

شبکه های کامپیوتری

نویسنده

اندرواس.تنن باوم

فهرست مطالب

۱	مقدمه	۱۱
۱-۱	کاربردهای شبکه های کامپیوتری	۱۲
۱-۱-۱	کاربردهای تجاری	۱۲
۱-۱-۲	کاربردهای خانگی	۱۵
۱-۱-۳	کاربران ستار	۱۸
۱-۱-۴	تبعات اجتماعی	۲۰
۲-۱	سخت افزار شبکه	۲۲
۱-۲-۱	شبکه های محلی	۲۳
۲-۲-۱	شبکه های شهری	۲۵
۳-۲-۱	شبکه های گسترده	۲۶
۴-۲-۱	شبکه های بیسیم	۲۷
۵-۲-۱	شبکه های خانگی	۲۹
۶-۲-۱	شبکه های	۳۱
۳-۱	نرم افزار شبکه	۳۲
۱-۳-۱	سلسله مراتب پروتکل ها	۳۲
۲-۳-۱	ملاحظات در طراحی لایه ها	۳۶
۳-۳-۱	سرویسهای اتصال-گرا و غیر متصل	۳۷
۴-۳-۱	عملکردهای پایه سرویس	۳۹
۵-۳-۱	رابطه سرویس و پروتکل	۴۱
۴-۱	مدلهای مرجع	۴۲
۱-۴-۱	مدل مرجع OSI	۴۲
۲-۴-۱	مدل مرجع TCP/IP	۴۵
۳-۴-۱	مقایسه مدل های OSI و TCP/IP	۴۷
۴-۴-۱	نگاهی انتقادی به مدل OSI و پروتکل های آن	۴۹
۵-۴-۱	نگاهی انتقادی به مدل TCP/IP	۵۱
۵-۱	شبکه های نمونه	۵۲
۱-۵-۱	اینترنت	۵۲
۲-۵-۱	شبکه های اتصال-گرا: X.25 ، Frame Relay	۵۲
۶-۱	ATM و	۶۰
۳-۵-۱	اینترنت	۶۵
۴-۵-۱	شبکه های محلی بیسیم: 802.11	۶۷
۶-۱	استانداردهای شبکه	۷۰
۱-۶-۱	مراجع مسئول استانداردهای مخابرات	۷۰
۲-۶-۱	مراجع مسئول استانداردهای بین المللی	۷۲
۳-۶-۱	مراجع مسئول استانداردهای اینترنت	۷۴
۷-۱	واحدهای اندازه گیری	۷۵
۸-۱	طرح کلی مباحث کتاب	۷۶
۹-۱	خلاصه	۷۸
۷۹	مسائل	
۲	لایه فیزیکی	۸۳
۱-۲	مبانی نظری مخابرات داده	۸۳
۱-۱-۲	آنالیز فوریه	۸۳
۲-۱-۲	محدودیت پهنای باند	۸۴
۳-۱-۲	حداکثر نرخ داده در یک کانال	۸۶
۲-۲	رسانه انتقال هدایت پذیر	۸۷
۱-۲-۲	رسانه مغناطیسی	۸۷
۲-۲-۲	زوج تابیده	۸۸
۳-۲-۲	کابل کواکسیال	۸۹
۴-۲-۲	فیبر نوری	۹۰
۳-۲	انتقال بیسیم	۹۶
۱-۳-۲	طیف الکترومغناطیس	۹۶
۲-۳-۲	مخابرات رادیویی	۹۹
۳-۳-۲	مخابرات مایکروویو	۱۰۰
۴-۳-۲	امواج مادون قرمز و میلیمتری	۱۰۲
۵-۳-۲	مخابرات امواج نوری	۱۰۳
۴-۲	ماهواره های مخابراتی	۱۰۳
۱-۴-۲	ماهواره های زمین ثابت	۱۰۵
۲-۴-۲	ماهواره های مدار متوسط	۱۰۸

فهرست مطالب ۷

۴-۳	پروتکل های پنجره لغزنده	۱۹۶
۱-۴-۳	پروتکل پنجره لغزنده ایست	۱۹۸
۲-۴-۳	پروتکل «N» تا به عقب برگرد»	۲۰۱
۳-۴-۳	پروتکل تکرار انتخابی	۲۰۸
۵-۳	ارزیابی پروتکل ها	۲۱۴
۱-۵-۳	مدل ماشین حالت محدود	۲۱۴
۲-۵-۳	مدل شبکه پتری	۲۱۷
۶-۳	چند نمونه از پروتکل های لینک داده	۲۱۹
۱-۶-۳	HDLC - کنترل سطح بالای لینک داده	۲۱۹
۲-۶-۳	لایه پیوند داده در اینترنت	۲۲۲
۷-۳	خلاصه	۲۲۷
	مسائل	۲۲۷

۴ زیر لایه دسترسی به لایه انتقال .. ۲۳۱

۱-۴	مسئله تخصیص کانال	۲۳۲
۲-۴	پروتکل های دسترسی چندگانه	۲۳۵
۱-۲-۴	ALOHA	۲۳۵
۲-۲-۴	پروتکل های دسترسی چندگانه با قابلیت شنود	
	سیگنال حامل (CSMA)	۲۳۹
۳-۲-۴	پروتکل های بدون تصادم	۲۴۲
۴-۲-۴	پروتکل های با رقابت محدود	۲۴۵
۵-۲-۴	پروتکل های دسترسی چندگانه مبتنی بر تقسیم	
	طول موج	۲۴۹
۶-۲-۴	پروتکل های بی سیم برای شبکه محلی	۲۵۲
۳-۲-۴	اترنت	۲۵۵
۱-۳-۴	کابل کشی اترنت	۲۵۶
۲-۳-۴	کدینگ منچستر	۲۵۹
۳-۳-۴	پروتکل زیر لایه MAC در اترنت	۲۶۰
۴-۳-۴	الگوریتم عقب گرد نمایی	۲۶۲
۵-۳-۴	کارا ئی (بازده) اترنت	۲۶۳
۶-۳-۴	اترنت مبتنی بر سوئیچ	۲۶۵
۷-۳-۴	اترنت سریع	۲۶۷
۸-۳-۴	اترنت گیگابیت	۲۷۱
۹-۳-۴	IEEE 802.2: کنترل منطقی لینک	۲۷۵
۱۰-۳-۴	نگاهی به گذشته اترنت	۲۷۶
۴-۴	شبکه های محلی بی سیم	۲۷۶
۱-۴-۴	پشته پروتکلی 802.11	۲۷۷

۳-۴-۲	ماهواره های مدار پائین	۱۰۸
۴-۴-۲	ماهواره یافبر	۱۱۱
۵-۲	شبکه تلفن عمومی	۱۱۲
۱-۵-۲	ساختار سیستم تلفن	۱۱۲
۲-۵-۲	تلفن و سیاست	۱۱۵
۳-۵-۲	مدار های پایانی: مودم، ADSL، و بیسیم	۱۱۷
۴-۵-۲	ترانک ها و مالتی پلکس کردن	۱۲۹
۵-۵-۲	سوئیچینگ	۱۳۷
۶-۲	شبکه تلفن همراه	۱۴۲
۱-۶-۲	تلفن های همراه نسل اول: صدای آنالوگ	۱۴۳
۲-۶-۲	تلفن های همراه نسل دوم: صدای	
	دیجیتال	۱۴۷
۳-۶-۲	تلفن های همراه نسل سوم: صدای دیجیتال و	
	داده	۱۵۵
۷-۲	تلویزیون کابلی	۱۵۸
۱-۷-۲	تلویزیون با آنتن مرکزی	۱۵۸
۲-۷-۲	اینترنت کابلی	۱۵۸
۳-۷-۲	تخصیص طیف فرکانسی	۱۶۰
۴-۷-۲	مودم های کابلی	۱۶۱
۵-۷-۲	مودم کابلی یا ADSL ؟	۱۶۳
۸-۲	خلاصه	۱۶۴
	مسائل	۱۶۵

۳ لایه پیوند داده .. ۱۷۱

۱-۳	ملاحظات طراحی لایه پیوند داده	۱۷۱
۱-۱-۳	سرویس هایی که به لایه شبکه داده	
	می شود	۱۷۲
۲-۱-۳	فریم بندی	۱۷۴
۳-۱-۳	کنترل خطا	۱۷۷
۴-۱-۳	کنترل جریان	۱۷۸
۴-۳	کشف و تصحیح خطا	۱۷۹
۱-۲-۳	گدهای تصحیح خطا	۱۷۹
۲-۲-۳	گدهای کشف خطا	۱۸۱
۳-۳	چند پروتکل ساده لینک داده	۱۸۵
۱-۳-۳	پروتکل یکطرفه نامقید	۱۸۹
۲-۳-۳	پروتکل توقف-انتظار یکطرفه	۱۹۱
۳-۳-۳	پروتکل یکطرفه برای کانال های نويز دار	۱۹۳

۲۷۸ ۲-۴-۴ لایه فیزیکی در 802.11	۳۳۴ دیتاگرام
۲۸۰ ۳-۴-۴ پروتکل زیر لایه MAC در 802.11	۳۳۶ ۲-۵ الگوریتمهای مسیریابی
۲۸۴ ۴-۴-۴ ساختار فریم 802.11	۳۳۸ ۱-۲-۵ اصل بهینگی
۲۸۶ ۵-۴-۴ خدمات	۳۳۹ ۲-۲-۵ مسیریابی مبتنی بر کوتاهترین مسیر
۲۸۷ ۵-۴-۴ بی سیم با باند گسترده	۳۴۱ ۳-۲-۵ الگوریتم سیل آسا (Flooding)
۲۸۸ ۶-۴-۴ مقایسه 802.11 با 802.16	۳۴۳ ۴-۲-۵ مسیریابی بردار فاصله
۲۸۹ ۶-۴-۴ پشته پروتکلی 802.16	۳۴۷ ۵-۲-۵ مسیریابی حالت لینک
۲۹۰ ۳-۵-۴ لایه فیزیکی در 802.16	۳۵۳ ۶-۲-۵ مسیریابی سلسله مراتبی
۲۹۲ ۴-۵-۴ پروتکل زیر لایه MAC در 802.16	۳۵۵ ۷-۲-۵ مسیریابی فراگیر (Broadcast Routing)
۲۹۴ ۵-۵-۴ ساختار فریم در 802.16	۳۵۷ ۸-۲-۵ مسیریابی چندپخشی
۲۹۵ ۶-۴-۴ بلوتوث (Bluetooth)	۳۵۹ ۹-۲-۵ مسیریابی برای ماشینهای متحرک
۲۹۶ ۶-۴-۴ معماری بلوتوث	۳۶۳ ۱۰-۲-۵ مسیریابی در شبکه های ویژه
۲۹۷ ۶-۴-۴ کاربردهای بلوتوث	۳۶۸ ۱۱-۲-۵ جستجوی گره در شبکه های همتابه
۲۹۸ ۳-۶-۴ پشته پروتکلی بلوتوث	۳۶۸ همتابه
۳۰۰ ۴-۶-۴ لایه رادیویی در بلوتوث	۳۷۳ ۳-۵ الگوریتمهای کنترل ازدحام
۳۰۰ ۵-۶-۴ لایه باند پایه در بلوتوث	۳۷۵ ۱-۳-۵ اصول کلی در کنترل جریان
۳۰۱ ۶-۶-۴ لایه L2CAP در بلوتوث	۳۷۷ ۲-۳-۵ سیاستهای پیشگیری از ازدحام
۳۰۱ ۷-۶-۴ ساختار فریم در بلوتوث	۳۷۷ ۳-۳-۵ کنترل ازدحام در زیر شبکه های مدار
۳۰۳ ۷-۴ هدایت در سطح لایه پیوند داده ها	۳۷۸ مجازی
۳۰۵ ۱-۷-۴ پلهانی از 802.x به 802.y	۳۸۰ ۴-۳-۵ کنترل ازدحام در زیر شبکه های دیتاگرام
۳۰۷ ۲-۷-۴ بهم بندی شبکه ها به صورت محلی	۳۸۳ ۵-۳-۵ دور ریختن بار (Load Shedding)
۳۰۹ ۳-۷-۴ پلهای مبتنی بر درخت پوشا	۳۸۵ ۶-۳-۵ کنترل لرزش (Jitter Control)
۳۱۰ ۴-۷-۴ پلهای راه دور (Remote Bridges)	۳۸۶ ۴-۵ کیفیت خدمات (Quality of Service)
۳۱۰ ۵-۷-۴ تکرار کننده، هاب، پل، سوئیچ، مسیریاب و دروازه	۳۸۶ ۱-۴-۵ نیازها
۳۱۱ ۶-۷-۴ شبکه های محلی مجازی (Virtual LANs)	۳۸۶ ۲-۴-۵ راهکارهای دستیابی به کیفیت خوب
۳۱۴ ۸-۴ خلاصه	۳۸۷ خدمات
۳۲۲ مسائل	۴۰۰ ۳-۴-۵ خدمات مجتمع (Integrated Services)
۳۲۴ لایه شبکه	۴۰۲ ۴-۴-۵ خدمات متمایز
۳۲۹ ۱-۵ مسائل طراحی لایه شبکه	۴۰۵ ۵-۴-۵ سوئیچ برچسب و MPLS
۳۲۹ ۱-۱-۵ هدایت (سوئیچینگ) بسته به روش ذخیره و هدایت	۴۰۹ ۵-۵ بهم بندی شبکه ها (Internetworking)
۳۲۹ ۲-۱-۵ خدمات ارائه شده برای لایه انتقال	۴۱۰ ۱-۵-۵ شبکه ها از چه دیدگاهی متفاوتند؟
۳۳۱ ۳-۱-۵ پیاده سازی خدمات بی اتصال	۴۱۱ ۲-۵-۵ چگونگی اتصال شبکه ها به یکدیگر
۳۳۳ ۴-۱-۵ پیاده سازی خدمات اتصال گرا	۴۱۳ ۳-۵-۵ مدارات مجازی الحاق شده
۵-۱-۵ ۵-۱-۵ مقایسه زیر شبکه های مدار مجازی و	۴۱۴ ۴-۵-۵ بهم بندی شبکه های بدون اتصال
	۴۱۶ ۵-۵-۵ ایجاد تونل (Tunneling)
	۴۱۷ ۶-۵-۵ مسیریابی بین شبکه های بهم متصل
	۴۱۸ ۷-۵-۵ قطعه قطعه سازی بسته ها

۵۲۶	۳-۴-۶ پروتکل انتقال بی درنگ	۴۲۲	۶-۵ لایه شبکه در اینترنت
۵۳۰	۵-۶ پروتکل های لایه انتقال در اینترنت: TCP	۴۲۴	۱-۶ پروتکل IP
۵۳۰	۱-۵-۶ مقدمه ای بر TCP	۴۲۸	۲-۶-۵ آدرسهای IP
۵۳۱	۲-۵-۶ مدل خدمات TCP	۴۴۱	۳-۶-۵ پروتکل های کنترل اینترنت
۵۳۳	۳-۵-۶ پروتکل TCP	۴۶۵	۴-۶-۵ OSPF: پروتکل مسیریابی برای دروازه های
۵۳۴	۴-۵-۶ سرآیند قطعه TCP	۴۴۷	درونی
۵۳۸	۵-۵-۶ برقراری اتصال TCP	۵۶۵	BGP: پروتکل مسیریابی برای دروازه
۵۳۹	۶-۵-۶ خاتمه دادن به اتصال TCP	۴۵۳	خارجی
۵۴۰	۷-۵-۶ مدلسازی فرآیند مدیریت اتصال در TCP	۴۵۵	۶-۶-۵ ارسال چندپخش در اینترنت
۵۴۲	۸-۵-۶ سیاستهای انتقال در TCP	۴۵۶	۷-۶-۵ IP متحرک (Mobile IP)
۵۴۶	۹-۵-۶ کنترل ازدحام در TCP	۴۵۸	۸-۶-۵ IPv6
۵۴۹	۱۰-۵-۶ مدیریت تایمرها در TCP	۴۶۸	۷-۵ خلاصه
۵۵۲	۱۱-۵-۶ TCP و UDP بی سیم	۴۶۸	مسائل
۵۵۴	۱۲-۵-۶ TCP تراکنشی (Transactional TCP)		
۵۵۶	۶-۶ مسائل مرتبط با کارآیی	۴۷۵	۶ لایه انتقال
۵۵۷	۶-۶ مشکلات کارآیی در شبکه های کامپیوتری	۴۷۵	۱-۶ خدمات انتقال (The Transport Service)
۵۵۹	۲-۶ اندازه گیری کارآیی شبکه	۴۷۵	۱-۶ خدمات ارائه شده به لایه های بالاتر
۵۶۱	۳-۶ طراحی سیستم برای کارآیی بهتر	۴۷۵	۲-۶ عملکردهای اولیه و توابع بنیانی لایه
۵۶۵	۴-۶ پردازش سریع TPDU	۴۷۷	انتقال
۵۶۹	۵-۶ پروتکل های برای شبکه های گیگابیتی	۴۸۱	۳-۶ سوکت های برکلی (Berkeley Socket)
۵۷۳	۷-۶ خلاصه	۴۸۱	۴-۶ مثالی از برنامه نویسی سوکت: یک
۵۷۳	مسائل	۴۸۲	سرویس دهنده اینترنتی فایل
		۴۸۷	۲-۶ مؤلفه های هر پروتکل انتقال
۵۷۹	۷ لایه کاربرد	۴۸۸	۱-۲-۶ آدرس دهی
۵۷۹	۱-۷ سیستم نام ناحیه DNS	۴۹۱	۲-۲-۶ برقراری اتصال
۵۸۰	۱-۱-۷ فضای نام DNS	۴۹۷	۳-۲-۶ خاتمه اتصال
۵۸۲	۲-۱-۷ رکوردهای منابع	۵۰۱	۴-۲-۶ کنترل جریان و بافرسازی
۵۸۵	۳-۱-۷ سرویس دهنده نام	۵۰۶	۵-۲-۶ مالتی پلکسینگ (تسهیم)
۵۸۷	۲-۷ پست الکترونیک	۵۰۷	۶-۲-۶ جبران از کارافتادگی (Crash Recovery)
۵۸۸	۱-۲-۷ معماری و سرویسها	۵۱۰	۳-۶ یک پروتکل ساده انتقال
۵۸۹	۲-۲-۷ عامل کاربر	۵۱۰	۱-۳-۶ توابع اولیه ارائه خدمات در مثال فوق
۵۹۲	۳-۲-۷ فرمت پیامها	۵۱۲	۲-۳-۶ واحد انتقال در مثال فوق
۵۹۸	۴-۲-۷ انتقال پیام	۵۳۳	۳-۳-۶ بررسی مثال فوق از دید «ماشین حالت
۶۰۱	۵-۲-۷ تحویل نهایی	۵۱۹	محدوده
۶۰۶	۳-۷ تارنمای جهانی - وب	۵۲۱	۴-۶ پروتکل های لایه انتقال در اینترنت: UDP
۶۰۷	۱-۳-۷ بررسی ساختاری	۵۲۲	۱-۴-۶ مقدمه ای بر UDP
۶۲۲	۲-۳-۷ سندهای وب استاتیک	۵۲۳	۲-۴-۶ فراخوانی پروسیجرهای راه دور (RPC)

۷۴۷	۴-۴-۸ حمله روز تولد (The birthday Attack)	۶۳۴	۳-۳-۷ سندهای وب دینامیک
۷۵۰	۵-۸ مدیریت کلیدهای عمومی	۶۴۱	۴-۳-۷ پروتکل انتقال اَیر متن - HTTP
۷۵۰	۱-۵-۸ گواهینامه‌ها (Certificates)	۶۴۵	۵-۳-۷ بهبود کارایی
۷۵۲	۲-۵-۸ X.509	۶۵۱	۶-۳-۷ وب پیسیم
۷۵۳	۳-۵-۸ زیرساخت کلید عمومی	۶۶۱	۴-۷ چند رسانه‌ای
۷۵۷	۶-۸ امنیت ارتباطات	۶۶۲	۱-۴-۷ مقدمه‌ای بر صدای دیجیتال
۷۵۷	۱-۶-۸ IPsec	۶۶۳	۲-۴-۷ فشرده‌سازی صدا
۷۶۲	۲-۶-۸ دیوارهای آتش (Firewalls)	۶۶۵	۳-۴-۷ صدای جویباری
۷۶۵	۳-۶-۸ شبکه‌های خصوصی مجازی (VPN)	۶۶۹	۴-۴-۷ رادیوی اینترنتی
۷۶۶	۴-۶-۸ امنیت شبکه‌های بی‌سیم	۶۷۱	۵-۴-۷ صدای روی IP
۷۷۱	۷-۸ پروتکل‌های احراز هویت	۶۷۷	۶-۴-۷ مقدمه‌ای بر ویدئو
۷۷۸	۱-۷-۸ احراز هویت براساس کلید مشترک و	۶۸۱	۷-۴-۷ فشرده‌سازی ویدئو
۷۷۲	سری	۶۸۷	۸-۴-۷ پخش فیلم بر حسب تقاضا
۷۷۷	۲-۷-۸ ایجاد کلید مشترک: مبادله کلید به روش	۶۹۳	۹-۴-۷ ستون فقرات چندپخش - Mbone
۷۷۷	«دیفی-هلمن»	۶۹۷	۵-۷ خلاصه
۷۷۹	۳-۷-۸ احراز هویت توسط مرکز توزیع کلید	۶۹۷	مسائل
۷۸۲	۴-۷-۸ احراز هویت با استفاده از Kerberos		
۷۸۲	۵-۷-۸ احراز هویت با استفاده از رمزنگاری با کلید		
۷۸۵	عمومی		
۷۸۶	۸-۸ امنیت نامه‌های الکترونیکی		
۷۸۶	۱-۸-۸ PGP (Pretty Good Privacy)		
۷۹۱	۲-۸-۸ PEM (Privacy Enhanced Mail)		
۷۹۱	۳-۸-۸ S/MIME		
۷۹۲	۹-۸ امنیت وب		
۷۹۲	۱-۹-۸ تهدیدها		
۷۹۳	۲-۹-۸ نامگذاری مطمئن		
۸۰۱	۳-۹-۸ SSL: لایه سوکنهای امن		
۸۰۵	۴-۹-۸ امنیت کدهای متحرک		
۸۰۸	۱۰-۸ زمینه‌ها و پی‌آمدهای اجتماعی		
۸۰۸	۱-۱۰-۸ حریم خصوصی افراد (Privacy)		
۸۱۱	۲-۱۰-۸ آزادی بیان		
۸۱۳	۳-۱۰-۸ مالکیت معنوی (Copyright)		
۸۱۶	۱۱-۸ خلاصه		
۸۱۷	مسائل		
۸۲۳	واژه‌نامه		
۸۲۹	محتویات دیسک فشرده همراه کتاب		

مقدمه

هر یک از سه قرن گذشته را با یک تکنولوژی خاص بعنوان نماد آن قرن می‌شناسیم. قرن هیجدهم عصر سیستمهای بزرگ مکانیکی و انقلاب صنعتی بود، و قرن نوزدهم عصر بخار. تکنولوژی کلیدی قرن بیستم نیز جمع‌آوری، پردازش و توزیع اطلاعات بود. شبکه‌های گسترده و بین‌المللی تلفن، اختراع رادیو و تلویزیون، تولد و گسترش باورنکردنی صنعت کامپیوتر، و پرتاب ماهواره‌های مخابراتی از نمادهای این عصر هستند.

با رشد سریع تکنولوژیهای جمع‌آوری، پردازش و توزیع اطلاعات، این زمینه‌ها بسرعت در هم ادغام شده، و تفاوت‌های آنها در حال محو شدن است. شرکتی که در اقصی نقاط دنیا شعبه و نمایندگی دارند، می‌توانند فقط با فشار یک دکمه از آخرین وضعیت دفاتر خود (حتی دور افتاده‌ترین آنها) مطلع شوند. اما جالب اینجاست که رشد تقاضا برای روشهای پیشرفته‌تر پردازش اطلاعات همیشه یک گام از سرعت رشد این تکنولوژیها جلوتر است. با اینکه صنعت کامپیوتر از صنایع دیگر (از جمله صنایع اتومبیل، و حمل و نقل هوایی) نسبتاً جوانتر است، اما در مدتی بس کوتاه به پیشرفتهای چشمگیری دست یافته است. در دو دهه اول، سیستمهای کامپیوتری بسیار متمرکز بودند، و معمولاً در یک اتاق بزرگ جا می‌گرفتند. کم نبودند مراکزی که این اتاقها دیوارهای شیشه‌ای داشتند، و بازدیدکنندگان با حیرت این موجودات عجیب‌الخلقه الکترونیکی را برانداز می‌کردند. دانشگاهها و شرکت‌های متوسط معمولاً یکی دو کامپیوتر بیشتر نداشتند، و تعداد شرکت‌هایی که استطاعت خرید بیش از یک دوجین از آنها را داشته باشند، چندان زیاد نبود. هیچکس (شاید غیر از نویسندگان داستانهای علمی-تخیلی) حتی نمی‌توانست تصور کند که تا قبل از پایان قرن بیستم کامپیوترهایی با همان قدرت را بتوان روی یک تمبر پستی جای داد، و آنها را بصورت انبوه تولید کرد.

پیوند فرخنده کامپیوتر و مخابرات اتفاقی بود که هر دو صنعت را دچار تحولات عظیم کرد. اکنون دیگر مفهوم اتاقی با یک کامپیوتر بزرگ بنام «مرکز کامپیوتر»، که افراد کارهایشان را به آنجا می‌آورند، بکلی منسوخ شده است. مدل قدیمی کامپیوتر بزرگی که تمام کارهای محاسباتی سازمان را انجام می‌دهد، اکنون جای خود را به تعداد زیادی کامپیوتر کوچک متصل به هم داده است. به این سیستمها شبکه‌های کامپیوتری (computer networks) گفته می‌شود؛ موضوع این کتاب نیز طراحی و ساختار این شبکه‌هاست.

در این کتاب هر جا از «شبکه کامپیوتری» سخن می‌گوئیم، منظورمان مجموعه‌ای از کامپیوترهای مستقل است، که با یک تکنولوژی واحد به هم متصل شده‌اند. دو کامپیوتر وقتی «به هم متصلند»، که بتوانند با یکدیگر اطلاعات رد و بدل کنند. الزامی نیست که این اتصال از طریق سیمهای مسی باشد؛ فیبرهای نوری، امواج مایکروویو و مادون قرمز، و ماهواره‌های مخابراتی هم می‌توانند عامل این ارتباط باشند. بعداً خواهیم دید که اندازه، شکل و ساختار

شبکه ها می توانند بسیار متفاوت باشد. همچنین بسیاری از افراد وقتی می شنوند که اینترنت یا وب هیچکدام شبکه کامپیوتری نیستند متعجب می شوند؛ اما در پایان این کتاب علت آنرا هم خواهید فهمید. فعلاً همین قدر کافیه بدانید که: اینترنت یک شبکه نیست، بلکه شبکه ایست از شبکه ها، و وب نیز یک سیستم توزیع شده است که بر پایه اینترنت کار می کند.

لازم است همین جا به یک اشتباه رایج بین دو اصطلاح شبکه کامپیوتری و سیستم توزیع شده (distributed system) اشاره کنم. یک سیستم توزیع شده مجموعه ایست از چندین کامپیوتر مستقل، که کاربر آنرا به شکل یک سیستم واحد و متجانس می بیند. در این سیستمها معمولاً یک لایه نرم افزاری (روی سیستم عامل) بنام میان افزار (middleware) است، که مدل موردنظر را پیاده سازی می کند. وب (World Wide Web) نمونه ای از یک سیستم توزیع شده است، که در آن همه چیز از دیدگاه کاربر یک سند (صفحه وب) بنظر می رسد. در شبکه کامپیوتری این تجانس، مدل و نرم افزار وجود ندارد. کاربران بطور مستقیم با کامپیوترها در تماسند، و هیچ کوششی برای ایجاد تجانس بین آنها صورت نمی گیرد. کاربر بروشنی تفاوت های نرم افزاری و سخت افزاری کامپیوترها را می بیند، و اگر بخواهد برنامه ای را روی یکی از کامپیوترها اجرا کند، باید ابتدا وارد آن شود (log on). در حقیقت، یک سیستم توزیع شده نرم افزاریست که روی شبکه کار می کند، و تجانس و شفافیت آن توسط این نرم افزار تأمین می شود. به همین دلیل تفاوت سیستم توزیع شده با یک شبکه بیشتر در نرم افزار (بویژه سیستم عامل) نهفته است تا سخت افزار.

با این همه، شباهت های زیادی نیز بین این دو وجود دارد. مثلاً، سیستم های توزیع شده و شبکه ها هر دو به انتقال فایل نیاز دارند؛ تفاوت در اینست که این کار را چه کسی انجام می دهد، سیستم یا کاربر. با آنکه این کتاب درباره شبکه های کامپیوتری است، بسیاری از مطالب آن در سیستم های توزیع شده نیز مصداق دارد. برای کسب اطلاعات بیشتر درباره سیستم های توزیع شده به (Tanenbaum and Van Steen, 2002) نگاه کنید.

۱-۱ کاربردهای شبکه های کامپیوتری

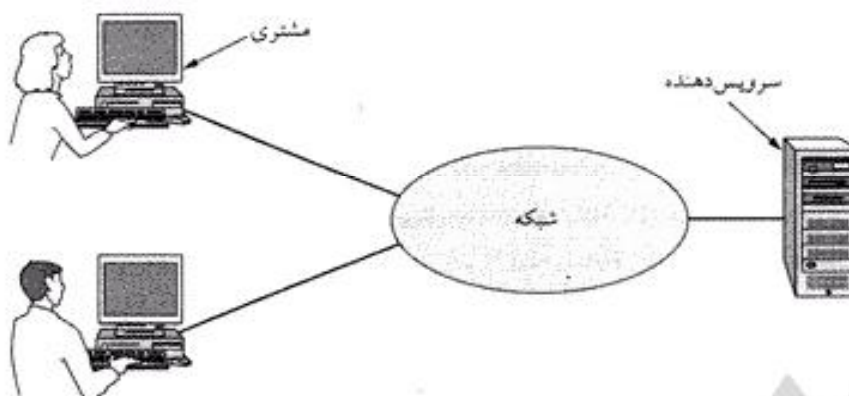
قبل از پرداختن به جزئیات فنی، بهتر است کمی درباره اینکه چرا مردم به شبکه های کامپیوتری اهمیت می دهند و چرا از آنها استفاده می کنند، صحبت کنیم (چرا که اگر کسی به شبکه اهمیت نمی داد، اصلاً شبکه ای ساخته نمی شد). ابتدا از کاربردهای سنتی (از قبیل شرکتها و افراد) شروع می کنیم، و سپس به کاربردهای جدیدتر (مانند شبکه های متحرک و خانگی) خواهیم پرداخت.

۱-۱-۱ کاربردهای تجاری

اکثر شرکتها تعداد زیادی کامپیوتر برای کارهای مختلف (تولید، انبارداری، فروش، و حسابداری) دارند. شاید در ابتدا این کامپیوترها از یکدیگر جدا باشند، ولی در مرحله ای از کار برای یکپارچه کردن اطلاعات کل شرکت، مدیریت تصمیم می گیرد تا آنها را به هم متصل کند.

به بیان کلی تر، اشتراک منابع (resource sharing) به ما اجازه می دهد تا برنامه ها، تجهیزات و بخصوص داده ها را (صرفنظر از موقعیت فیزیکی افراد و منابع) در اختیار همه آنها بی که به این شبکه متصلند، قرار دهیم. ساده ترین مثال آن، چاپگر است که برای تمام کارکنان یک دفتر به اشتراک گذاشته شده است. پیداست که تک تک این افراد به یک چاپگر اختصاصی نیاز ندارند، و علاوه بر آن یک چاپگر شبکه اغلب ارزانتر، سریعتر و کم هزینه تر از تعداد زیادی چاپگرهای پراکنده است.

با این حال، اشتراک اطلاعات بسیار مهمتر از اشتراک تجهیزات فیزیکی (مانند چاپگر، اسکنر، و CD نویس) است. امروزه تمام شرکت های بزرگ و متوسط (و بسیاری از شرکت های کوچک) پشددت به اطلاعات کامپیوتری خود



شکل ۱-۱. شبکه‌ای با دو مشتری و یک سرویس‌دهنده.

(از قبیل اطلاعات مشتریان، انبار، سند های مالی و حساب داری، و اطلاعات مالیاتی) وابسته‌اند. بانکی که تمام کامپیوترهای آن از کار افتاده باشند، پنج دقیقه هم نمی‌تواند دوام بیاورد. حتی شرکت های کوچکی مانند آژانس های مسافرتی و دفاتر خدمات حقوقی نیز بشدت به اطلاعات کامپیوتری خود متکی هستند.

در یک شرکت کوچک تمام کامپیوترها به احتمال زیاد در یک دفتر (و یا حداکثر یک ساختمان) قرار دارند، در حالیکه کامپیوترهای یک شرکت بزرگ می‌تواند در یک شهر یا کشور (و حتی در قاره های مختلف) پراکنده باشد. در این حالت، ممکنست مدیر فروشی که در نیویورک نشسته، به موجودی انبار شرکت در سنگاپور نیاز داشته باشد. بعبارت دیگر، حتی ۱۵۰۰۰ کیلومتر فاصله هم نباید خللی در دسترسی به اطلاعات وارد کند. در واقع می‌توان گفت، ما بدنبال «از بین بردن فاصله ها» هستیم.

در ساده ترین شکل، اطلاعات شرکت می‌تواند در یک یا چند پایگاه داده متمرکز باشد، و کارمندان شرکت بایستی بتوانند از راه دور به آنها دسترسی داشته باشند. در این مدل، اطلاعات در کامپیوترهای پُر قدرتی بنام سرویس دهنده (server) - که اغلب در یک مرکز و تحت کنترل سرپرست سیستم قرار دارند - نگهداری می‌شوند. کارمندان نیز، که در اینجا به آنها مشتری (client) گفته می‌شود، از راه دور و از پای کامپیوترهای معمولی خود به این اطلاعات دسترسی پیدا می‌کنند. (گاهی به فردی که از کامپیوتر استفاده می‌کند، نیز «مشتری» گفته می‌شود؛ بهر حال، از فحوای متن باید بتوانید متوجه شوید که منظور کامپیوتر است یا کاربر.) اتصال کامپیوترهای سرویس دهنده و مشتری از طریق شبکه صورت می‌گیرد (شکل ۱-۱ را ببینید). در این شکل شبکه به صورت یک بیضی ساده نشان داده شده است؛ وقتی بخواهیم شبکه را بصورت کلی و انتزاعی (و بدون هیچگونه جزئیاتی) نشان دهیم، از این روش استفاده خواهیم کرد.

به این آرایش مدل مشتری-سرویس دهنده (client-server model) گفته می‌شود، و در بسیاری از شبکه های کوچک و بزرگ کاربرد دارد چون مستقل از فاصله است. وب نیز بر مبنای مدل مشتری-سرویس دهنده ساخته شده است؛ وقتی یک صفحه وب را باز می‌کنید، در واقع آنرا از سرویس دهنده وب دریافت کرده، و در کامپیوتر خود (که در اینجا مشتری است) نمایش می‌دهید. در اکثر مواقع یک سرویس دهنده می‌تواند به تعداد زیادی مشتری سرویس بدهد.

اکثر مدل مشتری-سرویس دهنده را دقیقتر بررسی کنیم، متوجه می‌شویم که دو پروسس (process) در آن دخیل هستند: یک پروسس روی کامپیوتر مشتری، و دیگری روی کامپیوتر سرویس دهنده. ارتباط از لحظه ای آغاز می‌شود، که پروسس مشتری از طریق شبکه یک پیام به پروسس سرویس دهنده فرستاده، و سپس به انتظار پاسخ

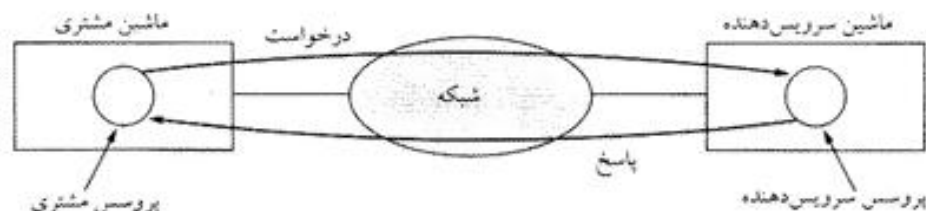
آن می ماند. وقتی پروسس سرویس دهنده درخواست مشتری را دریافت کرد، کار خواسته شده را انجام می دهد (یا اطلاعات خواسته شده را تهیه می کند)، و پاسخ را به مشتری پس می فرستد. این فرآیند را در شکل ۱-۲ ملاحظه می کنید.

گاهی در یک شبکه کامپیوتری رابطه بین افراد اهمیت بیشتری نسبت به تبادل اطلاعات بین کامپیوترها دارد. چنین شبکه ای در واقع یک رسانه ارتباطی (communication medium) است. امروزه دیگر تقریباً هیچ شرکتی را نمی توان یافت که از سرویس پست الکترونیک (ایمیل: e-mail) استفاده نکند، و در واقع بسیاری از ارتباطات روزمره کارمندان از همین طریق صورت می گیرد. این روش آنقدر ساده و کارآمد است که خود باعث بروز مشکلات جدیدی شده است، چون رؤسای شرکتها هم یاد گرفته اند چطور فقط با فشار یک دکمه می توانند پیامهای (اغلب بی محتوای) خود را به این طرف و آن طرف بفرستند!

اما ایمیل تنها شکل از ارتباطات پیشرفته ای نیست که به لطف شبکه های کامپیوتری ممکن شده است. در یک شبکه، دو نفر که فاصله زیادی هم از یکدیگر دارند، می توانند بطور مشترک روی یک گزارش یا مقاله کار کنند. وقتی یکی از آنها تغییری در این گزارش می دهد، دیگری بلافاصله آنرا خواهد دید (و دیگر نیازی نیست روزها به انتظار بستگی چشم به در بدوزد). با این روش دیگر نیازی نیست غصه هماهنگ کردن کارمندانی که هر کدام ساز خود را می زنند، بخورید.

یکی دیگر از امکانات ارتباطی شبکه ها، کنفرانس ویدئویی (video conferencing) است. به کمک این تکنولوژی، کارمندانی که هزاران کیلومتر از هم فاصله دارند، می توانند یکدیگر را ببینند، صدای هم را بشنوند، و با حتی مطالب خود را روی یک تخته سیاه مجازی بنویسند. کنفرانس ویدئویی جانشین بسیار مناسبی برای کنفرانسهای واقعی (که متضمن تحمل هزینه های سفر است) می باشد. گاهی گفته می شود که صنعت ارتباطات و حمل و نقل با یکدیگر مسابقه مرگ وزندگی گذاشته اند، و هر کدام پیروز شود، دیگری را از میدان بدر خواهد کرد. اتفاق دیگری که این روزها شتاب بیشتری گرفته، امکان تجارت الکترونیک بین شرکتهای کوچک و بزرگ است. برای مثال، سازندگان کامپیوتر، اتومبیل و هواپیما می توانند قطعات مورد نیاز خود را از طریق شبکه های کامپیوتری به سازندگان این نوع قطعات سفارش دهند، و سپس آنها را مونتاژ و تبدیل به محصول نهایی کنند. سفارش و خرید قطعات در لحظه نیاز (زمان واقعی) لزوم نگهداری و انبار کردن مقدار زیادی از آنها را مستفی می کند.

گرایش تجاری دیگری که حتی اهمیت بیشتری پیدا کرده، فروش محصولات روی اینترنت است. این روزها شرکتهای بسیاری (از قبیل خطوط هوایی، کتابفروشیها، و فروشندگان محصولات فرهنگی) به فروش محصولات خود از طریق اینترنت روی آورده اند. این شاخه از تجارت (که به تجارت الکترونیک - electronic commerce یا e-commerce - معروف است) در آینده رشد بسیار بیشتری خواهد کرد.



شکل ۱-۲. مدل مشتری-سرویس دهنده بر «درخواست و پاسخ» مبتنی است.

۲-۱-۱ کاربردهای خانگی

سال ۱۹۷۷، وقتی از کین اولین (رئیس شرکت Digital Equipment Corporation - که پس از IBM بزرگترین شرکت کامپیوتری دنیا محسوب می شد) پرسیدند چرا وارد بازار کامپیوترهای شخصی نمی شود، وی پاسخ داد: "هیچ دلیلی ندارد که هر کس توی خانه اش یک کامپیوتر داشته باشد." تاریخ ثابت کرد که اولین اشتباه می کرد، و اکنون دیگر شرکت DEC وجود خارجی ندارد. اما چرا مردم برای کارهای خانگی خود کامپیوتر می خرند؟ نوشتن نامه، مقاله و حتی کتاب (و تا یادم نرفته، بازی) یکی از مهمترین دلایل آن است؛ اما این وضعیت امروزه در حال تغییر است. شاید مهمترین دلیل خرید کامپیوترهای خانگی در سالهای اخیر اینترنت باشد. کارهای که این قبیل افراد با کامپیوتر خود انجام می دهند، عمدتاً عبارتند از:

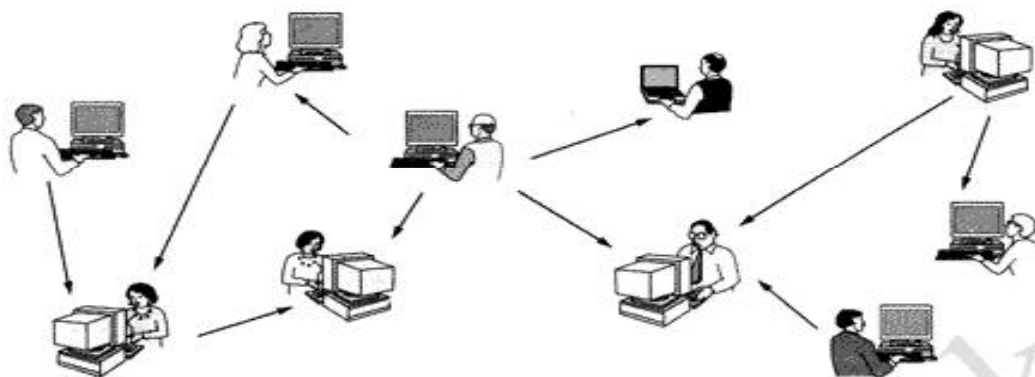
۱. دسترسی به اطلاعات پراکنده در سراسر دنیا
۲. ارتباطات دو جانبه
۳. سرگرمیهای تعاملی
۴. تجارت الکترونیک

امروزه منبع بسیار عظیمی از اطلاعات در تمامی زمینه ها (از قبیل هنر، تجارت، آشپزی، بهداشت، تاریخ، سرگرمی، علم، ورزش و تفریحات سالم - و البته گاهی ناسالم!) روی اینترنت وجود دارد، که می توان به آنها دسترسی پیدا کرد. روزنامه های بسیاری روی اینترنت منتشر می شوند، که می توان اخبار را بدلتخواه و بصورت گزینشی از آنها بدست آورد. حتی می توانید کاری کنید که مقاله دلخواه شما وقتی خواب هستید، از اینترنت بار شده و سپس چاپ شود، تا موقع صبحانه بتوانید با خیال راحت آنرا بخوانید. (به این ترتیب روزنامه فروشهای بیچاره بیکار خواهند شد، ولی مطبوعات هم هیچ وقت دل خوشی از آنها نداشتند.)

بعد از روزنامه ها و مجلات الکترونیکی نوبت کتابخانه های دیجیتال است. بسیاری از سازمانهای علمی معتبر مانند ACM (www.acm.org) و IEEE (www.computer.org) مدتهاست که انتشارات و کنفرانسهای متعددی روی اینترنت برگزار می کنند؛ و این روند سرعت در حال گسترش است. بنظر می رسد که رواج کتابخوانی اینترنتی فقط به قیمت، اندازه و وزن کامپیوترهای کتابی بستگی دارد. (شاید هنوز عده ای به این آینده شک داشته باشند؛ اما بهتر است آنها بلایی را که دستگاه چاپ گوتنبرگ سر کتابهای زیبای خطی آورد، بیاد بیاورند.)

تمام کاربردهایی که در بالا نام بردیم، متضمن ارتباط فرد با یک منبع اطلاعات روی اینترنت بود. اما روش دیگری نیز برای برقراری ارتباط از طریق اینترنت وجود دارد، و آن ارتباط فرد به فرد است (این پاسخ تکنولوژی قرن بیست و یکم است به تلفن قرن نوزدهمی). امروزه میلیونها نفر در سراسر دنیا بطور روزمره از ایمیل استفاده می کنند؛ صوت و تصویر هم مدتهاست به جزیی جدایی ناپذیر از آن تبدیل شده است (و باید بزودی منتظر بوهای اینترنتی هم باشیم!).

این روزها همه نوجوانانی که سری میان سرها در می آورند، به برنامه های پیام رسان قوری (instant messaging) معتاد هستند. این برنامه ها (که از برنامه talk در سیستمهای یونیکس مشتق شده اند) به افراد امکان می دهند پیامهای متنی خود را بلافاصله (و بدون تأخیر زمانی) با هم مبادله کنند. نسخه هایی از این برنامه ها که به چندین نفر اجازه می دهند تا در آن واحد با هم گفتگو کنند، به اتاق گفتگو (chat room) معروفند. گروههای خبری (newgroup) از امکانات قدیمی و پرمطرفدار اینترنت است، که امکان بحث درباره موضوعات بسیار متنوعی را به شما می دهد. در این سیستم پیامی که می فرستید، به تمام آنهایی که عضو گروه خبری هستند خواهد رسید (خوششان بیاید، یا نیاید). مبادلات گروه خبری (بر خلاف اتاق گفتگو) بصورت بلافاصله و در زمان واقعی نیست، و پیامها در نقطه ای ذخیره می شوند، تا کاربر بتواند هر زمان که خواست آنها را بخواند.



شکل ۳-۱. در یک سیستم همتا-همتا مشتری یا سرویس دهنده ثابتی وجود ندارد.

نوع دیگری از امکانات ارتباطی موجود در اینترنت، ارتباط همتا-به-همتا (peer-to-peer) است. این مدل تفاوت اساسی با مدل مشتری-سرویس دهنده دارد (به Parameswaran et al., 2001 نگاه کنید). در این مدل ارتباط افراد در یک گروه غیرثابت و ناپایدار صورت می گیرد (شکل ۳-۱ را ببینید). در واقع، هر فرد می تواند مستقیماً با هر فرد (یا افراد) دیگر تماس برقرار کند، و چیز ثابتی بعنوان سرویس دهنده یا مشتری وجود ندارد. بزرگترین نمونه ارتباط همتا-به-همتا در حوالی سال ۲۰۰۰ با سرویسی بنام Napster شکل گرفت؛ این سرویس در اوج خود امکان ارتباط بیش از ۵۰ میلیون نفر را فراهم می آورد، که بصورت غیر قانونی به رد و بدل کردن موزیک مبادرت می کردند (این بزرگترین نقض حق التألیف در تاریخ موسیقی بود؛ به Lam and Tan, 2001 و Macedonia, 2000 نگاه کنید). ایده کار نسبتاً ساده بود: هر نفر می توانست آهنگهایی را که در هارد دیسک خود داشت، در پایگاه داده مرکزی Napster ثبت کند؛ افرادی هم که بدنبال آهنگ خاصی بودند، این پایگاه داده را جستجو کرده، و بعد مستقیماً به سراغ آن می رفتند. Napster ادعا می کرد که هیچ حق التألیفی را نقض نمی کند، چون اساساً آهنگها در کامپیوترهای آن ذخیره نمی شوند؛ اما دادگاه با این نظر موافق نبود، و حکم به تعطیلی آن داد. سیستمهای همتا-به-همتا جدید با هوشتر شده اند، چون پایگاه داده مرکزی را حذف کرده اند و بجای آن این اطلاعات در کامپیوتر تک تک افراد ذخیره می شود، و آنها لیستی از افراد مجاور خود را هم در اختیار دارند. در این روش جستجو کمی بیشتر طول می کشد (که بار آن هم بر دوش کامپیوترهاست)، ولی در نهایت به همان اندازه مؤثر است.

همه برنامه های همتا-به-همتا هم غیرقانونی نیستند. برای مثال، برنامه هایی هستند که اجازه می دهند تا آهنگها و فیلمهای مجاز و حتی عکسهای خانوادگی خود را روی اینترنت به اشتراک بگذارید، و یا بازیهای دسته جمعی انجام دهید. در حقیقت، پرفرودارترین کاربرد اینترنت، یعنی ایمیل، ذاتاً یک سیستم همتا-به-همتا است، و بنظر می رسد در آینده این سیستمها حتی گسترده تر شوند.

جرایم الکترونیکی به دزدی آثار و نقض حق التألیف محدود نمی شود، و این روزها قمارخانه ها هم پایشان به اینترنت باز شده است. کامپیوترها قادرند هر کاری انجام دهند، پس چرا قمار نکنند؟ البته قمار در بسیاری از کشورها غیرقانونی است، اما مشکل اینجا است که در چند جا هم قانونیست (مانند انگلستان)، و صاحبان کازینوها به امکانات بالقوه اینترنت برای قمار واقف شده اند. اما اگر کازینو و قمارباز در دو کشور متفاوت (که قوانین متفاوتی هم در زمینه قمار دارند) باشند، چطور؟ سؤال خوبیست!

ارتباطات اینترنتی در زمینه تماسهای تلفنی، ویدئویی و رادیو نیز تحولات وسیعی ایجاد کرده اند. آموزش از

راه دور (telelearning) نیز یکی دیگر از امکاناتیست که اینترنت عرضه کرده است. (تصورش را بکنید که ساعت ۸ صبح سر کلاس درس حاضر باشید، بدون اینکه لازم باشد قبل از آن از رختخواب بیرون بیایید!) بنظر می رسد در دراز مدت اینترنت بزرگترین نقش را در بهبود ارتباطات انسانی بازی کند.

سومین دسته از کاربردهای خانگی شبکه های کامپیوتری، صنعت سرگرمی و تفریحات (با رشدی سرسام آور) است. داغترین بحث در این زمینه پخش فیلم برحسب تقاضا (video on demand) است. شاید تا ده سال دیگر راحتی بتوانید فیلم دلخواه خود را انتخاب کرده، و همان لحظه روی صفحه تلویزیون تماشا کنید. فیلمهای جدید تعاملی (interactive) خواهند بود، بدین معنا که بیننده می تواند مسیر سناریو را بدلتخواه خود تغییر دهد. تلویزیون زنده (شرکت مستقیم و بلافاصله در مسابقات و شوهای تلویزیونی) نیز یکی دیگر از امکانات آینده است.

بازیهای تعاملی یکی دیگر از امکانات شبکه است که شاید آینده آن حتی از پخش فیلم برحسب تقاضا نیز داغتر باشد. حتی همین حالا هم گروههای بزرگی از جوانان ماجراجو شب و روز مشغول بازی موش و گربه و جنگهای هوایی، زمینی و دریایی در زوایای تاریک و دورافتاده این دنیای مجازی (اینترنت) هستند. اگر آینده بتواند امکانات پخش سه بعدی و کیفیت بالا را عرضه کند، دیگر این دنیای مجازی هیچ چیز از دنیای واقعی کم نخواهد داشت.

چهارمین دسته از کاربردهای شبکه، شاید وسیعترین آنها باشد: خرید از خانه (home shopping). امروزه میلیونها نفر در سراسر جهان هر روز مایحتاج خود را بطور مستقیم از اینترنت تهیه می کنند، و هرگز پا از خانه بیرون نمی گذارند (حداقل برای خرید). هزاران شرکت بزرگ و کوچک کاتالوگ محصولات خود را بصورتی جذاب روی اینترنت گذاشته اند، و برای خرید هر یک از آنها کافیسیت روی جنس موردنظر یک کلیک کنید. کالایی را خریدید، و لی نمی دانید چطور کار می کند؟ نگران نباشید، باز هم اینترنت به شما کمک می کند، و هر اطلاعات و راهنمایی که بخواهید در اختیارتان قرار می دهد.

می خواهید صورتحسابهای خود را پرداخت کنید؟ از آخرین وضعیت حسابهای بانکی خود مطلع شوید؟ و یا سرمایه گذاری جدیدی بکنید؟ باز هم اینترنت در خدمت شماست. امروزه میلیونها نفر در سراسر جهان کارهای مالی و بانکی خود را بصورت الکترونیکی انجام می دهند، و با تقویت مسائل امنیتی شبکه این روند حتی گسترش بیشتری نیز خواهد یافت.

یکی از زمینه هایی که شاید هیچکس تصور اینترنتی شدن آنرا نمی کرد، سمساری بود. حراج اینترنتی اشیاء دست دوم اینک به یکی از تجارتهای بزرگ تبدیل شده است. بر خلاف تجارت الکترونیک معمولی که از مدل مشتری - سرویس دهنده استفاده می کند، حراج اینترنتی در واقع یک سیستم همتا-به-همتا یا خریدار-به-خریدار (consumer-to-consumer) است. امروزه با اصطلاحات زیادی از این دست برخورد می کنید، که در آنها بجای "to" از "2" استفاده می شود (چون تلفظ آنها یکسان است). در شکل ۱-۴ تعدادی از رایجترین این اصطلاحات را ملاحظه می کنید.

شکی نیست که کاربردهای شبکه و اینترنت در آینده سرعت افزایش خواهد یافت، و در زمینه هایی رسوخ خواهد کرد که امروز حتی به تصور کسی نمی آید. چه کسی در سال ۱۹۹۰ می توانست تصور کند که بخش بزرگی از درآمد شرکتهای تلفن از محل پیامهای کوتاهی باشد که دانش آموزان دبیرستانی بطور خستگی ناپذیر و در حالیکه سوار اتوبوس مدرسه هستند، با تلفن همراه خود برای دوستانشان می فرستند؟ این شرکتهای پخوبی می دانند که سرویس پیام کوتاه (Short Message Service - SMS) بسیار سودآور است.

شبکه های کامپیوتری برای افرادی که دور از شهرها زندگی می کنند، نیز مفید است. اینان می توانند راحت در

اصطلاح	نام کامل	مثال
B2C	فروشنده-به-خریدار	خرید کتاب روی اینترنت
B2B	فروشنده-به-فروشنده	خرید قطعات یدکی توسط تولیدکننده
G2C	دولت-به-خریدار	توزیع فرمهای مالیاتی از طریق اینترنت
C2C	خریدار-به-خریدار	حراج اشیاء دست دوم
P2P	همتا-به-همتا	اشتراک فایل

شکل ۴-۱. برخی از انواع تجارت الکترونیک.

روستاهای خود زندگی کنند، و در عین حال به تمام امکانات شهرهای بزرگ هم دسترسی داشته باشند. دانشگاههای آینده به احتمال زیاد حالت ملی و محلی خود را از دست داده، و بصورت بین المللی درخواست خواهند آمد. درمان از راه دور (telemedicine) امروزه به کنترل بیماران محدود می شود، ولی چه کسی می تواند امکانات بالقوه آنرا پیش بینی کند. (یا مثلاً، چقدر خوب می شد اگر می توانستیم یک دوربین دیجیتالی در یخچال خود نصب کنیم، تا هر وقت شیر تمام شد بتوانیم سر راه خانه شیر بخریم!)

۳-۱-۱ کاربردان سیار

کامپیوترهای سیار، مانند کامپیوترهای کتابی و دستیاران دیجیتالی (PDA)، یکی از سریعترین رشدها را در صنعت کامپیوتر تجربه می کنند. اغلب دارندگان این وسایل میل دارند حتی وقتی از خانه دور و یا در سفر هستند، با کامپیوتر خانگی یا دفتری خود ارتباط داشته باشند. در این قبیل موارد دیگر شبکه های کابلی محلی از اعراب ندارد، و باید به فکر شبکه های بیسیم (wireless network) باشیم. در این قسمت نگاهی به کاربردهای شبکه های بیسیم خواهیم داشت.

جالبترین کاربرد شبکه های بیسیم در ایجاد دفاتر سیار است. اغلب افراد میل دارند در سفر همان کارهایی را انجام دهند که در دفتر کار خود انجام می دهند (ایمیل و فکس بفرستند، تلفن راه دور بزنند، فایل های خود را باز کنند، و یا در وب گشت بزنند)، و اصلاً هم کاری به این ندارند که کجا هستند! برای مثال، امروزه در اغلب کنفرانسهای کامپیوتری گردانندگان کنفرانس یک شبکه بیسیم در محوطه کنفرانس راه می اندازند، و هر کسی می تواند با استفاده از یک مودم بیسیم به اینترنت دسترسی پیدا کند. بسیاری از دانشگاهها هم در محوطه خوابگاهی خود شبکه های بیسیم دارند، و به دانشجویان امکان می دهند تا زیر درختان محوطه نشسته و ایمیل های خود را چک کنند، و یا در کتابخانه دانشگاه دنبال مقاله بگردند.

شبکه های بیسیم در امور حمل و نقل (کشتیها، کامیونها و تاکسیها) تحولی بزرگ ایجاد کرده اند. برای مثال، در بسیاری از شهرهای بزرگ رانندگان تاکسی مستقل بوده و عضو هیچ شرکت یا اتحادیه ای نیستند. وقتی کسی به تاکسی نیاز دارد، به یک سرویس مرکزی تلفن می کند، و مشخصات وی (از قبیل مبدأ و مقصد) توسط این سرویس به تمام تاکسیها ارسال می شود. اولین تاکسی که مایل به انجام این سرویس باشد، با فشار یک دکمه اعلام آمادگی کرده و به سراغ مسافر می رود.

شبکه های بیسیم از نظر نظامی نیز اهمیت فوق العاده ای دارند. هیچ ارتشی نمی تواند در جنگهای بزرگ به شبکه های عمومی تکیه کند، و بهتر است شبکه ای خاص خود بر پا کند؛ و چه چیزی بهتر از یک شبکه بیسیم. با وجود شباهتهای بسیار بین شبکه های بیسیم و کامپیوترهای سیار، آنها یکی نیستند (شکل ۱-۵ را ببینید). به تفاوت شبکه های بیسیم ثابت (fixed wireless) و شبکه های بیسیم سیار (mobile wireless) توجه کنید. در بسیاری از دفاتر، کامپیوترهای کتابی سیار بصورت ثابت به شبکه محلی متصل شده اند؛ از طرف دیگر کامپیوتری که با استفاده از مودم به شبکه وصل می شود، سیار است - ولی مسلماً به آن بیسیم نمی توان گفت.

کاربردها	سیار	بیسیم
کامپیوترهای رومیزی در دفتر کار	No	No
کامپیوتری که با مودم به شبکه وصل شده	Yes	No
شبکه بیسیم مستقر در یک ساختمان	No	Yes
دفاتر سیار؛ PDA ها	Yes	Yes

شکل ۱-۵. ترکیب شبکه های بیسیم و کامپیوترهای سیار.

از طرف دیگر، هر شبکه بیسیمی الزاماً سیار نیست. ساختمانهای بسیاری وجود دارند، که بدلیل مشکلات کابل کشی از شبکه های بیسیم استفاده می کنند. امروزه نصب شبکه های بیسیم بسیار ساده شده است، و دردسرهای کابل کشی را هم ندارد.

البته ترکیب بیسیم با کامپیوترهای سیار نیز عملیست، و امروزه کاربردهای مهمی دارد. کسانی که در انبارهای بزرگ یا فروشگاههای شلوغ کار می کنند، تجربه استفاده از این سیستمهای ترکیبی را دارند. این کامپیوترها با اطلاعات ورودی کمی که می گیرند، و با اتصال بیسیم به پایگاه داده مرکزی، کار خود را با سرعت و دقت انجام می دهند. با رشد تکنولوژی بیسیم، مسلماً کاربردهای آن نیز گسترش خواهد یافت. اجازه دهید نگاهی به این احتمالات بیندازیم.

پارکومترهای بیسیم کار دولت و افراد را راحتتر خواهند کرد. این پارکومترها می توانند کارت اعتباری نیز قبول کرده، و سرعت اعتبار آنرا چک کنند؛ و وقتی مدت پارک تمام شد، اگر هنوز اتومبیل شما آنجا بود، نزدیکترین پلیس را خبر می کنند تا آنرا جریمه کند! تخمین زده شده که فقط پلیس ایالات متحده می تواند از این طریق ۱۰ میلیارد دلار بر درآمد خود بیفزاید (Harte et al., 2000). این روش اثرات مثبت زیست محیطی نیز دارد، چون رانندگان خودروها مطمئنند که راهی برای فرار از دست پلیس ندارند، و به ناچار به وسایل نقلیه عمومی روی می آورند.

امروزه ماشینهای خودکار فروش غذا، نوشابه و چیزهای دیگر در همه جا یافت می شوند. اما غذا و نوشابه که از آسمان وارد این ماشینها نمی شود؛ یک مأمور سوار بر کامیون هر از چند گاهی به این ماشینها سرکشی می کند، تا در صورت نیاز آنها را پُر کند. اگر این ماشینها هر روز موجودی خود را از طریق شبکه بیسیم به مرکز اطلاع دهند، مأمور ما می داند سراغ کدام ماشینها باید برود، و چه مقدار کالا احتیاج دارد (و حتی می تواند مسیر حرکت خود را به بهترین نحو برنامه ریزی کند). البته این اطلاعات را از طریق خطوط تلفن هم می توان به مرکز منتقل کرد؛ اما کشیدن یک خط تلفن برای هر ماشین (آن هم برای یک تماس در روز) اصلاً مقرون بصرفه نیست.

زمینه دیگری که شبکه های بیسیم می توانند باعث صرفه جویی شوند، قرائت کتورهای مختلف خانگی است. اگر هر کتور آب، برق و گاز اطلاعات خود را از طریق شبکه بیسیم به شرکت مربوطه منتقل کند، دیگر نیازی به مراجعه کتورخوانها به درب منازل نیست. به همین ترتیب، اگر آشکارسازهای دود و حرارت آلام خود را (بجای راه انداختن آژیر و سر و صدا) مستقیماً به مراکز آتش نشانی بفرستند، بسیار مؤثرتر خواهد بود. با کاهش قیمت دستگاههای رادیویی (که مبنای شبکه های بیسیم هستند)، وسایل اندازه گیری و گزارش دهی بیشتری به استفاده از آنها روی خواهند آورد.

زمینه دیگری از کاربردهای شبکه های بیسیم (که از مدتها قبل نیز انتظار آن می رفت)، ادغام تلفنهای همراه و PDA ها با کامپیوترهای بیسیم است. اولین PDA های بیسیم قادر بودند صفحات ساده وب را روی صفحات کوچک خود نمایش دهند. این سیستم که WAP 1.0 (Wireless Application Protocol) نام داشت، بدلیل خوانا نبودن صفحات، پهنای باند کم، و سرویس ضعیف با شکست مواجه شد. ولی با سرویسها و وسایل جدید WAP 2.0 اوضاع مسلماً بهتر خواهد شد.

ترکیب این تکنولوژیها می تواند به سرویس جدیدی منجر شود که می توان آنرا تجارت سیار (mobile commerce) نامید (Senn, 2000). این پدیده در واقع عامل ترکیب کننده تکنولوژی PDA های بیسیم با تجارت الکترونیک است، که این روزها همه از آن سهم می خواهند، و این امیدواری وجود دارد که افراد به خرید و انجام کارهای بانکی با آن روی آورند. PDA بیسیم می تواند در فروشگاهها و مراکز خرید بعنوان عامل انتقال پول و یا کارت اعتباری عمل کند (بدین ترتیب که این هزینه ها بعداً با صورتحساب تلفن پرداخت شود). نکته مثبت این روش برای فروشندگان آن است که هزینه های کار با شرکتهای اعتباری را به مقدار زیادی پائین می آورد. البته عیب بزرگی نیز دارد: خریداران می توانند قبل از خرید با PDA بیسیم خود قیمتها را با فروشندگان دیگر مقایسه کنند. اگر شرکتهای تلفن سرویس قرائت بارکد را هم به این PDA ها اضافه کنند، که دیگر اوضاع خرابتر می شود! چون دیگر حتی با مخفی کردن قیمتها هم نمی توان جلوی مقایسه قیمتها توسط خریدار را گرفت (کاری که خیلی از فروشندگان به آن امید بسته اند)!

از آنجائیکه محل این دستگاهها همیشه برای اپراتور سیستم مشخص است، می توان سرویسهای خاصی را در اختیار کاربران آنها گذاشت؛ مثلاً، می توان آدرس نزدیکترین کتابفروشی یا رستوران چینی را در اختیار وی گذاشت، و یا آخرین پیش بینی وضعیت هوا را به وی اعلام کرد.

امکانات و کاربردهای این سرویس جدید می تواند بسیار فراتر از مثالهای ساده فوق باشد؛ و خوبی قضیه اینست که کاربران تلفنهای موبایل عادت دارند برای هر چیزی پول بدهند (درست بر خلاف کاربران اینترنت که همه چیز را مجانی می خواهند)! اگر یک سایت اینترنتی برای قبول کارت اعتباری از شما درخواست پول کند، فوراً فریادتان بلند خواهد شد، ولی اگر همین اتفاق روی سرویسهای موبایل بیفتد، بدون هیچ اعتراضی قبول می کنید (البته فعلاً).

بگذارید کمی هم به آینده نگاه کنیم: شبکه های شخصی (Personal Area Network)، و کامپیوترهای پوشیدنی (wearable computer). به تازگی IBM ساعتی ساخته که سیستم عامل لینوکس (Linux) روی آن اجرا می شود، و می تواند به اینترنت وصل شده و ایمیل رد و بدل کند (Narayanawami et al., 2002). در آینده دیگر چیزی بعنوان کارت ویزیت بین افراد رد و بدل نخواهد شد، و آنها می توانند با یک تماس ساعت مچی تمام اطلاعات طرف مقابل را دریافت کنند. به احتمال زیاد کامپیوترهای پوشیدنی (که اطلاعات زیستی فرد را در خود دارند) جای کارتهای مغناطیسی را برای ورود به مکانهای حساس خواهند گرفت، و یا می توانند در هر لحظه مکان فرد را اعلام کنند. امکانات این سیستمها تقریباً بی شمار است.

ساعتها و رادیوهای هوشمند را همه ما سالهاست از طریق فیلمهای جیمز باند می شناسیم، ولی آیا تا بحال گرد و غبار هوشمند به گوشتان خورده است؟ محققان دانشگاه برکلی اخیراً یک کامپیوتر بیسیم ساخته اند، که در مکعبی با ابعاد 1 mm جای می گیرد (Warneke et al., 2001). با این کامپیوترها می توان مسیر حرکت چمدان در فرودگاه ها، و یا جانوران (مثلاً، پرندگان مهاجر) را با دقت کنترل کرد.

۱-۱-۴ تبعات اجتماعی

گسترش روزافزون شبکه های کامپیوتری باعث ایجاد مسائل اجتماعی، اخلاقی و سیاسی خاص خود شده است که در این قسمت برخی از آنها را بررسی می کنیم. یکی از امکانات شبکه های کامپیوتری تبادل آزاد و سریع اطلاعات و اخبار است. البته تا وقتی این پیامها در محدوده های فنی باقی بماند، مشکل چندانی وجود نخواهد داشت؛ در دسر وقتی شروع می شود که صحبت به مسایل حساس (از قبیل سیاست، مذهب یا سکس) کشیده شود.

نظرها و دیدگاههایی که توسط اعضای یک گروه خبری پست می شود، ممکنست برای افراد دیگر بسیار ناهنجار و موهن باشد؛ و وقتی پیامها به متن محدود نشود، کار بدتر هم خواهد شد. امروزه بر راحتی می توان

عکسهای بسیار واضح و با کیفیت عالی (و یا حتی کلیپهای ویدئویی کوچک) را از طریق اینترنت منتشر کرد. برخی از افراد در زندگی به فلسفه «زندگی کن، و بگذار زندگی کنند» معتقدند، اما عده زیادی هم هستند که احساس می کنند به برخی مطالب (مانند حمله به کشورها یا مذاهب دیگر، صور قبیحه و غیره) نباید اجازه انتشار داد. کشورهای مختلف هم دارای قوانین متفاوتی در این زمینه ها هستند؛ و جدال از همین جا شروع می شود. در این میان بسیاری از افراد اپراتورهای شبکه را (مانند روزنامه ها و مجلات) مسئول محتویات شبکه می دانند، اما واقعیت اینست که یک شبکه بیشتر شبیه اداره تلفن و پست است تا روزنامه یا مجله (و نمی توان آنرا مسئول چیزهایی که از این طریق مبادله می شود، دانست). از طرف دیگر، اگر اپراتور شبکه اجازه داشته باشد مطالب را سانسور کند، به احتمال زیاد (برای فرار از متهم شدن) روی کوچکترین چیزها هم انگشت خواهد گذاشت، و بدین ترتیب حقوق افراد در زمینه آزادی بیان از بین خواهد رفت. بحث موافق و مخالف همچنان ادامه دارد (و براحتی می توان حدس زد که به این زودی ها هم به نتیجه نخواهد رسید).

بحث جالب دیگر حقوق و رابطه کارگر و کارفرماست. بسیاری از افراد در محل کار خود ایمیل می فرستند، و یا ایمیل های رسیده را می خوانند. برخی از کارفرمایان ادعا می کنند که آنها حق دارند ایمیل های کارمندان خود را بخوانند و یا آنها را سانسور کنند؛ و صد البته کارمندان با این حرفها موافق نیستند!

حتی اگر بپذیریم کارفرمایان در ادعای خود محق هستند، آیا می توان این رابطه را به دانشجوی و دانشگاه (و یا دانش آموز و مدرسه) تعمیم داد؟ در سال ۱۹۹۴ دانشگاه کارنگی سلون تصمیم گرفت برخی از پیامهای رسیده را که در آنها به موضوعات سکسی پرداخته شده بود، سانسور کند (با این استدلال که این مطالب برای افراد زیر ۱۸ سال مناسب نیست). سالها طول کشید تا پس لرزه های این اقدام فروکش کند.

و از همه مهمتر رابطه دولت با شهروندان است. اداره آگاهی فدرال ایالات متحده (FBI) سالهاست سیستمی را در مراکز ارائه سرویس اینترنت (ISP) نصب کرده، که به آن اجازه می دهد تا ایمیل های ورودی و خروجی را تجسس کند (Blaze and Bellovin, 2000; Sobel, 2001; Zacks, 2001). نام این سیستم Carnivore (گلی گوشتخوار) بود، که بدلیل حساسیتهای ایجاد شده در جامعه به نام کم ضررتر DCS1000 تغییر داده شد (با این حال کار آن همچنان جاسوسی در ایمیل های مردم بود). طبق اصلاحیه چهارم قانون اساسی ایالت متحده آمریکا، دولت بدون مجوز قانونی حق تجسس در احوال شخصی افراد را ندارد. اینکه این قانون نوشته شده در قرن هیجدهم هنوز در قرن بیست و یک اعتبار دارد یا خیر، را آینده روشن خواهد کرد.

فضولی در کار مردم به دولت محدود نمی شود؛ بخش خصوصی هم از این گناه در امان نیست. برای مثال، مرورگرهای وب از فایل های کوچکی بنام کوکی (cookie) استفاده می کنند که اطلاعات شخصی افراد را در اختیار شرکتها می گذارد، و حتی می تواند منجر به افشای شماره کارتهای اعتباری و اطلاعات محرمانه دیگر روی اینترنت شود (Berghel, 2001).

در شبکه های کامپیوتری می توان پیامهای بدون نام و نشانی فرستاد، که در جای خود می تواند مفید باشد. مثلاً، کارمندان، دانشجویان و یا مردم عادی می توانند بدین طریق اعمال خلاف رؤسای شرکتها، استادان و سیاستمداران را به اطلاع عموم برسانند بدون آنکه از اقدامات تلافی جوینان آنها ترسی به دل راه دهند. (البته در بسیاری از کشورها از این قبیل اطلاعات بدون منبع نمی توان در دادگاه بعنوان مدرک جرم استفاده کرد).

وضعیت فعلی شبکه های کامپیوتری شبیه موقعیت کتابهای چاپی در ابتدای اختراع این صنعت بود: افراد عادی وسیله ای بدست آورده بودند تا با آن صدای خود را به گوش دیگران برسانند. اما این آزادی با خود تبعات اجتماعی، سیاسی و اخلاقی خاصی بدنبال داشت، که همچنان لاینحل باقی مانده است.

زندگی همیشه بدین منوال است: هر سکه ای دو رو دارد. اینترنت هم از این قاعده مستثنی نیست. امکان

دسترسی سریع و آسان به اطلاعات ارمغان اینترنت است. ولی خروارها اطلاعات منفی، غلط و گمراه کننده وجه دیگر آن است. راهنمایی بهداشتی که تازگی در اینترنت خوانده اید (و احتمالاً می خواهید به آن عمل کنید)، ممکنست از یک برنده جایزه نوبل آمده باشد، یا یک دانش آموز بازیگوش دبیرستانی.

شبکه های کامپیوتری انواع جدیدی از جرم و رفتارهای ضداجتماعی را نیز با خود آورده اند. هر روز که صندوق پستی خود را باز می کنید، دهها و صدها پیام مزخرف و بدردنخور در آن می بینید (و حتی کم کم به آن عادت کرده اید). این نتیجه کار افرادیست که میلیونها آدرس ایمیل را روی یک CD جمع کرده، و (بدون رضایت صاحبان این آدرسها) به این و آن می فروشند. و تازه اینها دسته بی آزارها هستند؛ این روزها کسی پیدا نمی شود که صابون و پیروسهایی که از طریق ایمیل منتشر می شوند، به تنش نخورده باشد.

دزدی هویت یکی دیگر از خطرات سرقت اطلاعات از طریق اینترنت است. (درباره سرقت های ادبی و نقض گسترده و وسیع قانون حق التألیف در اینترنت قبلاً هم صحبت کردیم.)

بسیاری از این مشکلات حاصل ضعف (و یا عدم رعایت) مسائل امنیتی در اینترنت است. اگر تمام پیامهای ایمیل بصورت رمز در آیند، سرقت اطلاعات بسیار مشکلتر خواهد شد (این تکنولوژی توسعه زیادی یافته، که در فصل ۸ مفصلاً به آن خواهیم پرداخت). مشکل اینجاست که بالا بردن سطح امنیتی مترادف است با بالا رفتن هزینه، و این چیزی نیست که به آسانی پذیرفته شود. تعداد زیادی از این مسائل نیز به مشکلات و باگهای موجود در نرم افزارها مربوط می شود، که خود حاصل بزرگتر و پیچیده تر شدن آنهاست. اگر روی برنامه ها بر حسب بزرگی و پیچیدگی آنها مالیات بسته شود، شاید این مشکل تا حدی حل شود؛ البته خیلی ها این راه حل را نمی پسندند! پس دادن پول برنامه های معیوب نیز می تواند راه حل خوبی باشد، فقط مشکل اینجاست که چنین قانونی ظرف یک سال کل صنعت نرم افزار را ورشکست خواهد کرد!

۲-۱ سخت افزار شبکه

اکنون وقت آنست که توجه خود را از مسائل متفرقه به موضوع اصلی (یعنی همان شبکه های کامپیوتری) معطوف کنیم. هیچ طبقه بندی پذیرفته شده ای که در برگیرنده تمام انواع شبکه های کامپیوتری باشد، وجود ندارد، ولی در این میان می توان به دو عامل مهم توجه کرد: تکنولوژی انتقال و اندازه شبکه. اجازه دهید این دو را جداگانه بررسی کنیم. امروزه دو تکنولوژی انتقال بیش از همه گسترش یافته و فراگیر هستند:

۱. ارتباطات پخش (broadcast)
۲. ارتباطات همتا به همتا (peer-to-peer)

شبکه های پخش (broadcast network) دارای یک کانال مخابراتی هستند که بین همه کامپیوترهای شبکه به اشتراک گذاشته شده است. هر یک از کامپیوترها می توانند پیامهای خود را در بسته (packet) های کوچک مخابره کنند، و تمام کامپیوترهای دیگر این پیامها را دریافت خواهند کرد. آدرس کامپیوتری که این بسته در حقیقت برای وی ارسال شده، در بخشی از پیام نوشته می شود. هر کامپیوتری به محض دریافت بسته، آدرس گیرنده را چک می کند؛ اگر پیام برای او باشد، آنرا پردازش می کند؛ ولی اگر پیام متعلق به دیگری باشد، بسادگی آنرا نادیده می گیرد. بعنوان مقایسه، فرض کنید کسی در انتهای راهرویی که در دو طرف آن پُر از اتاقهای متعدد است، فریاد بزند «آقای واتسون، بیاید. با شما کار دارم.» با اینکه این پیام به گوش همه افراد می رسد، فقط آقای واتسون به آن پاسخ می دهد و دیگران توجهی به آن نخواهد کرد. یا وقتی در سالن انتظار فرودگاه اعلام می شود که «مسافران پرواز ۶۴۴ به خروجی ۱۲ مراجعه کنند»، فقط آنهایی که بلیط این پرواز را دارند، عکس العمل نشان می دهند.

در شبکه های پخش با تعبیه یک کُد خاص در فیلد آدرس (address field) می توان یک پیام را به تمام

کامپیوترها ارسال کرد. چنین پیامی را همه کامپیوترها متعلق به خود تلقی کرده، و آنرا می خوانند. به این تکنیک پخش (broadcasting) گفته می شود. در برخی از سیستمهای پخش امکان ارسال پیام به دسته ای از کامپیوترها نیز وجود دارد، که به آن پخش گروهی (multicasting) می گویند. بدین منظور، معمولاً از یک بیت خاص در فیلد آدرس استفاده می شود، و همه آنهایی که این بیت در آنها وجود دارد عضو گروه محسوب شده و پیام را می گیرند. در شبکه های همتا به همتا (peer-to-peer network) بین تک تک کامپیوترها مسیر ارتباطی مستقل وجود دارد. البته وقتی یک بسته بخواهد از کامپیوتری به کامپیوتر دیگر برود، احتمالاً سر راه خود از چند ماشین بینابینی نیز عبور خواهد کرد. معمولاً در این قبیل شبکه ها مسیرهای متعددی بین دو کامپیوتر خاص می توان برقرار کرد، که از نظر طول مسیر با هم تفاوت دارند، و یافتن کوتاهترین مسیر یکی از مسایل مهم در این گونه شبکه ها است. بعنوان یک قاعده کلی (البته با استثنای متعدد)، شبکه های کوچک، متمرکز و محلی از نوع پخش هستند، و شبکه های بزرگ و گسترده از نوع همتا به همتا. به ارتباط همتا به همتا گاهی پخش تکمی (unicasting) نیز گفته می شود. روش دیگر طبقه بندی شبکه ها اندازه شبکه است. در شکل ۱-۶ نوعی طبقه بندی بر اساس اندازه را مشاهده می کنید.

در بالا شبکه های شخصی (Personal Area Network) را می بینید (شبکه هایی که متعلق به یک فرد خاص هستند). ارتباط بیسیم بین ماوس، کی بورد، چاپگر، PDA و کامپیوتر از این نوع است. بعد از آن شبکه های محلی (LAN)، شهری (MAN) و گسترده (WAN) می آیند. در آخر هم شبکه ها (شبکه ای که هر نقطه از آن خود یک شبکه کامل است) - و اینترنت معروفترین نمونه آن است - می آید. در این طبقه بندی فاصله کامپیوترها اهمیت زیادی دارد، چون تکنولوژی ارتباطی به شدت به آن وابسته است. در این کتاب درباره تمام این شبکه ها صحبت خواهیم کرد. در زیر هر یک از این شبکه ها را مختصراً معرفی می کنیم.

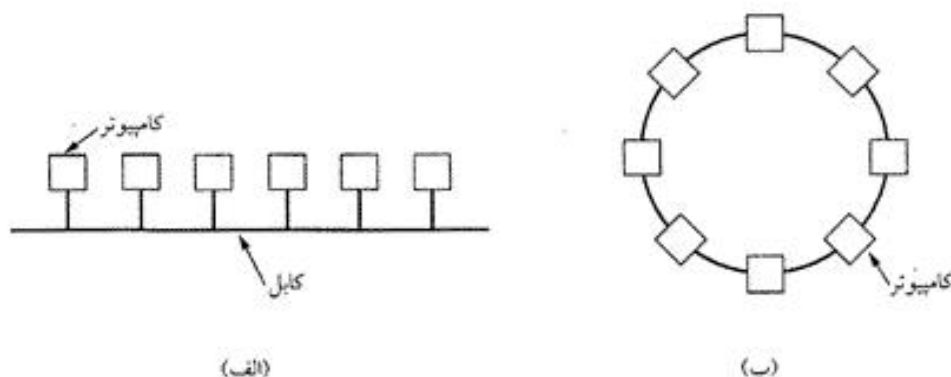
۱-۲-۱ شبکه های محلی (Local Area Network)

شبکه محلی، یا LAN، شبکه ایست خصوصی واقع در یک ساختمان یا مجتمع، که حداکثر ابعاد آن یکی دو کیلومتر باشد. از این نوع شبکه معمولاً برای متصل کردن کامپیوترهای یک شرکت و به اشتراک گذاشتن منابع (مانند چاپگر) یا مبادله اطلاعات استفاده می شود. یک شبکه LAN سه مشخصه اصلی دارد، که آنرا از سایر انواع شبکه متمایز می کند: (۱) اندازه، (۲) تکنولوژی انتقال اطلاعات، و (۳) توپولوژی (topology).

اندازه LAN بسیار محدود است، بگونه ای که زمان انتقال سیگنالها در آن (حتی در بدترین شرایط) بسیار کم و قابل قبول پیش بینی است. دانستن این محدودیت ها برای طراحی شبکه بسیار مهم و اساسی است، و باعث ساده تر شدن مدیریت شبکه نیز می شود.

فاصله پردازنده ها	محل نسبی پردازنده ها	نمونه
1 m	روی یک میز	شبکه شخصی
10 m	یک اتاق	شبکه محلی
100 m	یک ساختمان	.
1 km	یک مجتمع	.
10 km	یک شهر	شبکه شهری
100 km	یک کشور	شبکه گسترده
1000 km	یک قاره	.
10,000 km	کره زمین	اینترنت

شکل ۱-۶. طبقه بندی شبکه ها بر اساس اندازه و فاصله پردازنده ها.



شکل ۷-۱. دو شبکه پخش. (الف) باس. (ب) حلقوی.

تکنولوژی انتقال اطلاعات در LAN معمولاً به کابل متکیست (و از این نظر بسیار شبیه شبکه های تلفن است). سرعت انتقال اطلاعات در LAN بین ۱۰ تا ۱۰۰ میلیون بیت در ثانیه (که با Mbps مشخص می شود)، تأخیر انتشار در آن کم (در حد میکرو یا نانو ثانیه)، و خطا در آن بسیار اندک است. LAN های جدیدتر به سرعت 10 Gbps نیز دست یافته اند. سرعت انتقال در شبکه معمولاً با واحد مگابیت بر ثانیه (1,000,000 bits/sec) یا گیگابیت بر ثانیه (1,000,000,000 bits/sec) اندازه گیری می شود.

توپولوژی های مختلفی برای شبکه های محلی پخش وجود دارد، که در شکل ۷-۱ دو تا از آنها را می بینید. در یک شبکه باس (bus network - شبکه ای با کابل کشی خطی) در هر لحظه فقط یکی از کامپیوترها مجاز به استفاده از خط و ارسال اطلاعات است، و تمام ماشینهای دیگر بایستی در این مدت از ارسال هر گونه اطلاعات خودداری کنند. در این قبیل شبکه ها بایستی مکانیزمی برای حل اختلاف (در مواقعی که دو کامپیوتر همزمان با هم شروع به ارسال می کنند) وجود داشته باشد. این مکانیزم می تواند متمرکز (centralized) یا توزیع شده (distributed) باشد. یکی از مکانیزمهای حل اختلاف در شبکه های باس پخش IEEE 802.3 نام دارد (که به اترنت - Ethernet - نیز معروف است)، و با کنترل غیرمتمرکز در سرعتهای 10 Mbps تا 10 Gbps کار می کند. کامپیوترهای یک شبکه اترنت در هر زمانی می توانند اقدام به ارسال کنند، ولی اگر تصادفی بین آنها پیش آمد، هر یک از آنها مدتی (که بصورت تصادفی تعیین می شود) صبر کرده و دوباره سعی خواهد کرد.

نوع دیگری از شبکه های پخش، شبکه حلقوی (ring network) است. در یک شبکه حلقوی، هر بیت اطلاعات بصورت مستقل (و بدون اینکه بخواهد منتظر سایر بیت های بسته ای که به آن تعلق دارد، شود) در شبکه منتشر می شود. با توجه به سرعت بالای انتشار الکترونها در محیطهای رسانا، هر بیت حتی قبل از انتشار بیت های بعدی، می تواند پاره ها محیط شبکه را دور بزند. در این نوع شبکه هم بایستی مکانیزمی برای حل اختلاف بین کامپیوترهای متخاصم وجود داشته باشد. اغلب این مکانیزمها به نوعی نوبت بندی متکی هستند. یکی از این مکانیزمها IEEE 802.5 (یا IBM Token Ring) است، که در سرعتهای 4 Mbps و 16 Mbps کار می کند. FDDI یکی دیگر از شبکه های پخش حلقوی است.

نوع دیگری از تقسیم بندی شبکه های پخش بر حسب نحوه اختصاص کانال است، و به استاتیک و دینامیک تقسیم می شود. در اختصاص کانال استاتیک هر کامپیوتر برای مدت زمانی محدود و مشخص کانال را در دست می گیرد، و فقط در این بُرش زمانیست که می تواند اطلاعات ارسال کند. در این روش پهنای باند کانال بشدت هدر می رود، چون بسیار پیش می آید که وقتی نوبت به یک کامپیوتر می رسد، چیزی برای گفتن ندارد. به همین دلیل، سیستمهای امروزی اغلب پهنای باند را بصورت دینامیک (بر حسب نیاز) تخصیص می دهند.

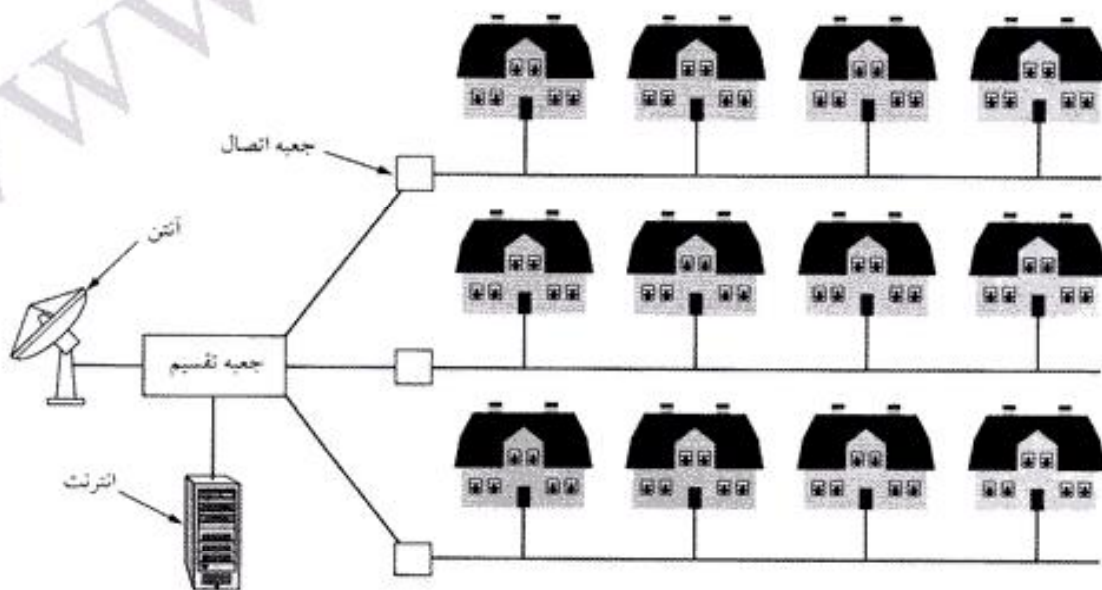
تخصیص دینامیک کانال خود بر دو نوع است: متمرکز و غیرمتمرکز. در نوع متمرکز یک موجودیت مشخص (بنام واحد تصمیم گیرنده - Arbitration Unit) وجود دارد، که درخواست ها را دریافت کرده، و بر اساس نوعی الگوریتم داخلی نوبت ها را تعیین می کند. در تخصیص کانال غیرمتمرکز این موجودیت تصمیم گیرنده وجود ندارد، و تصمیم گیری بر عهده تک تک کامپیوترهاست. شاید فکر کنید این روش جز هرج و مرج نتیجه ای ندارد، ولی چنین نیست (در آینده خواهید دید که الگوریتمهایی وجود دارند که می توانند به اوضاع سر و سامان بدهند).

۲-۲-۱ شبکه های شهری (Metropolitan Area Network)

شبکه شهری، یا MAN، شبکه ایست که یک شهر را پوشش می دهد. شبکه های تلویزیون کابلی بهترین نمونه MAN هستند. اولین شبکه های تلویزیون کابلی در نقاط کور شهرها راه اندازی شدند، بدین ترتیب که یک آنتن مرکزی و بزرگ در محلی که فرستنده اصلی را می دید نصب، و از این آنتن کابلهایی به مشترکان محروم از برنامه های تلویزیونی کشیده می شد.

در ابتدا این سیستمها بطور اختصاصی برای هر محل ساخته می شد، ولی بزودی شرکتهای بزرگ بوی پول را از آن استشمام کردند، و با کسب اجازه دولت تمام شهر را زیر پوشش کابلهای خود بردند. این شبکه ها برای پخش برنامه هم برنامه ریزی خاصی دارند، مثلاً یک شبکه فقط اخبار پخش می کند، دیگری فقط برنامه های ورزشی دارد، و آن یکی فقط آشپزی. این شبکه ها بسیار تخصصی بودند، و تا اواخر دهه ۱۹۹۰ فقط برنامه های تلویزیونی پخش می کردند.

با شروع گرایش عمومی به اینترنت، گردانندگان این شبکه ها بزودی دریافتند که با تغییری مختصر در سیستمهای خود می توانند از قسمتهای بالاستفاده پهنای باند برای ارائه سرویسهای دوطرفه اینترنت بهره ببرند. از این لحظه بود که شبکه های تلویزیون کابلی تبدیل به شبکه های شهری (MAN) شدند. در شکل ۸-۱ نمایی تقریبی از یک شبکه شهری را ملاحظه می کنید. در این شکل می بینید که ابتدا سیگنالهای تلویزیونی و اینترنتی ترکیب شده و به یک مرکز فوق-توزیع (head end) می روند، تا از آنجا در خانه های مشترکان توزیع شوند. (در فصل ۲ باز هم به این مبحث خواهیم پرداخت).



شکل ۸-۱ یک شبکه شهری مبتنی بر تلویزیون کابلی.

تلویزیون کابلی تنها مثال زنده MAN نیست. اخیراً تحقیقاتی بر روی اینترنت بیسیم پرسرعت (high-speed wireless Internet) انجام شده، که نتیجه آن نوع دیگری از MAN خواهد بود. با این استاندارد که IEEE 802.16 نام دارد، در فصل ۲ بیشتر آشنا خواهید شد.

۳-۲-۱ شبکه های گسترده (Wide Area Network)

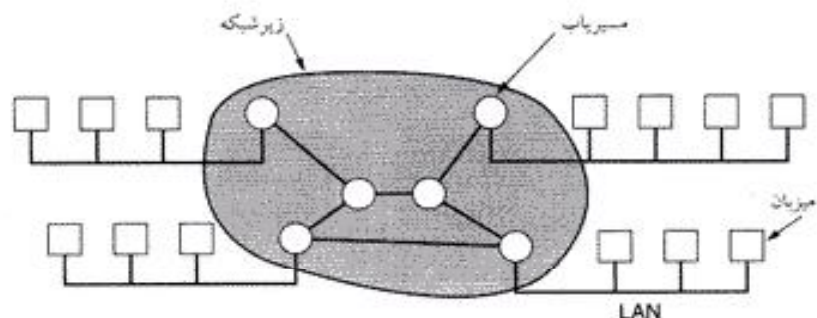
شبکه گسترده، یا WAN، گستره جغرافیایی بزرگی (مانند یک کشور یا قاره) دارد. در این نوع شبکه کامپیوترهایی هستند که برنامه های کاربردی روی آنها اجرا می شود، و معمولاً به آنها میزبان (host) می گویند. این کامپیوترها توسط زیرشبکه های مخابراتی (communication subnet) - یا بطور مختصر، زیرشبکه - به هم متصل می شوند. میزبانها متعلق به افراد هستند، در حالیکه زیرشبکه اغلب به شرکت های مخابرات تعلق دارد. وظیفه زیرشبکه انتقال پیام از یک میزبان به میزبان دیگر است. جدا کردن این دو بخش (میزبانها و زیرشبکه) طراحی شبکه های WAN را تا حد زیادی ساده می کند.

در اغلب شبکه های گسترده، زیرشبکه از دو بخش مجزا تشکیل می شود: خطوط انتقال (transmission lines) و تجهیزات سوئیچینگ (switching elements). خطوط انتقال وظیفه رد و بدل کردن اطلاعات را بر عهده دارند، و می توان برای ایجاد آنها از سیم مسی، فیبر نوری یا حتی امواج رادیویی استفاده کرد. تجهیزات سوئیچینگ کامپیوترهای خاصی هستند که ارتباط بین خطوط انتقال را برقرار می کنند. وقتی داده ها از یک خط وارد می شود، این کامپیوتر باید مسیر خروجی آنها مشخص کند. این کامپیوترهای سوئیچینگ به نامهای مختلفی خوانده می شوند، که می توان از معروفترین آنها به مسیریاب (router) اشاره کرد.

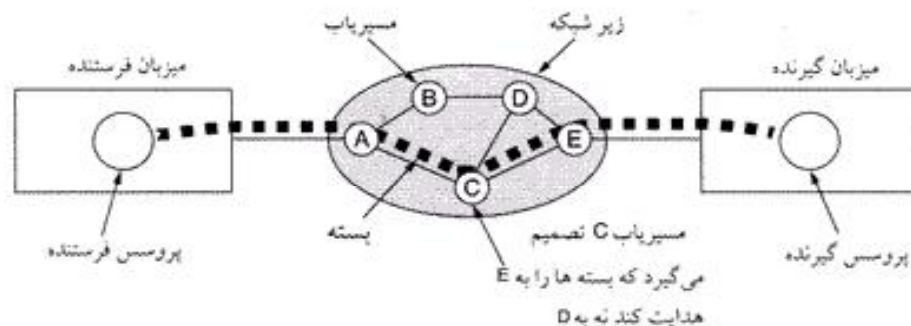
در این مدل (شکل ۹-۱) معمولاً هر کامپیوتر میزبان در یک شبکه محلی قرار دارد که از طریق یک مسیریاب به قسمتهای دیگر متصل می شود (البته در مواردی میزبان می تواند مستقیماً نیز به مسیریاب وصل باشد). به مجموعه خطوط مخابراتی و مسیریاب ها (منهای کامپیوترهای میزبان) زیرشبکه گفته می شود.

معنای اولیه زیرشبکه همان است که در بالا گفته شد، یعنی مجموعه خطوط مخابراتی و مسیریاب ها که وظیفه آنها انتقال اطلاعات از یک میزبان به میزبان دیگر است. اما سالها بعد از این اصطلاح در ارتباط با آدرس دهی شبکه ها نیز استفاده شد (فصل ۵ را ببینید). متأسفانه هنوز اصطلاح مناسبی برای کاربرد اولیه آن پیدا نشده است، و ما هم (با کمی شک و تردید) آنها را در هر دو مورد بکار خواهیم برد، که با توجه به موضوع بحث می توان معنای مورد نظر را استنباط کرد.

در بسیاری از WAN ها تعداد زیادی خطوط انتقال وجود دارد، که هر کدام یک جفت مسیریاب را به هم وصل می کنند. اگر دو مسیریاب که اتصال فیزیکی مستقیم ندارند، بخواهند با یکدیگر ارتباط برقرار کنند، باید این کار را بصورت غیرمستقیم (از طریق مسیریاب های دیگر) انجام دهند. وقتی یک بسته داده در مسیر خود (از مسیریاب



شکل ۹-۱. ارتباط بین کامپیوترهای میزبان و LAN ها در یک زیرشبکه.



شکل ۱-۱۰. استریم (جریان) بسته ها از مبدأ به مقصد.

مبدأ به مسیر یاب مقصد) از چند مسیر یاب بینایی عبور می کند، ابتدا بصورت کامل دریافت و ذخیره شده، و پس از آزاد شدن خط خروجی به سمت مقصد فرستاده می شود. زیر شبکه هایی که بر اساس این قاعده عمل می کنند، به زیر شبکه ذخیره-ارسال (store-and-forward) یا سوئیچ بسته (packet-switched) معروفند. تقریباً تمامی شبکه های WAN (بجز شبکه های ماهواره ای) از این نوع هستند. اگر اندازه بسته ها کوچک و یکسان باشد، به آنها سلول (cell) نیز گفته می شود.

به دلیل اهمیت مفهوم زیر شبکه سوئیچ بسته، لازمست کمی بیشتر درباره آن توضیح دهیم. وقتی پروسی در یک میزبان می خواهد پیامی به میزبان دیگر بفرستد، ابتدا آنرا به بسته های کوچکتر (که پشت سر هم شماره گذاری می شوند) تقسیم می کند. این بسته ها بصورت مستقل به طرف مقابل ارسال می شوند، و بعد از رسیدن تمامی آنها به مقصد، در آنجا دوباره به یکدیگر مونتاژ شده و پیام اصلی را می سازند (شکل ۱-۱۰ را ببینید).

در این شکل تمام بسته ها از طریق مسیر ACE به مقصد رسیده اند (در حالیکه مسیرهای ABDE و ACDE نیز وجود داشت). در برخی از شبکه ها این یک الزام است، یعنی تمام بسته های یک پیام باید از یک مسیر عبور کنند، در حالیکه در شبکه های دیگر این بسته ها می توانند از مسیرهای مختلف عبور کنند. البته، اگر مسیری بهترین مسیر ممکن باشد (مانند ACE در اینجا)، همه بسته ها از آن مسیر عبور خواهند کرد، حتی اگر شبکه چنین الزامی را تحمیل نکرده باشد.

تصمیم گیری درباره مسیر ارسال بسته ها امری داخلی است، یعنی هر مسیر یاب خود درباره آن تصمیم می گیرد. وقتی یک بسته به مسیر یاب A می رسد، این مسیر یاب A است که تصمیم می گیرد آنرا از طریق خط متصل به B بفرستد یا از خط متصل به C. مسیر یاب ها برای تصمیم گیری درباره مسیر بسته ها از الگوریتمهای مسیر یابی (routing algorithm) استفاده می کنند، که درباره آنها در فصل ۵ صحبت خواهیم کرد.

تمام شبکه های WAN از نوع سوئیچ بسته نیستند، مانند سیستمهای ماهواره ای. در این سیستمها هر روتر آنتنی دارد که از طریق آن اطلاعات را به ماهواره می فرستد، یا اطلاعات ارسالی آن را دریافت می کند. تمام مسیر یاب های این مجموعه می توانند به ماهواره گوش کنند (و حتی برخی از آنها به اطلاعات ارسالی از مسیر یاب های همسایه نیز گوش می کنند). البته شبکه هایی هم وجود دارد که فقط برخی از مسیر یاب های آن (و نه همه آنها) ارتباط ماهواره ای دارند. شبکه های ماهواره ای ذاتاً از نوع پخش هستند، و اغلب در جاهایی بکار می روند که این طریقه پخش اهمیت داشته باشد.

۴-۲-۱ شبکه های بیسیم (Wireless Network)

مخابرات دیجیتال بیسیم ایده جدیدی نیست. گد موری که فیزیکدان ایتالیایی گاگلیلمو مارکونی در سال ۱۹۰۱ از یک کشتی به ساحل مخابره کرد، را می توان اولین پیام دیجیتال بیسیم محسوب کرد. سیستمهای جدید مخابرات

بسیم فقط کارایی بهتری دارند، اما ایده اصلی در واقع همان است. در ساده‌ترین صورت، شبکه‌های بیسیم را می‌توان به سه دسته بزرگ تقسیم کرد:

۱. ارتباطات بین سیستمی
۲. LAN های بیسیم
۳. WAN های بیسیم

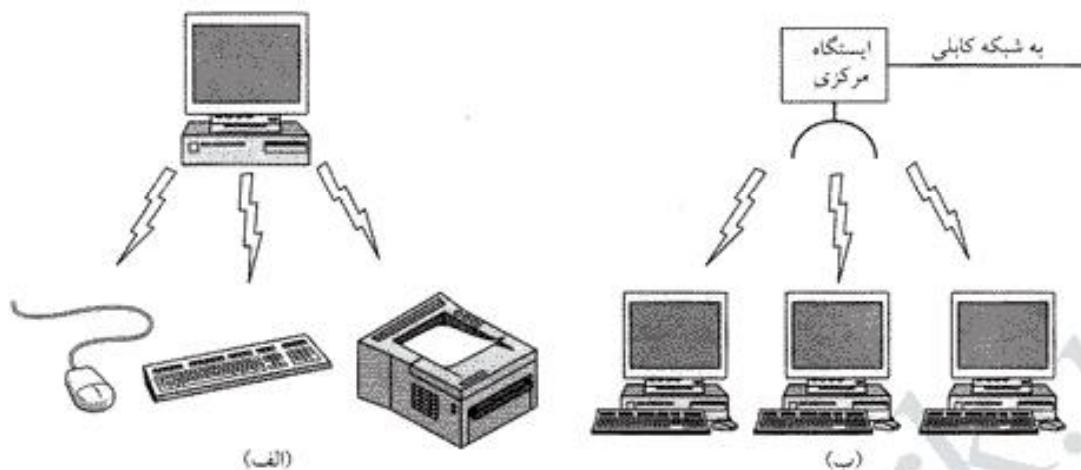
ارتباطات بین سیستمی (system interconnection) یعنی برقراری ارتباط بین قطعات داخلی یک کامپیوتر با استفاده از امواج رادیویی کوتاه بُرد. تقریباً هر کامپیوتری یک مانتیور، صفحه کلید یا ماوس دارد که معمولاً با کابل به آن متصل می‌شوند. برای بسیاری از کاربران خانگی (و حتی اداری) وصل کردن این کابلها (با اینکه آنها طوری طراحی شده‌اند که نتوان هیچکدام را به دیگری وصل کرد) یک کار شاق است، و برای این کار دست به دامان تکنسین‌های کامپیوتر می‌شوند. به همین علت، برخی از شرکت‌های سازنده کامپیوتر دور هم جمع شدند، و یک شبکه بیسیم با بُرد کوتاه بنام بلوتوث (Bluetooth) اختراع کردند که این قطعات را بدون استفاده از سیم به کامپیوتر متصل می‌کند. تکنولوژی بلوتوث اجازه می‌دهد تا دستگاههایی مانند چاپگر، دوربین دیجیتال، گوشی، و اسکنر نیز (با قوار گرفتن در بُرد امواج آن) به کامپیوتر متصل شوند. برای این کار به هیچ اتصال فیزیکی یا حتی نصب درایور نیاز نیست، و فقط کافیس دستگاه را روشن کرده و در بُرد کامپیوتر قرار دهید، تا کار کنند. برای بسیاری از کاربران این یک مزیت خارق‌العاده است.

ارتباطات بین سیستمی اساساً بر الگوی اصلی-پیرو (master-slave) مبتنی است (شکل ۱-۱۱ الف). در این سیستم، کامپیوتر اصلی است و با وسایل جانبی بعنوان رعایای خود صحبت می‌کند. این کامپیوتر اصلی است که به رعایا می‌گوید از چه آدرسی استفاده کنند، کی حرف بزنند، چه مدت حرف بزنند، روی چه فرکانسی صحبت کنند، و مانند آن. در باره تکنولوژی بلوتوث در فصل ۴ مفصلاً صحبت خواهیم کرد.

نوع دیگر ارتباطات بیسیم، شبکه محلی بیسیم (یا LAN بیسیم) است. در این سیستم هر کامپیوتر یک مودم رادیویی و یک آنتن دارد، که به وسیله آن با کامپیوترهای دیگر ارتباط برقرار می‌کند. در اغلب این سیستمها یک آنتن مرکزی روی پشت بام وجود دارد (شکل ۱-۱۱ ب)، که ارتباط بین کامپیوترها را تسهیل می‌کند، اما اگر شبکه باندازه کافی کوچک باشد، آنها می‌توانند مستقیماً با هم حرف بزنند. این نوع شبکه در دفاتر کوچک، خانه‌ها و جاهایی که کابل کشی مشکل است، سرعت در حال گسترش است. مهمترین استاندارد LAN های بیسیم IEEE 802.11 نام دارد، که در اغلب سیستمها از آن استفاده می‌شود.

نوع سوم ارتباطات بیسیم، سیستمهای WAN بیسیم است. شبکه رادیویی بکار رفته در سیستمهای تلفن همراه از این نوع است. این سیستمها اکنون نسل سوم خود را پشت سر می‌گذارند. نسل اول آنالوگ بود و فقط برای صدا از آن استفاده می‌شد. نسل دوم با اینکه دیجیتال شده بود، ولی باز هم فقط از صدا پشتیبانی می‌کرد. نسل سوم نیز دیجیتال است، و اینک همزمان از صدا و دیتا پشتیبانی می‌کند. WAN های بیسیم اساساً تفاوتی با LAN بیسیم ندارند، و فقط بُرد آنها بیشتر و البته نرخ انتقال داده‌ها کمتر است. LAN های بیسیم می‌توانند داده‌ها را با سرعتهایی در حد 50 Mbps (در محدوده چند ده متر) منتقل کنند. نرخ انتقال داده‌ها در WAN های بیسیم بزحمت به 1 Mbps می‌رسد، ولی بُرد آنها بجای متر با کیلومتر سنجیده می‌شود. در فصل ۲ درباره این سیستمها بسیار خواهیم گفت.

علاوه بر این شبکه‌های کم سرعت، اکنون WAN های بیسیم پُر ظرفیت نیز در دست توسعه است. دسترسی پُر سرعت به اینترنت از منزل و دفتر کار بدون استفاده از خطوط تلفن، از اولین کاربردهای این شبکه‌هاست. استاندارد این سیستم (که به آن سرویس توزیع چند نقطه‌ای محلی گفته می‌شود) IEEE 802.16 نام دارد، که درباره آن در فصل ۴ بیشتر صحبت خواهیم کرد.



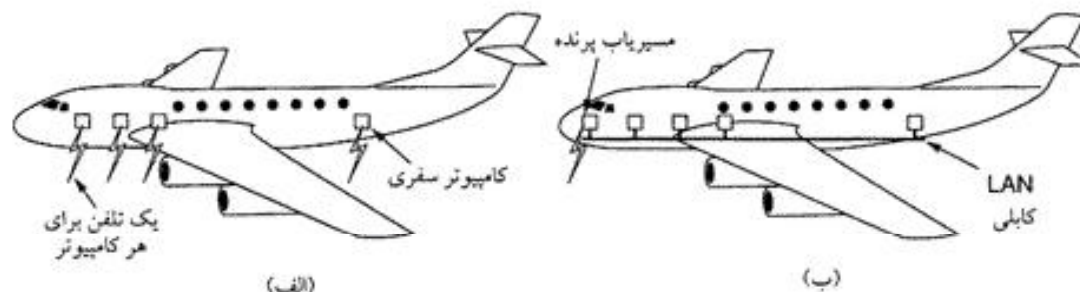
شکل ۱-۱۱. (الف) پیکربندی بلوتوث. (ب) LAN بیسیم.

تقریباً تمام شبکه‌های بیسیم باید در جایی به یک شبکه معمولی متصل شوند، تا بتوانند وظایف خود را انجام دهند؛ این کار را به روشهای مختلف می‌توان انجام داد. برای مثال، در شکل ۱-۱۲ الف هواپیمایی را می‌بینید، که در آن تعدادی از مسافران مودمهای خود را به تلفنهای تعبیه شده در صندلی‌های هواپیما وصل کرده‌اند؛ این تماسها بکلی از هم مستقلند. اما در شکل ۱-۱۲ ب روش بهینه‌تری را ملاحظه می‌کنید، که در آن هر صندلی یک اتصال اینترنت دارد، و مجموعه آنها تشکیل یک LAN معمولی می‌دهند؛ و این LAN به یک مسیر یاب بیسیم وصل است که ارتباط آنها را با دنیای خارج برقرار می‌کند.

بسیاری افراد فکر می‌کنند که بیسیم موج آینده است (برای مثال، Leeper, 2001؛ Bi et al., 2001؛ Varshey and Vetter, 2000)، ولی حداقل یک صدای مخالف هم وجود دارد. باب متکالف، مخترع اینترنت، می‌گوید: «کامپیوترهای موبایل بیسیم مثل دستشویی‌های متحرک هستند - آنها فقط بدرد پیک نیک، کارگاه‌های ساختمانی و اردوهای کوتاه مدت می‌خورند. نصیحت من اینست که در خانه خود کابل اینترنت بکشید، و منتظر آینده بمانید» (Metcalf, 1995). شاید تاریخ بعدها این اظهار نظر را در کنار جمله تاریخی تی جی واتسون رنيس IBM در سال ۱۹۴۵ بگذارد، که در پاسخ اینکه چرا IBM وارد بازار کامپیوتر نمی‌شود، گفته بود: «دنیا تا سال ۲۰۰۰ به چهار یا پنج کامپیوتر بیشتر نیاز نخواهد داشت».

۵-۲-۱ شبکه‌های خانگی (Home Network)

از هم اکنون می‌توان شبکه‌های خانگی را در افق آینده دید. ایده اصلی آن است که تمام یک وسایل یک خانه بتوانند



شکل ۱-۱۲. (الف) کامپیوترهای سفری منفرد. (ب) یک LAN پرنده.

با یکدیگر ارتباط برقرار کنند، و بتوان آنها را از طریق اینترنت کنترل کرد. این یکی از آن چیزهایی است که هیچکس متظر آن نبوده (مانند، کنترل از راه دور تلویزیون و تلفن همراه)، ولی وقتی آمد دیگر هیچکس نمی تواند زندگی بدون آن را تصور کند.

وسایل زیادی را می توان در یک شبکه به هم متصل کرد، که از واضحترین آنها می توان به موارد زیر اشاره کرد:

۱. کامپیوترها (رومیزی، سفری، PDA، وسایل جانبی)
۲. وسایل سرگرمی (تلویزیون، DVD، ویدئو، دوربین دیجیتالی، استریو، MP3)
۳. وسایل مخابراتی (تلفن معمولی و همراه، فکس، دستگاههای ارتباط داخلی)
۴. لوازم خانگی (میکروویو، یخچال، ساعت، بخاری، تهویه مطبوع، چراغ)
۵. وسایل اندازه گیری از راه دور (آلارم دود یا دزدی، قرائت کنتور، ترموستات، دوربین اتاق بچه)

شبکه های خانگی به بسیاری از خانه ها راه یافته است؛ در این خانه ها وسایلی وجود دارد که یک ارتباط پرسرعت اینترنت را بین چند کامپیوتر به اشتراک می گذارند. با گسترش سرگرمی ها روی اینترنت، بزودی شاهد تلویزیونها و استریوهایی خواهیم بود که مستقیماً به اینترنت متصلند (و این ارتباط دو جانبه خواهد بود، چرا که شاید شما هم مایل باشید فیلمها و موزیکهای خود را با دوستان و آشنایان به اشتراک بگذارید). مخابرات بین المللی از هم اکنون یک کالای در دسترس است، ولی بزودی این سرویسها بصورت دیجیتالی و از طریق اینترنت ارائه خواهند شد. امروزه کمتر خانه ای را می توان یافت که یک دوجین ساعت نداشته باشد، و با آمدن بهار و پائیز صاحبخانه مجبور است آنها را دستی جلو یا عقب بکشد؛ اگر تمام این ساعتها به اینترنت متصل باشند، می توان آنها را بصورت خودکار تنظیم کرد. کنترل خانه و مشاهده اتفاقاتی که در غیبت ما در آن می گذرد، یکی از آرزوهای دیرینه ماست، که اینک به واقعیت تبدیل شده است. (دیگر می توانید با خیال راحت به سینما بروید، و بچه ها در خانه تنها بگذارید!) شاید فکر کنید هر یک از این کاربردها به شبکه ای مجزا نیاز دارد، اما یکپارچه کردن آنها احتمالاً ایده بهتری است.

شبکه های خانگی تفاوت های ذاتی با سایر انواع شبکه دارد. اول اینکه نصب آن نباید پیچیده باشد. آنهایی که در این سالها درگیر کار نصب شبکه بوده اند، با جوابهای زیر (وقتی یک مشتری با مشکلی مواجه شده و به شما تلفن می زند) کاملاً آشنا هستند: (۱) دفترچه راهنما را با دقت بخوانید، (۲) کامپیوتر را دوباره بوت کنید، (۳) تمام سخت افزارها و نرم افزارهای اضافی - آنهایی که مال شرکت ما نیست! - را حذف کنید، (۴) جدیدترین درایور را از سایت وب ما بردارید و نصب کنید، و بالاخره وقتی هیچکدام از این کارها فایده ای نبخشد، (۵) کامپیوتر را فرمت و ویندوز را از نو نصب کنید. اگر به کسی که یک یخچال اینترنتی خریده، بگوئید آخرین ویرایش سیستم عامل یخچال اینترنتی را بار کرده و نصب کند، مسلماً باعث خوشحالی وی نخواهد شد! آنهایی که کامپیوتر می خرند، عادت دارند با سیستمهایی سر و کله بزنند که کار نمی کند، ولی خریداران اتومبیل، تلویزیون و یخچال این حرفها سرشان نمی شود؛ آنها انتظار دارند وسیله ای که خریده اند از اول بسماله بدون هیچ مشکلی کار کند.

دوم اینکه شبکه های خانگی باید بتوانند تحت هر شرایطی کار کنند. حتی همین حالا هم یک دستگاه کولر گازی با فقط چهار دکمه LOW، MEDIUM، HIGH، OFF، یک دفترچه راهنمای ۳۰ صفحه ای دارد. تصورش را بکنید وقتی این کولر اینترنتی شود، فقط قسمت امنیت آن دستکم ۳۰ صفحه خواهد بود! و این از حد تحمل قسمت اعظم کاربران این قبیل دستگاهها خارج است.

قیمت پائین سومین عامل برای موفقیت شبکه های خانگی است. خیلی از افراد حاضر نیستند ۵۰ دلار پول اضافه برای یک ترموستات اینترنتی بدهند، که چپ، فقط درجه حرارت خانه خود را از اداره چک کنند! حتی ۵ دلار را هم شاید خیلی ها بزور بدهند.

از آنجائیکه دنیای آینده دنیای چند رسانه ایست، عامل چهارم در مقبولیت شبکه های خانگی بالا بودن پهنای باند آنهاست. مطمئن باشید هیچکس یک تلویزیون اینترنتی که فیلمها را با وضوح 320×240 پیکسل و ۱۰ فریم در ثانیه نشان می دهد، نخواهد خرید. حتی اینترنت (که در اغلب ادارات از آن استفاده می شود) نیز مناسب چند رسانه ای نیست. شبکه های خانگی برای آن که بتوانند فراگیر شوند، باید پهنای باند بیشتر را با قیمتی کمتر ارائه کنند.

پنجم، امکان گسترش شبکه های خانگی است. این به معنای آن است که جنگ استانداردها باید یک بار و برای همیشه حل شود. اگر امروز به مشتریان خود توصیه کنید وسایل IEEE 1394 (که به فایر وایر - FireWire - معروف است) را بخرند، و سال آینده بگوئید امسال مد USB 2.0 است، مطمئن باشید همه آنها را پرانده اید. استانداردهای ارتباطی باید سالها (و کابل کشی، دهها سال) ثابت و بدون تغییر باقی بماند.

امنیت و قابلیت اعتماد ششمین عامل موفقیت شبکه های خانگی است. اینکه چند تا ایمیل و پروسی از دست بدهید یک چیز است، و به باد رفتن تمام خانه در اثر لو رفتن کدهای امنیتی آن، یک چیز دیگر.

در این میان یکی از سوالات جالب اینست که شبکه های خانگی باید بیسیم باشند یا باسیم. همین حالا هم در اغلب خانه ها شش شبکه وجود دارد: شبکه برق، تلفن، تلویزیون کابلی، آب، گاز و فاضلاب. اضافه کردن شبکه هفتم به خانه های نوساز هزینه چندانی ندارد، ولی در خانه های موجود کاری پُرخرج خواهد بود. شبکه های بیسیم کم خرج تر هستند، ولی شبکه های کابلی از نظر امنیتی بسیار قابل اعتمادترند. امواج رادیویی بسادگی از محدوده خانه خارج می شوند، و شاید یک همسایه فضول بتواند ایمیل های شما را بخواند، و یا دزدکی از اشتراک اینترنت شما استفاده کند. در فصل ۸ درباره روشهای رمزنگاری برای مقابله با این مشکلات صحبت خواهیم کرد، ولی در شبکه های خانگی موضوع امنیت از اهمیت چندگانه ای برخوردار است.

بطور خلاصه، شبکه های خانگی علاوه بر امکاناتی که ارائه می کنند، چالشهایی را نیز با خود به همراه دارند. اغلب این چالشها به مدیریت ساده، قابلیت اعتماد، ایمنی بالا (بویژه در مورد کاربران غیر حرفه ای)، کارایی بالا و قیمت پائین مربوط می شوند.

۶-۲-۱ شبکه شبکه ها (Internetwork)

شبکه های متعددی با نرم افزارها و سخت افزارهای بسیار مختلف در سراسر دنیا وجود دارد، و بسیار پیش می آید که کاربری از یک شبکه بخواهد با کاربران شبکه های دیگر ارتباط برقرار کند. برای انجام این خواسته بایستی شبکه های مختلف (که بعضاً با هم ناسازگار هم هستند) با وسایلی بنام دروازه gateway - که می تواند سخت افزاری یا نرم افزاری باشد - به هم متصل شده، و داده ها از فرمتی به فرمت دیگر تبدیل شود. به مجموعه ای از این شبکه های بهم پیوسته شبکه شبکه ها (internetwork یا internet) گفته می شود. کلمه internet وقتی با i نوشته می شود، معنای عام می دهد، ولی با I همان شبکه جهانی اینترنت از آن مستفاد می شود.

متداولترین شکل شبکه شبکه ها عبارتست از تعدادی LAN که با ارتباطات WAN به هم متصل شده اند. در حقیقت، اگر در شکل WAN ۹-۱ را به جای subnet (زیر شبکه) قرار دهیم، هیچ چیز تغییر نخواهد کرد. در این مورد تنها تفاوت تکنیکی بین WAN و subnet وجود یا عدم وجود کامپیوترهای میزبان (host) است: اگر سیستم ناحیه خاکستری فقط از مسیر یاب تشکیل شده باشد، این یک زیر شبکه است؛ اگر در آن میزبان هم وجود داشته باشد، یک WAN است. تفاوت واقعی در مالکیت و طرز استفاده است.

اغلب افراد مفاهیم زیر شبکه (subnet)، شبکه (network) و شبکه شبکه ها (internetwork) را با هم اشتباه می کنند. زیر شبکه (که بیشتر در شبکه های گسترده مفهوم پیدا می کند) مجموعه ایست از مسیر یابها و خطوط محابراتی متعلق به راهبر شبکه. برای مقایسه، شبکه تلفن شهری از یک سری مراکز سوئیچینگ، خطوط مخابراتی

پُرسرعت، و خطوط کم سرعت که مشترکان را به مرکز وصل می کند، تشکیل می شود. خطوط پُرسرعت و مراکز سونیچینگ (که متعلق به شرکت مخابرات هستند) همان زیرشبکه را می سازند. تلفنهای مشترکان که متعلق به افراد (و معادل میزبان) است، جزئی از زیرشبکه نیستند. مجموعه این دو (زیرشبکه و میزبانها) شبکه را می سازد. در شبکه های LAN (که فقط کامپیوتر و کابل است) زیرشبکه وجود ندارد.

وقتی چند تا از این شبکه ها به هم متصل می شوند، یک شبکه شبکه ها شکل می گیرد. با این تعریف، مجموعه دو LAN، یا یک LAN و یک WAN، را هم می توان شبکه شبکه ها نامید، ولی در صنعت کامپیوتر توافق یکپارچه ای درباره این اصطلاحات وجود ندارد. بعنوان یک قاعده کلی، اگر چند شرکت یا سازمان مختلف هر یک (به هزینه خود) قسمتی از یک شبکه را بنا کنند، با یک شبکه شبکه ها سروکار داریم. همچنین اگر تکنولوژی زیربنایی در قسمتهای مختلف متفاوت باشد، به احتمال زیاد دو شبکه داریم.

۳-۱ نرم افزار شبکه

در اولین شبکه های کامپیوتری سخت افزار از اهمیت ویژه ای برخوردار بود، و به نرم افزار فقط بعنوان چیزی که باید بعداً به آن فکر می شد، نگاه می کردند. اما این استراتژی دیگر کارایی ندارد. امروزه نرم افزار شبکه بسیار ساخت یافته است، که در این قسمت آنرا بررسی می کنیم. روشهای مورد بحث در این قسمت سنگ بنای کتاب را تشکیل می دهند، و در آینده بسیار به آنها مراجعه خواهیم کرد.

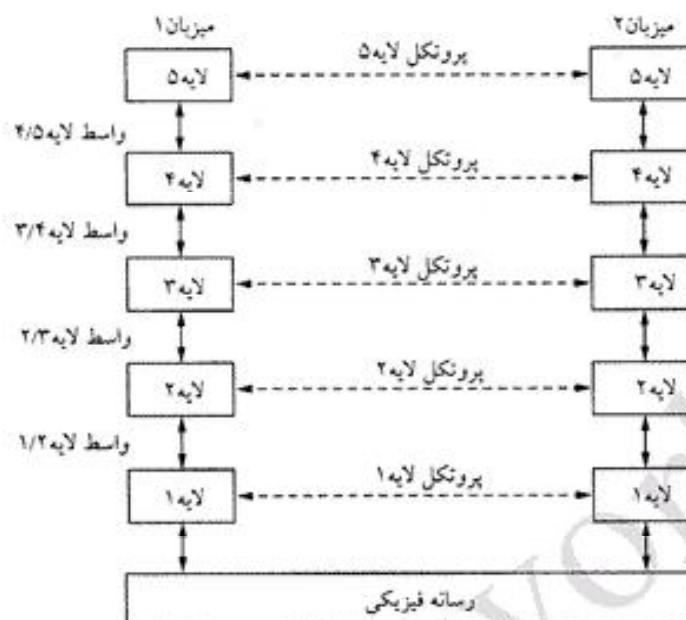
۱-۳-۱ سلسله مراتب پروتکل ها

برای کاهش پیچیدگیهای طراحی، اغلب شبکه ها بصورت مجموعه ای از چند لایه (layer) یا سطح (level) - که هر کدام روی دیگری قرار می گیرند - طراحی می شوند. تعداد لایه ها، نام هر لایه، محتوای آن، و کاری که هر لایه انجام می دهد، از شبکه ای به شبکه دیگر متفاوت است. وظیفه هر لایه ارائه سرویسهای خاص به لایه های بالاتر، و پنهان کردن جزئیات کار از دید آنهاست. در این مفهوم، هر لایه یک ماشین مجازی (virtual machine) است که سرویسهای خاصی را در اختیار لایه های بالاتر می گذارد.

این یکی از مفاهیم آشنا و کلیدی در کلیه علوم کامپیوتری است، که با نامهایی از قبیل پنهان کردن اطلاعات (information hiding)، انواع داده مجرد (abstract data types)، کپسولی کردن داده ها (data encapsulation) و برنامه نویسی شیءگرا (object-oriented programming) شناخته می شود. ایده اصلی این است که یک قطعه نرم افزار (یا سخت افزار) سرویسی را به کاربران خود عرضه کند، ولی جزئیات کار (از قبیل حالت داخلی خود و الگوریتمهای بکار رفته) را از آنها مخفی نگه دارد.

لایه n یک ماشین همیشه با لایه n ماشین دیگر حرف می زند. قواعد و قراردادهای این ارتباط را پروتکل لایه n (layer n protocol) می نامند. در ساده ترین حالت، پروتکل (protocol) عبارتست از قراردادهای توافق شده بین دو طرف برای برقراری و پیشبرد یک ارتباط. بعنوان مقایسه، وقتی یک خانم به یک آقا معرفی می شود، آن خانم می تواند دستش را پیش بیاورد؛ و بنوبه خود آن آقا هم می تواند دست خانم را (اگر یک شاهزاده اروپایی در یک میهمانی رسمی باشد) ببوسد، یا فقط با او دست بدهد (اگر آن خانم یک وکیل آمریکایی در یک جلسه کاری باشد). سرپیچی از پروتکل ها برقراری ارتباط را بسیار دشوار (اگر نگوئیم، غیرممکن) خواهد کرد.

در شکل ۱-۱۳ یک شبکه پنج لایه به تصویر کشیده شده است. به اجزایی که در یک لایه هستند، همتا (peer) گفته می شود. این همتاها می توانند پروسس های نرم افزاری، وسایل سخت افزاری، و یا حتی دو انسان باشند. بعبارت دیگر، این همتاها هستند که با استفاده از پروتکل با هم رابطه برقرار می کنند.



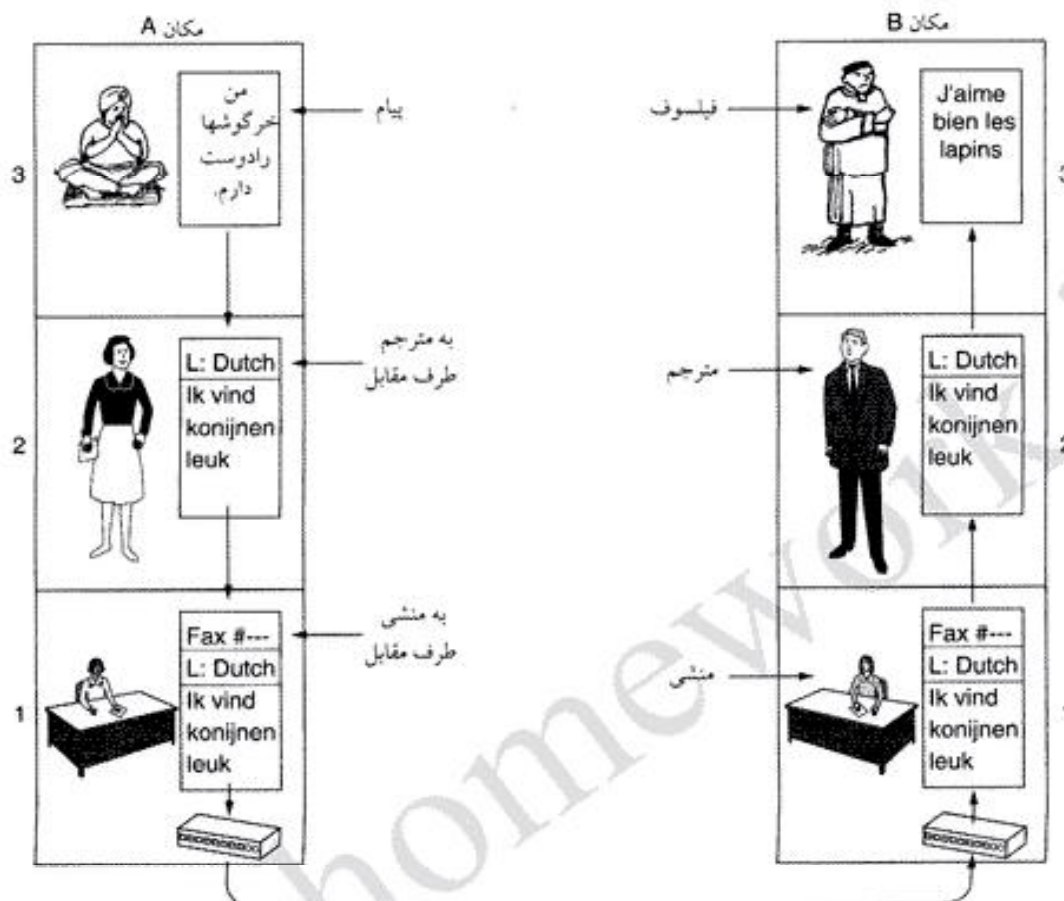
شکل ۱-۱۳. لایه‌ها، پروتکل‌ها، واسط‌ها.

در حقیقت، داده‌ها هرگز مستقیماً از لایه ۱ یک ماشین به لایه ۱ ماشین دیگر منتقل نمی‌شوند. بلکه، هر لایه داده‌ها (و اطلاعات کنترلی) را به لایه زیرین خود می‌دهد، تا به پایین‌ترین لایه برسد. در زیر پایین‌ترین لایه (لایه ۱) رسانه فیزیکی (physical medium) قرار دارد، که داده‌ها را جابجا می‌کند. در شکل ۱-۱۳ ارتباط مجازی لایه‌ها با خط چین و ارتباط واقعی و فیزیکی با خط ممتد نشان داده شده است.

بین هر زوج از لایه‌های مجاور واسط (interface) قرار دارد. واسط مشخص می‌کند که هر لایه چه سرویس‌ها و عملکردهای پایه‌ای در اختیار لایه بالاتر می‌گذارد. تعریف واسط‌های مناسب از مهمترین وظایف طراحان شبکه است. لازمه این امر آنست که وظایف هر لایه دقیقاً مشخص و شناخته شده باشد. علاوه بر به حداقل رساندن اطلاعات رد و بدل شده بین لایه‌ها، یک واسط شسته و رفته کار عوض کردن پیاده‌سازی لایه‌ها را نیز آسان می‌کند، چون تنها کاری که باید کرد این است که پیاده‌سازی جدید دقیقاً همان سرویس‌های پیاده‌سازی قدیمی را به همسایگان خود ارائه کند. در حقیقت، پیاده‌سازی‌های متعددی در شبکه‌های مختلف وجود دارد، که هیچ خللی در ارتباط لایه‌ها ایجاد نمی‌کنند.

به مجموعه لایه‌ها و پروتکل‌ها معماری شبکه (network architecture) می‌گویند. مشخصه‌های یک معماری باید آنچنان دقیق و جامع باشد که طراح شبکه بتواند نرم‌افزارها و سخت‌افزارهای لازم برای کارکرد صحیح آنرا فراهم آورد. جزئیات پیاده‌سازی و مشخصات واسط‌ها هرگز جزء معماری شبکه نیست، چون آنها باید در دل ماشین مخفی باشند (و از خارج دیده نشوند). حتی لازم نیست واسط‌ها در تمام ماشینهای یک شبکه یکسان باشند، مشروط باینکه تمام این ماشینها بتوانند از تمام پروتکل‌ها استفاده کنند. به مجموعه پروتکل‌هایی که در یک سیستم خاص بکار می‌روند (یک پروتکل در هر لایه)، پشته پروتکل (protocol stack) گفته می‌شود. پروتکل، پشته پروتکل و معماری شبکه از مهمترین موضوعات مورد بحث این کتاب هستند.

شاید یک مثال بتواند در روشن شدن مفهوم ارتباط چند لایه کمک کند. دو فیلسوف را، که اولی فقط زبانهای اردو و انگلیسی می‌داند و دیگری فقط چینی و فرانسه، در نظر بگیرید (آنها معادل پروسسهای همتا در لایه ۳ هستند). از آنجائیکه این دو فیلسوف نمی‌توانند مستقیماً با هم حرف بزنند، دو مترجم استخدام می‌کنند



شکل ۱-۱۴. معماری فیلسوف-مترجم-منشی.

(پروسسهای همتا در لایه ۲)، که آنها هم بنویسند خود هر کدام یک منشی دارند (پروسسهای همتا در لایه ۱). فیلسوف ۱ میل دارد علاقه خود به *oryctolagus cuniculus* را به فیلسوف ۲ (همتای خود) ابلاغ کند. برای اینکار، از طریق واسط ۲/۳ یک پیام بزرگان انگلیسی با مضمون "I like rabbits" به مترجم خود می فرستد (شکل ۱-۱۴ را ببینید). مترجم ها بین خود توافق کرده اند که به زبان هلندی حرف بزنند، پس پیام فوق به "Ik vind konijnen leuk" تبدیل می شود. انتخاب زبانی که همتاهای لایه ۲ با آن صحبت کنند، بر عهده خود آنان (پروسسهای این لایه) است.

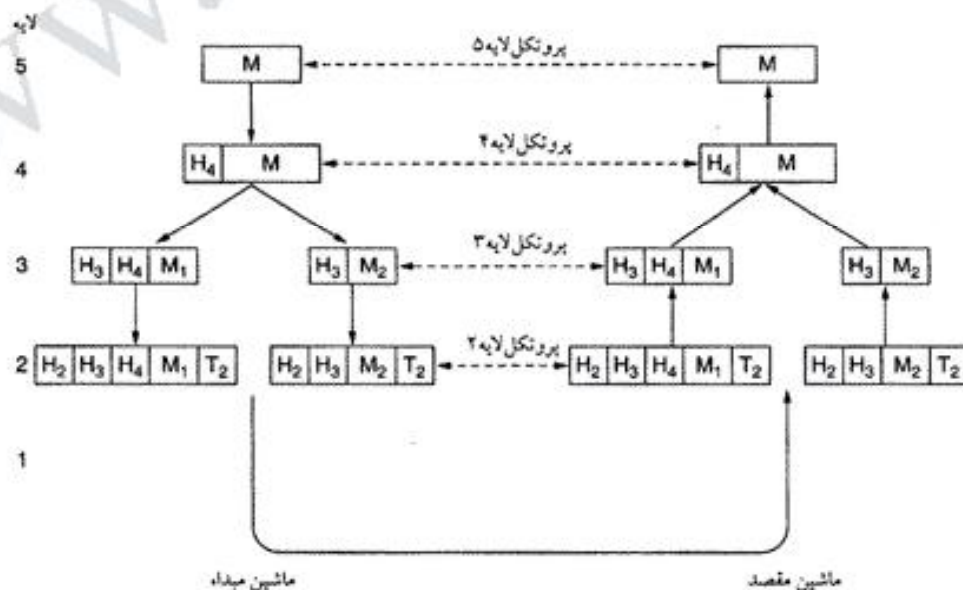
مترجم، سپس، این پیام را به منشی خود می دهد تا مثلاً از طریق فکس (پروتکل لایه ۱) ارسال شود. وقتی این پیام به منشی سمت مقابل رسید، آنرا به مترجم (لایه ۲) تحویل داده، و مترجم نیز پس از ترجمه به زبان فرانسه آنرا، از طریق واسط ۲/۳، به فیلسوف ۲ می دهد. توجه کنید که تا زمانیکه واسط ها تغییر نکرده باشند، پروتکل ها کاملاً از یکدیگر مستقل هستند. برای مثال، مترجم ها می توانند زبان توافقی خود را به فنلاندی تغییر دهند، مشروط باینکه واسط آنها با لایه های ۱ و ۳ هیچ تغییری نکند. یا اینکه منشی ها می توانند از فکس به ایمیل یا تلفن سوئیچ کنند، بدون اینکه این کار هیچ تأثیری روی لایه های دیگر بگذارد. هر پروسس می تواند اطلاعات دلخواه خود را (که فقط پدرد پروسس همتای آن می خورد) به پیام اضافه کند. این اطلاعات به لایه های بالاتر منتقل نخواهند شد. حال یک مثال فنی تر را در نظر می گیریم: نحوه ارتباط در یک شبکه پنج لایه (شکل ۱-۱۵ را ببینید). برنامه ای

که در لایه ۵ اجرا می شود، یک پیام (M) تولید کرده و برای ارسال به لایه ۴ می دهد. لایه ۴ یک سرآیند (header) به این پیام اضافه و آنرا به لایه ۳ تحویل می دهد. سرآیند حاوی اطلاعات کنترلی بین لایه های متناظر است؛ برای مثال، لایه ۴ می تواند به هر پیام یک عدد ترتیبی نسبت بدهد، تا اگر لایه های پائینتر آنها را بدون نظم و ترتیب ارسال کردند، لایه متناظر در سمت مقابل بتواند آنها را به ترتیب صحیح بازبینی کند. در برخی از لایه ها، این سرآیندها حاوی اندازه بسته، زمان ارسال و اطلاعاتی از این قبیل است.

در بسیاری از شبکه اندازه پیام در لایه ۴ هیچ محدودیتی ندارد، ولی (تقریباً همیشه) این محدودیت در لایه ۳ وجود دارد. در نتیجه، لایه ۳ باید پیام را به قطعات کوچکتر بشکند، و به هر قطعه یک سرآیند لایه ۳ اضافه کند. در این مثال، پیام M به دو قطعه M_1 و M_2 شکسته شده است.

سپس، لایه ۳ این بسته ها را به لایه بعدی (یعنی لایه ۲) تحویل می دهد. لایه ۲ علاوه بر اضافه کردن سرآیند خاص خود به ابتدای هر بسته، به انتهای آنها نیز یک دنباله (trailer) چسبانده، و آنها را به لایه بعدی (که لایه انتقال فیزیکی است) می دهد. در سمت مقابل، ماشین که این بسته ها را دریافت می کند، آنها را لایه به لایه بالا می فرستد، و هر لایه (قبل از تحویل به لایه بالاتر) اطلاعات خاص خود را از بسته ها برمی دارد. بدین ترتیب، هیچ لایه ای سرآیندها و دنباله های لایه های زیرین خود را دریافت نخواهد کرد.

نکته مهمی که در شکل ۱-۱۵ باید بدان توجه کنید، رابطه ارتباطات مجازی و حقیقی بین ماشینها، و تفاوت پروتکل ها و واسطه ها است. برای مثال، پروتکل های همتا در لایه ۴ فکر می کنند که ارتباط بین آنها یک ارتباط «افقی» است، که با استفاده از پروتکل لایه ۴ برقرار می شود. هر یک از این پروتکل ها احتمالاً ابزارهایی بنام *SendToOtherSide* (برای ارسال پیام به سمت مقابل) یا *GetFromOtherSide* (برای دریافت پیام از سمت مقابل) نیز دارند، و حتی نمی دانند که پیامهای خود را از طریق واسطه ۳/۴ به لایه پائین تر می دهند، نه به طرف مقابل. تجزیه ی بودن رابطه پروتکل های همتا یک نکته کلیدی در طراحی شبکه است. با این تمهید، کار بسیار پیچیده و دشواری مانند طراحی کامل یک شبکه به کارهای کوچکتر و ساده تری (مانند طراحی لایه های جداگانه) شکسته می شود.



شکل ۱-۱۵. انتقال اطلاعات در یک شبکه پنج لایه.

با اینکه نام این بخش «نرم افزار شبکه» است، اما باید متوجه باشید که لایه های پائین تر معمولاً بصورت سخت افزاری پیاده سازی می شوند. با این حال، اینها نیز الگوریتمهای پیچیده نرم افزاری هستند، که فقط در سخت افزار حک و ثابت شده اند.

۲-۳-۱ ملاحظات در طراحی لایه ها

برخی از مفاهیم کلیدی طراحی شبکه های کامپیوتری در لایه های مختلفی حضور دارند، که در زیر برخی از مهمترین آنها را بطور مختصر بررسی خواهیم کرد.

هر لایه به مکانیزمی برای شناسایی فرستنده (sender) و گیرنده (receiver) نیاز دارد. از آنجائیکه یک شبکه معمولاً تعداد زیادی کامپیوتر دارد، و در هر کامپیوتر پروسس های متعددی در حال اجرا هستند، باید ابزاری وجود داشته باشد که هر پروسس بتواند پروسس همتای خود را دقیقاً شناسایی و مشخص کند. بعبارت دیگر، برای تعیین دقیق مقصد به یک نظام آدرس دهی (addressing) نیاز داریم.

نحوه انتقال داده ها نیز نیازمند قواعد و مقررات خاص خود است. در برخی از سیستمها داده ها فقط در یک جهت حرکت می کنند، اما در برخی دیگر در هر دو جهت. همچنین پروتکل باید تعیین کند که هر ارتباط فیزیکی معادل چند کانال منطقی است، و اولویت بندی آنها چگونه است. در بسیاری از شبکه ها هر ارتباط فیزیکی حداقل دو کانال منطقی را شامل می شود، یکی برای داده های معمولی و دیگری برای داده های اضطراری.

مسئله مهم دیگر کنترل خطا (error control) است، چون هیچ ارتباط فیزیکی صد در صد کامل و عاری از خطا نیست. گداهای بسیاری برای کشف و تصحیح خطا وجود دارد، ولی هر دو طرف باید بر سر یکی از آنها توافق کنند. همچنین گیرنده باید بتواند به طریقی به فرستنده اعلام کند که کدام پیامها را درست دریافت کرده و کدامها را غلط.

همانطور که قبلاً هم گفتیم، در بسیاری از موارد بسته های تشکیل دهنده یک پیام بصورت منظم ارسال نمی شوند؛ در این حالت، پروتکل باید طوری طراحی شده باشد که گیرنده بتواند قطعات پیام را دوباره بنحو صحیح بهم بچسباند. یک راه حل شماره گذاری قطعات (بسته های) پیام است، ولی باز این سؤال باقیست که گیرنده باید چگونه با این بسته ها عمل کند.

یکی از مسائل که در تمام لایه ها وجود دارد اینست که، فرستنده با چه سرعتی باید اطلاعات را ارسال کند تا گیرنده های گند در گرداب داده ها غرق نشوند. برای این مشکل نیز راه حل های مختلفی وجود دارد که بعداً مفصلاً درباره آنها توضیح خواهیم داد. برخی از این راه حل ها شامل نوعی فیدبک (بازخور) از گیرنده به فرستنده است، که در هر لحظه وضعیت گیرنده را (بطور مستقیم یا غیر مستقیم) به فرستنده اعلام می کند. در برخی دیگر، دو طرف از قبل بر سر نرخ انتقال اطلاعات توافق می کنند. این مبحث کنترل جریان (flow control) نام دارد.

مسئله دیگری که در لایه های متعدد باید حل شود اینست که، اغلب پروسسها قادر نیستند پیامهایی با هر طول دلخواه دریافت کنند. این وضعیت باعث شده تا مکانیزمهایی برای شکستن پیامها به قطعات کوچک، ارسال، و سپس مونتاژ آنها در مقصد ابداع شود. مشکل دیگری که از اینجا ناشی می شود آنست که برخی از پروسسها اصرار دارند پیامها را آنقدر ریز کنند، بگونه ای که کارایی کل سیستم را مختل می کند. در اینجا راه حل چسباندن چند قطعه به یکدیگر و شکستن دوباره آنها در مقصد است.

در اغلب موارد ایجاد کانالهای ارتباطی جداگانه برای هر زوج پروسس کاری پرهزینه (و گاهی غیرممکن) است. در این موارد، لایه های پائین تر برای برقراری ارتباط بین چند پروسس مستقل، از یک کانال استفاده می کنند. این عمل (که مالتی پلکس - multiplexing - و دمالتی پلکس - demultiplexing - نام دارد) معمولاً بصورت شفاف انجام می شود، که در اینصورت می توان آنرا در هر لایه ای پیاده سازی کرد. برای مثال، وقتی تعداد خطوط

مخابراتی موجود محدود است، در لایه فیزیکی از تکنیکهای مالتی پلکس استفاده می شود. وقتی بین مبدأ و مقصد مسیرهای مختلفی وجود دارد، یکی از آنها باید انتخاب شود. گاهی تصمیم گیری در این مورد باید در چند لایه انجام شود. مثلاً، برای ارسال اطلاعات از لندن به رُم اولاً (در لایه های بالاتر) باید تصمیم بگیریم که از مسیر فرانسه استفاده کنیم یا از مسیر آلمان؛ سپس در لایه های پایینتر انتخاب مسیرهای خلوت تر از میان مسیرهای موجود پیش می آید. به این مبحث مسیریابی (routing) گفته می شود.

۳-۳-۱ سرویسهای اتصال-گرا و غیر متصل

هر لایه می تواند دو نوع سرویس در اختیار لایه بالاتر از خود بگذارد: سرویس اتصال-گرا (connection-oriented) و سرویس غیر متصل (connectionless). در این قسمت این سرویس ها و تفاوت های آنها را بررسی خواهیم کرد.

سرویس اتصال-گرا بر اساس مدل سیستمهای تلفن کار می کند. وقتی می خواهید با یک نفر تماس بگیرید، گوشی تلفن را برداشته، شماره می گیرید، صحبت می کنید، و بعد گوشی را می گذارید. در یک سرویس اتصال-گرا هم ابتدا اتصال برقرار شده، و بعد از تبادل اطلاعات مورد نظر، اتصال قطع می شود. مهمترین نکته در مورد سرویسهای اتصال-گرا اینست که آنها مانند یک لوله عمل می کنند: فرستنده از یک طرف داده ها (پیست ها) را به داخل لوله می فرستند، و گیرنده در طرف دیگر آنها را می گیرد. در اغلب موارد داده ها بهمان ترتیبی که فرستاده شده اند، دریافت می شوند.

در برخی موارد بعد از برقراری اتصال، فرستنده، گیرنده و زیر شبکه ابتدا یک سری مذاکرات اولیه (negotiation) انجام می دهند تا بر سر مواردی از قبیل حداکثر اندازه پیامها، کیفیت سرویس مورد نظر، و مانند آن توافق کنند. معمولاً، یک طرف پیشنهادی می دهد و طرفهای دیگر آنرا قبول یا رد کرده، و یا بکلی پیشنهاد جدیدی ارائه می کنند.

از سوی دیگر، سرویس غیر متصل بر اساس مدل پست بنا شده است. هر پیام (نامه) دارای آدرس مشخصی است، و مسیری که برای رسیدن به مقصد طی می کند، کاملاً مستقل از پیامهای دیگر است. معمولاً وقتی دو نامه به یک مقصد می فرستید، اولین نامه زودتر از دومی به آنجا می رسد؛ ولی گاهی پیش می آید که اولی با تأخیر و بعد از دومی به مقصد برسد.

برای هر سرویس می توان یک کیفیت سرویس (quality of service) در نظر گرفت. برخی از سرویسها مطمئن و قابل اعتماد هستند، بگونه ای که هیچ داده ای در حین انتقال از بین نمی رود. یک سرویس قابل اعتماد معمولاً بگونه ای طراحی می شود که گیرنده دریافت صحیح داده ها را به فرستنده اعلام کند. این تصدیق دریافت (acknowledgement) باعث تحمیل یک بار اضافی و تأخیر در انتقال پیامها می شود، که اغلب ارزش آنرا دارد، ولی گاهی به زحمتش نمی ارزد.

انتقال فایل (file transfer) از جمله مواردیست که به یک سرویس مطمئن اتصال-گرا نیاز دارد؛ صاحب فایل معمولاً میل دارد تمام بیت های فایلش (با همان نظم و ترتیب) به مقصد برسد. کمتر کسی را پیدا می کنید که راضی شود یک فایل قروقاطی دریافت کند، حتی اگر اینکار به معنای سرعت بیشتر باشد.

سرویس اتصال-گرای قابل اعتماد بر دو گونه مختلف است: توالی پیام (message sequence) و جریان بایت (byte stream). در سرویس توالی پیام حد و مرز پیامها همیشه حفظ می شود: وقتی دو پیام ۱۰۲۴ بایتی می فرستید، طرف مقابل همیشه دو پیام ۱۰۲۴ بایتی دریافت خواهد کرد، نه یک پیام ۲۰۴۸ بایتی، یا چهار پیام ۵۱۲ بایتی. اما در سرویس دوم، چیزی بنام حد و مرز پیام وجود ندارد، و فقط جریانی از بایتها دیده خواهد شد. در این حالت وقتی ۲۰۴۸ بایت به مقصد می رسد، به هیچ طریقی نمی توان گفت که آیا این یک پیام ۲۰۴۸ بایتی بوده، یا دو

پیام ۱۰۲۴ بایتی، و یا ۲۰۴۸ پیام ۱ بایتی. برای مثال، اگر بخواهید صفحات یک کتاب را به یک دستگاه حروفچینی الکترونیکی بفرستید، شاید برایتان مهم باشد که حد و مرز هر صفحه مشخص باشد. از طرف دیگر، وقتی از راه دور به یک کامپیوتر متصل می شوید، جریان بایت ها از کامپیوتر مبدأ به مقصد تمام آن چیز است که نیاز دارید، و حد و مرز پیامها هیچ اهمیتی ندارد.

اما همانطور که گفتیم، در برخی از موارد تأخیری که در نتیجه تصدیق دریافت (acknowledgement) پدید می آید، غیر قابل قبول است؛ مکالمه دیجیتالی یکی از این موارد است. اغلب کاربران ترجیح می دهند گاهی صدای طرف مقابل را با نویز بشنوند، تا اینکه (در نتیجه مکانیزم تصدیق دریافت) مکالمه با تأخیر و وقفه انجام شود. و یا در کنفرانسهای ویدئویی کمی برفک و نویز قابل تحمل تر است، تا اینکه تصاویر با پرشهای اعصاب خردکن دریافت شود.

از طرف دیگر، همه کاربردها به ارتباط متصل نیاز ندارند؛ پُست الکترونیک (ایمیل) یکی از مواردیست که نیازی به سرویس اتصال گرا ندارد، و بویژه در مواقعی که هزینه عاملی تعیین کننده است، سرویس قابل اعتماد نیز چندان الزامی نیست. در این موارد فقط کافیست اولویت تحویل پیام بالا باشد. سرویس غیر متصل غیر قابل اعتماد (سروسی که به تصدیق دریافت از طرف مقابل متکی نیست) اغلب بعلت شباهتی که با سیستم تلگراف دارد، سرویس دیتاگرام (datagram service) نامیده می شود.

اما گاهی با اینکه نیازی به برقراری یک اتصال نیست (مثلاً برای ارسال یک پیام کوتاه)، ولی قابل اعتماد بودن ارتباط اهمیت دارد. در این قبیل موارد می توان از سرویس دیتاگرام همراه با تصدیق دریافت (acknowledged datagram service) استفاده کرد. این مانند پُست کردن یک نامه سفارشی است، که فرستنده می خواهد از رسیدن نامه بدست گیرنده مطمئن شود. با دریافت این تأییدیه، فرستنده مطمئن می شود که نامه گم نشده و بدست طرف مقابل رسیده است.

سرویس دیگری نیز وجود دارد، که سرویس درخواست-پاسخ (request-reply service) نام دارد. در این سرویس، فرستنده دیتاگرامی که حاوی یک درخواست است ارسال می کند، و پاسخ آنرا می گیرد. برای مثال، وقتی پیامی به کتابخانه محلی می فرستید و سؤال می کنید که «زبان سواحیلی در کدام کشور تکلم می شود»، از این سرویس استفاده می کنید. از سرویس درخواست-پاسخ معمولاً در سیستمهای مشتری-سرویس دهنده استفاده می شود: مشتری درخواست خود را به سرویس دهنده فرستاده، و پاسخ آنرا دریافت می کند. در شکل ۱-۱۶ خلاصه ای از سرویسهای توضیح داده شده در این قسمت را مشاهده می کنید.

		مثال	سرویس
		چند صفحه متوالی	استریم پیام قابل اعتماد
اتصال گرا		ورود از راه دور	استریم بایت قابل اعتماد
		صدای دیجیتالی	اتصال غیر قابل اعتماد
			دیتا گرام غیر قابل اعتماد
غیر متصل		ایمیل ثبت شده	دیتا گرام تصدیق شده
		جستجوی پایگاه داده	درخواست - پاسخ

شکل ۱-۱۶. شش نوع سرویس مختلف.

شاید در نگاه اول تعجب کنید که اصولاً چرا باید از یک سرویس غیر قابل اعتماد استفاده کنیم، و اصلاً چه کسی چنین چیزی را لازم دارد؟ اول از همه اینکه، امکان دارد در مواردی سرویس قابل اعتماد اساساً در دسترس نباشد. برای مثال، اینترنت یک ارتباط قابل اعتماد نیست، و گاهی ممکنست داده‌ها در حین انتقال صدمه ببینند؛ این بر عهده پروتکل‌های لایه‌های بالاتر است که این مشکل را حل کنند. دوم اینکه، تأخیر ذاتی سرویسهای مبتنی بر تصدیق دریافت در مواردی (مانند برنامه‌های چند رسانه‌ای) پذیرفتنی نیست. به این دلایل، وجود سرویسهای قابل اعتماد و غیر قابل اعتماد هر دو لازم است.

۴-۳-۱ عملکردهای پایه سرویس

هر سرویس با یک سری عملکردهای پایه (primitives) که در اختیار کاربر خود می‌گذارد، شناخته می‌شود. این عملکردهای پایه یا خود کاری را انجام می‌دهند، و یا انجام کاری را در طرف مقابل گزارش می‌کنند. اگر پشته پروتکل (protocol stack) جزئی از سیستم عامل باشد (که اغلب نیز چنین است)، عملکردهای پایه نیز معمولاً جزء فراخوانی‌های سیستم (system call) هستند. این فراخوانی‌ها باعث فعال شدن کُدی در هسته سیستم عامل شده، و ارسال بسته‌های پیام انجام می‌شود.

عملکردهای پایه هر سرویس به خصلت آن سرویس بستگی دارد. برای مثال، عملکردهای پایه یک سرویس اتصال-گرافاوت از سرویسهای غیر متصل است. در شکل ۱-۱۷ حداقل عملکردهای پایه لازم برای پیاده‌سازی یک سرویس اتصال-گرای جریان بایت قابل اعتماد را در یک محیط مشتری-سرویس دهنده ملاحظه می‌کنید. طرز استفاده از این عملکردهای پایه مانند زیر است. ابتدا، کامپیوتر سرویس دهنده LISTEN را اجرا می‌کند تا نشان دهد که آماده پذیرش ارتباطات ورودی است. عملکرد LISTEN معمولاً بصورت یک فراخوانی مسدودشونده (blocking) پیاده‌سازی می‌شود؛ بدین معنا که بعد از اجرای این عملکرد، پروسس سرویس دهنده تا زمان دریافت درخواست اتصال مسدود می‌شود.

سپس، مشتری عملکرد CONNECT را اجرا می‌کند تا به سرویس دهنده متصل شود. فراخوانی CONNECT معمولاً باید مشخص کند که مقصد اتصال کجاست، بهمن دلیل ممکنست پارامتری داشته باشد که آدرس سرویس دهنده را بدست می‌دهد. با این عمل، سیستم عامل مشتری پیامی را به همتای خود می‌فرستد، و درخواست اتصال می‌کند - این مرحله با شماره (1) در شکل ۱-۱۸ نشان داده شده است. پس از آن، پروسس مشتری تا زمان دریافت پاسخ به حالت تعلیق (suspend) درمی‌آید. وقتی این بسته به سرویس دهنده رسید، توسط سیستم عامل پردازش می‌شود، و وقتی می‌بیند یک درخواست اتصال است، بدنبال یک پروسس شنونده (listener) می‌گردد. اگر چنین پروسسی را پیدا کند، آنرا از حالت انسداد خارج کرده، و یک پیام تصدیق دریافت (acknowledgement) به مشتری پس می‌فرستد - مرحله (2). دریافت این پیام توسط مشتری باعث می‌شود تا پروسس از حالت تعلیق در آید. در این لحظه پروسسهای سرویس دهنده و مشتری هر دو در حال اجرا هستند، و

مفهوم	عملکرد پایه
انتظار برای دریافت اتصال	LISTEN
برقراری ارتباط با همتای منتظر	CONNECT
انتظار برای دریافت اتصال	RECEIVE
ارسال پیام به همتا	SEND
پایان اتصال	DISCONNECT

شکل ۱-۱۷. پنج عملکرد پایه لازم برای پیاده‌سازی یک سرویس اتصال-گرای ساده.



شکل ۱-۱۸. تبادل بسته ها در یک شبکه اتصال-گرای مشتری-سرویس دهنده.

اتصال برقرار شده است. توجه به این نکته ضروریست که پیام تصدیق دریافت (2) توسط کُد پروتکل (که در سطح هسته - kernel level - اجرا می شود) ایجاد می شود، نه کُد عملکرد پایه (که یک پروسس سطح کاربر - user level - است). اگر هنگام دریافت درخواست اتصال توسط سرویس دهنده، هیچ پروسس شنونده ای وجود نداشته باشد، نتیجه نامشخص است. در برخی از سیستمها، این درخواست برای مدتی در صف (queue) می ماند، به امید اینکه شاید یک پروسس LISTEN اجرا شود.

این فرآیند بسیار شبیه تماس تلفنی مشتری با مدیر قسمت پشتیبانی مشتریان در یک شرکت است. نشستن مدیر قسمت پشتیبانی در کنار تلفن بنوعی اعلام آمادگی برای دریافت تقاضاها است (همان پروسس LISTEN). سپس یکی از مشتریان زنگ می زند (پروسس CONNECT)؛ و به محض اینکه مدیر پشتیبانی گوشی تلفن را برداشت، ارتباط برقرار می شود.

قدم بعدی را باید سرویس دهنده بردارد: اجرای عملکرد RECEIVE برای دریافت اولین درخواست. معمولاً سرویس دهنده این کار را بلافاصله بعد از برطرف شدن انسداد (و حتی قبل از اینکه پاسخ تصدیق دریافت آن به مشتری برسد) انجام می دهد. فراخوانی RECEIVE نیز جزء پروسسهای مسدودشونده است. وقتی مشتری اعلام آمادگی سرویس دهنده را دریافت کرد، درخواست خود را در قالب یک عملکرد SEND به آن می فرستد - مرحله (3).

رسیدن بسته SEND به سرویس دهنده آنرا از حالت انسداد خارج کرده، و باعث می شود تا بتواند به درخواست مشتری رسیدگی کند. پس از آماده شدن پاسخ، سرویس دهنده آنرا با اجرای عملکرد SEND به مشتری پس می فرستد - مرحله (4). رسیدن این بسته به مشتری باعث می شود تا از حالت انسداد خارج شده، و محتوای پاسخ را بررسی کند. اگر مشتری درخواست های بیشتری داشته باشد، آنها را در همین مرحله انجام می دهد؛ در غیر اینصورت، با اجرای عملکرد DISCONNECT اتصال را قطع می کند. فراخوانی DISCONNECT نیز معمولاً یک فراخوانی مسدودکننده است، و باعث تعلیق پروسس مشتری و اعلام پایان ارتباط به سرویس دهنده می شود - مرحله (5). پروسس سرویس دهنده با دریافت پیام DISCONNECT از مشتری، عملکرد DISCONNECT را در سمت خود اجرا کرده، و (پس از اعلام به مشتری) ارتباط را قطع می کند. وقتی بسته سرویس دهنده به مشتری رسید، مشتری نیز اتصال را رها می کند - مرحله (6). این بود ماجرای ساده یک ارتباط اتصال-گرا!

البته، زندگی همیشه این قدر شیرین نیست، و خیلی چیزها می تواند آنرا تلخ کند. برای مثال، فرض کنید عملکرد CONNECT قبل از LISTEN اجرا شود، یکی از بسته ها وسط راه گم شود، و اتفاقاتی از این قبیل. بعداً درباره این مسائل بیشتر صحبت خواهیم کرد، اما فعلاً به همین ارتباط ساده شکل ۱-۱۸ دقت کنید.

با توجه به این نکته که برای کامل شدن سیکل این پروتکل به شش بسته نیاز داریم، شاید پرسید که چرا بجای آن از یک ارتباط غیر متصل (که فقط به دو بسته - یک درخواست و یک پاسخ - نیاز دارد) استفاده نکنیم. جواب اینست

که در یک دنیای بی عیب و نقص می توان چنین کرد؛ اما آیا اوضاع همیشه بر وفق مراد ماست؟ اگر فایل ها خیلی بزرگ باشند، خط ارتباطی پُر از خطا باشد، بسته ها مدام گم شوند، چکار باید کرد؟ اگر پای صدها و هزارها بسته در میان باشد، و فقط چند بسته ناقابل گم شود، مشتری چگونه باید این موضوع را متوجه شود؟ مشتری از کجا بفهمد آخرین بسته ای که گرفته، واقعاً آخرین بسته بوده، یا خیلی ساده ارتباط قطع شده است؟ فرض کنید، مشتری فایل دومی را هم درخواست می کند، و بعد یکبار بسته ای می رسد که شماره ۱ دارد. آیا این اولین بسته فایل دوم است، یا اولین بسته فایل اول که تا حالا سرگردان بوده، و همین الان راه خود را پیدا کرده؟ خلاصه، در دنیای واقعی یک شبکه غیرقابل اعتماد مبتنی بر پروتکل درخواست-پاسخ نمی تواند کافی باشد. در فصل ۳ پروتکل هایی را که برای رفع این مشکلات ابداع شده اند، به تفصیل بررسی خواهیم کرد. فعلاً همین قدر کفایت بدانید که، گاهی پروسه هایی که به روش جریان بایت با هم ارتباط برقرار می کنند، بهترین گزینه اند.

۵-۳-۱ رابطه سرویس و پروتکل

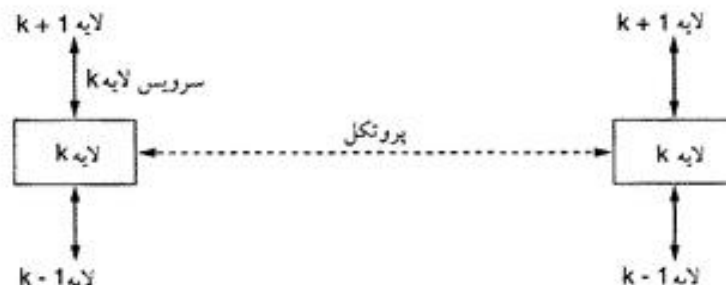
سرویس و پروتکل دو چیز متفاوتند، که اغلب افراد آنها را با هم اشتباه می گیرند. اما این تفاوت بقدری مهم است، که جا دارد باز هم بر آن تأکید کنیم. سرویس (service) عبارتست از مجموعه ای از عملکردهای پایه که یک لایه در اختیار لایه بالاتر از خود قرار می دهد. سرویس فقط می گوید که یک لایه چه کارهایی می تواند برای کاربر خود انجام دهد، ولی هیچ چیز درباره چگونگی آن نمی گوید. سرویس در واقع به واسطه بین دو لایه مربوط می شود، که در آن لایه پائین تر ارائه دهنده سرویس و لایه بالاتر مصرف کننده سرویس است.

اما، پروتکل (protocol) عبارتست از مجموعه قواعد حاکم بر فرمت، مفهوم و نحوه تبادل بسته ها و پیامها بین دو لایه همتا. در واقع این پروتکل است که سرویسهای تعریف شده در هر لایه را پیاده سازی می کند. همتایان هر لایه می توانند پروتکل ارتباطی خود را عوض کنند، مشروط بر اینکه سرویسهایی که به کاربران خود می دهند، تغییری نکنند. با این تعریف، سرویس و پروتکل کاملاً از هم مستقل هستند.

همانطور که در شکل ۱-۱۹ می بینید، سرویس به واسطه دو لایه مربوط می شود، در حالیکه پروتکل پیامهای مبادله شده بین دو لایه همتا (در دو کامپیوتر مختلف) را کنترل می کند. بسیار اهمیت دارد که این دو مفهوم را با هم مخلوط نکنید.

یک مثال مشابه از زبانهای برنامه نویسی می تواند در درک تمایز سرویس و پروتکل کمک کند. سرویس در واقع شبیه نوع داده (data type) یا شیء (object) در زبانهای شیء گرا است. با اینکه می دانیم یک شیء چه خواصی دارد، اما نمی دانیم آنها را چگونه پیاده سازی کرده است. این پیاده سازی همان پروتکل است که از چشم کاربر مخفی می ماند.

البته در بسیاری از پروتکل های قدیمی این تمایز بروشنی وجود ندارد. در این پروتکلها هر تغییری در پروتکل بلافاصله به چشم کاربر خواهد آمد. اما امروزه طراحان شبکه سعی می کنند این تمایز را رعایت کنند.



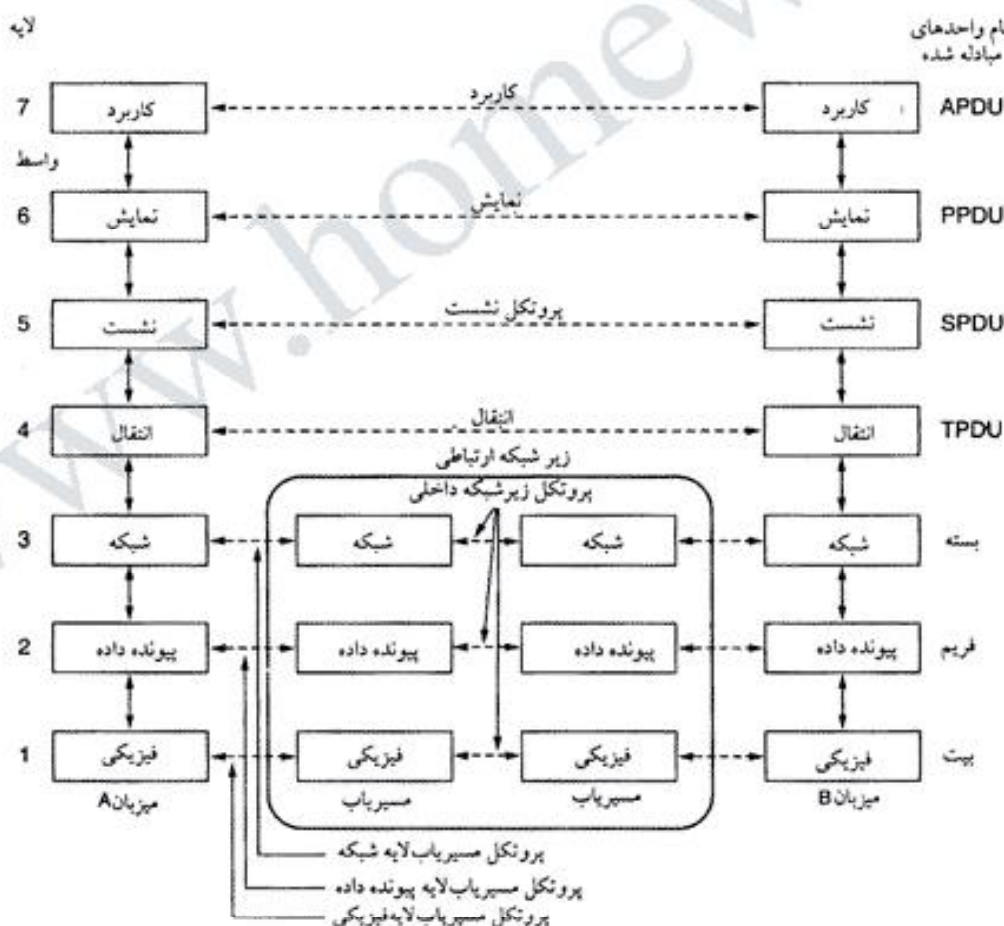
شکل ۱-۱۹. رابطه بین سرویس و پروتکل.

۴-۱ مدل‌های مرجع

حال که با شبکه‌های چند لایه بصورت تئوری آشنا شدید، وقت آنست که نگاهی به چند نمونه از این نوع شبکه‌ها بیندازیم. در دو قسمت آینده دو تا از مهمترین معماریهای شبکه، مدل مرجع OSI و مدل مرجع TCP/IP، را بررسی خواهیم کرد. با اینکه پروتکل‌های مدل OSI امروزه بندرت مورد استفاده عملی دارند، اما این مدل همچنان معتبر، و مشخصات لایه‌های آن از اهمیت زیادی برخوردار است. وضعیت مدل TCP/IP بر عکس است: با اینکه خود این مدل کمتر مورد استفاده قرار می‌گیرد، ولی پروتکل‌های آن کاربرد وسیعی دارند. به همین دلیل، مدل‌های فوق را مفصلاً مورد بررسی قرار داده‌ایم.

۴-۱-۱ مدل مرجع OSI

در شکل ۱-۲۰ مدل OSI را (منهای لایه رسانه فیزیکی) ملاحظه می‌کنید. این مدل بر اساس نظرات پیشنهادی سازمان بین‌المللی استاندارد (International Standards Organization - ISO) - بعنوان اولین استاندارد بین‌المللی شبکه‌های چند لایه - توسعه داده شد (Day and Zimmermann, 1983). این مدل در سال ۱۹۹۵ مورد تجدیدنظر قرار گرفت (Day, 1995). این مدل که نام کامل آن مدل مرجع ارتباطات سیستم‌های باز (ISO) (ISO - Open System Interconnection) است، با ارتباطات سیستم‌های باز - سیستم‌هایی که قادر به ارتباط با سیستم‌های دیگر هستند - سروکار دارد؛ ولی ما آنرا بسادگی مدل OSI خواهیم نامید.



شکل ۱-۲۰. مدل مرجع OSI.

مدل OSI هفت لایه دارد. اما، چرا هفت لایه؟ برخی از مهمترین دلایل انتخاب هفت لایه عبارتند از:

۱. هر کجا به تجرید خاصی نیاز باشد، باید یک لایه ایجاد کرد.
۲. هر لایه باید وظیفه کاملاً مشخصی را انجام دهد.
۳. وظیفه هر لایه باید با در نظر گرفتن تعریف پروتکل‌های استاندارد بین‌المللی انتخاب شود.
۴. مرزهای هر لایه باید طوری انتخاب شود، که کمترین انتقال اطلاعات از آنها لازم باشد.
۵. تعداد لایه‌ها باید آنقدر زیاد باشد، که نیازی به تعریف توابع مشابه در یک لایه نباشد؛ و باید آنقدر کم باشد که معماری شبکه بیش از حد بزرگ و پیچیده نشود.

در زیر لایه‌های مدل OSI را (از پائین به بالا) مورد بحث قرار داده‌ایم. توجه داشته باشید که مدل OSI خود یک معماری شبکه نیست، چون هیچ سرویس یا پروتکلی در آن تعریف نمی‌شود. این مدل فقط می‌گوید که هر لایه چه کاری باید انجام دهد. با اینکه سازمان استانداردهای بین‌المللی (ISO) پروتکل‌های هر لایه را نیز تعریف کرده است، ولی آنها جزء مدل OSI نیستند، و جداگانه بصورت استانداردهای بین‌المللی منتشر می‌شوند.

لایه فیزیکی

لایه فیزیکی (physical layer) وظیفه انتقال بیت‌های خام را از طریق کانال مخابراتی بر عهده دارد. مهمترین نکته در طراحی این لایه اینست که وقتی یک طرف یک بیت 1 می‌فرستد، طرف مقابل یک بیت 1 دریافت کند، نه یک بیت 0. سوالات اساسی در این لایه عبارتند از اینکه، برای 1 و 0 از چه ولتاژهایی استفاده کنیم، هر بیت باید چند نانوثانیه (یک میلیاردیم ثانیه) روی خط دوام بیاورد، آیا انتقال همزمان در هر دو جهت امکانپذیر باشد یا خیر، اتصال اولیه چگونه شروع شود و چگونه پایان یابد، رابط شبکه (network connector) چند پایه باید داشته باشد و وظیفه هر پایه چیست. مسائل طراحی در این لایه عمدتاً از نوع مکانیکی، الکتریکی، تایمینگ (همزمانی)، و رسانه فیزیکی انتقال (که زیر لایه فیزیکی قرار دارد) هستند.

لایه پیوند داده

مهمترین وظیفه لایه پیوند داده (data link layer) عبارتست از تبدیل خط فیزیکی پُر از خطا به یک خط ارتباطی عاری از خطا برای لایه بالاتر، یعنی لایه شبکه. لایه پیوند داده این کار را با شکستن داده‌های ورودی به بسته‌های کوچک چند صد یا هزار بایتی (که فریم داده - data frame - نامیده می‌شوند)، و ارسال آنها انجام می‌دهد. وقتی گیرنده هر بسته را دریافت می‌کند، یک فریم تصدیق دریافت (acknowledgement frame) به فرستنده باز پس می‌فرستد، تا آنرا از دریافت صحیح بسته مطلع کند.

مسئله دیگری که در لایه پیوند داده (و حتی لایه‌های بالاتر) باید حل شود اینست که، چگونه یک گیرنده کُند را با یک فرستنده سریع هماهنگ کند. برای این منظور باید مکانیزمی تعبیه شود تا فرستنده در هر لحظه از مقدار بافر (buffer - حافظه موقت) گیرنده مطلع باشد. در اغلب موارد این دو ویژگی - کنترل جریان اطلاعات و مقابله با خطا - در هم ادغام می‌شوند.

لایه پیوند داده در شبکه‌های پخش (broadcast) باید با مسئله دیگری نیز دست و پنجه نرم کند: کنترل دسترسی به یک کانال مشترک. برای این منظور از زیرلایه‌ای بنام کنترل دسترسی رسانه (medium access control) در لایه پیوند داده استفاده می‌شود.

لایه شبکه

لایه شبکه (network layer) عملکرد زیرشبکه را کنترل می‌کند. یکی از مسائلی که باید در این لایه حل شود، نحوه مسیریابی بسته‌ها از مبدأ به مقصد است. این مسیرها می‌توانند مسیرهای استاتیک باشند (مسیرهایی که بطور

ثابت و بندرت متغیر در شبکه تعبیه شده اند)، یا مسیرهای نیمه استاتیک (مسیرهایی که در ابتدای هر نشست تعیین و مشخص می شوند)، و یا مسیرهای دینامیک (مسیرهایی که در هر لحظه و برای هر بسته از نو - و با توجه به بار شبکه - جستجو و مشخص می شوند).

اگر تعداد بسته های در حال حرکت در یک زیر شبکه بیش از حد باشد، آنها راه یکدیگر را بند آورده و وضعیتی را بوجود می آورند که به آن گلوگاه (bottleneck) یا ازدحام (congestion) گفته می شود. کنترل این وضعیت نیز بر عهده لایه شبکه است. به بیان کلی، کیفیت سرویس (تاخیر انتشار بسته ها، زمان انتقال آنها، و حالت های گذرا در شبکه) همگی جزء مسئولیتهای لایه شبکه است.

اگر یک بسته برای رسیدن به مقصد خود باید از یک شبکه خارج و وارد شبکه دیگری شود، مسائل جدیدی بروز خواهد کرد. اول اینکه، امکان دارد آدرس دهی در این شبکه ها متفاوت باشد. دوم اینکه، ممکن است شبکه دوم این بسته را بکلی نپذیرد، چون مثلاً اندازه آن بیش از حد بزرگ است؛ و یا اینکه پروتکلها با هم فرق داشته باشند، و مسائلی از این قبیل. حل همه این مسائل (و بهم پیوستن شبکه های ناهمگن) نیز از وظایف لایه شبکه است. در شبکه های بخشی مسیریابی بسته ها ساده است، بهمین دلیل لایه شبکه با بسیار کوچک است یا اصلاً نیست.

لایه انتقال

اصلی ترین وظیفه لایه انتقال (transport layer) گرفتن داده ها از لایه بالاتر، تقسیم آن به قطعات کوچکتر (در صورت نیاز)، ارسال آن به لایه شبکه، و حصول اطمینان از دریافت صحیح آنها در طرف مقابل است. علاوه بر آن، همه این کارها باید بگونه ای مؤثر و طوری انجام شود که لایه های بالاتر را از تغییرات اجتناب ناپذیر در سخت افزار ایزوله کند.

این لایه همچنین تعیین می کند که چه سرویس هایی باید در اختیار لایه نشست (و از آنجا، در اختیار کاربران شبکه) قرار گیرد. متداولترین نوع انتقال کانالهای نقطه-به-نقطه عاری از خطاست، که در آن باینها بهمان ترتیبی که فرستاده شده اند، در طرف مقابل دریافت می شوند. با این حال انواع دیگری از سرویسهای انتقال وجود دارد، که از میان آنها می توان به انتقال پیام بدون اطمینان از دریافت منظم آنها، و ارسال همزمان پیامهای بخشی به چندین نقطه اشاره کرد. نوع این سرویسها در لحظه برقراری اتصال مشخص می شود. (البته باید تصریح کنیم که کانال عاری از خطا در دنیای واقعی وجود خارجی ندارد، و آنچه از این اصطلاح برداشت می شود پائین بودن نرخ خطاست، بگونه ای که بتوان در عمل آنرا نادیده گرفت).

لایه انتقال یک لایه نقطه-به-نقطه واقعی است، که در آن کامپیوتر فرستنده (مبدأ) مستقیماً با کامپیوتر گیرنده (مقصد) ارتباط دارد. در لایه های پائینتر ارتباط ماشین مبدأ معمولاً با ماشینهای همسایه (و نه ماشین مقصد) است. تفاوت لایه های ۱ تا ۳ (که بصورت زنجیره ای هستند) با لایه های ۴ تا ۷ (که نقطه-به-نقطه هستند) در شکل ۱-۲۰ نشان داده شده است.

لایه نشست

لایه نشست (session layer) اجازه می دهد تا بین کاربران در ماشینهای مختلف نشست برقرار شود. نشست سرویسهای مختلفی ارائه می کند، از جمله: کنترل دیالوگ (dialog control - کنترل اینکه نوبت چه کسی است)، مدیریت نشانه (token management - جلوگیری از تداخل اعمال مهم)، و سنکرون کردن (synchronization - کنترل عملیات انتقال طولی مدت، و از سرگیری آن از نقطه قطع شده در صورت بروز اختلال).

لایه نمایش

بر خلاف لایه های پائینتر، که عمدتاً با بیت ها سروکار دارند، لایه نمایش (presentation layer) توجه خود را

روی ساختار پیامها و مفهوم آنها متمرکز می‌کند. برای اینکه کامپیوترهایی با ساختارهای متفاوت داده بتوانند با هم ارتباط برقرار کنند، ساختار پیامهای مبادله شده بایستی کاملاً مشخص و استاندارد باشد. وظیفه لایه نمایش مدیریت این ساختارها در سطح بالاست.

لایه کاربرد

بسیاری از پروتکل‌های مورد نیاز کاربران در لایه کاربرد (application layer) قرار دارد، که از معروفترین آنها می‌توان به پروتکل HTTP (HyperText Transfer Protocol) - پروتکل اصلی وب - اشاره کرد. وقتی مرورگر وب می‌خواهد صفحه‌ای را بار کند، نام آن صفحه را با استفاده از پروتکل HTTP به سرور می‌دهد و وب می‌فرستد؛ سرور صفحه وب نیز با همین پروتکل صفحه را به مرورگر برمی‌گرداند. پروتکل انتقال فایل (FTP)، پروتکل انتقال خبر (NNTP)، و پروتکل‌های پست الکترونیک (SMTP و POP) نیز جزء پروتکل‌های کاربردی هستند.

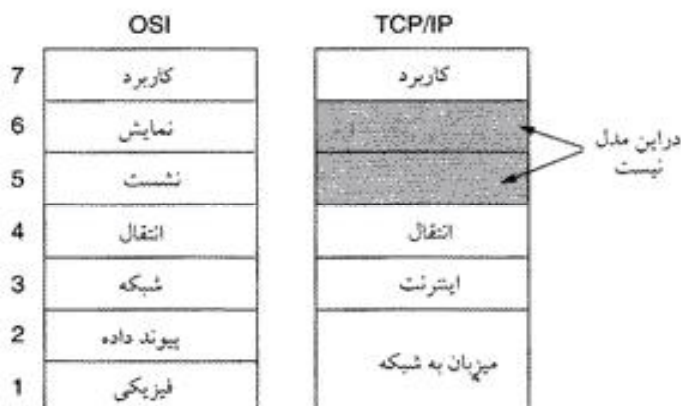
۲-۴-۱ مدل مرجع TCP/IP

اکنون اجازه دهید به مدل مرجع بکار رفته در پدربزرگ شبکه‌های کامپیوتری، آرپانت (ARPANET)، و خلف آن اینترنت، بپردازیم. آرپانت یک شبکه تحقیقاتی بود که توسط DoD (وزارت دفاع ایالات متحده آمریکا) پایه‌ریزی شد. بتدریج صدها دانشگاه و مرکز دولتی بوسیله خطوط اجاره‌ای تلفن (leased line) به این شبکه ملحق شدند. با پیشرفت مخابرات رادیویی و ماهواره، مشکلاتی در پروتکل‌های ارتباطی آرپانت بوجود آمد، که انتخاب یک معماری مرجع جدید را الزامی می‌کرد. یکی از اولین هدفهای آرپانت ارتباط یکپارچه شبکه‌های مختلف بود، که بالاخره (بعد از بررسی چندین پروتکل) توسط مدل مرجع TCP/IP محقق شد. این مدل برای اولین بار توسط (Cerf and Kahn, 1974) تعریف شد، که در سال ۱۹۸۵ مورد تجدیدنظر قرار گرفت (Leiner et al., 1985). فلسفه طراحی مدل مرجع TCP/IP در (Clark, 1988) مورد بحث قرار گرفته است.

دغدغه همیشگی وزارت دفاع این بوده که بخشی از شبکه و وسایل با ارزش آن در یک لحظه (احتمالاً در یک حمله اتمی) نیست و نابود شود، و بهمین دلیل همواره تأکید داشته که این شبکه باید بگونه‌ای طراحی شود که حتی در صورت از بین رفتن بخشی از زیر شبکه‌های آن، بتواند بدون وقفه به کار خود ادامه دهد. بعبارت دیگر، هدف این است که دو کامپیوتر مادامیکه که کار می‌کنند، باید بتوانند با هم ارتباط داشته باشند (حتی اگر تعدادی از ماشینهای واسط بین آنها از مدار خارج شوند). علاوه بر آن، این مدل باید بتواند از عهده طیف وسیعی از کاربردهای متنوع (از انتقال فایل گرفته، تا مکالمه زمان واقعی) برآید.

لایه اینترنت

تمام این الزامات باعث شد تا در نهایت یک شبکه سوئیچینگ بسته (packet-switching) مبتنی بر یک لایه ارتباطات غیر متصل (connectionless) انتخاب شود. این لایه، که لایه اینترنت (internet layer) نام دارد، سنگ بنای معماری TCP/IP است. وظیفه این لایه اینست که به ماشینها اجازه دهد بسته‌های خود را روی شبکه و به سمت مقصد بفرستند. این لایه رسیدن پیامها را با همان ترتیبی که فرستاده شده‌اند، تضمین نمی‌کند؛ وظیفه مرتب کردن پیامها (در صورت نیاز) بر عهده لایه‌های بالاتر است. (دقت کنید که در اینجا «اینترنت» یک کلمه عام است.) لایه اینترنت تا حد زیادی شبیه سیستم پست است. اگر چند نامه را به مقصد کشوری دیگر در صندوق پست بیدازید، با کمی شانس همه آنها به دست گیرنده خواهند رسید. البته احتمال دارد که هر یک از این نامه از مسیر متفاوتی به مقصد رسیده باشد، که این موضوع از دید کاربران پنهان است (و علاقه‌ای هم به دانستن آن ندارند). فرمت بسته‌های پیام و پروتکل آنها در لایه اینترنت تعریف می‌شود، که IP (Internet Protocol) نام دارد.



شکل ۱-۲۱. مدل مرجع TCP/IP.

وظیفه لایه اینترنت اینست که بسته‌های IP را به مقصد پُرساند. مسیریابی بسته‌ها (و جلوگیری از ازدحام در مسیرهای شلوغ) برعهده این لایه است، که از این نظر می‌توان آنرا معادل لایه شبکه در مدل OSI دانست (شکل ۱-۲۱ را ببینید).

لایه انتقال

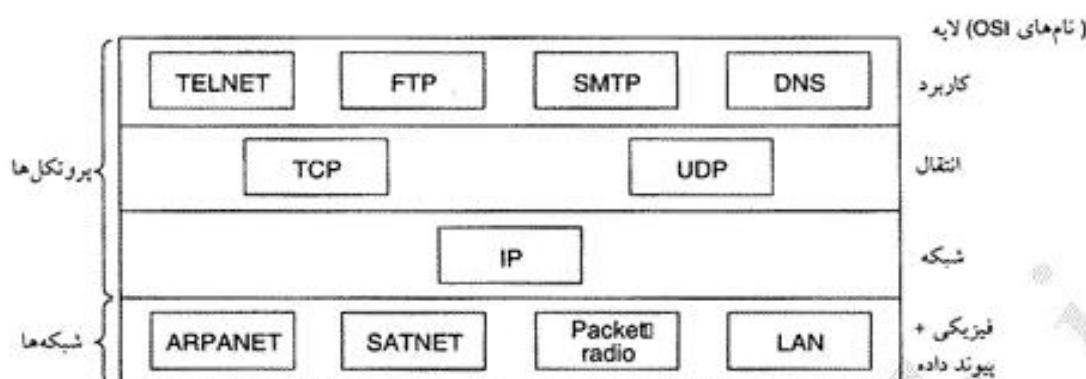
لایه بالای لایه اینترنت در مدل TCP/IP، لایه انتقال (transport layer) نام دارد. این لایه شبیه لایه انتقال در مدل OSI است، و اجازه می‌دهد تا عناصر هم‌تا در کامپیوترهای مبدأ و مقصد با هم مکالمه انجام دهند. در این لایه دو پروتکل انتقال نقطه-به-نقطه تعریف شده است. پروتکل اول، که TCP (Transmission Control Protocol) نام دارد، یک پروتکل اتصال-گرای قابل اعتماد است که اجازه می‌دهد تا جریانی از بایتهای بدون خطا از یک کامپیوتر در اینترنت به کامپیوتر دیگر فرستاده شود. این پروتکل جریان بایتهای را بصورت بسته بسته در آورده، و به لایه اینترنت تحویل می‌دهد. در ماشین مقصد عکس این عمل انجام می‌شود: بسته‌ها به هم چسبانده شده، و بصورت جریانی از بایتهای به لایه بالاتر فرستاده می‌شود. در پروتکل TCP کنترل جریان داده‌ها (flow control) نیز وجود دارد، بدین معنا که فرستنده داده‌ها را سریعتر از آنچه گیرنده توان دریافت آنرا دارد، ارسال نخواهد کرد.

پروتکل دوم این لایه، که UDP (User Datagram Protocol) نام دارد، یک پروتکل غیرمتصل غیر قابل اعتماد است، که در مواردی که نیازی به سخت‌گیریهایی TCP نیست از آن استفاده می‌شود. این پروتکل بیشتر در مواردی که سرعت اهمیت بیشتری دارد تا دقت (مانند انتقال صوت و تصویر)، یا در جاهائیکه فرآیند درخواست-پاسخ فقط یک بار انجام می‌شود، بکار می‌رود. در شکل ۱-۲۲ رابطه پروتکل‌های IP، TCP و UDP را مشاهده می‌کنید. پروتکل IP اکنون در شبکه‌های بسیاری پیاده‌سازی شده است.

لایه کاربرد

مدل TCP/IP لایه‌های نشست یا نمایش ندارد، یعنی در واقع معتقد است که نیازی به آنها نیست. تجربه مدل OSI نیز نشان می‌دهد که این نظر درست است، و این دو لایه بندرت کاربردی پیدا کرده‌اند.

در بالای لایه انتقال لایه کاربرد (application layer) قرار می‌گیرد، که تمام پروتکل‌های سطح بالا در آن قرار دارند. پروتکل‌های ترمینال مجازی (TELNET)، انتقال فایل (FTP) و پست الکترونیک (SMTP) از پروتکل‌هایی هستند که از سالها قبل در این لایه پیاده‌سازی شده‌اند (شکل ۱-۲۲). پروتکل ترمینال مجازی اجازه



شکل ۱-۲۲. پروتکل ها و شبکه ها در مدل TCP/IP.

می دهد تا کاربر وارد کامپیوترهای راه دور شده، و با آنها مانند یک کامپیوتر محلی کار کند. پروتکل انتقال فایل نیز ابزار است مؤثر برای انتقال اطلاعات از یک ماشین به ماشین دیگر. پست الکترونیک در ابتدا چیزی بیش از یک انتقال فایل ساده نبود، ولی بعدها یک پروتکل خاص بنام SMTP برای آن توسعه داده شد. اکنون پروتکل های معروف دیگری نیز در این لایه وجود دارند، که برخی از آنها عبارتند از: پروتکل نام ناحیه (DNS) برای ترجمه نام کامپیوترها به آدرس شبکه، پروتکل انتقال خبر (NNTP) برای خواندن مقالات پوزنت (USENET)، پروتکل انتقال صفحات ابرمتن (HTTP) برای خواندن صفحات وب، و دهها پروتکل دیگر.

لایه میزبان - به - شبکه

در زیر لایه اینترنت یک شکاف بزرگ دیده می شود؛ در واقع مدل TCP/IP درباره این قسمت تا حد زیادی سکوت کرده است، و فقط انتظار دارد که میزبان بنحوی به شبکه وصل شده، و بتواند بسته های IP را ارسال کند. پروتکل انجام این کار در مدل TCP/IP تعریف نمی شود، و در موارد مختلف متفاوت است (حتی کتابها و مقالاتی که درباره TCP/IP نوشته شده اند، بندرت در این باره صحبت می کنند).

۳-۴-۱ مقایسه مدل های OSI و TCP/IP

مدل های TCP/IP و OSI نقاط مشترک زیادی دارند. هر دوی آنها مبتنی بر مجموعه ای از پروتکل های مستقل هستند، و عملکرد لایه ها نیز تا حد زیادی شبیه یکدیگر است. برای مثال، در هر دو مدل لایه های بالای لایه انتقال (و از جمله خود آن) بصورت نقطه به - نقطه عمل می کنند، مستقل از شبکه هستند، و سرویسهای خود را (به شکلی کاربرد - گرا) در اختیار لایه های بالاتر می گذارند.

علیرغم شباهتهای اساسی، این دو مدل تفاوتهای بسیاری نیز با هم دارند، که در این قسمت به آنها خواهیم پرداخت. شایان ذکر است که ما در اینجا مدلها را با هم مقایسه می کنیم، نه مجموعه پروتکل های آنها را (در این باره نیز بعداً صحبت خواهیم کرد). برای یک مقایسه کامل و جامع بین TCP/IP و OSI به کتاب (Piscitello and Chapin, 1993) مراجعه کنید.

در مدل OSI سه مفهوم محوری وجود دارد:

۱. سرویس (service)
۲. واسط (interface)
۳. پروتکل (protocol)

شاید بزرگترین دستاورد مدل OSI روشن ساختن مفاهیم فوق (و تفکیک آنها) باشد. هر لایه سرویسهایی در اختیار لایه های بالاتر از خود قرار می دهد. تعریف این سرویسها فقط می گوید که یک لایه چه کاری انجام می دهد، و هیچ حرفی درباره نحوه انجام آنها و چگونگی استفاده از سرویسها نمی زند.

تعریف چگونگی دسترسی به سرویسهای یک لایه بر عهده واسطه است. واسطه پارامترهای ورودی لازم، و نتیجه ای را که باید منتظر آن باشید، تعریف می کند. حتی واسطه هم نمی گوید که یک لایه چگونه کار خود را انجام می دهد.

و بالاخره، کاری که لایه انجام می دهد را پروتکل های آن لایه تعریف می کنند. یک لایه مادامیکه کارش را بدرستی انجام دهد، می تواند از هر پروتکلی استفاده کند. تغییر پروتکل های یک لایه هیچ تأثیری روی ارتباط آن با لایه های بالاتر نخواهد گذاشت.

ایده های فوق بسیار شبیه مفاهیم مدرن برنامه نویسی شیء گرا هستند. هر شیء، مانند یک لایه، متدها (عملکردها)ی دارد که اشیاء دیگر از آنها استفاده می کنند. نحوه استفاده از این متدها در واقع همان سرویسها نیست که این شیء در اختیار دیگران می گذارد. ورودیها و خروجیهای شیء واسطه آن با دنیای خارج هستند. کد اجرایی شیء نیز شبیه همان پروتکل است، که نحوه عملکرد آن از دید دیگران مخفی است.

در مدل اولیه TCP/IP تمایز بین سرویسها، واسطه ها و پروتکلها واضح و مشخص نبود، اگرچه افرادی (با توجه به تجربه موفق OSI) سعی کرده بودند آنها هر چه بیشتر شبیه OSI کنند. برای مثال، لایه اینترنت فقط دو سرویس واقعی بنامهای SEND IP PACKET و RECEIVE IP PACKET داشت. با توجه به این وضع، پروتکل های OSI بهتر از TCP/IP مخفی شده اند، و امکان تغییر آنها براحتی وجود دارد، چیزی که هدف غایی طراحی لایه ای محسوب می شود.

مدل OSI قبل از اختراع پروتکل های آن طراحی و ابداع شد. این بدان معناست که مدل OSI وابستگی و تمایل خاصی به هیچ مجموعه پروتکلی ندارد، چیزی که در سایر مدلها بسیار دیده می شود. البته این وضعیت یک نقطه ضعف نیز دارد، و آن اینست که طراحان تجربه چندان در زمینه موضوع کار ندارند، و واقعاً نمی دانند کدام عملکرد را باید در کدام لایه قرار دهند. برای مثال، لایه پیوند داده در ابتدا فقط برای شبکه های نقطه به نقطه طراحی شده بود، و وقتی شبکه های پخش و وارد بازار شد، مجبور شدند یک زیر لایه به آن اضافه کنند.

وقتی افراد شروع به طراحی شبکه با استفاده از مدل OSI و پروتکل های موجود کردند، بزودی دریافتند که این شبکه ها با سرویسهای مورد نیاز انطباق ندارند (۱)، بنابراین مجبور شدند زیر لایه های زیادی به آن وصله پینه کنند. بالاخره، کمیته استاندارد مقرر کرد که هر کشور برای خود یک شبکه منطبق با مدل OSI (تحت نظارت دولت) داشته باشد - شبکه ای که به هیچ عنوان آینده (اینترنت) در آن دیده نشده بود. خلاصه، کارها آنطوری که انتظار داشتند از آب در نیامد.

در مورد TCP/IP وضع بر عکس بود: اول پروتکلها اختراع و توسعه داده شدند، و سپس مدلی برای توصیف آنها ساخته شد. هیچ مشکلی در زمینه انطباق پروتکلها با مدل وجود نداشت؛ همه چیز جفت و جور بود. تنها مشکل این بود که این مدل با هیچ مجموعه پروتکل دیگری جور در نمی آمد. این بدان معنا بود که مدل TCP/IP بدرد توصیف شبکه های غیر TCP/IP نمی خورد.

جدای از مسائل فلسفی قضیه، تفاوت دیگر در تعداد لایه های این دو مدل است: مدل OSI هفت لایه دارد و مدل TCP/IP چهار لایه. لایه های شبکه، انتقال و کاربرد در هر دو مشترکند، ولی لایه های دیگر فرق دارند.

تفاوت دیگر در زمینه ارتباطات اتصال-گرا و غیر متصل است. مدل OSI از هر دو نوع ارتباط اتصال-گرا و غیر متصل در لایه شبکه پشتیبانی می کند، ولی در لایه انتقال فقط سرویس اتصال-گرا دارد (چون این سرویس در

معرض دید کاربران است). مدل TCP/IP در لایه شبکه فقط سرویس غیرمتصل دارد، ولی در لایه انتقال از هر دو نوع ارتباط پشتیبانی می‌کند، و دست کاربر را برای انتخاب باز می‌گذارد (که بویژه برای پروتکل‌های ساده درخواست-پاسخ بسیار مهم است).

۱-۴. نگاهی انتقادی به مدل OSI و پروتکل‌های آن

مدل‌های OSI و TCP/IP (و پروتکل‌هایشان) هیچکدام کامل نیستند، و جا دارد برخی از نقاط ضعف آنها را برشماریم. در این قسمت و قسمت آینده، برخی از نقاط ضعف مهم مدل‌های OSI و TCP/IP را بررسی خواهیم کرد. با مدل OSI شروع می‌کنیم.

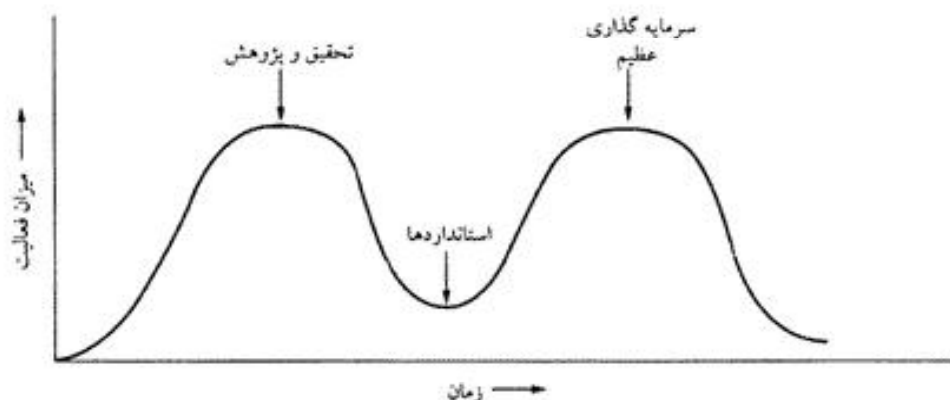
در زمان چاپ ویرایش دوم این کتاب (سال ۱۹۸۹)، بسیاری از متخصصان برجسته شبکه بر این باور بودند که آینده در بست متعلق به مدل OSI و پروتکل‌های آن است، و هیچ چیز نمی‌تواند در مقابل پیشرفت آن مقاومت کند. اما این اتفاق نیفتاد. چرا؟ نگاهی به گذشته درس‌های بسیاری برای چشمان عبرت‌بین دارد، که می‌توان آنها را چنین خلاصه کرد:

۱. زمان نامناسب
۲. تکنولوژی نامناسب
۳. پیاده‌سازی نامناسب
۴. سیاست‌های نامناسب

زمان نامناسب

اولین عامل شکست مدل OSI زمان نامناسب بود. زمانی که یک استاندارد وضع می‌شود، اهمیت حیاتی در موفقیت یا عدم موفقیت آن دارد. دیوید کلارک از دانشگاه M.I.T فرضیه‌ای در زمینه استانداردها دارد که به ملاقات فیل‌ها معروف است، و در شکل ۱-۲۳ آنرا مشاهده می‌کنید.

این شکل میزان فعالیت‌های حول یک موضوع جدید را نشان می‌دهد. وقتی موضوعی برای اولین بار کشف می‌شود، گرداگرد آنرا سیلی از فعالیت‌های تحقیقی (به شکل بحث، مقاله و سخنرانی) فرا می‌گیرد. بعد از مدتی این موج فروکش می‌کند، و بعد از اینکه صنعت به آن موضوع علاقمند شد، موج سرمایه‌گذاری‌ها از پی می‌آید. بسیار مهم است که در نقطه تلاقی این دو فیل (موج تحقیق و موج سرمایه‌گذاری) استانداردها بطور کامل وضع شوند. اگر استاندارد زودتر از موعد (قبل از پایان تحقیقات) نوشته شود، خطر آن هست که موضوع بدرستی



شکل ۱-۲۳. فرضیه ملاقات فیل‌ها

درک نشده باشد، و استاندارد ضعیف از آب در آید. اگر استاندارد دیرتر از موعد (بعد از شروع موج سرمایه‌گذاری) نوشته شود، شرکت‌های بسیاری قبلاً - در مسیرهای مختلف - در آن سرمایه‌گذاری کرده‌اند، و این خطر هست که استاندارد آنها را نادیده بگیرد. اگر فاصله این دو فیل خیلی کم باشد (همه عجله داشته باشند که زودتر کار را شروع کنند)، خطر آن هست که استاندارد دنیسان بین آنها له شوند.

اکنون معلوم شده است که پروتکل‌های استاندارد OSI بین فیله‌ها له شدند. وقتی پروتکل‌های OSI پایه عرصه وجود گذاشتند، پروتکل‌های رقیب (TCP/IP) مدتها بود که در دانشگاه‌ها و مراکز تحقیقاتی پذیرفته شده بودند. با اینکه هنوز موج سرمایه‌گذاری صنعتی در TCP/IP شروع نشده بود، اما بازار آکادمیک آنقدر بزرگ بود که شرکت‌های بسیاری را تشویق به تولید محصولات TCP/IP کند. و وقتی OSI بالاخره از راه رسید، کسی نبود که داوطلبانه از آن پشتیبانی کند. همه منتظر بودند دیگری قدم اول را بردارد؛ قدمی که هرگز برداشته نشد، و OSI در نقطه خفه شد.

تکنولوژی نامناسب

دلیل دیگری که OSI هرگز با نگرش آن بود که، این مدل و پروتکل‌های آن هر دو ناقص و معیوب بودند. انتخاب هفت لایه برای این مدل بیشتر یک انتخاب سیاسی بود تا فنی، و در حالیکه دو لایه آن (نشست و نمایش) تقریباً خالی بودند، در لایه‌های دیگر (لینک داده و شبکه) جای نفس کشیدن نبود.

مدل OSI (و سرویس‌ها و پروتکل‌های آن) بطرزی باورنکردنی پیچیده است. اگر کاغذهای چاپی این استاندارد را روی هم بچینید، ارتفاع آن از نیم متر هم بیشتر خواهد شد! پیاده‌سازی پروتکل‌های OSI بسیار دشوار، و عملکرد آنها ناقص است. در این رابطه، نقل جمله جالبی از پاول موکاپتریس (Rose, 1993) خالی از لطف نیست:

سؤال: از ترکیب یک گانگستر با یک استاندارد بین‌المللی چه چیزی بدست می‌آید؟

جواب: کسی پیشنهادی به شما می‌کند که از آن سر در نمی‌آورد.

مشکل دیگر مدل OSI، علاوه بر غیر قابل فهم بودن آن، اینست که برخی از عملکردهای آن (مانند آدرس‌دهی، کنترل جریان داده‌ها، و کنترل خطا) در تمام لایه‌ها تکرار می‌شود. برای مثال، سالتزر و همکارانش (1984) نشان دادند که کنترل خطا باید در بالاترین لایه انجام شود تا بیشترین تأثیر را داشته باشد، بنابراین تکرار آن در لایه‌های پایینتر نه تنها غیرضروری است، بلکه باعث افت کارایی هم خواهد شد.

پیاده‌سازی نامناسب

با توجه به پیچیدگی بیش از حد مدل OSI و پروتکل‌های آن، جای تعجب نبود که اولین پیاده‌سازیهای آن حجیم، سنگین و کند باشد. آنهایی که با این مدل کار کرده بودند، بزودی پشیمان شدند، و طولی نکشید که کلمه OSI مترادف شد با «کیفیت بد». بعدها محصولات بهتری به بازار آمد، اما آوازه منفی OSI فراموش نشد.

از طرف دیگر، اولین پیاده‌سازی TCP/IP (که بخشی از سیستم عامل یونیکس برکلی بود) بسیار خوب از کار در آمد (و لازم به گفتن نیست که مجانی هم بود). افراد بسیاری سرعت شروع به استفاده از آن کردند، هواخواه آن شدند، آنرا توسعه دادند، و این باعث شد که باز هم به خیل طرفداران آن اضافه شود. در اینجا، برخلاف OSI، ماریج رو به بالا می‌رفت، نه پائین.

سیاست‌های نامناسب

بدلیل اولین پیاده‌سازی TCP/IP، بسیاری از افراد (بویژه در محیط‌های دانشگاهی) تصور می‌کردند که TCP/IP جزئی از یونیکس است، و یونیکس هم در آن دوران محبوبیتی فوق‌العاده داشت.

از سوی دیگر، این عقیده رواج داشت که OSI یک مخلوق دولتی (مخصوصاً دولتهای اروپایی و آمریکایی) است البته این عقیده فقط تا حدی درست بود، اما همین تصور هم که عده‌ای دیوانسالار دولتی بخواهد یک

استاندارد فنی را بزور جا بیندازند، باعث شد تا برنامه‌نویسان و طراحان شبکه تمایلی به همکاری از خود نشان ندهند. زبانهای برنامه‌نویسی PL/1 (که در دهه ۱۹۶۰ از سوی IBM بعنوان زبان آینده توسعه داده شد) و Ada (که وزارت دفاع آمریکا حامی آن بود) بهمین دلیل دچار سرنوشتی مشابه شدند.

۵-۴-۱ نگاهی انتقادی به مدل TCP/IP

مدل TCP/IP و پروتکل‌های آن نیز مشکلات خاص خود را دارند. اول اینکه، در این مدل مفاهیم سرویس، واسط و پروتکل بروشنی از یکدیگر تفکیک نشده‌اند؛ کاری که در مدل OSI بخوبی انجام شده است. به همین دلیل نمی‌توان از TCP/IP بعنوان ابزاری برای طراحی و توسعه شبکه‌های جدید استفاده کرد.

دوم اینکه، مدل TCP/IP به هیچ عنوان یک مدل کلی نیست، و نمی‌توان از آن برای توصیف شبکه‌های غیر TCP/IP استفاده کرد. برای مثال، توصیف بلوتوث با مدل TCP/IP بکلی غیرممکن است.

سوم اینکه، با در نظر گرفتن مفاهیم شبکه‌های چند لایه، لایه میزبان-به-شبکه اساساً یک لایه واقعی نیست، بلکه فقط یک واسط (بین لایه‌های شبکه و لینک داده) است. در واقع، این یکی از مهمترین جاهانیست که مدل TCP/IP مفاهیم واسط و لایه را با هم قاطی کرده است.

چهارم اینکه، در مدل TCP/IP هیچ تمایزی بین لایه‌های فیزیکی و لینک داده نیست (و حتی حرفی از آنها بهمان نیامده است). اینها دو لایه کاملاً متفاوت هستند - لایه فیزیکی با مشخصات کابل و فیبر نوری و کانالهای مخابراتی سروکار دارد، در حالیکه وظیفه لایه پیوند داده شکستن داده‌ها به قطعات کوچکتر و اطمینان از تحویل صحیح آنها به مقصد است. در یک مدل کامل این دو لایه باید از هم جدا باشند؛ کاری که در مدل TCP/IP انجام نشده است.

و بالاخره اینکه، اگر چه پروتکل‌های TCP و IP بسیار خوب طراحی و پیاده‌سازی شده‌اند، بسیاری از دیگر پروتکل‌های این مدل چنین نیستند، و اغلب توسط دانشجویان کنجکاو (و در ساعات بیکاری) نوشته شده‌اند. این پروتکلها بعلت انتشار سریع (که اغلب دلیلی جز مجانی بودن ندارد) سرعت جامی افتند، و بهمین دلیل جایگزین کردن آنها بسیار دشوار می‌شود. برخی از این پروتکلها امروز چیزی جز شرمساری نیستند. مثلاً، پروتکل TELNET اساساً برای ترمینالهای کُند و سنگین تله‌تایپ نوشته شد، و هیچ نشانی از گرافیک و ماوس در آن نیست. اما، بعد از ۲۵ سال خیلی‌ها همچنان از آن استفاده می‌کنند.

بطور خلاصه، مدل OSI (علیرغم برخی از مشکلات آن، و منهای لایه‌های نشست و نمایش) ثابت کرده که بهترین ابزار برای توصیف شبکه‌های کامپیوتریست - اما متأسفانه همچنان روی کاغذ باقی مانده است. از طرف دیگر، با اینکه چیزی بنام مدل TCP/IP در واقع وجود خارجی ندارد، اما پروتکل‌های آن در مقیاس وسیعی مورد استفاده قرار می‌گیرند. از آنجائیکه در دنیای کامپیوتر هر کسی حرف خود را می‌زند، ما هم در این کتاب از یک مدل اصلاح شده OSI استفاده خواهیم کرد، ولی توجه خود را بیشتر روی TCP/IP و پروتکل‌های وابسته به آن معطوف می‌کنیم - البته درباره پروتکل‌های جدید مانند 802، SONET و بلوتوث هم صحبت خواهیم کرد. در حقیقت، مدلی که ما در این کتاب از آن استفاده خواهیم کرد، یک مدل ترکیبی (شکل ۱-۲۴) است.

5	لایه کاربرد
4	لایه انتقال
3	لایه شبکه
2	لایه پیوند داده
1	لایه فیزیکی

شکل ۱-۲۴. یک مدل مرجع ترکیبی، که در این کتاب از آن استفاده خواهد شد.

۵-۱ شبکه های نمونه

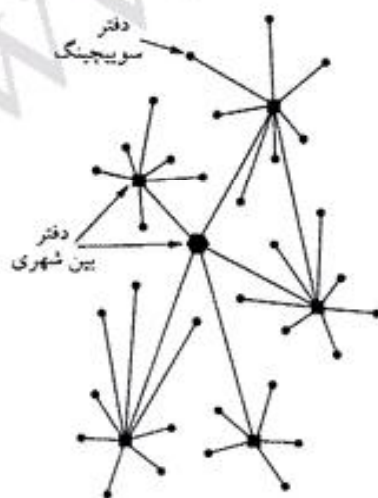
بحث شبکه های کامپیوتری انواع مختلفی از شبکه ها (کوچک و بزرگ، شناخته شده و مجهور) را در بر می گیرد. این شبکه ها در اندازه، کاربرد و تکنولوژیهای پیکار رفته با هم متفاوتند. برای آن که تصویری از تنوع موجود در شبکه های کامپیوتری داشته باشید، در این قسمت نگاهی به چند نمونه از آنها خواهیم انداخت. این قسمت را با معرفی اینترنت (که احتمالاً شناخته شده ترین شبکه هاست) شروع می کنیم، و نگاهی به تاریخچه، سیر تکامل و تکنولوژی آن خواهیم داشت. پس از آن به سراغ ATM (که تفاوت های چشمگیری با اینترنت دارد، و حتی می توان گفت ضد آن است) می رویم. و بالاخره، نگاهی به IEEE 802.11 (استاندارد شبکه های محلی بیسیم) می اندازیم.

۱-۵-۱ اینترنت

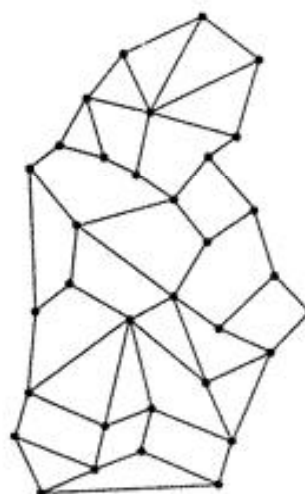
اینترنت (Internet) در واقع اصلاً یک شبکه نیست، بلکه مجموعه ایست از شبکه های مختلف که از پروتکل های خاصی استفاده کرده، و سرویس های مشخصی را ارائه می کند. ویژگی غیرعادی اینترنت اینست که توسط فرد خاصی طراحی نشده، و هیچکس هم آنرا کنترل نمی کند. برای درک بهتر این مطلب، اجازه دهید بینیم اینترنت از کجا شروع شد، و علت آن چه بود. یکی از جالبترین تاریخچه های اینترنت را می توانید در کتاب جان ناقتون - 2000 ببینید. این کتاب نه تنها برای افراد عادی، بلکه برای مورخان نیز جلب است؛ برخی از مطالب ذیل از این کتاب اقتباس شده است. البته کتابهای فنی بیشتری نیز درباره اینترنت و پروتکل های آن نوشته شده، که از آن میان می توان به (Maufer, 1999) اشاره کرد.

آرپانت (ARPANET)

داستان ما از اواخر دهه ۱۹۵۰ شروع می شود. در اوج جنگ سرد، وزارت دفاع ایالات متحده آمریکا به فکر ایجاد یک شبکه فرماندهی و کنترل افتاد که بتواند حتی در مقابل حملات هسته ای دوام بیاورد. در آن زمان تمامی مخابرات نظامی به شبکه تلفن عمومی متکی بود، که مستعد آسیب تشخیص داده شده بود. با یک نگاه به شکل ۱-۲۵ (الف) می توانید مبنای این استدلال را دریابید. در این شکل نقاط سیاه نماینده مراکز سونیچینگ شهری



(الف)



(ب)

شکل ۱-۲۵. (الف) ساختار شبکه تلفن. (ب) طرح بارن برای یک سیستم سونیچینگ توزیع شده.

هستند که هزاران خط تلفن از آنها منشعب می شود. این مراکز نیز بنوبه خود به مراکز بین شهری بزرگتر متصل هستند، که در مجموع شبکه تلفن کشوری را می سازند. آسیب پذیری این سیستم از آنجا ناشی می شد که تخریب چند مرکز بین شهری کلیدی می توانست تماس تلفنی را در کل کشور مختل کند.

در سال ۱۹۶۰ وزارت دفاع قراردادی را با شرکت راند (RAND Corporation) امضا کرد، که در آن وظیفه یافتن یک راه حل به آن محول شده بود. یکی از متخصصان این شرکت، بنام پل بارن (Paul Baran)، طرح یک شبکه توزیع شده (distributed) و تحمل پذیر خطا (fault-tolerant) را پیشنهاد کرد، که آنرا در شکل ۱-۲۵ (ب) می بینید. از آنجائیکه در این شبکه طول مسیر بین مراکز سونیچینگ طولانیتر از آن بود که بتوان از سیگنالهای آنالوگ استفاده کرد، بارن پیشنهاد کرد در این سیستم از تکنولوژی سونیچینگ بسته دیجیتال (digital packet-switching) استفاده شود. بارن گزارشات متعددی برای وزارت دفاع نوشت، و جزئیات سیستم پیشنهادی خود را تشریح کرد. مقامات رسمی پتاگون به ایده نهفته در این سیستم علاقمند شدند، و از AT&T (که در آن زمان انحصار شبکه تلفن کشوری را در دست داشت) خواستند که یک نمونه اولیه از آن بسازد. AT&T طرح بارن را رد کرد؛ بزرگترین و ثروتمندترین شرکت دنیا تحمل نمی کرد که یک جوان تازه از راه رسیده به آنها بگوید چگونه شبکه تلفن بسازند! آنها ادعا کردند که طرح بارن قابل اجرا نیست، و بدین ترتیب ایده آن را در نطفه خفه کردند.

سالها گذشت، و وزارت دفاع همچنان بدنبال سیستم فرماندهی و کنترل ایده آل خود بود. برای درک بهتر اتفاقات بعدی، باید کمی به عقب برگردیم: به اکتبر ۱۹۵۷، زمانی که اتحاد جماهیر شوروی (سابق) با پرتاب اولین قمر مصنوعی بنام اسپوتنیک در مسابقه فضایی از ایالات متحده پیشی گرفت. آیزنهاور، رئیس جمهور وقت ایالات متحده، در جستجو برای یافتن علت عقب افتادگی کشورش، با وحشت دریافت که نیروهای زمینی، دریایی و هوایی آمریکا مشغول دعوا بر سر تقسیم بودجه تحقیقاتی پتاگون هستند. وی بلافاصله تصمیم گرفت که یک مرکز واحد برای تحقیقات نظامی بوجود آورد؛ مرکزی که آرپا (آژانس پروژه های تحقیقاتی پیشرفته Advanced Research Projects Agency - ARPA) نام گرفت. آرپا هیچ دانشمند یا آزمایشگاهی نداشت؛ در واقع، آرپا چیزی نبود جز یک دفتر هماهنگی کوچک با بودجه ای ناچیز (البته با معیارهای پتاگون). آرپا کارش را با عقد قرارداد با واگذاری امتیاز به شرکتها یا دانشگاههایی که ایده های جالبی داشتند، انجام می داد.

در سالهای اول، آرپا بیشتر سعی داشت خطوط کلی مأموریت خود را روشن و ترسیم کند، ولی در سال ۱۹۶۷ توجه مدیرعامل آن، لاری رابرتس، به موضوع شبکه جلب شد. او با متخصصان بسیاری مشورت کرد؛ و یکی از همین متخصصان، بنام ولسی کلارک، بود که پیشنهاد ایجاد یک زیر شبکه سونیچینگ بسته را مطرح کرد (شکل ۱-۱۰).

بعد از مقداری بحثهای اولیه، رابرتس ایده را پسندید و آنرا طی یک مقاله نسبتاً مبهم به گردهمایی اصول سیستم عامل (که در اواخر ۱۹۶۷ در گاتلین بورگ، تنسی برگزار شده بود) ارائه کرد (Roberts, 1967). در میان ناباوری رابرتس، مقاله دیگری نیز به این کنفرانس ارائه شده بود که نه تنها سیستم مشابهی را توصیف می کرد، بلکه حتی صحبت از پیاده سازی آن تحت مدیریت فردی بنام دونالد دیویس در آزمایشگاه ملی فیزیک (NPL) در انگلستان به میان آمده بود. سیستم NPL در واقع سیستمی در سطح ملی نبود، بلکه فقط چند کامپیوتر را در محوطه NPL به هم متصل می کرد، اما نکته مهم این بود که نشان می داد سونیچینگ بسته در عمل کار می کند. از همه جالبتر اینکه، سیستم NPL بر اساس کارهای بارن پایه گذاری شده بود. وقتی رابرتس از گاتلین بورگ برگشت، دیگر مصمم بود چیزی را بسازد که بعدها به آرپانت (ARPANET) معروف شد.

این زیر شبکه تعدادی مینی کامپیوتر بنام IMP (Interface Message Processor) را با خطوط انتقال

56-kbps به هم متصل می کرد. برای رسیدن به قابلیت اعتماد بالا، هر IMP به حداقل دو IMP دیگر متصل می شد. این زیرشبکه در واقع یک زیرشبکه دیتاگرام (datagram subnet) بود، بنابراین اگر تعدادی از خطوط با IMP ها از بین می رفتند، پیامها می توانستند از طریق مسیرهای جایگزین به مقصد برسند.

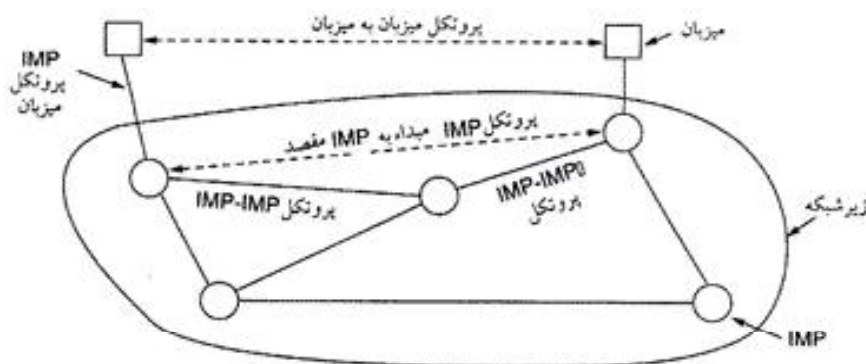
هر گره (node) این شبکه عبارت بود از یک کامپیوتر میزبان و یک IMP، که با سیمی کوتاه به هم وصل می شدند. هر میزبان می توانست پیامهایی تا سقف ۸۰۶۳ بیت به IMP خود بفرستد، و این IMP سپس پیام را به بسته های ۱۰۰۸ بیتی شکسته و آنها را بصورت مستقل به سمت مقصد می فرستاد. هر بسته قبل از اینکه به گره بعدی هدایت شود، بایستی بطور کامل دریافت می شد؛ بدین ترتیب، آرپانت اولین زیرشبکه سوئیچینگ بسته بود که بصورت ذخیره-هدایت (store-and-forward) کار می کرد.

پس از آن آرپا مناقصه ای برای ساخت این زیرشبکه اعلام کرد، که دوازده شرکت اسناد آنرا خریدند. بعد از بررسی پیشنهادات رسیده، در دسامبر ۱۹۶۸ آرپا شرکت BBN را (که یک شرکت مشاوره در کمبریج، ماساچوست بود) برای ساخت این زیرشبکه و نوشتن نرم افزارهای آن برگزید. شرکت BBN مینی کامپیوترهای اصلاح شده هانیول DDP-316 را (که ۱۲ کیلوبایت حافظه ۱۶ بیتی داشت) بعنوان IMP انتخاب کرد. از آنجائیکه قطعات مکانیکی ذاتاً غیر قابل اعتماد فرض می شدند، این IMP ها اصلاً دیسک نداشتند، و با خطوط اجاره ای 56-kbps به هم متصل می شدند. با اینکه امروزه حتی بچه ها هم دیگر خطوط 56-kbps را قبول ندارند (و به کمتر از ADSL راضی نمی شوند)، آنروزها خطوط 56-kbps بالاترین چیزی بود که می شد آرزو کرد.

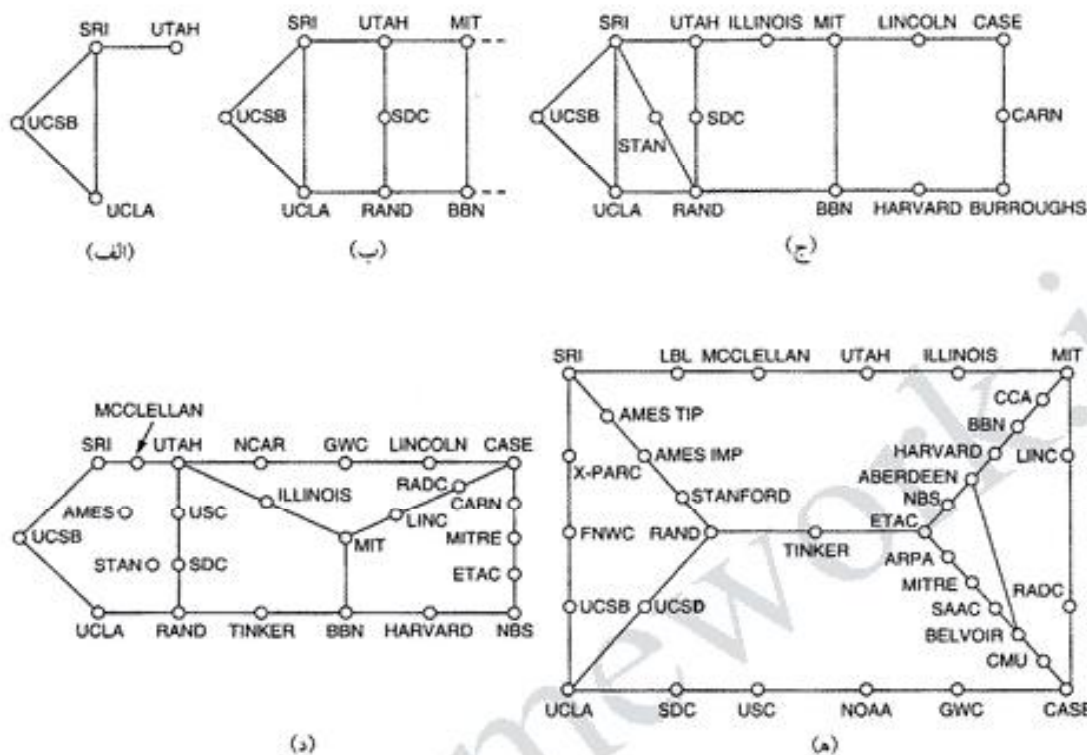
نرم افزار نیز در دو بخش مجزا طراحی شد: زیرشبکه، میزبان. نرم افزار زیرشبکه عبارت بود از پروتکل ارتباط IMP با میزبان، پروتکل IMP-IMP، و نرم افزاری برای بهبود ارتباط IMP مبدأ با IMP مقصد. در شکل ۱-۲۶ طراحی اولیه آرپانت را ملاحظه می کنید.

در خارج از زیرشبکه، میزبانها نیز به نرم افزار نیاز داشتند: پروتکل ارتباط میزبان با IMP، پروتکل میزبان-میزبان، و نرم افزارهای کاربردی. بزودی معلوم شد که BBN احساس می کند با گرفتن پیام در نقطه واسط میزبان-IMP، و تحویل آن در نقطه IMP-میزبان سمت مقابل کارش پایان یافته است.

اما رابرتس مشکل دیگری داشت: کامپیوترهای میزبان هم نیازمند نرم افزار بودند. برای حل این مشکل، در تابستان ۱۹۶۹ رابرتس همایشی از متخصصان شبکه (که عمدتاً دانشجویان تازه فارغ التحصیل بودند) در اسنوبرد، یوتا تشکیل داد. این دانشجویان فکر می کردند کسی وجود دارد که طرح کلی شبکه را برای آنها توضیح دهد، و بعد از آن می توانستند نوشتن نرم افزار را شروع کنند. آنها بسیار شگفت زده شدند وقتی فهمیدند که نه متخصصی برای توضیح طرح شبکه وجود دارد، و نه اساساً چیزی بنام طرح شبکه! آنها دریافتند که باید کار را از صفر شروع کنند.



شکل ۱-۲۶. طراحی اولیه آرپانت.



شکل ۱-۲۷. مراحل رشد آرپانت. (الف) دسامبر ۱۹۶۹. (ب) ژوئیه ۱۹۷۰. (ج) مارس ۱۹۷۱. (د) آوریل ۱۹۷۲. (ه) سپتامبر ۱۹۷۲.

با وجود همه این مشکلات، بالاخره آرپا موفق شد در دسامبر ۱۹۶۹ یک شبکه آزمایشی متشکل از چهار گره دانشگاهی UCSB، UCLA، SRI و یوتا (راه اندازی کند. علت انتخاب این چهار دانشگاه آن بود که همگی آنها قراردادهای متعددی با آرپا داشتند، و از طرف دیگر (صرفاً برای زورآزمایی فنی) کامپیوترهای آنها بکلی با هم سازگار بود. با نصب IMP های جدید این شبکه گسترش یافت، و بزودی سراسر ایالات متحده را تحت پوشش گرفت. در شکل ۱-۲۷ رشد آرپانت را در طی سه سال پس از تولد آن ملاحظه می کنید.

آرپا برای کمک به رشد این نوزاد تازه متولد شده (آرپانت)، در زمینه شبکه های ماهواره ای و مخابرات رادیویی نیز سرمایه گذاری انجام داد. در یک آزمایش معروف، با استفاده از یک شبکه رادیویی پیامی از یک کامیون در حال حرکت در جاده های کالیفرنیا به دانشگاه SRI، و از آنجا از طریق آرپانت به ساحل شرقی ایالات متحده فرستاده شد، که سپس از آنجا از طریق شبکه ماهواره ای به دانشگاه کالج در لندن هدایت شد. بدین ترتیب محققانی که در کامیونی در جاده های کالیفرنیا نشسته بودند، توانستند با کامپیوترهایی در لندن کار کنند.

این آزمایش همچنین نشان داد که پروتکل های موجود آرپانت برای کار روی شبکه های مختلف مناسب نیستند. این نتایج منجر به تحقیقات بیشتر روی پروتکل ها شد، که با اختراع TCP/IP و پروتکل های آن به اوج رسید (Cerf and Kahn, 1974). مدل TCP/IP بویژه برای ارتباطات روی شبکه های مختلف و ناهمگن (که آرپانت روز بروز به سمت آن حرکت می کرد) طراحی شده بود.

بمنظور تشویق و ترغیب پذیرش این پروتکل های جدید، آرپا قراردادهایی با شرکت BBN و دانشگاه کالیفرنیا در برکلی (UCB) منعقد کرد، تا این پروتکل ها را با یونیکس برکلی یکپارچه کنند. محققان برکلی هم کار خود را با

نوشتن برنامه‌های واسط شبکه (که به سوکت - socket - معروف شدند)، و برنامه‌های کاربردی و مدیریتی بنحوی احسن انجام دادند.

زمانه نیز با TCP/IP یار بود؛ بسیاری از دانشگاهها تازه کامپیوترهای جدید VAX را خریده، و آنها را در شبکه‌های LAN به هم متصل کرده بودند، اما هیچ نرم‌افزاری برای شبکه کردن آنها نداشتند. وقتی یونیکس 4.2BSD (با پروتکل‌های TCP/IP، سوکتها و نرم‌افزارهای کمکی خود) بعنوان یک بسته نرم‌افزاری کامل به بازار آمد، بلافاصله مورد قبول جامعه دانشگاهی قرار گرفت. از همه مهمتر اینکه، با TCP/IP می‌شد به آرپانت وصل شد، اتفاقی که بسیاری منتظر آن بودند.

در دهه ۱۹۸۰ شبکه‌های بسیاری (به‌ویژه شبکه‌های محلی) به آرپانت ملحق شدند. با افزایش تعداد کامپیوترهای آرپانت مشکل جدیدی پدید آمد، و آن پیدا کردن یک کامپیوتر در میان خیل عظیم کامپیوترها بود. برای حل این مشکل سیستم نام ناحیه (Domain Name System - DNS) ابداع شد، که نام کامپیوترها را به آدرس IP آنها تبدیل می‌کرد. از آن به بعد، DNS تبدیل به یک پایگاه داده عمومی و توزیع شده شد، که علاوه بر آدرس IP کامپیوترها، اطلاعات دیگری را نیز در اختیار کاربران خود قرار می‌داد. در فصل ۷ درباره DNS مفصلاً صحبت خواهیم کرد.

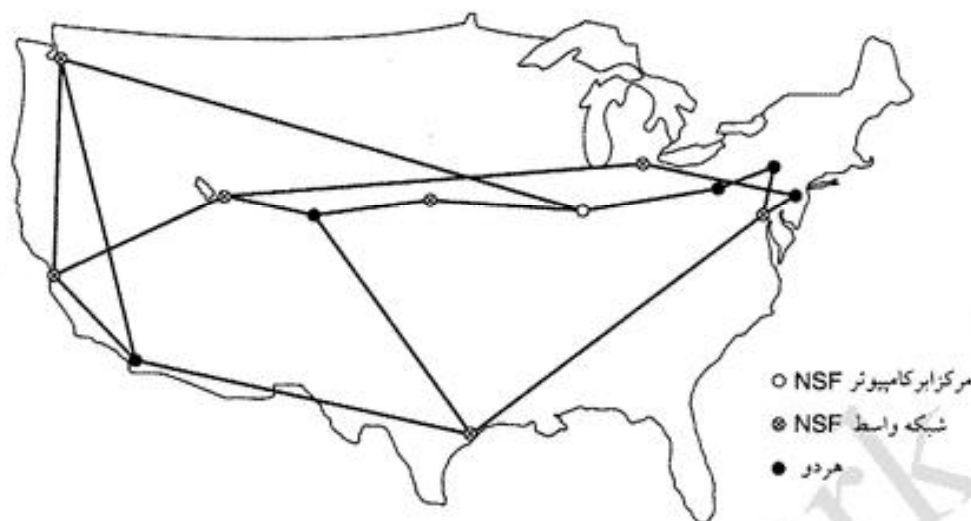
NSFNET

در اواخر دهه ۱۹۷۰ بنیاد ملی علوم ایالات متحده (U.S. National Science Foundation - NSF) شاهد تأثیر روزافزون آرپانت بر تحقیقات دانشگاهی بود. اما هر دانشگاهی که می‌خواست به آرپانت دسترسی داشته باشد، بایستی قراردادی با وزارت دفاع داشته باشد (که بسیاری از آنها نداشتند). پاسخ NSF به این وضعیت، راه‌اندازی شبکه‌ای مشابه آرپانت بود که تمام دانشگاهها به آن دسترسی داشته باشند. بمنظور ایجاد زیربنایی محکم برای این شبکه، NSF با متصل کردن شش ابرکامپیوتر خود در دانشگاههای سان‌دیوگو، بولدر، کامپاین، پیتمبورگ، ایتاکا و پرینستون، یک ستون فقرات (backbone) بوجود آورد. هر یک از این ابرکامپیوترها یک برادر کوچکتر (یک مینی‌کامپیوتر LSI 11، معروف به فازبال) داشت. این فازبال‌ها به خطوط اجاره‌ای 56-kbps متصل بودند، و زیرشبکه را می‌ساختند - یعنی، شبکه NSF از نظر سخت‌افزاری شبیه آرپانت بود. اما، تکنولوژی نرم‌افزاری آن با آرپانت متفاوت بود؛ فازبال‌ها از همان ابتدا به TCP/IP صحبت می‌کردند، که آنرا تبدیل به اولین شبکه گسترده TCP/IP می‌کرد.

بعدها NSF تعداد زیادی شبکه منطقه‌ای تأسیس کرد، که به هزاران دانشگاه، آزمایشگاه تحقیقاتی، کتابخانه، و موزه اجازه می‌داد تا به هر یک از ابرکامپیوترهای آن دسترسی داشته باشند، یا اینکه مستقیماً با یکدیگر تماس برقرار کنند. این شبکه (شامل ستون فقرات و شبکه‌های محلی) NSFNET نامیده شد. از طریق لینکی بین یک IMP و یک فازبال در دانشگاه کارنگی-ملون، NSFNET به آرپانت نیز متصل شده بود. اولین ستون فقرات NSFNET را در شکل ۱-۲۸ مشاهده می‌کنید.

NSFNET یک موفقیت آنی بود، و از همان ابتدا با تراکم کاری روبرو شد. NSF بلافاصله به فکر گسترش NSFNET افتاد، و به همین منظور قراردادی با کنسرسیوم MERIT بست. برای ایجاد دومین ستون فقرات، کانالهای فیبرنوری با ظرفیت 448-kbps از MCI (که اکنون در WorldCom ادغام شده است) اجاره شد. برای مسیریاب‌های شبکه نیز از IBM PC-RT استفاده شد. این شبکه نیز بسیار زود با تراکم کاری روبرو شد، و در سال ۱۹۹۰ ظرفیت ستون فقرات آن به 1.5-Mbps ارتقاء داده شد.

با ادامه رشد NSFNET، بزودی NSF متوجه شد که دولت نمی‌تواند برای همیشه به سرمایه‌گذاری در شبکه ادامه دهد. از طرف دیگر، شرکتهای تجاری نیز مایل بودند به شبکه NSFNET ملحق شوند، ولی مقررات NSF



شکل ۱-۲۸. ستون فقرات NSFNET در سال ۱۹۸۸.

کاربردهای انتفاعی شبکه را ممنوع کرده بود. متعاقب آن، NSF بعنوان اولین قدم بسوی تجاری کردن شبکه، شرکت‌های IBM، MERIT و MCI را به ایجاد یک مؤسسه غیرانتفاعی (بنام Advanced Networks and Services - ANS) ترغیب کرد. در سال ۱۹۹۰، ANS کنترل NSFNET را بدست گرفت، و با ارتقاء لینکهای 1.5-Mbps به 45-Mbps شبکه ANSNET را بوجود آورد. این شرکت بعد از ۵ سال کار به America AOL (Online) فروخته شد. اما در آن زمان دیگر شرکت‌های بسیاری سرویسهای تجاری IP ارائه می‌کردند، و روشن شده بود که دولت باید پای خود را از تجارت شبکه بیرون بکشد.

برای تسهیل امور (و اطمینان از اینکه تمام شبکه‌های منطقه‌ای می‌توانند با هم تماس بگیرند)، NSF چهار قرارداد با شرکت‌های بزرگ برای ایجاد نقطه دسترسی شبکه (Network Access Point - NAP) امضا کرد. این چهار شرکت عبارت بودند از: PacBell (در منطقه سانفرانسیسکو)، Ameritech (در منطقه شیکاگو)، MFS (در منطقه واشینگتن دی.سی.)، Sprint (در منطقه نیویورک). هر اپراتور شبکه که بخواهد سرویسهای ستون فقرات به شبکه‌های منطقه‌ای NSF بدهد، بایستی به تمام NAP ها متصل باشد.

بدین ترتیب، هر بسته که بخواهد از یک منطقه به منطقه دیگر برود، می‌تواند از هر یک از این ستونهای فقرات استفاده کند، که نتیجه آن ایجاد رقابت برای سرویس بهتر و قیمت کمتر است. با این تمهید، ستون فقرات منحصر بفرد دولتی جای خود را به یک زیرساخت متنوع و رقابتی داد. بسیاری از افراد دولت فدرال را به گناه عدم خلاقیت سرزنش می‌کنند، ولی در واقع این بنیاد ملی علوم و وزارت دفاع بودند که زیرساخت‌های اینترنت را شکل داده و سپس اداره آنرا به بخش خصوصی سپردند.

در دهه ۱۹۹۰ مناطق و کشورهای بسیاری، با تأثیرپذیری از الگوی آرپانت و NSFNET، شبکه‌های ملی تحقیقاتی خود را بوجود آوردند. در اروپا، این شبکه‌ها (که EuropaNET و EBONE نام داشتند) از لینکهای 2-Mbps شروع کردند، و به 34-Mbps ارتقاء یافتند. در آنجا نیز زیرساخت‌های شبکه بتدریج به بخش خصوصی محول شد.

کاربردهای اینترنت

بعد از آنکه در اول ژانویه ۱۹۸۳ بعنوان تنها پروتکل رسمی آرپانت معرفی شد، تعداد شبکه‌ها، کامپیوترها و کاربران متصل به آن بسرعت افزایش یافت؛ و وقتی آرپانت و NSFNET به هم متصل شدند، رشد آن

حالت نمایی بخود گرفت. بسیاری از مناطق و کشورها (از جمله کانادا، اروپا و اقیانوسیه) به شبکه ملحق شدند. در اواسط دهه ۱۹۸۰ دیگر افراد به این مجموعه به عنوان شبکه ای از شبکه ها (که بعدها به اینترنت معروف شد) نگاه می کردند، بدون آنکه هیچگونه پخشنامه رسمی در کار باشد، یا حتی مراسم افتتاحیه ای (با قیچی و نوارهای رنگی، و تشویق و هورا) برگزار شده باشد.

چیزی که اینترنت را به هم متصل نگه می دارد، مدل TCP/IP و مجموعه پروتکل های آن است. پذیرش TCP/IP باعث شد تا سرویس های جهانی بتوانند جنبه عملی بخود بگیرند.

اما واقعاً «روی اینترنت بودن» چه معنایی دارد؟ طبق تعریف ما، ماشینی روی اینترنت است که مجموعه پروتکل های TCP/IP را اجرا کند، یک آدرس IP داشته باشد، و بتواند بسته های IP را به تمام ماشین های دیگری که روی اینترنت هستند، بفرستد. صرف توانایی ارسال و دریافت ایمیل به معنای بودن روی اینترنت نیست، چون سرویس های ایمیل می تواند به شبکه های خارج از اینترنت هدایت شود. با این حال، اوضاع با وضعیتی که در حال حاضر وجود دارد (میلیونها کامپیوتر شخصی می توانند با مودم به یک ISP وصل شده، یک آدرس IP موقتی بگیرند، و بسته های IP رد و بدل کنند)، کمی مغشوش و مبهم است. اما، مادامیکه این کامپیوترها به مسیر پاب ISP متصل هستند، پُر بیراه نیست که آنها را روی اینترنت بدانیم.

اینترنت سستی (از ۱۹۷۰ تا اوایل دهه ۱۹۹۰) چهار کاربرد عمده داشت:

۱. ایمیل (e-mail) - نوشتن، ارسال و دریافت نامه های پُست الکترونیک از همان روزهای اول آرپانت جزء سرویس های آن بود، و همچنان یکی از محبوبترین هاست. امروزه بسیاری از افراد روزانه دهها و صدها ایمیل دریافت می کنند، و به آن بعنوان دریچه ای برای ارتباط با دنیای خارج نگاه می کنند - بسیار بیشتر از تلفن یا پُست معمولی.
۲. اخبار (news) - گروه خبری (newsgroup) یک محفل اختصاص یافته برای تبادل پیام در یک زمینه خاص است. امروزه هزاران گروه خبری در زمینه های فنی و غیر فنی (از جمله کامپیوتر، علوم، هنر و سیاست) وجود دارند. هر گروه خبری برای خود قواعد و مقرراتی دارد، که سرپیچی از آنها را بر نمی تابد.
۳. ورود از راه دور (remote login) - هر روز هزاران نفر در سراسر دنیا برای ورود به کامپیوترهای دیگر از طریق اینترنت (البته آنهایی که حق ورود به آنها را داشته باشند)، از برنامه هایی مانند rlogin، telnet یا ssh استفاده می کنند.
۴. انتقال فایل (file transfer) - با استفاده از برنامه های FTP، کاربران اینترنت می توانند فایل های خود را از یک ماشین به ماشین دیگر کپی کنند. جریان انتقال دانش از این طریق بسیار گسترده و متنوع است.

تا اوایل دهه ۱۹۹۰ اینترنت جولانگاه دانشگاهیان، کارمندان دولت و محققان صنعتی بود؛ اما یک برنامه کاربردی جدید بنام WWW (World Wide Web) این وضعیت را بکلی تغییر داد، و میلیونها نفر افراد عادی نیز توانستند به کاربران حرفه ای اینترنت ملحق شوند. این برنامه، که توسط تام برنرز-لی (Tom Berners-Lee) از فیزیکدانان مرکز تحقیقات هسته ای اروپا (CERN) ابداع شد، هیچ یک از سرویس های اینترنت را عوض نکرد، ولی کاربرد آنها را ساده تر کرد. به کمک این تکنولوژی جدید، و برنامه مرورگر موزائیک (Mosaic browser)، که توسط مارک آندرسن در مرکز ملی کاربردهای ابرکامپیوتر (NCSA) نوشته شد، WWW ایجاد سایتهایی متشکل از صفحات مختلف (و با اطلاعاتی در قالبهای متن، تصویر، صدا، و حتی ویدئو)، با لینک هایی به صفحات دیگر را امکانپذیر کرد. با کلیک کردن روی یک لینک (link)، کاربر مستقیماً به صفحه ای که مشخص شده، می پرد. حتماً سایتهای بسیاری متعلق به شرکتهای بزرگ را دیده اید، که می توانید با کلیک کردن هر یک از لینک های آن، به صفحه مربوطه (مثلاً، صفحه مربوط به محصولات شرکت، لیست قیمت های آن، پشتیبانی فنی، فروش و غیره) وارد شوید. در زمانی کوتاه صفحات جدید و متنوعی به WWW اضافه شد، صفحاتی مانند نقشه شهرها و کشورها.

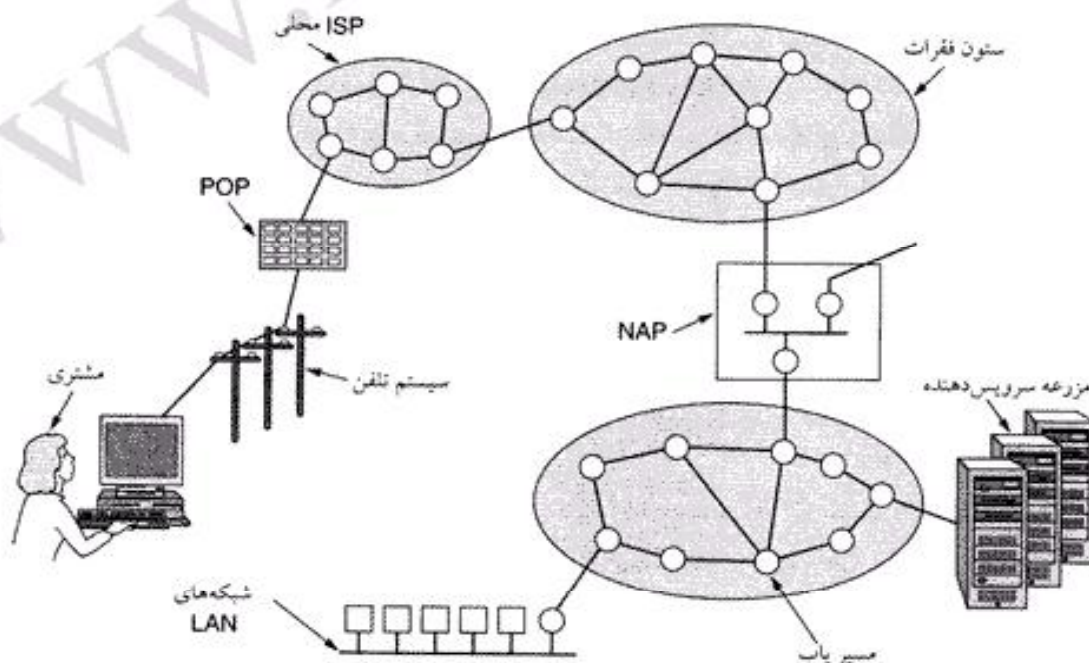
جدول قیمت سهام، کاتالوگ کارتهای کتابخانه‌ها، برنامه‌های رادیویی، و حتی متن کامل کتابهایی که از شمول قانون حق‌التألیف خارج شده‌اند (مانند کتابهای مارک تواین، چارلز دیکنز، و امثالهم). حتی بسیاری از افراد عادی نیز برای خود سایت (صفحات خانگی) ساخته‌اند.

موتور محرکه این رشد، شرکت‌های ارائه‌دهنده سرویس اینترنت (ISP - Internet Service Provider) بودند. این شرکت‌ها به افراد اجازه می‌دادند تا از خانه و با کامپیوترهای شخصی خود به اینترنت متصل شده، و از سرویسهای آن استفاده کنند. در دهه ۱۹۹۰، این شرکت‌ها هر ساله برای دهها میلیون نفر امکان دسترسی اینترنت فراهم کردند، و چهره آنرا از محیطی دانشگاهی و نظامی به یک شبکه عمومی تغییر دادند. تعداد دقیق کاربران اینترنت در حال حاضر معلوم نیست، ولی محققان سر به صدها میلیون نفر می‌زنند، و خیلی زود از مرز یک میلیارد خواهد گذشت.

معماری اینترنت

در این قسمت سعی می‌کنیم تصویری کلی از اینترنت بدست دهیم (شکل ۱-۲۹ را ببینید). به دلیل شباهتها و تداخل وظایف زیادی که بین شرکت‌های مخابرات و ISP ها وجود دارد، امروزه اوضاع بسیار در هم و مغشوش است، و بسختی می‌توان گفت کی چکاره است - به همین دلیل توضیحات ذیل ساده‌تر از آن چیز است که در واقعیت وجود دارد. اجازه دهید شکل ۱-۲۹ را جزء به جزء بررسی کنیم.

بهترین نقطه برای شروع، خانه مشتری (client) است. در اینجا فرض را بر این گذاشته‌ایم که مشتری با استفاده از یک مودم و خط تلفن به ISP متصل می‌شود. مودم (modem) وسیله‌ایست که سیگنالهای دیجیتالی کامپیوتر را به سیگنالهای آنالوگ تبدیل می‌کند، تا این سیگنالها بتوانند بدون اعوجاج روی خطوط تلفن مستقل شوند. این سیگنالها در نقطه تماس ISP (که به POP - Point Of Presence - معروف است) مجدداً تبدیل به سیگنالهای دیجیتال شده، و وارد شبکه منطقه‌ای ISP می‌شود. از این نقطه به بعد سیستم کاملاً دیجیتال است، و بر مبنای سوئیچ بسته کار می‌کند. اگر این ISP همان شرکت مخابرات باشد، POP در مرکز سوئیچینگ تلفن واقع



شکل ۱-۲۹. یک تصویر کلی از اینترنت.

خواهد بود؛ اما اگر ISP و شرکت مخابرات یکی نباشند، POP یک مرکز سوئیچینگ کوچک بین راهی خواهد بود، که از آنجا به شبکه تلفن وصل می‌شود.

شبکه منطقه‌ای هر ISP از چند مسیر یاب، که به شهرهای مختلف تحت پوشش آن ISP سرویس می‌دهند، تشکیل می‌شود. اگر مقصد بسته ارسال شده از مشتری یکی از کامپیوترهای واقع در همان شبکه منطقه‌ای ISP باشد، بلافاصله به آن تحویل داده می‌شود. ولی اگر چنین نباشد، بسته به اپراتور ستون فقرات ISP داده خواهد شد. اپراتور ستون فقرات (backbone operator) بالاترین نقطه این زنجیره است (شرکتهای At&T و Sprint جزء اپراتورهای عمده هستند). هر اپراتور یک شبکه بزرگ از ستونهای فقرات بین‌المللی (مشکل از هزاران مسیر یاب که با فیبرهای نوری پرسرعت به هم متصلند) را اداره می‌کند. شرکتهای بزرگ و آنهایی که سرویسهای میزبانی اینترنت ارائه می‌کنند، معمولاً مستقیماً به ستون فقرات متصل هستند. اپراتورهای ستون فقرات این نوع خدمات را تشویق می‌کنند، و برای آن تسهیلات ویژه‌ای بنام کاریر کرایه‌ای فراهم می‌آورند، که باعث ارتباط نزدیک با ستون فقرات از سرعتهای بسیار بالایی برخوردارند.

اگر مقصد بسته ارسال شده یکی از ISP های متصل به ستون فقرات باشد، به نزدیکترین مسیر یاب فرستاده می‌شود و از آنجا بدست وی خواهد رسید. با این حال در دنیا ستونهای فقرات متعددی (با سرعتهای مختلف) وجود دارند، و احتمال دارد که این بسته وارد یکی از این شاهراههای رقیب شود. برای اینکه بسته‌ها بتوانند براحتی بین شاهراهها حرکت کنند، تمام آنها باید به یک NAP متصل باشند - NAP چیزی نیست بیش از اتاقی پُر از مسیر یاب‌های متعدد (حداقل یکی به ازای هر شاهراه)، که در یک LAN ساده به هم متصل شده‌اند. شاهراههای بزرگ، غیر از اتصال از طریق NAP، معمولاً بصورت مستقیم نیز به شاهراههای دیگر راه دارند (که به آن ارتباط دوجانبه گفته می‌شود). یکی از تناقضهای بزرگ اینترنت آن است که شرکتهایی که در انتظار عموم با هم رقابت سخت دارند، در خفا با یکدیگر ارتباطات نزدیک و تنگاتنگ برقرار می‌کنند (Metz, 2001).

این هم از مروری اجمالی بر اینترنت. البته در فصلهای آینده تک تک این اجزاء، الگوریتمها، و پروتکلها را مفصلاً بررسی خواهیم کرد. نکته‌ای که جالبست بدانید اینست که امروزه بسیاری از شرکتهای ارتباطات داخلی شبکه خود را بر اساس مدل و تکنولوژیهای اینترنت بنا می‌کنند، چیزی که به اینترنت (intranet) معروف است.

۲-۵-۱ شبکه‌های اتصال-گرا: X.25، Frame Relay و ATM

از همان اولین روزهایی که شبکه پا به عرصه وجود گذاشت، جنگ بین طرفداران زیرشبکه‌های اتصال-گرا و شبکه‌های غیرمتصل (دیتاگرام) نیز شروع شد. مهمترین برگ برنده طرفداران زیرشبکه‌های غیرمتصل همان آرپانت/اینترنت است. بیاد دارید که قصد اولیه وزارت دفاع آمریکا از بنیانگذاری آرپانت، ایجاد شبکه‌ای بود که بتواند در مقابل ضربات هسته‌ای (و منهدم شدن بخش بزرگی از خطوط و تجهیزات انتقال) دوام بیاورد (در واقع، هدف اصلی این طرح بالا بردن ضریب تحمل خرابی شبکه بود). این رهیافت منجر به طراحی شبکه‌ای شد که در آن هر بسته راه خود را مستقل از بسته‌های دیگر طی می‌کند. بدین ترتیب، اگر تعدادی از مسیر یاب‌های شبکه از مدار خارج شوند، مادامیکه شبکه بتواند مسیرهای جدید خود را از نو پیگیربندی کند، در ارسال بسته‌ها از مبدأ به مقصد خللی پیش نخواهد آمد.

طرفداران زیرشبکه‌های اتصال-گرا معمولاً همان شرکتهای تلفن هستند. در این سیستم، آغازکننده ارتباط قبل از آنکه بتواند ارسال اطلاعات را شروع کند، بایستی منتظر برقراری ارتباط مستقیم با طرف مقابل بماند. این ارتباط فیزیکی در تمام طول تماس برقرار می‌ماند، و تمام بسته‌های اطلاعات از همین مسیر واحد عبور خواهند کرد. اگر هر یک از تجهیزات این مسیر به هر دلیلی از کار بیفتند، تماس قطع خواهد شد - چیزی که وزارت دفاع مسلماً نمی‌پسندد.

پس علت علاقه شرکت های تلفن به این سیستم چیست؟ دو دلیل اصلی این علاقه عبارتند از:

۱. کیفیت سرویس
۲. حسابداری مصرف کنندگان

در شبکه های اتصال-گرا هر تماس مقداری از منابع زیر شبکه (از قبیل توان پردازشی مسیریاب ها) را بخود اختصاص می دهد، و در صورتیکه این منابع به حالت اشباع برسند، تماس جدید امکانپذیر نبوده، و کاربر با بوق اشغال روبرو خواهد شد. در این روش تماس ها (بدلیل اختصاص منابع کافی) از کیفیت بالایی برخوردار هستند. از طرف دیگر، اگر در شبکه های غیر متصل تعداد زیادی بسته به یکباره وارد یک مسیریاب شوند، ممکنست برخی از آنها (بدلیل کمبود امکانات پردازشی) در داخل روتر از بین بروند. البته فرستنده متوجه این نقص خواهد شد، و بسته های گمشده را از نو ارسال خواهد کرد، ولی همین موضوع باعث افت کیفیت شبکه (بویژه در مورد صدا و تصویر) می شود. لازم به گفتن نیست که شرکت های تلفن پیش از هر چیز نگران کیفیت صدا هستند، و به همین دلیل همچنان زیر شبکه های اتصال-گرا را ترجیح می دهند.

دلیل دومی که شرکت های تلفن سرویس های اتصال-گرا را بیشتر می پسندند، امکان صدور صورتحساب برای مشترکان است (کاری که مدتهاست به آن عادت کرده اند). هزینه تماس های بین شهری و خارج از کشور معمولاً بر حسب مدت مکالمه محاسبه می شود (علت اتخاذ این روش هم بیشتر سادگی آن بوده است). اگر تماس مستقیمی بین دو طرف مکالمه برقرار نباشد، طبعاً این شرکتها نمی توانند برای مشترکان خود صورتحساب صادر کنند.

سیستم های محاسبه صورتحساب هزینه بسیار سنگینی به شرکت های تلفن تحمیل می کند. اگر یک شرکت تلفن بر اساس یک شارژ ثابت ماهیانه از مشترکان خود پول دریافت کند، علیرغم بالا رفتن مصرف مشترکان، می تواند پول زیادی صرفه جویی کند. اما عوامل زیادی (که بیشتر آنها هم سیاسی هستند) با اتخاذ این روش مخالفت می کنند. جالبست بدانید که در اغلب سیستم های مشابه (مانند تلویزیون های کابلی و برخی پارک های تفریحی) از روش محاسبه ثابت استفاده می شود. در این سیستمها هم امکان پرداخت به ازای مصرف وجود دارد، ولی بدلیل هزینه بالای صدور صورتحساب معمولاً از آن اجتناب می شود.

به دلایل فوق، جای تعجب نیست که شرکت های تلفن طرفدار زیر شبکه های اتصال-گرا باشند؛ اما تعجب برانگیز این است که، اینترنت هم دارد به همین سمت پیش می رود - البته با این استدلال که این کار باعث بالا رفتن کیفیت سرویس های صدا و تصویر آن خواهد شد. اکنون اجازه دهید چند شبکه اتصال-گرا را بهتر بشناسیم.

X.25 و Frame Relay

اولین شبکه اتصال-گرا که وارد سرویس عمومی شد، شبکه X.25 بود. این شبکه در اوایل دهه ۱۹۷۰، و در زمانی طراحی شد که شرکت های تلفن بصورت انحصاری عمل می کردند، و هر کشور شبکه ملی خاص خود را داشت. برای استفاده از X.25، ابتدا کامپیوتر مبدأ با ماشین مقصد تماس تلفنی برقرار می کرد. از آنجائیکه در آن واحد تماس های مختلفی می توانست وجود داشته باشد، به هر تماس تلفنی یک شماره داده می شد. بسته های داده بسیار ساده بودند: یک سرآیند ۳ بیتی و بدنه ای متشکل از ۱۲۸ بایت. سرآیند (header) تشکیل می شد از یک شماره تماس ۱۲ بیتی، یک شماره ترتیب بسته (packet sequence number)، یک عدد تصدیق دریافت (acknowledgement number)، و چند بیت متفرقه. شبکه های X.25 به مدت نزدیک به یک دهه با موفقیتی نسبی کار کردند.

در دهه ۱۹۸۰ شبکه های X.25 جای خود را به نوع جدیدی از شبکه های اتصال-گرا بنام frame relay (رله فریم) دادند. این شبکه جدید اساساً هیچ نوع کنترل خطا و کنترل جریان نداشت، و بسته ها به همان ترتیب

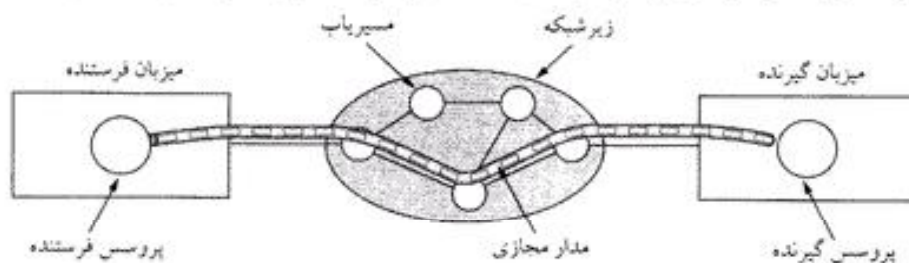
دریافت در مقصد تحویل می شدند (البته اگر به مقصد می رسیدند). این سه خصوصیت (فقدان کنترل خطا، فقدان کنترل جریان، و تحویل ترتیبی بسته ها) شبکه های frame relay را بسیار شبیه یک LAN بزرگ می کند، و در واقع بزرگترین کاربرد آن هم همین است: اتصال چند LAN دور از هم، و ایجاد یک LAN بزرگ. شبکه های frame relay هم نسبتاً موفق بودند، و هنوز در برخی جاها از آنها استفاده می شود.

حالت انتقال آسنکرون (ATM)

یکی دیگر از شبکه های اتصال-گرا (که اهمیت بسیار بیشتری نیز دارد) شبکه حالت انتقال آسنکرون (Asynchronous Transfer Mode - ATM) است. علت این نامگذاری عجیب آن است که در شبکه های تلفن اکثر تماسها بصورت سنکرون (synchronous - وابسته به پالس ساعت) هستند، در حالیکه ATM چنین نیست. شبکه ATM در اوایل دهه ۱۹۹۰ طراحی شد، و سروصدای زیادی نیز با کرد (Ginburg, 1996; Gorlaski, 1995; Ibe, 1997; Kime et al., 1994; Stallings, 2000). شبکه های ATM با ادعای ادغام تمام انواع شبکه و سیستم های مخابراتی (صدا، داده، تلویزیون کابلی، تلکس، تلگراف، کبوترهای نامه بر، قوطیهای حلی سیمی، طبل های آفریقایی، علامت های دودی سرخپوستان، و خلاصه هر چیزی که به نوعی اطلاعات منتقل می کند) به میدان آمدند - اتفاقی که هرگز نیفتاد. علت آن هم تا تقریباً شبیه همان بلایی بود که سر OSI آمد (زمان نامناسب، تکنولوژی بد، پیاده سازی نامناسب، و سیاست های غلط). شرکت های اینترنتی که منتظر وسیله ای بودند تا شرکت های تلفن را در همان راند اول از پا در آورند، به ATM امید بستند. اما این امید دیری نپایید، و شرکت های اینترنتی (حتی سرسخت ترین آنها) بزودی دریافتند که تا رسیدن به سرویس های مطلوب راه درازی در پیش دارند. البته ATM از OSI بسیار موفقتر بود، و حتی امروز هم در شبکه های تلفن (و برای انتقال بسته های IP) مورد استفاده قرار می گیرد. از آنجائیکه این زیرشبکه فقط برای ارتباطات داخلی بکار می رود، اغلب کاربران معمولی از وجود آن اطلاعی ندارد، ولی ATM زنده و سر حال است.

مدار مجازی ATM

از آنجائیکه شبکه های ATM از نوع اتصال-گرا هستند، برای برقراری ارتباط اولیه ابتدا باید یک بسته خاص بفرستند. با عبور این بسته از زیرشبکه، تمام مسیر یاب هایی که در مسیر آن قرار دارند، آنرا در جدول های خود ثبت می کنند و منابع لازم را برای آن کنار می گذارند. به ارتباطی که بدین طریق برقرار می شود، مدار مجازی (virtual circuit) می گویند، چون بسیار شبیه مدار های فیزیکی در شبکه های تلفن است (شکل ۱-۳۰ را ببینید). بسیاری از شبکه های ATM از مدار های مجازی دائمی بین دو نقطه پشتیبانی می کنند (که بسیار شبیه خطوط اجاره ای در سیستم تلفن معمولی است). هر اتصال (موقت یا دائم) دارای یک شماره شناسایی است. بعد از برقراری ارتباط، دو طرف می توانند شروع به فرستادن داده کنند. ایده اصلی در ATM ارسال داده ها در بسته های کوچک و با اندازه ثابت، بنام سلول (cell)، است. هر سلول ۵۳ بایت طول دارد، که ۵ بایت آن سرآیند، و



شکل ۱-۳۰. یک مدار مجازی.

بایت	5	48
سرآیند	داده های کاربر	

شکل ۱-۳۱. یک سلول ATM.

۴۸ بایت باقیمانده داده هاست (شکل ۱-۳۱). شماره شناسایی اتصال در سرآیند سلولها نوشته می شود، بطوریکه تمام مسیرهای مسیریابی می توانند تشخیص دهند که هر سلول متعلق به کدام اتصال است، و چگونه باید آنرا هدایت کنند. هدایت سلولها بصورت سخت افزاری (و با سرعت فوق العاده بالا) صورت می گیرد - در واقع، علت اصلی اندازه ثابت سلولها در شبکه های ATM اینست که ساخت مسیرهای سخت افزاری برای آن بسیار ساده است (هدایت بسته های IP با اندازه متغیر به مسیرهای نرم افزاری نیاز دارد، که بسیار کندتر هستند).

مزیت دیگر ATM توانایی آن در ارسال همزمان یک سلول به مسیرهای مختلف است - که این ویژگی در سیستمهای پخش تلویزیونی بسیار مفید است. از طرف دیگر، کوچک بودن سلولها باعث می شود تا هیچ خطی برای مدت طولانی اشغال نشود، و کیفیت سرویس افزایش یابد.

در ATM تمام سلولها از یک مسیر به مقصد هدایت می شوند. البته تضمینی برای رسیدن یک سلول به مقصد وجود ندارد، ولی ترتیب آنها حتماً رعایت می شود. اگر سلول ۲ بعد از سلول ۱ فرستاده شده باشد، به همان ترتیب به مقصد می رسند، و هرگز سلول ۲ پیش از سلول ۱ به مقصد نخواهد رسید. اگر هر یک از این سلولها (با هر دوی آنها) در بین راه از بین بروند، این بر عهده پروتکل های لایه های بالاتر است که آنها را بازیابی کنند. از این نظر ATM حداقل یک پله بالاتر از اینترنت می ایستد (که نه ترتیب بسته ها ضمانت می شود، نه حتی رسیدن صحیح و سالم آنها).

سازماندهی شبکه های ATM شبیه WAN های قدیمی (متشکل از خطوط تلفنی و سونیچها) است. متداولترین سرعتها در شبکه های ATM عبارتند از 155-Mbps و 622-Mbps (البته ATM از سرعت های بالاتر هم پشتیبانی می کند). علت انتخاب سرعت 155-Mbps آنست که تلویزیونهای با وضوح بالا (HDTV) به چنین سرعتی نیاز دارند - مقدار دقیق این سرعت 155.52-Mbps است، که دقیقاً معادل سرعت سیستم AT&T SONET می باشد. علت انتخاب سرعت 622-Mbps نیز اینست که از ترکیب چهار کانال 155.52-Mbps یک کانال 622-Mbps بوجود می آید.

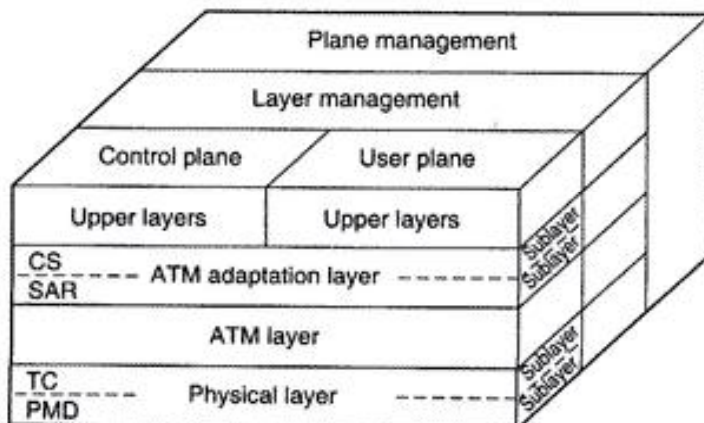
مدل مرجع ATM

شبکه ATM برای خود یک مدل مرجع مستقل دارد، که با مدل های OSI و TCP/IP فرق دارد - این مدل را در شکل ۱-۳۲ ملاحظه می کنید. این مدل سه لایه دارد: لایه فیزیکی، لایه ATM، لایه انطباق ATM (و هر چند لایه که کاربر مایل باشد روی این لایه ها سوار کند).

لایه فیزیکی با مشخصات فیزیکی سیستم (ولتاژها، زمان بندی بیت ها و غیره) سروکار دارد. مدل ATM هیچ پشتیبانی در مورد این مشخصات ندارد، و می گوید که ارسال سلولها می تواند بصورت مستقیم یا از طریق سیستمهای انتقال دیگر انجام شود. بعبارت دیگر، ATM مستقل از سیستم انتقال است.

لایه ATM با خود سلولها و انتقال آنها سروکار دارد. ایجاد و رها کردن مدار مجازی، تعریف فیلدهای سرآیند سلول، و کنترل ازدحام (congestion control) از وظایف این لایه است.

از آنجائیکه اکثر برنامه های کاربردی تمایلی به کار کردن با بسته هایی به کوچکی سلولهای ATM ندارند، یک لایه دیگر بالای لایه ATM تعبیه شده تا این قبیل برنامه ها بتوانند بسته های بزرگتری به ATM بفرستند. این لایه



CS: زیر لایه همگرایی
SAR: زیر لایه قطعه بندی و مونتاژ
TC: زیر لایه همگرایی انتقال
PMD: زیر لایه فیزیکی

شکل ۱-۳۲. مدل مرجع ATM.

(در سمت فرستنده) بسته های داده را به سلولهای ۵۳ بیتی می شکند، و در طرف گیرنده آنها را دوباره سر هم می کند. نام این لایه، لایه انطباق (ATM Adaptation Layer - AAL) است. بر خلاف مدل های قبلی که دو بُعدی بودند، مدل ATM یک مدل مرجع سه بُعدی است (شکل ۱-۳۲ را ببینید). صفحه کاربر (user plane) با انتقال داده، کنترل جریان، تصحیح خطا، و دیگر عملکردهای کاربر سروکار دارد. از طرف دیگر، صفحه کنترل (control plane) مدیریت اتصال را بر عهده دارد. وظیفه صفحه های مدیریت لایه (layer management) و مدیریت صفحه (plane management) مدیریت منابع سیستم و هماهنگ کردن لایه های بینایی است.

لایه های فیزیکی و AAL هر یک به دو زیر لایه تقسیم شده اند، یکی در پائین برای انجام عملکردهای محوله، و دیگری در بالا برای ارتباط با لایه بالاتر (که زیر لایه همگرایی - convergence sublayer - خوانده می شود). وظیفه هر یک از این لایه ها و زیر لایه ها را در شکل ۱-۳۳ ملاحظه می کنید.

OSI لایه	ATM لایه	ATM زیر لایه	کارکرد
3/4	AAL	CS	واسط استاندارد
		SAR	قطعه بندی و مونتاژ
2/3	ATM		کنترل جریان تولید سرآیند سلول مدار مجازی - مدیریت مسیر مالی پلکس/مالی پلکس سلول
2	Physical	TC	ایزوله کردن سرعت سلول تولید مجموع تطبیقی تولید سلول بسته بندی و باز کردن بسته ها تولید فریم
1		PMD	زمان بندی بیت دسترسی فیزیکی

شکل ۱-۳۳. وظایف لایه ها و زیر لایه های ATM.

زیرلایه PMD (Physical Medium Dependent) مستقیماً به کابل شبکه وصل می‌شود، و کار آن ارسال و دریافت بیت‌ها و ایجاد همزمانی بین آنهاست. هر نوع کابل و سیستم انتقال زیرلایه PMD خاص خود را دارد. زیرلایه دیگر لایه فیزیکی، TC (Transmission Convergence) نام دارد. وقتی یک سلول ارسال می‌شود، لایه TC آنرا بصورت جریانی از بیت‌ها به لایه PMD می‌فرستد - که این کاری ساده است. در طرف گیرنده، لایه TC باید جریان بیت‌هایی را که از لایه PMD دریافت می‌کند، دوباره بصورت سلول در آورد - بعبارت دیگر باید بتواند ابتدا و انتهای هر سلول را بدرستی تشخیص دهد. در مدل ATM این کار در لایه فیزیکی انجام می‌شود، وظیفه‌ی که در مدل OSI (و تقریباً تمام مدل‌های دیگر) بر عهده لایه پیوند داده است.

مانطور که قبلاً هم گفتیم، لایه ATM مدیریت ایجاد و انتقال سلولها را بر عهده دارد. مهمترین بخش از وظایف ATM نیز در همین لایه صورت می‌گیرد. این لایه تلفیقی است از لایه‌های لینک داده و شبکه در مدل OSI - که در ضمن هیچ زیرلایه‌ای هم ندارد.

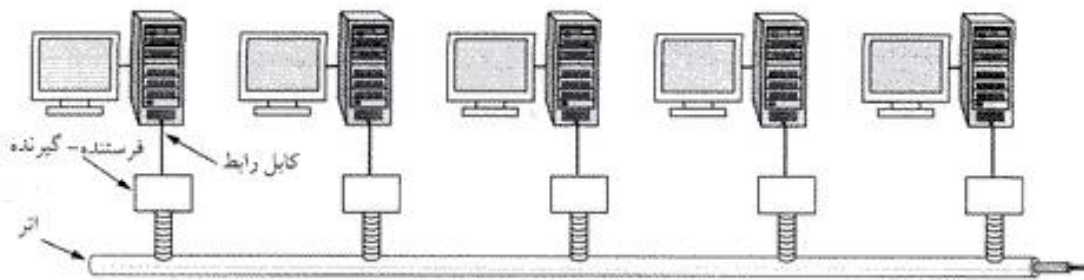
لایه AAL به دو زیرلایه SAR (Segmentation And Reassembly) و CS (Convergence Sublayer) تقسیم شده است. لایه پائینی (SAR) در طرف فرستنده بسته‌های داده را به سلول می‌شکند، و در طرف گیرنده دوباره آنها را به هم می‌چسباند. لایه بالایی (CS) وظیفه ارائه سرویسهای مختلف به برنامه‌های کاربردی را بر عهده دارد.

از آنجائیکه ATM در سراسری زوال قرار دارد، در این کتاب بیش از این درباره آن صحبت نخواهیم کرد. با این حال، بدلیل نصب در مقیاس وسیع، به احتمال زیاد تا چند سال دیگر نیز دوام خواهد آورد. برای کسب اطلاعات بیشتر درباره ATM به (Dobrowski and Grise, 2001; Gadecki and Heckart, 1997) مراجعه کنید.

۳-۵-۱ اینترنت

اینترنت و ATM هر دو شبکه‌های گسترده هستند، ولی در هر شرکت، سازمان و دانشگاه تعداد زیادی کامپیوتر وجود دارد که باید به هم متصل شوند. از همین جاست که نیاز به شبکه‌های محلی شکل می‌گیرد. در این قسمت می‌خواهیم کمی درباره متداولترین شبکه محلی، یعنی اینترنت (Ethernet)، صحبت کنیم. داستان ما از ایالت بکر و دست نخورده‌هاوایی، در اوایل دهه ۱۹۷۰، شروع می‌شود - در اینجا منظور از «بکر و دست نخورده» فقدان شبکه تلفن است. با اینکه نبود شبکه تلفن برای کسانی که هاوایی را برای استراحت انتخاب می‌کردند، یک مزیت بود، ولی برای محقق بنام نورمن آبرامسون و همکارانش در دانشگاه هاوایی که می‌خواستند کاربران جزایر دورافتاده هاوایی را به کامپیوتر مرکزی در هونولولو (مرکز ایالت هاوایی) متصل کنند، چندان خوشایند نبود. کشیدن کابل از وسط اقیانوس آرام مسلماً نمی‌توانست راه حل مشکل آنها باشد، پس باید فکر دیگری می‌کردند.

یکی از راه‌حل‌هایی که آنها پیدا کردند، استفاده از امواج رادیویی با بُرد کوتاه بود. هر ترمینال به یک رادیوی کوچک وصل می‌شد، که دو فرکانس داشت: فرکانس ارسال (به کامپیوتر مرکزی)، فرکانس دریافت (از کامپیوتر مرکزی). وقتی کاربر می‌خواست به کامپیوتر مرکزی وصل شود، روی فرکانس ارسال یک بسته می‌فرستاد. اگر کس دیگری در همان لحظه در حال ارسال نبود، بسته مزبور به کامپیوتر مرکزی می‌رسید، و کاربر می‌توانست بسته تصدیق دریافت (acknowledgement) آنرا روی فرکانس دریافت بگیرد. اما اگر کانال اشغال بود، ترمینال بسته تصدیق دریافت را نمی‌گرفت، و متوجه می‌شد که باید دوباره سعی کند. از آنجائیکه روی کانال دریافت فقط یک کامپیوتر (کامپیوتر مرکزی) مجاز به ارسال اطلاعات بود، هرگز در آن انسداد پیش نمی‌آمد. این سیستم، که آلوهانت (ALOHA NET) نام گرفت، در شرایط ترافیک پائین خوب کار می‌کرد، ولی در ترافیک بالا بشدت ناکارآمد می‌شد.



شکل ۱-۳۴. معماری اترنت اولیه.

در همان زمان، دانشجویی بنام باب متکالف که تازه از M.I.T فارغ التحصیل شده بود، به قصد ادامه تحصیل در مقطع دکترا وارد دانشگاه هاروارد شد. باب در حین مطالعات خود با کارهای آبرامسون آشنا، و بشدت به آن علاقمند شد. این علاقه تا آن حد بود که باب تصمیم گرفت قبل از شروع به کار در مرکز تحقیقات زیراکس در پالو آلتو (Xerox PARC)، تابستان را در هاوایی بگذراند و با آبرامسون کار کند. وقتی باب متکالف به Xerox PARC برگشت، متوجه شد که محققان آنجا روی پروژه های کار می کنند که بعدها به کامپیوتر شخصی (Personal Computer - PC) معروف شد. ولی این یک ماشین ایزوله و جدا از همه جا بود. باب، با استفاده از تجارب آبرامسون، و به کمک یکی از همکارانش بنام دیوید باگز، اولین شبکه محلی را طراحی و پیاده سازی کرد (Metcalfe and Boggs, 1976).

آنها این سیستم را (بیاد ماده ای خیالی بنام اتر - ether - که تا مدتها تصور می شد محیط انتشار امواج الکترومغناطیس است) اترنت نامیدند. (بعد از کشف معادلات انتشار امواج الکترومغناطیس توسط فیزیکدان بریتانیایی، جیمز کلارک ماکسول، دانشمندان فرض را بر این گذاشتند که این امواج در محیطی بنام اتر منتشر می شوند. فقط بعد از آزمایشات معروف مایکلسون-مورلی بود که فیزیکدانان دریافتند امواج الکترومغناطیس می توانند در خلأ منتشر شوند.)

رسانه انتشار در این سیستم خلأ نبود، بلکه از یک رشته کابل هم محور (coaxial) ضخیم بطول حداکثر ۲/۵ کیلومتر (با یک تکرار کننده در هر ۵۰۰ متر) استفاده می شد (این کابل در واقع همان اتر محسوب می شد) - شکل ۱-۳۴ را ببینید. حداکثر تا ۲۵۶ کامپیوتر را می شد به این کابل متصل کرد. به چنین کابلی، که چندین ماشین بصورت موازی به آن متصل شده اند، کابل چنداتصال (multidrop cable) گفته می شود. اترنت با سرعت 2.94 Mbps کار می کرد. اترنت یکی از اشکالات عمده آلو هانت را نیز برطرف کرده بود: هر کامپیوتر قبل از ارسال بسته خود ابتدا به کابل گوش می کرد، تا مطمئن شود کس دیگری در همان لحظه در حال استفاده از آن نیست. اگر چنین بود، کامپیوتر تا خالی شدن خط کار خود را عقب می انداخت. بدین ترتیب، اترنت توانست با اجتناب از تداخل های بی مورد به کارایی بالاتری دست یابد. البته آلو هانت امکان انجام چنین کاری را نداشت، چون فرستنده یک جزیره نمی توانست امواج فرستنده های جزایر دیگر را بشنود.

علیرغم اینکه کامپیوترهای اترنت تا وقتی خط خالی نباشد، اقدام به ارسال نمی کنند، اما مسئله دیگری ممکنست بروز کند: اگر چند کامپیوتر هم آن منتظر خالی شدن خط باشند، و به محض اینکه خط خالی شد، در یک لحظه شروع به ارسال اطلاعات خود کنند، چه خواهد شد؟ راه حل این مشکل آن است که هر کامپیوتر در تمام لحظات ارسال اطلاعات خود به خط گوش کند، و اگر متوجه تداخل امواج شد، ابتدا به دیگران اخطار می فرستد، و سپس برای مدتی کوتاه (که مقدار آنرا بطور تصادفی تعیین می کند) کنار می کشد، و بعد از این مدت دوباره سعی

می‌کند؛ اگر در مرتبه بعد باز هم تداخل پیش آمد، مدت انتظار را دو برابر می‌کند، تا بالاخره یکی از آنها فرصت ارسال امواج را پیدا کند.

ایترنت زیراکس چنان موفق بود که در سال ۱۹۷۸ شرکت‌های DEC، ایستل و زیراکس استاندارد ی بنام DIX برای اینترنت 10-Mbps وضع کردند. استاندارد DIX در سال ۱۹۸۳ با دو تغییر جزئی به استاندارد IEEE 802.3 تبدیل شد.

متأسفانه زیراکس همواره یکی از شرکت‌هایی بوده است که اختراعات بسیار مهمی (مانند PC، اینترنت و ماوس) ز آنجا منشأ گرفته، ولی نتوانسته اقدامی برای تجاری کردن آنها بعمل آورد. وقتی زیراکس علاقه چندانی به اینترنت نشان نداد (و فقط در استاندارد کردن آن همکاری کرد)، باب متکالف شرکتی بنام 3Com تأسیس کرد تا بتواند برای PC ها کارت شبکه اینترنت تولید کند.

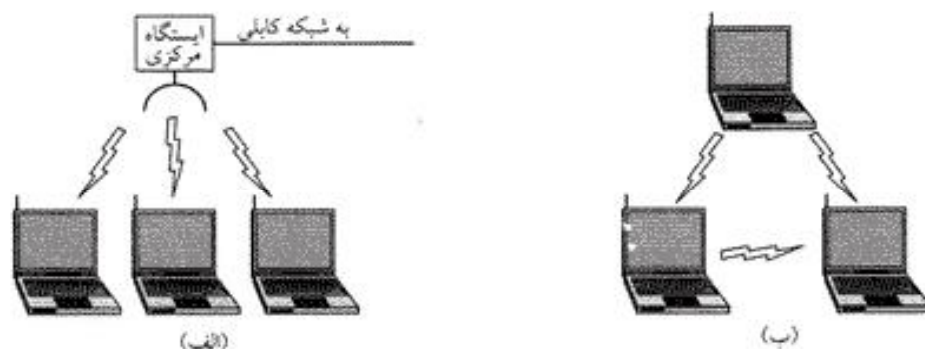
اینترنت به رشد و توسعه خود ادامه داد (رشدی که همچنان ادامه دارد)، و امروزه به سرعت‌های 100-Mbps و 1000-Mbps دست یافته است - سرعتی که انتظار می‌رود باز هم افزایش یابد. کابل کشی، سوئیچینگ و سایر جنبه‌های اینترنت نیز بهبود یافته، و ویژگی‌های جدیدی به آن اضافه شده است. در فصل ۲ مفصلاً درباره اینترنت صحبت خواهیم کرد.

البته همین جا لازم به توضیح است که اینترنت (IEEE 802.3) تنها استاندارد LAN نیست. استانداردهای خط توکین (IEEE 802.4 - Token Bus) و حلقه توکین (IEEE 802.5 - Token Ring) نیز از جمله استانداردهای معروف LAN هستند. وجود سه استاندارد کمابیش ناسازگار برای شبکه‌های محلی بیشتر از اینکه مسئله‌ای فنی باشد، موضوعی سیاسی است. در همان زمان که اینترنت در حال استاندارد شدن بود، جنرال موتورز نیز در کار ایجاد شبکه‌ای بود که از همان توپولوژی اینترنت (یعنی کابل خطی) استفاده می‌کرد، ولی نوبت ارسال هر کامپیوتر با استفاده از بسته خاصی بنام توکین، که بین کامپیوترها دست به دست می‌گشت، تعیین می‌شد. هر کامپیوتر فقط زمانی می‌توانست اقدام به ارسال اطلاعات کند که توکین را در اختیار داشته باشد، و بدین ترتیب مشکل تداخل حل می‌شد. جنرال موتورز اعلام کرد که این تکنیک برای خط تولید کارخانجات اتومبیل سازی ضرورت مطلق دارد، و حاضر نبود حتی یک میلیمتر از موضع خود عقب‌نشینی کند. با وجود چنین ادعایی، 802.4 اکنون از صفحه روزگار محو شده است.

غول صنعت کامپیوتر، IBM هم سوگلی خود را داشت: حلقه توکین. تنها تفاوت این سیستم با شبکه جنرال موتورز آن بود که در اینجا کابل شبکه یک مسیر بسته (حلقه) را تشکیل می‌داد. بر خلاف 802.4، 802.5 هنوز در برخی از سایت‌های IBM مورد استفاده است (ولی خارج از IBM هیچکس از آن استفاده نمی‌کند). تحقیقاتی در زمینه نسل جدید و پرسرعت حلقه توکین با سرعت گیگابیت (802.5v) در جریان است، ولی بنظر نمی‌رسد بتواند با اینترنت رقابت کند. خلاصه اینکه، در جنگی که بین اینترنت، خط توکین و حلقه توکین در گرفت، اینترنت پیروز شد، چون اولین و در ضمن از رقبایش بهتر بود.

۵-۱ شبکه‌های محلی بیسیم: 802.11

تقریباً همزمان با به بازار آمدن کامپیوترهای کتابی، بسیاری افراد این رؤیا را در سر می‌پروراندند که بتوانند به محض ورود به جایی که دسترسی اینترنت وجود دارد، بلافاصله و بنحوی جادویی کامپیوترشان به اینترنت متصل شود - و همیشه وقتی رؤیایی وجود دارد، افرادی هم هستند که به فکر محقق کردن آن بیفتند. عملی ترین رهیافتی که برای به فعل در آوردن این ایده وجود داشت، مجهز کردن کامپیوترها به فرستنده-گیرنده‌های راد، بی بُرد کوتاه بود - از همین جا بود که شرکت‌های متعددی بسرعت شبکه‌های محلی بیسیم را وارد بازار کردند.



شکل ۱-۳۵. شبکه بیسیم (الف) با ایستگاه مرکزی، و (ب) بدون ایستگاه مرکزی.

مشکل اصلی این بود که هیچکدام از این شبکه های بیسیم با هم سازگار نبودند، و کامپیوترهایی که به بیسیم های مختلف مجهز بودند، نمی توانستند با هم ارتباط برقرار کنند. بالاخره همه به این نتیجه رسیدند که استاندارد کردن شبکه های بیسیم می تواند ایده خوبی باشد، و بدنبال آن کمیته استاندارد IEEE مأمور تدوین این استاندارد شد - استاندارد 802.11 نام گرفت، و در میان عموم به WiFi معروف است. این یکی از استانداردهای مهم صنعت کامپیوتر است، و شایسته توجه کافی.

استاندارد پیشنهاد شده باید در دو حالت کار می کرد:

۱. در شرایط وجود یک ایستگاه مرکزی

۲. در شرایط فقدان ایستگاه مرکزی

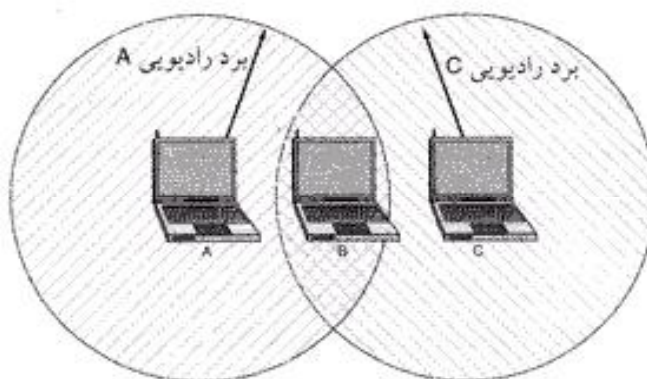
در حالت اول، تمام پیامها باید از طریق ایستگاه مرکزی، که در استاندارد 802.11 به آن نقطه دسترسی (access point) گفته می شود، مبادله شوند. اما در حالت دوم، کامپیوترها مستقیماً با یکدیگر ارتباط برقرار می کنند. این دو حالت را در شکل ۱-۳۵ مشاهده می کنید.

شروع می شود، و اعداد 1 تا 10 قبلاً استفاده شده بود، نام این استاندارد 802.11 شد. ولی بقیه کار به همین سادگی نبود.

برخی از مهمترین چالشهایی که کمیته تدوین استاندارد با آنها روبرو بود، عبارت بودند از: انتخاب یک باند فرکانسی مناسب (که ترجیحاً بین المللی نیز باشد)؛ محدود بودن بُرد سیگنالهای رادیویی؛ تأمین ایمنی مناسب؛ عمر محدود باتری در کامپیوترهای کتابی؛ مسائل بهداشتی (هنوز این مسئله بدرستی روشن نشده که آیا امواج رادیویی سرطانزا هستند یا خیر)؛ پیامدهای متحرک بودن کامپیوترها؛ و بالاخره، ایجاد سیستمی که از نظر پهنای باند ارزش اقتصادی داشته باشد.

با توجه به غالب بودن اینترنت در زمان تدوین این استاندارد، کمیته تصمیم گرفت که 802.11 باید در لایه های بالاتر از لایه پیوند داده با اینترنت سازگار باشد. بویژه، ارسال بسته های IP در شبکه های بیسیم بایستی دقیقاً به همان روش اینترنت باشد.

با این وجود، لایه های فیزیکی و لینک داده در این دو سیستم تفاوت های اساسی با هم دارند. اول اینکه، در اینترنت هر کامپیوتر قبل از شروع به ارسال اطلاعات به اتر (کابل شبکه) گوش می کند، و فقط در صورت خالی بودن آن شروع به ارسال می کند. در شبکه های بیسیم کار به همین سادگی نیست. برای درک علت آن، به شکل ۱-۳۶ نگاه کنید. فرض کنید کامپیوتر A در حال ارسال اطلاعات به کامپیوتر B است، ولی بُرد امواج آن به کامپیوتر C نمی رسد. اگر در این لحظه C بخواهد چیزی به B بفرستد، باید به اتر (در اینجا، فضا) گوش کند، ولی آیا عدم



شکل ۱-۳۶. گاهی بُرد امواج رادیویی برای پوشش دادن به تمام شبکه کافی نیست.

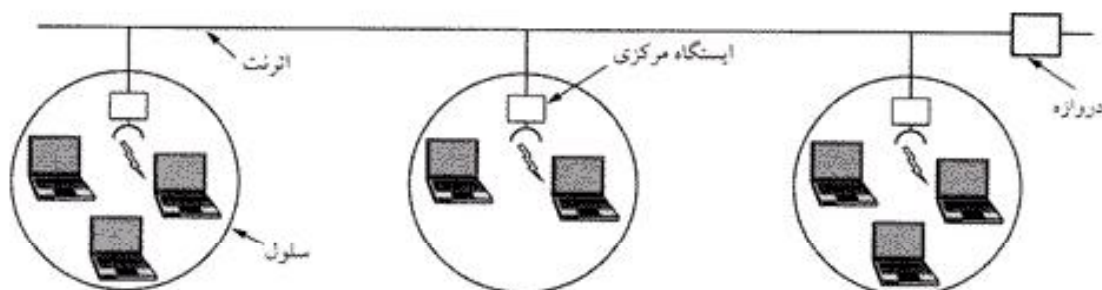
دریافت امواج به معنای آن است که می‌تواند با اطمینان شروع به ارسال کند؛ مسلماً خیر. استاندارد 802.11 باید این مسئله را حل می‌کرد.

مسئله دومی که باید حل می‌شد این بود که اجسام سخت امواج رادیویی را منعکس می‌کنند، و یک موج می‌تواند چندین بار (و از مسیرهای مختلف) به گیرنده برسد. این تداخل امواج باعث بروز حالتی می‌شود که به آن محوشدگی چندمسیره (multipath fading) می‌گویند.

مسئله سوم این است که بسیاری از نرم‌افزارهای موجود اساساً با چنین وضعیتی (متحرک بودن کامپیوتر) آشنا نیستند. برای مثال، بسیاری از برنامه‌ها دارای لیستی از چاپگرها هستند که می‌توانند روی آنها چاپ کنند. وقتی یکی از این برنامه‌ها (که روی یک کامپیوتر کتابی نصب شده) وارد محیط جدیدی می‌شود، نمی‌تواند تشخیص دهد که فهرست قبلی چاپگرها دیگر در این وضعیت اعتبار ندارد.

مسئله چهارم این است که وقتی یک کامپیوتر از بُرد یک ایستگاه مرکزی خارج و وارد محدوده ایستگاه دیگری می‌شود، باید مکانیزمی برای این جابجایی وجود داشته باشد. با اینکه تلفنهای همراه دارای چنین مکانیزمی هستند، اما این اتفاق در اینترنت نمی‌افتد، و باید بطریقی حل شود. یکی از راه‌حل‌های این مسئله، متصل کردن ایستگاههای مرکزی به یکدیگر از طریق کابل است (شکل ۱-۳۷ را ببینید). از دید دنیای خارج، این شبکه کاملاً شبیه یک شبکه اینترنت واحد است. نقطه اتصال سیستم 802.11 با دنیای خارج را درگاه (portal) می‌گویند.

بعد از مدتی کار سخت، کمیته موفق به تدوین استانداردی شد که این مسائل (و بسیاری مسائل دیگر) در آن حل شده بود. این شبکه بیسیم با سرعتهای 1-Mbps و 2-Mbps کار می‌کرد. تقریباً بلافاصله، همه لب به شکایت گشودند که این سرعتها بسیار کم است، و کار بر روی استانداردهای سریعتر آغاز شد. در همین زمان کمیته استاندارد به دو قسمت تقسیم شد، که در سال ۱۹۹۹ هر کدام استاندارد جداگانه‌ای وضع کردند. استاندارد



شکل ۱-۳۷. یک شبکه 802.11 چندسلولی.

802.11a از باند فرکانسی وسیعتری نسبت به 802.11 استفاده می کند، و سرعت آن به 54-Mbps می رسد. استاندارد 802.11b در همان باند فرکانسی 802.11 کار می کند، ولی با استفاده از مدولاسیون متفاوت به سرعت 11-Mbps دست می یابد. همانطور که می بینید، هر دوی این استانداردها از 802.11 (و حتی از اینترنت اولیه) سریعترند، ولی هنوز بدرستی معلوم نیست که کدامیک از آنها برنده نهایی این مسابقه اند. برای اینکه اوضاع از این هم پیچیده تر شود، کمیته 802.11 استاندارد جدیدی بنام 802.11g تدوین کرده، که در باند فرکانسی 802.11b، ولی با مدولاسیون 802.11a، کار می کند. در فصل ۴ مفصلاً درباره 802.11 صحبت خواهیم کرد.

شکی نیست که 802.11 انقلاب جدیدی در دنیای کامپیوتر و اینترنت بپا کرده است. فرودگاهها، ایستگاههای قطار، بنادر، هتلها، فروشگاهها، دانشگاهها، مراکز آموزشی (و حتی کافه های کوچک) بسرعت در حال نصب شبکه های 802.11 هستند. در واقع، 802.11 همان چیزی را به اینترنت داده است، که کامپیوترهای کتابی به دنیای کامپیوتر دادند: تحرک پذیری.

۶-۱ استانداردهای شبکه

تعداد زیادی سازنده و تأمین کننده قطعات و تجهیزات شبکه وجود دارد، که فکر می کنند می دانند چگونه باید کار خود را انجام دهند. اما بدون یک عامل هماهنگ کننده، این وضعیت می تواند به یک آشوب واقعی بینجامد، و در نهایت هیچ کاری هم انجام نشود. تنها راه برای خلاصی از چنین وضعیتی، توافق بر سر استانداردهای شبکه است. استانداردها نه تنها اجازه می دهد تا تجهیزات مختلف بتوانند با هم کار کنند، بلکه فروش محصولات منطبق با استاندارد را نیز افزایش می دهند - و فروش بیشتر یعنی تولید انبوه، کاهش هزینه ها، طراحی بهتر، کاهش قیمت ها، و افزایش مجدد درخواست (و مگر اقتصاد چیزی غیر از این است). در این قسمت نگاهی به استانداردهای بین المللی (که از اهمیت زیادی برخوردارند، ولی کمتر شناخته شده اند) خواهیم داشت.

استانداردها بر دو نوعند: استانداردهای بالفعل (de facto)، و استانداردهای قانونی (de jure). استانداردهای بالفعل آنهایی هستند که بدون هیچ طرح رسمی بوجود آمده و پذیرفته شده اند. کامپیوترهای سازگار با IBM PC (که به اختصار PC خوانده می شوند) از جمله استانداردهای بالفعل هستند، چون شرکتهای بسیاری تصمیم گرفتند تا کپی های دقیق و کاملی از این نوع کامپیوتر بسازند. یکی دیگر از استانداردهای بالفعل، سیستم عامل یونیکس (UNIX) است، که در دانشکده های کامپیوتر بعنوان سیستم عامل استاندارد پذیرفته شده است. از طرف دیگر، استانداردهای قانونی آنهایی هستند که توسط مراجع مسئول بین المللی پذیرفته شده اند. مراجع بین المللی استاندارد به دو دسته تقسیم می شوند: آنهایی که طبق معاهدات بین المللی تأسیس شده اند، و آنهایی که بصورت داوطلبانه شکل گرفته اند. در زمینه استانداردهای شبکه های کامپیوتری، سازمانهایی از هر دو دسته وجود دارند، که در زیر آنها را معرفی خواهیم کرد.

۱۶-۱ مراجع مسئول استانداردهای مخابرات

وضعیت قانونی شرکتهای تلفن از کشوری به کشور دیگر بطرز چشمگیری متفاوت است. در یک سو ایالات متحده آمریکا قرار دارد، که در آن متجاوز از ۱۵۰۰ شرکت خصوصی در این زمینه مشغول به کار هستند. شرکت AT&T، قبل از آنکه در سال ۱۹۸۴ طبق قانون ضد تراست به دو بخش تقسیم شود، بزرگترین شرکت خصوصی دنیا بود، و مخابرات این کشور را تحت سلطه کامل خود داشت. این شرکت سرویس تلفن ۸۰ درصد مناطق آمریکا را تأمین می کرد، در حالیکه بقیه شرکتهای تنها ۲۰ درصد بازار (آن هم اغلب در مناطق روستایی) را در اختیار داشتند. بعد از تقسیم AT&T، این شرکت اجازه یافت تا فقط در مخابرات راه دور فعالیت کند (آن هم در رقابت با

سایر شرکتها). هفت شرکت منطقه ای بل (Bell)، که از AT&T منشعب و با شرکت های دیگر متحد شده بودند، نیز خدمات تلفن شهری و تلفن همراه را بر عهده گرفتند. بدلیل ادغام ها و انشعابات متعدد، صنعت مخابرات در ایالات متحده اکنون از وضعیت ثابتی برخوردار است.

در ایالات متحده، به شرکت هایی که سرویس های مخابراتی در اختیار عموم قرار می دهند، کاربر عمومی گفته می شود. نوع سرویس ها و تعرفه خدمات این شرکتها توسط کمیسیون فدرال مخابرات (برای تماس های بین ایالتی و بین المللی) و کمیسیون های ایالتی (برای تماس های داخل ایالتی) تعیین می شود.

در سوی دیگر، کشورهایی قرار دارند که در آنها تمامی سرویس های مخابراتی (از جمله، پست، تلگراف، تلفن، و حتی رادیو و تلویزیون) تحت انحصار مطلق دولت قرار دارد. اکثر کشورهای دنیا هم از این دسته اند. در برخی از این کشورها، مخابرات را یک شرکت ملی اداره می کند، و در برخی دیگر دولت مستقیماً (از طریق وزارتخانه ای بنام پ.ت.ت: پست-تلگراف-تلفن) کارها را در دست دارد. اما، در کل دنیا حرکت یکپارچه ای به سمت آزادسازی، رقابت و حذف انحصار دولت آغاز شده است. امروزه اکثر کشورهای اروپایی مخابرات خود را به بخش خصوصی سپرده اند، و در دیگر کشورها این فرآیند (هر چند کند و بطنی) ادامه دارد.

با این همه شرکت و سرویس های مخابراتی، وجود نوعی استاندارد بین المللی برای تضمین ارتباط بین افراد (و کامپیوترها) از کشوری به کشور دیگر ضروری بنظر می رسد. البته این نیاز از مدتها قبل آشکار شده بود: در سال ۱۸۶۵، نمایندگانی از کشورهای مختلف اروپایی آنچه را که امروز ITU (اتحادیه بین المللی مخابرات - International Telecommunication Union) نامیده می شود، بنا نهادند. وظیفه این اتحادیه استاندارد کردن مخابرات بین المللی (که در آن زمان فقط شامل تلگراف می شد) بود. حتی در آن زمان نیز مشخص بود که اگر کلیه کشورها در مورد استفاده از یک کد یکسان (که همان کد مورس بود) به توافق نرسند، مشکلات عدیده ای بروز خواهد کرد. با ورود تلفن به صحنه مخابرات بین المللی، ITU وظیفه استاندارد کردن آنرا نیز بر عهده گرفت. در سال ۱۹۴۷، ITU بصورت یکی از سازمان های تابعه سازمان ملل متحد درآمد. اتحادیه بین المللی مخابرات (ITU) سه بخش عمده دارد:

۱. بخش مخابرات رادیویی (ITU-R)
۲. بخش تدوین استانداردهای مخابراتی (ITU-T)
۳. بخش توسعه (ITU-D)

وظیفه ITU-R تخصیص فرکانس های رادیویی به متقاضیان در سراسر دنیاست. بخش ITU-T (که بیشتر مد نظر ماست) وظیفه تدوین استاندارد برای سیستم های مخابراتی (تلفن و داده) را بر عهده دارد. از سال ۱۹۵۳ تا ۱۹۹۳، ITU-T با نام CCITT (که از نام فرانسوی آن - Comité Consultatif International Télégraphique et Téléphonique - گرفته شده بود) شناخته می شد. در اول مارس ۱۹۹۳، CCITT برای کاهش بوروکراسی ساختار اداری اش را تغییر داد، و نام خود را نیز ITU-T گذاشت. ITU-T و CCITT هر دو توصیه هایی را در زمینه مخابرات تلفنی و داده منتشر می کردند. امروزه نیز افراد بسیاری به توصیه های CCITT مراجعه می کنند، اگر چه از سال ۱۹۹۳ این توصیه ها نام ITU-T را بر خود دارند. اعضای ITU-T به چهار دسته تقسیم می شوند:

۱. دولت ها
۲. اعضای بخش
۳. اعضای وابسته
۴. نمایندگی ها

تقریباً تمام کشورهای عضو سازمان ملل متحد (یعنی حدود ۲۰۰ کشور) در ITU-T عضویت دارند. از آنجائیکه ایالات متحده آمریکا وزارتخانه ای بنام «پست و تلگراف و تلفن» ندارد، فرد دیگری باید نمایندگی آنرا در ITU-T بر عهده بگیرد، که این وظیفه به وزارت امور خارجه محول شده است (شاید به این دلیل که با کشورهای خارجی سروکار دارد). تعداد اعضای بخش در ITU-T به حدود ۵۰۰ می رسد، که شرکتهای تلفن (مانند AT&T، ودفون، ورلدکام)، تولیدکنندگان تجهیزات مخابراتی (مانند سیسکو، نوکیا، نورتل)، تولیدکنندگان کامپیوتر (مانند کامپک، سان، توشیبا)، سازندگان چیپ های میکروالکترونیک (مانند اینتل، موتورولا، IT)، شرکتهای رسانه ای (مانند AOL Time Warner، CBS، سونی)، و سایر شرکتهای علاقمند (مانند بوئینگ، سامسونگ، زیراکس) از آن جمله اند. تعدادی از سازمانهای علمی غیرانتفاعی و کنسرسیوم های صنعتی (مانند IFIP و IATA) نیز در ITU-T عضو بخش هستند. اعضای وابسته شرکتهای کوچکتري هستند، که به یکی از مباحث خاص ITU-T علاقمند هستند. نمایندگی ها نیز آنهایی هستند که بر امور مخابراتی نظارت دارند، مانند کمیسیون مخابرات فدرال ایالات متحده (FCC).

وظیفه ITU-T ارائه توصیه های فنی در زمینه تلفن، تلگراف و مخابرات داده است. این توصیه ها اغلب بصورت استانداردهای جهانی پذیرفته می شوند، مانند V.24 (در ایالات متحده به نام EIA RS-232 نیز شناخته می شود) که نقش و وظیفه پایه های رابط های مودمها و ترمینالهای آسنکرون را مشخص می کند.

این نکته را باید تذکر داد که ITU-T فقط توصیه های فنی ارائه می کند، و دولتها می توانند آنها را بپذیرند یا نپذیرند (همانطور که می دانید دولتها مثل بچه های ده-یازده ساله هستند، یعنی دوست ندارند کسی به آنها دستور بدهد). البته کشوری که استاندارد غیر از استاندارد سایر کشورها را قبول کند، در عمل خود را از سایر کشورهای دنیا ایزوله کرده است - کاری که شاید فقط کره شمالی مایل به انجام آن باشد. احتمالاً نامگذاری استانداردهای ITU-T بنام «توصیه» فقط کلکی برای آرام کردن ملی گرایان کشورهای مختلف است.

کار واقعی ITU-T توسط ۱۴ گروه مطالعاتی (که اغلب آنها نزدیک به ۴۰۰ نفر عضو دارند) انجام می شود. سرفصل هایی که این گروه های مطالعاتی بررسی می کنند، از استانداردهای تهیه صورتحساب تلفن گرفته تا سرویسهای چندرسانه ای متغیر است. برای اینکه هر گروه بتواند کار خود را بهتر انجام دهد، به چند دسته کاری تقسیم می شود، که هر یک از این دسته ها بنوبه خود به چند تیم کارشناسی، و هر تیم تخصصی نیز به گروه های تخصصی تقسیم می شوند. (مثل اینکه هیچ وقت نمی توان از بوروکراسی خلاص شد!)

علیرغم این بوروکراسی، ITU-T کار خود را بخوبی انجام می دهد، و از اول تأسیس آن تاکنون نزدیک به ۳۰۰۰ توصیه (که بالغ بر ۶۰،۰۰۰ صفحه می شود) تدوین کرده است، که بسیاری از آنها بطور گسترده ای مورد استفاده قرار گرفته اند. برای مثال، استاندارد V.90 (که درباره مودمهای 56-kbps است) از توصیه های ITU-T می باشد.

با رشد و توسعه روزافزون مخابرات راه دور و جهانی شدن آن، استانداردها روز به روز اهمیت بیشتری می یابند، و سازمانهای بیشتری مایلند در تدوین آنها شرکت داشته باشند. برای کسب اطلاعات بیشتر درباره ITU (Irmer, 1994) مراجعه کنید.

۲-۶-۱ مراجع مسئول استانداردهای بین المللی

استانداردهای بین المللی توسط سازمان بین المللی استاندارد (ISO - که البته نام واقعی آن سازمان بین المللی برای تدوین استاندارد، IOS، است) تهیه و منتشر می شوند. ISO یک سازمان غیرپیمانی داوطلبانه است، در سال ۱۹۴۶ تأسیس شده، و دارای ۸۹ عضو می باشد. سازمانهای استاندارد ایالات متحده آمریکا (ANSI)، انگلستان (BSI)، فرانسه (AFNOR)، آلمان (DIN) و هشتاد و پنج کشور دیگر از اعضای ISO هستند.

تنوع استانداردهایی که ISO منتشر می‌کند، واقعاً جالب و حیرت‌آور است - استانداردهایی وجود دارد که کسی حتی فکر استاندارد بودن آنها را نمی‌کند، مانند استاندارد دانه‌های کاکائو (ISO 2451)، تورهای ماهیگیری (ISO 1530)، لباسهای زیر زنانه (ISO 4416) و غیره. تعداد استانداردهای ISO بالغ بر ۱۳,۰۰۰ است (که استانداردهای OSI نیز از آن جمله‌اند). ISO نزدیک به ۲۰۰ کمیته فنی (TC) دارد، که به ترتیب زمان ایجاد شماره‌گذاری شده‌اند و هر کدام در موضوعی خاص تخصص دارند. برای مثال، TC1 با استانداردهای پیچ و مهره سروکار دارد، و TC97 با استانداردهای صنعت کامپیوتر و پردازش اطلاعات. هر TC به چند زیرکمیته (SC)، و هر زیرکمیته به چند گروه کاری (WG) تقسیم می‌شود.

کار اصلی ISO در WG ها (که متجاوز از ۱۰۰,۰۰۰ عضو داوطلب در سراسر دنیا دارند) انجام می‌شود. بسیاری از این «داوطلبان» در استخدام شرکت‌هایی هستند که محصول آنها قرار است استاندارد شود؛ برخی دیگر نیز مقامات رسمی کشورهایی هستند که مایلند روش ساخت محصولات کشورشان بصورت استانداردهای جهانی در آید. در برخی از WG ها مقامات متخصصان دانشگاهی نیز حضور دارند.

در زمینه استانداردهای صنعت مخابرات، ISO و ITU-T اغلب با یکدیگر تشریک مساعی دارند (در واقع، یکی از اعضای ITU-T است)، تا از توسعه استانداردهای ناسازگار اجتناب شود.

نماینده ایالات متحده آمریکا در ISO مؤسسه ملی استانداردهای آمریکا (ANSI) است، که برخلاف نامش یک مؤسسه غیردولتی و غیرانتفاعی است، و اعضای آن عبارتند از تولیدکنندگان، کاربرهای عمومی و شرکت‌های ذینفع. ISO اغلباً استانداردهایی که توسط ANSI وضع می‌شود، را بعنوان استانداردهای بین‌المللی می‌پذیرد. رویه‌ای که در ISO برای پذیرش استانداردها مورد استفاده قرار می‌گیرد، بگونه‌ایست که بیشترین توافق اعضا را بدنبال داشته باشد. این رویه با اعلام نیاز یکی از سازمانهای عضو به یک استاندارد جدید آغاز می‌شود. بدنبال آن یک گروه کاری تشکیل می‌شود، تا برای استاندارد جدید یک پیش‌نویس کمیته (CD) تهیه کند. این CD بین سازمانهای عضو توزیع می‌شود، و آنها شش ماه فرصت دارند تا انتقادات خود را مطرح کنند. اگر اکثریت اعضا با این CD موافق باشند، یک سند اصلاح شده بنام پیش‌نویس استاندارد جهانی (DIS) برای اظهارنظر و رأی‌گیری منتشر می‌شود. بر اساس نتایج بدست آمده از این دور، متن نهایی بعنوان استاندارد جهانی (IS) تنظیم، اصلاح و منتشر می‌شود. در مورد استانداردهایی که اختلاف نظر جدی بین اعضا وجود داشته باشد، امکان دارد CD یا DIS چندین بار اصلاح و به رأی گذاشته شود، تا بتواند رأی کافی بدست آورد، و این فرآیندی است که ممکنست سالها طول بکشد.

در ایالات متحده آمریکا، مؤسسه ملی استانداردها و تکنولوژی (National Institute of Standards and Technology - NIST) که از توابع وزارت بازرگانی است، سازمان ملی استاندارد محسوب می‌شود، و تمام خریدهای دولتی (به استثنای خریدهای نظامی، که استانداردهای خاص خود را دارند) بایستی مطابق استانداردهای آن باشد.

یکی دیگر از بازیگران بزرگ در صحنه استانداردهای جهانی، مؤسسه مهندسان برق و الکترونیک (IEEE) است، که بزرگترین سازمان حرفه‌ای دنیا محسوب می‌شود. علاوه بر انتشار کتب و مجلات متعدد و برگزاری صدها کنفرانس علمی در سال، IEEE یک گروه تدوین استاندارد نیز دارد که در زمینه مهندسی برق و کامپیوتر فعالیت می‌کند. برای مثال، کمیته 802 وظیفه تدوین استاندارد شبکه‌های LAN را در IEEE بر عهده دارد، که گروه‌های کاری آنرا در شکل ۱-۳۸ ملاحظه می‌کنید. میزان موفقیت گروه‌های کاری 802 چندان بالا نیست، و داشتن عدد 802.x تضمینی برای موفقیت یک استاندارد نیست. البته استثنایی هم در این زمینه وجود دارد، که استانداردهای 802.3 و 802.11 از آن جمله‌اند.

شماره	سرفصل
802.1	معماری LAN
802.2 ↓	کنترل لینک منطقی
802.3 *	اترنت
802.4 ↓	باس توکن
802.5	حلقه توکن
802.6 ↓	دو پاس-دو صف
802.7 ↓	گروه مشورتی تکنولوژی های بخشی
802.8 ↑	گروه مشورتی تکنولوژی های فیزیکی
802.9 ↓	LAN ایزو سنکرون (برای کاربردهای زمان واقعی)
802.10 ↓	شبکه مجازی و امنیتی
802.11 *	LAN بی سیم
802.12 ↓	تقدم تقاضا (خاص)
802.13	عدد نحس (کسی آنرا نمی خواهد)
802.14 ↓	مودم کابلی
802.15 *	شبکه های شخصی (بلووث)
802.16 *	بی سیم پاند وسیع
802.17	حلقه بسته برگشتی

شکل ۱-۳۸. گروه های کاری 802. گروه های مهم با * مشخص شده اند. آنهایی که با ↓ مشخص شده اند، به خواب زمستانی رفته اند، و گروه هایی که با ↑ مشخص شده اند، مدتهاست از صفحه روزگار محو شده اند.

۳-۶-۱ مراجع مسئول استانداردهای اینترنت

اینترنت نیز مکانیزم های استاندارد سازی خاص خود را دارد، که با ISO و ITU-T بسیار متفاوت است. مهمترین تفاوت آنها اینست که افرادی که در گردهمایی های ISO و ITU-T شرکت می کنند، کت و شلوار رسمی می پوشند، ولی شرکت کنندگان در نشست های استاندارد سازی اینترنت لباس جین به تن می کنند (البته اگر تابستان نباشد، و محل اجلاس هم کنار دریا نباشد).

در گردهمایی های ISO و ITU-T اغلب نمایندگان شرکها و دولتها (که استاندارد سازی شغل آنهاست) شرکت می کنند. آنها به استاندارد بعنوان «یک چیز آسمانی» نگاه می کنند، و حاضرند از جانشان برای آن مایه بگذارند. از طرف دیگر، اینترنت ها بی نظمی را به عنوان یک اصل پذیرفته اند. اما از آنجائیکه بدون حداقلی از اشتراکات اصولاً امکان برقراری هیچ نوع تماسی وجود ندارد، استاندارد را (در کمال تأسف) چیزی لازم می دانند. وقتی آرپانت بوجود آمد، وزارت دفاع ایالات متحده آمریکا کمیته ای رسمی را مأمور نظارت بر آن کرد. در سال ۱۹۸۳، این کمیته به هیئت نظارت بر فعالیتهای اینترنتی (Internet Activity Board - IAB) تغییر نام داد، و وظیفه همسو نگه داشتن آرپانت و اینترنت بر عهده آن گذاشته شد (کاری که می توان آنرا به چوپانی یک گله گربه تشبیه کرد!). بعدها نام این کمیته به هیئت مدیره معماری اینترنت (Internet Architecture Board) تغییر کرد، بگونه ای که حروف اختصاری آن همچنان IAB باقی ماند.

به هر یک از نزدیک به ده عضو IAB موضوع مهمی برای دنبال کردن محول شد. این اعضا سالی چند بار با هم ملاقات و تبادل نظر می کردند، و گزارش کار خود را به NFS و وزارت دفاع (که پشتیبانی مالی آنها بر عهده داشتند) ارائه می کردند. وقتی نیاز به تدوین استاندارد جدیدی حس می شد، اعضای IAB آنرا با جار و جنجال زیاد اعلام

می کردند، تا دانشجویانی که موتور نرم افزاری اینترنت محسوب می شدند، بتوانند آنرا پیاده سازی کنند. تبادل اطلاعات فنی توسط مقالات و گزارشهایی بنام نظرخواهی (Request For Comment - RFC) انجام می شد. این RFC ها (که به ترتیب انتشار شماره گذاری می شوند) در سایت www.ietf.org/rfc نگهداری می شوند، تا همه کسانی که مایلند بتوانند به آنها دسترسی داشته باشند. هم اکنون متجاوز از ۳۰۰۰ RFC در این سایت موجود است، که در این کتاب به بسیاری از آنها اشاره خواهیم کرد.

تا سال ۱۹۸۹ اینترنت چنان گسترش یافته بود، که این روش نسبتاً رسمی دیگر نمی توانست بکار آید. در آن زمان شرکتهای بسیاری محصولات TCP/IP خود را وارد بازار کرده بودند، و نمی خواستند آنها را بصرف اینکه چند پژوهشگر به خیال خود ایده های بهتری دارند، عوض کنند. در تابستان ۱۹۸۹، ساختار سازمانی IAB تغییر کرد، و به دو بخش نیروی پژوهشی اینترنت (Internet Research Task Force - IRTF) و نیروی مهندسی اینترنت (Internet Engineering Task Force - IETF) تقسیم شد، و نیروهای جدیدی (غیر از مجامع تحقیقاتی) وارد آن شدند. در سالهای اول این سازمان ها دارای گردش بسته بودند: هر عضو برای دو سال انتخاب می شد، و اعضای جدید فقط توسط اعضای قدیمی منسوب می شدند. بعدها، انجمن اینترنت (Internet Society) بوجود آمد، که تمامی علاقمندان اینترنت می توانستند در آن عضویت یابند. انجمن اینترنت از جهاتی شبیه IEEE یا ACM است، و توسط یک هیئت امنای انتخابی، که اعضای IAB را منسوب می کند، اداره می شود. هدف از تقسیم IAB آن بود که IRTF بتواند به تحقیقات بلند مدت مشغول شود، در حالیکه IETF به کارهای مهندسی کوتاه مدت اشتغال دارد. IETF خود به چند گروه کاری تقسیم می شد، که هر یک روی موضوعی خاص کار می کردند. به منظور هماهنگ کردن فعالیتهای این گروه ها، مدیران گروه های کاری نشستهای متعددی برگزار می کردند. سرفصلهایی که گروه های کاری IETF روی آنها کار می کردند، عبارت بودند از: برنامه های کاربردی جدید، گردآوری اطلاعات کاربران، یکپارچه سازی OSI، هدایت و آدرس دهی، امنیت، مدیریت شبکه، و استاندارد سازی. تعداد این گروه های کاری بتدریج افزایش یافت، و اکنون متجاوز از ۷۰ گروه کاری ذیل IETF مشغول به کار هستند.

علاوه بر آن، فرآیند استاندارد سازی رسمی تری (که شبیه ISO بود) پذیرفته شد. برای آن که یک ایده جدید به صورت استاندارد پیشنهادی (Proposed Standard) در آید، بایستی بطور کامل در یک RFC تشریح شود، و در جامعه اینترنت نیز علاقه کافی نسبت به آن وجود داشته باشد. سپس، برای آنکه این استاندارد به مرحله پیش نویس استاندارد (Draft Standard) ارتقاء یابد، بایستی بصورت واقعی پیاده سازی شده و به مدت حداقل ۴ ماه در ۲ سایت اینترنتی بطور همه جانبه تست شود. اگر IAB متقاعد شود که ایده اصلی خوب است و نرم افزار نیز بخوبی کار می کند، می تواند این RFC را رسماً بعنوان یک استاندارد اینترنتی اعلام کند. برخی از این استانداردها حتی بصورت استانداردهای نظامی (MIL-STD) در آمده اند، و رعایت آنها در محصولاتی که به وزارت دفاع فروخته می شوند، اجباریست. دیوید کلارک نقل قول جالبی درباره استانداردهای اینترنت دارد که اکنون بسیار معروف است: «اجماع نظری ناقص، به همراه کدی که کار می کند».

۷. واحدهای اندازه گیری

برای اجتناب از هر گونه سوء تفاهم، لازم است تأکید کنیم که در این کتاب بجای واحدهای اندازه گیری انگلیسی (با آن اندازه ها و ضرایب عجیب و غریب) از سیستم متریک استفاده شده است. در شکل ۱-۳۹ پیشنوذهای اصلی سیستم متریک را مشاهده می کنید. اغلب از حروف اختصاری این پیشنوذهای استفاده خواهیم کرد، و واحد اعداد بزرگتر از ۱ با حروف بزرگ می نویسیم، مانند KB و MB (البته با یک استثناء تاریخی: حروف اختصاری

پیشوند	عدد اعشاری	توان	پیشوند	عدد اعشاری	توان
Kilo	1,000	10^3	milli	0.001	10^{-3}
Mega	1,000,000	10^6	micro	0.000001	10^{-6}
Giga	1,000,000,000	10^9	nano	0.000000001	10^{-9}
Tera	1,000,000,000,000	10^{12}	pico	0.000000000001	10^{-12}
Peta	1,000,000,000,000,000	10^{15}	fermi	0.000000000000001	10^{-15}
Exa	1,000,000,000,000,000,000	10^{18}	atto	0.000000000000000001	10^{-18}
Zetta	1,000,000,000,000,000,000,000	10^{21}	zepto	0.000000000000000000001	10^{-21}
Yotta	1,000,000,000,000,000,000,000,000	10^{24}	yocto	0.000000000000000000000001	10^{-24}

شکل ۱-۳۹. پیشوندهای اصلی سیستم متریک.

کیلو بیت/ثانیه بصورت kbps نوشته می‌شود). برای مثال، یک خط انتقال 1-Mbps در هر ثانیه 10^6 بیت اطلاعات را منتقل می‌کند، و 100 psec معادل 10^{-10} ثانیه است. برای تمایز بین پیشوندهای «میلی» و «میکرو» - که هر دو با m شروع می‌شوند -، اولی را با "m" و دومی را با "μ" نمایش می‌دهیم.

همچنین اشاره به این نکته خالی از فایده نیست که، واحدهایی که در صنعت کامپیوتر برای اندازه‌گیری ظرفیت حافظه، دیسک، و فایل بکار برده می‌شوند، کمی با سایر واحدها تفاوت دارند. در اینجا، «کیلو» بجای 10^3 (معادل 1000) معنای 2^{10} (یا 1024) می‌دهد، چون ظرفیت حافظه همیشه توانی از ۲ است - بنابراین، وقتی گفته می‌شود 1-KB، بمعنای ۱۰۲۴ بایت است، نه ۱۰۰۰ بایت. به همین ترتیب، 1-MB معادل 2^{20} (1,048,576 بایت) است، 1-GB معادل 2^{30} (1,073,741,824 بایت)، و 1-TB معادل 2^{40} (1,099,511,627,776 بایت). از طرف دیگر، در صحبت از سرعت انتقال داده‌ها، دیگر اعداد توانی از ۲ نیستند؛ برای مثال، خط 1-kbps دقیقاً 1000 بیت/ثانیه داده منتقل می‌کند، و یک شبکه 1-Mbps دقیقاً با سرعت 1,000,000 بیت/ثانیه کار می‌کند. متأسفانه، بسیاری از افراد قادر به تفکیک این دو نیستند (بویژه در مورد ظرفیت دیسکها). برای اجتناب از هر گونه ابهام، در این کتاب واحدهای KB، MB، و GB بترتیب معادل 2^{10} ، 2^{20} ، و 2^{30} بایت، و واحدهای kbps، Mbps، و Gbps بترتیب معادل 10^3 ، 10^6 ، و 10^9 بیت/ثانیه در نظر گرفته خواهند شد.

۸-۱ طرح کلی مباحث کتاب

در این کتاب شبکه‌های کامپیوتری از جنبه نظری و عملی مورد بررسی قرار گرفته‌اند. اکثر فصول کتاب با بحث در کلیات موضوع مورد نظر شروع شده، و با بررسی مثالها و نمونه‌های عملی ادامه می‌یابد. این نمونه‌ها از اینترنت و شبکه‌های بیسیم انتخاب شده‌اند، چون در عین اهمیت بسیار با یکدیگر تفاوت‌های اساسی نیز دارند. هر گاه لازم بوده به مثالهای دیگر نیز استناد کرده‌ایم.

در این کتاب کار خود را بر اساس مدل ترکیبی شکل ۱-۲۴ بنا نهاده‌ایم، و از فصل آینده بحث درباره سلسله مراتب پروتکل‌های این مدل را (از پائین به بالا) شروع خواهیم کرد. در فصل ۲ ابتدا پیش‌زمینه‌ای درباره سیستمهای مخابرات داده (که شامل سیستمهای کابلی، بیسیم و ماهواره‌ای می‌شود) بدست خواهیم داد. این سیستمها که به لایه فیزیکی مربوط می‌شوند، بیشتر از جنبه معماری مورد بحث قرار گرفته‌اند تا جنبه سخت‌افزاری. در این فصل نمونه‌های متعددی را مورد بررسی قرار داده‌ایم، از شبکه‌های سونیچینگ تلفن معمولی و تلفن همراه گرفته، تا

شبکه های تلویزیون کابلی.

در فصل ۳ لایه پیوند داده و پروتکل های آن (بهمراه تحلیل این پروتکلها) به کمک مثالهای متعدد مورد بحث قرار گرفته است. پس از آن، تعدادی از پروتکل های مهم این لایه، از جمله HDLC (که در شبکه های با سرعت کم تا متوسط مورد استفاده قرار می گیرد) و PPP (که در اینترنت کاربرد دارد) را مورد بررسی قرار خواهیم داد.

فصل ۴ با زیرلایه دسترسی داده (که بخشی از لایه پیوند داده است) سروکار دارد. بحث اصلی در اینجا نحوه دسترسی به شبکه در جاهانیست که کانالهای فیزیکی شبکه به صورت اشتراکی مورد استفاده قرار می گیرند (مانند شبکه های LAN و برخی از شبکه های ماهواره ای). در این فصل نیز نمونه های متعددی (از جمله اترنت، LAN بیسیم، MAN بیسیم، بلوتوث، و شبکه های ماهواره ای) مورد بررسی قرار گرفته است. درباره پل (bridge) و سوئیچ (switch) نیز در همین فصل صحبت کرده ایم.

فصل ۵ به لایه شبکه، بویژه مسیریابی (routing) و الگوریتم های آن (استاتیک و دینامیک)، اختصاص دارد. حتی با بهترین الگوریتم های مسیریابی، اگر بار بیش از حد به یک شبکه تحمیل شود، امکان بروز ازدحام (congestion) در آن وجود دارد، بنابراین یکی از مباحث مهم این فصل ازدحام و راههای جلوگیری از آن (و از آن هم فراتر، تضمین نوعی کیفیت سرویس) است. نحوه اتصال شبکه های غیرمتجانس و رفع مشکلات آن از دیگر مباحث این فصل است. لایه شبکه از اهمیت زیادی در اینترنت برخوردار است.

در فصل ۶ درباره لایه انتقال (و بویژه پروتکل های اتصال-گرا) مفصلاً صحبت خواهیم کرد. حتی یکی از سرویسهای ساده این لایه بصورت عملی (بهمراه کد آن) ارائه شده است، تا با نحوه پیاده سازی آن بیشتر آشنا شوید. پروتکل های اینترنتی TCP و UDP (و کارایی آنها) نیز مفصلاً در این فصل مورد بررسی قرار گرفته اند. از دیگر مباحثی که در فصل ۶ مطرح خواهد شد، شبکه های بیسیم است.

فصل ۷ با لایه کاربرد (و پروتکلها و برنامه های آن) سروکار دارد. اولین مبحث این فصل، دفترچه تلفن اینترنت یعنی DNS است. ایمیل و پروتکل های آن مبحث بعدی فصل ۷ است. پس از آن سراغ وب می رویم، و مباحثی مانند محتویات استاتیک و دینامیک، فعل و انفعالات سمت مشتری و سرور دهنده، پروتکل های آن، کارایی وب، و وب بیسیم را مورد بررسی قرار خواهیم داد. در پایان نیز، درباره شبکه های چندرسانه ای (صدا و تصویر جویباری - streaming -، و رادیوی اینترنتی) صحبت خواهیم کرد.

فصل ۸ درباره امنیت شبکه است. از آنجائیکه مباحث این فصل با تمام لایه ها سروکار دارد، آنرا در آخر (بعد از پایان بحث لایه های شبکه) آورده ایم. این فصل را با معرفی تکنولوژیهای رمزنگاری (cryptography) آغاز کرده ایم، و نشان داده ایم که چگونه می توان به کمک این تکنولوژیها امنیت مخابرات داده، ایمیل و وب را تأمین کرد. این فصل را با بحثی درباره تقابل بین امنیت و حریم خصوصی افراد، آزادی بیان، سانسور (و سایر موضوعاتی که با امنیت شاخ به شاخ می شوند) به پایان رسانده ایم.

در فصل ۹ مقالات و کتابهایی را که می توانید برای مطالعه بیشتر درباره موضوعات هر فصل به آنها مراجعه کنید، به همان ترتیب فصول کتاب آورده ایم. در این فصل یک کتابنامه مفصل در زمینه موضوعات مطرح شده در کتاب نیز آورده شده است.

در سایت وب مؤلف کتاب

<http://www.prenhall.com/tanenbaum>

می توانید لینکهای متعددی به درسنامه ها، صفحات پرسش و پاسخ، شرکتها، کنسرسیومهای صنعتی، سازمانهای حرفه ای، سازمانهای استاندارد، تکنولوژیها، و مقالات تخصصی پیدا کنید.

۹-۱ خلاصه

از شبکه های کامپیوتری می توان برای مقاصد مختلفی (در شرکتها، یا برای افراد عادی) استفاده کرد. در شرکتها، شبکه می تواند دسترسی به منابع اطلاعاتی را برای تمام کارکنان فراهم آورد. در این شبکه ها معمولاً از مدل مشتری-سرویس دهنده (که در آن منابع مشترک روی کامپیوترهای قدرتمندی موسوم به سرویس دهنده - server - قرار می گیرند) استفاده می شود. شبکه برای افراد عادی امکان دسترسی به منابع اطلاعاتی یا تفریحی را فراهم می آورد. امروزه افراد بسیاری با یک مودم از منزل خود به شرکتهای خدمات اینترنتی (ISP) متصل شده، و از امکانات آن استفاده می کنند. یکی از زمینه هایی که امروزه سرعت رو به گسترش است، شبکه های بیسیم است، که امکان دسترسی به اینترنت را حتی از تلفنهای همراه به افراد می دهد.

در یک تقسیم بندی بسیار کلی، شبکه ها را می توان به شبکه های محلی (LAN)، شبکه های شهری (MAN)، شبکه های گسترده (WAN) و شبکه های اینترنت (internetwork) تقسیم کرد، که هر کدام برای خود دارای ویژگیها، سرعت، تکنولوژی و جایگاه خاصی می باشد. شبکه محلی (LAN) سرعت بالایی دارد، ولی معمولاً به یک ساختمان مفرد محدود می شود. محدوده جغرافیایی شبکه شهری (MAN) - همانطور که از نام آن برمی آید - یک شهر است؛ تلویزیون کابلی نمونه ای از شبکه های شهری است. شبکه های گسترده (WAN) یک کشور یا قاره را در بر می گیرند. شبکه های LAN و MAN سوئیچ نشده هستند (یعنی، مسیریاب - router - ندارند)، در حالیکه شبکه های WAN سوئیچ شده اند. امروزه شبکه های بیسیم (به ویژه LAN بیسیم) از محبوبیت روزافزونی برخوردارند. از اتصال چند شبکه به یکدیگر نیز یک شبکه شبکه ها شکل می گیرد.

نرم افزار شبکه از پروتکلها (که قواعد حاکم بر ارتباط پروسس ها را تعیین می کنند) تشکیل می شود. پروتکلها یا اتصال-گرا (connection-oriented) هستند یا غیرمتصل (connectionless). اغلب شبکه ها از مدل سلسله مراتبی پروتکلها استفاده می کنند، که در آن هر لایه سرویسهایی را در اختیار لایه های بالاتر قرار می دهد، و آنرا از جزئیات کار در لایه های پائینتر ایزوله می کند. مجموعه پروتکل (protocol stack) های مهم امروزی عمدتاً بر مبنای دو مدل OSI و TCP/IP بنا شده اند. هر دوی این مدلها دارای لایه های شبکه، انتقال و کاربرد هستند، ولی در لایه های دیگر با هم فرق دارند. هنگام طراحی پروتکلها باید به مسائلی از قبیل مالتی پلکس کردن (multiplexing)، کنترل جریان (flow control)، کنترل خطا (error control) و مانند آنها توجه ویژه مبذول داشت. (بخش عمده ای از این کتاب به پروتکلها و طراحی آنها اختصاص دارد).

وظیفه شبکه ارائه سرویس به کاربران است، که این سرویسها نیز می توانند اتصال-گرا یا غیرمتصل باشند. در برخی شبکه ها، یک لایه سرویس غیرمتصل در اختیار می گذارد، در حالیکه لایه بالاتر سرویس اتصال-گرا ارائه می کند.

معروفترین شبکه های موجود عبارتند از: اینترنت، ATM، اترنت و LAN بیسیم (IEEE 802.11). اینترنت محصول تکامل شبکه آرپانت (ARPANET - شبکه سوئیچ بسته وزارت دفاع ایالات متحده آمریکا) بود. اینترنت در واقع امروزه شبکه ایست از هزاران شبکه دیگر، نه یک شبکه واحد. مشخصه اصلی اینترنت استفاده از مجموعه پروتکل TCP/IP است. شبکه های ATM بیشتر در سیستمهای تلفن، و برای مخابرات راه دور، بکار می روند. اترنت (Ethernet) نیز محبوبترین تکنولوژی LAN است، که در اغلب شرکتها و دانشگاهها از آن استفاده می شود. و بالاخره، شبکه های LAN بیسیم که (با سرعت خیره کننده 54 Mbps) سرعت در حال گسترش هستند.

برای آنکه چند کامپیوتر بتوانند با هم ارتباط برقرار کنند، مهمترین موضوع وجود استانداردهای متعدد (سخت افزاری و نرم افزاری) است. وظیفه تدوین استاندارد برای شبکه های کامپیوتری بر عهده سازمانهایی از قبیل ITU-T، ISO، IEEE و IAB گذشته شده است.

مسائل

۱. فرض کنید سگی از نژاد سنت برنارد (نوعی سگ قوی هیکل که در کشور کوهستانی سوئیس برای نجات کوهنوردان ساخته دیده تربیت می شود) بنام پرنی دارید، و او را برای حمل جعبه ای حاوی سه نوار 8mm آموزش داده اید (به هر حل پُر شدن دیسک هم نوعی موقعیت اورژانس است). هر نوار 7 GB ظرفیت دارد، و پرنی می تواند تحت هر شرایطی با سرعت 18 km/h حرکت کند. تا چه فاصله ای نرخ انتقال اطلاعات پرنی همچنان از خط انتقالی با سرعت 150 Mbps (بدون در نظر گرفتن سرآیند) بیشتر است؟
۲. یکی از گزینه های رقیب شبکه های LAN سیستم های تسهیم زمانی (time sharing) بزرگ (بهمراه ترمینالهایی که کاربران از آنها استفاده می کنند) است. دو مزیت شبکه های LAN مبتنی بر مدل مشتری-سرویس دهنده را نسبت به سیستم های فوق بیان کنید.
۳. کارایی سیستم های مشتری-سرویس دهنده به دو ویژگی مهم شبکه وابسته است: پهنای باند (bandwidth - حداکثر تعداد بیت هایی که شبکه می تواند در هر ثانیه منتقل کند)، و زمان تأخیر (latency time - مدت زمانی که طول می کشد تا اولین بیت از مشتری به سرویس دهنده - یا بالعکس - برسد). دو مثال از شبکه ای با پهنای باند زیاد و زمان تأخیر زیاد، و پهنای باند کم و زمان تأخیر کم بزنید.
۴. غیر از پهنای باند زیاد و زمان تأخیر کم، چه شرایط دیگری باید فراهم باشد تا یک شبکه کیفیت مناسبی برای سرویس صدای دیجیتال داشته باشد؟
۵. یک از عوامل مهم در میزان تأخیر سیستم های سوئیچینگ بسته مبتنی بر مدل ذخیره-هدایت، مدت زمانیست که صرف ذخیره و هدایت بسته در یک سوئیچ می شود. آیا زمان سوئیچینگ 10 μ sec برای یک سیستم مشتری-سرویس دهنده که بین کالیفرنیا و نیویورک کار می کند، عامل مهمی محسوب می شود. سرعت انتشار امواج الکترومغناطیس در سیم مسی را 0.667 سرعت نور در نظر بگیرید.
۶. یک سیستم مشتری-سرویس دهنده از شبکه ماهواره ای، که ماهواره آن در ارتفاع 40,000 km سطح زمین قرار گرفته، استفاده می کند. زمان تأخیر پاسخ در بهترین حالت چقدر است؟
۷. در آینده ای نه چندان دور مردم می توانند مستقیماً از طریق ترمینالهای متصل به شبکه های کامپیوتری به لوايح و طرح های قانونی رأی بدهند، و دیگر نیازی به مجالس قانونگذاری وجود نخواهد داشت. جنبه های مثبت چنین دموکراسی مستقیمی نسبتاً روشن است؛ کمی درباره نقاط منفی آن بحث کنید.
۸. می خواهیم پنج مسیر یاب را در یک زیر شبکه نقطه-به-نقطه به هم وصل کنیم. هر زوج از این مسیر یاب ها را می توان با یک خط پُر سرعت، خطی با سرعت متوسط، و یا یک خط کم سرعت به هم متصل کرد، و یا اینکه اصلاً آنها را به هم وصل نکرد. اگر یک کامپیوتر بتواند هر توپولوژی را در 100 ms طراحی و بررسی کند، چه مدت طول می کشد تا تمام توپولوژی های ممکن را بررسی کند؟
۹. تعداد 1 - 2ⁿ مسیر یاب در یک درخت باینری متقارن (یک مسیر یاب در هر گره) به هم متصل شده اند. مسیر یاب i برای ارتباط با مسیر یاب j، باید ابتدا پیام خود را به ریشه این درخت بفرستد، تا از آنجا بدست مسیر یاب j برسد. (با فرض اینکه تمام مسیر یاب یکسان هستند) عبارتی بنویسید که تعداد متوسط پرش های لازم برای رسیدن پیام یک مسیر یاب به مسیر یاب دیگر را (برای n های بزرگ) بدست دهد.
۱۰. یکی از نقاط منفی زیر شبکه های پخش ظرفیتی است که در اثر اقدام به پخش همزمان توسط چند کامپیوتر از دست می رود (تصادم - collision). اجازه دهید مسئله را ساده کرده، و فرض کنیم زمان به برش های مساوی تقسیم شده، و در هر برش زمانی n کامپیوتر با احتمال p اقدام به پخش روی شبکه می کنند. چه

- کسری از برشهای زمانی در اثر بروز حالت تصادم تلف خواهند شد؟
۱۱. دو دلیل برای استفاده از پروتکل های لایه ای ارائه کنید.
۱۲. رئیس یک شرکت رنگسازی ایده جدیدی برای تولید یک محصول بدیع و نو ظهور دارد. موضوع را با اداره حقوقی شرکت در میان می گذارد، و آنها هم از اداره مهندسی درخواست کمک می کنند. سرپرست اداره مهندسی درباره زوایای مختلف این طرح با یکی از هم تایان خود در شرکتی دیگر مشورت می کند، که وی نیز موضوع را به اداره حقوقی شرکت خود منتقل می کند. در پایان نیز، رؤسای دو شرکت بر سر جزئیات مالی معامله با یکدیگر به مذاکره می پردازند. آیا می توان این سناریو را مطابق با پروتکل های چند لایه مدل OSI دانست؟
۱۳. دو تفاوت عمده بین ارتباطات اتصال-گرا و غیر متصل چیست؟
۱۴. دو شبکه سرویس های اتصال-گرای قابل اعتماد ارائه می کنند: یکی از آنها بصورت استریم بایت، و دیگری بصورت استریم پیام. آیا این دو یکسان هستند؟ اگر پاسخ مثبت است، چه دلیلی برای نامگذاری جداگانه آنها وجود دارد؟ اگر خیر، تفاوت آنها در چیست؟
۱۵. در بحث پروتکل های شبکه، «مذاکره» (negotiation) چه معنایی دارد؟ یک مثال بزنید.
۱۶. در شکل ۱-۱۹ یک سرویس نشان داده شده است. آیا سرویس دیگری در این شکل وجود دارد؟ اگر آری، کجا؟ اگر خیر، چرا؟
۱۷. در برخی از شبکه ها، لایه پیوند داده با درخواست ارسال مجدد فریم هایی که بدرستی دریافت نشده اند، نوعی کنترل خطا انجام می دهد. اگر احتمال خراب شدن یک فریم p باشد، متوسط تعداد دفعاتی که یک فریم باید فرستاده شود، چقدر است؟ فرض کنید فریم های تصدیق دریافت (acknowledgement) هرگز خراب نمی شوند.
۱۸. کدامیک از لایه های OSI وظایف ذیل را بر عهده دارند:
(الف) تقسیم استریم بیتها به فریم
(ب) تعیین مسیری در زیر شبکه، که باید از آن استفاده شود
۱۹. اگر واحد تبادل داده در لایه پیوند داده فریم، و در لایه انتقال بسته نام داشته باشد، فریمها در بسته پیچیده می شوند، یا بسته ها در فریم؟ توضیح دهید.
۲۰. ساختار سلسله مراتبی پروتکل های یک سیستم n لایه دارد. برنامه های کاربردی در این سیستم پیام هایی بطول M بایت تولید می کنند، و در هر لایه یک سرآیند h بایتی به پیام لایه بالاتر اضافه می شود. چه کسری از پهنای باند شبکه را این سرآیندها اشغال می کنند؟
۲۱. دو شباهت و دو تفاوت مدل های مرجع را OSI و TCP/IP را بیان کنید.
۲۲. تفاوت اصلی TCP و UDP چیست؟
۲۳. زیر شبکه شکل ۱-۲۵ (ب) طوری طراحی شده که در مقابل حملات هسته ای دوام بیاورد. چند بمب اتمی لازم است تا این زیر شبکه به دو قسمت کاملاً مجزا تقسیم شود؟ فرض کنید برای نابود کردن هر گره و تمام لینک های متصل به آن یک بمب اتمی کافیت.
۲۴. تخمین زده شده است که اینترنت هر ۱۸ ماه دو برابر می شود. (و با اینکه کسی واقعاً تعداد آنها را نمی داند طبق برآوردهای تخمینی در سال ۲۰۰۱ تعداد کامپیوترهای اینترنت ۱۰۰ میلیون بوده است. با استفاده از این

مفروضات، تعداد کامپیوترهای اینترنت در سال ۲۰۱۰ را محاسبه کنید. آیا این عدد را باور می‌کنید؟ توضیح دهید.

۲۵. هنگام تبادل فایل بین دو کامپیوتر، دو استراتژی تصدیق دریافت ممکن است. در روش اول، فایل به قطعاتی تقسیم شده، و برای هر قطعه تصدیق دریافت جداگانه مطالبه می‌شود، اما برای کل فایل، خیر. در روش دوم، برای تک تک قطعات تصدیق دریافت مطالبه نمی‌شود، ولی در پایان دریافت کل فایل باید به تأیید طرف مقابل برسد. درباره این دو روش بحث کنید.

۲۶. چرا ATM از سلولهای کوچک و با اندازه ثابت استفاده می‌کند؟

۲۷. هر بیت در استاندارد اولیه 802.3 چند متر طول داشت؟ سرعت کار این شبکه را 10 Mbps، و سرعت سیگنالهای الکتریکی در کابل کواکسیال را 2/3 سرعت نور در خلاء فرض کنید.

۲۸. ابعاد یک تصویر 1024×768 پیکسل است، و هر پیکسل آن 3 بایت جا می‌گیرد (و فرض کنید این تصویر فشرده نشده است). انتقال این تصویر روی یک خط 56-kbps چقدر طول می‌کشد؟ روی یک خط 1-Mbps چقدر؟ روی شبکه اینترنت 10-Mbps چقدر؟ و روی شبکه اینترنت 100-Mbps چقدر؟

۲۹. اینترنت و شبکه‌های بیسیم شباهتها و تفاوتهایی دارند. یکی از ویژگیهای اینترنت اینست که در هر لحظه فقط یک فریم می‌تواند روی شبکه منتقل شود. آیا در شبکه‌های 802.11 نیز چنین است؟ توضیح دهید.

۳۰. نصب شبکه‌های بیسیم بسیار ساده است، که همین باعث ارزانی آنها شده است، چون معمولاً هزینه‌های نصب در مقابل هزینه تجهیزات رقم قابل توجهی را تشکیل می‌دهد. با این حال، این نوع شبکه معایبی نیز دارد. دو تا از این معایب را نام ببرید.

۳۱. دو مزیت و دو عیب برای استاندارد کردن پروتکل‌های شبکه بشمارید.

۳۲. وقتی یک سیستم از دو قسمت ثابت و متحرک تشکیل می‌شود (مانند درایو CD، و دیسک CD-ROM)، استاندارد کردن آن از اهمیت زیادی برخوردار است، تا محصولات شرکتهای مختلفی که این قطعات را تولید می‌کنند، با هم سازگار باشد. سه مثال از خارج صنعت کامپیوتر بزنید، که چنین استانداردی در آنها وجود دارد. حال سه مثال از خارج صنعت کامپیوتر بزنید، که چنین استانداردی در آنها وجود نداشته باشد.

۳۳. تمام فعالیتهای خود را که در طول شبانه روز بنحوی با شبکه سروکار دارد، فهرست وار بنویسید. اگر این شبکه‌ها به یکباره از کار بیفتند، چه تأثیری روی زندگی شما خواهد گذاشت؟

۳۴. شبکه‌هایی را که در محل کار یا تحصیل خود می‌شناسید (به همراه نوع، توپولوژی، و روش سونیچینگ آنها)، مشخص کنید.

۳۵. برنامه ping به شما اجازه می‌دهد تا بسته‌ای را به یک مقصد مشخص فرستاده، و زمان رفت و برگشت آنرا محاسبه کنید. از این برنامه برای تعیین زمان رفت و برگشت بسته‌ها به چند نقطه مختلف استفاده کنید. با استفاده از این اطلاعات، نمودار زمان انتقال بسته‌ها روی اینترنت (در جهت رفت) را بر حسب فاصله رسم کنید. بهتر است برای این منظور از دانشگاههای مهم و سرشناس (که محل آنها دقیقاً مشخص است) استفاده کنید. برای مثال، سایت berkeley.edu در برکلی - کالیفرنیا است، سایت mit.edu در کمبریج - ماساچوست،

سایت vu.nl در آمستردام-هلند، سایت www.usyd.edu.au در سیدنی-استرالیا، و سایت www.uct.ac.za در کیپ تاون-آفریقای جنوبی.

۳۶. سری به سایت IETF (به آدرس www.ietf.org) بزنید، و ببینید چکار می کنند. یکی از پروژه هایی را که به آن علاقه دارید، انتخاب کرده، و درباره آن مقاله ای نیم صفحه ای بنویسید.

۳۷. استانداردسازی در دنیای شبکه های کامپیوتری بسیار مهم است، و همانطور که دیدید سازمانهای ITU و ISO این وظیفه را بر عهده دارند. به سایت وب این سازمانها (www.iso.org و www.itu.org) رفته، و با کارهای استانداردسازی آنها آشنا شوید. گزارش کوتاهی درباره انواع چیزیهایی که این دو سازمان استاندارد کرده اند، بنویسید.

۳۸. اینترنت از تعداد زیادی شبکه های مختلف تشکیل شده است، که آرایش آنها توپولوژی اینترنت را مشخص می کند. در خود اینترنت اطلاعات زیادی در زمینه توپولوژی آن وجود دارد. با استفاده از یک موتور جستجو (search engine) در باره این موضوع تحقیق کرده، و خلاصه ای از یافته های خود را در یک گزارش بنویسید.

لایه فیزیکی

در این فصل پانین ترین لایه سلسله مراتب شکل ۱-۲۴ را مورد بررسی قرار خواهیم داد. مشخصات مکانیکی، الکتریکی و تایمینگ (همزمانی) شبکه در این لایه تعریف می شود. برای شروع کمی درباره تئوری مخابرات صحبت می کنیم، فقط برای اینکه نشان دهیم دست طبیعت چه محدودیتهایی را در زمینه انتقال داده ها به ما تحمیل کرده است.

سپس با رسانه های فیزیکی که برای انتقال داده ها از آنها استفاده می شود، آشنا می شوید: رسانه های هدایت پذیر (مانند سیم مسی و فیبر نوری)، بیسیم (امواج رادیویی زمینی)، و ماهواره. آشنایی با این رسانه ها برای شناخت تکنولوژیهای مدرن مخابراتی اهمیت اساسی دارد.

در ادامه سه نمونه از سیستمهای مخابراتی که در شبکه های کامپیوتری کاربرد گسترده ای دارند، را مفصلاً مورد بررسی قرار خواهیم داد: سیستم تلفن ثابت، شبکه تلفن همراه، و تلویزیون کابلی. در تمام این سیستمها از فیبر نوری بعنوان ستون فقرات (backbone) استفاده می شود، ولی تکنولوژی پیکار رفته در آخرین قطعه اتصال (از مرکز تلفن یا ایستگاه توزیع تا مصرف کننده) در آنها متفاوت است.

۱.۲ مبانی نظری مخابرات داده

برای انتقال اطلاعات روی سیم می توان از متغیرهای الکتریکی مانند ولتاژ یا جریان استفاده کرد. با نمایش تغییرات این ولتاژ یا جریان بصورت تابعی از زمان، می توان رفتار سیگنال را مدل سازی و تحلیل ریاضی کرد. موضوع این بخش تحلیل ریاضی سیگنالهای الکتریکی است.

۱.۱.۲ آنالیز فوریه

در اوایل قرن نوزدهم میلادی، ریاضیدان فرانسوی ژان بپتیست فوریه ثابت کرد که هر تابع متناوب، $g(t)$ ، با دوره تناوب T را می توان به صورت مجموع بینهایت جمله سینوسی و کسینوسی نوشت:

$$g(t) = \frac{1}{T} c + \sum_{n=1}^{\infty} a_n \sin(\gamma n \pi f t) + \sum_{n=1}^{\infty} b_n \cos(\gamma n \pi f t) \quad (1-2)$$

که در آن $f = 1/T$ فرکانس اصلی، a_n و b_n ضرایب هارمونی (جمله n ام تابع سینوس و کسینوس، و c یک عدد ثابت است. به تجزیه یک تابع به مجموع بینهایت جمله سینوسی و کسینوسی سری فوریه (Fourier series) گفته می شود. اگر ضرایب جملات سری فوریه یک تابع و دوره تناوب آن (T) معلوم باشد، می توان این تابع را با

استفاده از معادله (۱-۲) ایجاد کرد.

اگر یک سیگنال داده دارای دوره محدودی باشد (که همیشه هم چنین است)، می توان آنرا تکرار بینهایت یک الگوی مشخص فرض کرد (بعبارت دیگر، این موج در فاصله زمانی T تا $2T$ با فاصله 0 تا T کاملاً یکسان است). برای محاسبه ضریب a_n کافیهست دو طرف معادله (2-1) را در $\sin(2\pi kft)$ ضرب کرده، و سپس در فاصله 0 تا T از آن انتگرال بگیریم. از آنجائیکه

$$\int_0^T \sin(2\pi kft) \sin(2\pi nft) dt = \begin{cases} 0 & \text{for } k \neq n \\ T/2 & \text{for } k = n \end{cases}$$

فقط جمله ای که ضریب a_n دارد، باقی می ماند، و جملات b_n بکلی از بین خواهند رفت. به همین ترتیب، اگر دو طرف معادله (2-1) را در $\cos(2\pi kft)$ ضرب کرده، و سپس در فاصله 0 تا T از آن انتگرال بگیریم، ضریب b_n بدست خواهد آمد. برای بدست آوردن ثابت c هم کافیهست از تابع $g(t)$ در همین فاصله زمانی انتگرال بگیریم. خلاصه عملیات فوق را در فرمولهای ذیل مشاهده می کنید:

$$a_n = \frac{2}{T} \int_0^T g(t) \sin(2\pi nft) dt \quad b_n = \frac{2}{T} \int_0^T g(t) \cos(2\pi nft) dt \quad c = \frac{2}{T} \int_0^T g(t) dt$$

۲.۱.۲ محدودیت پهنای باند

اما برای اینکه ببینید اینها چه ربطی به مخابرات داده دارد، اجازه دهید یک مثال بزنیم: مخابره کاراکتر آسکی حرف "b" که بصورت ۸ بیتی کُد شده است. حرف "b" در استاندارد آسکی ۸ بیتی بصورت 01100010 کُد می شود، و اینها بیتهایی هستند که باید مخابره شوند. در نمودار سمت چپ شکل ۱-۲ (الف) ولتاژهای خروجی کامپیوتر برای حرف "b" را می بینید. با استفاده از آنالیز فوریه، ضرایب ثابت این سیگنال بصورت زیر بدست می آیند:

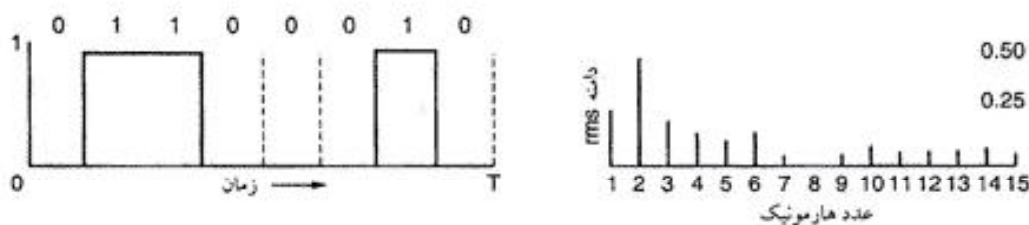
$$a_n = \frac{1}{\pi n} [\cos(\pi n/4) - \cos(3\pi n/4) + \cos(6\pi n/4) - \cos(7\pi n/4)]$$

$$b_n = \frac{1}{\pi n} [\sin(3\pi n/4) - \sin(\pi n/4) + \sin(7\pi n/4) - \sin(6\pi n/4)]$$

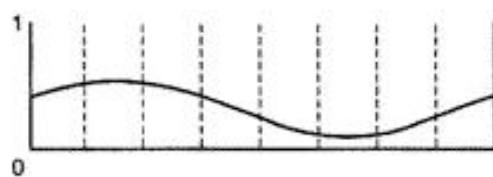
$$c = 3/4$$

در نمودار سمت راست شکل ۱-۲ (الف) مقدار rms ضرایب فوریه، یعنی $\sqrt{a_n^2 + b_n^2}$ ، چند هارمونی اول این سیگنال را مشاهده می کنید. علت اهمیت این ضرایب آنست که انرژی انتشار سیگنال با مجذور آنها (در یک فرکانس خاص) متناسب است.

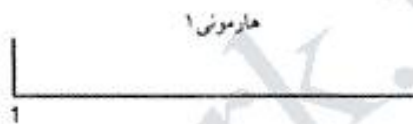
از طرف دیگر می دانیم که هیچ سیستم پخش بدون اتلاف انرژی نیست، ولی اگر تمام ضرایب فوریه به یکسان تضعیف شوند، سیگنال حاصله فقط ضعیفتر می شود، و بهیچوجه دچار اعوجاج (تغییر شکل) نخواهد شد. متأسفانه، در سیستمهای پخش مختلف ضرایب فوریه به میزانهای متفاوت تضعیف می شوند، و همین باعث بروز اعوجاج در شکل موج خواهد شد. معمولاً، تضعیف موج فقط از فرکانس خاصی به بالا (که به فرکانس قطع - f_c - معروف است و با هرتز - Hz - یا سیکل بر ثانیه - cycles/sec - سنجیده می شود) روی می دهد، و زیر این فرکانس تضعیف موج وجود ندارد. طیف فرکانسی که موج می تواند بدون تضعیف منتشر شود، به پهنای باند (bandwidth) معروف است. از آنجائیکه در عمل فرکانس قطع عددی دقیق و قطعی نیست، پهنای باند بعنوان فرکانسی که در آن فقط نیمی از انرژی موج عبور می کند، تعریف می شود.



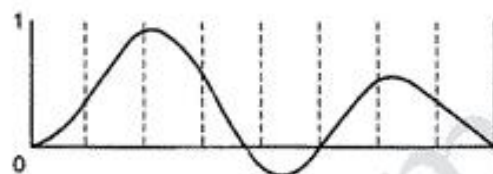
(الف)



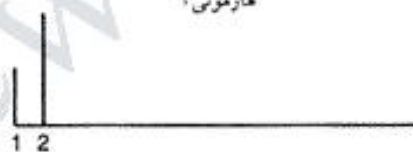
(ب)



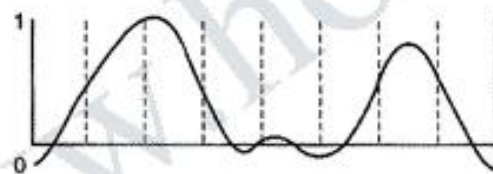
هارمونی ۱



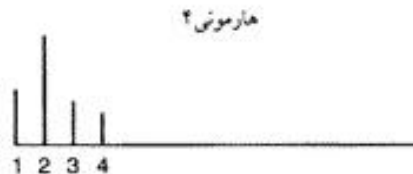
(ج)



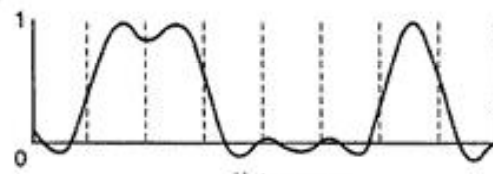
هارمونی ۲



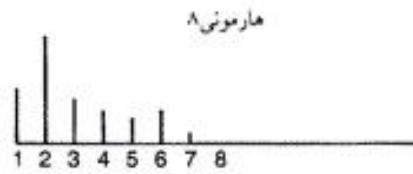
(د)



هارمونی ۴



(ه)



هارمونی ۸

شکل ۲-۱. (الف) یک سیگنال باینری و ضرایب فوریه rms آن. (ب)-(ه)

تقریب های متوالی سیگنال اولیه.

پهنای باند یکی از خواص فیزیکی رسانه انتقال است، و معمولاً به نوع، شکل، ضخامت و طول آن بستگی دارد. در برخی موارد با قرار دادن یک فیلتر پهنای باندی را که در اختیار مشتری است، عملاً محدود می کنند. برای مثال، پهنای باند یک خط تلفن در فواصل کوتاه می تواند به 1 MHz برسد، ولی شرکت های تلفن با قرار دادن

فیلترهای ویژه آنرا به حدود 3100 Hz محدود می کنند. این پهنای باند برای انتقال واضح و مفهوم صدا کافیست، و از اتلاف منابع نیز جلوگیری می کند.

حال اجازه دهید ببینیم اگر پهنای باند خط انتقال آنقدر کم باشد که فقط فرکانسهای پائین بتوانند منتقل شوند (بعبارت دیگر فقط هارمونی های اول معادله (2-1) اجازه عبور داشته باشند)، سیگنال شکل ۲-۱ (الف) به چه صورتی در می آید. در شکل ۲-۱ (ب) شکل موج را در حالتی که فقط هارمونی اول (هارمونی اصلی، f) اجازه عبور دارد، می بینید. در شکلهای ۲-۱ (ج) - (ه) نیز شکل موج برای پهنای باندهای بالاتر نشان داده شده اند. اگر نرخ انتقال اطلاعات b bits/sec باشد، زمان لازم برای ارسال (مثلاً) ۸ بیت معادل $8/b$ sec است، بنابراین فرکانس هارمونی اصلی $b/8$ Hz خواهد بود. همانطور که گفتیم، فرکانس یک خط تلفن معمولی (که به خط رده صوتی - voice-grade - معروف است) بصورت مصنوعی کمی بالای 3000 Hz نگه داشته می شود. این محدودیت باعث می شود تا بالاترین هارمونی که می تواند از این خط عبور کند، تقریباً $3000/(b/8)$ یا $24000/b$ باشد.

در شکل ۲-۲ اعداد محاسبه شده برای برخی از مهمترین نرخهای انتقال را می بینید. همانطور که از این شکل برمی آید، اگر بخواهیم داده ها را با سرعت 9600 bps روی یک خط رده صوتی بفرستیم، موج ما از شکل ۲-۱ (الف) به شکل ۲-۱ (ه) تبدیل خواهد شد - که باعث می شود تشخیص بیت های اولیه قدری مشکل شود. از همینجا پیداست که برای سرعت های بالای 38.4 kbps هیچ شانس برای ارسال داده های باینری وجود ندارد (حتی اگر خط ماکاملاً بدون نویز باشد). بعبارت دیگر، محدود کردن پهنای باند باعث محدود شدن نرخ انتقال اطلاعات (حتی در بهترین کانالها) خواهد شد. با این حال، راههایی وجود دارد (مثلاً، استفاده از چند سطح ولتاژ بجای در سطح ولتاژ) که می توان به نرخهای انتقال بالاتری دست یافت، و در ادامه همین فصل درباره آنها صحبت خواهیم کرد.

۳-۱-۲ حداکثر نرخ داده در یک کانال

در سال ۱۹۲۴ یکی از مهندسان AT&T بنام هنری نایکوئیست دریافت که حتی بهترین کانال انتقال هم ظرفیت محدودی دارد، و توانست معادله حداکثر نرخ انتقال داده یک کانال بدون نویز را برای یک پهنای باند مشخص بدست آورد. در سال ۱۹۴۸ کلود شانون کار نایکوئیست را تکمیل کرد، و معادله وی را برای حالتی که کانال در معرض نویز تصادفی (نویز ترمودینامیک یا حرارتی) است، توسعه دهد (Shannon, 1948). در این قسمت کارهای این دو دانشمند را بطور مختصر بررسی خواهیم کرد.

تعداد هارمونی فرستاده شده	اولین هارمونی (Hz)	T (msec)	Bps
80	37.5	26.67	300
40	75	13.33	600
20	150	6.67	1200
10	300	3.33	2400
5	600	1.67	4800
2	1200	0.83	9600
1	2400	0.42	19200
0	4800	0.21	38400

شکل ۲-۲. رابطه بین نرخ انتقال داده و هارمونی ها.

نایکونیست ثابت کرد که اگر سیگنالی از یک فیلتر پائین‌گذر (low-pass filter) با پهنای باند H عبور داده شود، سیگنال خروجی (فیلتر شده) را می‌توان با داشتن فقط $2H$ نمونه در ثانیه بازسازی کرد. نمونه‌برداری از خط با نرخ بالاتر از $2H$ دقت بیشتری به همراه نخواهد داشت، چون فرکانسهای بالاتری که بدین ترتیب می‌توان بازیابی کرد، قبلاً توسط فیلتر حذف شده‌اند. اگر این سیگنال دارای V سطح مجزا باشد، طبق قضیه نایکونیست:

$$\text{maximum data rate} = 2H \log_2 V \text{ bits/sec}$$

برای مثال، یک کانال 3-kHz بدون نویز نمی‌تواند سیگنالهای باینری (دو سطح ولتاژ) بالاتر از 6000 bps را انتقال دهد.

تا اینجا کانالها را بدون نویز فرض کردیم، که در واقع چنین نیست. اگر در کانال انتقال نویز تصادفی (حرارتی) وجود داشته باشد، اوضاع بسرعت رو به وخامت خواهد گذاشت (و می‌دانیم که هیچ گریزی از نویز حرارتی، که حاصل حرکت مولکولهاست، نیست). مقدار نویز حرارتی با نسبت توان سیگنال به توان نویز سنجیده می‌شود، و به نسبت سیگنال به نویز (signal-to-noise ratio) معروف است. اگر توان سیگنال را با S و توان نویز را با N نشان دهیم، نسبت سیگنال به نویز S/N خواهد بود. معمولاً آنچه که به عنوان نسبت سیگنال به نویز داده می‌شود، نه خود S/N بلکه $10 \log_{10} S/N$ است، که به دسی‌بل (decibel) معروف است و با dB نشان داده می‌شود. اگر نسبت سیگنال به نویز 10 باشد، معادل 10 دسی‌بل خواهد بود، نسبت 100 معادل 20 دسی‌بل است، نسبت 1000 معادل 30 دسی‌بل، و الی آخر. سازندگان تقویت‌کننده‌های استریو اغلب پهنای باندی را که دستگاه آنها بصورت خطی عمل می‌کند، بصورت فرکانس 3 دسی‌بل در هر انتهای طیف مشخص می‌کنند (فرکانس 3 دسی‌بل فرکانسی است که ضریب تقویت دستگاه نصف می‌شود، چون $\log_{10} 3 = 0.5$).

شانون معادله نایکونیست را برای کانالهای نویزدار بصورت زیر تصحیح کرد:

$$\text{maximum number of bits/sec} = H \log_2 (1 + S/N)$$

که در آن H پهنای باند کانال (بر حسب هرتز)، و S/N نسبت سیگنال به نویز است. برای مثال، در کانالی با پهنای باند 3000 Hz که دارای نویز حرارتی 30 dB است (نویز معمول در بخش آنالوگ سیستمهای تلفن)، هرگز نمی‌توان بیش از 30,000 bps داده ارسال کرد (بدون توجه به اینکه تعداد سطوح ولتاژ این سیگنال یا تعداد نمونه‌برداری‌ها چقدر باشد).

معادله شانون از تئوری اطلاعات (information theory) استنتاج شده، و برای هر کانالی که تحت تأثیر نویز حرارتی باشد صادق است. کانالی که این قاعده را نقض کند، باید معادل یک ماشین حرکت دائم فرض کرد (ماشینی که طبق اصول ترمودینامیک نمی‌تواند وجود خارجی داشته باشد). البته باید توجه داشتید که این عدد فقط حد بالای ظرفیت کانال را مشخص می‌کند، و سیستمهای واقعی بندرت به این حد دست پیدا می‌کنند.

۲-۲ رسانه انتقال هدایت‌پذیر

وظیفه لایه فیزیکی انتقال بیت‌های خام از یک ماشین به ماشین دیگر است. برای اینکار از رسانه‌های فیزیکی مختلفی می‌توان استفاده کرد، که هر کدام پهنای باند، تأخیر انتشار، هزینه، و سهولت نصب و نگهداری خاص خود را دارند. این رسانه‌ها را می‌توان به دو دسته کلی هدایت‌پذیر (مانند سیم مسی یا فیبر نوری) و هدایت‌ناپذیر (مانند امواج رادیویی و لیزری) تقسیم‌بندی کرد. در قسمتهای آینده این رسانه‌ها را مورد بررسی قرار خواهیم داد.

۱-۲-۲ رسانه مغناطیسی

یکی از متداولترین راههای انتقال اطلاعات از کامپیوتری به کامپیوتر دیگر، نوشتن آنها روی نوار مغناطیسی، دیسک، CD یا DVD و سپس خواندن آنها در کامپیوتر مقصد است. اگرچه این روش بخوبی استفاده از مخابرات

ماهواره ای نیست، ولی هزینه آن بسیار کمتر است، بویژه در مواردی که پهنای باند زیاد و قیمت کم جزء عوامل کلیدی باشند.

اجازه دهید با یک مثال ساده مطلب را روشن تر کنیم. امروزه نوارهای مغناطیسی با ظرفیت 200 GB بوفور در بازار یافت می شوند، و یک جعبه مقوایی به ابعاد $60 \times 60 \times 60$ cm می تواند 1000 تا از این نوارها را در خود جای دهد، که بدین ترتیب ظرفیت آن به 200 TB (تراپایت، معادل هزار گیگابایت) یا 1600 Tbit (که معادل 1.6 Pbit - پتابایت - است) می رسد. این بسته را می توان با استفاده از پست سریع السیر در کمتر از ۲۴ ساعت به هر نقطه ای در ایالات متحده آمریکا تحویل داد. بدین ترتیب، پهنای باند مؤثر این سیستم انتقال 1600 terabits/86,400 sec یا 19 Gbps است - و اگر فاصله این دو نقطه فقط یک ساعت باشد، پهنای باند تا 400 Gbps نیز افزایش خواهد یافت. هیچ شبکه کامپیوتری حتی نمی تواند به این سرعت نزدیک شود.

برای شبکه های بانکی که در هر روز باید چندین گیگابایت اطلاعات را از نقطه ای به نقطه دیگر منتقل کنند (تا در صورت بروز فجایع طبیعی نیز بتوانند به کار خود ادامه دهند) هیچ چیز نمی تواند جای نوار مغناطیسی را بگیرد. البته شبکه ها هر روز سریعتر می شوند، ولی ظرفیت نوارهای مغناطیسی نیز رو به افزایش است.

در مورد هزینه نیز وضعیت مشابهی وجود دارد. قیمت هر نوار مغناطیسی 200 GB چیزی حدود 40 دلار است (البته در خریدهای عمده)، و از هر نوار می توان حداقل ۱۰ بار استفاده کرد. بنابراین، هزینه هر بار استفاده از جعبه نوار ۱۰۰۰ تا 4000 دلار خواهد شد. اگر 1000 دلار نیز برای حمل و نقل به آن اضافه کنیم (که البته معمولاً بسیار کمتر است)، هزینه کل به 5000 دلار برای انتقال 200 TB اطلاعات می رسد - و این یعنی ۳ سنت (معادل 0.01 دلار) برای هر گیگابایت، که هیچ شبکه ای نمی تواند با آن رقابت کند. نتیجه اخلاقی داستان:

هرگز پهنای باند کامیونی پُر از نوارهای مغناطیسی را که سرعت در بزرگراه در حال حرکت است، دست کم نگیرید.

۲-۲-۲ زوج تابیده

با اینکه پهنای باند نوار مغناطیسی بسیار عالیست، تأخیر انتشار آن (زمانی که طول می کشد تا اولین بیت اطلاعات به مقصد برسد) ناامیدکننده است - در شبکه ها معمولاً با میلی ثانیه سروکار داریم نه روز و ساعت! در بسیاری از کاربردها برقرار بودن دائمی ارتباط یک امر حیاتی است. یکی از قدیمی ترین (و همچنان متداولترین) رسانه های انتقال زوج تابیده (twisted pair) است. زوج تابیده عبارتست از یک زوج سیم مسی عایق دار (ضخامت 1 mm)، که صورت مارپیچ به دور یکدیگر تابیده اند (مانند زنجیره مولکول DNA). علت تابیدن سیمها آنست که دو سیم مسی معمولی مانند یک آنتن عمل کرده، و انرژی تلف می کنند. در حالت تابیده امواج سیمها یکدیگر را خنثی کرده، و تشعشع به حداقل می رسد.

بیشترین کاربرد زوج تابیده در شبکه های تلفن است: تقریباً تمام تلفنها با استفاده از زوج تابیده به مرکز تلفن وصل می شوند. از زوجهای تابیده می توان بطول چندین کیلومتر بدون نیاز به تقویت کننده استفاده کرد، ولی برای مسافتهای طولانیتر به تکرارکننده (repeater) نیاز هست. تعداد زیادی زوج تابیده که در یک غلاف محافظ جمع شده باشند، تشکیل یک کابل زوج تابیده را می دهند. اگر سیمهای این کابل بصورت دو به دو به یکدیگر نتابیده باشند، تداخل شدیدی بین آنها بوجود خواهد آمد.

از زوجهای تابیده برای انتقال سیگنالهای آنالوگ و دیجیتال می توان استفاده کرد. پهنای باند این خطوط به ضخامت سیمها و مسافت بستگی دارد، و در فواصل کوتاه (دو تا سه کیلومتر) می توان به پهنای باند چندین مگابیت بر ثانیه دست یافت. بدلیل کارایی کافی و هزینه پائین، از زوج تابیده بنحوی گسترده ای استفاده شده است، و بنظر می رسد تا سالها نیز این وضعیت ادامه یابد.



(الف)

(ب)

شکل ۲-۳. (الف) زوج تاییده Cat 3. (ب) زوج تاییده Cat 5.

انواع مختلفی از کابلهای زوج تاییده وجود دارد، که دو تا از آنها در شبکه های کامپیوتری از اهمیت بیشتری برخوردار هستند. در کابلهای Category 3 (که به Cat 3 نیز معروف است) سیمها با شدت کمتری به هم تاییده اند. کابل Cat 3 از چهار زوج تاییده، که در یک غلاف پلاستیکی قرار می گیرند، تشکیل می شود. تا قبل از سال ۱۹۸۸ تقریباً در همه جا از این نوع کابل استفاده می شد. کابل Cat 3 می توانست چهار خط تلفن معمولی (یا دو خط تلفن مرکب) را از ایستگاه تلفن به نقطه مورد نظر برساند.

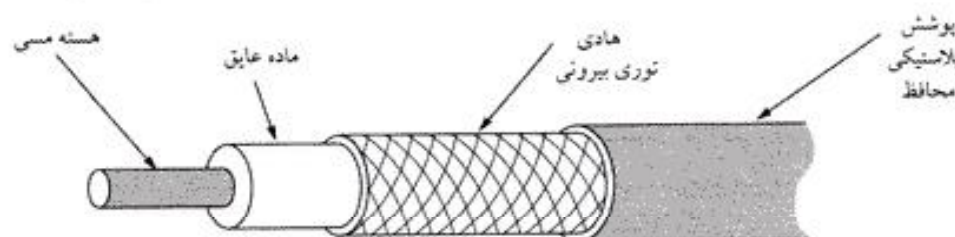
در سال ۱۹۸۸ کابل پیشرفته تر Category 5 (یا همان Cat 5) وارد بازار شد. این کابل شبیه Cat 3 است، با این تفاوت که تعداد دورهای آن در واحد طول بیشتر بوده، و بهمین دلیل تداخل سیگنال در آن کاهش یافته و برای شبکه های پرسرعت مناسبتر است. کابلهای Cat 6 و Cat 7 نیز در حال آمدن به بازار هستند، که سرعت آنها بترتیب به 250 MHz و 600 MHz می رسد (برای مقایسه، سرعت کابلهای Cat 3 و Cat 5 بترتیب 16 MHz و 100 MHz است).

تمام این کابلها در بازار بنام UTP (زوج تاییده بدون زره - Unshielded Twisted Pair) شناخته می شوند، تا از کابلهای STP (زوج تاییده زره دار - Shielded Twisted Pair) که IBM در اوایل دهه ۱۹۸۰ معرفی کرد (و بدلیل گرانی و سختی کار با آن، استقبال چندانی از این نوع کابل نشد)، متمایز باشند. در شکل ۲-۳ دو نوع زوج تاییده Cat 3 و Cat 5 (و تفاوت آنها) را ملاحظه می کنید.

۲-۲-۲ کابل کواکسیال

یکی دیگر از رسانه های رایج کابل کواکسیال (هم محور - coaxial) است (که اغلب به آن کابل کواکس می گویند). این کابل بدلیل داشتن زره (غلاف فلزی) کارایی بهتری نسبت به زوج تاییده (هم از نظر سرعت، هم از نظر مسافت) دارد. کابل کواکسیال دو نوع دارد. اولی کابل 50-ohm است، که از ابتدا برای مخابرات دیجیتال در نظر گرفته شده؛ و دومی کابل 75-ohm، که ابتدا برای مخابرات آنالوگ و تلویزیون کابلی بکار می رفت، ولی امروزه با گسترش اینترنت کابلی از اهمیت روزافزونی برخوردار شده است. این تمایز بیش از آن که فنی باشد، جنبه تاریخی دارد: امپدانس آنتنهای دوقطبی قدیمی 300-ohm بود، و ترانسفورماتورهای تطبیق امپدانس 4:1 بوفور در بازار یافت می شد.

کابل کواکسیال تشکیل می شود از یک سیم مسی سخت بعنوان هسته (core)، یک لایه عایق استوانه ای به دور این هسته، یک لایه توری فلزی که به دور عایق بافته شده، و لایه پلاستیکی محافظ خارجی (شکل ۲-۴).



شکل ۲-۴. ساختمان کابل کواکسیال.

ساختمان و نحوه عایق بندی کابل کواکسیال باعث شده تا این نوع کابل از نظر سرعت و مصنوعیت در مقابل نویز کارایی بسیار خوبی داشته باشد. پهنای باند کابل های کواکسیال به کیفیت مواد آن، طول کابل و نسبت سیگنال به نویز امواج ارسالی بستگی دارد، و در کابل های جدید به 1 GHz نیز می رسد. از کابل کواکسیال بیشتر در سیستم های تلفن راه دور استفاده می شد، که امروزه بتدریج جای خود را به فیبر های نوری می دهد. با این حال، در شبکه های شهری و تلویزیون کابلی هنوز هیچ رقیبی برای کابل کواکس وجود ندارد.

۴-۲-۲ فیبر نوری

بسیاری از افرادی که در صنعت کامپیوتر کار می کنند، از شتاب خیره کننده رشد تکنولوژی آن به خود می بالند. اولین کامپیوتر شخصی (PC) که IBM در سال ۱۹۸۱ به بازار عرضه کرد، با سرعت ساعت 4.77 MHz کار می کرد؛ بعد از گذشت بیست سال، امروزه PC ها می توانند با سرعتی متجاوز از 2 GHz کار کنند، و این یعنی ۲۰ برابر شدن سرعت در هر ۱۰ سال - چندان هم بد نیست!

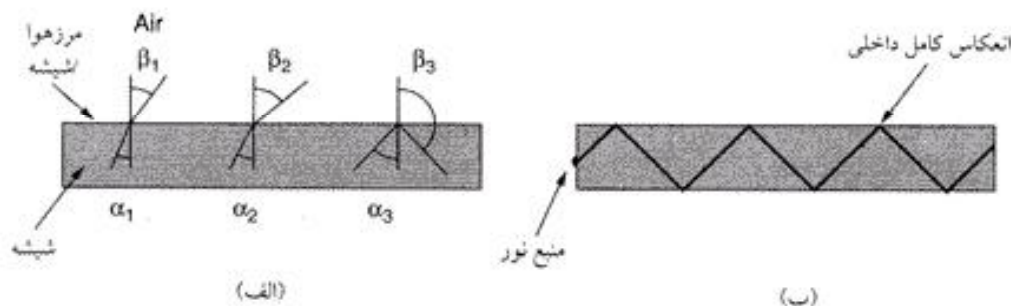
در همین دوره، سرعت مخابرات راه دور از 56-kbps (آرپانت) به 1 Gbps (مخابرات نوری جدید) رسیده است - این یعنی رشدی معادل ۱۲۵ برابر در هر دهه (در حالیکه میزان خطا هم از یک بیت در هر 10^8 بیت به تقریباً صفر رسیده).

اما پیشرفت هم مرزهای فیزیکی خاص خود را دارد؛ CPU ها مدتیست با موانعی از قبیل محدودیت سرعت نور و مشکل اتلاف گرما روبرو هستند. در حالیکه این مسائل در مخابرات وجود ندارد، و تکنولوژی فیبر نوری می تواند ب راحتی به پهنای باندی فراتر از 50,000 Gbps (یا 50 Tbps) دست یابد (و از هم اکنون بسیاری افراد با جدیت بدنبال تکنولوژی های بهتر نیز هستند). سرعت فعلی مخابرات فیبر نوری به چیزی در حدود 10 Gbps محدود می شود، ولی علت اصلی آن محدودیت سرعت تبدیل سیگنال های الکتریکی به پالس های نوری است، نه محدودیت ذاتی فیبر های نوری (البته در آزمایشگاه ها به سرعت 100 Gbps نیز دست یافته اند).

در مسابقه بین کامپیوتر و مخابرات، مسلماً مخابرات برنده است. البته هنوز عده زیادی از دانشمندان و مهندسان کامپیوتر (که به قانون نایکوئیست و شانون فکر می کنند، و نمی دانند که این قانون ها مربوط به سیگنال های مسی است) نمی توانند تصور پهنای باند نامحدود را به ذهن خود راه دهند. قانون آینده باید این باشد که سرعت کامپیوترها هرگز به گرد شبکه ها نیز نخواهد رسید، و باید از هر نوع پردازشی روی اطلاعات شبکه خودداری کرد (حتی اگر به معنای تلف شدن مقداری از پهنای باند باشد). اما اجازه دهید ببینیم فیبر نوری چگونه کار می کند.

یک سیستم انتقال نوری سه مولفه کلیدی دارد: منبع نور، رسانه انتقال، آشکارساز. طبق قرارداد، یک پالس نوری معادل بیت 1 و فقدان نور معادل بیت 0 است. رسانه انتقال یک رشته (فیبر) فوق العاده نازک شیشه است. آشکارساز (detector) نیز دستگاهیست که با برخورد نور به آن یک پالس الکتریکی تولید می کند. با قرار دادن یک منبع نور در یک سر فیبر نوری و یک آشکارساز در سمت دیگر، یک سیستم انتقال نوری یکطرفه خواهیم داشت که سیگنال الکتریکی را گرفته، آنرا به پالس های نوری تبدیل کرده، و در طرف دیگر پالس دریافتی را به سیگنال های الکتریکی تبدیل می کند.

اما می دانیم که نور از شیشه خارج می شود، و چنین سیستمی بکلی بی فایده خواهد بود. اینجاست که یکی از قوانین جالب فیزیک نور به کمک ما می آید. این قانون (که به قانون شکست نور معروف است) می گوید که وقتی پرتو نور از یک محیط (مثلاً، شیشه) وارد محیط دیگر (مثلاً، هوا) می شود، در مرز این دو محیط دچار خمیدگی یا شکست (refraction) می شود. به شکل ۲-۵ (الف) نگاه کنید؛ در این شکل یک پرتو نور می بینید که با زاویه α به مرز شیشه و هوا برخورد کرده، و با زاویه β از شیشه خارج و وارد هوا می شود. مقدار خمیدگی یا شکست پرتو نور به خواص فیزیکی دو محیط (بویژه ضریب شکست آنها) بستگی دارد. اگر زاویه برخورد نور از یک مقدار



شکل ۲-۵. (الف) سه مثال از پرتوهای نوری که به مرز شیشه-هوا برخورد کرده، و شکسته می‌شوند. (ب) پرتو نور بدلیل شکست کلی در داخل شیشه گرفتار شده است.

بحرانی بیشتر باشد، پرتو نور دچار شکست کلی شده و دوباره به داخل شیشه برمی‌گردد، و هرگز وارد هوا نخواهد شد. نوری که با این زاویه (یا بیشتر از آن) به داخل شیشه تابانده شود، برای همیشه در آن محبوس می‌شود، و می‌تواند مسافت‌های طولانی را بدون اتلاف انرژی در فیبر نوری بپیماید - شکل ۲-۵ (ب) را ببینید. در شکل ۲-۵ (ب) فقط یک پرتو نور نشان داده شده است، ولی از آنجائیکه هر پرتو نوری که با زاویه بالاتر از زاویه بحرانی به مرز شیشه و هوا برخورد کند به داخل شیشه برمی‌گردد، در هر لحظه پرتوهای متعددی با زاویه‌های مختلف در داخل فیبر نوری به بالا و پائین حرکت می‌کنند، و اصطلاحاً گفته می‌شود که هر یک از این پرتوها دارای حالت (mode) خاص خود است. به فیبری که چنین ویژگی داشته باشد، فیبر چندحالت (multimode fiber) گفته می‌شود.

ولی اگر قطر فیبر فقط چند برابر طول موج نور باشد، پرتو نور فقط می‌تواند در جهت مستقیم (بدون جهش به بالا و پائین) حرکت کند. به چنین فیبری که بصورت یک موج‌بر (wave guide) عمل می‌کند، فیبر تک‌حالت (single-mode fiber) گفته می‌شود. فیبرهای تک‌حالته گرانتر از فیبرهای چندحالته هستند، ولی در مسافت‌های طولانی اغلب از این نوع فیبرها استفاده می‌شود. فیبرهای تک‌حالته امروزی می‌توانند با ظرفیتهایی تا 50 Gbps و بطول 100 کیلومتر بدون نیاز به تقویت‌کننده کار کنند. (در آزمایشگاهها برای مسافت‌های کوتاهتر به پهنای باند بالاتری نیز دست یافته‌اند.)

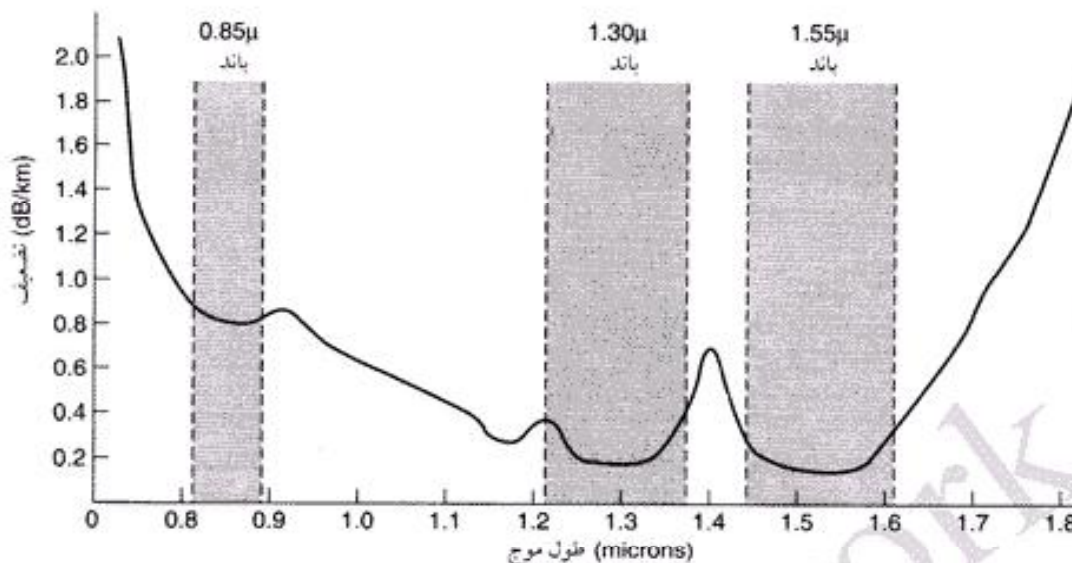
عبور نور در دورن فیبر نوری

فیبرهای نوری از شیشه ساخته می‌شوند و خود شیشه هم از شن، ماده‌ای که بوفور در طبیعت یافت می‌شود. شیشه برای اولین بار در مصر باستان ساخته شد، ولی این شیشه‌ها بقدری کدر بودند که در ضخامتهای بیشتر از 1 mm نور را از خود عبور نمی‌دادند. شیشه‌ای که برای استفاده در پنجره‌ها مناسب باشد، در دوره رنسانس ساخته شد. شیشه‌ای که در ساخت فیبرهای نوری بکار می‌رود آنقدر شفاف است که اگر اقیانوس را با آن پر کنیم، کف آن بوضوح دیده خواهد شد.

تضعیف نور (attenuation) در شیشه به طول موج آن و برخی از خواص فیزیکی شیشه بستگی دارد. در شکل ۲-۶ میزان تضعیف نور در فیبرهای نوری بر حسب دسی‌بل بر کیلومتر (خطی) نشان داده شده است. معادله تضعیف نور چنین است:

$$\text{attenuation (dB)} = 10 \log_{10} \frac{\text{transmitted power}}{\text{received power}}$$

برای مثال، اگر قدرت دریافتی در خروجی نصف ورودی باشد، میزان تضعیف سیگنال ۲-۶ است. در شکل ۲-۶ نمودار تضعیف نور را در ناحیه مادون قرمز طیف، که در عمل نیز از آن



شکل ۲-۶. تضعیف نور عبوری از فیبر نوری در ناحیه مادون قرمز.

استفاده می شود، می بینید. طول موج نور مرئی (0.4-0.7 میکرون، یا 400-700 نانومتر) قدری کوتاهتر از نور مادون قرمز است ($1 \mu\text{m} = 10^{-6} \text{ m}$ و $1 \text{ nm} = 10^{-9} \text{ m}$).

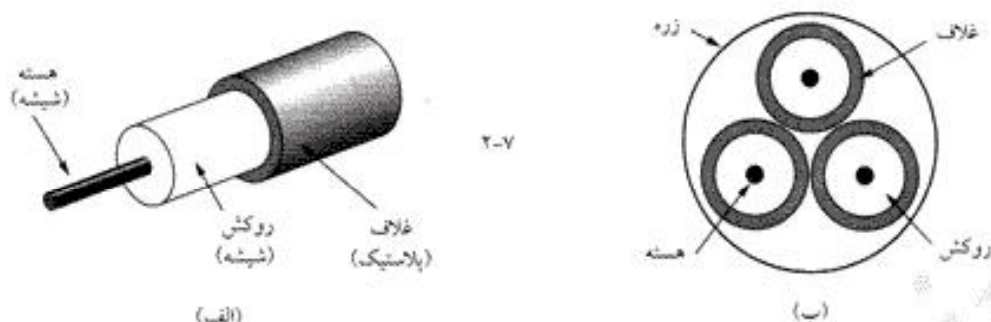
برای مخابرات نوری سه باند از طول موجهای نور مادون قرمز مورد استفاده قرار می گیرد، که بترتیب حول طول موجهای 0.85، 1.30 و 1.55 میکرون متمرکز شده اند. دو باند آخر دارای مشخصات تضعیف خوبی (کمتر از ۵ درصد در هر کیلومتر) هستند. باند 0.85 میکرون میزان تضعیف بالاتری دارد، ولی در این طول موج می توان تجهیزات نوری و الکترونیکی را از یک ماده نیمه هادی واحد (گالیوم آرسنید) ساخت. پهنای این باندها همگی بین 25,000 GHz تا 30,000 GHz است.

وقتی پالسهای نور از فیبر نوری عبور می کنند، پهنای آنها زیاد می شود، که این پدیده به پراکنش کروماتیک (chromatic dispersion) معروف است، و مقدار آن به طول موج نور بستگی دارد. یکی از راههای جلوگیری از تداخل این پالسهای پهن شده، افزایش فاصله آنهاست، و این کار نیز فقط با کاهش نرخ ارسال سیگنال ممکن است. خوشبختانه با کشف این موضوع که می توان با تولید پالسهای که تقارن کسینوس هذلولی دارند، اثر پراکنش کروماتیک را بکلی حذف کرد، امکان آن بوجود آمده که پالسهای نوری را بدون تغییر شکل محسوس تا مسافتهای هزاران کیلومتری مخابره کرد. امروزه تلاشهای زیادی در دست انجام است تا این پالسها را، که به آنها سولیتون (soliton) گفته می شود، از حالت آزمایشگاهی خارج و در عمل بکار بگیرند.

کابل فیبر نوری

کابل فیبر نوری بسیار شبیه کابل کواکسیال است، با این تفاوت که غلاف توری فلزی بیرونی را ندارد (به شکل ۲-۷ (الف) نگاه کنید). رشته شیشه ای که نور از آن عبور می کند، در مرکز کابل قرار دارد. در فیبرهای چندحالتی قطر این رشته معمولاً $50 \mu\text{m}$ (تقریباً معادل ضخامت موی سر انسان) است، و در فیبرهای تک حالتی قطر آن به 8 تا 10 میکرون می رسد.

هسته فیبر نوری با یک روکش شیشه ای (با ضریب شکست کمتر) پوشانده می شود، تا تمام نور در رشته مرکزی باقی بماند. پوشش بیرونی نیز (که اغلب پلاستیکی است) نقش محافظ را بازی می کند. معمولاً چند کابل تک رشته را در یک غلاف گرد هم می آورند (شکل ۲-۷ (ب) را ببینید).



شکل ۲-۷. (الف) نمای کناری یک فیبر منفرد. (ب) سطح مقطع کابلی با سه فیبر.

کابلهای زمینی معمولاً در عمق یک متری سطح زمین دفن می‌شوند (که در آنجا از آسیب پیلهای مکانیکی یا موشهای جونده در امان نیستند). در قسمتهای کم عمق ساحل، کابلهای زیر دریایی را در کانالهای مخصوص پنهان می‌کنند، ولی در آبهای عمیق (که کندن کانال عملی نیست) آنها را آزاد در کف دریا رها می‌کنند (و با این کار آنها را در معرض آسیب از طرف کشتی‌های ماهیگیری و اسکونیدهای غول‌پیکر قرار می‌دهند). سه روش برای متصل کردن فیبرهای نوری وجود دارد. در روش اول، به انتهای کابل پایانه‌های مخصوص وصل کرده، و آنها را به سوکت فیبر نوری متصل می‌کنند. این پایانه‌ها ۱۰ تا ۲۰ درصد نور را تلف می‌کنند، ولی کار با آنها بسیار ساده است.

دوم اینکه، می‌توان رشته‌ها را بطور مکانیکی بهم متصل کرد. در این روش دو سر رشته‌ها را که با دقت بریده شده‌اند، در یک غلاف روبروی هم قرار می‌دهند، و آنها را در جای خود محکم می‌کنند. در این روش با عبور دادن نور و تنظیم رشته‌ها می‌توان به حداکثر سیگنال عبوری دست یافت. اتلاف نور در این روش فقط ۱۰ درصد است، و یک فرد تعلیم دیده می‌تواند در عرض ۵ دقیقه چنین اتصالی را بوجود آورد.

در روش سوم، دو سر رشته‌ها ذوب و در هم فرو برده می‌شود، تا یک اتصال یکپارچه بوجود آید. این بهترین نوع اتصال است، چون رشته‌ها در واقع یکی می‌شوند، ولی حتی در این روش هم مقداری افت توان وجود دارد. در هر سه روش فوق، محل اتصال می‌تواند مقدارن از نور را بازتابش کند، که این نور با سیگنال اصلی تداخل خواهد کرد.

تبدیل سیگنالهای الکتریکی به پالسهای نوری معمولاً به دو روش صورت می‌گیرد: لیزرهای نیمه‌هادی و LED (Light Emitting Device). هر کدام از این منابع نوری ویژگیهای خاص خود را دارند (به شکل ۲-۸ نگاه کنید). طول موج یک منبع نور را می‌توان با قرار دادن تداخل‌سنج (interferometer) هایی از نوع فابری-پروت (Fabry-Perot) یا ماخ-زئندر (Mach-Zehnder) بین منبع نور و فیبر نوری تنظیم کرد. تداخل‌سنج فابری-پروت یک حفره تشدید (resonant cavity) ساده است، که از دو آینه موازی تشکیل می‌شود. نور بصورت عمود به این آینه‌ها تابانده می‌شود، و فقط طول موجهایی که مضرب صحیحی از طول حفره باشند، می‌توانند از آن خارج شوند. در تداخل‌سنج ماخ-زئندر نور به دو پرتو جداگانه تقسیم می‌شود، که پس از طی مسافتی کوتاه دوباره با هم ترکیب می‌شوند؛ این دو پرتو فقط در طول موجهای خاصی با یکدیگر همگام هستند. در انتهای دیگر فیبر نوری یک فتودیود (photodiode) قرار می‌گیرد، که با هر پالس نوری یک سیگنال الکتریکی تولید می‌کند. زمان پاسخ این نوع دیودها معمولاً ۱ nsec است ($1 \text{ nsec} = 10^{-9} \text{ sec}$)، که باعث می‌شود نرخ داده‌ها به 1 Gbps محدود شود. از آنجائیکه نویز حرارتی در اینجا هم می‌تواند باعث بروز مشکل می‌شود، پالس نوری باید آنقدر انرژی داشته باشد که بتوان آنرا از نویز تشخیص داد (در فیبرهای نوری می‌توان ضریب خطا را با افزایش انرژی پالسهای نوری بمیزان دلخواه پائین آورد).

لیزر نیمه هادی	LED	آیتم
زیاد	کم	سرعت داده
چند حالت یا تک حالت	چند حالت	نوع فیبر
بلند	کوتاه	فاصله
کم	زیاد	طول عمر
قابل توجه	کم	حساسیت به دما
زیاد	کم	قیمت

شکل ۲-۸. مقایسه ای بین لیزرهای نیمه هادی و LED بعنوان منبع نور.

شبکه های فیبر نوری

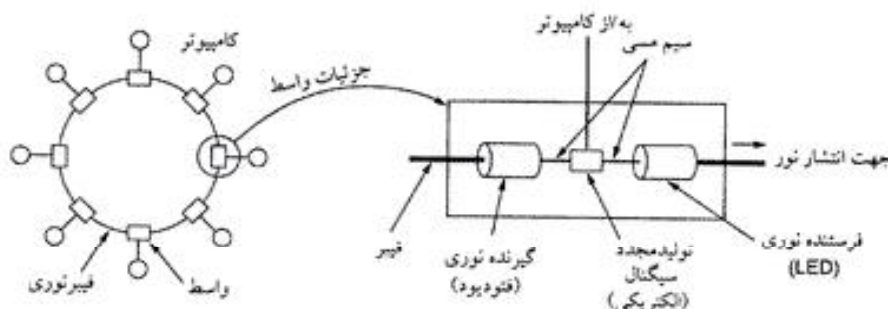
از فیبر نوری، علاوه بر مخابرات راه دور، در شبکه های محلی نیز می توان استفاده کرد - اگرچه این کار نسبت به اینترنت با مشکلات بیشتری همراه است. یک شبکه حلقوی را در نظر بگیرید (شکل ۲-۹) - این شبکه در واقع مجموعه ایست از چند اتصال نقطه-به-نقطه. هر کامپیوتر یک اتصال T (T junction) به شبکه دارد، که می تواند پالسهای نور را از خود عبور دهد، یا آنها را دریافت کند.

دو نوع اتصال وجود دارد: غیرفعال (passive) و فعال (active). در اتصال غیرفعال دو تویی به فیبر اصلی متصل می شود، که یکی LED یا دیود لیزری دارد (برای ارسال)، و دیگری فتودیود (برای دریافت). خود تویی هیچ عنصر فعالی ندارد و به همین دلیل فوق العاده قابل اعتماد است، چون خراب شدن LED یا فتودیود باعث قطعی شبکه نخواهد شد و فقط همان یک کامپیوتر از شبکه قطع می شود.

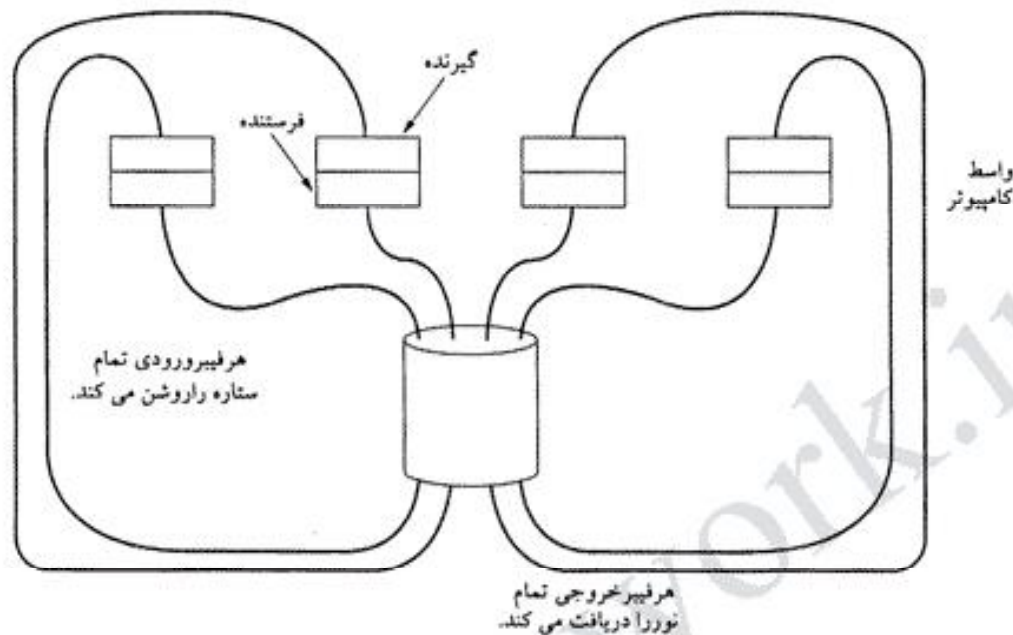
اتصال دیگری که در شکل ۲-۹ می بینید، تکرارکننده فعال (active repeater) است. در اینجا، نور ورودی ابتدا به سیگنال الکتریکی تبدیل شده، در صورت نیاز تقویت و دوباره بصورت نور منتشر می شود. اتصال به کامپیوتر توسط یک سیم مسی ساده که از تقویت کننده سیگنال منشعب شده، برقرار می شود. امروزه تقویت کننده های تمام نوری (که مرحله تبدیل به سیگنال الکتریکی - و بالعکس - در آنها حذف شده) نیز به بازار آمده اند؛ پهنای باند این تکرارکننده ها بسیار بالاست.

اگر یک تکرارکننده فعال خراب شود، حلقه شکسته شده و کل شبکه از کار می افتد. از طرف دیگر به علت تقویت سیگنال، فاصله کامپیوترها در این شبکه می تواند به چندین کیلومتر برسد، و از نظر تعداد کامپیوترها هم هیچ محدودیتی وجود ندارد. (در نوع غیرفعال، هر اتصال مقداری از توان نوری را هدر می دهد، بنابراین تعداد کامپیوترها و طول کل حلقه بشدت محدود است.)

توپولوژی حلقه تنها راه برای ایجاد شبکه های محلی با فیبر نوری نیست، و می توان به کمک سخت افزارهای خاص (انتشاردهنده های نوری) ساختاری موسوم به ستاره غیرفعال (passive star) بوجود آورد (شکل ۲-۱۰).



شکل ۲-۹. یک شبکه حلقوی فیبر نوری، با تکرارکننده های فعال.



شکل ۲-۱۰. شبکه ای با ساختار ستاره غیر فعال.

انتشار دهنده نوری یک استوانه شیشه ای خاص است، که فیبرهای ورودی (گیرنده) به یک طرف و فیبرهای خروجی (فرستنده) به طرف دیگر آن جوش خورده اند. وقتی یک کامپیوتر سیگنالی می فرستد، پالس نوری حاصله از طریق این استوانه در تمام فیبرهای متصل به گیرنده ها پخش می شود. در واقع، ستاره غیر فعال پالسهای ورودی را ترکیب کرده، و نور حاصله را روی تمام فیبرها می فرستد. از آنجائیکه انرژی نوری دریافتی در ستاره غیر فعال بین تمام رشته ها پخش می شود، تعداد کامپیوترهای متصل به این شبکه محدود (و وابسته به حساسیت فتودیودها) است.

مقایسه فیبر نوری و سیم مسی

در اینجا بد نیست مقایسه ای بین فیبر نوری و سیم مسی داشته باشیم؛ حتماً آموزنده خواهد بود. فیبر نوری مزایای متعددی نسبت به سیم مسی دارد. از همه مهمتر اینکه پهنای باند آن بسیار بیشتر از سیم مسی است، و همین دلیل کافیت تا در شبکه های پرسرعت انتخاب اول باشد. دیگر اینکه بدلیل اتلاف قدرت ناچیز فقط در هر ۵۰ کیلومتر به تکرار کننده نیاز دارد، در حالیکه این مسافت برای سیم مسی ۵ کیلومتر بیشتر نیست، که این باعث صرفه جویی زیادی در هزینه ها خواهد شد. مزیت دیگر فیبر نوری عدم تأثیر پذیری آن نسبت به نویز و تداخل های الکترومغناطیسی است. مقاومت عالی شیشه در مقابل خوردگی های شیمیایی نیز یکی دیگر از نقاط قوت فیبر نوری محسوب می شود.

جالب است بدانید که علت علاقه شرکت های تلفن به فیبر نوری اساساً چیز دیگریست: نازکی و سبکی. کانالهای زیرزمینی که شرکت های تلفن به نقاط مختلف حفر کرده اند، اکنون تا حد اشباع پُر شده اند، و دیگر جای خالی برای اشعاعات جدید ندارند. جایگزین کردن کابل های مسی با فیبر نوری، علاوه بر خالی کردن این کانالها، درآمد جدیدی نیز برای این شرکت ها به همراه دارد: فروش مس خالص به پالایشگاه های مس. علاوه بر آن، فیبرهای نوری بسیار سبکتر از سیمهای مسی هستند: هر کیلومتر از کابل مسی با هزار زوج تابیده نزدیک به ۸۰۰۰ کیلوگرم وزن

دارد، در حالیکه وزن کابل فیبر نوری با همان ظرفیت فقط ۱۰۰ کیلوگرم است. کاهش وزن نیز معادل است با کاهش نیاز به وسایل مکانیکی سنگین برای نصب و پشتیبانی سیستم، بگونه ای که امروزه دیگر در مسیرهای جدید فقط از فیبر نوری استفاده می شود.

آخر اینکه، نور از فیبرهای نوری نشت نمی کند، و گرفتن اشعاع غیرمجاز از آن بسیار مشکل است. این ویژگی باعث شده تا فیبر نوری از ایمنی بسیار بالایی در مقابل سارقان اطلاعات برخوردار باشد.

البته فیبر نوری چندان بی عیب و نقص هم نیست. اول اینکه، این تکنولوژی هنوز بسیار جدید است و حتی بسیاری از مهندسين نیز توانایی کار با آن را ندارند. دیگر اینکه فیبر نوری بسیار آسیب پذیرتر از سیم مسی است، و حتی خم کردن بیش از حد باعث خرابی آن می شود. از آنجائیکه انتقال نوری ذاتاً یک طرفه است، برای ارتباط دوطرفه باید از دو رشته فیبر (یا یک رشته فیبر با دو باند فرکانسی) استفاده کرد. دست آخر اینکه، تجهیزات ارتباطی نوری گرانتر از انواع الکتریکی آن است. با این وجود، آینده مخابرات در فواصلی حتی بیش از چند متر از آن فیبرهای نوری است. برای بحثی جامع درباره جنبه های مختلف فیبر نوری و شبکه های آن، به (Hecht, 2001) مراجعه کنید.

۳-۲ انتقال بیسیم

یکی از پدیده های عصر ما معنادان اینترنتی است: کسانی که می خواهند بیست و چهار ساعته برخط (on-line) باشند. برای این قبیل افراد (که دائماً در حال جابجا شدن هستند) دیگر زوج نابیده، کابل کوآکس و فیبر نوری کاربرد ندارد. آنها می خواهند بدون مقید شدن به هیچیک از سیستمهای مخابراتی دست و پا گیر، اطلاعات مورد نیازشان را روی کامپیوتر کیفی، جیبی و حتی مچی (۱) دریافت کنند؛ و مخابرات بیسیم تنها چیزیست که می تواند توقعات اینها را برآورده کند. در این قسمت نگاهی کلی به مخابرات بیسیم خواهیم انداخت، نه فقط بخاطر اینکه عده ای دوست دارند کنار دریا هم از دوستان اینترنتی شان جدا نشوند، بلکه به دلیل آنکه این تکنولوژی نقش مهمی در زندگی امروزی ما بازی می کند.

برخی افراد معتقدند که در آینده فقط دو نوع مخابرات وجود خواهد داشت: فیبر نوری و بیسیم. تمام تجهیزات ثابت (کامپیوترهای رومیزی، تلفن و فکس) از فیبر نوری، و تجهیزات متحرک از بیسیم بهره خواهند گرفت. حتی در مواردی می توان از بیسیم برای تجهیزات ثابت استفاده کرد. مثلاً، اگر کابل کشی به یک ساختمان (بعلت وجود موانع طبیعی مانند کوه، جنگل و باتلاق) مشکل باشد، استفاده از بیسیم ترجیح دارد. همانطور که قبلاً هم گفتیم، مخابرات بیسیم دیجیتال برای اولین بار در مجمع الجزایر هاوایی بکار گرفته شد.

۱-۳-۲ طیف الکترومغناطیس

امواج الکترومغناطیس حاصل حرکت الکترونها هستند، که می توانند در فضا (حتی در خلأ) منتشر شوند. وجود چنین امواجی اولین بار بصورت تئوری در سال ۱۸۶۵ توسط فیزیکدان انگلیسی جیمز کلرک ماکسول پیش بینی شد، و در سال ۱۸۸۷ فیزیکدان آلمانی هاینریش هرتز موفق شد آنها را مشاهده کند. تعداد نوسانهای یک موج در ثانیه فرکانس، f ، نامیده می شود، و واحد آن (به افتخار هاینریش هرتز) هرتز (Hz) نامگذاری شده است. فاصله بین دو قله (یا حضیض) متوالی موج را نیز طول موج می گویند، و آنرا با حرف یونانی λ نشان می دهند.

اگر آنتنی با اندازه مناسب به یک مدار الکتریکی وصل شود، می تواند امواج الکترومغناطیسی منتشر کند، که این امواج را می توان در فواصل مناسب دریافت کرد. تمام سیستمهای مخابرات بیسیم بر این اساس کار می کنند.

امواج الکترومغناطیسی (صرفنظر از فرکانس آنها) در خلأ با سرعت ثابتی (سرعت نور) حرکت می کنند، که مقدار تقریبی آن 3×10^8 m/sec است، و با c نشان داده می شود (سرعت نور حد در طبیعت است، یعنی

هیچ چیز نمی تواند سریعتر از آن حرکت کند.) سرعت این امواج در سیم مسی یا فیبر نوری به حدود $\frac{2}{3}c$ کاهش می یابد، که تا حدی نیز به فرکانس وابسته است.

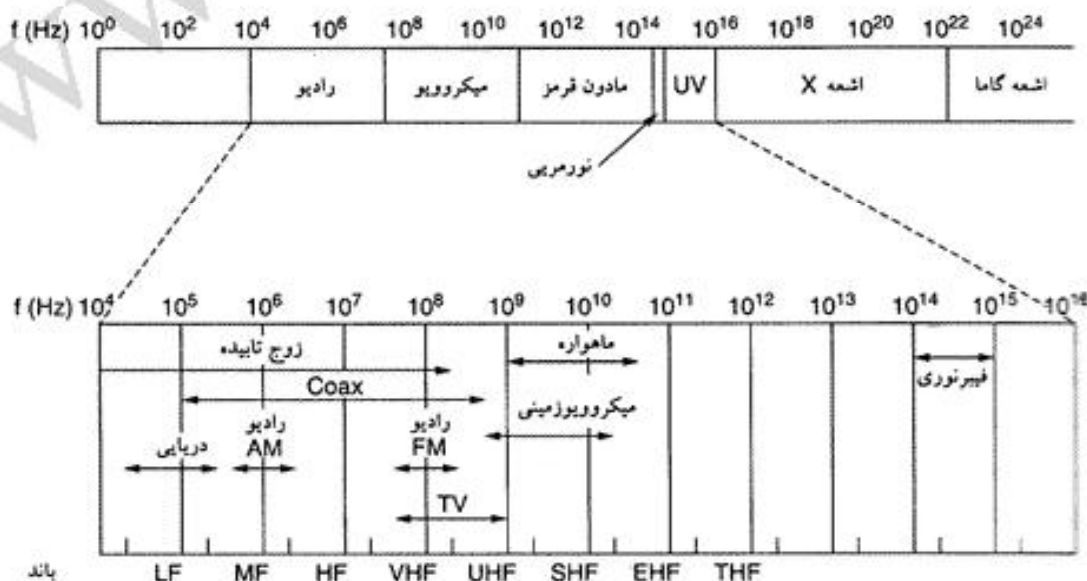
رابطه اساسی بین f ، λ و c (در خلأ) چنین است:

$$\lambda f = c \quad (2-2)$$

از آنجائیکه c ثابت است، با داشتن f می توان λ را محاسبه کرد، و بالعکس. اگر λ را برحسب متر و f را برحسب MHz داشته باشیم، رابطه بالا بصورت $\lambda f \approx 300$ در می آید. برای مثال، در فرکانس 100-MHz طول موج 3 m است، که در فرکانس 1000-MHz طول موج آن به 0.3 m کاهش می یابد؛ موجی با طول موج 0.1 m فرکانسی معادل 3000-MHz (در خلأ) دارد.

در شکل ۱۱-۲ طیف الکترومغناطیس را ملاحظه می کنید. از ناحیه های رادیویی، مایکروویو، مادون قرمز، و مرئی این طیف می توان (با مدولاسیون دامنه، فرکانس، و فاز) برای انتقال اطلاعات استفاده کرد. نور ماوراء بنفش، اشعه X و اشعه گاما بدلیل فرکانس بالاتر برای منظور بهتر هستند، ولی از آنجائیکه تولید و مدولاسیون آنها مشکل است، از اجسام غیرشفاف عبور نمی کنند، و برای سلامتی موجودات زنده خطرناکند، کاربرد مخابراتی ندارند. باندهایی که در پائین شکل ۱۱-۲ می بینید، رسماً توسط ITU (برحسب طول موج) تقسیم بندی و نامگذاری شده اند. اصطلاحات LF، MF و HF بترتیب معادل فرکانس پائین، فرکانس متوسط و فرکانس بالا هستند. پیداست وقتی این نامگذاریها انجام می شد، کسی تصور فرکانسهای بالاتر از 10 MHz را هم نمی کرد، و به همین دلیل امروزه برای باندهای بالاتر از آن از نامهای VHF (فرکانس خیلی بالا)، UHF (فرکانس بسیار بالا)، SHF (فرکانس فوق العاده بالا)، EHF (فرکانس شدیداً بالا)، و THF (فرکانس خارق العاده بالا) استفاده می شود - خوشبختانه برای فرکانسهای بالاتر هنوز نامی انتخاب نشده، و شما هم می توانید ذوق خود را در این زمینه امتحان کنید.

مقدار اطلاعاتی که یک موج الکترومغناطیس می تواند حمل کند، به پهنای باند آن بستگی دارد. با تکنولوژی امروزی، می توان در فرکانسهای پائین بازای هر هرتز از پهنای باند ۲ تا ۳ بیت، و در فرکانسهای بالا تا ۸ بیت



شکل ۱۱-۲. طیف الکترومغناطیس و کاربردهای مخابراتی آن.

اطلاعات منتقل کرد، بنابراین یک کابل کواکسیال با پهنای باند 750 MHz می تواند در هر ثانیه تا چندین گیگابایت اطلاعات منتقل کند. با یک نگاه به شکل ۲-۱۱ می توان براحتی دریافت که چرا فیبر نوری این همه طرفدار دارد.

اگر معادله (2-2) را برای f حل کرده و سپس از آن نسبت به λ دیفرانسیل بگیریم، خواهیم داشت

$$\frac{df}{d\lambda} = -\frac{c}{\lambda^2}$$

حال اگر بجای دیفرانسیل تفاضل محدود و قدر مطلق مقادیر را در نظر بگیریم، داریم

$$\Delta f = \frac{c\Delta\lambda}{\lambda^2}$$

بنابراین با داشتن پهنای یک باند، $\Delta\lambda$ ، می توان فرکانس متناظر با آن باند، Δf ، و از آنجا نرخ داده آن، را محاسبه کرد. هر چه پهنای یک باند بزرگتر باشد، نرخ انتقال داده آن باند بیشتر است. برای مثال، باند $1.30 \mu\text{m}$ را در شکل ۲-۶ در نظر بگیرید. در اینجا، $\lambda = 1.3 \times 10^{-6} \text{ m}$ و $\Delta\lambda = 0.17 \times 10^{-6} \text{ m}$ است، بنابراین Δf تقریباً 30 THz خواهد شد. با چنین پهنای باندی (و با فرض 8 bits/Hz) می توان به نرخ انتقال داده 240 Tbps دست یافت. در اکثر سیستمهای انتقال از باندهای فرکانس باریک استفاده می شود (یعنی، $1 \ll \frac{\Delta f}{f}$)، تا گیرنده بتواند بیشترین توان را دریافت کند. اما گاهی نیز استفاده از باند فرکانسی وسیع ضروریست، که پر دو نوع است. در نوع طیف گسترده با پرش فرکانسی (frequency hopping spread spectrum)، فرستنده در هر ثانیه صدها بار فرکانس خود را عوض می کند (بعبارت دیگر از فرکانسی به فرکانس دیگر می پرد). این تکنیک بویژه در مخابرات نظامی کاربرد دارد، چون تشخیص فرکانس فرستنده بسیار مشکل است و پارازیت انداختن روی آن نیز تقریباً غیر ممکن است. در این روش تضعیف موج در اثر انعکاس نیز وجود ندارد، چون وقتی موج انعکاسی (کمی بعد از موج اصلی) به گیرنده می رسد، فرکانس آن تغییر کرده است و فرکانسهای قبلی را دیگر قبول نمی کند. در سالهای اخیر این تکنیک در محصولات تجاری هم کاربرد پیدا کرده است - برای مثال، بلوتوث و 802.11 هر دو از این تکنیک استفاده می کنند.

اختراع این تکنیک نیز داستانی جالب دارد، که شنیدن آن خالی از لطف نیست. یکی از دو مخترع تکنیک طیف گسترده با پرش فرکانسی، خانم هدی لامار هنرپیشه اتریشی الاصل است. شوهر اول این خانم، که سازنده تسلیحات نظامی بود، برای وی توضیح داد که چگونه می توان با استفاده از سیگنالهای رادیویی از درها و موشک ها را کنترل کرد. وقتی خانم لامار فهمید که شوهرش به هیتلر اسلحه می فروشد، دچار وحشت شد، با لباس مبدل گریخت، و به هالیوود رفت تا به حرفه مورد علاقه اش یعنی هنرپیشگی ادامه دهد. لامار به کمک دوستش جورج آنتیل (که یک آهنگساز بود) و برای کمک به متفقین، تکنیک پرش فرکانسی را اختراع کرد. طرح اولیه آنها دارای ۸۸ فرکانس (به تعداد کلیدهای پیانو) بود، که به شماره 2,292,387 در اداره ثبت اختراعات ایالات متحده آمریکا به ثبت رسید. با این حال، آنها نتوانستند مفید و عملی بودن این اختراع را به نیروی دریایی آمریکا بقبولانند، و هرگز پولی بابت آن دریافت نکردند. تنها سالها پس از سپری شدن از حق الاختراع آن بود که این تکنیک مورد توجه محافل علمی قرار گرفت.

نوع دوم باندهای فرکانسی وسیع، طیف گسترده با توالی مستقیم (direct sequence spread spectrum) نام دارد، که در آن سیگنال روی طیف وسیعی از فرکانسها پخش می شود. این تکنیک نیز امروزه، بویژه در تلفنهای همراه نسل دوم، کاربردهای تجاری پیدا کرده است، و با آمدن نسل سوم تلفنهای همراه تسلط آن بر بازار کامل خواهد شد، زیرا دارای کارایی طیفی خوب، مصونیت در برابر نویز عالی و بسیاری ویژگیهای دیگر است. در برخی از شبکه های محلی بیسیم نیز از تکنیک طیف گسترده با توالی مستقیم استفاده شده است. در قسمتهای آینده همین

فصل باز هم به مبحث طیف گسترده برمی گردیم؛ اگر به تکنیکهای مخابرات طیف گسترده و تاریخچه آن علاقمند هستید، کتاب (Scholtz, 1982) را جالب خواهید یافت.

فعلاً فرض را بر این می گذاریم که همه جا از باند فرکانسی باریک استفاده می شود، و بحث خود را با کاربرد بخشهای مختلف طیف الکترومغناطیس شکل ۲-۱۱ ادامه می دهیم - از رادیو شروع می کنیم.

۲-۳-۲ مخابرات رادیویی

امواج رادیویی کاربرد گسترده ای در مخابرات (در فضاها یا سربو شده یا باز) دارند، چون باسانی می توان آنها را تولید کرد، بُرد زیادی دارند، و از ساختمانها و موانع عبور می کنند. امواج رادیویی همه -طرفه هستند، یعنی در تمام جهات منتشر می شوند، بنابراین نیازی به تنظیم دقیق موقعیت گیرنده و فرستنده نیست.

البته همه -طرفه بودن امواج رادیویی همیشه هم خوب نیست. در دهه ۱۹۷۰، شرکت جنرال موتورز تصمیم گرفت اتومبیلهای کادیلاک جدید خود را به ترمز ضد-قفل کامپیوتری (ABS) مجهز کند. در این سیستم وقتی راننده ترمز می گیرد، کامپیوتر مرکزی پالسهای به ترمزها می فرستد، که آنها را چندین بار در ثانیه باز و بسته می کند، و ترمزها دیگر قفل نمی کنند. مدتی بعد در یک روز آفتابی، یکی از پلیسهای گشت بزرگراه ایالت اوهایو تصمیم گرفت رادیوی بیسیم جدید خود را امتحان کرده و با مرکز فرماندهی تماس بگیرد. با این کار، اتومبیل کادیلاکی که در نزدیکی وی حرکت می کرد، بلافاصله مثل یک اسب چموش شروع به حرکات عجیبی کرد. وقتی افسر پلیس اتومبیل خاطی را متوقف کرد، راننده ادعا کرد که هیچ کاری انجام نداده و اتومبیل بیکپاره دیوانه شده است.

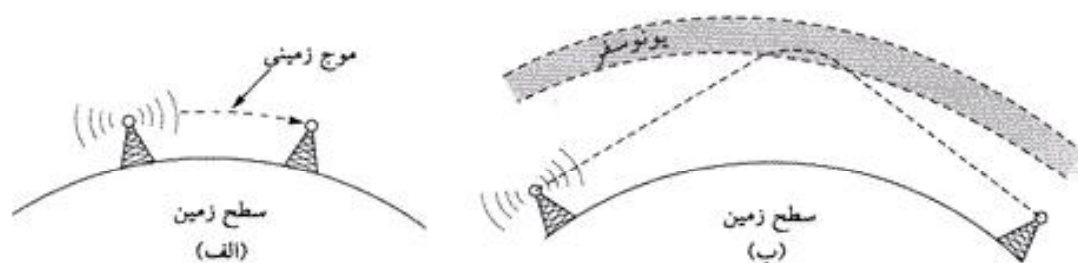
کمی بعد ماجرا روشتتر شد: اتومبیلهای کادیلاک همه جا خوب کار می کردند، و فقط در بزرگراههای ایالت اوهایو، آن هم وقتی پلیس می دیدند، دیوانه می شدند. تا مدتها جنرال موتورز سرگردان بود و نمی توانست بفهمد چرا کادیلاک در همه جا جز بزرگراههای ایالت اوهایو خوب کار می کند (کادیلاک حتی در خیابانها و جاده های معمولی اوهایو هم خوب کار می کرد). فقط بعد از تحقیقات گسترده بود که معلوم شد، مدارهای سیمکشی کادیلاک برای فرکانس سیستم رادیویی جدید پلیس بزرگراه اوهایو تبدیل به یک آنتن خوب می شود، و پالسهای این رادیو سیستم ABS را فعال می کند.

ویژگیهای امواج رادیویی به فرکانس آنها وابسته است. در فرکانسهای پائین، امواج براحتی از موانع عبور می کنند، ولی توان آنها بر حسب فاصله سرعت افت می کند (با ضریب $1/r^2$). در فرکانسهای بالا، امواج رادیویی به خط مستقیم حرکت می کنند، و در برخورد با موانع منعکس می شوند. حتی قطرات باران هم امواج با فرکانس بالا را جذب می کند. امواج رادیویی در تمام فرکانسها در معرض تداخل ناشی از کارکرد وسایل الکتریکی (مانند موتور) هستند.

بدلیل بُرد زیاد امواج رادیویی، خطر تداخل آنها در سرتاسر دنیا وجود دارد، به همین دلیل دولتها نحوه استفاده از فرستنده های رادیویی را (البته با یک استثنا) پشت کنترل می کنند.

امواج رادیویی در باندهای VLF، LF و MF از انحنا ی زمین تبعیت می کنند - به شکل ۲-۱۲ (الف) نگاه کنید. این امواج نزدیک به ۱۰۰۰ کیلومتر بُرد دارند. ایستگاههای رادیویی AM در باند MF کار می کنند، و به همین دلیل بُرد آنها چنین زیاد است. امواج رادیویی در این باند براحتی از موانع و ساختمانها عبور می کنند، و علت دریافت آنها در داخل ساختمانها نیز همین است. تنها مشکل این امواج پهنای باند کم آنهاست (معادله ۲-۳ را ببینید).

در باندهای HF و VHF، آن قسمت از امواج رادیویی که نزدیک سطح زمین حرکت می کنند، جذب زمین می شوند. ولی بخشی از امواج که به سمت فضا می روند، پس از برخورد با یونوسفر (ionosphere - لایه ای از جو زمین به ارتفاع ۱۰۰ تا ۵۰۰ کیلومتر، که حاوی ذرات باردار است) به طرف زمین برمی گردند؛ شکل ۲-۱۲ (ب) را



شکل ۲-۱۲. (الف) در باندهای VLF، LF و MF امواج رادیویی از انحناى زمین تبعیت می کنند. (ب) در باندهای HF و VHF آنها بین زمین و یونوسفر رفت و برگشت می کنند.

نگاه کنید. در شرایط خاصی این امواج می توانند چندین بار بین زمین و یونوسفر رفت و برگشت کرده، و صدها کیلومتر دورتر دریافت شوند. طرفداران رادیو آماتوری و مخابرات نظامی از باندهای HF و VHF استفاده می کنند.

۳-۳-۲ مخابرات مایکروویو

در فرکانسهای بالای 100 MHz امواج تقریباً به خط مستقیم حرکت می کنند، و می توان آنها را دقیقاً روی یک نقطه متمرکز کرد. متمرکز کردن تمام انرژی موج در یک پرتو باریک (با استفاده از آنتنهای بشقابی) باعث بالا رفتن نسبت سیگنال به نویز می شود، ولی از طرف دیگر لازم است تا گیرنده و فرستنده بدقت تنظیم شوند. علاوه بر آن، باریک بودن پرتوها باعث می شود تا تداخل فرستنده ها به حداقل برسد، مشروط بر اینکه کمی با هم فاصله داشته باشند. تا قبل از اختراع فیبر نوری، این امواج (که به مایکروویو معروفند) ستون فقرات مخابرات راه دور محسوب می شدند. در واقع، یکی از شرکت های رقیب AT&T بنام MCI شبکه گسترده ای از برج های مایکروویو با فواصل ده ها کیلومتر ایجاد کرده بود (حتی نام این شرکت - Microwave Communications, Inc. - هم بنوعی گویای استفاده از امواج مایکروویو است). MCI مدتهاست به فیبر نوری روی آورده، و اکنون در WorldCom ادغام شده است.

از آنجائیکه امواج مایکروویو به خط مستقیم حرکت می کنند، اگر فاصله ایستگاه مبدأ و مقصد زیاد باشد، انحناى زمین مانع از رسیدن امواج خواهد شد. به همین دلیل لازم است تا در فاصله بین آنها از برج های تکرارکننده استفاده شود. هر چه ارتفاع این برجها بیشتر باشد، فاصله آنها می تواند زیادتر باشد. فاصله دو ایستگاه تقریباً با توان دوم ارتفاع آنها رابطه دارد؛ برای مثال، دو برج ۱۰۰ متری می توانند چیزی در حدود ۸۰ کیلومتر فاصله داشته باشند. امواج مایکروویو، برخلاف امواج رادیویی فرکانس پائین، نمی توانند بخوبی از موانع عبور کنند. علاوه بر این، با وجود متمرکز شدن موج در فرستنده، امواج مایکروویو در طول مسیر دچار پراکندگی جزیی می شوند. قسمتی از موج مایکروویو نیز که توسط لایه های پائین جو منعکس می شود، فاصله بیشتری را طی کرده و هنگام رسیدن به گیرنده دیگر با موج اصلی هم فاز نیست، و باعث خستگی شدن آن می شود. این پدیده که به محوشدگی چندمسیره (multipath fading) معروف است، یکی از مشکلات جدی در مخابرات رادیویی محسوب می شود، و به وضع هوا و فرکانس موج بستگی دارد. در برخی از موارد، ۱۰ درصد ظرفیت کانال برای مقابله با این وضعیت کنار گذاشته می شود، تا در صورت بروز محوشدگی چندمسیره بتواند موقتاً از فرکانسهای جایگزین استفاده کند. تقاضا برای پهنای باند بیشتر باعث شده تا فرکانسها هر روز بالا و بالاتر برود. امروزه باندهای 10 GHz نیز در حال کار هستند، ولی از فرکانس 4 GHz به بالا مشکل جدیدی خودنمایی می کند: جذب شدن انرژی امواج توسط

آب. طول موج این قبیل امواج فقط چند سانتی متر است، و براحتی جذب قطرات باران می شوند. این پدیده شاید برای کسانی که می خواهند یک اجاق مایکروویو بزرگ بسازند و پرندگان را در هوا کباب کنند خوب باشد، ولی برای مخابرات مایکروویو یک مشکل جدی است. تنها راه حل این مشکل (مانند محوشدگی چندمسیره) قطع کردن کانالهای باران زده، و استفاده از کانالهای جایگزین است.

خلاصه اینکه، استفاده از امواج مایکروویو در مخابرات راه دور، تلفنهای همراه، و تلویزیون چنان گسترش یافته، که امروزه دیگر در طیف مایکروویو جایی برای استفاده بیشتر نمانده است. مخابرات مایکروویو مزایای زیادی نسبت به فیبر نوری دارد. اول اینکه نیازی به تصرف زمین برای حفر کانال و کشیدن کابل نیست، و با خرید چند تکه زمین کوچک در فواصل ۵۰ کیلومتری و نصب چند برج مایکروویو، می توان یک سیستم مخابراتی مستقل ایجاد کرد. بهمین خاطر بود که MCI توانست سرعت بعنوان یکی از غولهای صنعت تلفن راه دور قد علم کند. (یکی دیگر از شرکتهای بزرگ مخابراتی یعنی Sprint، مسیر کاملاً متفاوتی در پیش گرفت: این شرکت که متعلق به راه آهن پاسیفیک جنوبی بود، از زمینهای حاشیه خطوط آهن استفاده کرده، و کابلهای خود را در آنجا دفن کرد.)

دیگر اینکه، مایکروویو نسبتاً ارزان است. نصب چند برج ساده برای نصب آنتنها (که می تواند فقط یک دکل ساده یا چهار سیم مهارکننده باشد)، بسیار ارزانتر از خرید زمین در مناطق شلوغ شهری یا صعب العبور کوهستانی (و یا حتی اجاره خطوط تلفنی از شرکتهای تلفن) است.

طیف الکترومغناطیس و سیاست

برای جلوگیری از هرج و مرج و آشوب، توافقهایی در سطح ملی و بین المللی برای نحوه استفاده از فرکانسها شکل گرفته است. از آنجائیکه هر کس پهنای باند بیشتری می خواهد، دولتها مجبور به دخالت هستند، و پهنای باند لازم برای رادیوی AM و FM، تلویزیون، تلفنهای همراه، شرکتهای تلفن، پلیس، ناوبری هوایی و دریایی، مصارف نظامی و دولتی (و خلاصه، همه آنها) که فرکانس می خواهند را به آنها اختصاص می دهند. در سطح بین المللی نیز یکی از شعبات ITU-R (بنام WARC) به هماهنگی این اقدامات اختصاص یافته، تا کارکرد وسایل مخابراتی را در کشورهای مختلف تضمین کند. با این همه، توصیه های ITU-R الزام آور نیست، و گاهی پیش می آید که FCC (Federal Communication Commission) - سازمانی که مسئول تخصیص باند های فرکانسی در ایالات متحده آمریکا است) توصیه های آنرا نادیده بگیرد، تا (مثلاً) یک طیف فرکانسی را به یک گروه قدرتمند سیاسی اختصاص دهد.

علاوه بر اختصاص قسمتی از طیف فرکانسی به یک کاربرد خاص (مثلاً، تلفن همراه)، باید مشخص شود که کدام حامل (carrier) ها مجاز به استفاده از این فرکانسها هستند. در گذشته، برای این منظور سه روش کاربرد گسترده تری داشت. در قدیمی ترین آنها، که به مسابقه زیبایی معروف بود، هر حامل بایستی توضیح می داد که چرا پیشنهاد آنها بیشترین نفع را برای جامعه دارد - و این مقامات دولتی بودند که بهترین پیشنهاد را انتخاب می کردند. از آنجائیکه انتخاب یک حامل منافع میلیاردری برای آن در بر داشت، در این روش رشوه، فساد و پارتی بازی پیدای می کرد. علاوه بر آن، حتی وقتی یک کارمند وظیفه شناس تشخیص می داد که پیشنهاد یک شرکت خارجی بهتر از شرکتهای داخلی است، قبولاندن آن به مقامات بالاتر کار ساده ای نبود.

مشکلات روش انتخاب دولتی، به روش دوم منجر شد: قرعه کشی بین حامل های متقاضی یک طیف فرکانسی. مشکل این روش آن بود که حتی شرکتهایی که هیچ نفعی در استفاده از طیف نداشتند، می توانستند در قرعه کشی شرکت کنند. بدین ترتیب حتی صاحب یک رستوران یا کفش فروشی هم می توانست برنده قرعه کشی شده، و بعد با خیال راحت و بدون هیچ خطری امتیاز خود را به شرکتهای ذینفع بفروشد و سود کلانی به جیب بزند.

بخشیدن چنین ثروتهای بادآورده ای به افراد زرنگ و هوشیار، باعث انتقادات زیادی شد، و منجر به اتخاذ روش سوم اعطای امتیاز حامل طیفهای فرکانسی گردید: حراج کردن پهنای باند. وقتی انگلستان در سال ۲۰۰۰ طیف فرکانسی تلفنهای همراه نسل سوم را به حراج گذاشت، پیش بینی می کرد که از این راه چهار میلیارد دلار عایدی داشته باشد. ولی در واقع از این مزایده چهل میلیارد دلار سود برد، چون شرکتهای زیادی (از ترس اینکه از قافله تلفن همراه عقب بمانند) به تکاپوی خرید طیف فرکانسی افتادند. این حادثه طمع سایر دولتها را تحریک کرد، و آنها هم به فکر راه انداختن مزایده فروش طیف فرکانسی افتادند. شرکتهای بسیاری به همین دلیل تا مرز ورشکستگی پیش رفتند، و آنهایی که ماندند باید سالها تلاش کنند تا پول از دست رفته را جبران کنند.

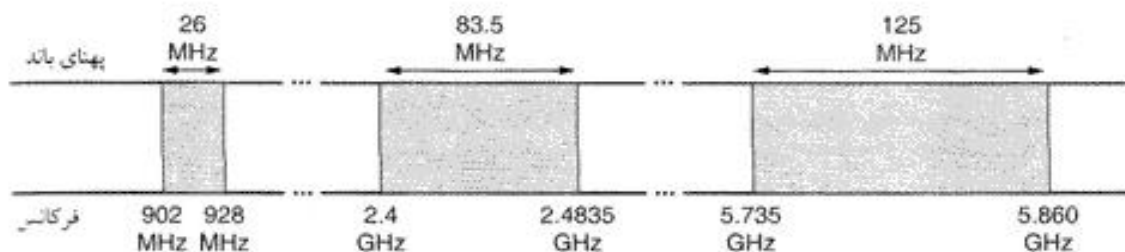
روش دیگر تخصیص فرکانس این است که اصلاً آنرا به کسی اختصاص ندهند: اجازه دهید هر کس با هر فرکانسی که می خواهد کار کند، فقط توان فرستنده های آنها را بگونه ای کنترل کنید که بُرد کمی داشته باشند، و با فرستنده های دیگر تداخل نکنند. بسیاری از دولتها بخشی از طیف فرکانسی خود را (که به باند ISM : صنعتی، علمی، پزشکی معروف است) با همین نیت آزاد می گذارند تا هر کس که مایل است از آنها استفاده کند. در باز کن های برقی، تلفنهای بیسیم خانگی، اسباب بازی های کنترل از راه دور، و بسیاری از وسایل خانگی مجهز به کنترل رادیویی از باندهای ISM استفاده می کنند. برای به حداقل رساندن تداخل بین این دستگاهها، FCC و سایر دولتها سازندگان این قبیل وسایل را مجبور می کنند تا از تکنیکهای طیف گسترده استفاده کنند.

محل طیف ISM از کشوری به کشور دیگر متفاوت است. برای مثال، در ایالات متحده آمریکا وسایلی با توان تشعشعی کمتر از ۱ وات می توانند (بدون نیاز به اخذ مجوز FCC) از باندهای مشخص شده در شکل ۲-۱۳ استفاده کنند. باند 900-MHz بهترین کارایی را دارد، ولی علاوه بر اینکه خیلی شلوغ است، در تمام دنیا هم آزاد نیست. باند 2.4-GHz در اغلب کشورهای دنیا آزاد است، ولی در معرض تداخل امواج اجاقهای مایکروویو و تأسیسات رادار قرار دارد. بلوتوث و برخی از شبکه های محلی 802.11 در این باند کار می کنند. باند 5.7-GHz نسبتاً جدید است و وسایل آن هنوز گران هستند، ولی از آنجائیکه استاندارد 802.11a در این باند کار می کند، بزودی شاهد رواج آن خواهیم بود.

۲-۴. امواج مادون قرمز و میلیمتری

امواج هدایت نشده مادون قرمز و میلیمتری کاربرد زیادی در مخابرات بُرد کوتاه دارند. دستگاههای کنترل از راه دور وسایل صوتی و تصویری همگی از امواج مادون قرمز استفاده می کنند. این دستگاهها جهت دار، ارزان و ساده هستند، ولی یک عیب عمده دارند: امواج مادون قرمز از اجسام صلب عبور نمی کند (بین دستگاه کنترل از راه دور و تلویزیون خود بایستید، و ببینید هنوز کار می کند یا نه). در حالت کلی، هر چه به سمت فرکانسهای طیف مرئی نور برویم، امواج بیشتر شبیه نور (و کمتر شبیه امواج رادیویی) عمل می کنند.

از طرف دیگر، عبور نکردن امواج مادون قرمز از موانع سخت یک مزیت محسوب می شود: شما که نمی خواهید تلویزیونتان با کنترل از راه دور همسایه روشن و خاموش شود. استراق سمع امواج مادون قرمز نیز



شکل ۲-۱۳. باندهای ISM در ایالات متحده آمریکا.

مشکلاتر از امواج رادیویی است، و بهمین دلیل از ایمنی بالاتری برخوردار است. بر خلاف امواج رادیویی، استفاده از امواج مادون قرمز هیچ نیازی به کسب مجوز از مقامات رسمی ندارد. با وجود برخی کاربردهای امواج مادون قرمز در صنعت کامپیوتر (مانند اتصال ماوس و چاپگر)، این امواج در صحنه مخابرات حرفی برای گفتن ندارند.

۵-۳-۲ مخابرات امواج نوری

ارسال علائم نوری قرنهایست که شناخته شده و بکار برده می شود. امروزه برای متصل کردن دو شبکه که در ساختمانهای جداگانه قرار دارند، از لیزرهایی که روی پشت بام ساختمانها قرار می گیرند، استفاده می شود. پرتوهای لیزر اساساً یکطرفه هستند، بنابراین هر ساختمان باید فرستنده و گیرنده لیزری جداگانه داشته باشد. پهنای باند این روش بسیار بالا و هزینه آن نیز بسیار پائین است، و علاوه بر اینکه نصب ساده ای (نسبت به تجهیزات مایکروویو) دارد، نیازی به مجوز FCC نیز ندارد.

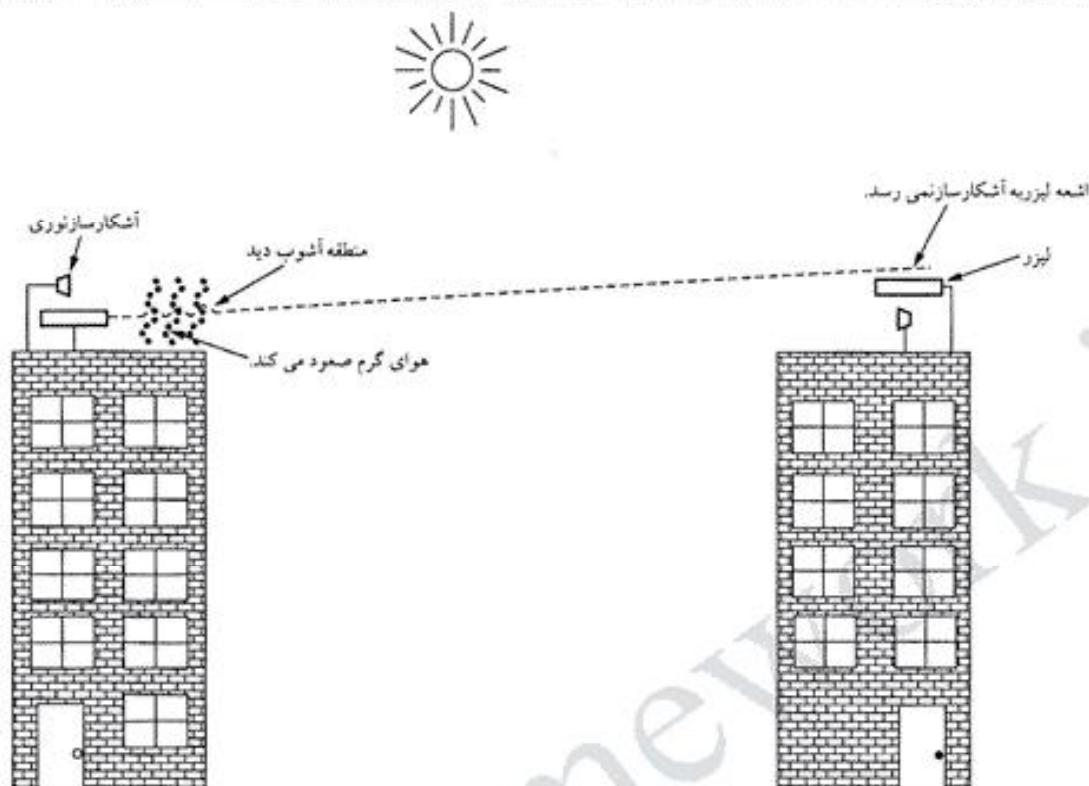
البته مزیت اصلی پرتو لیزر (یعنی باریک بودن آن) در اینجا یک نقطه ضعف محسوب می شود. هدف گیری یک پرتو لیزر به قطر ۱ میلی متر روی هدفی باندازه ته سوزن در فاصله ۵۰۰ متری حتی برای قهرمان تیراندازی المپیک هم دشوار است. در این سیستمها معمولاً از عدسیهای خاصی برای پراکنده کردن نور لیزر و پهن کردن پرتوهای آن استفاده می شود.

یکی دیگر از معایب لیزر اینست که نمی تواند در روزهای بارانی و مه آلود بخوبی عمل کند، ولی برای هوای صاف و آفتابی ایده آل است. تجربه جالبی که مؤلف این کتاب از چنین سیستمی دارد، می تواند آموزنده باشد. چندی قبل در یک کنفرانس علمی که در یکی از هتل های مدرن اروپایی ترتیب یافته بود، شرکت داشتم. مسئولین با درایت کنفرانس تعدادی کامپیوتر در یک اتاق نصب کرده بودند، تا مدعوین بتوانند در حین شرکت در سخنرانی ها ایمیل خود را نیز چک کنند. از آنجائیکه شرکت مخابرات محلی حاضر نشده بود تعداد زیادی خط تلفن را فقط برای سه روز برگزاری کنفرانس در اختیار برگزارکنندگان آن قرار دهد، آنها یک دستگاه لیزر روی پشت بام هتل قرار داده بودند، تا آنجا را به ساختمان دیارتمان کامپیوتر دانشگاه که در چند کیلومتری آن محل قرار داشت، وصل کنند. سیستم شب قبل از افتتاح کنفرانس تست شده بود، و همه چیز مرتب بنظر می رسید. ساعت ۹ صبح روز بعد (که روزی صاف و آفتابی بود) ارتباط بکلی قطع شد، و تمام آن روز نیز وصل نشد. عصر همان روز، برگزارکنندگان کنفرانس دوباره سیستم را تست کردند، و این بار هم همه چیز درست و عالی بود. اما صبح روز بعد (و روز بعد از آن) دوباره همان اتفاق تکرار شد. فقط بعد از پایان کنفرانس بود که علت کشف شد.

با شروع روز، گرمای خورشید باعث گرم شدن بام ساختمان شده، و هوای گرم به بالا صعود می کرد (شکل ۱۴-۲). این جریان هوای آشفته باعث می شد تا پرتو لیزر به مقدار ناچیز منحرف شده، و تمرکز آن روی گیرنده ساختمان هدف بهم بخورد. این دقیقاً همان پدیده ایست که باعث می شود تا ستارگان در شب چشمک بزنند، و یا در روزهای داغ تابستان در جاده ها همه چیز مواج بنظر برسد.

۴ ماهواره های مخابراتی

در سالهای ۱۹۵۰ و اوایل ۱۹۶۰ برخی افراد تلاش کردند تا با استفاده از بالونهای هوای گرم که پوششی فلزی داشتند، نوعی سیستم انعکاس امواج رادیویی بسازند. متأسفانه سیگنال برگشتی چنان ضعیف بود که به هیچ دردی نمی خورد. سپس نیروی دریایی آمریکا متوجه بالونی شد که همیشه در آسمان است: ماه؛ سیستمی که ساخته شد با استفاده از انعکاس امواج از ماه، بین کشتی ها و پایگاههای ساحلی ارتباط برقرار می کرد. پیشرفت بیشتر در مخابرات فضایی فقط زمانی ممکن شد که انسان اولین ماهواره های مخابراتی را به فضا

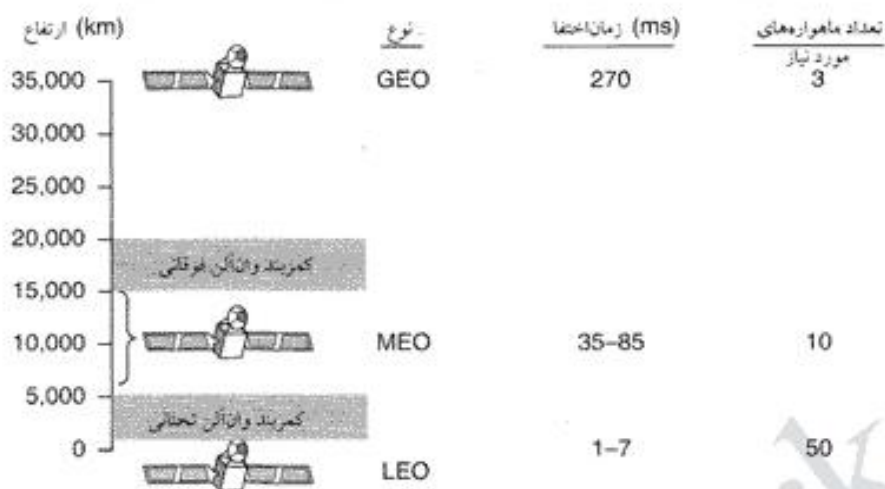


شکل ۲-۱۴. جریانهای همرفتی می تواند باعث انحراف پرتوهای لیزر شود.

پرتاب کرد. تفاوت کلیدی بین ماهواره های مصنوعی و اجسام فضایی این بود که ماهواره های ساخته دست بشر می توانستند قبل از برگشت دادن سیگنال آنها تقویت کنند - بدین ترتیب، یک کنجکاو عجیب تبدیل به یکی از ارکان زندگی بشر شد.

ماهواره های مخابراتی چند ویژگی دارند که آنها را برای کاربردهای مختلف جذاب می کند. یک ماهواره مخابراتی، در ساده ترین شکل خود، یک تکرارکننده مایکروویو بزرگ در آسمان است. چنین ماهواره ای دارای چندین ترانسپاندر (تکرارکننده - transponder) است، که به فرکانسهای خاصی گوش می کنند، سیگنال دریافتی را تقویت کرده، و سپس آنها را با فرکانس متفاوتی برمی گردانند (تا با سیگنال ورودی تداخل نکند). پرتو برگشتی می تواند وسیع باشد و بخش بزرگی از سطح کره زمین را بپوشاند، یا اینکه باریک باشد و فقط منطقه ای به قطر چند صد کیلومتر مربع را پوشش دهد. به این حالت شیپور خمیده (bent pipe) گفته می شود.

طبق قانون کپلر، دوره گردش مداری یک ماهواره با توان $\frac{3}{2}$ شعاع مدار آن متناسب است - هر چه مدار ماهواره بالاتر باشد، دوره گردش آن بیشتر است. در مدارهای نزدیک زمین دوره گردش ماهواره ها معمولاً ۹۰ دقیقه است. و آنها بسرعت از دید ایستگاههای زمینی خارج می شوند. برای پوشش دادن تمام سطح کره زمین با چنین ماهواره هایی، به تعداد زیادی از آنها نیاز است. در مداری به ارتفاع 35,800 km دوره گردش ماهواره ۲۴ ساعت (معادل یک دور گردش زمین به دور خود) است. اگر ارتفاع ماهواره را تا 384,000 km بالا ببریم، دوره گردش آن یک ماه خواهد شد - اگر باور ندارید، به ماه نگاه کنید؛ ارتفاع مداری ماه از سطح زمین دقیقاً همین مقدار است. دوره گردش ماهواره اهمیت زیادی دارد، ولی تنها عاملی نیست که تعیین می کند ماهواره کجا باید قرار گیرد. عامل مهم دیگر، وجود کمربند وان آلن (Van Allen belt) است - کمربند وان آلن لایه ای از ذرات باردار پُرانرژی است که توسط میدان مغناطیسی زمین بدام افتاده اند. هر ماهواره ای که در داخل این لایه پرواز کند.



شکل ۲-۱۵. ماهواره‌های مخابراتی و برخی از ویژگی‌های آنها، از جمله ارتفاع از سطح زمین، زمان تأخیر رفت و برگشت سیگنال، و تعداد ماهواره‌های که برای پوشش کل زمین لازم است.

بسرعت توسط ذرات باردار پُرانرژی آن از بین می‌رود. از ترکیب این دو عامل سه ناحیه مشخص می‌شود، که می‌توان ماهواره‌ها را با اطمینان خاطر در آنها قرار داد (شکل ۲-۱۵ را ببینید). در قسمتهای آینده ماهواره‌هایی را که در هر یک از این ناحیه‌ها قرار داده می‌شوند، تشریح خواهیم کرد.

۱-۴-۲ ماهواره‌های زمین ثابت

در سال ۱۹۴۵، نویسنده داستانهای علمی-تخیلی آرثور سی. کلارک با محاسبات خود نشان داد که اگر ماهواره‌ای در یک مدار استوایی و در ارتفاعی معادل ۳۵,۸۰۰ km قرار گیرد، نسبت به زمین ثابت بنظر خواهد رسید، و نیازی نیست که ایستگاه زمینی آنرا تعقیب کند (Clarke, 1945). وی در این مقاله یک سیستم کامل مخابراتی را با استفاده از این ماهواره‌های زمین ثابت (geostationary satellites) تشریح کرده بود، و از جمله مدار آنها، پانلهای خورشیدی (برای تأمین انرژی مورد نیاز)، فرکانسهای رادیویی، و نحوه پرتاب ماهواره‌ها را بدقت توضیح داده بود. متأسفانه، وی در این مقاله نتیجه گرفته بود که این طرح غیرعملی است، چون قرار دادن تقویت‌کننده‌های لامپ خلأ (که بسیار پرمصرف، حجیم و شکننده بودند) در مدار زمین غیرممکن است، و هرگز این ایده را دنبال نکرد، اگر چه چند داستان علمی-تخیلی در این زمینه نوشت.

اختراع ترانزیستور همه چیز را تغییر داد، و اولین ماهواره مخابراتی بنام تل‌استار (Telstar) در ژوئیه ۱۹۶۲ به فضا پرتاب شد. از آن زمان به بعد، ماهواره‌های مخابراتی صنعتی چندمیلیارد دلاری را بوجود آورده‌اند، که تنها زمینه سودآور تحقیقات فضایی است. به این ماهواره‌های بلندپرواز اغلب ماهواره‌های GEO (Geostationary Earth Orbit) گفته می‌شود.

با تکنولوژی موجود (برای جلوگیری از تداخل امواج) حداقل فاصله دو ماهواره زمین ثابت نمی‌تواند کمتر از ۲ درجه در صفحه استوایی باشد. بدین ترتیب، در هر زمان بیش از $180 = 360 / 2$ نمی‌توانند در مدار باشند. با این حال برای بالا بردن پهنای باند در دسترس، می‌توان در هر ترانسیلندر از فرکانسها و پولاریزاسیونهای مختلف استفاده کرد.

در این مورد هم برای جلوگیری از هرج و مرج در مدار زمین، تخصیص مکانهای مداری توسط ITU انجام می‌شود. پای سیاست به اینجا هم باز شده است: کشورهایی که بزرگتر از عصر حجر فاصله گرفته‌اند، سهم خود

را از مکانهای مداری طلب می کنند، تا بعد بتوانند آنرا به قیمت خوب به متقاضیان اجاره بدهند. کشورهایی هم هستند که ادعا می کنند قلمرو خاک آنها تا کره ماه ادامه دارد، و هیچکس حق ندارد در آسمان بالای سر آنها ماهواره داشته باشد. و برای شلوغی بیشتر اوضاع، فقط مخابراتی ها نیستند که بر سر تصاحب مدارها می جنگند: تلویزیونهای ماهواره ای، دولتها، و نظامیها هم می خواهند سهمی از مدارهای زمین داشته باشند.

ماهواره های امروزی بسیار بزرگ هستند، که وزن آنها گاهی به ۴۰۰۰ کیلوگرم می رسد، و دهها کیلووات انرژی الکتریکی مصرف می کنند (انرژی که توسط پانلهای خورشیدی تولید می شود). مدار و موقعیت این ماهواره ها تحت تأثیر جاذبه خورشید، ماه و سایر سیارات منظومه شمسی پیوسته در حال تغییر است، که این تأثیرات توسط موتورهای موشکی کوچکی که در آنها تعبیه شده، خنثی می شود (به این کار نگهداری ایستگاه - station keeping - می گویند). از آنجائیکه سوخت این موتورها محدود است، بعد از مدتی (که معمولاً حدود ۱۰ سال است) ماهواره بکلی از کنترل خارج و بلااستفاده می شود. ارتفاع چنین ماهواره هایی بتدریج کاهش می یابد، و پس از ورود به جو زمین می سوزند، و یا در برخورد با سطح زمین متلاشی می شوند.

مکانهای مداری تنها چیزی نیست که سر آن دعواست. فرکانسها نیز خواهان زیادی دارد، چون احتمال تداخل فرکانسهای ارسالی از ماهواره (downlink) با فرکانسهای زمینی وجود دارد. برای رفع این مشکل، ITU چند باند فرکانسی را به کاربردهای ماهواره ای اختصاص داده است، که آنها را در شکل ۲-۱۶ ملاحظه می کنید. باند C اولین باندی بود که به کاربردهای تجاری اختصاص یافت. این باند دو محدوده دارد: محدوده پائینی که برای ارسال از ماهواره (downlink) استفاده می شود، و محدوده بالایی که به ارسال به ماهواره (uplink) تخصیص یافته است. این باندها بسیار شلوغ هستند، چون در مخابرات مایکروویو زمینی هم از آنها استفاده می شود. باندهای L و S در سال ۲۰۰۰ طبق توافقات بین المللی اضافه شدند، اما این باندها نیز بسیار باریک و شلوغ هستند.

باند بعدی که در اختیار مخابرات تجاری قرار دارد، باند Ku (K under) است. این باند هنوز پُر نشده، و در فرکانسهای آن می توان ماهواره ها را تا فاصله ۱ درجه مداری به هم نزدیک کرد، ولی یک مشکل عمده دارد: باران. آب جاذب خوبی برای انرژی امواج در این طول موج است. خوشبختانه، طوفانهای بزرگ معمولاً به یک منطقه کوچک محدود هستند، و اگر بجای یک ایستگاه زمینی از چندین ایستگاه با فواصل زیاد استفاده کنیم، این مشکل مرتفع خواهد شد (که البته این راه حل هزینه بسیار بالایی دارد). باند Ka (K above) نیز به مصارف تجاری تخصیص یافته، ولی تجهیزات آن هنوز بسیار گران است. علاوه بر این باندهای تجاری، دهها باند دولتی و نظامی نیز وجود دارد.

یک ماهواره امروزی در حدود ۴۰ ترانسپاندر دارد، که پهنای باند هر کدام از آنها ۸۰-MHz است. معمولاً هر ترانسپاندر بعنوان یک شیپور خمیده عمل می کند، ولی در ماهواره های جدیدتر (که امکانات پردازشی بیشتری دارند) می توان بگونه ای دیگر نیز عمل کرد. در ماهواره های اولیه، تقسیم ترانسپاندرها به کانالهای مختلف استاتیکی بود: پهنای باند بصورت ساده به چند باند فرکانسی ثابت تقسیم می شد. امروزه، پرتو ترانسپاندر به بُرشهای زمانی

مشکلات	پهنای باند	ارسال به ماهواره	دریافت از ماهواره	باند
پهنای باند کم به شلوغ	15 MHz	1.6 GHz	1.5 GHz	L
پهنای باند کم به شلوغ	70 MHz	2.2 GHz	1.9 GHz	S
تداخل زمینی	500 MHz	6.0 GHz	4.0 GHz	C
باران	500 MHz	14 GHz	11 GHz	Ku
باران با قیمت بالای تجهیزات	3500 MHz	30 GHz	20 GHz	Ka

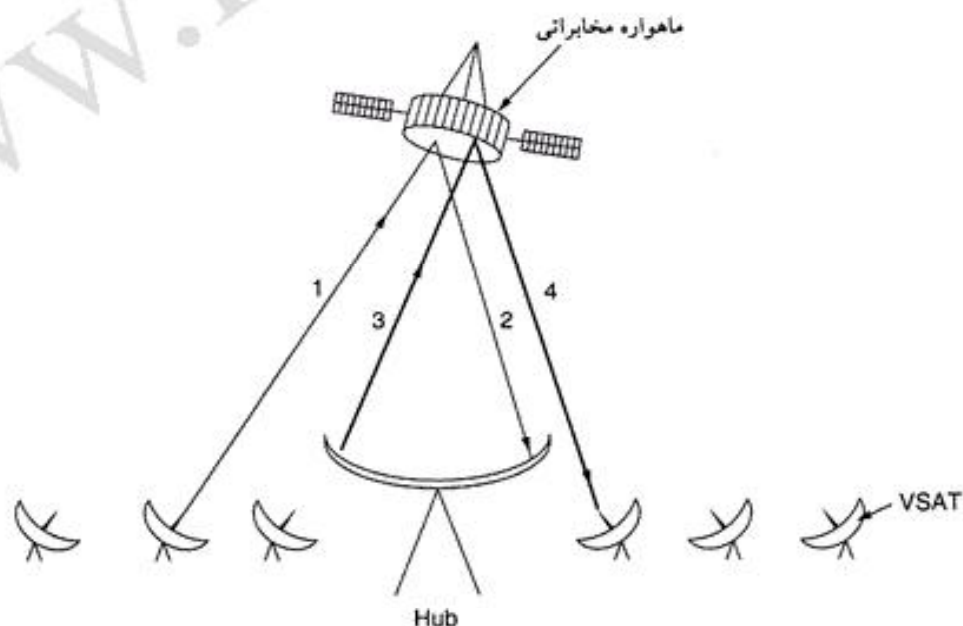
شکل ۲-۱۶. باندهای ماهواره ای.

تقسیم می‌شود، و هر کاربر می‌تواند بنوبت از آن استفاده کند. در قسمتهای آینده همین فصل این دو تکنیک (مالتی پلکس تقسیم فرکانسی و مالتی پلکس تقسیم زمانی) را به تفصیل بررسی خواهیم کرد.

اولین ماهواره‌های زمین ثابت یک پرتو فضایی واحد داشتند، که تقریباً $\frac{1}{3}$ سطح زمین را می‌پوشاند (به این ناحیه جای پای ماهواره - footprint - گفته می‌شود). اما با افت شدید قیمت، اندازه، و توان مصرفی وسایل میکروالکترونیک، استراتژیهای هوشمندانه تری امکان پذیر گردید. هر ماهواره به چندین آنتن و ترانسپاندر مجهز شد، و پرتو ارسالی از ماهواره چنان باریک شد که فقط منطقه کوچکی را در بر می‌گرفت، و بدین ترتیب امکان مخابره همزمان چندین سیگنال بوجود آمد. این پرتوهای نقطه‌ای (spot beam) معمولاً بیضی شکل هستند، و می‌توان آنها را بقدری باریک کرد که فقط ناحیه‌ای به قطر چند صد کیلومتر را پوشش دهند. برای مثال، یک ماهواره مخابراتی که فقط برای ایالات متحده آمریکا به فضا پرتاب شده، را می‌توان طوری تنظیم کرد که یک پرتو برای ۴۸ ایالت خاک اصلی آمریکا، یک پرتو برای آلاسکا، و یکی برای هاوایی داشته باشد.

یکی از پیشرفتهای جدید در دنیای مخابرات ماهواره‌ای، ایستگاههای ارزان قیمتی هستند که VSAT (ترمینال با باریکه بسیار کوچک - Very Small Aperture Terminal) نامیده می‌شوند (Abramson, 2000). این ترمینالهای بسیار کوچک آنتنهایی بقطر ۱ متر یا کمتر دارند (برای مقایسه، قطر آنتنهای استاندارد GEO ۱۰ متر است)، و قدرت تشعشعی آنها در حدود ۱ وات است. سرعت ارسال این سیستمها در حدود 19.2-kbps است، ولی می‌توانند تا 512-kbps یا بیشتر دریافت کنند. تلویزیونهای ارسال مستقیم ماهواره‌ای (DB-SAT) معمولاً از این تکنولوژی استفاده می‌کنند.

در بسیاری از سیستمهای VSAT، ایستگاههای زمینی بدلیل توان پائین نمی‌توانند مستقیماً با هم ارتباط برقرار کنند (البته از طریق ماهواره). در این موارد، از یک ایستگاه زمینی خاص، با آنتنی بزرگ و قوی بعنوان هاب (hub)، برای رله کردن سیگنالها از یک VSAT به VSAT دیگر استفاده می‌کنند (شکل ۲-۱۷). در این حالت، یکی از طرفین باید آنتنی بزرگ با یک تقویت کننده قوی داشته باشد. زمان تأخیر سیگنال در این روش بیشتر از ارتباط



شکل ۲-۱۷. چند سیستم VSAT به همراه یک هاب.

مستقیم است، ولی کاربران معمولاً هزینه کمتر را به کمی تأخیر ترجیح می دهند. سیستمهای VSAT بیشترین کاربرد را در مناطق روستایی دارد. متأسفانه، هنوز نیمی از جمعیت دنیا به تلفن دسترسی ندارند، و کشیدن خطوط تلفن به هزاران دهکده کوچک و دورافتاده از توان اغلب کشورهای جهان سوم خارج است - ولی نصب یک آنتن VSAT یک متری (که با سلولهای خورشیدی کار کند) کار چندان مشکلی نیست. VSAT تکنولوژی است که می تواند همه مردم دنیا را به هم وصل کند.

مخابرات ماهواره ای ویژگیهایی دارد که بشدت با ارتباطات نقطه-به-نقطه زمینی متفاوتند. اولین ویژگی اینست که، با آنکه سیگنالهای ماهواره ای با سرعت نور (نزدیک $300,000 \text{ km/sec}$) حرکت می کنند، (بعلمت فاصله زیاد ماهواره تا سطح زمین) ارتباط با ماهواره های GEO دارای زمان تأخیر سیگنال قابل ملاحظه ای است. بسته به فاصله کاربر از ایستگاه زمینی و زاویه ماهواره نسبت به افق در آن محل، زمان ارتباط نقطه-به-نقطه بین 25° تا 30° میلی ثانیه متغیر است. زمان متوسط این تأخیر 27° میلی ثانیه است (که برای سیستمهای VSAT دارای هاب به دو برابر، یعنی 54° میلی ثانیه، می رسد).

برای مقایسه بد نیست بدانید که، زمان تأخیر انتشار در لینکهای میکروویو زمینی حدود $3 \mu\text{sec/km}$ ، و برای کابلهای کواکسیال و فیبر نوری تقریباً $5 \mu\text{sec/km}$ است (امواج الکترومغناطیس در هوا سریعتر از مواد جامد حرکت می کنند).

ویژگی مهم دیگر ماهواره ها اینست که آنها ذاتاً رسانه های پخش هستند؛ فرستادن پیام برای یک نفر هیچ تفاوتی با هزاران نفر ندارد. در برخی از کاربردها (مانند تبلیغات اینترنتی) این ویژگی بسیار مفید است. با اینکه بوسیله ارتباطات نقطه-به-نقطه هم می توان چنین وضعیتی را شبیه سازی کرد، ولی این کار با استفاده از ماهواره بسیار ارزانتر تمام می شود. از سوی دیگر، از دیدگاه امنیت و حفظ حریم خصوصی افراد، ماهواره یک فاجعه تمام عیار است: هر کسی می تواند پیامهای خصوصی دیگران را بشنود. اینجاست که اهمیت رمزنگاری (encryption) روشن می شود.

در مخابرات ماهواره ای، هزینه انتقال پیام به فاصله فرستنده و گیرنده بستگی ندارد: تماس با آن سوی اقیانوسها از نظر هزینه هیچ فرقی با تماس با خانه روبرویی ندارد. مخابرات ماهواره ای از نظر نرخ خطا بسیار عالیست، و زمان به بهره برداری رسیدن آن نیز بسیار کوتاه است (نکته ای که در مخابرات نظامی بسیار اهمیت دارد).

۲-۴-۲ ماهواره های مدار متوسط

در مداری بسیار پائینتر از مدار زمین ثابت، و بین دو کمربند وان آلن، ماهواره های مدار متوسط که به MEO (Medium-Earth Orbit) معروفند، قرار می گیرند. این ماهواره ها بطور متوسط هر ۶ ساعت یکبار دور زمین می گردند، و بهمین دلیل ایستگاه زمینی باید آنها را تعقیب کند. بعلمت ارتفاع پائین، ماهواره های MEO جای پای کوچکتری نسبت به ماهواره های GEO دارند، ولی توان تشعشعی لازم برای ارسال به آنها نیز بسیار کمتر است. در حال حاضر از این ماهواره ها برای مقاصد مخابراتی استفاده نمی شود، بنابراین ما هم درباره آنها بیش از این صحبت نخواهیم کرد. ماهواره های ۲۴ گانه GPS (سیستم مکان یابی جهانی - Global Positioning System) که در ارتفاع $18,000 \text{ km}$ پرواز می کنند، در این دسته قرار می گیرند.

۳-۴-۲ ماهواره های مدار پائین

اگر باز هم پائینتر بیانیم، به ماهواره های مدار پائین LEO (Low-Earth Orbit) می رسیم. بدلیل سرعت زیاد گردش مداری این ماهواره ها، برای ایجاد سیستمی با پوشش جهانی به تعداد زیادی از آنها نیاز داریم. از طرف دیگر، بدلیل ارتفاع کم ماهواره های LEO، برای ارتباط با آنها به توان کمی نیاز است، و زمان رفت و برگشت

سیگنال نیز بسیار کم (در حدود چند میلی ثانیه) خواهد بود. در این قسمت سه سیستم ماهواره ای LEO را - که دو تای آنها به سرویس صدا و یکی به سرویسهای اینترنت اختصاص دارند - مورد بررسی قرار خواهیم داد.

ایریدیوم

همانطور که گفتیم بعلت سرعت زیاد گردش مداری ماهواره های LEO - و عبور سریع از مقابل آنتنهای زمینی - تا همین اواخر از این ماهواره ها برای مقاصد مخابراتی استفاده نمی شد. این وضعیت در سال ۱۹۹۰ تغییر کرد: در این سال موتورولا درخواستی برای دریافت مجوز پرتاب ۷۷ ماهواره مخابراتی LEO به FCC ارائه کرد - این پروژه ایریدیوم (Iridium) نام داشت (ایریدیوم عنصر ۷۷ام جدول تناوبی است). بعدها این پروژه تغییر کرد، و قرار شد از فقط ۶۶ ماهواره استفاده شود - بالطبع نام پروژه را هم باید به دیسپروسیوم (عنصر ۶۶ام جدول تناوبی) تغییر می دادند، اما نام دیسپروسیوم بیشتر شبیه یک بیماری خطرناک است تا یک سیستم ماهواره ای! ایده اصلی در این سیستم آن است که به محض خارج شدن یک ماهواره از دید آنتن زمینی، ماهواره دیگری بلافاصله جای آنرا می گیرد. این پیشنهاد به یک هیجان عمومی در میان شرکتهای مخابراتی دامن زد، و همه خواهان آن بودند که زنجیره ای از ماهواره های LEO به فضا پرتاب کنند.

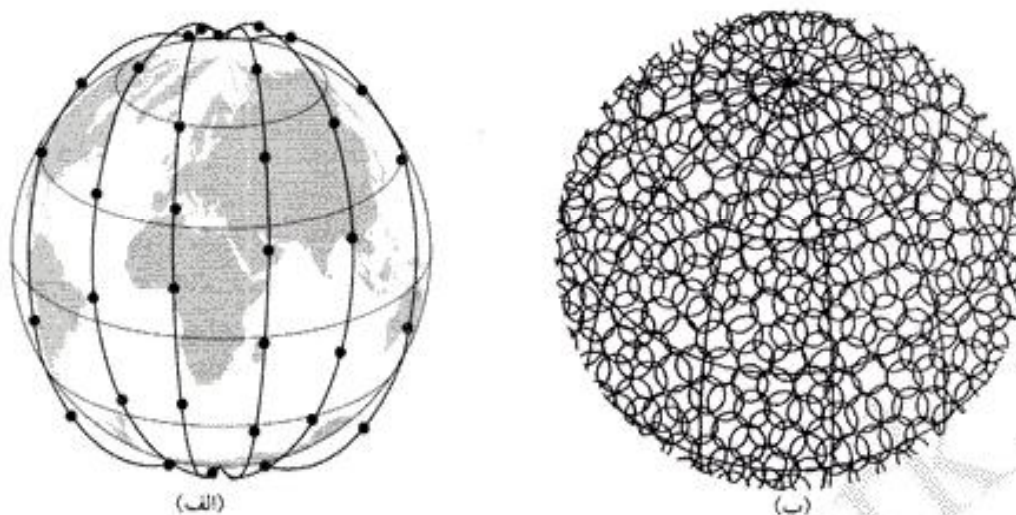
بعد از هفت سال اشتراک مساعی فنی و مالی بین چند شرکت بزرگ، بالاخره ماهواره های ایریدیوم در سال ۱۹۹۷ به فضا پرتاب شدند، و از نوامبر ۱۹۹۸ سرویسهای مخابراتی آنها شروع بکار کرد. اما متأسفانه بدلیل گسترش شبکه تلفنهای همراه، تقاضای ناچیزی برای این سرویسها در بازار وجود داشت، و متعاقب آن در آگوست ۱۹۹۹ پروژه ایریدیوم (طی یکی از فاجعه بارترین ورشکستگی های تاریخ) متوقف شد. ماهواره ها و سایر تجهیزات این پروژه (که ۵ میلیارد دلار ارزش داشت) به قیمت ۲۵ میلیون دلار به یک سرمایه گذار فروخته شد؛ در واقع، پروژه ایریدیوم را باید بزرگترین بازار اسقاطی فضایی بشمار آورد. سرویسهای ایریدیوم از مارس ۲۰۰۱ دوباره راه اندازی شد.

کار اصلی ایریدیوم ارائه سرویسهای مخابراتی از طریق تجهیزاتی که مستقیماً با ماهواره ارتباط برقرار می کنند، بود (و هست). ایریدیوم سرویسهای صدا، داده، فکس، پیجر، و هدایت و ناوبری را در تمام نقاط زمین (خشکی، دریا و هوا) در اختیار کاربران خود می گذارد. کاربران این سرویسها را عمدتاً دریانوردان، هوانوردان، کارکنان سکویهای اکتشاف نفت، و مسافران و جهانگردانی که به مناطق فاقد زیرساختهای مخابراتی (مانند کوه، صحرا، جنگل، و برخی از کشورهای جهان سوم) سفر می کنند، تشکیل می دهند.

ماهواره های ایریدیوم در مدار قطبی و در ارتفاع ۷۵۰ کیلومتری زمین پرواز می کنند. این ماهواره ها بصورت کمربندهایی که از قطبهای شمال - جنوب زمین می گذرند، آرایش یافته اند، و با یکدیگر ۳۲ درجه عرض جغرافیایی فاصله دارند - شکل ۲-۱۸ (الف) را ببینید. برای پوشش دادن تمام سطح زمین شش تا از این حلقه ها کفایت می کند. آنهایی که کمی با شیمی آشنا باشند، می توانند این ماهواره ها را الکترونهای یک اتم غول آسای دیسپروسیوم فرض کنند که بدور هسته اتم (کره زمین) در گردشند.

هر ماهواره حداکثر ۴۸ سلول (پرتو نقطه ای) دارد، که بدین ترتیب تعداد کل سلولها به ۱۶۲۸ می رسد، و می توانند کل سطح زمین را پوشش دهند - شکل ۲-۱۸ (ب) را ببینید. هر ماهواره ایریدیوم ۳۸۴۰ کانال - و کل سیستم ۲۵۳،۴۴۰ کانال - ظرفیت دارد، و می توان از آنها برای سرویسهای مختلف استفاده کرد.

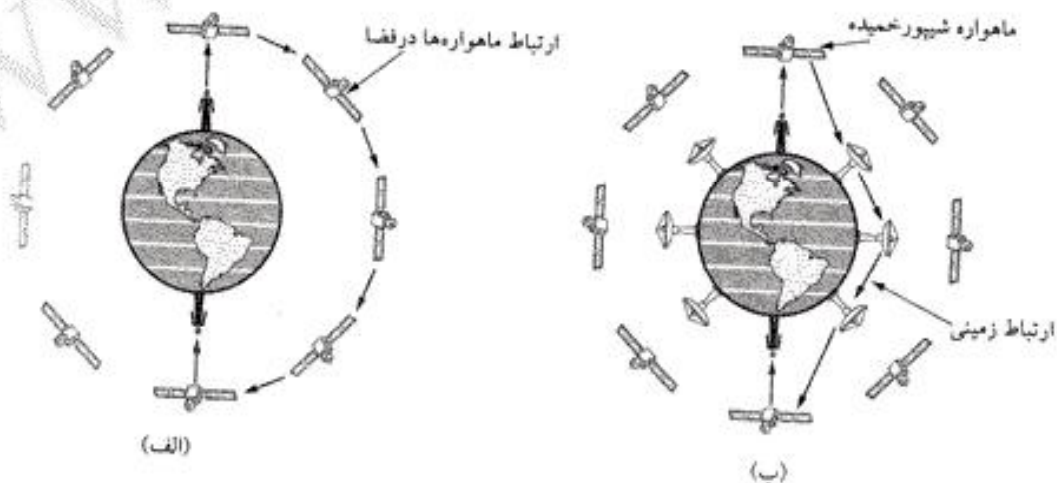
یکی از نکات جالب در مورد ایریدیوم اینست که ارتباط کاربرانی که از یکدیگر فاصله زیادی دارند، در فضا صورت می گیرد (بعبارت دیگر، رله کردن اطلاعات بین ماهواره ها در فضا انجام می شود)؛ شکل ۲-۱۹ (الف) را ببینید. همانطور که در این شکل می بینید، وقتی کاربری در قطب شمال بخواهد با قطب جنوب تماس بگیرد، سیگنال را به ماهواره ای که بالای سرش قرار دارد، می فرستد. سپس این سیگنال از یک ماهواره به ماهواره دیگر رله می شود، تا به قطب جنوب برسد.



شکل ۲-۱۸. (الف) ماهواره های ایریدیوم شش حلقه بدور زمین می سازند. (ب) تعداد سلولهای متحرک ایریدیوم به ۱۶۲۸ می رسد.

گلوبال استار

یکی از طرحهای رقیب ایریدیوم، پروژه گلوبال استار (Globalstar) است. این پروژه از ۴۸ ماهواره LEO تشکیل شده، ولی روش سوئیچینگ آن با ایریدیوم فرق دارد، و در آن از سوئیچینگ زمینی استفاده می شود - شکل ۲-۱۹ (ب) را ببینید. در اینجا سیگنالها از ماهواره مبدأ به یک ایستگاه زمینی بزرگ واقع در سانتا و رک شاپ هدایت شده، و از آنجا به نزدیکترین ایستگاه زمینی نزدیک مقصد فرستاده می شوند، تا بالاخره (بعد از ارسال به ماهواره) به گیرنده برسند. مزیت این روش آنست که (بر خلاف ایریدیوم) قسمت پیچیده کار روی زمین انجام می شود، و مدیریت آن ساده تر است. همچنین، با آنتنهای بزرگ زمینی (که سیگنالهای قویتری می فرستند، و می توانند سیگنالهای ضعیفتری دریافت کنند) از تجهیزات انفرادی ساده تری می توان استفاده کرد (توان تشعشعی تلفتها در حد چند میلی وات است، و حتی بعد از تقویت در ماهواره باز هم چندان قوی نیست).



شکل ۲-۱۹. (الف) رله سیگنال در فضا. (ب) رله سیگنال روی زمین.

تله‌دیزیک

کاربران اصلی ایریدیوم افرادی هستند که به جاهای پرت و دورافتاده می‌روند. نمونه دیگر سیستمهای ماهواره‌ای تله‌دیزیک (Teledesic) است، که کاربران اینترنت پرسرعت را (در تمام دنیا) هدف گرفته است. ایده این سیستم به کریگ مک‌کاو (پیشگام صنعت تلفنهای همراه) و بیل گیتس (بنیانگذار مایکروسافت) تعلق دارد، که از سرعت لاک‌پشتی دسترسی اینترنت ناراضی بودند. هدف سیستم تله‌دیزیک فراهم آوردن دسترسی به اینترنت با سرعتهای بالا (تا 100-Mbps ارسال، و تا 720-Mbps دریافت) برای میلیونها کاربر همزمان است. این سیستم از آنتنهای کوچک و ثابت (شبیه VSAT) استفاده می‌کند، و هیچ نیازی به ارتباطات تلفنی معمولی ندارد.

در طرح اولیه سیستم تله‌دیزیک قرار بود از ۲۸۸ ماهواره پرتو باریک، که در دوازده گروه در ارتفاع ۱۳۵۰ کیلومتری (درست زیر کمربند وان‌آلن تحتانی) پرواز می‌کنند، استفاده شود. ولی این طرح بعدها به ۳۰ ماهواره با جای پای بزرگتر تغییر کرد. ماهواره‌ها در باند نسبتاً خلوت Ka و بصورت سونیچینگ بسته‌ای کار خواهند کرد، و می‌توانند بسته‌ها را بین خود رد و بدل کنند. وقتی یک کاربر به پهنای باند نیاز پیدا می‌کند، درخواست خود را بصورت یک بسته می‌فرستد، و ۵۰ میلی‌ثانیه بعد پهنای باند موردنیاز بصورت خودکار به وی اختصاص داده می‌شود. اگر همه چیز طبق نقشه پیش برود، این سیستم در سال ۲۰۰۵ عملیاتی خواهد شد.

۴-۲-۴ ماهواره یا فیبر؟

مقایسه‌ای بین مخابرات ماهواره‌ای و زمینی آموزنده خواهد بود. تا همین ۲۰ سال پیش هیچکس شک نداشت که آینده مخابرات متعلق به ماهواره‌هاست، چون سیستمهای تلفن در ۱۰۰ سال گذشته تغییر عمده‌ای نکرده بودند، و بنظر نمی‌رسد در ۱۰۰ سال بعدی هم هیچ اتفاق خاصی بیفتد. شرکت‌های تلفن سرویسهای صوتی خوب (با قیمت مناسب) ارائه می‌کردند، و سرمایه‌گذاری آنها سودی تضمین شده داشت. برای آنهایی هم که سرویس داده می‌خواستند، مودمهای 1200-bps موجود بود - و این تمام بضاعت شرکت‌های تلفن بود.

اما در سال ۱۹۸۴ آتش رقابت ایالات متحده آمریکا و اروپا را در بر گرفت، و اوضاع تغییر کرد. شرکت‌های تلفن در سرویسهای راه‌دور فیبرهای نوری را جایگزین کابلهای مسی کردند، و سرویسهای با پهنای باند زیاد مانند ADSL (خط دیجیتال نامتقارن) در اختیار کاربران خود قرار دادند. تغییر اساسی دیگر کاهش نرخهای مخابرات راه‌دور بود، که تا آن زمان بطور مصنوعی و به نفع کاربران محلی بالا نگه داشته شده بود. بنظر می‌رسید فیبر نوری برنده جنگ باشد، ولی ماهواره هم مزایایی دارد که فیبر نمی‌تواند (و احتمالاً نخواهد توانست) با آنها رقابت کند. اول اینکه، پهنای باند فیبر (که شاید یک رشته آن از تمام ماهواره‌های پرتاب شده بیشتر باشد) در اختیار اغلب کاربران نیست. فیبر نوری بیشتر در مخابرات راه‌دور و برای متصل کردن شبکه‌های تلفن بکار می‌رود، تا رساندن پهنای باند بالا به کاربران منفرد. در حالیکه ماهواره‌ها (به کمک یک آنتن ساده در بالای پشت بام) می‌توانند مستقیماً و بدون هیچ واسطه‌ای پهنای باند زیاد را در اختیار تک تک کاربران قرار دهند. سیستم تله‌دیزیک بر اساس همین ایده شکل گرفته است.

مخابرات سیار دیگر قلمرو دست‌نیافتنی ماهواره‌هاست. امروزه بسیاری از افراد مایلند در حال قدم‌زدن، اتومبیل‌سواری، و حتی قایقرانی و پرواز به سرویسهای مخابراتی دسترسی داشته باشند. اینجا دیگر فیبر نوری بکلی بی‌استفاده است، و فقط مخابرات ماهواره‌ای می‌تواند راه چاره باشد. البته برای آنهایی که شعاع حرکتشان محدود است، می‌توان از ترکیب فیبر و بیسیمهای رادیویی استفاده کرد، ولی در هوا و دریا این روش هم دیگر کارایی ندارد.

مزیت دیگر ماهواره در مخابرات پخش (broadcasting) است. وقتی بخواهید یک پیام را در آن واحد به هزاران نفر برسانید (مانند ارسال نرخ کالا و ارز، یا قیمت سهام و اوراق قرضه)، هیچ چیز جای ماهواره را (از نظر

سهولت و هزینه) نمی‌گیرد.

پراکندگی جمعیت و سرزمین از دیگر عواملیست که بکارگیری مخابرات ماهواره‌ای را مقرون بصرفه می‌کند. برای مثال، کشور اندونزی برای هدایت ترافیک تلفن داخلی خود نیز از ماهواره استفاده می‌کند، چون پرتاب یک ماهواره بسیار ساده‌تر است، ناکشیدن کابل‌های زیر دریایی به ۱۳۶۷۷ جزیره‌ای که این کشور را تشکیل می‌دهند. در جاهایی که تملک زمین برای کشیدن کابل زمینی دشوار (و یا پرهزینه) است، نیز ماهواره مزیت نسبی دارد. سرعت نصب و راه‌اندازی سیستم نیز یکی از جاذبه‌های ماهواره بر کابل زمینی پیروز می‌شود. وقتی جنگی در می‌گیرد، و نیروهای نظامی به تماس با نقاط جدید نیاز پیدا می‌کنند، پرتاب یک ماهواره معمولاً سریع‌ترین راه‌حل ممکن است.

بطور خلاصه، بنظر می‌رسد که جریان اصلی مخابرات در آینده بر فیبرهای نوری و تلفنهای همراه متکیست، ولی در برخی از کاربردهای خاص ماهواره‌ها برتری دارند. اما همه اینها تابع یک چیز هستند: اقتصاد. با اینکه فیبر نوری پهنای باند بیشتری ارائه می‌کند، اما حرف آخر را قیمت می‌زند. اگر تکنولوژی جدیدی اختراع شود که هزینه پرتاب ماهواره‌ها را بشدت کاهش دهد (مانند شاتل‌های فضایی که بتوانند هر بار دهها ماهواره را به مدار ببرند)، یا کاربرد غیر منتظره‌ای برای ماهواره‌های مدار پائین پیدا شود، معلوم نیست که فیبر نوری در تمام زمینه‌ها برنده باشد.

۵-۲ شبکه تلفن عمومی

اگر بخواهیم دو کامپیوتر را که نزدیک به هم و یا در یک ساختمان هستند، به یکدیگر متصل کنیم، کار بسیار ساده است و فقط کافیست یک رشته کابل بین آنها بکشیم - این همان شبکه محلی یا LAN است. اما اگر فاصله کامپیوترها زیاد باشد، یا کابل بایستی از املاک خصوصی یا شهری عبور داده شود، هزینه انجام کابل کشی معمولاً به مانعی بزرگ تبدیل می‌شود (البته اگر این کار غیرقانونی نباشد، که در اغلب کشورها هست). در نتیجه، طراحان شبکه به تأسیسات مخابراتی موجود روی می‌آورند.

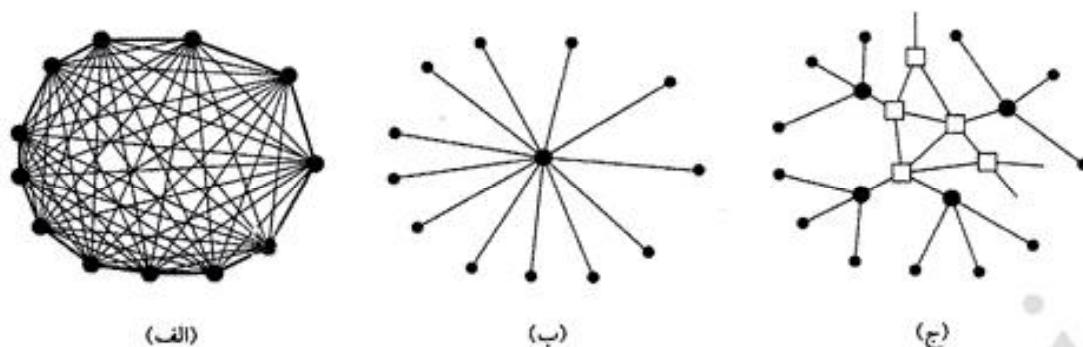
این تأسیسات بویژه شبکه تلفن عمومی - PSTN (Public Switched Telephone Network) - سالها قبل و با هدفی دیگر طراحی شده‌اند: انتقال صدای انسان بگونه‌ای کمابیش قابل تشخیص. کارایی این تجهیزات برای ارتباط کامپیوتر-کامپیوتر در بهترین حالت اغلب حاشیه‌ای است، اما با به بازار آمدن فیبر نوری و تکنولوژیهای دیجیتال این وضعیت سرعت در حال تغییر است. در هر حال، سیستم تلفن عمومی چنان با شبکه‌های (گسترده) کامپیوتری عجین است، که جا دارد وقت بیشتری به بررسی آن اختصاص دهیم.

برای بهتر شکافتن صورت مسئله، اجازه دهید مقایسه‌ای کلی (ولی روشن‌کننده) بین ارتباط دو کامپیوتر از طریق کابل مستقیم شبکه و از طریق خطوط تلفن داشته باشیم. کابل مستقیم شبکه می‌تواند داده‌ها را با سرعتی معادل 10^9 bps (یا حتی بیشتر) منتقل کند، در حالیکه حداکثر سرعت یک خط تلفن 56-kbps بیشتر نیست - تفاوتی در حد ۲۰,۰۰۰ برابر. تفاوت این دو مانند تفاوت سرعت یک مرغابی که سلاته سلاته در علفزار راه می‌رود، با موشکی است که به طرف ماه پرواز می‌کند. اگر به جای خط تلفن از اتصال ADSL استفاده کنیم، تفاوت سرعت به ۱۰۰۰ تا ۲۰۰۰ برابر خواهد رسید.

ناگفته پیداست که چنین سطح از تفاوتی برای طراحان سیستمهای کامپیوتری مشکل ساز است، و آنان تمام تلاش خود را برای بهینه کردن آن متمرکز کنند. در قسمتهای آینده سیستم تلفن و طرز کار آن را تشریح خواهیم کرد؛ برای اطلاعات بیشتر در این زمینه به (Bellamy, 2000) مراجعه کنید.

۱۵-۲ ساختار سیستم تلفن

بلافاصله بعد از آن که آلکساندر گراهام بل در سال ۱۸۷۶ (و درست چند ساعت زودتر از رقیبش، الیساگری)



شکل ۲-۲۰. (الف) شبکه‌ای با اتصالات داخلی کامل. (ب) سوئیچ مرکزی. (ج) سلسله مراتب دو سطحی.

اختراع خود را به ثبت رساند، سیل تقاضا برای آن سرازیر شد. تلفنهای اولیه صورت جفتی کار می‌کردند، و مشتریان مجبور بودند بین خودشان یک رشته سیم بکشند (مسیر برگشت الکترونها از زمین بود). اگر کسی می‌خواست با n نفر تماس تلفنی داشته باشد، مجبور بود به این n نقطه سیم بکشد. در کمتر از یک سال شهرها تبدیل شدند به جنگلی از سیمهای تو در تو که از خانه‌ای به خانه دیگر (از فراز ساختمانها و درختها) کشیده شده بود. خیلی زود معلوم شد که مدل اتصال هر تلفن به تمام تلفنهای دیگر مدلی عملی نیست - شکل ۲-۲۰ (الف). خوشبختانه بل خیلی زود متوجه این مشکل شد، و با تأسیس شرکتی بنام شرکت تلفن بل، اولین مرکز سوئیچینگ را در سال ۱۸۷۸ (در نیو هیون، کانیتیکات) راه‌اندازی کرد. این شرکت یک رشته کابل به خانه یا دفتر هر مشتری می‌کشید. برای برقراری تماس، مشتری دسته تلفن را می‌چرخاند تا زنگی در مرکز تلفن بصدا در آید و توجه اپراتور جلب شود؛ سپس این اپراتور ارتباط وی را توسط یک رشته کابل با مقصد موردنظر برقرار می‌کرد. این مدل را در شکل ۲-۲۰ (ب) ملاحظه می‌کنید.

بزودی در هر شهر و دهکده‌ای مراکز سوئیچینگ بل دایر شد، و بل مجبور شد برای برقراری تماسهای راه دور این مراکز را نیز به یکدیگر متصل کند. اما در اینجا هم همان مشکل قبلی خود را نشان داد: ارتباط مستقیم هر مرکز سوئیچینگ با تمام مراکز دیگر غیرممکن بود، بنابراین مراکز سوئیچینگ سطح دوم اختراع شد. پس از مدتی کوتاه تعداد مراکز سطح دوم نیز بشدت افزایش یافت - شکل ۲-۲۰ (ج) را ببینید. بعدها این سلسله مراتب تا پنج سطح بالا رفت.

تا سال ۱۸۹۰ این سیستم تلفن سه بخش عمده داشت: مراکز سوئیچینگ، سیمهایی که بین مراکز سوئیچینگ و مشترکان کشیده می‌شد (این سیمها دیگر سیمهای لخت با برگشت زمین نبود، بلکه تبدیل به سیمهای زوج تابیده عایق‌دار شده بود)، و اتصالات راه دور بین مراکز سوئیچینگ. با آن که پیشرفتهایی در هر یک از این سه بخش صورت گرفته، اما مدل اولیه بل در ۱۰۰ سال گذشته تقریباً بدون تغییر باقی مانده است. برای دیدن تاریخچه‌ای مختصر از سیستم تلفن بل، به (Hawley, 1991) نگاه کنید.

تا قبل از دو پاره شدن AT&T در سال ۱۹۸۴، سیستم تلفن سیستمی با سلسله مراتب چندسطحی (و با پراکندگی زیاد) بود. در بحث زیر این ساختار تا حد زیادی ساده شده، ولی عصاره اصلی آن همچنان حفظ شده است. هر تلفن با دو رشته سیم مسی به نزدیکترین ایستگاه پایانی (end office یا local central office) وصل می‌شود. فاصله این دو معمولاً بین ۱ تا ۱۰ کیلومتر است (و در شهرها کمتر از مناطق روستاییست). فقط در ایالات متحده آمریکا نزدیک به ۲۲,۰۰۰ ایستگاه پایانی وجود دارد. به اتصال دو سیمه بین ایستگاه پایانی و مشترک تلفن

اصطلاحاً مدار پایانی (local loop) گفته می شود. اگر مدارهای پایانی موجود در سراسر دنیا را بدنبال هم ردیف کنند، می تواند هزار بار فاصله زمین تا ماه را بپیماید.

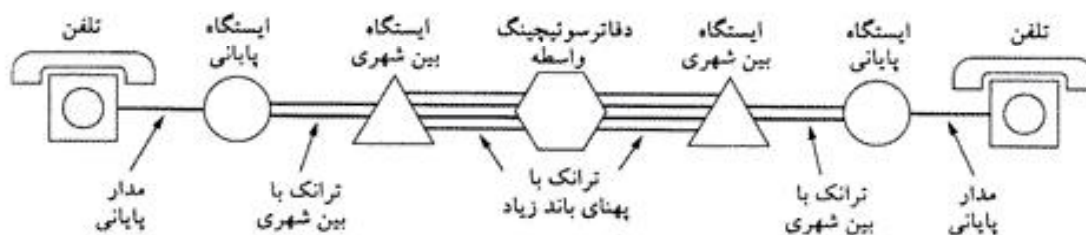
طبق یکی از تخمین ها، ۸۰ درصد ارزش سرمایه AT&T مس موجود در مدارهای پایانی آن است - بدین ترتیب، می توان AT&T را بزرگترین معدن مس دنیا بحساب آورد. خوشبختانه بورس کالا از این موضوع اطلاع چندانی ندارد - اگر این را می دانستند، بلافاصله AT&T را می خریدند، تمام سرویسهای تلفن را قطع می کردند، مس ها را در می آوردند، و دوباره به صنایع مس می فروختند.

وقتی یک مشترک به مشترک دیگری که به همان ایستگاه پایانی وصل است تلفن می زند، دستگاههای سوئیچینگ بین این دو مدار پایانی یک ارتباط الکتریکی مستقیم برقرار می کنند - این ارتباط مستقیم در تمام طول تماس برقرار می ماند.

اما اگر مشترک موردنظر متعلق به ایستگاه پایانی دیگری باشد، روش کار فرق می کند. هر ایستگاه پایانی دارای چند خط ارتباطی با یک یا چند مرکز سوئیچینگ، که اصطلاحاً ایستگاه بین شهری (toll office) نامیده می شوند، است - اگر این ایستگاه ها در یک منطقه باشند، به آنها ایستگاه شریک (tandem office) نیز گفته می شود. این خطوط ارتباطی را ترانک های مرتبط کننده بین شهری (toll connecting trunks) می نامند. اگر ایستگاههای پایانی تلفن کننده و تلفن شونده به یک ترانک وصل باشند (که در صورت نزدیکی آنها بسیار محتمل است)، امکان دارد ارتباط در همان ایستگاه بین شهری برقرار شود. در شکل ۲-۲۰ (ج) یک شبکه تلفن ساده را می بینید، که در آن تلفنها با نقاط سیاه کوچک، ایستگاههای پایانی با نقاط سیاه بزرگ، و ایستگاههای بین شهری با مربع نشان داده شده اند.

اگر تلفن کننده و تلفن شونده دارای ایستگاه بین شهری مشترک نباشند، مسیر ارتباطی بایستی در سطح بالاتری برقرار شود. ایستگاههای بین شهری از طریق ایستگاههای اولیه (primary office)، ناحیه ای (sectional office) و منطقه ای (regional office) به هم متصل می شوند. اتصال این ایستگاهها از طریق ترانک بین شهری (intertoll trunk یا interoffice trunk) برقرار می شود. نوع مراکز سوئیچینگ و توپولوژی آنها (مثلاً، اینکه دو ایستگاه ناحیه ای می توانند مستقیماً به یکدیگر وصل شوند، یا باید این ارتباط از طریق یک ایستگاه منطقه ای باشد؟) از کشوری به کشور دیگر فرق می کند، و به تراکم تلفن در آن کشور بستگی دارد. در شکل ۲-۲۱ نحوه هدایت یک تماس راه دور را ملاحظه می کنید.

در مخابرات راه دور از رسانه های مختلفی استفاده می شود. امروزه در مدارهای پایانی از کابل Cat 3 استفاده می شود، در حالیکه در سالهای اولیه اختراع تلفن از سیمهای بدون پوششی که با فاصله ۲۵ سانتیمتر بر فراز تیرهای تلفن کشیده می شد، استفاده می کردند. بین مراکز سوئیچینگ معمولاً کابل کواکسیال، مایکروویو یا فیبر نوری بکار برده می شود.



شکل ۲-۲۱. مدار هدایت یک تماس راه دور.

در گذشته انتقال سیگنالهای تلفن بصورت آنالوگ بود؛ ابتدا صدا به ولتاژ الکتریکی تبدیل شده، و سپس همین سیگنال روی خط تلفن ارسال می شد. با اختراع فیبر نوری، الکترونیک دیجیتال و کامپیوتر، امروزه دیگر تمام ترانک ها و سوئیچها دیجیتال هستند، و تنها قسمتی که هنوز از تکنولوژی آنالوگ استفاده می کند، همان مدار پایانی است. مزیت انتقال دیجیتال در اینست که دیگر مشکل اعوجاج سیگنال در تقویت کننده های مختلف وجود ندارد، و فقط کافیت بتوانیم 0 را از 1 تشخیص دهیم. انتقال دیجیتال مطمئنتر، ارزانتر، و نگهداری آن ساده تر است.

بطور خلاصه، سیستم تلفن از سه قسمت عمده تشکیل شده است:

۱. مدارهای پایانی (زوجهای تابیده آنالوگ که به خانه ها و دفاتر کشیده می شوند)
۲. ترانک ها (فیبرهای نوری دیجیتال که مراکز سوئیچینگ را به یکدیگر متصل می کنند)
۳. مراکز سوئیچینگ (مراکزی که تماسهای تلفنی را از یک خط اصلی به خط دیگر هدایت می کنند)

اجازه دهید قبل از پرداختن به ادامه این بحث، کمی هم درباره تلفن و سیاست صحبت کنیم. بعد از آن خواهیم دید که مدارهای پایانی و ترانک ها چگونه عمل می کنند، و سوئیچینگ چگونه انجام می شود.

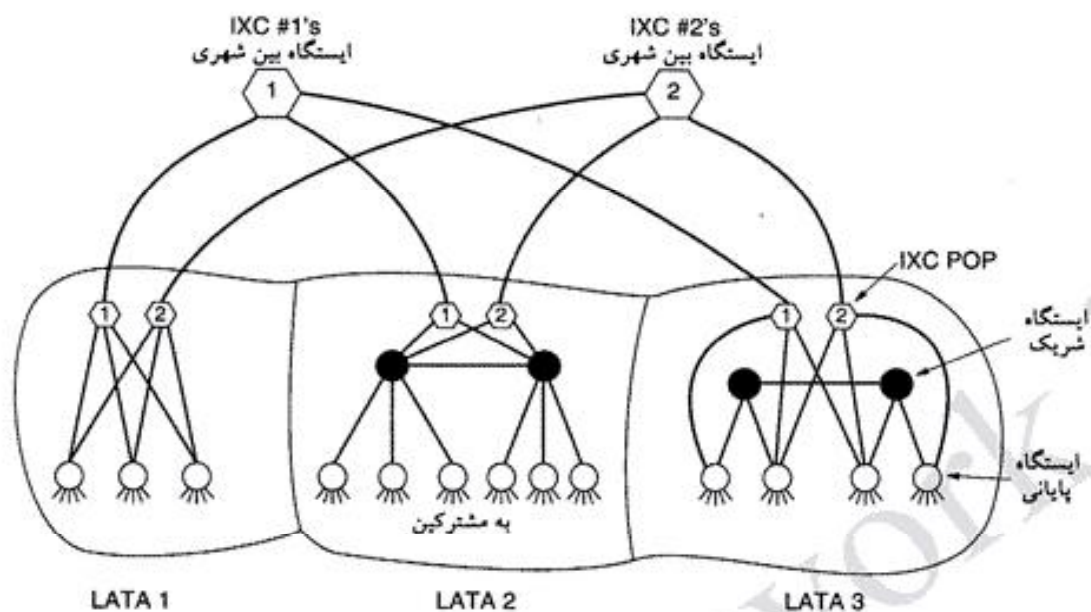
۲-۵-۲ تلفن و سیاست

تا قبل از سال ۱۹۸۴ برای چندین دهه انحصار تلفن شهری و راه دور ایالات متحده آمریکا در انحصار شرکت بل سیستم بود. در دهه ۱۹۷۰ دولت فدرال به این نتیجه رسید که این انحصار غیرقانونی است، و برای شکستن آن به دادگاه شکایت کرد. در اول ژانویه ۱۹۸۴ رأی دادگاه به نفع دولت فدرال صادر شد، و شرکت AT&T به شرکتهای BOC ۲۳، AT&T Long Lines (شرکتهای Bell Operating Company)، و چند شعبه کوچک دیگر شکسته شد. شرکتهای BOC برای آن که بتوانند در بازار رقابت کنند، هفت گروه منطقه ای (RBOC) تشکیل دادند. این رأی (که به Modified Final Judgment - MFJ - معروف است) مخابرات راه دور ایالات متحده را یک شبه دچار دگرگونی اساسی کرد.

دستور MFJ منجر به تغییرات مهمی در سرویسهای مخابراتی شد: افزایش رقابت بین شرکتهای مخابراتی، بهبود سرویسها، و کاهش قیمت مخابرات راه دور. از طرف دیگر قیمت سرویسهای شهری بشدت بالا رفت، چون این شرکتهای بایستی بنوعی کاهش درآمد خود را (که قبلاً از درآمد مخابرات راه دور تأمین می شد) جبران می کردند. بسیاری از کشورهای دیگر نیز در حال پیروی از این مدل هستند.

برای آن که مشخص شود چه کسی مجاز به انجام چه کاریست، ایالات متحده به ۱۶۴ LATA (مناطق محلی دسترسی و انتقال - Local Access and Transport Area) تقسیم شد. تقسیم بندی LATA ها تا حد زیادی (اما نه کاملاً) بر گندهای منطقه (area code) منطبق است. هر LATA دارای یک LEC (کاربر تبادل محلی - Local Exchange Carrier) است که انحصار کامل سرویسهای تلفن در آن منطقه را در اختیار دارد. مهمترین این LEC ها همان BOC ها هستند، ولی در برخی از مناطق شرکتهای مستقل نیز (که تعداد آنها به ۱۵۰۰ می رسد) فعالیت دارند.

کلیه ترافیک بین LATA ها توسط شرکتهایی بنام IXC (کاربر تبادل - InterExchange Carrier) انجام می شود. تا مدتی قبل AT&T Long Lines تنها IXC فعال در بازار بود، ولی اکنون شرکتهای بزرگی مانند Sprint و WorldCom نیز در این زمینه فعال هستند. یکی از دغدغه هایی که بعد از شکسته شدن AT&T وجود داشت آن بود که IXC ها در زمینه کیفیت خط، تعرفه ها، و تعداد رقمهای پیش شماره یکسان باشند. روش این در شکل ۲-۲۲ نشان داده شده است؛ در این شکل سه LATA با تعدادی ایستگاه پایانی می بینید. LATA های ۲ و ۳ با ایستگاههای بین شهری نیز ارتباط دارند.



شکل ۲-۲۲. ارتباط بین LATA ها و LEC ها و IXC ها

هر IXC که بخواهد مجری تماسهای یک LATA باشد، یک ایستگاه سونیچینگ بنام POP (نقطه تماس - Point Of Presence) در آنجا تأسیس می کند. وظیفه برقراری تماس IXC با ایستگاههای پایانی بر عهده LEC آن منطقه است (خواه بصورت مستقیم مانند LATA های ۱ و ۳، یا بصورت غیرمستقیم مانند LATA ۲). تماس (خواه فنی یا مالی) باید برای تمام IXC ها یکسان باشد. بدین ترتیب، مشترکی که مثلاً در LATA ۱ است، می تواند تصمیم بگیرد توسط کدام IXC با مشترکی در LATA ۳ تماس بگیرد.

در MFJ تصریح شده است که IXC ها نباید وارد بازار تلفن محلی شوند، و LEC ها هم از حضور در سرویسهای راه دور منع شده اند (البته آنها می توانند هر کار دیگری انجام دهند، مثلاً چپیس و پفک بفروشند). در سال ۱۹۸۴ این دستور نسبتاً واضح بود، ولی تکنولوژی همواره راههای جالبی برای منسوخ کردن قوانین پیدا می کند. از آنجائیکه تلفنهای همراه و تلویزیون کابلی مشمول دستور MFJ نمی شوند، LEC ها و IXC ها بسمت خرید این شرکتها (یا ادغام با آنها) روی آوردند.

در سال ۱۹۹۵ کنگره متوجه شد که جدا نگه داشتن حوزه فعالیت شرکتهای مختلف دیگر عملی نیست، و بهمین دلیل با تصویب یک لایحه به شرکتهای تلفن شهری، راه دور و تلویزیون کابلی اجازه داد تا وارد بازارهای یکدیگر شوند. ایده اصلی این لایحه اینست که سرویسهای صدا، داده و تلویزیون کابلی را یکپارچه کرده، و باعث رقابت بین شرکتهای مختلف (برای سرویس بهتر و قیمت کمتر) شود. این قانون از فوریه ۱۹۹۶ به اجرا گذاشته شد، و باعث شد تا تعدادی از BOC ها وارد حوزه IXC شوند، و از طرف دیگر شرکتهای تلویزیون کابلی نیز به ارائه سرویسهای تلفن بپردازند (و با LEC ها رقابت کنند).

یکی از نکات جالب قانون ۱۹۹۶ آنست که LEC ها بایستی شماره های تلفن را بگونه ای تنظیم کنند که قابل انتقال باشند. این بدان معناست که یک مشترک می تواند از این منطقه به منطقه دیگر برود، بدون آنکه نیازی باشد شماره تلفنش را عوض کند. این ویژگی باعث می شود تا افراد آزادی بیشتری در عوض کردن LEC خود داشته باشند، و در نتیجه تنوع رقابت داغتر شود. با قانون ۱۹۹۶، مخابرات ایالات متحده دستخوش تغییرات ساختاری شدیدی شده است. کشورهای بسیاری نیز ترغیب شده اند تا دست به چنین اقداماتی بزنند. البته در اغلب

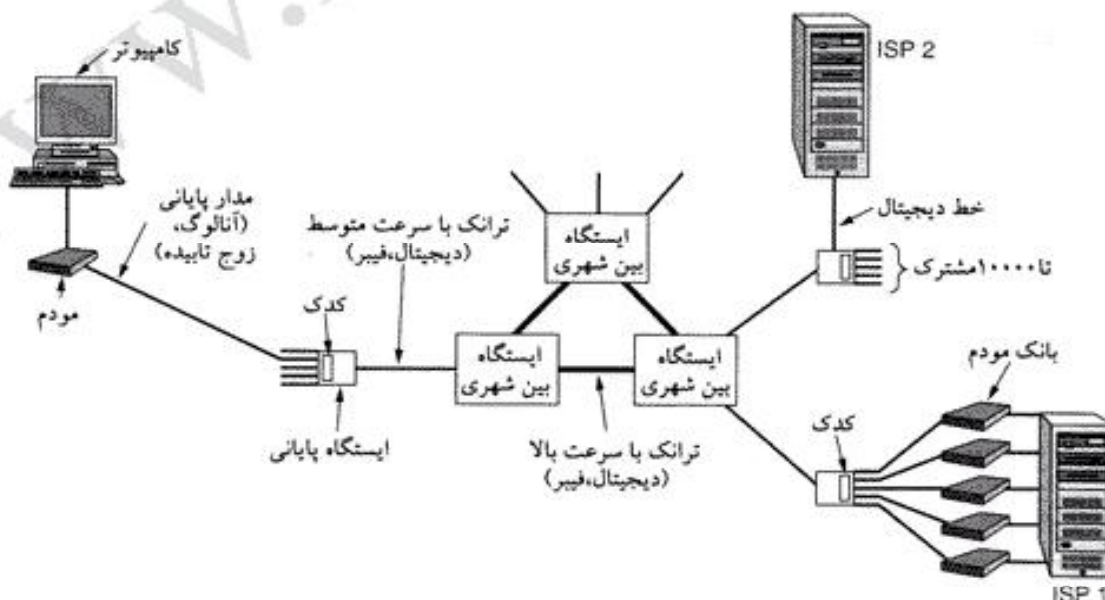
موارد کشورهای دیگر صبر می کنند تا نتیجه کار در ایالات متحده مشخص شود؛ اگر نتیجه مثبت بود، همان راه را می روند؛ اگر نتیجه منفی بود، راه دیگری را امتحان می کنند.

۳-۵-۲ مدارهای پایانی: مودم، ADSL، ویسیسم

اکنون زمان آن است که تا به بررسی نحوه کار سیستم تلفن بپردازیم. بخشهای اصلی این سیستم در شکل ۲-۲۳ نشان داده شده است. در این شکل مدارهای پایانی، ترانک ها، ایستگاه های بین شهری و ایستگاههای پایانی را می بینید. هر ایستگاه پایانی (در ایالات متحده و کشورهای بزرگ) تا ۱۰,۰۰۰ مدار پایانی دارد. در حقیقت تا همین اواخر، کد منطقه + شماره ناحیه مشخص کننده ایستگاه پایانی بود - برای مثال، 601-xxxx (212) یک ایستگاه پایانی با ۱۰,۰۰۰ مشترک (از 0000 تا 9999) است. با ایجاد شدن امکان رقابت بر سر سرویسهای محلی دیگر این وضعیت عملی نیست، چون شرکتهای زیادی طالب بدست آوردن کدهای محلی هستند. روشهای شماره گذاری نیز بایستی بکلی عوض شود، چون اغلب کدهای منطقه ای مصرف شده اند.

اجازه دهید با بخشی که بیشتر مردم با آن آشنا نیست، شروع کنیم: دو رشته سیمی که از ایستگاه پایانی شرکت تلفن به محل مشترک (خانه یا محل کار) کشیده می شود. به این دو رشته سیم اغلب «کیلومتر آخر» گفته می شود، اگرچه طول آن ممکنست به چندین کیلومتر هم برسد. در طول یکصد سال گذشته در این بخش از سیگنالهای آنالوگ استفاده شده است، و (بدلیل هزینه زیاد تجهیزات دیجیتال) احتمالاً در چند سال آینده نیز وضع به همین منوال خواهد بود. با این حال، در این آخرین سنگر آنالوگ نیز تغییرات شروع شده است. در این قسمت مدارهای پایانی و آخرین تحولات آنرا (با تأکید بر مخابرات داده بین کامپیوترها) بتفصیل بررسی خواهیم کرد.

وقتی یک کامپیوتر می خواهد داده های دیجیتال را روی یک خط تلفن آنالوگ ارسال کند، ابتدا باید آنها را به سیگنالهای آنالوگ تبدیل کند - کاری که با دستگاهی بنام مودم (modem) انجام می شود. در ایستگاه پایانی این اطلاعات مجدداً به سیگنالهای دیجیتال تبدیل شده، و برای ارسال روی ترانک راه دور (trunk) آماده می شود. اگر مقصد اطلاعات یک کامپیوتر باشد، سیگنال مجدداً به آنالوگ تبدیل می شود تا بتوان آنرا روی مدار پایانی



شکل ۲-۲۳. ترکیبی از انتقال دیجیتال و آنالوگ برای تماس کامپیوتر با کامپیوتر. تبدیل سیگنال بوسیله مودمها و کدکها انجام می شود.

آن ارسال کرد؛ و در مقصد یک مودم دیگر اطلاعات را از آنالوگ به دیجیتال تبدیل می کند. در شکل ۲-۲۳، ISP 1 (ارائه دهنده سرویسهای اینترنتی) یک بانک مودم دارد، که هر کدام از آنها به یک خط تلفن (مدار پایانی) مستقل متصلند. این ISP می تواند در آن واحد به تعداد مودمهای خود به افراد مختلف سرویس بدهد (البته با این فرض که کامپیوترهایش به اندازه کافی قوی باشند). این آرایش تا وقتی که مودمهای 56-kbps وارد بازار شدند، آرایش متداولی بود (بزودی علت آنرا خواهید فهمید).

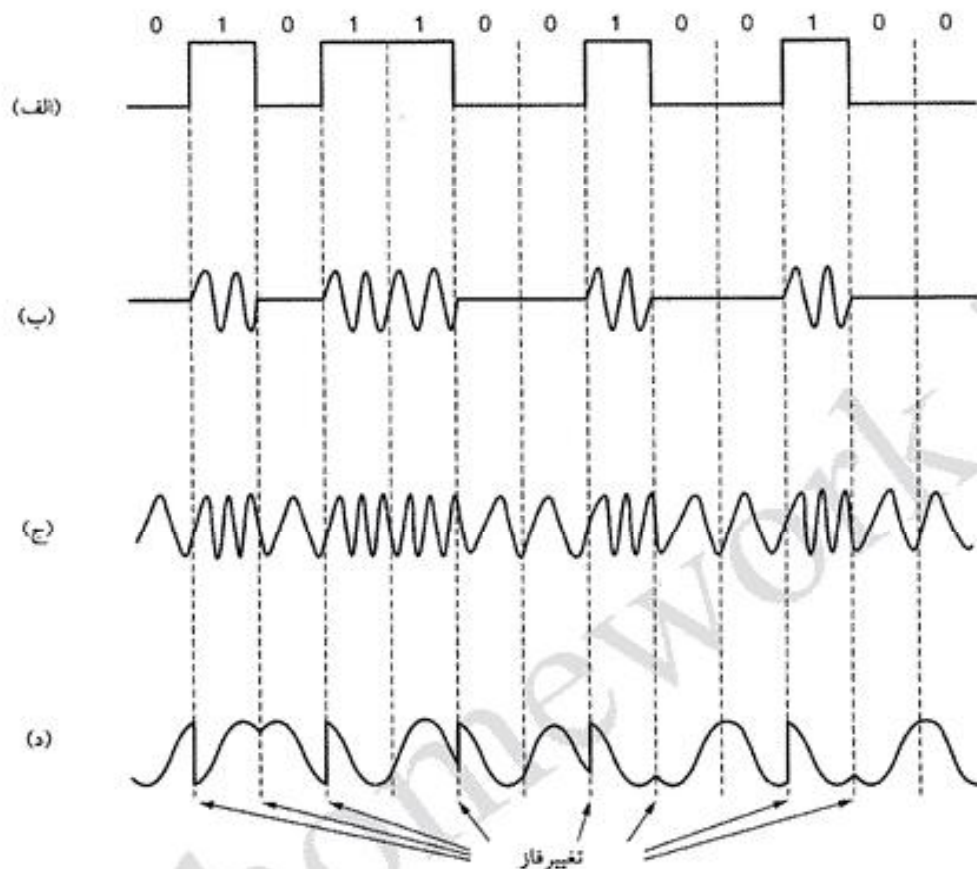
در مخابرات آنالوگ برای انتقال اطلاعات از ولتاژی که با زمان تغییر می کند، استفاده می شود. اگر رسانه انتقال کامل و بدون نقص باشد، گیرنده دقیقاً همان سیگنالی را دریافت خواهد کرد که فرستنده ارسال کرده است. اما متأسفانه چنین نیست، و گیرنده همان سیگنال ارسالی را دریافت نمی کند - و در مخابرات دیجیتال این یعنی خطا. انتقال سیگنالهای الکتریکی روی خطوط انتقال با سه مشکل عمده روبروست: تضعیف سیگنال، اعوجاج تأخیری، و نویز. وقتی یک سیگنال در رسانه انتقال منتشر می شود، انرژی خود را از دست می دهد که به آن تضعیف (attenuation) می گویند، و بر حسب دسی بل بر کیلومتر اندازه گیری می شود. میزان تضعیف یک سیگنال به فرکانس آن بستگی دارد. شاید فکر کنید که این وابستگی به فرکانس اشکال زیادی ایجاد نمی کند، ولی وقتی یک موج را بصورت مجموعه ای از مؤلفه های فوری در نظر بگیرید، تأثیر وابستگی به فرکانس خود را نشان خواهد داد. در واقع هر یک از مؤلفه های فوری به گونه ای متفاوت تضعیف می شوند، و ترکیب مجدد آنها در گیرنده موج کاملاً متفاوتی ایجاد می کند.

اما وضع از این هم بدتر است، چون مؤلفه های فوری با سرعتهای متفاوتی در رسانه انتقال (سیم) منتشر می شوند. این تفاوت سرعتها باعث اعوجاج تأخیری (delay distortion) سیگنال در گیرنده می شود. مشکل دیگر نویز (noise) - انرژی ناخواسته از منابعی غیر از فرستنده - است. نویز حرارتی حاصل حرکات تصادفی الکترونها در سیم است، و بکلی نمی توان از آن اجتناب کرد. نویز القایی نیز حاصل القای ولتاژ در اثر عبور جریان از سیمهای مجاور است. گاهی پیش آمده که وقتی تلفنی با کسی صحبت می کنید، مکالمه دیگری را نیز در زمینه می شنوید؛ علت این پدیده (که به همشنوایی - crosstalk - مشهور است) نویز القایی می باشد. در اثر قطع و وصل خطوط قدرت نویز دیگری بنام نویز ضربه ای روی خطوط مخابراتی القا می شود، که می تواند چندین بیت از اطلاعات را از بین ببرد.

مودم

بدلیل مشکلاتی که در بالا گفته شد (نویز و وابستگی تضعیف سیگنال و سرعت انتشار آن به فرکانس)، سعی می شود از سیگنالهایی با محدوده فرکانسی پائین استفاده شود. متأسفانه، شکل موج مربعی سیگنالهای دیجیتال دارای طیف فرکانسی وسیعی است، و بشدت در معرض تضعیف و اعوجاج تأخیری قرار دارد. این تأثیرات باعث شده تا سیگنالهای بیس باند (DC) فقط برای سرعتهای پائین و مسافتهای کوتاه مناسب باشد.

برای حل مشکل سیگنالهای DC بویژه در خطوط تلفن، از سیگنال AC استفاده می شود. در این جا یک تون ۱۰۰۰ تا ۲۰۰۰ هرتزی بعنوان موج حامل سینوسی (sine wave carrier) بکار برده می شود. برای انتقال اطلاعات می توان دامنه، فرکانس یا فاز این موج حامل را مدوله کرد. در مدولاسیون دامنه (amplitude modulation) از دو دامنه متفاوت بعنوان 0 و 1 استفاده می شود. در مدولاسیون فرکانس (frequency modulation) - که به کدگذاری با شیفت فرکانس (frequency shift keying) نیز معروف است - از دو تون متفاوت برای 0 و 1 استفاده می شود. (در صنعت مخابرات، اصطلاحات مدولاسیون و کدگذاری معادل یکدیگرند.) در مدولاسیون فاز (phase modulation) موج حامل در فواصل یکنواخت 0 یا 180 درجه شیفت پیدا می کند. با استفاده از شیفتهای 45، 135، 225 یا 315 درجه ای می توان در هر فاصله زمانی بجای یک بیت، ۲



شکل ۲-۲۴. (الف) سیگنال باینری. (ب) مدولاسیون دامنه. (ج) مدولاسیون فرکانس. (د) مدولاسیون فاز.

بیت اطلاعات منتقل کرد. همچنین، وجود تغییر فاز در انتهای هر فاصله زمانی تشخیص مرزهای آنها را برای گیرنده آسانتر می‌کند.

شکل ۲-۲۴ این سه نوع مدولاسیون را نشان می‌دهد. در شکل ۲-۲۴ (ب) یکی از دامنه‌ها صفر و دیگری غیر صفر است. در شکل ۲-۲۴ (ج) از دو فرکانس متفاوت برای نمایش ۱ و ۰ استفاده شده است. در شکل ۲-۲۴ (د) در هر فاصله زمانی وجود (یا عدم وجود) تغییر فاز نشان‌دهنده تغییر مقدار بیت (یا عدم تغییر آن) است.

دستگاهی که جریان بیت‌ها را بعنوان ورودی گرفته، و با ایجاد یک موج حامل و اعمال یکی از انواع مدولاسیون (یا ترکیبی از آنها) یک خروجی آنالوگ تولید می‌کند (و یا بر عکس، با گرفتن موج آنالوگ اطلاعات دیجیتال را از آن استخراج می‌کند)، مودم (modulator-demodulator) نامیده می‌شود. مودم بین کامپیوتر (دیجیتال) و سیستم تلفن (آنالوگ) قرار می‌گیرد.

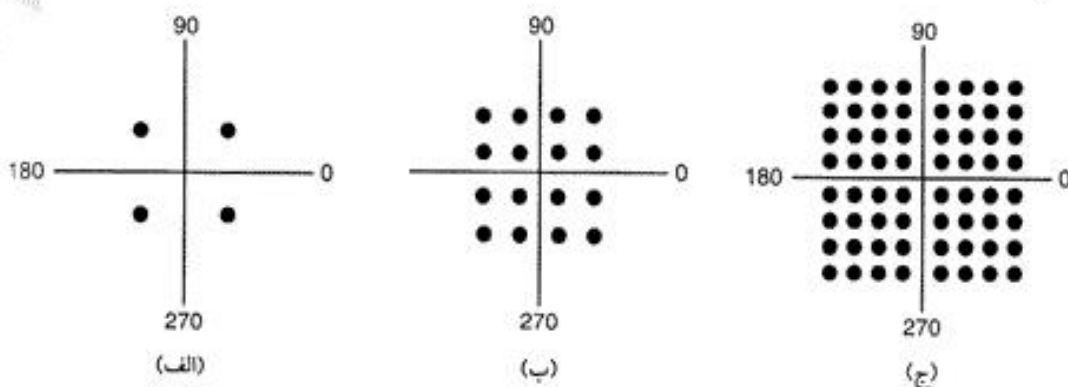
بالا بردن سرعت بسادگی و فقط با زیاد کردن نرخ نمونه‌برداری (sampling rate) ممکن نیست. قضیه نایکوئیست می‌گوید که برای یک خط کامل 3000 Hz (که خطوط تلفن مسلماً چنین نیستند)، حداکثر نرخ نمونه‌برداری 6000 Hz است. در عمل، اکثر مودمها با نرخ 2400 times/sec نمونه‌برداری می‌کنند، ولی سعی می‌کنند در هر نمونه‌برداری بیت‌های بیشتر را بخوانند.

به تعداد نمونه ها در ثانیه باد (baud) گفته می شود، و در هر باد یک سمبل (symbol) فرستاده می شود. بنابراین، یک خط n -baud در هر ثانیه n سمبل ارسال می کند (برای مثال، یک خط 2400-baud در هر 416.667 μ sec یک سمبل می فرستد). اگر این سمبل فقط حاوی ولتاژ 0 v (برای نمایش 0 منطقی) یا 1 v (برای نمایش 1 منطقی) باشد، سرعت مودم 2400 bps خواهد بود. اما اگر از ولتاژهای 0، 1، 2 و 3 ولت استفاده کنیم، هر سمبل می تواند حاوی 2 بیت باشد، و سرعت انتقال اطلاعات مودم به 4800 bps می رسد. در مدولاسیون فاز چهار درجه ای نیز می توان در هر سمبل 2 بیت ارسال کرد، و بدین ترتیب سرعت ارسال داده دو برابر سرعت باد خواهد بود. این تکنیک که کاربرد زیادی گسترده ای نیز دارد، QPSK (گذرانی با شیفت فاز چهارگانه - Quadrature Phase Shift Keying) نامیده می شود.

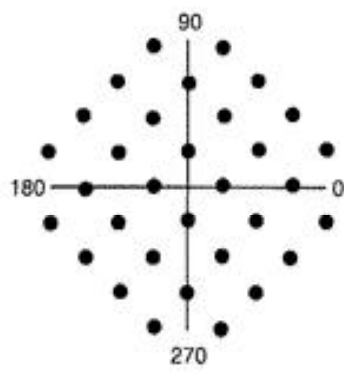
مفاهیم پهنای باند، باد، سمبل، و نرخ بیت بسیار با هم اشتباه می شوند، بنابراین اجازه دهید یک بار دیگر آنها را بیان کنیم. پهنای باند یک رسانه محدوده فرکانسی است که می تواند سیگنال را با کمترین تضعیف عبور دهد. این یکی از ویژگیهای فیزیکی رسانه انتقال است، و با هرتز (Hz) سنجیده می شود. به تعداد نمونه برداری در هر ثانیه باد گفته می شود، و هر نمونه حاوی یک قطعه از اطلاعات (یا سمبل) است. بنابراین، نرخ باد (baud rate) و نرخ سمبل (symbol rate) در واقع یکی هستند. تعداد بیت بر سمبل توسط نوع مدولاسیون (مثلاً، QPSK) تعیین می شود. نرخ بیت (bit rate) مقدار اطلاعاتیست که روی یک کانال فرستاده می شود، و برابر است با نرخ سمبل (symbol/sec) $\times$ تعداد بیت در هر سمبل (bits/symbol).

تمام مودمهای پیشرفته برای ارسال بیشترین بیتهای ممکن در هر باد، از مدولاسیونهای ترکیبی (چند دامنه ای و چند فازی) استفاده می کنند. در شکل ۲-۲۵ (الف) نقاطی را می بینید که در فواصل یکسان از مبدأ مختصات (دامنه یکسان)، و با زاویه ها (فازها) 45، 135، 225 و 315 درجه قرار گرفته اند (فاز هر نقطه زاویه ایست که با جهت مثبت محور x می سازد). در شکل ۲-۲۵ (الف) چهار ترکیب معتبر وجود دارد، که بدین ترتیب می توان در هر سمبل 2 بیت ارسال کرد - این همان QPSK است.

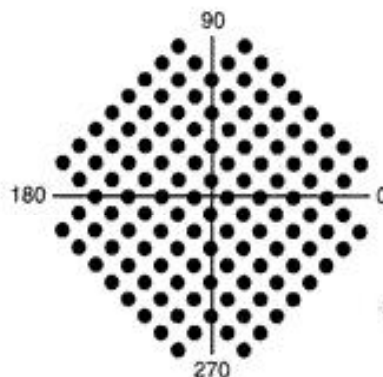
در شکل ۲-۲۵ (ب) مدولاسیون دیگری را می بینید، که در آن از چهار دامنه و چهار فاز مختلف استفاده شده و 16 ترکیب معتبر بدست می دهد. با این مدولاسیون می توان در هر سمبل چهار بیت ارسال کرد، و به آن QAM-16 (مدولاسیون دامنه چهارگانه - Quadrature Amplitude Modulation) گفته می شود (گاهی نیز 16-QAM خوانده می شود). با مدولاسیون QAM-16 می توان روی یک خط 2400-baud تا سرعت 9600 bps داده ارسال کرد.



شکل ۲-۲۵. (الف) مدولاسیون QPSK. (ب) مدولاسیون 16-QAM. (ج) مدولاسیون 64-QAM.



(ب)



(ج)

شکل ۲-۲۶. (الف) استاندارد V.32 برای 9600 bps. (ب) استاندارد V.32 bis برای 14,400 bps.

در شکل ۲-۲۵ (د) مدولاسیون دیگری را می بینید که شامل دامنه های بیشتری است. در این طرح ۶۴ ترکیب ممکنه وجود دارد، که بدین ترتیب می توان به ازای هر سمبل ۶ بیت را ارسال کرد. این مدولاسیون QAM-64 نام دارد (از مدولاسیون QAM با مراتب بالاتر نیز استفاده می شود).

دیاگرامهایی مانند شکل ۲-۲۵ که ترکیبات ممکنه دامنه و فاز را نشان می دهند، به دیاگرام فلکی (constellation diagram) معروفند. هر استاندارد مودمهای سرعت بالا دارای دیاگرام فلکی خاص خود است، و فقط قادر به ارتباط با مودمهاییست که دارای دیاگرام مشابهی باشند (البته این مودمها می توانند تمام سرعتهای پائینتر را شبیه سازی کنند).

با بالا رفتن تعداد نقاط در دیاگرام فلکی حتی نویزهای کوچک نیز می توانند باعث بروز خطا در آشکارسازی دامنه یا فاز سیگنال شوند، که بدنبال آن پهنای زیادی از دست می رود. برای کاهش احتمال خطا، در مودمهای سرعت بالا با اضافه کردن بیت های اضافی نوعی تصحیح خطا (error correction) انجام می شود. به این روش TCM (مدولاسیون کدگذاری تاروودی - Trellis Coded Modulation) می گویند. برای مثال، استاندارد V.32 از ۳۲ نقطه فلکی برای ارسال ۴ بیت داده و یک بیت برابری (parity) در هر سمبل استفاده کرده، و با نرخ 2400-baud به سرعت 9600 bps (با تصحیح خطا) دست می یابد. دیاگرام فلکی این استاندارد را در شکل ۲-۲۶ (الف) مشاهده می کنید. (چرخش ۴۵ درجه ای حول محور مختصات فقط بدلیل مهندسی اتخاذ شده است، و هیچگونه تأثیری روی ظرفیت اطلاعات ندارد).

بعد از 9600 bps قدم بعدی 14,400 bps است، که استاندارد آن V.32 bis نامیده می شود. برای رسیدن به این سرعت باید (با نرخ 2400-baud) در هر سمبل ۶ بیت داده و ۱ بیت برابری ارسال شود. دیاگرام فلکی این مودم را، که (وقتی از QAM-128 استفاده شود) ۱۲۸ نقطه دارد، در شکل ۲-۲۶ (ب) می بینید. مودمهایی که قابلیت فکس دارند، برای ارسال فکس از این استاندارد استفاده می کنند. مدولاسیون QAM-256 در هیچ یک از مودمهای تلفنی استفاده نمی شود، ولی در شبکه های کابلی کاربرد دارد (بعداً آنرا خواهید دید).

استاندارد مودم تلفنی بعدی V.34 است، که با سرعت 28,800 bps (12 data bits/symbol در 2400-baud) کار می کند. آخرین مودم در این سری استاندارد V.34 bis نام دارد، که به سرعت 33,600 bps (14 data bits/symbol در 2400-baud) دست پیدا می کند.

برای رسیدن به سرعتهای بیشتر، بسیاری از مودمها از تکنیکهای فشرده سازی استفاده می کنند تا به سرعتهای

بالتر از 33,600 bps برسند. از سوی دیگر، تقریباً تمام مودمها قبل از شروع ارسال داده‌ها کیفیت خط را چک می‌کنند، و اگر نقصی در کیفیت خط وجود داشته باشد، سرعت خود را آنقدر پائین می‌آورند تا ارسال مطمئن داده‌ها امکانپذیر باشد. به همین دلیل، سرعت مؤثر یک مودم می‌تواند کمتر، مساوی، و یا بیشتر از سرعت رسمی آن باشد.

تمام مودمهای جدید (با استفاده از فرکانسهای متفاوت برای ارسال و دریافت) اجازه می‌دهند تا ارسال و دریافت همزمان انجام شود. به چنین ارتباطی دو-طرفه همزمان (full duplex) گفته می‌شود، مانند یک اتوبان دو-باند. اگر در هر لحظه فقط از یک طرف ارتباط ممکن باشد، به آن دو-طرفه ناهمزمان (half duplex) گفته می‌شود (مانند راه‌آهن‌های یک-خطه). و اگر ارتباط فقط در یک جهت مجاز باشد، به آن یک‌طرفه (simplex) می‌گویند (مانند یک خیابان یک‌طرفه). یک رشته فیبر نوری که در یک سمت فقط دیود لیزری و در سمت دیگر فقط آشکارساز نوری دارد، نیز سیستمی یک‌طرفه است.

همانطور که قبلاً گفتیم، طبق قانون شانون سرعت انتقال روی خطوط تلفن از 35-kbps نمی‌تواند فراتر رود، به همین مودمهای استاندارد از سرعت 33,600 bps بالاتر نمی‌روند. شاید بپرسید پس مودمهای 56-kbps چطور کار می‌کنند؟ کمی صبر کنید، به آن هم خواهیم رسید.

اما این حد 35-kbps از کجا آمده است؟ این محدودیت به طول متوسط مدارهای پایانی و کیفیت آنها بستگی دارد. به شکل ۲-۲۳ نگاه کنید: ارتباطی که از کامپیوتر سمت چپ شروع شده و به 1 ISP ختم شود، از دو مدار پایانی آنالوگ (یکی در مبدأ و دیگری در مقصد) عبور می‌کند. هر یک از این حلقه‌ها میزان نویز سیگنال را بالا می‌برند. اگر بتوانیم یکی از این حلقه‌ها را حذف کنیم، می‌توانیم حداکثر سرعت را به دو برابر برسانیم.

این دقیقاً همان کاریست که 2 ISP انجام داده: این ISP ارتباط خود با نزدیکترین ایستگاه پایانی را بصورت کاملاً دیجیتال در آورده است. سیگنال دیجیتال ترانک مستقیماً به 2 ISP فرستاده شده، و مودمها و کودکهای آنالوگ بکلی حذف شده‌اند. بدین ترتیب، حداکثر سرعت ارتباط با این ISP می‌تواند به 70-kbps نیز برسد. بین دو نقطه که از مودم و خطوط آنالوگ استفاده می‌کنند، حداکثر سرعت همان 33.6 kbps است.

اما علت اینکه مودمهای 56-kbps می‌توانند در عمل کار کنند، به قضیه نایکوئیست برمی‌گردد. پهنای باند کانالهای تلفنی (بانضمام باند محافظ) 4000 Hz است، و طبق قضیه نایکوئیست حداکثر نرخ نمونه‌برداری در چنین کانالی 8000 خواهد بود. در ایالات متحده آمریکا تعداد بیت‌های هر نمونه ۸ است، که (با احتساب یک بیت کنترلی) حداکثر سرعت مجاز به 64,000 bits/sec می‌رسد. در اروپا از بیت کنترلی استفاده نمی‌شود، و می‌توان تمام بیتها را به داده‌ها اختصاص داد، بنابراین حداکثر سرعت در آنجا می‌تواند تا 64,000 bits/sec افزایش یابد، ولی طبق یک توافق بین‌المللی همان سرعت 56,000 انتخاب شده است.

این استاندارد V.90 نامیده می‌شود. در این استاندارد سرعت ارسال کاربر به ISP همان 33.6 kbps است، ولی سرعت دریافت از ISP به 56-kbps می‌رسد (چون معمولاً آنچه که کاربر از ISP می‌گیرد، بسیار بیشتر از آن چیزیست که به آن می‌فرستند). از لحاظ نظری امکان رساندن کانال ارسال به ISP تا 56-kbps نیز وجود داشت، ولی (بعلت نویزی بودن خطوط تلفنی) تصمیم گرفته شد تا این کانال به 33.6 kbps محدود شده، و پهنای باند آن به کانال دریافت از ISP داده شود تا احتمال رسیدن آن به سرعت 56-kbps افزایش یابد.

قدم بعدی در این بازی استانداردها، V.92 است. مودمهای این استاندارد می‌توانند تا 48-kbps روی کانال ارسال داده بفرستند (البته مشروط باینکه خط تلفن توانایی آنرا داشته باشد). زمان شناسایی کیفیت خط و تعیین سرعت مناسب در این استاندارد نیز بسیار کمتر از مودمهای دیگر است (تقریباً نصف ۳۰ ثانیه‌ای که مودمهای دیگر صرف این کار می‌کنند). و بالاخره، مودمهای استاندارد V.92 اجازه می‌دهند تا یک تماس تلفنی بتواند تماس اینترنتی را قطع کند، مشروط باینکه خط دارای سرویس انتظار مکالمه (call waiting) باشد.

خط مشترک دیجیتال (DSL)

وقتی شرکت‌های تلفن بالاخره موفق شدند سرویس 56-kbps ارائه کنند، خیلی به خود غرّه شدند؛ اما در همان حال، شرکت‌های تلویزیون کابلی ارتباطاتی با سرعت 10 Mbps (و ماهواره‌ها سرعت ارسال 50 Mbps) عرضه می‌کردند. با داغ شدن بازار اینترنت، شرکت‌های تلفن (LEC) دریافتند که برای رقابت به محصول جدیدی نیاز دارند. پاسخ آنها به این وضعیت ارائه سرویس‌های دیجیتال روی مدارهای پایانی بود. این سرویس‌ها پهنای باند بیشتری داشتند و به آنها سرویس باند-وسیع (broadband) گفته می‌شد (اگرچه این نامگذاری بیشتر جنبه تبلیغاتی داشت، تا فنی).

در ابتدا این سرویس‌ها بسیار متنوع بودند، و تحت نام xDSL (خط مشترک دیجیتال: Digital Subscriber Line - که در آن x نوع سرویس را مشخص می‌کند) دسته‌بندی می‌شدند. مهمترین این سرویس‌ها - که عامل اصلی موفقیت آن هم بود - ADSL (DSL نامتقارن - Asymmetric DSL) نام دارد، که در زیر با آن آشنا خواهید شد. از آنجائیکه ADSL هنوز در حال توسعه و تکامل است و استانداردهای آن هنوز بطور کامل تدوین نشده، ممکنست در آینده تغییری در آن رخ دهد، ولی تصویر کلی همان است که خواهید دید. برای اطلاعات بیشتر درباره ADSL به (Summers, 1999; Vetter et al., 2000) نگاه کنید.

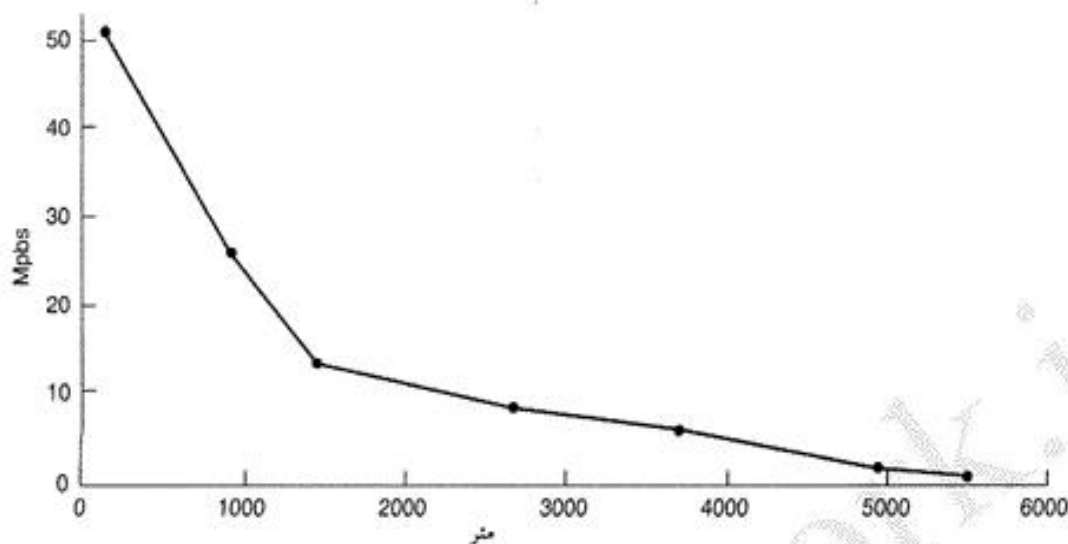
علت آن که مودم‌ها اینقدر کُده هستند اینست که، شبکه تلفن اساساً برای انتقال صدای انسان طراحی و توسعه داده شده، و سرویس‌های داده فرزندخوانده آن محسوب می‌شود. در نقطه‌ای که مدار پایانی وارد ایستگاه پایانی تلفن می‌شود، فیلترهایی قرار داده شده که تمام فرکانس‌های زیر 300 Hz و بالای 3400 Hz را تضعیف می‌کنند. البته قطع فرکانس بیکباره صورت نمی‌گیرد - از لحاظ فنی، 300 Hz و 3400 Hz نقاط 3 dB هستند - بهمین دلیل پهنای باند را معمولاً 4000 Hz فرض می‌کنند، اگرچه فاصله نقاط 3 dB فقط 3100 Hz است. داده نیز به همین باند باریک محدود است.

برای اجتناب از این وضعیت در سرویس xDSL، خط مشترک بدون عبور از فیلترهای مزبور مستقیماً به نوع خاصی از سوئیچ‌ها متصل می‌شود، تا بتواند از تمام ظرفیت مدار پایانی استفاده کند. در این حالت دیگر محدودیت پهنای باند فقط به خواص فیزیکی مدار پایانی بستگی دارد، نه به محدوده‌ای که فیلترها بطور مصنوعی برای آن بوجود آورده‌اند.

متأسفانه، ظرفیت مدار پایانی نامحدود نیست، و به عواملی از قبیل طول خط، ضخامت سیم، و کیفیت کلی آن بستگی دارد. در شکل ۲-۲۷ نمودار پهنای باند بر حسب مسافت را ملاحظه می‌کنید - در اینجا فرض شده که سایر عوامل بهینه هستند (سیم‌های نو، کابل‌های نه چندان قهقور، و غیره).

نمودار فوق مشکل اصلی شرکت‌های تلفن را بخوبی نشان می‌دهد: شعاع ارائه این سرویس به مشترکان بشدت محدود است. این بدان معناست که وقتی کاربری که خارج از این شعاع زندگی می‌کند، برای دریافت سرویس xDSL مراجعه کند، باید با کمال تأسف از او عذرخواهی کنیم که امکان ارائه سرویس به وی را نداریم. برای بیشتر کردن شعاع سرویس، باید سرعت آنرا پائین بیاوریم، ولی پائین آوردن سرعت همان و از دست دادن جذابیت همان. اینجاست که تکنولوژی مغلوب اقتصاد می‌شود. (راه حل دیگر آنست که ایستگاه‌های کوچک و پراکنده‌ای در نقاط نزدیک به هم تأسیس کنیم، که البته این هم اقتصادی نیست).

سرویس xDSL با اهداف مشخصی طراحی شده است: اول اینکه، این سرویس‌ها باید بتوانند با خطوط زوج تأیید Cat 3 کار کنند؛ دوم اینکه، این سرویس‌ها نباید هیچ اختلالی در دستگاه‌های تلفن و فکس معمولی بوجود آورند؛ سوم اینکه، باید از 56-kbps سریعتر باشند؛ و چهارم اینکه، این سرویس‌ها باید دائماً برقرار باشند، و هزینه آنها هم ثابت (و مثلاً ماهیانه) باشد - نه مانند تلفن‌های معمولی، بصورت دقیقه‌ای.

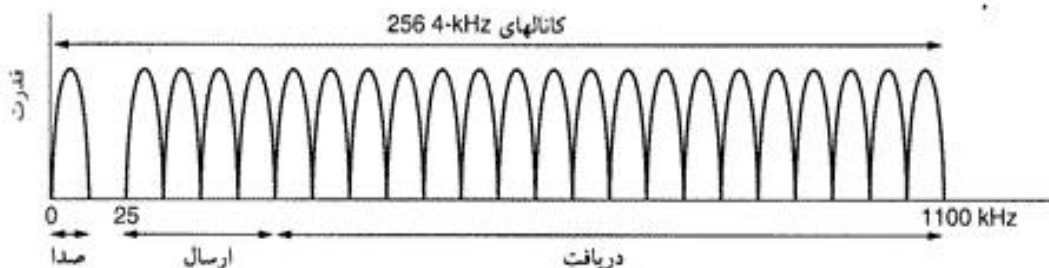


شکل ۲-۲۷. نمودار پهنای باند سرویس DSL بر حسب طول سیم در کابل Cat 3 UTP.

اولین سرویس ADSL توسط AT&T ارائه شد، که در آن پهنای باند موجود در مدار پایانی (که تقریباً 1.1 MHz است) به سه باند تقسیم شده بود: باند POTS (سرویس تلفن معمولی - Plain Old Telephone Service)، باند ارسال از کاربر (به ایستگاه پایانی)، و باند ارسال به کاربر (از ایستگاه پایانی). تکنیک تقسیم پهنای باند به فرکانسهای مختلف مالتی پلکس تقسیم فرکانس (frequency division multiplexing) نام دارد، که بعداً درباره آن مفصلاً صحبت خواهیم کرد. شرکت های بعدی برای ارائه این سرویس از تکنیک متفاوتی استفاده کردند، و از آنجائیکه احتمالاً این روش غالب خواهد شد، ابتدا آنرا توضیح می دهیم.

این روش که DMT (تون چندگانه گسسته - Discrete MultiTone) نامیده می شود، در شکل ۲-۲۸ نشان داده شده است. در حقیقت، کاری که در اینجا انجام شده تقسیم پهنای باند موجود (1.1 MHz) به 256 کانال مستقل 4312.5 Hz است. از کانال 0 بعنوان POTS (سرویس تلفن معمولی) استفاده می شود. کانالهای 1-3 خالی رها شده اند، تا تداخلی بین صدا و داده پیش نیاید. از 250 کانال باقیمانده، یکی برای کنترل ارسال از کاربر و یکی برای کنترل ارسال به کاربر تخصیص یافته، و از بقیه کانالها می توان برای داده استفاده کرد.

در تئوری می توان از هر یک از این کانالها برای ارتباط دو-طرفه همزمان استفاده کرد، ولی هارمونیک ها، همشنوایی و اثرات دیگر باعث می شود تا در عمل ظرفیت سیستم بسیار کمتر باشد. این ارائه دهنده سرویس است که تعیین می کند چند کانال برای ارسال از کاربر اختصاص یافته، و چند کانال برای ارسال به کاربر. از نظر تکنیکی می توان این نسبت را بصورت 50-50 تعریف کرد، ولی از آنجائیکه اغلب کاربران اطلاعات دریافتی خیلی بیشتری دارند، ۸۰ الی ۹۰ درصد پهنای باند به دریافت کاربر اختصاص داده می شود. از اینجا است که حرف "A" (بمعنای



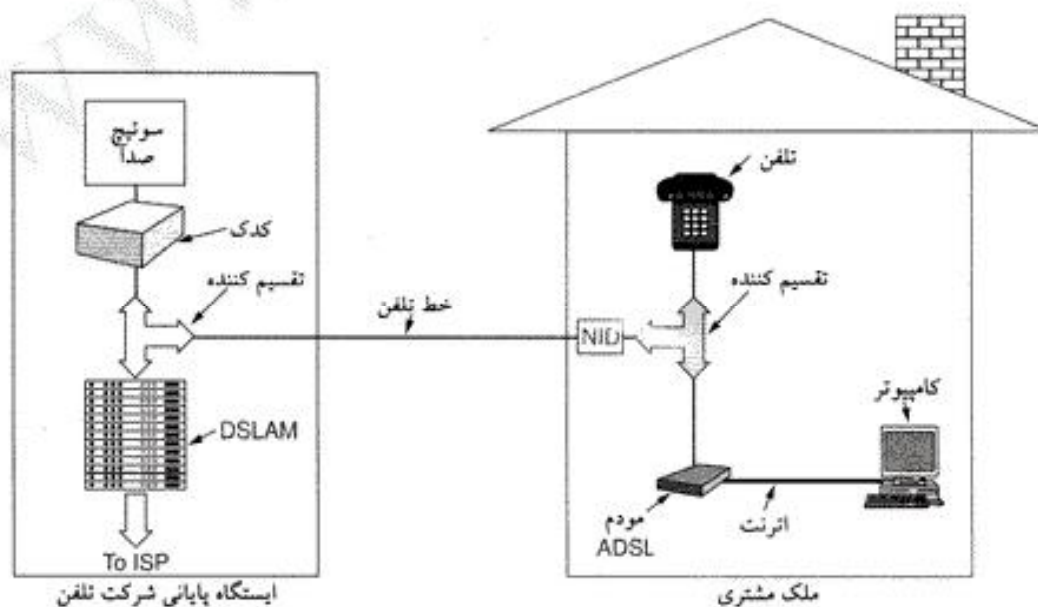
شکل ۲-۲۸. عملکرد ADSL با مدولاسیون تون چندگانه گسسته.

"نامتقارن" در ADSL ظاهر می‌شود. در اغلب موارد ۳۲ کانال به ارسال کاریر، و بقیه به دریافت آن اختصاص می‌یابد. به منظور افزایش پهنای باند، امکان اختصاص چند کانال فوقانی به ارتباط دوطرفه نیز وجود دارد، ولی این روش به تجهیزات اضافی برای خنثی کردن پژواک (echo) در خط نیاز دارد.

در استاندارد ADSL (ANSI T1.413 و ITU G.992.1) تا 8 Mbps برای دریافت کاریر و تا 1 Mbps برای ارسال کاریر مجاز است. اما، کمتر شرکتی پیدا می‌کنید که چنین سرویسهایی ارائه کند. در عمل، این سرویسها معمولاً بصورت 512 kbps دریافت و 64 kbps ارسال (سرویس معمولی)، و 1 Mbps دریافت و 256 kbps ارسال (سرویس ویژه) ارائه می‌شوند.

در هر کانال از مدولاسیونی شبیه V.34 (با نرخ نمونه برداری 4000-baud بجای 2400-baud) استفاده می‌شود. کیفیت خط در هر کانال مستقلاً (وبصورت پیوسته) ارزیابی شده، و در صورت نیاز سرعت مجدداً تنظیم می‌شود، بنابراین سرعت کانالها می‌تواند کاملاً متفاوت باشد. برای ارسال داده‌ها از مدولاسیون QAM با نرخ 15 bits/baud (شبیه شکل ۲-۲۵ ب) استفاده می‌شود. برای مثال، اگر 224 کانال دریافت با نرخ 15 bits/baud در 4000-baud داشته باشیم، پهنای باند دریافتی معادل 13.44 Mbps خواهیم داشت. در عمل، نرخ سیگنال به-نویز هرگز آنقدر خوب نیست که اجازه چنین سرعتهای را بدهد، ولی در مسافتهای کوتاه و با کابل کشی مناسب می‌توان به سرعت 8 Mbps رسید، که همین برای استقبال عامه کافیست.

در شکل ۲-۲۹ یک طرح ADSL را مشاهده می‌کنید. در این طرح، تکنسین شرکت تلفن بایستی یک NID (دستگاه واسط شبکه - Network Interface Device) در محل سکونت مشتری نصب کند. این جعبه پلاستیکی کوچک انتهای مایملک شرکت تلفن و شروع مایملک مشتری را مشخص می‌کند. کنار NID (یا حتی گاهی بصورت ترکیبی با آن) یک تقسیم‌کننده (splitter - فیلتر آنالوگی که باند POTS را از باند داده جدا می‌کند) نصب می‌شود. سیگنال POTS به دستگاه تلفن، و سیگنال داده به یک مودم ADSL داده می‌شود. مودم ADSL در واقع یک DSP (پردازشگر سیگنال دیجیتال - Digital Signal Processor) است، که بگونه‌ای تنظیم شده تا بعنوان 256 مودم QAM موازی (با فرکانسهای مختلف) کار کند. از آنجائیکه اکثر مودمهای ADSL بصورت خارج از



شکل ۲-۲۹. یک طرح ساده ADSL.

کامپیوتر هستند، ارتباط آنها با کامپیوتر نیز باید از نوع پرسرعت باشد. برای این منظور از پورتهای اترنت یا USB استفاده می شود. بدون شک در آینده شاهد به بازار آمدن مودمهای داخلی ADSL نیز خواهیم بود. در انتهای دیگر خط (در ایستگاه پایانی)، یک تقسیم کننده متناظر قرار می گیرد. در اینجا، سیگنال صدا از داده جدا شده و به سوییچ های معمولی تلفن فرستاده می شود. سیگنال بالای 26 kHz نیز جدا شده، و به دستگاهی بنام DSLAM (مالتی پلکسر دسترسی DSL - DSL Access Multiplexer) که بسیار شبیه مودم ADSL است، می رود. پس از تبدیل این سیگنال به جریان بیت های دیجیتال، از آنجا به ISP فرستاده می شود. جداسازی کامل سیستم صدا از ADSL، پیاده سازی آنرا برای شرکتهای تلفن بسیار ساده کرده است؛ تمام کاری که باید انجام دهد اینست که در سمت خود یک تقسیم کننده و DSLAM، و در سمت مشتری یک تقسیم کننده ساده نصب کند. در سیستمهای پرسرعت دیگر (مانند، ISDN) تجهیزات بسیار پیچیده تری باید نصب شود.

یکی از معایب طرح شکل ۲-۲۹ وجود تجهیزاتی است که باید در محل سکونت مشترک نصب شود (NID و تقسیم کننده). این کار مستلزم آنست که یک تکنسین به آنجا مراجعه کرده، و وسایل را نصب کند. به همین دلیل استاندارد جدیدی که به تقسیم کننده نیازی ندارد، توسعه داده شده است. نام غیررسمی این استاندارد G.lite است، ولی رسماً به آن ITU G.992.2 گفته می شود. این طرح شبیه شکل ۲-۲۹ است، که فقط تقسیم کننده حذف شده، و از خط تلفن بهمان صورت موجود استفاده می شود. تنها تفاوت اینست که باید میکروفیلترهایی را بین پرز تلفن و تجهیزات (دستگاه تلفن و مودم ADSL) قرار داد. میکروفیلتر تلفن یک فیلتر پائین گذر (low-pass) است که فرکانسهای بالای 3400 Hz را حذف می کند؛ میکروفیلتر مودم ADSL یک فیلتر بالاگذر (high-pass) است که فرکانسهای زیر 26 kHz را حذف می کند. با این حال کارایی G.lite مانند زمانی که تقسیم کننده بکار می رود، نیست و تنها می تواند به سرعت 1.5 MHz برسد (ولی هزاران مراجعه مستقیم به منازل مشترکان را حذف می کند). در استاندارد G.lite به تقسیم کننده ایستگاه پایانی همچنان نیاز هست.

باید بیاد داشت که، ADSL یک استاندارد لایه فیزیکی است، و آنچه که روی آن اجرا می شود به کاربر تلفن بستگی دارد. یکی از این کاربردها ATM است، چون در ATM امکان کنترل کیفیت وجود دارد و بسیاری از شرکتهای تلفن زیرساختهای آنرا در اختیار دارند.

مدار پایانی بیسیم

از سال ۱۹۹۶ در ایالات متحده آمریکا (و کمی بعد در کشورهای دیگر)، شرکتهایی که مایل بودند وارد رقابت با انحصارگر تلفن شهری (که به ILEC شهرت داشت) شوند، اجازه آنرا یافتند. شرکتهای تلفن راه دور (IXC ها) اولین کسانی بودند، که وارد گود شدند. هر IXC که می خواست وارد بازار تلفن شهری شود، باید کارهای ذیل را انجام می داد. اول، زمین یا ساختمانی برای تأسیس ایستگاه پایانی (end office) بخرد یا اجاره کند. دوم، تجهیزات و سوییچهای لازم را خریداری و در ایستگاه پایانی نصب کند. سوم، یک (یا چند) رشته فیبر نوری بین این ایستگاه پایانی و نزدیکترین ایستگاه بین شهری (tool office) بکشد، تا مشترکان آن بتوانند به شبکه تلفن کشوری دسترسی داشته باشند. چهارم، در صدد جلب مشتری برآید (با تبلیغ و ارائه سرویسهای بهتر و قیمت کمتر). و قسمت سخت کار همین جاست.

فرض کنید یک مشتری پیدا شده و می خواهد از شرکت تلفن شهری جدید (که آنرا CLEC می نامیم) سرویس تلفن بخرد. این شرکت چگونه باید مشتری را به ایستگاه پایانی خود - که خیلی هم برای آن خرج کرده - وصل کند؟ خریدن زمین در تمام طول مسیر، کندن کانال، و کشیدن یک خط تلفن بسیار پرهزینه است. بسیاری از این CLEC ها راه ساده تر و کم هزینه تری پیدا کرده اند: WLL (مدار پایانی بیسیم - Wireless Local Loop).

از یک نظر تلفنهای ثابت با مدار پایانی بیسیم شبیه تلفنهای همراه هستند، ولی سه تفاوت فنی مهم و اساسی با آنها دارند: اول، مشترکان WLL دسترسی اینترنت پرسرعت (با سرعتی حداقل معادل ADSL) می خواهند. دوم، این قبیل مشترکان انتظار ندارند (و اغلب اجازه نمی دهند) یک آنتن بشقابی بزرگ بالای بام خانه هایشان نصب شود. سوم، جای این تلفنها اساساً ثابت است، و مشکلات جابجا شدن سیگنال را (که در تلفنهای همراه وجود دارد، و بعداً در همین فصل با آنها آشنا خواهید دید) ندارند. بدین ترتیب یک صنعت کاملاً جدید متولد می شود: بیسیم ثابت (سرویس تلفن معمولی و اینترنت روی مدار پایانی بیسیم).

با اینکه آغاز یکار جدی WLL از سال ۱۹۹۸ است، ولی منشأ آن به سال ۱۹۶۹ برمی گردد. در این سال، FCC دو کانال تلویزیونی (هر یک با پهنای باند 6 MHz در فرکانس 2.1 GHz) به مصارف آموزشی اختصاص داد. در سالهای بعد تعداد این کانالها به ۳۳ (و پهنای باند مجموع آنها به 198 MHz در 2.5 GHz) افزایش یافت.

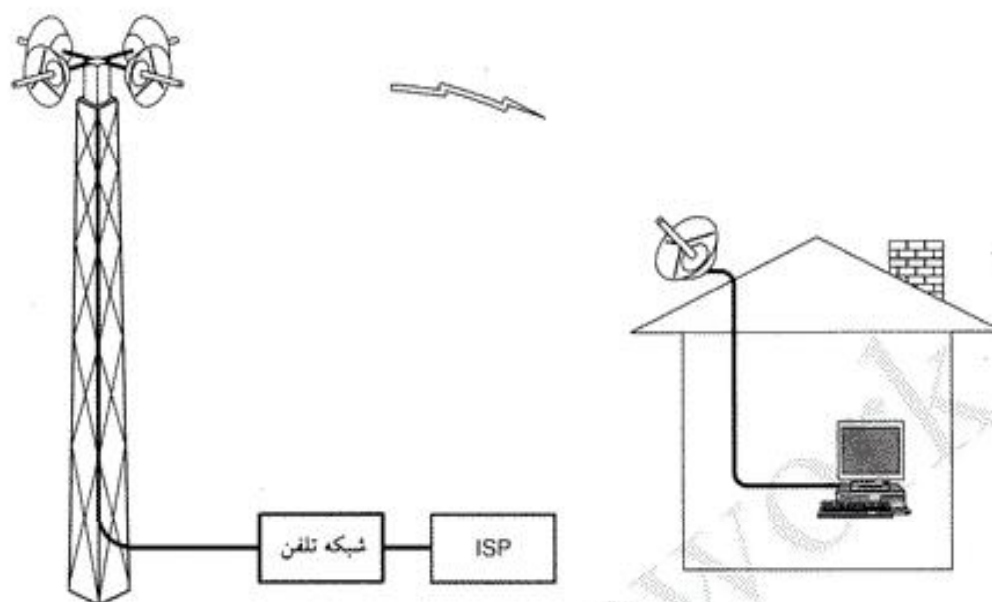
تلویزیون آموزشی هرگز پا نگرفت و در سال ۱۹۹۸، FCC این کانالها را پس گرفت و به ارتباطات رادیویی دوطرفه اختصاص داد - که بلافاصله در مدارهای پایانی بیسیم یکار گرفته شد. در این فرکانسها، طول موج امواج 10-12 cm است. بُرد این امواج حدود ۵۰ کیلومتر است، و از گیاهان و قطرات باران نسبتاً خوب عبور می کنند. سرویس جدیدی، که MMDS (سرویس توزیع چندکاناله چندنقطه‌ای - Multichannel Multipoint Distribution Service) نام گرفت، تمامی پهنای باند 198 MHz را مصرف کرد؛ MMDS را می توان (مانند پسر عمیش LMDS - که در همین قسمت آنرا توضیح می دهیم) یک شبکه شهری (MAN) در نظر گرفت.

مزیت بزرگ این سرویس تکنولوژی جا افتاده و موجود بودن تجهیزات آن است؛ و بزرگترین عیب آن این است که پهنای باند متوسطی دارد، که آن هم باید بین افراد زیادی در یک محدوده جغرافیایی وسیع به اشتراک گذاشته شود.

پهنای باند پائین MMDS باعث جذب شرکتهای تلفن به امواج میلیمتری شد. در فرکانسهای 28-31 GHz در آمریکا (و 40 GHz در اروپا) تمام باندها آزاد هستند، چون تکنولوژی تجهیزات نیمه هادی که در این فرکانسها کار کنند، پیچیده و گرانقیمت است. اما این مشکل هم با اختراع مدارات مجتمع گالیوم-آرسناید (Ga-As IC) حل شده، و راه برای ارتباطات رادیویی در باندهای میلیمتری هموار شده است. در پاسخ به تقاضاهای موجود، FCC یک پهنای باند 1.3 GHz به سرویس بیسیم جدید بنام LMDS (سرویس توزیع چندنقطه‌ای محلی - Local Multipoint Distribution Service) اختصاص داد - که این بزرگترین پهنای باندی بود که FCC تا به آن روز به یک کاربرد خاص اختصاص می داد. پهنای باند مشابهی نیز در اروپا (در فرکانس 40 GHz) به این سرویس اختصاص داده شده است.

عملکرد LMDS را در شکل ۲-۳۰ ملاحظه می کنید. در این شکل یک برج مخابراتی می بینید، که تعدادی آنتن در جهتهای مختلف روی آن نصب شده است. از آنجائیکه امواج میلیمتری شدت خطی هستند، هر آنتن زاویه محدودی (که به قطاع معروف است) را پوشش می دهد، که مستقل از سایر آنتنهاست. در این فرکانس، بُرد امواج 2-5 km است، و بهمین دلیل برای پوشش کامل یک شهر به تعداد زیادی از این برجهای مخابراتی نیاز هست.

در سرویس LMDS نیز، مانند ADSL، تخصیص پهنای باند نامتقارن (و به نفع کانال دریافت) است. با تکنولوژی موجود، هر قطاع می تواند تا 36 Mbps روی کانال دریافت کاربر و تا 1 Mbps روی کانال ارسال کاربر ظرفیت داشته باشد، که بین تمام کاربران موجود در آن قطاع به اشتراک گذاشته می شود. اگر هر کاربر در هر دقیقه سه صفحه 5-KB دریافت کند، بطور متوسط 2000 bps از پهنای باند را اشغال می کند، که بدین ترتیب همزمان حداکثر 18,000 کاربر می توانند در هر قطاع کار کنند. البته برای قابل قبول بودن سرویس، نباید در هر لحظه بیش از ۹۰۰۰ کاربر فعال داشته باشیم. با فرض داشتن چهار قطاع در یک برج (مانند شکل ۲-۳۰)، تعداد کاربرانی که در



شکل ۲-۳۰. معماری یک سیستم LMDS.

هر لحظه می توانند فعال باشند به ۳۶,۰۰۰ می رسد - و اگر فرض کنیم در ساعات اوج مصرف از هر سه کاربر یکی فعال است، تعداد کل مشترکانی که یک برج مخابراتی در شعاع ۵ کیلومتری می تواند پوشش دهد، به ۱۰۰,۰۰۰ بالغ می شود. (بر مبنای همین محاسبات بود که تعداد زیادی از CLEC ها به این نتیجه رسیدند که با یک سرمایه گذاری متوسط در برجهای مخابراتی امواج میلیمتری، می توانند در بازار تلفن شهری و اینترنت با شرکتهای تلویزیون کابلی رقابت کنند.)

اما، LMDS هم مشکلات خاص خود را دارد. اول اینکه، امواج میلیمتری کاملاً خطی هستند، و هیچ مانعی در خط دید برج مخابراتی و آنتن گیرنده نایستی وجود داشته باشد. دیگر اینکه، برگ درختان جاذب خوبی برای این امواج است، بنابراین برج فرستنده بایستی آنقدر مرتفع باشد که درختان مانع امواج آن نشوند (و باید توجه داشت مسیری که در زمستان صاف است، ممکنست در تابستان که درختان پراز برگ هستند، چنین نباشد). باران نیز امواج میلیمتری را جذب می کند. این مشکل را می توان تا حدی با گداهای تصحیح خطا، و افزایش توان تشعشعی فرستنده در روزهای بارانی جبران کرد. معهذ، سرویس LMDS در مناطق خشک شانس بیشتری برای موفقیت دارد، تا مناطق مرطوب و پُر باران.

مدار پایانی بیسیم بدون وجود استانداردهای معتبر شانس برای موفقیت ندارد، چون فقط در صورت وجود این استانداردهاست که سازندگان تجهیزات الکترونیکی رغبت ساخت وسایل مورد نیاز این صنعت را پیدا می کنند، و مشترکان نیز می توانند با خیال راحت (بدون نگرانی از اینکه با عوض کردن شرکت تلفن، وسایل را هم باید تعویض کنند) مشترک این سرویسها شوند. بمنظور تدوین استانداردهای LMDS، کمیته 802.16 توسط IEEE تأسیس شد، که این استاندارد را (که شبکه شهری بیسیم نامیده می شود) در آوریل ۲۰۰۲ عرضه کرد.

استاندارد IEEE 802.16 تلفن دیجیتال، دسترسی اینترنت، ارتباط بین شبکه های محلی، ایستگاههای پخش برنامه های رادیویی و تلویزیونی (و چندین کاربرد دیگر) را در بر می گیرد. در فصل ۴ درباره این استاندارد بیشتر صحبت خواهیم کرد.

۴-۵-۲ ترانک ها و مالتی پلکس کردن

ساخت و ساز نقش مهمی در اقتصاد سیستمهای تلفن بازی می کند. نصب یک خط اصلی با پهنای باند زیاد (high-bandwidth trunk) بین دو مرکز سوییچینگ به همان اندازه یک خط اصلی با پهنای باند کم (low-bandwidth trunk) پول لازم دارد، چون هزینه اصلی در اینجا هزینه حفر کانالهاست نه هزینه کابل مسی یا فیبر نوری. به همین دلیل، شرکت های تلفن از روشهای پیچیده ای برای ارسال همزمان چندین مکالمه روی یک خط فیزیکی استفاده می کنند، که به این روشها مالتی پلکس (multiplex) گفته می شود. تکنیکهای مالتی پلکس کردن به دسته بزرگ تقسیم می شوند: FDM (مالتی پلکس تقسیم فرکانس - Frequency Division Multiplexing) و TDM (مالتی پلکس تقسیم زمان - Time Division Multiplexing). در FDM، طیف فرکانسی به باندهای مختلفی تقسیم می شود، و هر کاربر انحصاراً از یک باند استفاده می کند. در TDM، هر کاربر برای لحظه ای کوتاه (تورش کوچکی از زمان) کل پهنای باند را در اختیار می گیرد.

در رادیوهای AM هر دو نوع مالتی پلکس دیده می شود. طیف تخصیص داده شده به هر کانال (ایستگاه) در حدود 1 MHz (تقریباً بین 500-1500 kHz) است، و هر ایستگاه در باند اختصاصی خود کار می کند. بین ایستگاهها نیز آنقدر فاصله وجود دارد، که با هم تداخل نکنند. این سیستم نمونه ای از FDM است. علاوه بر آن، در برخی از کشورها هر ایستگاه دارای دو زیرکانال منطقی است: کانال موزیک و کانال آگهی. فرستنده در برشهای کوتاه زمانی (و بصورت یک در میان) موزیک و آگهی پخش می کند، که در این حالت بصورت TDM کار می کند. در ادامه، ابتدا مالتی پلکس تقسیم فرکانس (و کاربرد آن در فیبرهای نوری، که به مالتی پلکس تقسیم طول موج معروفست) را خواهید دید. سپس با مالتی پلکس تقسیم زمان (و یکی از کاربردهای پیشرفته آن در فیبرهای نوری، بنام SONET) آشنا می شوید.

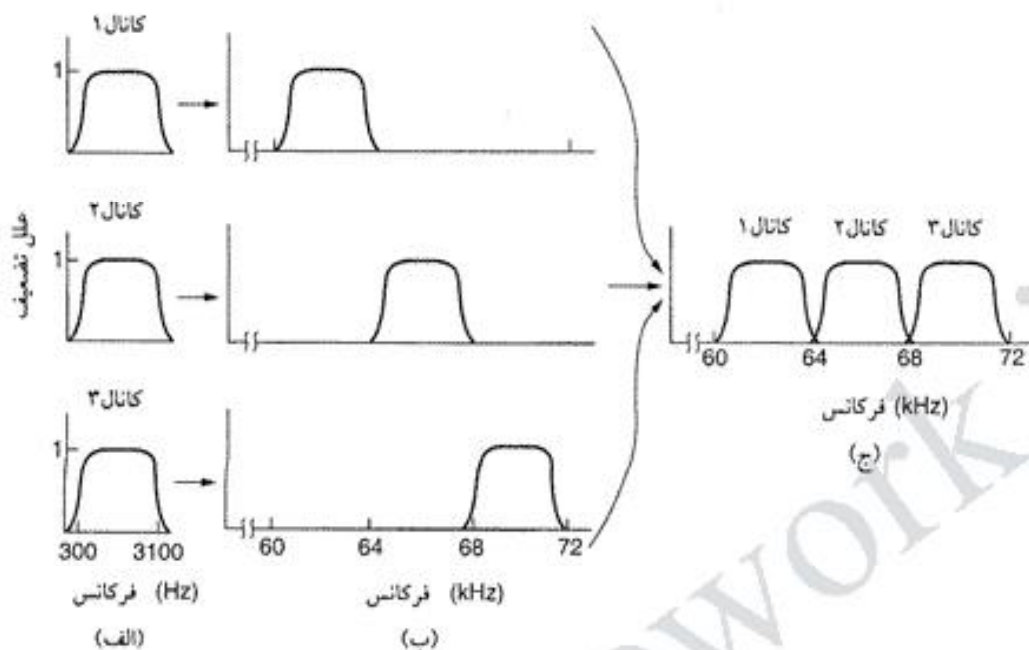
مالتی پلکس تقسیم فرکانس

در شکل ۲-۳۱ مالتی پلکس ۳ کانال صوتی با استفاده از FDM نشان داده شده است. پهنای باند هر کانال با استفاده از فیلترهای مخصوص به 3100 Hz محدود شده، ولی هنگام مالتی پلکس کردن پهنای هر کانال 4000 Hz در نظر گرفته می شود تا بین آنها تداخل پیش نیاید. در اولین قدم، فرکانس هر کانال به مقدار مشخصی (که با سایر کانالها متفاوت است) بالا برده می شود. سپس می توان این کانالها را با هم ترکیب کرد، چون دیگر هیچکدام از آنها فرکانس یکسانی ندارند و با هم مخلوط نمی شوند. دقت کنید که با وجود در نظر گرفتن یک حاشیه امنیتی برای هر کانال، آنها تا حدی روی هم می افتند، چون نقطه قطع فیلترها کاملاً تیز نیست. این روی هم افتادگی باعث بروز نوعی نویز غیرحرارتی در کانالهای مجاور می شود.

روشهای FDM که در سرتاسر دنیا بکار برده می شوند، تا حدی استاندارد شده اند. یکی از این استانداردها مالتی پلکس کردن دوازده کانال صوتی 4000-Hz روی باند 60-108 kHz است. این واحد گروه (group) نامیده می شود. گاهی اوقات از باند 12-60 kHz نیز بعنوان یک گروه استفاده می شود. بسیاری از شرکت های تلفن سرویسهای خطوط اجاره ای 48 kbps تا 56 kbps را بصورت همین گروه ها به مشتریان خود ارائه می کنند. از مالتی پلکس پنج گروه (یعنی ۶۰ کانال صوتی) یک فوق گروه (supergroup) بوجود می آید؛ و از مالتی پلکس پنج (در استاندارد CCITT) یا ده (در استاندارد شرکت پل) فوق گروه یک ابرگروه (mastergroup) شکل می گیرد. در برخی از استانداردها حتی تا ۲۳۰,۰۰۰ کانال صوتی در یک باند مالتی پلکس می شود.

مالتی پلکس تقسیم طول موج

برای کانالهای فیبر نوری از نوع دیگری از مالتی پلکس تقسیم فرکانس استفاده می شود، که مالتی پلکس تقسیم



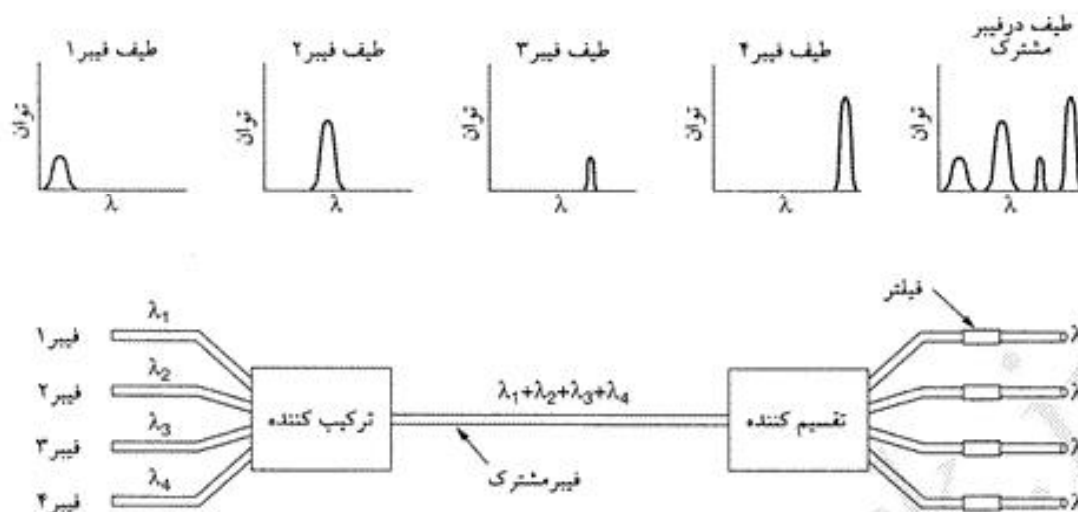
شکل ۲-۳۱. مالتی پلکس تقسیم فرکانس. (الف) کانالهای اولیه. (ب) بالا بردن فرکانس کانالها. (ج) کانالهای مالتی پلکس شده.

طول موج (Wavelength Division Multiplexing) نام دارد. اصول WDM در شکل ۲-۳۲ نشان داده شده است. در اینجا پرتوهای چهار فیبر نوری، که هر کدام طول موج متفاوتی دارند، وارد یک ترکیب کننده نوری شده، و برای ارسال آماده می شوند. در انتهای دیگر نیز عمل عکس انجام شده، و پرتوها مجدداً تفکیک می شوند. این کار توسط فیلترهای مخصوصی که فقط به یک طول موج اجازه عبور می دهند، انجام می شود.

در واقع هیچ چیز جدیدی در این تکنیک وجود ندارد، و این یک مالتی پلکس تقسیم فرکانس در فرکانسهای بسیار بالاست. مادامیکه هر کانال دارای فرکانس (یا، طول موج) خاص خود باشد، می توان آنها را با هم ترکیب (مالتی پلکس) کرد. تنها فرق این روش با FDM الکتریکی اینست که در سیستمهای نوری از فیلترهای انکساری منفعل استفاده می شود، که بسیار قابل اطمینان هستند.

سرعت پیشرفت تکنولوژی WDM آنقدر زیاد است که صنعت کامپیوتر به گرد آن هم نمی رسد. این تکنولوژی در سال ۱۹۹۰ اختراع شد، و اولین سیستم تجاری WDM دارای هشت کانال 2.5 Gbps بود. در سال ۱۹۹۸، سیستمهایی با ۴۰ کانال 2.5 Gbps به بازار عرضه شد، و در سال ۲۰۰۱ شاهد سیستمهایی با ۹۶ کانال 10 Gbps بودیم، که ظرفیت کل آنها به 960 Gbps می رسد (این پهنای باند برای ارسال ۳۰ فیلم کامل - با فرمت MPEG-2 - در هر ثانیه کافیت). در آزمایشگاهها سیستمهایی با ۲۰۰ کانال نیز تست شده اند. وقتی تعداد کانالها خیلی زیاد باشد، و طول موجها فاصله کمی (در حد 0.1 nm) با هم داشته باشند، به آن DWDM (Wavelength Division Multiplexing) چگال - Dense WDM گفته می گویند.

علت محبوبیت سیستمهای WDM اینست که انرژی هر فیبر نوری فقط چند گیگاهرتز پهنای دارد. چون در حال حاضر امکان تبدیل سریعتر سیگنالهای الکتریکی به پالسهای نوری (و بالعکس) وجود ندارد. با ترکیب چند کانال با طول موجهای مختلف، پهنای باند بصورت خطی افزایش پیدا می کند. از آنجائیکه پهنای باند یک فیبر نوری در حدود 25,000 GHz است (شکل ۲-۶ را ببینید)، ظرفیت آن می تواند به ۲۵۰۰۰ کانال 10-GHz برسد - و این



شکل ۲-۳۲. مالتی پلکس تقسیم طول موج.

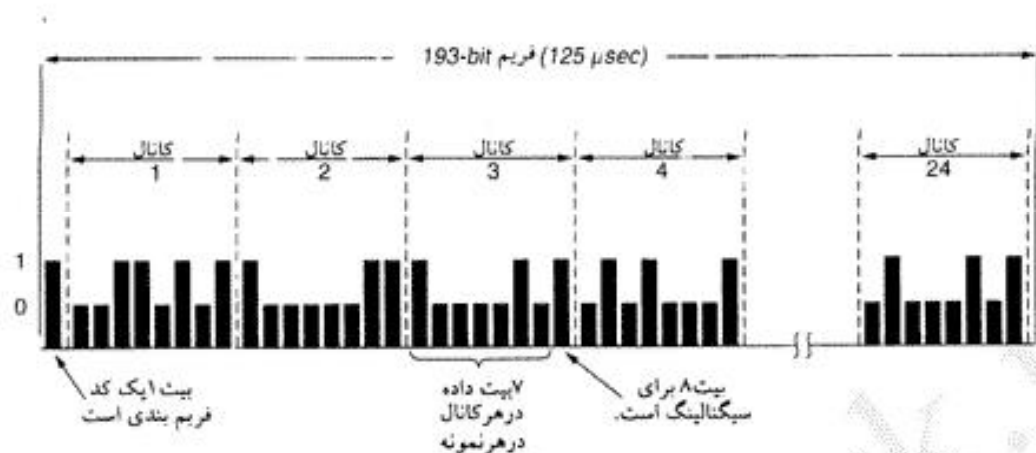
در 1 bit/Hz است، که البته نرخهای بالاتر هم امکان دارد. پیشرفت جدیدی که در این زمینه بدست آمده، تقویت کننده های تمام نوری است. در گذشته، لازم بود تا در فواصل ۱۰۰ کیلومتری پالسهای نوری به سیگنالهای الکتریکی تبدیل شده، و پس از تقویت دوباره به پالسهای نوری تبدیل و ارسال شوند. امروزه، تقویت کننده های تمام نوری می توانند (بدون نیاز به تبدیل کننده های نوری-الکتریکی) عمل تقویت را در هر ۱۰۰۰ کیلومتر انجام دهند.

مثال شکل ۲-۳۲ یک سیستم ثابت است: بیت های فیبر ۱ به خروجی ۳ می روند، بیت های فیبر ۲ به خروجی ۱ می روند، و الی آخر. ولی، ساخت سیستم های سوییچینگ WDM نیز امکان پذیر است. در این قبیل دستگاهها، می توان فیلترهای خروجی را با استفاده از تداخل سنجهای فابری-پروت یا ماخ-زندر تنظیم کرد. برای کسب اطلاعات بیشتر در زمینه WDM و کاربرد آن در سوییچینگ بسته اینترنت، به (Elmirghani and Mouftah, 2000; Hunter and Andonovic, 2000; Listani et al., 2001) مراجعه کنید.

مالتی پلکس تقسیم زمان

تکنولوژی WDM بسیار جالب و مهیج است، ولی هنوز هزاران هزار کیلومتر سیم مسی در شبکه تلفن موجود است که باید فکری هم بحال آنها کرد. با اینکه FDM هنوز کاربرد گسترده ای در کابلهای مسی و کانالهای مایکروویو دارد، اما این تکنولوژی اساساً آنالوگ است و نمی توان آنرا با کامپیوتر انجام داد. ولی TDM (مالتی پلکس تقسیم زمانی - Time Division Multiplexing) را می توان بطور کامل دیجیتالی کرد، و به همین دلیل در سالهای اخیر کاربرد گسترده ای یافته است. متأسفانه، از TDM فقط برای داده های دیجیتال می توان استفاده کرد، و از آنجائیکه مدارهای پایانی سیگنالهای آنالوگ تولید می کنند، باید این سیگنالها را در ایستگاه پایانی به دیجیتال تبدیل کرد، و سپس روی ترانکها (که دیجیتال هستند) ارسال کرد.

در این قسمت نحوه دیجیتالی کردن سیگنالهای آنالوگ صدا، و ترکیب آنها برای ارسال روی ترانکهای دیجیتال را توضیح می دهیم - سیگنالهایی که کامپیوترها از طریق مودم ارسال می کنند نیز آنالوگ است، بنابراین توضیحات زیر در مورد آنها هم صادق است. سیگنالهای آنالوگ در ایستگاه پایانی توسط دستگاهی بنام کدیک (codec) دیجیتال شده، و یکسری اعداد ۸ بیتی تولید می شود. این کدک در هر ثانیه ۸۰۰۰ نمونه می گیرد (در هر



شکل ۲-۲۳. کاربرد T1 (1.544 Mbps).

۱۲۵ میکروثانیه یک نمونه)، چون طبق قضیه نایکوئیست این مقدار برای گرفتن تمام اطلاعات کانالی با پهنای ماند 4-kHz کافیست - با نرخ نمونه برداری پائینتر اطلاعات از دست می رود، و با نرخ بالاتر اطلاعات بیشتری بدست نمی آید. این تکنیک که PCM (مدولاسیون کد پالس - Pulse Code Modulation) نامیده می شود، قلب سیستمهای جدید تلفن است. در نتیجه، تمام فواصل زمانی در سیستمهای تلفن مضاربی از 125 μsec هستند. با ورود تکنولوژی مخابرات دیجیتال، CCITT نتوانست بر سر استاندارد بین المللی برای آن به توافق برسد، به همین دلیل سیستمهای دیجیتال متعددی در سراسر دنیا مورد استفاده قرار گرفتند که عمدتاً با هم ناسازگار بودند. تکنیکی که در آمریکای شمالی و ژاپن از آن استفاده شد، کاربرد T1 است که آنرا در شکل ۲-۲۳ ملاحظه می کنید. (بخواهیم دقیقتر صحبت کرده باشیم، این فرمت DS1 و کاربرد آن T1 خوانده می شوند، ولی در اینجا قصد نداریم با آنچه در صنعت جا افتاده مخالفت کنیم) هر کاربرد T1 عبارتست از ۲۴ کانال صوتی که با یکدیگر مالتی پلکس شده اند. معمولاً، این سیگنالهای آنالوگ در فواصل زمانی منظم و متوالی به یک کدک داده شده و دیجیتایز می شوند (بجای آنکه از ۲۴ کدک استفاده کرده، و خروجی آنها را ترکیب کنیم)، هر یک از این ۲۴ کانال بنوبت ۸ بیت به خروجی می دهند. با احتساب ۷ بیت داده و یک بیت کنترل، در هر کانال $7 \times 8000 = 56,000$ bps داده و $1 \times 8000 = 8000$ bps اطلاعات کنترلی خواهیم داشت.

هر فریم دارای $24 \times 8 = 192$ بیت است، که (با احتساب یک بیت اضافی برای فریم بندی) 193 بیت در هر 125 μsec خواهیم داشت، که بدین ترتیب نرخ داده 1.544 Mbps به می رسد. بیت ۱۹۳ ام برای سنکرون کردن فریم مورد استفاده قرار می گیرد. این بیت شکل 0101010101... بخود می گیرد، و گیرنده بکمک این بیت می تواند از سنکرون بودن اطلاعات اطمینان یابد. اگر گیرنده همزمانی خود را با فرستنده از دست دهد، می تواند با جستجوی این طرح بیت دوباره با فرستنده سنکرون شود. کاربران آنالوگ بکلی نمی توانند چنین طرحی از بیت ها تولید کنند، چون این طرح بیت معادل موج سینوسی 4000-Hz است که حذف خواهد شد. کاربران دیجیتال می توانند چنین طرحی را تولید کنند، ولی مشکل اینجاست که این طرح با لغزش فریم نیز می تواند ظاهر شود. برای بازیابی سریعتر سیستم در صورت بروز چنین لغزش هایی، هنگامی که از T1 فقط برای ارسال داده استفاده می شود، کانال ۲۴ ام به یک طرح خاص سنکرون کردن اختصاص می یابد (و فقط در ۲۳ کانال داده ارسال می شود).

و قتر بالاخره CCITT بر سر استانداردهای PCM بتوافق دست یافت، احساس کرد که ۸۰۰۰ بیت کنترلی

خیلی زیاد است. بنابراین استاندارد 1.544-Mbps بر اساس بسته های داده ۸ بیتی بنا شده نه ۷ بیتی؛ عبارت دیگر، هر سیگنال آنالوگ بجای ۱۲۸ سطح به ۲۵۶ سطح مجزا دیجیتایز می شود. دو ویرایش (ناسازگار) از این استاندارد تهیه شد. در سیگنالینگ کانال مشترک (common-channel signaling) بیت اضافی (که در اینجا بجای ابتدا به انتهای فریم ۱۹۳ بیتی چسبانده می شود) در فریمهای فرد مقدار 10101010... بخود می گیرد. و در فریمهای زوج (تمام کانالها) حاوی اطلاعات سیگنالینگ است.

در ویرایش دیگر، سیگنالینگ وابسته به کانال (channel-associated signaling)، هر کانال دارای زیرکانال سیگنالینگ مخصوص بخود است. در هر زیرکانال یک بیت از هر هشت بیت داده در هر شش فریم به سیگنالینگ اختصاص داده می شود. بنابراین از هر شش فریم پنج تای آنها ۸ بیتی و یکی ۷ بیتی است. یکی دیگر از پیشنهادات CCITT، کاربرد PCM با ظرفیت 2.048 Mbps است که E1 نام دارد. این کاربرد دارای ۳۲ بسته ۸ بیتی در هر 125 μsec است، که ۳۰ تای آنها برای داده و دو تا برای سیگنالینگ بکار می روند. در هر گروه چهار فریمی ۶۴ بیت سیگنالینگ وجود دارد، که نیمی از آن به سیگنالینگ وابسته به کانال اختصاص داده شده. و نیمی دیگر برای سنکرون کردن فریمها کنار گذاشته شده است (کشورهای مختلف می توانند از این نیمه بدخواه استفاده کنند). خارج از آمریکای شمالی و ژاپن بجای T1 از کاربرد E1 استفاده می شود.

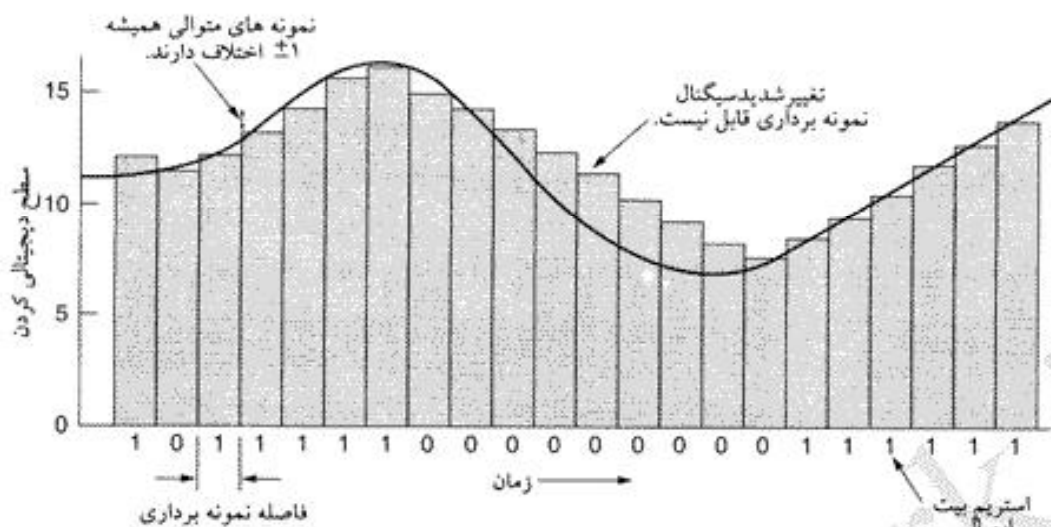
بعد از دیجیتایز کردن سیگنال صوتی، می توان با استفاده از تکنیکهای آماری تعداد بیتهای مورد نیاز هر کانال را کاهش داد (فشرده سازی). از این تکنیکها برای دیجیتایز کردن هر نوع سیگنال آنالوگ می توان استفاده کرد. تمام تکنیکهای فشرده سازی بر این اصل استوارند که تغییرات سیگنال نسبت به فرکانس نمونه برداری نسبتاً کمتر است، بنابراین قسمت اعظم سطوح دیجیتال ۷ یا ۸ بیتی تکراری و زائد هستند.

در یکی از این روشها، که مدولاسیون کد پالس تفاضلی (differential pulse code modulation) نام دارد، نه خود دامنه دیجیتایز شده بلکه تفاوت آن با مقدار قبلی بکار برده می شود. از آنجائیکه در یک مقیاس ۱۲۸ تایی پرشهایی بیشتر از ± 16 ممکن نیست، بجای ۷ بیت به فقط ۵ بیت نیاز داریم. اگر سیگنال در مواردی خاص بیش از ۱۶ سطح اختلاف داشته باشد، برای نمونه برداری آن به بیش از یک دوره زمانی نیاز هست. خطایی که بدین ترتیب روی می دهد، در دیجیتایز کردن صدا قابل چشم پوشی است.

در نوع اصلاح شده این تکنیک فشرده سازی، هر نمونه باید با نمونه قبلی ± 1 اختلاف داشته باشد. در این شرایط بیتی که فرستاده می شود برای آنست که مشخص کند، نمونه جدید بیشتر از قبلی است یا کمتر از آن. این تکنیک را، که مدولاسیون دلتا (delta modulation) نامیده می شود، در شکل ۲-۳۴ ملاحظه می کنید. مانند سایر تکنیکهای فشرده سازی (که فرض را بر تغییرات اندک در سیگنال آنالوگ می گذارند) مدولاسیون دلتا هم در تغییرات شدید سیگنال دچار خطا می شود (که در چنین مواردی اطلاعات از دست می رود).

روش بهبود یافته PCM تفاضلی بر پیش بینی مقدار بعدی سیگنال با استفاده از یرون بایی چند مقدار قبلی، و سپس محاسبه تفاضل این مقدار پیش بینی شده با مقدار واقعی سیگنال استوار است - به این روش کد کردن پیشگویانه (predictive encoding) گفته می شود. البته در این روش فرستنده و گیرنده هر دو باید از روش پیش بینی یکسانی استفاده کنند. روش کد کردن پیشگویانه باعث کم شدن اندازه اعدادی که باید کد شوند، می شود و به همین دلیل تعداد بیتهای ارسالی کاهش خواهد یافت.

تکنیک TDM اجازه می دهد تا چندین کاربرد T1 روی یک مرتبه بالاتر مالتی پلکس شوند - در شکل ۲-۳۵ این تکنیک را ملاحظه می کنید. در اینجا چهار کانال T1 (سمت چپ) روی یک کانال T2 مالتی پلکس شده اند. در کانال T1 مالتی پلکس بصورت بایت به بایت انجام می شود، ولی از T2 به بعد مالتی پلکس بیت به بیت صورت می گیرد. چهار استریم T1 که هر کدام 1.544 Mbps هستند، بایستی خروجی 6.167 Mbps تولید کند.



شکل ۲-۳۴. مدولاسیون دلتا.

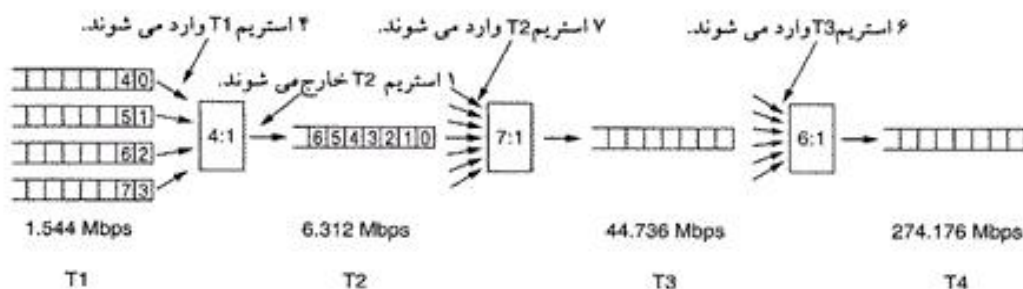
ولی ظرفیت T2 در واقع 6.312 Mbps است. پیت های اضافی برای فریم بندی و بازیابی کانال در صورت لغزش کاربر منظور شده اند. کاربرهای T1 و T3 در میان مشترکین شناخته شده تر هستند، چون T2 و T4 بیشتر برای مصارف داخلی سیستم تلفن بکار می رود.

در مرحله بعد، هفت استریم T2 بصورت بیت به بیت روی یک استریم T3 مالتی پلکس می شوند؛ از ترکیب شش استریم T3 هم یک T4 بوجود می آید. در هر مرحله مقدار کمی سرآیند نیز برای فریم بندی و تصحیح خطای لغزش کاربر اضافه می شود.

همانطور که بر سر کاربر اصلی توافق کمی بین ایالات متحده و سایر کشورهای جهان وجود دارد، نحوه مالتی پلکس آنها نیز استاندارد دقیقی ندارد. کمتر کسی را در دنیا پیدا می کنید که با ضرایب پیشنهادی آمریکا (۴، ۷ و ۶) موافق باشد، به همین دلیل CCITT از مضرب یکنواخت ۴ در تمام مراحل استفاده می کند. همچنین نحوه فریم بندی و بازیابی کانال هم در ایالات متحده و CCITT متفاوت است. در استاندارد پیشنهادی CCITT تعداد کانالها به ترتیب ۳۲، ۱۲۸، ۵۱۲، ۲۰۴۸ و ۸۱۹۲ است، که با سرعت های 2.048 Mbps، 8.848 Mbps، 34.304 Mbps، 139.264 Mbps و 565.148 Mbps کار می کنند.

SONET/SDH

در روزهای اولیه فیبر نوری، هر شرکت تلفن برای خود یک سیستم TDM نوری اختصاصی داشت. بعد از تجزیه



شکل ۲-۳۵. مالتی پلکس استریم T1 روی کاربرهای مرتبه بالاتر.

AT&T در سال ۱۹۸۴، شرکتهای تلفن شهری مجبور شدند به کاربرهای بین شهری مختلف که هر کدام سیستم TDM نوری خاص خود را داشت متصل شوند، به همین دلیل به فکر استاندارد کردن آن افتادند. در ۱۹۸۵، شرکت Bellcore (بازوی تحقیقاتی شرکتهای RBOC) کار روی استاندارد بنام SONET (شبکه نوری سنکرون - Synchronous Optical Network) را شروع کرد. بعدها CCITT نیز وارد این معرکه شد، که نتیجه آن استانداردهای موازی SONET، G.707 و G.709 بود. توصیه های CCITT که SDH (سلسله مراتب دیجیتال سنکرون - Synchronous Digital Hierarchy) خوانده می شوند، فقط چند تفاوت جزئی با SONET دارند. امروزه تقریباً تمامی ترافیک تلفن راه دور در ایالات متحده و اکثر نقاط دنیا، از SONET در لایه فیزیکی ترانک استفاده می کنند. برای کسب اطلاعات بیشتر درباره SONET به (Bellamy, 2000; Goralski, 2000; Shepard, 2001) مراجعه کنید.

در طراحی SONET چهار هدف اصلی مد نظر بوده است. اول و از همه مهمتر، SONET بایستی کاری می کرد که کاربرهای مختلف بتوانند با هم کار کنند. برای رسیدن به این هدف به یک استاندارد سیگنالینگ (در زمینه های طول موج، تایمینگ، ساختار فریم بندی و غیره) نیاز بود.

دوم، SONET باید راهی برای یکپارچه کردن سیستمهای دیجیتال آمریکایی، اروپایی و ژاپنی (که همگی از کانالهای PCM 64-kbps، ولی با روشهای مختلف و ناسازگار، استفاده می کردند) پیدا می کرد.

سوم، SONET باید راهی برای مالتی پلکس کردن کانالهای دیجیتال متعدد پیدا می کرد. در زمان تدوین SONET سریعترین کاربر دیجیتال، که کاربرد گسترده ای در ایالات متحده داشت، T3 با سرعت 44.736 Mbps بود. کاربر T4 تعریف شده، ولی هنوز بطور کامل عملیاتی نشده بود (برای بالاتر از آن هم هنوز کسی فکری نکرده بود). بخشی از مأموریت SONET ادامه راه تا gigabits/sec و بالاتر بود. همچنین باید راهی برای مالتی پلکس کردن کانالهای گندتر در یک کانال SONET پیش بینی می شد.

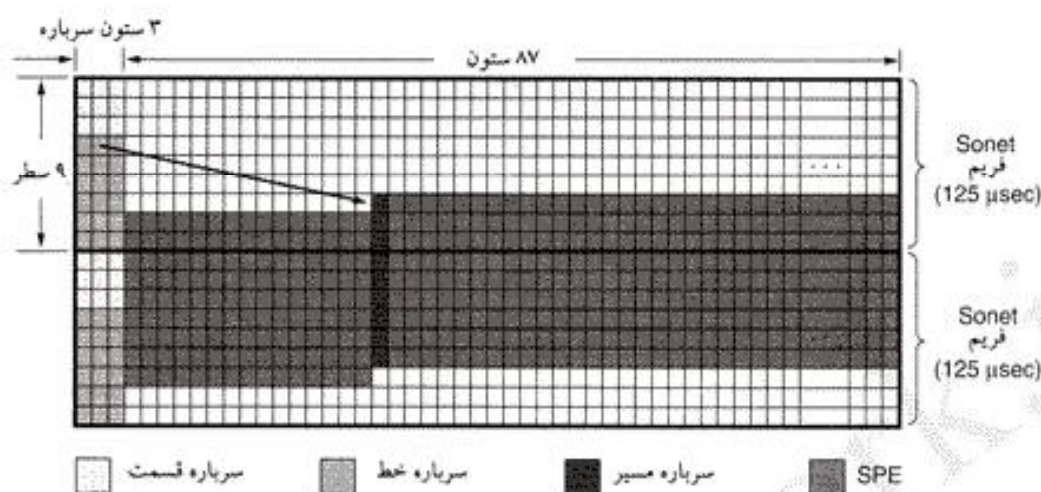
چهارم، SONET باید از یک سیستم جامع عملیات، مدیریت و نگهداری (OAM) پشتیبانی می کرد - وظیفه ای که در سیستمهای قبلی توجه چندانی به آن نشده بود.

از همان ابتدا تصمیم بر آن شد تا SONET یک سیستم ساده TDM باشد، چگونه ای که تمام پهنای باند فیبر نوری یک کانال واحد را تشکیل دهد، و هر زیرکانال برای کسری از ثانیه کانال را در اختیار بگیرد. بدین ترتیب، SONET یک سیستم سنکرون خواهد بود، که توسط یک ساعت اصلی (با دقت 1 در 10^9) کنترل می شود. در یک خط SONET بیت ها با فواصل زمانی فوق العاده دقیق (که توسط ساعت اصلی کنترل می شود) ارسال می شوند. وقتی بعدها سوئیچینگ سلول بعنوان مبنای ATM انتخاب شد، برای تمایز آن با روش سنکرون SONET، از کلمه آسنکرون (Asynchronous Transfer Mode) استفاده شد. در SONET، فرستنده و گیرنده به یک ساعت مشترک وابسته اند، در حالیکه در ATM چنین نیست.

در SONET فریمها 810 بیتی هستند، که در فواصل زمانی $125 \mu\text{sec}$ ارسال می شوند. از آنجائیکه SONET یک سیستم سنکرون است، خواه داده ای باشد یا نباشد، فریمها روی خط فرستاده می شوند. با این نرخ (8000 frames/sec)، SONET کاملاً با کانالهای PCM سازگار خواهد بود.

فریمهای ۸۱۰ بیتی SONET را می توان جدولی با ۹۰ ستون و ۹ سطر قرض کرد. بنابراین در هر ثانیه ۸۰۰۰ فریم $810 \times 8 = 6480$ بیتی ارسال می شود، که سرعت کل آنرا به 51.84 Mbps می رساند. این کانال پایه SONET است، که STS-1 (سیگنال انتقال سنکرون ۱ - Synchronous Transport Signal-1) نامیده می شود. تمام ترانک های SONET مضاربی از STS-1 هستند.

همانطور که در شکل ۲-۳۶ می بینید، سه ستون اول هر فریم به اطلاعات مدیریت سیستم اختصاص یافته اند.



شکل ۲-۳۶. دو فریم متوالی SONET.

در سه سطر اول این بایتهای محتوی سرآیند قسمت (section overhead) هستند، و در شش سطر بعدی محتوی سرآیند خط (line overhead). سرآیند قسمت در ابتدا و انتهای هر قسمت ایجاد و چک می شود، و سرآیند خط در ابتدا و انتهای خط.

فرستنده SONET فریمهای ۸۱۰ بیتی را بصورت متوالی (back-to-back) ارسال می کند، حتی اگر هیچ داده ای در کار نباشد - که در این حالت اطلاعات ساختگی ارسال می شود. از دید گیرنده این یک استریم پیوسته از بایتهاست، پس چگونه می تواند تشخیص دهد ابتدای هر فریم کجاست؟ پاسخ اینست که دو بایت ابتدایی هر فریم طرح خاصی دارند، که گیرنده می تواند آنرا تشخیص دهد. بمحض یافتن این طرح، گیرنده با فرستنده سنکرون می شود. شاید بگویند که احتمال دارد این طرح بیت ها در داده های کاربر نیز وجود داشته باشد، اما از آنجائیکه داده های چندین کاربر در یک فریم مالتی پلکس می شوند (و به دلایل دیگر) این اتفاق نخواهد افتاد.

ستونهای باقیمانده (۸۷ ستون) حاوی داده های کاربر هستند ($87 \times 9 \times 8 \times 8000 = 50.112 \text{ Mbps}$)، که به آنها SPE (بسته کاری سنکرون - Synchronous Payload Envelope) گفته می شود. اما داده های کاربر همیشه از سطر ۱ ستون ۴ شروع نمی شوند - SPE می تواند از هر کجای فریم شروع شود. اشاره گر نقطه شروع SPE در اولین سطر از سرآیند خط نوشته می شود. اولین ستون SPE سرآیند مسیر (path overhead) است - که سرآیند مسیر برای پروتکل های نقطه به نقطه.

با این تمهید (شروع SPE از هر نقطه فریم SONET، و امکان ادامه آن در فریمهای بعدی) انعطاف پذیری سیستم بسیار بالا می رود (شکل ۲-۳۶ را ببینید). برای مثال، اگر پس از شروع یک فریم SONET (و ارسال مقداری اطلاعات ساختگی) داده های واقعی از راه برسند، می توان بجای انتظار تا شروع فریم بعدی، آنها را از همان نقطه وارد فریم کرد.

سلسله مراتب مالتی پلکس SONET را در شکل ۲-۳۷ ملاحظه می کنید؛ نرخها از STS-1 تا STS-192 تعریف شده اند. هر STS- n یک کاربر نوری متناظر بنام OC- n دارد، که بیت به بیت با آن یکسان است (باستثنای یک جابجایی کوچک که برای سنکرون شدن لازم است). نامگذاری در استاندارد SDH متفاوت است و از OC-3 شروع می شود، چون در سیستمهای CCITT نرخی نزدیک به 51.84 Mbps وجود ندارد. علت وجود کاربر OC-9 هم وجود یک خط اصلی پُر سرعت در ژاپن است، که چنین سرعتی دارد. از کاربرهای OC-9 و OC-18

SONET		SDH	سرعت داده (Mbps)		
الکترونیکی	نوری	نوری	ناخالص	SPE	کاربر
STS-1	OC-1		51.84	50.112	49.536
STS-3	OC-3	STM-1	155.52	150.336	148.608
STS-9	OC-9	STM-3	466.56	451.008	445.824
STS-12	OC-12	STM-4	622.08	601.344	594.432
STS-18	OC-18	STM-6	933.12	902.016	891.648
STS-24	OC-24	STM-8	1244.16	1202.688	1188.864
STS-36	OC-36	STM-12	1866.24	1804.032	1783.296
STS-48	OC-48	STM-16	2488.32	2405.376	2377.728
STS-192	OC-192	STM-64	9953.28	9621.504	9510.912

شکل ۲-۳۷. نرخهای مالتی پلکس SONET و SDH.

در ژاپن استفاده می شود. نرخ داده ناخالص شامل تمام سرآیندها می شود؛ در نرخ داده SPE سرآیندهای قسمت و خط حذف شده اند؛ و در نرخ داده کاربر تمام سرآیندها حذف، و فقط ۸۶ ستون محاسبه شده است. اگر یک کاربر مالتی پلکس نشده و فقط شامل داده های یک منبع باشد، یک حرف c (بمعنای پیوسته) به انتهای نام آن اضافه می شود. برای مثال، OC-3 نشان دهنده یک کاربر 155.52-Mbps است که از مالتی پلکس شدن سه خط OC-1 حاصل شده، ولی OC-3c استریمی است از یک منبع واحد با نرخ 155.52 Mbps. سه استریم OC-1 در یک استریم OC-3c بصورت ستون-در-میان چیده می شوند: ستون ۱ از استریم ۱، سپس ستون ۱ از استریم ۲، سپس ستون ۱ از استریم ۳، بدینال آن ستون ۲ از استریم ۱، والی آخر - که بدین ترتیب فریمی مرکب از ۲۷ ستون و ۹ سطر بوجود می آید.

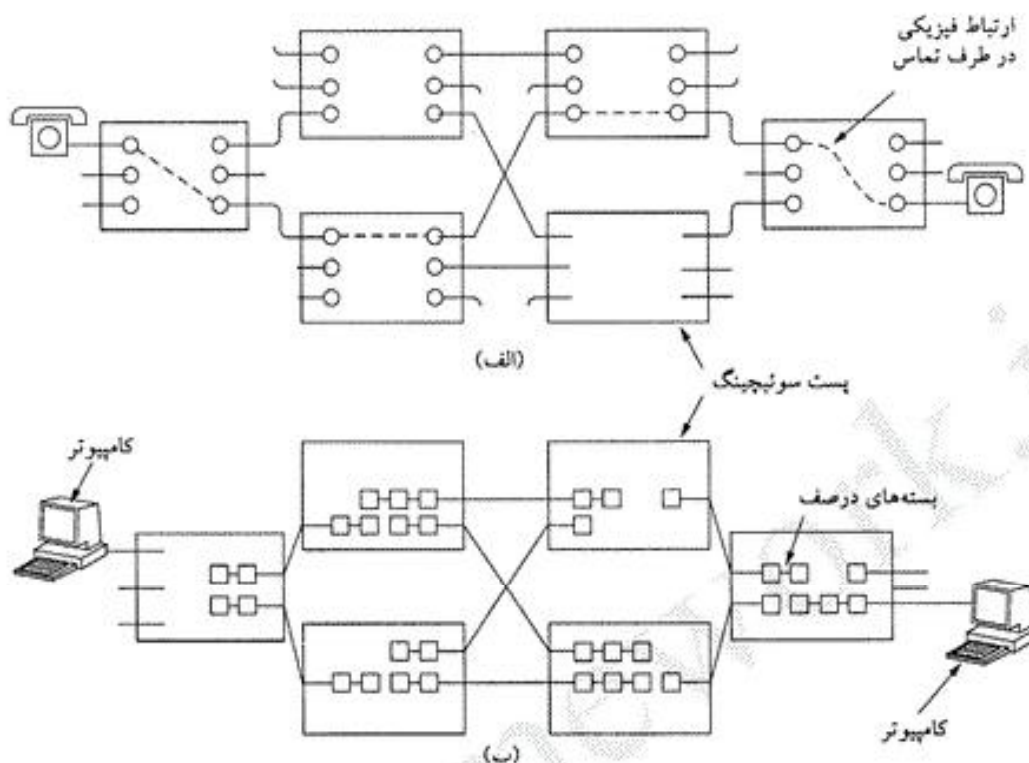
۵-۵-۲ سوئیچینگ

از دید اکثر مهندسان مخابرات، شبکه تلفن دو بخش عمده دارد: خارجی (مدارهای پایانی و ترانک ها - تجهیزاتی که در فضای باز و بیرون مرکز تلفن نصب می شوند) و داخلی (سوئیچها - تجهیزاتی که در فضای بسته و داخل مرکز تلفن نصب می شوند). تا اینجا با بخش خارجی آشنا شدیم؛ اکنون وقت آنست که نگاهی به داخل مرکز تلفن بیندازیم.

امروزه از دو تکنیک متفاوت سوئیچینگ استفاده می شود: سوئیچینگ مداری (circuit switching) و سوئیچینگ بسته ای (packet switching). ابتدا هر دو تکنیک را مختصراً معرفی می کنیم، و سپس مفصلاً به سوئیچینگ مداری (که تکنیک اصلی شبکه تلفن است) می پردازیم. در فصلهای آینده درباره سوئیچینگ بسته ای بیشتر صحبت خواهیم کرد.

سوئیچینگ مداری

وقتی یک تماس تلفنی می گیرید، دستگاههای سوئیچینگ سیستم تلفن در صدد یافتن یک مسیر فیزیکی بین شما و تلفن مقصد بر می آیند. به این تکنیک که آنرا در شکل ۲-۳۸ (الف) ملاحظه می کنید، سوئیچینگ مداری گفته می شود. هر یک از مستطیلهایی که در این شکل می بینید، یک مرکز سوئیچینگ (شهری، بین شهری، و غیره) است. در این مثال، هر مرکز سوئیچینگ سه خط ورودی و سه خط خروجی دارد. وقتی تماس تلفنی از یکی از این مراکز سوئیچینگ می گذرد، یک ارتباط فیزیکی بین آن خط ورودی و یکی از خطوط خروجی برقرار می شود، که در این



شکل ۲-۳۸. (الف) سوئیچینگ مداری. (ب) سوئیچینگ بسته ای.

شکل با خط چین نشان داده شده است.

در روزهای اولیه تلفن، این ارتباط توسط اپراتور و یکمک یک سیم فنی که پریز ورودی را به خروجی متصل می کرد، انجام می شد. اختراع دستگاه سوئیچینگ خودکار تلفن داستان جالبی دارد: این دستگاه در قرن نوزدهم در ایالت میسوری توسط فردی بنام آلمون ب. استراوگر، که شغل وی کفن و دفن بود، اختراع شد. در آن روزها وقتی کسی می مرد، یکی از بازماندگان وی با اپراتور تلفن شهر تماس می گرفت و می گفت، «لطفاً مرا به یک مؤسسه کفن و دفن وصل کنید». در شهر آقای استراوگر دو مؤسسه کفن و دفن وجود داشت، و از شانس بد این آقای اپراتور تلفن همسر رقیب بود. آقای استراوگر خیلی زود دریافت که اگر می خواهد ورشکست نشود، باید یک دستگاه سوئیچینگ خودکار تلفن اختراع کند - و این کار را کرد. همه آنهايي که در سراسر دنیا با دستگاههای سوئیچینگ خودکار تلفن سروکار دارند، آنها را با نام «دستگاه استراوگر» می شناسند. (تاریخ نمی گوید آیا این خانم بعد از بیکار شدن توانست شغلی مانند اپراتور اطلاعات تلفن - که باید به سؤالاتی از قبیل «لطفاً شماره یک مؤسسه کفن و دفن را بدهید» پاسخ دهد - بدست آورد یا خیر؟)

البته شکل ۲-۳۸ (الف) بسیار ساده شده است، چون مسیر فیزیکی بین دو تلفن می تواند از لبتکهای مایکروویو یا فیبر نوری (که هزاران تماس تلفنی روی آنها مالتی پلکس می شود) عبور کند. با این حال مفهوم کلی آن همچنان معتبر است: وقتی تماس تلفنی برقرار می شود، یک مسیر فیزیکی بین دو دستگاه تلفن بوجود می آید که تا آخر تماس باقی می ماند.

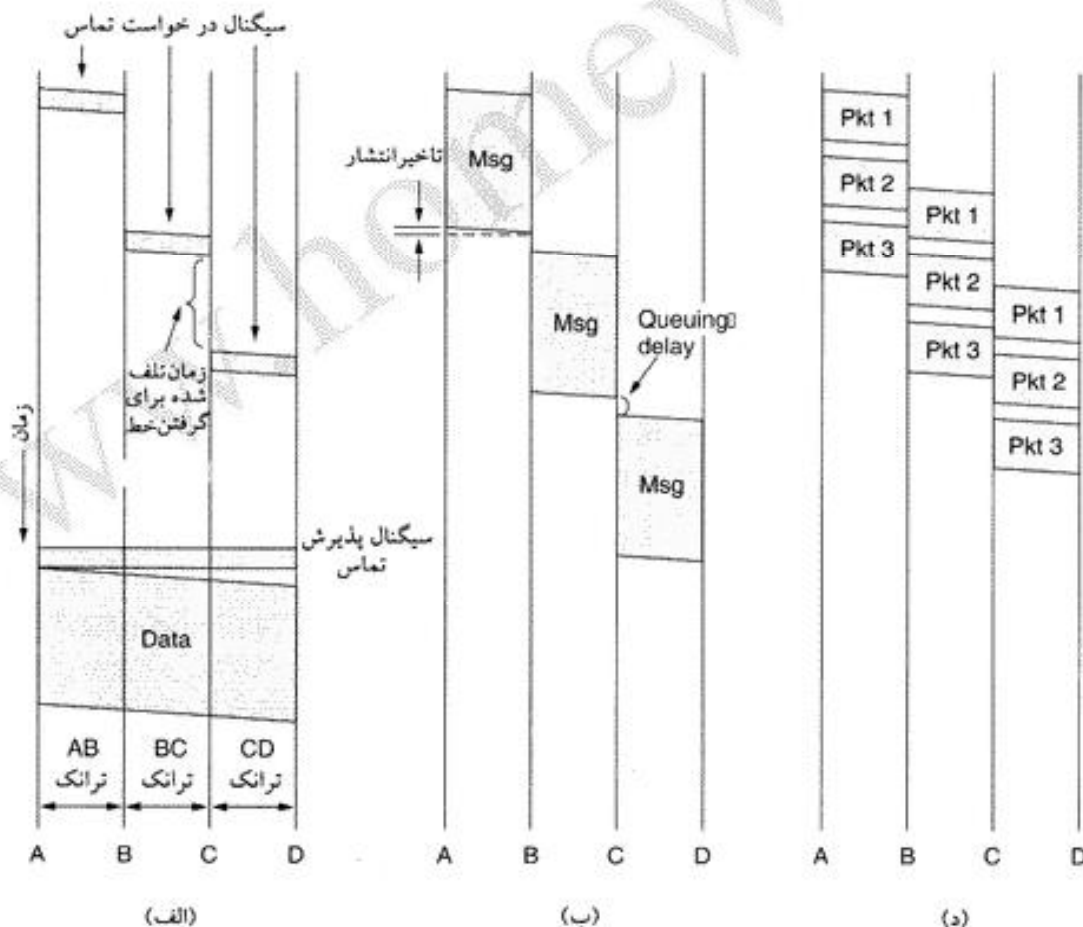
در شکل ۲-۳۸ (ب) روش جایگزین سوئیچینگ مداری، که سوئیچینگ بسته ای نام دارد، را می بینید. در این تکنولوژی هر بسته مستقلاً فرستاده می شود، بدون آنکه از قبل مسیری به آن اختصاص داده شده باشد؛ این بر عهده هر بسته است که راه خود را به سمت مقصد پیدا کند.

مهمترین ویژگی سوئیچینگ بسته‌ای اینست که قبل از ارسال هر گونه داده‌ای، بایستی یک مسیر نقطه-به-نقطه بین مبدأ و مقصد برقرار شده باشد. فاصله زمانی بین شماره‌گیری در مبدأ و زنگ خوردن تلفن مقصد بر راحتی می‌تواند به ۱۰ ثانیه برسد (که این زمان در تماسهای راه‌دور و بین‌المللی حتی بیشتر است). در این مدت سیستم تلفن بدنبال یافتن یک مسیر مناسب است (شکل ۲-۳۸ الف) را ببینید). توجه داشته باشید که قبل از شروع ارسال داده، سیگنال درخواست (request) باید تمام مسیر را تا مقصد طی کرده، و تصدیق (acknowledgement) آن بازگردد. این تأخیر در بسیاری از کاربردهای کامپیوتری (مانند بررسی اعتبار مشتری در خریدهای اعتباری) قابل قبول نیست.

اما بمحض آنکه مدار برقرار شد، دیگر تأخیر چندانی بین گیرنده و فرستنده وجود ندارد (فقط زمان تأخیر انتشار امواج الکترومغناطیس، که آن هم چیزی در حدود $5 \mu\text{sec/km}$ است). دیگر اینکه (بدلیل وجود مدار اختصاصی)، در سوئیچینگ مداری پدیده‌ای بنام ازدحام (congestion) وجود ندارد (البته بدلیل اینکه ظرفیت خطوط و مراکز سوئیچینگ نامحدود نیست، قبل از برقراری مدار همیشه احتمال شنیدن بوق اشغال وجود دارد).

سوئیچینگ پیام

یکی دیگر از اشکال سوئیچینگ، که آنرا در شکل ۲-۳۹ (ب) ملاحظه می‌کنید، سوئیچینگ پیام



شکل ۲-۳۹. همزمانی رویدادها در (الف) سوئیچینگ مداری، (ب) سوئیچینگ پیام،

(ج) سوئیچینگ بسته‌ای.

(message switching) است. در این نوع از سوییچینگ نیز مسیر فیزیکی ثابتی بین فرستنده و گیرنده وجود ندارد. وقتی فرستنده یک بلوک از داده ها را ارسال می کند، این داده ها در اولین مرکز سوییچینگ (که همان راوتر است) ذخیره شده (store)، و سپس به مرکز بعدی هدایت (forward) می شود. هر بلوک ابتدا بطور کامل دریافت شده، از نظر خطا بررسی و سپس ارسال می شود. همانطور که در فصل قبل هم گفتیم، به شبکه ای که با این روش کار می کند شبکه ذخیره-هدایت (store-and-forward) گفته می شود.

اولین تجهیزات مخابرات الکترومغناطیسی تجهیزات سوییچینگ پیام بودند، که برای ارسال تلگرام بکار گرفته شدند. پیام ابتدا در اداره تلگراف روی نوارهای کاغذی سوراخ می شد، و بعد از خوانده شدن توسط دستگاههای خاص به مرکز بعدی فرستاده می شد، که در آنجا بصورت یک نوار کاغذی سوراخ شده از دستگاه بیرون می آمد. اپراتور مسئول دستگاه کاغذ را پاره کرده، و در یک دستگاه نوارخوان (tape reader) می گذاشت تا بتواند پیام را بخواند (هر خط مخابراتی یک دستگاه نوارخوان داشت). این هم در واقع نوعی سوییچینگ است، که به سوییچینگ نوار پاره (torn tape switching) معروف بود. نوارهای کاغذی و سوییچینگ پیام مدتهاست که از دور خارج شده اند، و ما هم بیش از این درباره آنها صحبت نخواهیم کرد.

سوییچینگ بسته ای

اندازه پیام در شبکه سوییچینگ پیام هیچ محدودیتی ندارد، و این به آن معناست که دستگاههای مسیریاب برای نگهداری پیامها به وسایل ذخیره سازی (از قبیل، دیسک) نیاز دارند. پیامد دیگر این روش آنست که یک پیام واحد می تواند آنقدر بزرگ باشد که برای دقایقی خط مسیریاب-مسیریاب را اشغال کند، که بدین ترتیب کاربرد سوییچینگ پیام را در ارتباطات تعاملی (interactive) بشدت محدود می کند. برای حل این مشکل، سوییچینگ بسته ای (packet switching) اختراع شد. در شبکه های سوییچینگ بسته ای روی اندازه بسته ها محدودیت شدیدی اعمال می شود، و به همین دلیل مسیریاب ها نیازی به دیسک برای ذخیره کردن بسته ها ندارند، و می توانند آنها در در حافظه اصلی خود ذخیره کنند. با محدود کردن اندازه بسته ها، و اطمینان از اینکه یک کاربر نمی تواند خط انتقال را برای مدتی طولانی - که البته در اینجا منظور از طولانی بیش از چند میلی ثانیه است - به انحصار خود در آورد، شبکه های سوییچینگ بسته ای برای کاربردهای تعاملی بسیار مناسبند. با مقایسه شکل های ۲-۳۹ (ب) و (ج) یکی دیگر از مزایای روش سوییچینگ بسته ای بر سوییچینگ پیام را مشاهده می کنید: بسته اول یک پیام چندبسته ای می تواند حتی قبل از رسیدن بسته دوم به مسیریاب بعدی فرستاده شود، که این زمان تأخیر را کاهش داده و کارایی سیستم را بالا می برد. به دلایل فوق، شبکه های کامپیوتری اغلب از سوییچینگ بسته ای استفاده می کنند؛ سوییچینگ مداری نیز در موارد خاصی بکار برده می شود، ولی سوییچینگ پیام هیچ کاربردی در شبکه های کامپیوتری ندارد.

سوییچینگ مداری و سوییچینگ بسته ای تفاوت های زیادی با یکدیگر دارند. برای مثال، در یک شبکه سوییچینگ مداری قبل از ارسال اطلاعات بایستی مدار فیزیکی بین فرستنده و گیرنده برقرار شده باشد، در حالیکه در شبکه های سوییچینگ بسته ای چنین الزامی وجود ندارد و ارسال بسته ها می تواند بلافاصله شروع شود.

پیامد لزوم برقراری مدار ثابت در سوییچینگ مداری، اختصاص پهنای باند در تمام طول مسیر بین فرستنده و گیرنده است: تمام بسته ها باید از این مسیر عبور کنند. از طرف دیگر وقتی تمام بسته ها مجبور به عبور از یک مسیر باشند، نمی توانند خارج از ترتیبی که ارسال شده اند، به مقصد برسند. در سوییچینگ بسته ای هیچ مسیر ثابتی وجود ندارد، و بسته ها می توانند از هر مسیری که (در آن لحظه خاص) در شبکه موجود است عبور کنند، و حتی خارج از نظم و ترتیب اولیه به مقصد برسند.

ویژگی تحمل خطا در شبکه های سونیچینگ بسته ای بسیار بهتر از شبکه های سونیچینگ مداری است - و در واقع دلیل اختراع آن هم همین بوده است. وقتی در شبکه سونیچینگ بسته ای یک مسیر یاب از کار می افتد، بسته ها می توانند از مسیرهای دیگری که وجود دارد، استفاده کرده و مسیر یاب مرده را دور بزنند.

البته وجود یک پهنای باند اختصاصی در شبکه های سونیچینگ مداری این مزیت را دارد که بسته ها بمحض رسیدن به یک مسیر یاب به مسیر یاب بعدی فرستاده می شود، و زمان تأخیر ارسال بشدت کاهش می یابد؛ در حالیکه در شبکه های سونیچینگ بسته ای چنین پهنای باندی اختصاصی وجود ندارد، و بسته ها باید تا رسیدن نوبت ارسال در صف منتظر بمانند.

وجود مدار اختصاصی در شبکه های سونیچینگ مداری بدان معناست که (بعد از برقراری مدار) دیگر حالت ازدحام - انتظار برای باز شدن راه - بروز نخواهد کرد. البته همیشه این احتمال وجود دارد که در شروع ارتباط بدلیل شلوغی شبکه، امکان اختصاص مدار وجود نداشته باشد؛ این نوع دیگری از ازدحام - انتظار برای تخصیص مدار - است.

پهنای باندی که به یک کاربر تخصیص داده می شود، در تمام مدت در اختیار وی است، حتی اگر هیچ چیز برای ارسال نداشته باشد. امکان استفاده از این مدار برای کاربران دیگر وجود ندارد. در شبکه های سونیچینگ بسته ای اتلاف پهنای باند به شکل فوق وجود ندارد، و بهمین دلیل کارایی کلی آن بسیار بهتر است. درک این تفاوت بین سونیچینگ بسته ای و سونیچینگ مداری بسیار مهم است: تفاوت تضمین سرویس به قیمت اتلاف منابع، با استفاده بهینه از منابع به قیمت عدم تضمین سرویس.

سونیچینگ بسته ای از تکنیک ذخیره - هدایت استفاده می کند. در این روش هر بسته قبل از ارسال به مسیر یاب بعدی باید بطور کامل دریافت و در حافظه مسیر یاب ذخیره شود. این روش تأخیر نسبتاً قابل ملاحظه ای ایجاد می کند؛ در حالیکه در سونیچینگ مداری، بیت ها بطور پیوسته روی مدار منتقل می شوند و چنین تأخیری وجود ندارد.

تفاوت دیگر اینست که سونیچینگ مداری بطور کامل شفاف است: فرستنده و گیرنده می توانند از هر نرخ بیت، فرمت، و یا روش فریم بندی که می خواهند استفاده کنند؛ کاربر در این مورد هیچ چیز نمی داند، و به آن اهمیتی هم نمی دهد. اما در سونیچینگ بسته ای این کاربر است که پارامترهای اصلی را تعیین می کند. تفاوت این دو تقریباً مانند جاده و راه آهن است: در جاده این مسافر است که سرعت، اندازه و نوع وسیله نقلیه را انتخاب می کند، در حالیکه در راه آهن انتخاب این پارامترها بر عهده شرکت مسافربری (کاربر) است. همین شفافیت است که به سیستم تلفن اجازه می دهد انواع اطلاعات (صوت، فکس و داده) را منتقل کند.

آخرین تفاوت سونیچینگ مداری و سونیچینگ بسته ای روش محاسبه هزینه است. در سونیچینگ مداری (بدلیل تاریخی) هزینه بر اساس مسافت و مدت محاسبه می شود. در تلفنهای همراه، مسافت (البته بااستثنای مکالمات بین المللی) نقشی ندارد، و مدت مکالمه نیز نقش ناچیزی دارد (برای مثال، تفاوت هزینه مکالمه در روز، شب یا ایام تعطیل). در سونیچینگ بسته ای، اساساً چیزی بنام مدت مکالمه وجود ندارد، و فقط گاهی حجم ترافیک نقشی در هزینه ها بازی می کند. در مصارف خانگی، معمولاً هزینه ها بصورت ماهیانه ثابت اخذ می شود، چون این روش برای ISP ها ساده تر است و کاربران نیز راحت تر با آن کنار می آیند، ولی کاربرهای اصلی شبکه هزینه ها را بر اساس حجم ترافیک دریافت می کنند. تفاوتهایی را که در این قسمت برشمردیم، بصورت خلاصه در شکل ۲-۴۰ ملاحظه می کنید.

سونیچینگ مداری و بسته آنقدر مهم هستند که بزودی دوباره به این مبحث برگشته، و تکنولوژیهای مختلف آنها را به تفصیل بررسی خواهیم کرد.

سوئیچینگ بسته ای	سوئیچینگ مداری	آیتم
لازم ندارد	لازم دارد	برقراری تماس
خیر	بلی	مسیر فیزیکی اختصاصی
خیر	بلی	تمام بسته ها از یک مسیر عبور می کنند
خیر	بلی	بسته ها به ترتیب دریافت می شوند
خیر	بلی	خرابی سوئیچ مرگ آور است
متغیر	ثابت	پهنای باند موجود
در هر بسته	در لحظه شروع	زمان از حاکم احتمالی
خیر	بلی	پهنای باند تلف شده
بلی	خیر	ذخیره - هدایت
خیر	بلی	شفافیت
به ازای هر بسته	در دقیقه	هزینه

شکل ۲-۴۰. مقایسه شبکه های سوئیچینگ مداری و سوئیچینگ بسته ای.

۶-۲ شبکه تلفن همراه

سیستم تلفن معمولی (حتی اگر سرعت آن به دهها گیگاهایت برسد) دسته خاصی از کاربران را هرگز راضی نخواهد کرد: آنهایی که در یک جا بند نمی شوند. امروزه مردم می خواهند از هر جایی که هستند (داخل هواپیما، سوار بر اتومبیل، کنار دریا، نوک قله کوهها و یا از اعماق جنگل) تلفن بزنند - و حتماً تا چند سال دیگر انتظار دارند از این نقاط ایمیل خود را نیز چک کنند، و یا در وب گشت بزنند. پیامد این گرایش رشد چشمگیر تلفنهای غیر ثابت در سالهای اخیر است. در این قسمت قصد داریم مبحث تلفن همراه را بررسی کنیم.

تلفنهای غیر ثابت به دو دسته بزرگ تقسیم می شوند: گوشی بیسیم (cordless phone)، و تلفن همراه (mobile phone) - که گاهی به آن تلفن سلولی (cell phone) نیز گفته می شود. گوشی بیسیم عبارتست از یک دستگاه مرکزی و یک گوشی متحرک، که برای مصارف خانگی (مسافتهای کوتاه) طراحی شده است. این وسیله هرگز کاربردی در شبکه نداشته، و ما هم بیش از این به آن نخواهیم پرداخت. تلفن همراه، که امروزه کاربرد گسترده ای در ارتباطات صدا و داده دارد، موضوع اصلی بحث ماست.

تکامل تلفنهای همراه سه نسل را با تکنولوژیهای متفاوت پشت سر گذاشته است:

۱. صدای آنالوگ
۲. صدای دیجیتال
۳. صدای دیجیتال و داده (اینترنت، ایمیل، و غیره)

با اینکه بیشتر تکنولوژی این سیستمها مورد علاقه ماست، اما جالبست بدانید که سیاست و تصمیمات اقتصادی چه تأثیری بر تکامل این سیستمها دارد. اولین سیستم تلفن همراه در ایالات متحده آمریکا و توسط AT&T اختراع شد، که FCC هم بلافاصله آنرا در تمام کشور اجباری کرد. در نتیجه، تمام ایالات متحده صاحب یک سیستم واحد تلفن همراه (آنالوگ) شد، و تلفنی که در کالیفرنیا خریده شده بود، در نیویورک هم کار می کرد. اما در اروپا اوضاع کاملاً عکس این بود، و هر کشوری سیستم خاص خود را طرحی کرد، که نتیجه آن یک هرج و مرج کامل بود.

اروپا از اشتباه خود درس گرفت و وقتی نوبت تلفن همراه دیجیتال رسید، تمام شرکتهای مخابرات دولتی دور

هم جمع شده و بر سر یک استاندارد واحد (GSM) به توفیق رسیدند، که در نتیجه تلفنهای موبایل اروپایی در تمام نقاط این قاره کار می‌کند. در همان زمان ایالات متحده به این نتیجه رسیده بود که دولت نباید در موضوع استاندارد دخالت کند، و در نتیجه سرنوشت تلفن همراه دیجیتال به بازار سپرده شد. این تصمیم باعث شد تا انواع مختلفی از تلفن همراه دیجیتال تولید و وارد بازار شود - ایالات متحده اکنون دو سیستم بزرگ تلفن همراه دیجیتال (بعلاوه یک سیستم کوچکتر) دارد، که هیچکدام با دیگری سازگار نیست.

اروپا فقط در یک دوره زمانی کوتاه در زمینه تعداد کاربران تلفن همراه از آمریکا عقب بود، ولی اکنون از آن بسیار پیش افتاده است، که یکی از دلایل آن سیستم یکپارچه تلفن همراه در اروپاست، اما دلایل دیگری هم برای آن وجود دارد. تفاوت دیگر سیستم تلفن همراه آمریکا و اروپا (که باعث سرافکنندگی آمریکایی‌هاست) شماره‌های آنهاست. در ایالات متحده شماره‌های تلفن همراه با شماره تلفنهای معمولی (ثابت) مخلوط است. هیچ راهی برای تماس گیرنده وجود ندارد تا تشخیص دهد شماره‌ای که دارد می‌گیرد (مثلاً، 234-5678 (212)، یک شماره معمولی (کم‌هزینه) است یا یک شماره تلفن همراه (با هزینه زیاد). برای عصبی‌تر کردن مردم، شرکت‌های تلفن قرار گذاشته‌اند تا هزینه تماس‌ها را به پای صاحب تلفن همراه بنویسند. بهمین دلیل اغلب مردم در خرید تلفن همراه تردید دارند، و از صورتحسابهای نجومی که ممکنست (در نتیجه تماس دیگران) برای آنها بیاید، می‌ترسند. در اروپا، شماره‌های تلفن همراه گد ناحیه مشخصی دارند (مانند تلفنهای 800 یا 900)، و بسادگی از تلفنهای معمولی قابل تشخیص هستند. هزینه تماس هم مثل تلفنهای معمولی به پای تماس گیرنده نوشته می‌شود (البته با استثنای تماسهای بین‌المللی، که هزینه تماس بین دو طرف تقسیم می‌شود).

دلیل دیگری که باعث پذیرش گسترده تلفن همراه در اروپا شده، ابداع تلفنهای همراه از پیش پرداخت شده (pre-paid) است، که تا ۷۵٪ تلفنهای این قاره را در برخی نقاط شامل می‌شود. این تلفن‌ها را می‌توان در هر فروشگاهی (بدون هیچگونه تشریفات خاص) خرید - فقط پولش را بده و استفاده کن. این تلفن‌ها معمولاً با اعتبار ۲۰ یا ۵۰ یورو (واحد پول اروپا) عرضه می‌شوند، و بعد از صفر شدن اعتبار می‌توان آنها را (با استفاده از یک PIN code سری) دوباره شارژ کرد. امروزه هر نوجوان (و حتی بچه‌ای) در اروپا یک چنین تلفن همراهی دارد، و والدین وی می‌توانند بسادگی محلی او را پیدا کنند (بدون اینکه از صورتحسابهای نجومی آن بترسند). این تلفن‌ها (البته اگر از آنها برای تماس گرفتن استفاده نشود) ماه‌ها دوام می‌آورند، چون نه هزینه ماهیانه دارند نه برای تلفنهایی که به آنها می‌شود، اعتباری کسر می‌شود.

۱.۶-۲ تلفن‌های همراه نسل اول: صدای آنالوگ

صحبت از سیاست و روشهای بازاریابی دیگر پس است؛ اجازه دهید به کار اصلی خود یعنی تکنولوژی بپردازیم. حتی از اوایل قرن بیستم نیز تلفنهای متحرک رادیو-سیسم در زمینه‌های نظامی و مسافرتها دریایی کارایی خود را به اثبات رسانده بودند. در سال ۱۹۴۶، اولین تلفنهای مخصوص اتومبیل در سنت لوئیس راه‌اندازی شدند. در این سیستم آنتنهای (فرستنده-گیرنده) بزرگی روی ساختمانهای بلند نصب شده بود، و ارسال و دریافت روی یک کانال واحد صورت می‌گرفت. صحبت کردن و شنیدن در آن واحد امکان نداشت: برای صحبت کردن باید یک دکمه را فشار می‌دادید، و یا رها کردن آن دیگر نمی‌توانستید حرف بزنید. تا اوایل دهه ۱۹۵۰، این سیستم، که به فشار بده-حرف بزن (push-to-talk) معروف بود، در بسیاری از شهرهای بزرگ آمریکا نصب شد. این سیستم امروزه هم در اتومبیلهای پلیس و تاکسی تلفنی مورد استفاده قرار می‌گیرد.

در دهه ۱۹۶۰، IMTS (سیستم تلفن همراه بهبود یافته - Improved Mobile Telephone System) نصب و راه‌اندازی شد. این سیستم نیز از فرستنده-گیرنده‌های قوی (۲۰۰ وات) در نقاط مرتفع استفاده می‌کرد، ولی برای ارسال و دریافت دو فرکانس متفاوت داشت، که بدین ترتیب نیازی به دکمه «فشار بده-حرف بزن» نبود. علاوه بر

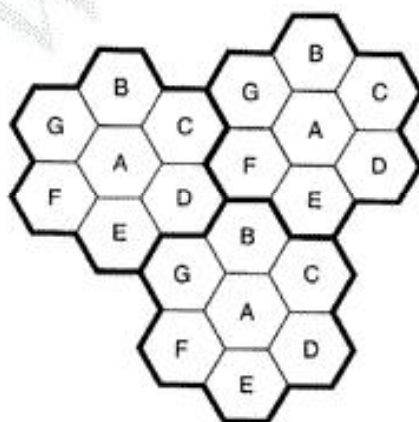
آن، برای ارسال نیز از فرکانسهای متعددی استفاده می شود، و خطر شنیده شدن صحبت دیگران (بر خلاف تلفنهای تاکسی تلفنی) نیز وجود نداشت.

سیستم IMTS از ۲۳ کانال (در طیف 150-450 Mhz) پشتیبانی می کرد. بدلیل کم بودن تعداد این کانالها، کاربران مجبور بودند برای شنیدن بسوق آزاد مدت زیادی صبر کنند. همچنین بدلیل قدرت زیاد فرستنده-گیرنده های این سیستم، تا شعاع چند صد کیلومتری در سیستمهای رادپویی اختلال ایجاد می کردند. همه این دلایل IMTS را به سیستمی غیر عملی تبدیل کرده بود.

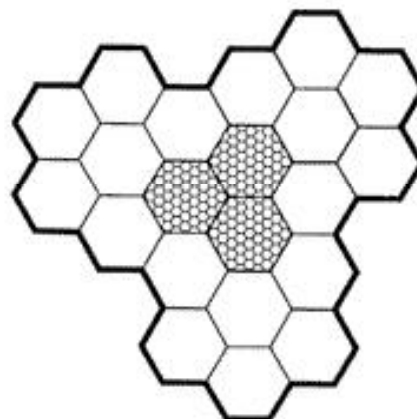
سیستم تلفن همراه پیشرفته

در سال ۱۹۸۲، با اختراع AMPS (سیستم تلفن همراه پیشرفته - Advanced Mobile Phone System) توسط شرکت Bell Labs اوضاع تغییر کرد. این سیستم در انگلستان (با نام TACS) و ژاپن (با نام MCS-L1) نیز نصب و بکار گرفته شد. با آنکه این سیستم دیگر تکنولوژی برتر محسوب نمی شود، اما از آنجائیکه بمنظور سازگاری با گذشته بسیاری از ویژگیهای بنیادی آن در تلفن همراه دیجیتال (DAMPS) لحاظ شده است، قدری درباره آن صحبت خواهیم کرد.

در تمام سیستمهای تلفن همراه، یک منطقه جغرافیایی به تعدادی سلول (cell) تقسیم می شود - که به همین دلیل به آن تلفن سلولی (cell phone) نیز می گویند. در AMPS هر سلول ۱۰ تا ۲۰ کیلومتر قطر دارد؛ در سیستمهای دیجیتالی این سلولها کوچکترند. هر سلول دارای تعدادی فرکانس است، که سلولهای همسایه از آنها استفاده نمی کنند. ایده کلیدی در سیستمهای سلولی که باعث شده تا ظرفیت آنها بسیار بیشتر از سیستمهای قبلی باشد، استفاده از سلولهای نسبتاً کوچک و بکارگیری فرکانسها در سلولهای نزدیک هم (البته، نه سلولهای مجاور) است. در حالیکه یک سیستم IMTS با قطر ۱۰۰ کیلومتر می تواند روی هر فرکانس یک تماس داشته باشد، یک سیستم AMPS قادر است (با جدا کردن فرکانس سلولهای مجاور) روی هر فرکانس ۱۰ تا ۱۵ تماس برقرار کند. هر چه اندازه سلولها کوچکتر (و تعداد آنها بیشتر) باشد، ظرفیت سیستم بالاتر خواهد رفت. با کوچکتر شدن سلولها می توان قدرت فرستنده-گیرنده ها را نیز کمتر کرد، و از تجهیزات ساده تر و ارزاتری استفاده کرد. طبق مقررات FCC، حداکثر قدرت تشعشعی دستگاههای تلفن همراه ۰.۶ وات، و فرستنده-گیرنده های مخصوص اتومبیل ۳ وات تعیین شده است.



(الف)



(ب)

شکل ۲-۴۱. (الف) فرکانسهای سلولهای مجاور یکسان نیستند. (ب) برای افزایش

تعداد کاربران، می توان از سلولهای کوچکتر استفاده کرد.

ایده تکرار فرکانسها را در شکل ۲-۴۱ (الف) ملاحظه می کنید. سلولها تقریباً به شکل دایره هستند، ولی برای سادگی کار بهتر است آنها را شش ضلعی فرض کنیم. در این شکل تمام سلولها هم اندازه اند، و در گروههای هفت تایی دسته بندی شده اند. هر حرف نشان دهنده مجموعه ای از فرکانسها است. توجه کنید که هر مجموعه فرکانسی با نزدیکترین سلول مشابه حداقل و سلول فاصله دارد، که این باعث به حداقل رسیدن تداخل فرکانسها خواهد شد.

یکی از بزرگترین معضلات سیستمهای تلفن همراه، یافتن نقاط مناسب برای نصب آنتنهای مرکزی سلولهاست. این مشکل باعث شده تا شرکتهای تلفن برای استفاده از مناره های رفیع کلیساها دست بدامن آنها شوند. وقتی در یک منطقه تعداد کاربران افزایش می یابد و سیستم دیگر جوابگوی بار مکالمات نیست، (با کم کردن قدرت آنتنها) هر سلول به سلولهای کوچکتر (microcell) تقسیم می شود، تا بتوان از فرکانسها بدفعات بیشتر استفاده کرد - شکل ۲-۴۱ (ب) را ببینید. در مواقعی که تعداد زیادی تلفن همراه برای مدتی کوتاه در یک نقطه گرد می آیند (مانند مسابقات ورزشی و کنسرتها موسیقی)، شرکتهای تلفن با استقرار آنتنهای متحرک (که ارتباط ماهواره ای دارند) موقتاً سلولهای کوچکتری ایجاد می کنند. اینکه اندازه یک سلول چقدر باید باشد، مسئله پیچیده ایست که در (Hac, 1995) درباره آن بحث شده است.

در مرکز هر سلول یک ایستگاه قرار دارد که تمام تلفنهای داخل سلول امواج خود را به آن می فرستند. در سیستمهای کوچک، تمام این ایستگاهها به یک دستگاه مرکزی بنام MTSO (مرکز سوئیچینگ تلفن همراه - Mobile Telephone Switching Office) یا MSC (مرکز سوئیچینگ همراه - Mobile Switching Center) متصل می شوند. در سیستمهای بزرگتر چندین MTSO وجود دارند، که بنوبه خود به یک MTSO بالاتر متصلند. MTSO ها، در واقع، همان ایستگاههای پایانی سیستم تلفن همراه هستند، که به حداقل یک ایستگاه پایانی سیستم تلفن ثابت نیز ارتباط دارند. ارتباط MTSO ها با تلفنهای همراه، با یکدیگر و با مراکز PSTN از طریق یک شبکه سوئیچینگ بسته ای صورت می گیرد.

هر تلفن همراه در هر لحظه در یک سلول (و تحت کنترل ایستگاه مرکزی آن) قرار دارد. وقتی این تلفن سلول را ترک می کند، ایستگاه مرکزی متوجه ضعیف شدن سیگنال آن شده و از تمام ایستگاههای مجاور میزان قدرت دریافتی از این تلفن را می پرسد. سپس، این ایستگاه کنترل تلفن مزبور را به ایستگاهی که بیشترین قدرت را گزارش کرده (و در واقع، تلفن وارد آن شده)، تحویل می دهد. رئیس جدید نیز به تلفن همراه اطلاع می دهد که (اگر مایل به ادامه مکالمه است) کانال خود را عوض کند (چون فرکانس قبلی آن در هیچیک از سلولهای همسایه وجود ندارد). این فرآیند (که به آن پاس کاری - handoff - گفته می شود) تقریباً 300 msec طول می کشد. تخصیص کانال توسط MTSO صورت می گیرد (چون ایستگاههای مرکزی چیزی جز رله های رادیویی ساده نیستند).

پاس کاری می تواند به دو طریق صورت گیرد. در پاس کاری نرم (soft handoff) تلفن همراه قبل از آن که ایستگاه قدیم آنرا قطع کند، به ایستگاه جدید متصل می شود. در این روش کاربر هیچ نوع قطعی احساس نمی کند. اما مشکل این روش آنست که دستگاه تلفن همراه باید بتواند در آن واحد خود را روی دو فرکانس (ایستگاه قبلی و ایستگاه جدید) تنظیم کند. تلفنهای همراه نسل اول و دوم هیچکدام قادر به انجام چنین کاری نیستند.

در پاس کاری سخت (hard handoff) ایستگاه قدیمی قبل از اتصال تلفن همراه به ایستگاه جدید، آنرا قطع می کند. اگر ایستگاه جدید به هر دلیلی (مثلاً، نداشتن باند خالی) نتواند تلفن را تحویل بگیرد، ارتباط کاربر یکباره قطع خواهد شد. این یکی از مشکلات اجتناب ناپذیر تلفنهای همراه نسل اول و دوم است.

کانال ها

سیستم AMPS دارای ۸۳۲ کانال دو طرفه همزمان است، که هر کانال خود از دو کانال یکطرفه ساده تشکیل می شود (۸۳۲ کانال دریافت و ۸۳۲ کانال ارسال). کانالهای یکطرفه دریافت روی فرکانسهای 824-849 MHz و کانالهای یکطرفه ارسال روی فرکانسهای 869-894 MHz کار می کنند (پهنای باند هر کانال 30 kHz است). همانطور که می بینید، AMPS از FDM برای تقسیم پهنای باند استفاده می کند.

امواج در فرکانس 800 MHz طول موجی در حدود 40 cm دارند، و به خط مستقیم حرکت می کنند. درختان و گیاهان این امواج را جذب می کنند، و در برخورد با زمین نیز منعکس می شوند. موجی که به ایستگاه مرکزی سلول می رسد، می تواند مستقیماً از گوشی تلفن همراه آمده باشد، و یا انعکاس آن از سطح زمین یا ساختمانها باشد، که این می تواند باعث پژواک (echo) یا محوشدگی چندمسیره (multipath fading) شود. گاهی حتی امکان شنیدن مکالماتی که در مسافتی طولانی چندین بار به سطح زمین خورده و منعکس شده، نیز وجود دارد.

۸۳۲ کانال AMPS به چهار دسته تقسیم می شوند:

۱. کانالهای کنترل (ایستگاه به تلفن همراه) برای مدیریت سیستم
۲. کانالهای فراخوانی (ایستگاه به تلفن همراه) برای اعلام اینکه مکالمه ای در انتظار کاربر است
۳. کانالهای دسترسی (دو طرفه) برای برقراری مکالمات و تخصیص کانال
۴. کانالهای داده (دو طرفه) برای صدا، فکس، یا داده

تعداد کانالهای کنترل ۲۱ عدد است، که بطور ثابت در PROM تلفنهای همراه نوشته شده است. از آنجائیکه استفاده از فرکانسهای مشابه در سلولهای مجاور مجاز نیست، تعداد کانالهای قابل استفاده در هر سلول بسیار کمتر از ۸۳۲ (و در واقع، چیزی نزدیک به ۴۵ کانال) است.

مدیریت مکالمه

هر تلفن همراه در سیستم AMPS دارای یک شماره سریال ۳۲ بیتی و یک شماره تلفن ۱۰ رقمی است، که در PROM آن نوشته شده است. شماره تلفن همراه دارای یک کد ناحیه ۳ رقمی (که بصورت ۱۰ بیتی کد شده) و یک شماره مشترک ۷ رقمی (که بصورت ۲۴ بیتی کد شده) - مجموعاً ۳۴ بیت - است.

وقتی تلفن روشن می شود، ۲۱ کانال کنترل از پیش برنامه ریزی شده را اسکن می کند، تا قویترین سیگنال را پیدا کند. سپس، شماره سریال ۳۲ بیتی و شماره تلفن ۳۴ بیتی خود را منتشر می کند. تمام اطلاعات کنترلی در AMPS دیجیتال هستند (برخلاف کانالهای صدا، که آنالوگ می باشند)، و این اطلاعات به دفعات و همراه با کدهای تصحیح خطا منتشر می شوند.

وقتی ایستگاه مرکزی سلول این اعلام را دریافت کرد، آنرا به MTSO می فرستد، که آن هم (پس از ثبت کاربر جدید) محل وی را به نزدیکترین MTSO اعلام می کند. در حالت عادی، یک تلفن همراه هر ۱۵ دقیقه خود را به ایستگاه مرکزی معرفی می کند.

برای برقراری یک تماس، کاربر (بعد از روشن کردن تلفن) شماره مورد نظر را وارد کرده، و دکمه CALL را فشار می دهد. تلفن این شماره را به همراه کد شناسایی خود روی یکی از کانالهای دسترسی (access channel) به ایستگاه مرکزی می فرستد. (اگر تداخلی پیش آید، تلفن این عملیات را بعداً تکرار می کند). وقتی ایستگاه مرکزی سلول این درخواست را دریافت کرد، به MTSO اطلاع می دهد. اگر تماس گیرنده یکی از مشترکین آن MTSO باشد، MTSO بدنبال یک کانال خالی می گردد تا به وی تخصیص دهد. اگر کانال خالی موجود بود، شماره آن روی کانال کنترل به تلفن برگشت داده می شود. با گرفتن کانال دسترسی از MTSO، تلفن همراه بطور خودکار به آن

کانال سوئیچ کرده و منتظر می ماند تا طرف مقابل گوشی را بردارد.

تماسهای ورودی به طریق دیگری عمل می کنند. تلفنهایی که بیکار هستند، دائماً به کانال فراخوانی (paging channel) گوش می کنند، تا پیامهایی را که برای آنان می رسد دریافت کنند. وقتی یک شماره تلفن همراه گرفته می شود (خواه از یک تلفن ثابت یا یک تلفن همراه دیگر)، یک بسته به MTSO ی آن تلفن ارسال می شود تا محل وی را پیدا کند. MTSO نیز که محل تمام مشترکین فعال خود را می داند، یک بسته به ایستگاه مرکزی آن تلفن می فرستد، که آن هم بتوی خود یک پیام (با مضمون: «تلفن 14، تو آنجایی؟») روی کانال فراخوانی متشتر می کند. تلفن مقصد هم با ارسال پاسخ "Yes" روی کانال دسترسی جواب می دهد. وقتی ایستگاه مرکزی پاسخ "Yes" را دریافت کرد، با پیامی مانند این عکس العمل نشان می دهد: «تلفن 14، روی کانال 3 به تماس جواب بده.» در اینجا، تلفن همراه به کانال 3 سوئیچ کرده، و شروع به زنگ زدن (یا نواختن یکی از ملودیهای عجیب و غریبی که این روزها همه جا شنیده می شود) می کند.

۲-۶-۲ تلفن های همراه نسل دوم: صدای دیجیتال

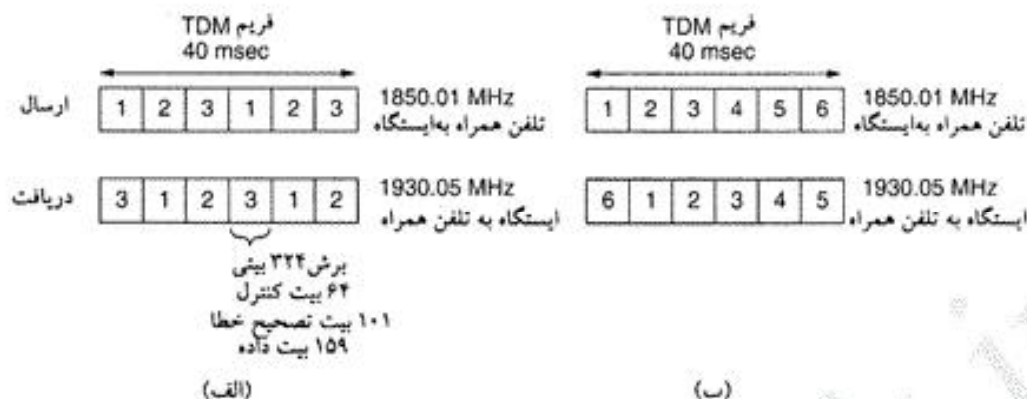
تلفن همراه نسل اول آنالوگ بود، و تلفن همراه نسل دوم دیجیتال. همانطور که نسل اول استاندارد جهانی نداشت، نسل دوم هم چنین استاندارد جهانی ندارد. امروزه چهار سیستم تلفن همراه نسل دوم مشغول بکار هستند: GSM، D-AMPS، CDMA و PDC. در این قسمت سه سیستم اول را بررسی خواهیم کرد؛ PDC فقط در ژاپن راه اندازی شده، و در واقع همان D-AMPS است که برای سازگاری با سیستمهای نسل اول آنالوگ ژاپن تغییراتی در آن صورت گرفته است. اصطلاح PCS (سرویسهای مخابرات شخصی - Personal Communications Services) از لحاظ فنی به معنای تلفن همراه دیجیتال با فرکانس 1900 MHz است، اما امروزه این کلمه عمدتاً به سیستمهای نسل دوم (یعنی، دیجیتال) تعبیر می شود.

D-AMPS : سیستم تلفن همراه پیشرفته دیجیتال

نسل دوم سیستمهای AMPS (که کاملاً دیجیتال است) D-AMPS (سیستم تلفن همراه پیشرفته دیجیتال - Digital Advanced Mobile Phone System) نام دارد. این سیستم تحت استاندارد بین المللی IS-54 (و خلف آن تحت استاندارد IS-136) تعریف شده است. سیستم D-AMPS بگونه ای طراحی شده که با AMPS سازگاری کامل داشته باشد، بهمین دلیل تلفنهای نسل اول و دوم می توانند همزمان در یک سلول کار کنند. D-AMPS همان کانالهای 30 kHz نسل اول (و دقیقاً با همان فرکانسها) استفاده می کند، بنابراین در یک سلول یک کانال می تواند آنالوگ کار کند، در حالیکه کانال مجاور در حال کار بصورت دیجیتال است. این MTSO ی مسئول سلول است که (بر اساس نسبت تلفنهای نسل اول و دوم در هر سلول) تعیین می کند کدام کانالها آنالوگ و کدام کانالها دیجیتال کار کنند - و (با تغییر این نسبت) می تواند ترکیب کانالها را بصورت دینامیک تغییر دهد.

بعد از عملیاتی شدن سرویسهای D-AMPS و به منظور پاسخگویی به تقاضای رو به فزونی بازار، باند فرکانسی جدیدی به آن تخصیص داده شد. کانالهای دریافت از کاربر (upstream) در محدوده 1850-1910 MHz، و کانالهای ارسال به کاربر (downstream) در محدوده 1930-1990 MHz هستند. طول موج در این باند فقط 16 cm است، بنابراین آنتن استاندارد (آنتن یک چهارم طول موج) تلفنهایی که در این باند کار می کنند، فقط 4 cm خواهد بود، و بهمین دلیل این تلفنها می توانند بسیار کوچکتر ساخته شوند. با این حال، اغلب تلفنهای D-AMPS هر دو باند فرکانسی (850 MHz و 1900 MHz) را پشتیبانی می کنند.

در یک تلفن همراه D-AMPS، سیگنال خروجی میکروفون بعد از دیجیتایز شدن با استفاده از مدلهایی که بسیار بهینه تر از مدولاسیون دلتا (delta modulation) و کد کردن پیشگویانه (predictive encoding) هستند،



شکل ۲-۴۲. (الف) یک کانال D-AMPS با سه کاربر. (ب) یک کانال D-AMPS با شش کاربر.

فشرده می شود. این مدل های فشرده سازی با استفاده از خصوصیات دستگاه صوتی انسان، پهنای باند لازم را از 56-kbps به 8 kbps یا کمتر تقلیل می دهند. این کار که توسط دستگاهی بنام vocoder انجام می شود (Bellamy, 2000) را ببینید، در دستگاه تلفن صورت می گیرد نه در ایستگاه مرکزی، که بدین ترتیب تعداد بیت های منتقل شده روی لینک هوایی کاهش خواهد یافت. در تلفن های ثابت این تکنیک هیچ مزیتی بدنبال ندارد، چون کاستن از ترافیک مدار پایانی هیچ تأثیری روی ظرفیت کل سیستم نخواهد داشت.

فشرده کردن صدای دیجیتال تلفن های همراه (همان کاری که در D-AMPS انجام می شود) مزیت فوق العاده ای به همراه دارد، چون می توان با استفاده از حالتی پلکس تقسیم زمانی (TDM) یک زوج فرکانسی (ارسال و دریافت) را بین سه کاربر به اشتراک گذاشت. هر زوج فرکانسی از 25 frame/sec (هر فریم معادل 40 msec) پشتیبانی می کند. هر فریم نیز بنوبه خود به شش بُرش زمانی 6.67 msec تقسیم می شود (شکل ۲-۴۲ (الف) را ببینید).

هر فریم به سه کاربر سرویس (ارسال و دریافت) می دهد، که به نوبت از آن استفاده می کنند. برای مثال در بُرش زمانی ۱ در شکل ۲-۴۲ (الف)، کاربر ۱ می تواند در حال ارسال به ایستگاه مرکزی باشد، در حالیکه در همان زمان کاربر ۳ در حال دریافت است. هر بُرش زمانی طولی معادل 324 بیت دارد، که 64 بیت آن برای مصارف کنترلی اختصاص یافته، و بقیه 260 بیت در اختیار کاربر است. از این 260 بیت، 101 بیت برای گداهای تصحیح خطا (که در لینک های هوایی پرنویز بسیار هم لازمند) بکار می رود، که بدین ترتیب فقط 159 بیت برای انتقال صدا باقی می ماند. با احتساب ۵۰ بُرش زمانی در هر ثانیه، پهنای باند موجود برای صدای فشرده فقط 8 kbps (پهنای باند PCM) است.

با استفاده از الگوریتم های فشرده سازی بهتر، می توان پهنای باند لازم را حتی به 4 kbps کاهش داد، تا شش کاربر بتوانند در آن واحد از یک فریم استفاده کنند (شکل ۲-۴۲ ب). از دید شرکتهای تلفن، توانایی فشرده کردن سه یا شش کاربر D-AMPS در کانالی که فقط یک کاربر AMPS می تواند از آن استفاده کند، یک بُرد واقعی است و دلیل محبوبیت PCM نزد شرکتهای تلفن همراه نیز همین است. البته کیفیت صدای 4 kbps هرگز به پای صدای 56 kbps نمی رسد، اما شرکتهای تلفن همراه هم کمتر روی کیفیت صدای سرویسهای خود تبلیغ می کنند. در مورد سرویسهای داده، کیفیت یک خط 8 kbps حتی با مودمهای 9600-bps نیز قابل مقایسه نیست.

ساختار کنترلی D-AMPS نسبتاً پیچیده است: هر ۱۶ فریم تشکیل یک ابرفریم (superframe) می دهند، که اطلاعات کنترلی بدفعات محدود در این ابرفریم گنجانده می شوند. در کل ۶ کانال کنترلی مورد استفاده قرار

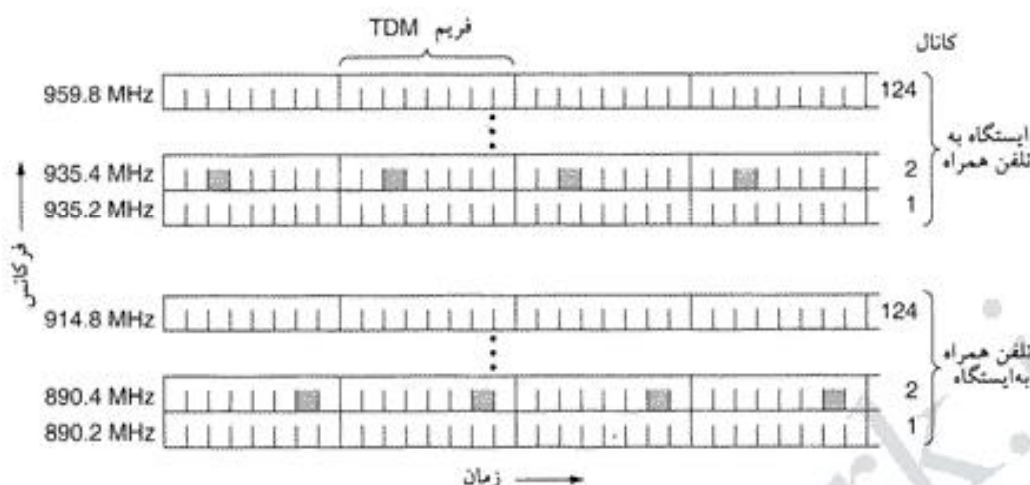
می گیرد: پیکربندی سیستم، کنترل بی درنگ (real-time) و غیر بی درنگ (nonreal-time)، فراخوانی (paging)، پاسخ دسترسی (access response) و پیام کوتاه (short message). طرز کار D-AMPS از نظر مفهومی شبیه AMPS است. وقتی گوشی تلفن همراه روشن می شود، با ایستگاه مرکزی تماس گرفته و بعد از معرفی خود، به کانالهای کنترلی گوش می کند. در این سیستم هم تعیین محل (سلول) تلفن همراه بر عهده MTSO است. یکی از تفاوت های D-AMPS با AMPS در نحوه پاس کاری است. در AMPS این کار بطور کامل بر عهده MTSO است و دستگاه تلفن هیچ دخالتی در آن ندارد. اما همانطور که در شکل ۲-۴۲ دیده می شود، در D-AMPS در $\frac{1}{3}$ از کل زمان تماس دستگاه تلفن هیچ کاری انجام نمی دهد، و می تواند از این زمان خالی برای اندازه گیری کیفیت خط استفاده کند. وقتی تلفن متوجه می شود که قدرت سیگنال در حال کاهش است، به MTSO اطلاع داده و MTSO هم ارتباط را قطع می کند، که در این لحظه تلفن می تواند به یک ایستگاه (فرکانس) قویتر سوییچ کند - به این تکنیک MAHO (پاس کاری یکمک تلفن همراه - Mobile Assisted HandOff) می گویند. در D-AMPS نیز (مانند AMPS) این فرآیند حدوداً 300 msec طول می کشد.

GSM: سیستم سراسری مخابرات همراه

سیستم D-AMPS بصورت گسترده در ایالات متحده (و بصورت اصلاح شده در ژاپن) در حال استفاده است. اما تقریباً در تمام نقاط دیگر دنیا از سیستمی بنام GSM (سیستم سراسری مخابرات همراه - Global System for Mobile Communications) استفاده می کنند، و حتی در آمریکا نیز این سیستم در نقاط محدودی راه اندازی شده است. GSM و D-AMPS از چند نظر شبیه یکدیگرند: هر دوی آنها سیستم های سلولی هستند؛ هر دوی آنها از FDM و فرکانس های متفاوت برای ارسال و دریافت (که فرکانس دریافت تلفن بیشتر از فرکانس ارسال آن است - 80 MHz بالاتر در D-AMPS و 55 MHz بالاتر در GSM) استفاده می کنند؛ و در هر دو سیستم چندین تلفن با استفاده از TDM مشترکاً از یک زوج فرکانس استفاده می کنند. اما در GSM پهنای کانالها بسیار بیشتر از D-AMPS (200 kHz در مقابل 30 kHz) و تعداد کاربران هر کانال نیز نسبتاً کمتر است (8 کاربر در مقابل 3 کاربر)، که در نتیجه نرخ داده آن بسیار بهتر از D-AMPS خواهد بود. در زیر توضیح مختصری درباره ویژگی های اصلی GSM خواهیم داد. اما توجه داشته باشید که استاندارد چاپی GSM متجاوز از ۵۰۰۰ برگ است، و در آن جنبه های فنی و مهندسی این سیستم به تفصیل تشریح شده، که ما در اینجا بد آنها نخواهیم پرداخت.

همانطور که قبلاً هم گفته شد (و در شکل ۲-۴۳ می بینید)، هر باند فرکانسی 200 kHz پهنای دارد. هر سیستم GSM دارای ۱۲۴ زوج کانال یکطرفه ساده (simplex) است، که هر کانال 200 kHz پهنای دارد و از ۸ ارتباط همزمان (بصورت مالتی پلکس تقسیم زمانی) پشتیبانی می کند. به هر ایستگاه فعال در هر بُرش زمانی یک زوج کانال تخصیص داده می شود، که بدین ترتیب تعداد کانالهای موجود به 992 می رسد. البته (بدلیل جلوگیری از تداخل فرکانسی ایستگاههای مجاور) این تعداد کانال برای تمام ایستگاهها قابل استفاده نیست. در شکل ۲-۴۳، هشت بُرش زمانی خاکستری می بینید که همگی متعلق به یک تماس هستند (۴ تا برای ارسال، و ۴ تا برای دریافت). ارسال و دریافت همزمان انجام نمی شود، چون دستگاههای رادیویی GSM برای سوییچ کردن فرکانس به زمان نیاز دارند، و نمی توانند همزمان هر دو کار را انجام دهند. اگر به یک ایستگاه فرکانس 890.4/935.4 MHz اختصاص داده شده باشد و بُرش زمانی 2 بخواهد چیزی به این ایستگاه بفرستد، باید از بُرشهای خاکستری رنگ قسمت پائین تصویر (به هر تعداد که نیاز دارد) برای کار خود استفاده کند.

بُرشهای TDM نشان داده شده در شکل ۲-۴۳ بخشی از یک سلسله مراتب پیچیده فریم بندی هستند. هر بُرش TDM دارای ساختار خاصیتیست که ترکیب آنها نیز به روشی خاص تشکیل یک فریم چندگانه می دهد. در شکل ۲-۴۴ شکل ساده شده ای از این ساختار سلسله مراتبی را ملاحظه می کنید. همانطور که در این شکل



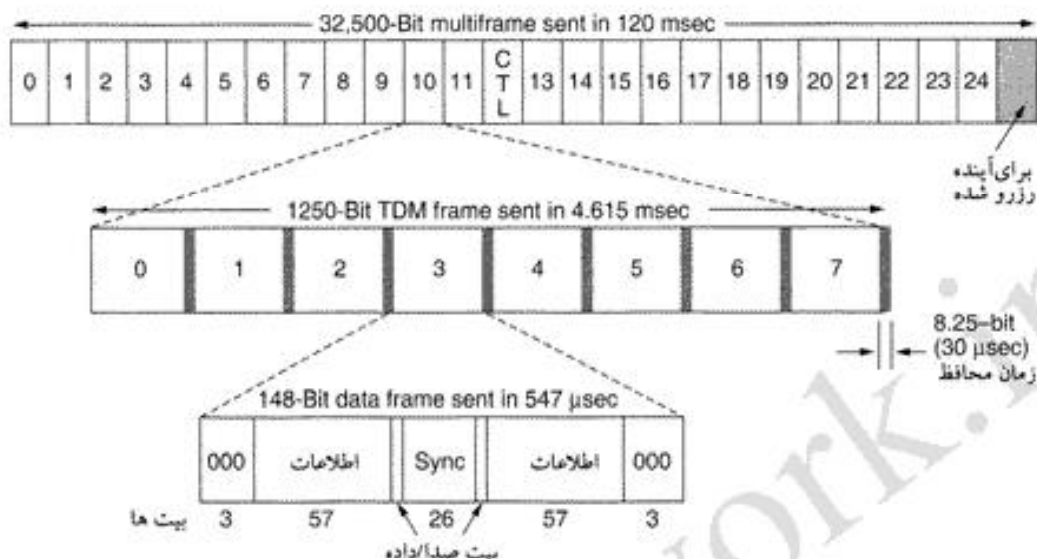
شکل ۲-۲۳. GSM از ۱۲۴ کانال فرکانسی استفاده می کند، که هر یک از آنها به ۸ بُرش زمانی تقسیم می شود.

می بینید، هر بُرش TDM از یک فریم داده ۱۴۸ بیتی تشکیل می شود، که کانال را برای مدت $577 \mu\text{sec}$ (شامل $30 \mu\text{sec}$ زمان حفاظت در هر بُرش) به اشغال خود در می آورد. به منظور همزمان کردن فریمها، هر فریم داده با ۳ بیت 0 شروع و ختم می شود. هر فریم دو فیلد اطلاعاتی (Information) ۵۷ بیتی دارد، که هر کدام دارای ۱ بیت کنترلی هستند که مشخص می کند این فیلد اطلاعاتی حاوی داده است یا صد. بین این فیلدهای اطلاعاتی یک فیلد نظم دهنده (Sync) ۲۶ بیتی وجود دارد که گیرنده از آن برای همزمان شدن با فرستنده استفاده می کند. ارسال هر فریم داده $547 \mu\text{sec}$ طول می کشد، ولی از آنجائیکه هر کانال بین ۸ ایستگاه به اشتراک گذاشته شده، فرستنده فقط در هر 4.615 msec می تواند یک فریم بفرستد. نرخ داده (ناخالص) هر کانال $270,833 \text{ bps}$ است که ۸ کاربر مشترکاً از آن استفاده می کنند. با یک تقسیم ساده مشخص می شود که نرخ داده ناخالص هر کاربر 33.854 kbps (بیش از دو برابر سیستمهای D-AMPS، یعنی 16.2 kbps) است. اما در اینجا هم مانند AMPS سرآیندها بخش زیادی از پهنای باند را می بلعند، و در نهایت 24.7 kbps برای داده واقعی کاربر (قبل از تصحیح خطا) باقی می ماند. آنچه پس از تصحیح خطا برای صدا باقی می ماند، 13 kbps است که باز هم کیفیت صدای بسیار بهتری نسبت به D-AMPS بدست می دهد (البته به قیمت مصرف پهنای باند بسیار بیشتر).

همانطور که در شکل ۲-۲۴ می بینید، هر ۸ فریم تشکیل یک فریم TDM و هر ۲۶ فریم TDM تشکیل یک فریم چندگانه 120-msec را می دهند. از ۲۶ فریم TDM، فریم ۱۲ برای کارهای کنترلی مورد استفاده قرار می گیرد، فریم ۲۵ نیز برای مصارف آنی کنار گذاشته شده است، و فقط ۲۴ فریم برای ارسال و دریافت باقی می ماند.

علاوه بر ساختار ۲۶ فریمی شکل ۲-۲۴، در GSM از ساختار دیگری با ۵۱ فریم نیز استفاده می شود. در این ساختار نیز برخی از فریمها کارکرد کنترلی (مدیریت سیستم) دارند. کانال کنترل پخش (broadcast control channel) یکی از این کانالهاست، که استریم خروجی پیوسته ایست که از ایستگاه مرکزی منتشر می شود و شامل هویت ایستگاه و اطلاعاتی درباره وضعیت کانالهای آن می شود. تمام دستگاههای تلفن همراه دائماً قدرت این سیگنال را چک می کنند تا بتوانند موقعیت خود را در میان سلولها تشخیص دهند.

کانال کنترل اختصاصی (dedicated control channel) یکی دیگر از کانالهای کنترلی است که برای به روز در آوردن موقعیت تلفنهای همراه، ثبت آنها و برقراری تماس بکار می رود. بویژه، هر ایستگاه مرکزی دارای پایگاه



شکل ۲-۴۴. بخشی از ساختار فریم بندی GSM.

داده‌ای از تمام تلفنهای موجود در محدوده قدرت خود است، که با استفاده از اطلاعاتی که روی این کانال فرستاده می‌شود، به روز در می‌آید.

دیگر کانال کنترلی، کانال کنترل مشترک (common control channel) است که به سه زیرکانال تقسیم می‌شود. اولین زیرکانال، کانال فراخوانی (paging channel) است که ایستگاه مرکزی از آن برای اعلام تماس ورودی استفاده می‌کند (و هر تلفن دائماً این کانال را چک می‌کند تا ببیند آیا تماسی دارد یا خیر). زیرکانال دوم، کانال دسترسی تصادفی (random access channel) است، که اجازه می‌دهد تا کاربران درخواستی برای یک بُرش از کانال کنترلی اختصاصی را روی آن ارسال کنند؛ کاربران از این بُرش برای برقراری تماس با دیگران استفاده می‌کنند. اگر دو کاربر همزمان وارد این کانال شوند، تداخل پیش می‌آید و باید (پس از کمی تأخیر) عملیات را از نو تکرار کنند. بُرش اختصاص داده شده به کاربر روی زیرکانال سوم، کانال اختصاص دسترسی (access grant channel)، به وی اعلام می‌شود.

CDMA: دسترسی چندگانه با تقسیم کد

سیستمهای D-AMPS و GSM هر دو سیستمهایی نسبتاً سستی و معمولی هستند، که از FDM و TDM برای تقسیم طیف فرکانسی به کانالها و تقسیم کانالها به بُرشهای زمانی استفاده می‌کنند. اما بازیگر سومی بنام CDMA (دسترسی چندگانه با تقسیم کد - Code Division Multiple Access) نیز در این صحنه حاضر است، که به روشی کاملاً متفاوت بازی می‌کند. وقتی CDMA برای اولین بار در صنعت مطرح شد، همان واکنشی را برانگیخت که پیشنهاد کریستف کلمب برای رسیدن به هندوستان از راه سفر به غرب، اما در نتیجه پایداری و مقاومت یک شرکت بنام Qualcomm، اکنون CDMA به جایی رسیده که نه تنها بعنوان یک سیستم قابل قبول مطرح است، بلکه به آن به چشم تنها مبنای مطمئن برای سیستمهای تلفن همراه نسل سوم نگاه می‌کنند. در ایالات متحده، CDMA حتی اکنون (در سیستمهای نسل دوم) نیز بعنوان رقیبی جدی برای D-AMPS مطرح است - برای مثال، شرکت Sprint PCS از CDMA استفاده می‌کند، در حالیکه AT&T Wireless از D-AMPS. استاندارد CDMA در سند IS-95 تدوین شده، و گاهی به این نام هم شناخته می‌شود؛ نام cdmaOne نیز یکی از نامهای تجاری آن است.

CDMA بکلی از AMPS، D-AMPS و GSM متفاوت است: بجای تقسیم طیف فرکانسی به کانالهای باریک، CDMA اجازه می دهد تا تمام تلفنها و ایستگاهها از تمام طیف فرکانسی برای ارسال و دریافت استفاده کنند. و برای تفکیک آنها از یکدیگر از تنوری رمزگذاری (coding theory) استفاده می کند. در سیستمهای CDMA این فرض که فریمهای تداخلی کرده غیر قابل استفاده اند را نیز بکلی کنار می گذارد. و بجای آن فرض می کند که این سیگنالها بصورت خطی با هم جمع می شوند.

برای درک بهتر CDMA از یک مثال آشنا استفاده می کنیم: سالتی پر از جمعیت که دو به دو مشغول صحبت هستند. TDM مانند آن است که این افراد را وسط سالن جمع کنید، ولی فقط به نوبت به آنها اجازه صحبت کردن بدهید. FDM مانند آن است که این افراد را با فاصله زیاد از یکدیگر بچینید. و به آنها اجازه دهید دو به دو همزمان (و البته مستقل) با هم صحبت کنند. CDMA نیز مانند این است که همه افراد را وسط سالن جمع کنید، ولی آنها (همزمان) به زبانهای مختلف با هم صحبت کنند. برای مثال، دو نفری که به فرانسه با هم حرف می زنند، هر چیزی غیر از کلمات فرانسوی را بعنوان پارازیت (نویز) شنیده می گیرند. نکته کلیدی در CDMA همین استخراج سیگنال مورد نظر (و دور زدن هر چیز دیگر) است. در زیر روش کار یک سیستم ساده شده CDMA را توضیح می دهیم.

در CDMA، هر بیت به m فاصله زمانی کوتاه، موسوم به چیب (chip)، تقسیم می شود. معمولاً هر بیت دارای ۶۴ یا ۱۲۸ چیب است، ولی در اینجا برای سادگی هر بیت را به فقط ۸ چیب تقسیم کرده ایم. به هر تلفن همراه (یا ایستگاه) یک کد m بیتی منحصر بفرد، که به آن توالی چیب (chip sequence) می گویند، اختصاص داده می شود. برای ارسال بیت ۱، ایستگاه توالی چیب خود را می فرستد، و برای ارسال بیت ۰ مکمل یک (one's complement) توالی چیب خود را - یک ایستگاه مجاز نیست هیچ چیز دیگری بفرستد. برای مثال، با فرض $m = 8$ ، اگر ایستگاهی بنام A دارای توالی چیب 00011011 باشد، برای ارسال بیت ۱ توالی 00011011 و برای ارسال ۰ توالی 11100100 را می فرستد.

افزایش مقدار اطلاعات ارسالی از b bits/sec به mb chips/sec فقط وقتی امکانپذیر است که پهنای باند موجود m برابر شود، که بدین ترتیب CDMA به سیستمی با طیف گسترده تبدیل می شود. اگر پهنای باند موجود برای ۱۰۰ ایستگاه 1 MHz باشد، با تکنیکهای FDM هر ایستگاه فقط 10 kHz در اختیار خواهد داشت و می تواند حداکثر 10 kbps (با فرض 1 bit/Hz) ارسال کند. با CDMA هر ایستگاه می تواند از تمامی پهنای باند 1-MHz استفاده کند، و به سرعت 1 megachips/sec برسد. اگر تعداد چیب بر بیت کمتر از ۱۰۰ باشد، پهنای باند مؤثر CDMA باز هم بیشتر از FDM است (و مشکلات تخصیص کانال نیز دیگر وجود ندارد).

در این قسمت برای درک بهتر مطلب روش دو-علامتی را بکار می بریم، یعنی بجای ۰ باینری از -۱ و بجای ۱ باینری از +۱ استفاده خواهیم کرد. توالی چیب ها را نیز در پرانتز نمایش می دهیم، که بدین ترتیب توالی چیب ایستگاه A به $(+1 +1 -1 -1 +1 -1 -1 +1)$ تبدیل می شود. در شکل ۲-۴۵ (الف) توالی چیب چهار ایستگاه بنامهای A، B، C و D را می بینید: در شکل ۲-۴۵ (ب) نیز همین توالی ها را به روش دو-علامتی نشان داده ایم. هر ایستگاه توالی چیب خاص خود را دارد. فرض می کنیم S بردار m چیبی ایستگاه S، و $\bar{S}$ بردار منفی (مکمل یک) آن است. تمام بردارهای توالی چیب متعامد (orthogonal) هستند، بعبارت دیگر ضرب داخلی نرمال شده هر دو بردار S و T $(S \cdot T)$ صفر است. با استفاده از روشی بنام کدهای والش (Walsh codes) می توان چنین بردارهایی تولید کرد. متعامد بودن دو بردار را به زبان ریاضی می توان چنین نوشت:

$$S \cdot T \equiv \frac{1}{m} \sum_{i=1}^m S_i T_i = 0$$

همانطور که خواهیم دید، متعامد بودن بردارهای یکی از ویژگیهای کلیدی این سیستم است. توجه داشته باشید که اگر $S \cdot T = 0$ ، آنگاه $S \cdot \bar{T} = 0$. ضرب داخلی نرمال شده هر بردار در خودش نیز 1 است:

$$S \cdot S = \frac{1}{m} \sum_{i=1}^m S_i S_i = \frac{1}{m} \sum_{i=1}^m S_i^2 = \frac{1}{m} \sum_{i=1}^m (\pm 1)^2 = 1$$

از آنجائیکه هر یک از جملات این دنباله 1 است، مجموع m جمله آن معادل m می شود، که بعد از ضرب در $\frac{1}{m}$ به عدد 1 خواهیم رسید. همچنین توجه داشته باشید که، $S \cdot \bar{S} = -1$.

همانطور که گفتیم، هر ایستگاه می تواند برای ارسال بیت 1 توالی چیب خود، و برای ارسال بیت 0 مکمل یک توالی چیب خود را بفرستد، و یا اصلاً سکوت کند و چیزی نفرستد. فعلاً فرض را بر این می گذاریم که تمام ایستگاهها سکرون هستند، یعنی ارسال توالی های خود را در یک زمان شروع می کنند.

وقتی دو یا چند ایستگاه بطور همزمان شروع به ارسال می کنند، سیگنال دو-علامتی آنها بصورت خطی با هم جمع می شود. برای مثال، اگر (در دوره زمانی یک چیب) سه ایستگاه سیگنال +1 و یک ایستگاه سیگنال -1 بفرستند، مجموع آنها +2 خواهد شد. می توان این وضعیت را مانند جمع ولتاژها تصور کرد: مجموع سه ولتاژ +1 ولت و یک ولتاژ -1 ولت معادل +2 ولت می شود. در شکل ۴۵-۲ (ج) شش نمونه از ارسال همزمان یک یا چند ایستگاه را ملاحظه می کنید. در مثال اول، ایستگاه C مبادرت به ارسال یک بیت 1 می کند، عبارت دیگر توالی چیب خود (01011100) را می فرستد. در مثال دوم، هر دو ایستگاه B و C یک بیت 1 می فرستند، که بدین ترتیب مجموع توالی های دو-علامتی آنها چنین خواهد شد:

$$(-1 -1 +1 +1 +1 +1 +1 -1) + (-1 +1 -1 +1 +1 +1 -1 -1) = (-2 \ 0 \ 0 \ 0 \ 2 \ 2 \ 0 -2)$$

A: 00011011
B: 00101110
C: 01011100
D: 01000010

(الف)

A: (-1 -1 -1 +1 +1 -1 +1 +1)
B: (-1 -1 +1 -1 +1 +1 +1 -1)
C: (-1 +1 -1 +1 +1 +1 -1 -1)
D: (-1 +1 -1 -1 -1 -1 +1 -1)

(ب)

شش مثال:

-- 1 -	C	$S_1 = (-1 +1 -1 +1 +1 +1 -1 -1)$
- 1 1 -	B + C	$S_2 = (-2 \ 0 \ 0 \ 0 \ 2 \ 2 \ 0 -2)$
1 0 --	A + B	$S_3 = (0 \ 0 -2 \ 2 \ 0 -2 \ 0 \ 2)$
1 0 1 -	A + B + C	$S_4 = (-1 +1 -3 \ 3 \ 1 -1 -1 +1)$
1 1 1 1	A + B + C + D	$S_5 = (-4 \ 0 -2 \ 0 \ 2 \ 0 \ 2 -2)$
1 1 0 1	A + B + C + D	$S_6 = (-2 -2 \ 0 -2 \ 0 -2 \ 4 \ 0)$

(ج)

$S_1 \cdot C = (1 +1 +1 +1 +1 +1 +1)/8 = 1$
 $S_2 \cdot C = (2 +0 +0 +0 +2 +2 +0)/8 = 1$
 $S_3 \cdot C = (0 +0 +2 +2 +0 -2 +0 -2)/8 = 0$
 $S_4 \cdot C = (1 +1 +3 +3 +1 -1 +1 -1)/8 = 1$
 $S_5 \cdot C = (4 +0 +2 +0 +2 +0 -2 +2)/8 = 1$
 $S_6 \cdot C = (2 -2 +0 -2 +0 -2 -4 +0)/8 = -1$

(د)

شکل ۴۵-۲. (الف) توالی چیب باینری چهار ایستگاه. (ب) توالی چیب دو-علامتی همان ایستگاهها. (ج) شش نمونه از ارسال همزمان ایستگاهها. (د) استخراج سیگنال ایستگاه C.

در مثال سوم، ایستگاه A یک بیت 1 و ایستگاه B یک بیت 0 می فرستد، و بقیه ایستگاهها ساکت هستند. در مثال چهارم، ایستگاههای A و C یک بیت 1 و ایستگاه B یک بیت 0 می فرستند. در مثال پنجم، هر چهار ایستگاه یک بیت 1 می فرستند؛ و بالاخره در مثال آخر، ایستگاههای A ، B و D یک بیت 1 می فرستند و ایستگاه C یک بیت 0. توجه داشته باشید که در تمام مثالهای فوق، توالی های S_1 تا S_8 فقط یک بیت را نشان می دهند.

برای تشخیص و استخراج استریم بیت های یک ایستگاه باید توالی چیب آن ایستگاه را از قبل بدانیم. این کار را می توان با محاسبه ضرب داخلی نرمال شده توالی چیب دریافت شده (جمع خطی سیگنال تمام ایستگاهها در آن لحظه) با توالی چیب ایستگاه موردنظر انجام داد. اگر توالی دریافت شده را S بنامیم، و بخواهیم توالی چیب (سیگنال ارسالی) ایستگاه C را از آن بیرون بکشیم، کافیست ضرب داخلی نرمال شده $S \cdot C$ را محاسبه کنیم.

برای اینکه ببینید این روش چگونه کار می کند، مثال چهارم شکل ۲-۴۵ (ج) را در نظر بگیرید. چیزی که گیرنده دریافت می کند، $S = A + \bar{B} + C$ است، و برای استخراج سیگنال C بایستی عبارت $S \cdot C$ را محاسبه کند:

$$S \cdot C = (A + \bar{B} + C) \cdot C = A \cdot C + \bar{B} \cdot C + C \cdot C = 0 + 0 + 1 = 1$$

همانطور که می بینید، دو جمله اول (بدلیل متعامد بودن بردارها) صفر شده اند. از اینجا می توانید علت انتخاب بردارهای متعامد را دریابید.

این وضعیت را می توان بصورت سه سیگنال مستقل نیز در نظر گرفت: سیگنالها بصورت مستقل و جدا از هم دریافت شده، و پس از محاسبه ضرب داخلی، با هم جمع می شوند. بعلت متعامد بودن بردارها، تمام ضربهای داخلی (بجز $C \cdot C$) صفر خواهند شد. بعبارت دیگر، تقدم ضرب داخلی یا جمع تأثیری بر نتیجه نهایی ندارد.

برای درک بهتر روش از رمز خارج کردن سیگنالها به شکل ۲-۴۵ (د) نگاه کنید. فرض کنید گیرنده می خواهد از شش سیگنال S_1 تا S_8 بیت های ارسال شده از ایستگاه C را استخراج کند. برای این کار، گیرنده تک تک سیگنالهای S را در بردار C (شکل ۲-۴۵ ب) ضرب کرده، و سپس $1/8$ آنرا محاسبه می کند (چون، $m = 8$). همانطور که می بینید، گیرنده توانسته است بطور صحیح بیت های ارسالی از C را استخراج کند.

در یک سیستم ایده آل (بدون نویز) CDMA تعداد ایستگاهها را می توان به تعداد دلخواه زیاد کرد (همانگونه که در یک کانال بدون نویز نایکونیست می توان نرخ نمونه برداری را بدلولخواه افزایش داد). اما در عمل، محدودیتهای فیزیکی ظرفیت سیستم را بشدت پائین می آورند. اول اینکه، فرض کردیم تمام چیب ها از نظر زمانی سنکرون هستند، در حالیکه این وضعیت عملاً غیرممکن است. بهترین کاری که می توان کرد اینست که فرستنده با ارسال یک توالی چیب از پیش تعریف شده (که بانداژه کافی طولانی است) به گیرنده امکان دهد تا خود را با فرستنده سنکرون کند. در این حالت تمام سیگنالهای غیرسنکرون بعنوان نویز تلقی خواهند شد. اگر تعداد این سیگنالهای غیرسنکرون چندان زیاد نباشد، الگوریتم فوق همچنان بخوبی کار خواهد کرد. در زمینه برهم نهی (superposition) توالی های چیب با سطح نویز تحقیقات تئوریک مفصلی انجام شده است (Pickholtz et al., 1982). همانطور که می توان انتظار داشت، هر چه توالی چیب طولانی تر باشد، احتمال استخراج آن در محیط های پرنویز بیشتر خواهد بود. حتی می توان در توالی بیت ها از گدهای تصحیح خطا نیز استفاده کرد - البته در توالی چیب هرگز از گدهای تصحیح خطا استفاده نمی شود.

یکی دیگر از مفروضات ضمنی بحث فوق یکسان بودن قدرت سیگنالهاییست که به گیرنده می رسند. در سیستمهای تلفن همراه (و از جمله CDMA) فاصله تلفنهای همراه از ایستگاه مرکزی (و در نتیجه قدرت تشعشعی آنها) متغیر است، و قدرت سیگنالهایی که از تلفنها به ایستگاه مرکزی می رسد یکسان نیست. یکی از روشهای ابتکاری برای حل این مشکل آنست که تلفنهای همراه با کاهش سطح سیگنال دریافتی از ایستگاه مرکزی، توان تشعشعی خود را بالا ببرند؛ بعبارت دیگر، هر چه از ایستگاه مرکزی دور می شوند، سیگنال قویتری بفرستند.

ایستگاه مرکزی نیز می تواند صریحاً به تلفنهای همراه فرمان دهد تا (بسته به فاصله شان از ایستگاه) قدرت سیگنالهای خود را افزایش یا کاهش دهند.

همچنین فرض کردیم که گیرنده از هویت فرستنده آگاه است. از نظر تئوری این امکان هست که (با فرض وجود قدرت محاسباتی)، یک گیرنده به تمام سیگنالها گوش دهد و الگوریتمهای از رمز خارج کردن را روی تمام آنها اجرا کند. اما مثل معروفی هست که می گوید، «حرف زدن همیشه از عمل کردن ساده تر است». CDMA پیچیدگیهای دیگری نیز دارد، که در این بحث مختصر آنها را نادیده گرفتیم. با این حال، CDMA یکی از هوشمندانه ترین سیستمهای مخابرات بیسیم محسوب می شود، که سرعت در حال گسترش است. این سیستم معمولاً از یک باند 1.25 MHz (برخلاف 30 kHz در D-AMPS و 200 kHz در GSM) استفاده می کند، و تعداد کاربرانی که پشتیبانی می کند، از هر دو سیستم قبلی بیشتر است. در عمل، پهنای باند موجود برای هر کاربر در سیستمهای CDMA حتی از GSM هم بهتر است.

یکی از بهترین منابع موجود در این زمینه (Lee and Miller, 1998) است. روشهای دیگری نیز در (Crespo et al., 1995) و (Sari et al., 2000) تشریح شده اند، که البته فهم آنها به دانش زیادی در زمینه مهندسی مخابرات نیاز دارد.

۳-۶-۲ تلفن های همراه نسل سوم: صدای دیجیتال و داده

آینده تلفن همراه چیست؟ اجازه دهید نگاه سریعی به آن بیندازیم. صنعت تلفن همراه عوامل محرک و پیشبرنده متعددی دارد. اول، ترافیک داده مدتهاست از ترافیک صدا پیشی گرفته و همچنان به رشد تصاعدی خود ادامه می دهد، ولی ترافیک صدا سالهاست دیگر رشد چندانی را تجربه نمی کند. بسیاری از متخصصان پیش بینی می کنند که بزودی در تلفنهای همراه نیز شاهد پیشی گرفتن ترافیک داده از ترافیک صدا خواهیم بود. دوم، صنایع تلفن، تفریحات و کامپیوتر همگی دیجیتالی شده اند و روز بروز بیشتر به یکدیگر نزدیک می شوند. مردم مدتهاست با شور و هیجان از دستگاه کوچک، سبک و قابل حملی صحبت می کنند که بتواند بعنوان تلفن، پخش CD، پخش DVD، ترمینال ایمیل و وب، وسیله بازی و سرگرمی، واژه پرداز (و غیره و غیره) کار کند، و قادر باشد در هر نقطه ای بصورت بیسیم و با پهنای باند بالا به اینترنت متصل شود. این دستگاه رؤیایی همان تلفن همراه نسل سوم است؛ برای اطلاعات بیشتر (Huber et al., 2000; 2000) and Sarikaya را ببینید.

در سال ۱۹۹۲، ITU کوشید این رؤیا را کمی بیشتر به واقعیت نزدیک کند، و به همین منظور طرح اولیه ای بنام IMT-2000 (که IMT مخفف سیستم مخابرات تلفن همراه بین المللی - International Mobile Telecommunication - است) منتشر کرد. عدد 2000 سه چیز را نشان می داد: (۱) سالی که این سیستم بایستی عملیاتی شود، (۲) فرکانسی (بر حسب MHz) که سیستم تحت آن کار می کند، و (۳) پهنای باند این سرویس (بر حسب kHz).

البته IMT-2000 به هیچیک از این اهداف نرسید. در سال ۲۰۰۰ هیچ سیستمی نصب و راه اندازی نشد. ITU پیشنهاد کرده بود که دولتها طیف 2 GHz را برای این منظور کنار بگذارند تا این سرویس بتواند بین کشورهای مختلف بدون اشکال کار کند؛ تنها کشوری که به این توصیه عمل کرد، چین بود. و بالاخره، مشخص شد که در حال حاضر پهنای باند 2 Mbps برای کاربرانی که بیش از حد متحرک هستند، عملی نیست (علت آن هم دشواری پاس کاری این قبیل کاربران است). پهنای باند عملی تر عبارتست از 2 Mbps برای کاربران ثابت خانگی (که می تواند با ADSL رقابت کند)، 384 kbps برای کاربرانی که قدم می زنند، و 144 kbps برای آنهایی که سوار اتومبیل هستند. با این حال، حوزه فعالیت 3G (نسل سوم) بسیار پُر جنب و جوش است. نسل سوم شاید کمی دیرتر از راه برسد و کمی کمتر از آنچه که انتظار می رود باشد، ولی حتماً می آید.

سرویسهایی که قرار است شبکه IMT-2000 در اختیار کاربران خود بگذارد، عبارتند از:

۱. انتقال صدا با کیفیت عالی.
۲. پیام رسانی (سرویس که جایگزین ایمیل، فکس، SMS، chat و غیره خواهد شد).
۳. مالتی مدیا (پخش موسیقی، تماشای ویدئو، فیلم، تلویزیون و غیره).
۴. دسترسی اینترنت (گشت و گذار در وب، از جمله صفحاتی که صدا، تصویر و فیلم دارند).

سرویسهای دیگری از قبیل کنفرانس ویدئویی (video conferencing)، حضور از راه دور (telepresence)، بازیهای گروهی و تجارت-همراه (m-commerce): پرداخت بهای اجناس خریداری شده در فروشگاه با نزدیک کردن تلفن همراه به صندوق) را نیز می توان از این شبکه انتظار داشت. علاوه بر آن، تمام این سرویسها بین المللی خواهند بود (یعنی در تمام نقاط دنیا می توان به آنها دسترسی داشت: در جاهایی که خطوط ارتباطی زمینی وجود ندارد، تلفن همراه بطور خودکار از لینکهای ماهواره ای استفاده خواهد کرد)، و همیشه با کیفیت تضمین شده در دسترس هستند.

ITU برای IMT-2000 تکنولوژی واحدی را در نظر گرفته است، بگونه ای که تلفنهای همراه بتوانند در هر نقطه ای از دنیا کار کنند (مانند دستگاههای ضبط صوت و پخش CD، نه تلویزیونها و تلفنهای همراه امروزی). یکسان شدن تکنولوژی، علاوه بر تسهیل کار شرکتهای مخابرات، افراد بیشتری را به استفاده از این سرویسها تشویق خواهد کرد. جنگ تکنولوژیها (مانند آنچه بین ویدئوهای Betamax و VHS رخ داد) هیچگاه به نفع صنعت و تجارت نبوده است.

پیشنهادهای متعددی ارائه شد، که بعد از غربال شدن آنها، سرانجام دو طرح باقی ماند. طرح اول، بنام W-CDMA (CDMA پهن باند - Wideband CDMA)، از طرف شرکت سوئدی اریکسون ارائه شد. این طرح از توالی مستقیم با طیف گسترده (مانند آنچه در بالا گفتیم) استفاده می کند. این سیستم در یک باند 5-MHz کار می کند، و بگونه ای طراحی شده که بتواند با شبکه های GSM (البته نه GSM قدیمی) کار کند. تلفنهای این سیستم می توانند بدون اختلال در ارتباط یک سلول W-CDMA را ترک کرده و وارد یک سلول GSM شوند. اتحادیه اروپا بشدت از این سیستم پشتیبانی می کند، و به آن نام UMTS (سیستم مخابرات تلفن همراه جهانی - Universal Mobile Telecommunications System) را داده است.

طرح دیگر که از طرف شرکت Qualcomm (مخترع CDMA) پیشنهاد شده، CDMA2000 نام دارد. این سیستم نیز اساساً همان توالی مستقیم با طیف گسترده (شکل اصلاح شده استاندارد IS-95) است، که با آن سازگاری دارد. CDMA2000 نیز از یک باند 5-MHz استفاده می کند، ولی نمی تواند با سلولهای GSM (و طبیعتاً، با سلولهای D-AMPS) پاس کاری انجام دهد. CDMA2000 تفاوتهای دیگری، از قبیل نرخ چیب، زمان فریم، طیف فرکانسی، و روش سنکرون شدن، نیز با W-CDMA دارد.

اگر مهندسان اریکسون و Qualcomm را در اتفاقی حبس کنند و از آنها بخواهند یک سیستم مشترک طراحی کنند، به احتمال زیاد می توانند. بهر حال، هر دو سیستم از تکنیکهای CDMA و یک کانال 5-MHz استفاده می کنند، و چیزهای دیگر هم مسلماً ارزش خودکشی ندارند. در اینجا هم (مثل همیشه) مشکل نه فنی و مهندسی بلکه سیاسی است. اروپا سیستمی می خواهد که با GSM سازگاری داشته باشد، و ایالات متحده آمریکا دنبال سیستمی است که با IS-95 سازگار باشد. هر دو نیز از شرکتهای محلی خود دفاع می کنند: اروپا از اریکسون (که شرکتی سوئدی است) و آمریکا از Qualcomm (که در کالیفرنیا است). اریکسون و Qualcomm دعوای حقوقی بیشماری نیز بر سر حق اختراع CDMA داشته اند.

بالاخره در مارس ۱۹۹۹ اریکسون Qualcomm را خرید، و دعاوها خاتمه یافت. آنها بر سر یک استاندارد 3G نیز به توافق رسیدند، ولی این استاندارد ناسازگاری های زیادی داشت. اما این منازعات بالاخره به پایان می رسد، و بزودی شاهد تلفن ها و سرویس های 3G خواهیم بود.

در باره سیستم های 3G مطالب زیادی نوشته شده، و برخی آنرا بعنوان یکی از بزرگترین اختراعات بشری ستوده اند. برای نمونه به (Collins and Smith, 2000; De Vriendt et al., 2002; Harte et al., 2002; 2002; Lu, and Sarikaya, 2000) اشاره می کنیم. اما 3G منتقدانی نیز دارد که معتقدند اساساً راه را اشتباه می رود، و می توان از میان آنها از (Garber, 2002; and Goodman, 2000) نام برد.

عده ای هم که از جنگ 3G خسته شده اند، به فکر افتادند تا قدم کوچکی به سمت آن بردارند، و نام آنرا هم 2.5G گذاشتند (اگر چه شاید نام 2.1G برای آن مناسب تر است). یکی از این سیستم ها EDGE (نسخه داده بهبود یافته برای تکامل GSM - Enhanced Data rates for GSM Evolution) نام دارد، که اساساً چیزی نیست جز همان GSM با bits/ baud بیشتر. مشکل اینجا است که bits/ baud بیشتر یعنی خطای بیشتر، و به همین دلیل EDGE در سرعت های متفاوت از 9 روش مختلف برای مدولاسیون و تصحیح خطا استفاده می کند.

یکی دیگر از طرح های 2.5G، GPRS (سرویس عمومی بسته رادیویی - General Packet Radio Service) نام دارد، که عبارتست از یک شبکه سوئیچینگ بسته ای که روی GSM یا D-AMPS کار می کند. این سیستم اجازه می دهد تا تلفن های همراه در سلول های صوتی بسته های IP رد و بدل کنند. GPRS برای این منظور از بُرش های زمانی و فرکانس های اختصاصی استفاده می کند، که تعداد و محل آنها (به نسبت ترافیک صوت و داده در سلول) بطور دینامیک توسط ایستگاه مرکزی تعیین می شود.

بُرش های زمانی موجود به چندین کانال منطقی تقسیم شده، و به مصارف مختلف می رسند. تخصیص بُرش های زمانی به هر کانال توسط ایستگاه مرکزی صورت می گیرد. یکی از این کانال های منطقی برای ارسال بسته ها از ایستگاه مرکزی به تلفن های همراه است، و هر بسته می داند که مقصد آن کجاست. برای ارسال یک بسته IP، تلفن همراه درخواستی برای تخصیص یک یا چند بُرش زمانی به ایستگاه مرکزی می فرستد. اگر این درخواست بدون مشکل به ایستگاه مرکزی برسد، ایستگاه مرکزی فرکانس و بُرش های زمانی تخصیص یافته را به تلفن همراه اعلام می کند. همین که بسته های IP ارسالی از تلفن همراه به ایستگاه مرکزی رسید، ایستگاه مرکزی (از طریق ارتباطاتی که دارد) آنها را به اینترنت منتقل می کند. از آنجائیکه GPRS فقط لایه ایست روی سیستم های صوتی موجود، می توان آنرا در بهترین حالت یک قدم به سمت 3G بشمار آورد.

با اینکه شبکه های 3G هنوز بطور کامل راه اندازی نشده اند، برخی از محققان آنرا موضوعی تحقق یافته (که دیگر ارزش توجه ندارد) تلقی کرده و سراغ سیستم های نسل چهارم (4G) رفته اند (Berezdivin et al., 2002; Guo and Chaskar, 2002; Huang and Zhuang, 2002; Kellerer et al., 2002; and Misra et al., 2002). برخی از مشخصات پیشنهادی سیستم های 4G عبارتند از: پهنای باند زیاد، دسترسی در همه جا، یکپارچگی کامل با شبکه های کابلی (و بویژه شبکه های IP)، مدیریت تطبیقی منابع و طیف فرکانسی، رادیوی نرم افزاری، و سرویس های مالی مدیا با کیفیت عالی.

از سوی دیگر، با توجه به تعداد بسیار زیاد شبکه های محلی بیسیم 802.11 که امروزه نصب شده، برخی معتقدند 3G به دنیا نیامده، مرده است. اینها می گویند، «هر جا که بروید بالاخره تحت پوشش یک شبکه 802.11 هستید، و این همان چیزیست که 3G می خواهد با آن همه منت به شما بدهد.» پنج سال دیگر معلوم می شود حق با چه کسی بوده است.

۷-۲ تلویزیون کابلی

تا اینجا سیستمهای تلفن ثابت و بیسیم را بررسی کردیم. هر دوی این سیستمها مسلماً نقش مهمی در آینده شبکه بازی خواهند کرد. اما بازیگر دیگری نیز در زمینه شبکه های ثابت وارد صحنه شده، و هر روز اهمیت بیشتری می یابد: **تلویزیون کابلی (Cable TV)**. امروزه افراد بسیاری سرویسهای تلویزیون و اینترنت خود را از طریق کابل دریافت می کنند، و شرکت های تلویزیون کابلی نیز با جدیت بدنبال سهم بیشتری از بازار هستند. در این قسمت تلویزیون کابلی را از دیدگاه شبکه مورد بررسی دقیقتر قرار خواهیم داد. برای کسب اطلاعات بیشتر نیز می توانید به (Laubach et al., 2001; Louis, 2002; Ovadia, 2001; and Smith, 2002) مراجعه کنید.

۱-۷-۲ تلویزیون با آنتن مرکزی

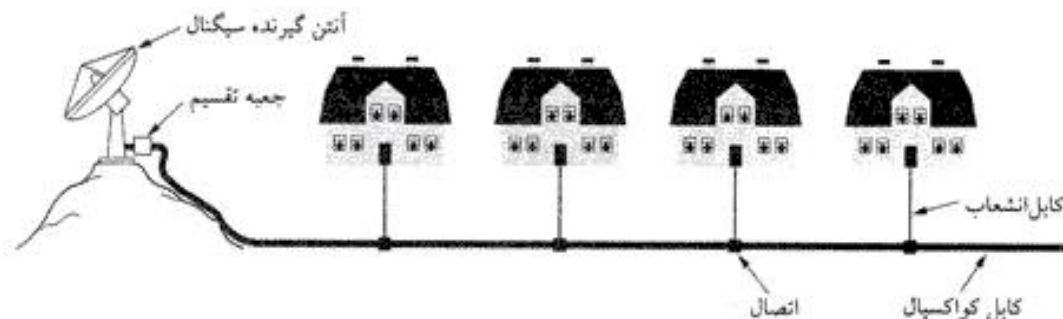
تلویزیون کابلی در اواخر دهه ۱۹۴۰ بعنوان راهی برای ارائه سرویسهای بهتر به مناطق روستایی و کوهستانی ابداع شد. این سیستم عبارت بود از یک آنتن بزرگ بر فراز نقطه ای مرتفع (برای دریافت امواج تلویزیونی)، یک تقویت کننده (موسوم به **جعبه تقسیم - head end**) برای تقویت سیگنال، و یک رشته کابل کواکسیال برای انتقال سیگنال تلویزیون به منازل؛ شکل ۲-۴۶ را ببینید.

در آن سالها به این سیستم **تلویزیون با آنتن مرکزی (Community Antenna Television)** گفته می شد. این سیستم بقدری ساده بود که هر کسی با کمترین تجربه سیمکشی نیز می توانست آنرا راه بیندازد، و هزینه آن هم معمولاً بین مشترکین سرشکن می شد. با اضافه شدن مصرف کنندگان فقط کافی بود انشعابهای بیشتری از کابل اصلی گرفته (و احیاناً تقویت کننده های بیشتری نصب) شود. این سیستم اساساً یکطرفه (از جعبه تقسیم سیگنال به مشترکین) بود، و تا دهه ۱۹۷۰ هزاران نمونه از آن نصب شده بود.

در سال ۱۹۷۴ شرکت رسانه ای تایم کانال جدیدی بنام **Home Box Office** راه اندازی کرد که فیلمهای سینمایی پخش می کرد، و فقط از طریق کابل قابل دسترسی بود. کانالهای کابلی دیگر نیز بسرعت راه افتادند که اخبار، مسابقات ورزشی، آشپزی (و بسیاری موضوعات دیگر) پخش می کردند. این تحول باعث دو تغییر اساسی در صنعت تلویزیون شد. اول اینکه، شرکت های بزرگ شروع به خریدن شبکه های کابلی موجود در شهرها کردند، و در بسیاری جاها نیز خود رأساً شروع به کابل کشی و جذب مشترکین جدید کردند. دوم اینکه، برای ارائه سرویس در سراسر کشور لازم بود بین شهرها نیز کابل کشی شود، پس شرکت های مزبور شروع به متصل کردن شهرها به یکدیگر کردند. این وضعیت درست مانند شبکه تلفن در ۸۰ سال قبل بود، که شرکت های تلفن برای ارائه سرویسهای راه دور بین شهرها کابل تلفن کشیدند.

۲-۷-۲ اینترنت کابلی

در طول سالها سیستمهای کابلی رشد کرد، و بین شهرهای مختلف فیبرهای نوری با پهنای باند زیاد کشیده شد. این

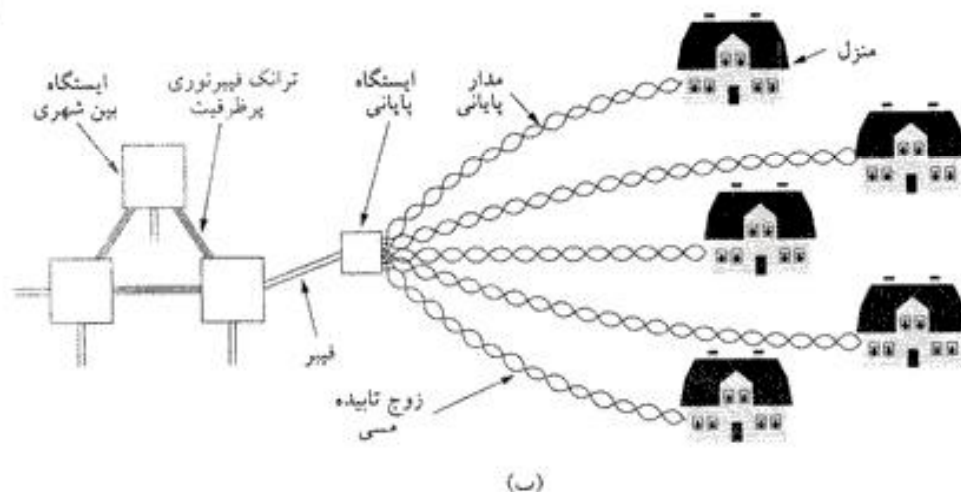
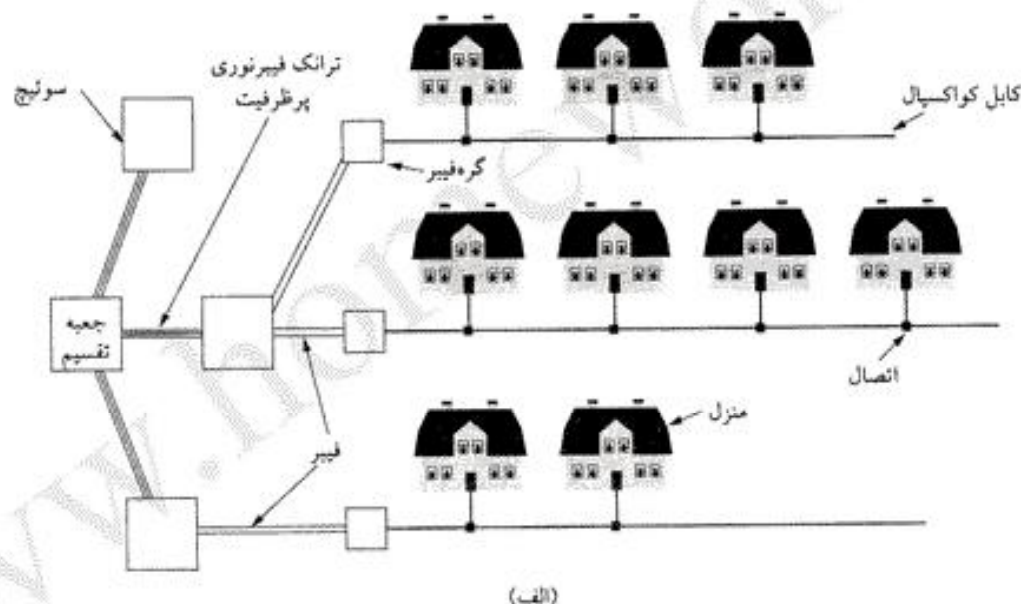


شکل ۲-۴۶. یک سیستم تلویزیون کابلی اولیه.

سیستم که ترکیبی بود از فیبر نوری (بین شهرها) و کابلهای کواکسیال (از مراکز توزیع تا منازل) HFC (آمیخته فیبر-کواکس - Hybrid Fiber Coax) نام دارد. در نقاطی که گره فیبر (fiber node) خوانده می شوند، سیگنالهای نوری به الکترونیکی (و بالعکس) تبدیل می شود. از آنجائیکه پهنای باند فیبر نوری بسیار بیشتر از کواکس است، یک گره فیبر می تواند تعداد زیادی کابل کواکسیال را تغذیه کند. در شکل ۲-۴۷ (الف) یک سیستم HFC مدرن را ملاحظه می کنید.

در سالهای اخیر، بسیاری از شرکت های کابلی تصمیم گرفته اند وارد تجارت دسترسی اینترنت (و همچنین تلفن) شوند. با این حال تفاوت های فنی شبکه های کابلی و تلفن چنان است که تاکنون مانع از تحقق کامل این خواسته شده است. یکی از مشکلاتی که می توان بعنوان نمونه به آن اشاره کرد، تقویت کننده های یکطرفه ای است که باید با تقویت کننده های دوطرفه جایگزین شوند.

اما تفاوت مهمتری بین سیستم HFC شکل ۲-۴۷ (الف) و سیستم تلفن (شکل ۲-۴۷ ب) وجود دارد، که برطرف کردن آن بسیار مشکلتر است. در سیستم HFC یک رشته کابل کواکسیال بین خانه های متعددی مشترک



شکل ۲-۴۷. (الف) تلویزیون کابلی، (ب) سیستم تلفن ثابت.

است، در حالیکه در سیستم تلفن شهری هر خانه دارای یک اتصال (مدار پایانی) خاص خود است. در پخش برنامه های تلویزیونی این مشترک بودن کابل اهمیت چندانی ندارد: یک برنامه پخش می شود، و اهمیتی ندارد که ۱۰ نفر آنرا می بینند یا ۱۰,۰۰۰ نفر. اما وقتی پای دسترسی اینترنت به میان می آید، ۱۰ نفر با ۱۰,۰۰۰ نفر فرق بسیاری با هم دارند. اگر یکی از کاربران بخواهد فایل بزرگی را از اینترنت بگیرد، پهنای باند لازم برای این کار از کاربران دیگر گرفته می شود. هر چه تعداد کاربران بیشتر شود، رقابت بر سر پهنای باند شدیدتر خواهد بود. در سیستم تلفن چنین مشکلی وجود ندارد: اگر شما مشغول گرفتن یک فایل بزرگ روی خط ADSL خود باشید، تأثیری روی کار همسایه تان نخواهد گذاشت. از طرف دیگر، پهنای باند یک کابل کواکس بسیار بیشتر از زوج-تابیده است.

شرکتهای کابلی برای غلبه بر این مشکل، کابلهای بلند را تکه تکه کرده و هر کدام را مستقیماً به یک گره فیبر متصل می کنند. اگر تعداد کاربران هر کابل کواکس خیلی زیاد نباشد، پهنای باند فیبر نوری عملاً نامحدود خواهد بود و می توان ترافیک را بخوبی مدیریت کرد. امروزه هر کابل کواکس به ۵۰۰ تا ۲۰۰۰ نفر سرویس می دهد، ولی با زیاد شدن تعداد مشترکین این سیستم شاید لازم باشد از کابلها و گره های فیبر بیشتری استفاده کرد.

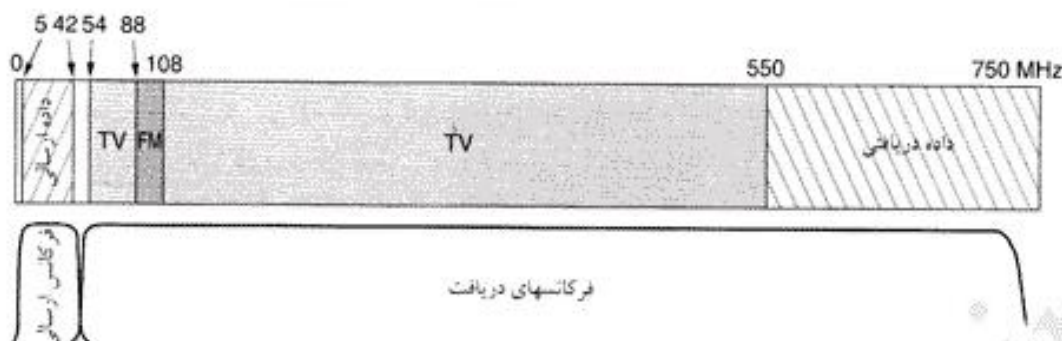
۳-۷-۲ تخصیص طیف فرکانسی

حذف تمام کانالهای تلویزیونی و یکارگیری زیرساخت های کابلی برای دسترسی اینترنت باعث نارضایتی تعداد زیادی از مشترکین خواهد شد، و به همین دلیل شرکتهای کابلی در این کار تردید دارند. علاوه بر آن، در اغلب شهرها قوانین سختگیرانه ای برای کنترل آنچه روی کابلها فرستاده می شود، وجود دارد، و حتی اگر شرکتهای کابلی واقعاً هم بخواهند اجازه چنین کاری را ندارند. در نتیجه، آنها باید راهی برای همزیستی تلویزیون کابلی و اینترنت روی یک کابل می یافتند.

کانالهای تلویزیون کابلی در منطقه آمریکای شمالی در ناحیه 54-550 MHz طیف قرار دارند (البته با استثنای محدوده 88-108 MHz که به رادیوی FM اختصاص دارد). هر کانال (با احتساب باندهای محافظ) 6 MHz پهنای دارد. در اروپا، حد پائین طیف معمولاً 65 MHz است و کانالها نیز 6-8 MHz پهنای دارند، چون سیستمهای تلویزیونی PAL و SECAM کیفیت بالاتری دارند. از قسمت پائین این باند استفاده نمی شود. کابلهای جدید می توانند بالاتر از 550 MHz (اغلب تا 750 MHz) نیز کار کنند. راه حل انتخاب شده این بود که کانالهای ارسال به اینترنت (upstream) در باند 5-42 MHz (در اروپا، کمی بالاتر) و کانالهای دریافت از اینترنت (downstream) در فرکانسهای بالاتر تعریف شوند. در شکل ۲-۴۸ طیف فرکانسی کابل کواکسیال را ملاحظه می کنید.

از آنجائیکه سیگنالهای تلویزیونی فقط در یک جهت (مرکز توزیع به منازل) منتشر می شوند، تقویت کننده های ارسال فقط در باند 5-42 MHz، و تقویت کننده های دریافت فقط در باند بالاتر از 54 MHz کار می کنند. همانطور که می بینید، بین باندهای ارسال و دریافت یک عدم تقارن وجود آمده، و باند دریافت از اینترنت بسیار وسیعتر است (البته این وضعیت چندان هم بد نیست، چون اغلب ترافیک اینترنت در همین جهت است). قبلاً هم دیدید که شرکتهای تلفن عمداً (و بدون اینکه اجبار تکنیکی وجود داشته باشد) سرویس DSL را بصورت نامتقارن در می آورند.

توانایی کابلهای بلند کواکسیال در انتقال سیگنالهای دیجیتال چندان بهتر از سیمهای زوج-تابیده نیست، و به همین دلیل آنها هم به نوعی مدولاسیون آنالوگ احتیاج دارند. برای مدولاسیون هر کانال 6 MHz (یا 8 MHz) از روش QAM-64 (و اگر کیفیت کابل فوق العاده خوب باشد، از QAM-256) استفاده می شود. با یک کانال 6-MHz و مدولاسیون QAM-64، نرخ انتقال داده ای معادل 36 Mbps بدست می آید، که اگر سرآیند از آن کسر شود، تقریباً 27 Mbps باقی می ماند. با مدولاسیون QAM-256 نرخ انتقال داده (بعد از کسر سرآیند) 39 Mbps



شکل ۲-۴۸. تخصیص فرکانس در یک سیستم تلویزیون کابلی برای دسترسی اینترنت.

خواهد بود. (ظرفیت سیستمهای اروپایی 1/3 بیشتر است.) در کانالهای ارسال به اینترنت بدلیل وجود نویز بالا (از منابع مایکروویو زمینی، و رادیوهای محلی) حتی QAM-64 نیز بخوبی کار نمی کند، و باید از روشهای محافظه کارانه تری (مانند QPSK) استفاده کرد. روش QPSK ظرفیت بسیار کمتری نسبت به مدولاسیون QAM دارد (شکل ۲-۲۵ را ببینید) - 2 bits/ baud بجای 6-8 bits/ baud. این اوصاف، عدم تقارن کانالهای ارسال و دریافت حتی از آنچه در شکل ۲-۴۸ دیده می شود، نیز بیشتر است. شرکت های کابلی مجبور شدند تقویت کننده های ساده خود را نیز با سیستم های دیجیتالی هوشمند جایگزین کنند. نام این تجهیزات نیز از جعبه تقسیم (headend) به CTMS (سیستم پایانه ای مودم کابلی - Cable Modem Termination System) تغییر کرده است، ولی ما در قسمتهای آتی همچنان از اصطلاح «جعبه تقسیم» برای اشاره به این تقویت کننده ها استفاده خواهیم کرد.

۴.۷.۲ مودمهای کابلی

مودم کابلی (cable modem) دستگاهیست با دو سر: یک سر به کامپیوتر وصل می شود، و سر دیگر به شبکه کابلی. در سالهای اولیه اینترنت کابلی، هر شرکت مودم کابلی خاصی داشت، که توسط تکنسینهای آن نصب می شد. اما بزودی معلوم شد که وجود سیستمی با استاندارد باز موجب گسترش رقابت در بازار، کاهش قیمت ها، و در نتیجه اقبال عمومی به این سرویسها خواهد شد. علاوه بر آن، خریدن یک مودم از فروشگاه و نصب آن توسط خود مشتری بسیار ساده تر از مراجعه یک تکنسین برای نصب هر مودم است (که هزینه بسیار بالایی نیز دارد). در نتیجه، شرکت های کابلی بزرگ برای تولید یک مودم کابلی استاندارد (و تست سازگاری آنها) با شرکتی بنام CableLabs متحد شدند. این استاندارد که DOCSIS (Data Over Cable Service Interface Specification) نام داشت، فقط نقطه شروعی بود برای جایگزینی مودمهای متنوع و ناسازگار - ویرایش اروپایی این استاندارد نیز EuroDOCSIS نام گرفت. اما ایده استاندارد کردن مودمهای کابلی برای همه شرکت های کابلی جالب نبود، چون آنها پول خوبی بابت اجازه دادن مودم به علاقمندان این سرویسها به جیب می زدند. یک استاندارد باز پای دهها تولیدکننده جدید را به این قلمرو باز می کرد، و به این تجارت پُر منفعت پایان می داد. ارتباط مودم به کامپیوتر ساده و سراسر است: در حال حاضر از اینترنت 10-Mbps (و گاهی هم USB) برای این منظور استفاده می شود. بطور مسلم در آینده شاهد کارتهای مودم کابلی (شبه کارتهای V.9x فعلی) نیز خواهیم بود.

ارتباط در سر دیگر پیچیده تر است، و بخش عمده ای از استاندارد آن به مهندسی رادیو مربوط می شود (که از

حاصله این کتاب خارج است). تنها نکته ای که باید بیاد داشته باشید این است که، یک مودم کابلی (مانند مودمهای ADSL) همیشه روشن و متصل است. ارتباط این مودمها به محض روشن شدن برقرار شده و تا زمان خاموش شدن در همین حالت باقی می ماند (هزینه آنها نیز بر حسب زمان محاسبه نمی شود).

برای درک بهتر طرز کار مودمهای کابلی، اجازه دهید ببینیم با روشن کردن آنها چه اتفاقی می افتد. مودم به محض روشن شدن تمام کانالهای دریافت را برای پیدا کردن بسته خاصی که در فواصل منظم از طرف منبع ارسال می شود (و حاوی پارامترهای سیستم برای مودمهایی که تازه روشن شده اند، می باشد) جستجو می کند. پس از پیدا کردن این بسته، مودم جدید حضور خود را از طریق یکی از کانالهای ارسال به منبع اعلام می کند. منبع نیز با تخصیص کانالهای ارسال و دریافت پاسخ مودم را می دهد. البته این کانالها هر زمان که منبع لازم ببیند (مثلاً، برای متعادل کردن بار)، می توانند عوض شوند.

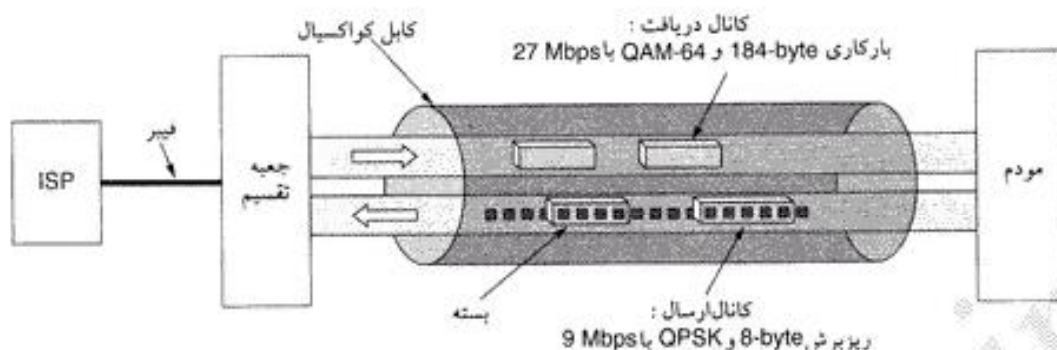
پس مودم با ارسال یک بسته خاص به منبع و محاسبه زمان رفت و برگشت آن، فاصله خود با منبع را تعیین می کند؛ به این کار تعیین فاصله (ranging) گفته می شود. این کار برای کارکرد صحیح کانالهای ارسال و همزمانی لازم است. این فاصله زمانی به ریزبُرشها (minislots) تقسیم می شود؛ هر بسته باید دقیقاً در یک یا چند ریزبُرش متوالی جا شود. منبع در فواصل منظم دور جدیدی از ریزبُرشها را اعلام می کند، ولی (بدلیل یکسان نبودن فاصله مودمها از منبع) این شلیک شروع مسابقه همزمان در تمام مودمها شنیده نخواهد شد. هر مودم، با دانستن فاصله خود از منبع، می تواند بفهمد که اولین ریزبُرش واقعاً در چه زمانی شروع شده است. طول ریزبُرشها به شبکه بستگی دارد (و ظرفیت آن معمولاً معادل ۸ بایت است).

در حین آماده سازی اولیه، منبع به هر مودم یک ریزبُرش تخصیص می دهد که می تواند از آن برای درخواست بهنای پاند ارسال استفاده کند. قاعدتاً تعداد مودمها از ریزبُرشها بیشتر است و یک ریزبُرش به چند مودم اختصاص داده می شود، که این می تواند موجب کشمکش بین آنها شود. وقتی کامپیوتر اطلاعاتی را برای ارسال به مودم می فرستد، مودم تعداد ریزبُرشهای لازم را محاسبه کرده و درخواست تخصیص آنها را از منبع می کند. اگر این درخواست پذیرفته شود، منبع ریزبُرشهای تخصیص داده شده را (روی یک از کانالهای دریافت) به مودم اعلام می کند. مودم نیز در ریزبُرشهای اختصاص یافته اطلاعات خود را ارسال می کند (و اگر به ریزبُرشهای بیشتری نیاز داشت، می تواند آنها را از طریق یکی از فیلدهای سرآیند درخواست کند).

اگر جوابی از منبع نرسید (که درخواست همزمان چند مودم می تواند یکی از غلتهای آن باشد)، مودم مدتی صبر کرده و دوباره اقدام خواهد کرد. اگر اقدام دوباره هم با شکست مواجه شد، مودم زمان انتظار را بیشتر می کند (بیاد دارید که این همان روش تخصیص بُرش زمانی در شبکه ALOHA است؛ از تکنیکهای اینترنت نمی توان استفاده کرد، چون مودم راهی برای گوش کردن به رسانه و تشخیص تصادم ندارد. در فصل ۴ این مبحث را مفصلتر بررسی خواهیم کرد).

مدیریت کانالهای دریافت از اینترنت بکلی متفاوت است. اول اینکه، در اینجا فقط یک فرستنده وجود دارد (منبع)، و بالطبع کشمکشی هم در میان نیست و منبع می تواند از تکنیکهای مالتی پلکس تقسیم زمانی آماری استفاده کند. دیگر اینکه، ترافیک دریافت از اینترنت معمولاً بسیار بیشتر از ترافیک ارسال به اینترنت است، و به همین دلیل از بسته هایی با طول ثابت (۲۰۴ بایت) استفاده می شود. قسمتی از این بسته سرآیندهاست (کُد تصحیح خطای Reed-Solomon و مانند آن)، و در نهایت ۱۸۴ بایت برای داده های کاربر باقی می ماند. این عدد برای سازگاری با تلویزیون دیجیتال MPEG-2 انتخاب شده، بنابراین فرمت کانالهای دریافت از اینترنت و تلویزیون یکسان است (شکل ۲-۴۹ را ببینید).

به فرآیند آماده سازی مودم برگردیم؛ همین که مودم فاصله خود را با منبع مشخص کرد، و کانالهای ارسال و



شکل ۲-۲۹. کانالهای ارسال و دریافت در منطقه آمریکای شمالی.

دریافت و ریزیرشها را گرفت، آماده است تا اطلاعات خود را بفرستد. اولین بسته‌ای که مودم می‌فرستد، بسته خاصیت حاوی درخواست یک آدرس IP از ISP با استفاده از پروتکل DHCP (که در فصل ۵ درباره آن توضیح خواهیم داد). همچنین وقت دقیق نیز از منبع پرسیده شد، و منبع به آن جواب می‌دهد.

در قدم بعد ایمنی اطلاعات بایستی تضمین شود. از آنجائیکه کابل مودم بین تعداد زیادی از افراد مشترک است، هر کسی می‌تواند اطلاعات تمام کاربران دیگر را بخواند. برای جلوگیری از سرک کشیدن دیگران، ترافیک در هر دو جهت (ارسال و دریافت) رمز می‌شود. بخشی از فرآیند آماده‌سازی مودم شامل ایجاد کلیدهای رمز است. شاید در نگاه اول این کار غیرممکن بنظر برسد؛ چگونه می‌توان در روز روشن و زیر نگاه هزاران غریبه کلیدهای رمز را رد و بدل کرد؟ این کار با استفاده از الگوریتمی بنام دیفی-هلمان (Diffie-Hellman) ممکن است، اما اجازه دهید توضیحات بیشتر را به فصل ۸ موکول کنیم.

در پایان، مودم هویت منحصر بفرد (شامل نام کاربر و کلمه عبور) خود را روی کانالی که اکنون از امنیت کافی برخوردار است، به ISP اعلام می‌کند. در اینجا فرآیند آماده‌سازی به پایان می‌رسد، و کاربر می‌تواند به کار عادی خود ادامه دهد.

به همین توضیح مختصر درباره مودمهای کابلی بسنده می‌کنیم، ولی می‌توانید اطلاعات تکمیلی را در (Adams and Dulchinos, 2001; Donaldson and Jones, 2001; and Dutta-Roy, 2001) بیابید.

۵-۷-۲ مودم کابلی یا ADSL ؟

کدامیک بهتر است: ADSL یا مودم کابلی؟ این مانند آنست که بپرسید کدام سیستم عامل، یا کدام زبان بهتر است. جواب به کسی که این سؤال را از او می‌پرسید، بستگی دارد. اما اجازه دهید ADSL و مودم کابلی را از چند نقطه نظر با هم مقایسه کنیم. هر دوی این سیستمها در بخش ستون فقرات از فیبر نوری استفاده می‌کنند، ولی بخش انتهایی آنها متفاوت است. مودم کابلی در بخش انتهایی (مدار پایانی) از کابل کواکسیال استفاده می‌کند، و ADSL از زوج-تابیده. از نظر تنوری، ظرفیت کواکس صدها برابر زوج-تابیده است، اما، تمام این ظرفیت در اختیار کاربر اینترنت نیست، چون پهنای باند کابل صرف چیزهای پدرنخوری (مثل برنامه‌های تلویزیونی) هم می‌شود.

در عمل، مقایسه ظرفیت این سیستمها بسیار مشکل است. در ADSL، شرکت سرویس دهنده ظرفیت خط را به صراحت مشخص می‌کند (مثلاً، 1 Mbps دریافت از اینترنت و 256 kbps ارسال به اینترنت)، و در اغلب موارد ۸۰٪ این ظرفیت نیز محقق می‌شود. اما شرکت‌های کابلی هرگز درباره ظرفیت سرویس خود صحبت نمی‌کنند، چون این ظرفیت به تعداد کاربرانی که در هر لحظه روی یک کابل هستند، بستگی دارد. گاهی کابل بهتر از ADSL است، و گاهی بدتر. اما چیزی را نمی‌توان از قبل پیش‌بینی کرد (و این از همه آزاردهنده‌تر است). در یک لحظه

کیفیت کابل خوب است، و لحظه بعد (وقتی یکی از آن خوره‌های اینترنت کامپیوتر خود را روشن می‌کند) غیر قابل تحمل.

در ADSL افزایش تعداد کاربران تأثیری روی کیفیت خط کاربران موجود نخواهد گذاشت، چون هر کاربر دارای یک خط مستقل است. در حالیکه در کابل این افزایش موجب افت کیفیت می‌شود. تنها راه مقابله با این مشکل هم انشعاب کابل‌های بیشتر از گره فیبر، و کم کردن تعداد کاربران هر خط است. اما این یعنی هزینه بیشتر، چیزی که صاحبان شرکت‌های کابلی در برابر آن مقاومت می‌کنند.

سیستم‌های تلفن همراه هم وضعیتی شبیه مودم کابلی دارد: گروهی از کاربران بطور مشترک از یک بخش خاص از پهنای باند استفاده می‌کنند. از آنجائیکه ترافیک صدا نسبتاً یکنواخت است، تقسیم پهنای باند بین کاربران (که با تکنیک‌های FDM و TDM انجام می‌شود) نیز یکنواخت و ثابت است. اما در ترافیک داده، تقسیم ثابت بهیچوجه کارایی ندارد. چون توزیع زمانی استفاده هر کاربر از کانال اختصاصی (روشی که در مودم کابلی بکار می‌رود) یکنواخت نیست، و موجب هدر رفتن منابع خواهد شد. با این همه، مودم کابلی حتی از این نظر نیز بیشتر شبیه تلفن همراه است تا تلفن ثابت.

سهولت دستیابی یکی دیگر از تفاوت‌های ADSL و مودم کابلی است. هر کاربری یک خط تلفن دارد، اما همه آنها آنقدر به ایستگاه تلفن نزدیک نیستند که بتوانند ADSL بگیرند. از طرف دیگر، کابل نیز چیزی نیست که همه داشته باشند، اما اگر از قبل کابل دارید و شرکت طرف قرارداد شما دسترسی اینترنت هم عرضه می‌کند، باید گفت شانس آورده‌اید. در اینجا دیگر فاصله تا گره فیبر یا منبع اهمیتی ندارد. باید خاطر نشان کرد که، از آنجائیکه کابل اساساً رسانه‌ای تلویزیونی است (و از ابتدا برای این منظور نصب شده)، در اغلب شرکت‌ها و دفاتر اداری وجود ندارد.

خطوط ADSL (بدلیل ماهیت نقطه-به-نقطه آنها) ذاتاً از ایمنی بالاتری نسبت به کابل برخوردارند. هر کاربری که به کابل دسترسی داشته باشد، می‌تواند تمام بسته‌های منتشر شده روی آنرا بخواند. البته همانطور که قبلاً گفتیم، تمام شرکت‌های معتبر سرویس‌های کابلی ترافیک را در هر دو جهت رمز می‌کنند. اما همین که کسی بتواند اطلاعات رمز شده شما را هم بگیرد، ضریب ایمنی را پائین می‌آورد.

سیستم تلفن عموماً قابل اعتمادتر از کابل است. برای مثال، شرکت‌های تلفن دارای سیستم‌های برق اضطراری هستند، و سرویس‌های آنها حتی در صورت بروز خاموشی کامل هم قطع نمی‌شود. اما در سیستم‌های کابلی، اگر برق هر یک از تقویت‌کننده‌های میانی در یکی از زنجیره‌ها قطع شود، ترافیک دریافت آن خط بکلی متوقف خواهد شد.

و بالاخره اینکه، اغلب شرکت‌هایی که سرویس ADSL ارائه می‌کنند، انتخاب ISP را بر عهده خود شما می‌گذارند (در برخی نقاط حتی اجبار قانونی برای این کار وجود دارد). در مورد شرکت‌های کابلی، اغلب خود آنها ارائه‌دهنده سرویس اینترنت هم هستند، و دست شما برای انتخاب ISP باز نیست. نتیجه نهایی این بحث آنست که، ADSL و کابل بیشتر به هم شبیه‌اند تا متفاوت. آنها سرویس‌های مشابهی ارائه می‌کنند، و با افزایش رقابت قیمت آنها هم احتمالاً روزی به یکدیگر نزدیکتر خواهد شد.

۸-۲ خلاصه

لایه فیزیکی اساس تمام شبکه‌هاست. طبیعت دو محدودیت بنیادی به تمام کانال‌های ارتباطی تحمیل کرده، و همین محدودیت‌هاست که پهنای باند آنها را مشخص می‌کند. این دو محدودیت عبارتند از: حد نایکوئیست (Nyquist limit) که با کانال‌های بدون نویز سروکار دارد، و حد شانون (Shannon limit) که به کانال‌های نویزدار مربوط می‌شود.

رسانه انتقال می تواند هدایت پذیر (guided) باشد یا هدایت ناپذیر (unguided). مهمترین رسانه های هدایت پذیر عبارتند از: زوج-تابیده، کابل کواکسیال، و فیبر نوری. رسانه های هدایت ناپذیر نیز عبارتند از: امواج رادیویی، مایکروویو، مادون قرمز، و لیزر. یکی از رسانه های انتقال رو به رشد ماهواره های مخابراتی (پویزه سیستم های مدار پائین - LEO) هستند.

سیستم تلفن یکی از کلیدی ترین اجزای شبکه های گسترده (WAN) است، که مهمترین عناصر آن عبارتند از: مدارهای پایانی (local loop)، ترانک ها (trunk)، و سوئیچ ها (switch). مدارهای پایانی مدارهای آنالوگ زوج-تابیده هستند، که برای انتقال داده های دیجیتال روی آنها باید از مودم (modem) استفاده کرد. ADSL با استفاده از تکنیک های مدولاسیون و تقسیم مدار پایانی به کانال های مجازی می تواند به سرعت 50 Mbps برسد. مدارهای پایانی بیسیم (WLL) یکی از تکنولوژی های جدیدیست که در آینده از آن (پویزه از LMDS) بیشتر خواهید شنید.

ترانک های شبکه تلفن دیجیتالی هستند، و از تکنیک های مالتی پلکس (شامل FDM، TDM و WDM) در آنها استفاده می شود. تکنیک های سوئیچینگ نیز بر دو نوع سوئیچینگ مداری (circuit switching) و سوئیچینگ بسته ای (packet switching) است، که هر دو اهمیت زیادی دارند.

برای کاربردهایی که تحرک زیادی دارند، سیستم تلفن ثابت چندان مناسب نیست. تلفن های همراه امروزه بطور گسترده ای برای ارتباطات صدا مورد استفاده قرار می گیرند، و در آینده نزدیک ترافیک داده نیز در آنها بحد قابل ملاحظه ای خواهد رسید. نسل اول تلفن های همراه آنالوگ بود، که عمدتاً به سیستم های AMPS متکی بود. نسل دوم تلفن همراه دیجیتال است، که در آن از سیستم های GSM، D-AMPS و CDMA استفاده می شود. نسل سوم تلفن های همراه نیز دیجیتال خواهد بود، و در آن CDMA پهن باند استفاده خواهد شد.

یکی از سیستم هایی که می توان از آن برای کاربردهای شبکه نیز بهره گرفت، تلویزیون کابلی (که از تلویزیون با آنتن مرکزی به سیستم های آمیخته فیبر-کواکس تکامل یافته) است. این سیستم پهنای باند بالقوه زیادی دارد، ولی پهنای باند واقعی آن به تعداد کاربران فعال (و اینکه مشغول چه کاری هستند) بستگی دارد.

مسائل

۱. ضرایب فوریه تابع $f(t) = t$ را محاسبه کنید ($0 \leq t \leq 1$).
۲. هر 1 msec از یک کانال بدون نویز 4-kHz نمونه برداری می شود. حداکثر نرخ داده این کانال چقدر است؟
۳. کانال های تلویزیونی 6 MHz پهنای دارند. اگر از یک سیگنال دیجیتال چهار سطحی استفاده کنیم، چند bit/sec می توان در این کانال مخابره کرد؟ فرض کنید کانال بدون نویز است.
۴. اگر یک سیگنال باینری در کانالی 3-kHz که نسبت سیگنال به نویز آن 20 dB است، مخابره شود، حداکثر نرخ داده قابل دستیابی چقدر است؟
۵. برای آن که بتوان کاربر T1 را روی یک خط 50-kHz قرار داد، نسبت سیگنال به نویز چقدر باید باشد؟
۶. فرق ستاره غیرفعال و تکرارکننده فعال در یک شبکه فیبر نوری چیست؟
۷. پهنای باند موجود در طیفی به پهنای $0.1 \mu\text{m}$ در طول موج $1 \mu\text{m}$ چقدر است؟
۸. می خواهیم یک سری تصاویر کامپیوتری اسکن شده را روی یک رشته فیبر نوری بفرستیم. وضوح هر تصویر 480×640 پیکسل، و هر پیکسل ۲۴ بیت است. تصاویر با سرعت ۶۰ صفحه بر ثانیه اسکن می شوند. پهنای باند مورد نیاز چقدر است؟ اگر از باند $1.30 \mu\text{m}$ استفاده کنیم، به چند میکرون از طول موج نیاز داریم؟
۹. آیا قضیه نایکونیست برای فیبرهای نوری هم صادق است، یا فقط برای کابلهای مسی کاربرد دارد؟
۱۰. در شکل ۲-۶، باند سمت چپ از بقیه باندها باریکتر است. چرا؟

۱۱. یک آنتن زمانی بهترین بهره را دارد که قطر آن معادل طول موج امواج رادیویی باشد. قطر قابل قبول آنتنها بین ۱ تا ۵ متر است. این قطر معادل کدام طیف فرکانسی است؟
۱۲. محوشدگی چندمسیره زمانی به حداکثر می رسد که دو موج با اختلاف فاز 180° درجه وارد گیرنده شوند. برای به حداکثر رسیدن محوشدگی چندمسیره در یک لینک مایکروویو 1-GHz بطول 50-km این مقدار چقدر باید باشد؟
۱۳. پرتو لیزری بقطر 1 mm روی آشکارسازی بقطر 1 mm که روی پشت بامی در فاصله 100 m قرار دارد، نشانه گرفته شده است. حداکثر انحراف زاویه ای (بر حسب درجه) چقدر باید باشد، تا پرتو لیزری هدف را گم نکند؟
۱۴. ۶۶ ماهواره پروژه ایریدیوم به شش کمربند بدور زمین تقسیم شده اند. در ارتفاعی که این ماهواره ها قرار دارند، دوره گردش مداری ۹۰ دقیقه است. متوسط زمان پاس کاری یک فرستنده زمینی بین دو ماهواره چقدر است؟
۱۵. ماهواره ای در مدار زمین ثابت با صفحه استوای زمین زاویه ϕ می سازد. آیا برای فردی که روی زمین در مدار ϕ درجه شمالی ایستاده، این ماهواره در آسمان ثابت بنظر می رسد؟ اگر نه، حرکت آنرا توضیح دهید.
۱۶. قبل از سال ۱۹۸۴ (وقتی هر ایستگاه پایانی با کد سه رقمی ناحیه و سه رقم اول شماره تلفن مشخص می شد) در سیستم تلفن چند کد ایستگاه پایانی می توانست وجود داشته باشد؟ کد ناحیه با عددی بین ۲ تا ۹ شروع می شد، رقم دوم می توانست ۰ یا ۱ باشد، و رقم سوم محدودیتی نداشت. دو رقم اول کد محلی نیز بایستی بین ۲ تا ۹، و رقم سوم می توانست هر عددی باشد.
۱۷. فقط با اطلاعاتی که در اینجا بدست آوردید، آیا می توانید بگویند حداکثر شماره تلفنهایی که (بدون تغییر در روش شماره گذاری یا اضافه کردن تجهیزات) می توان در ایالات متحده نصب کرد، چه تعداد است؟ آیا این تعداد شماره قابل دستیابی است؟ هر دستگاه فکس یا کامپیوتر را نیز یک تلفن در نظر بگیرید، و فرض کنید هر مشترک فقط یک دستگاه تلفن دارد.
۱۸. یک سیستم ساده تلفن را که در آن دو ایستگاه پایانی و یک ایستگاه بین شهری بوسیله خطوط دو-طرفه همزمان 1-MHz به هم متصل شده اند، در نظر بگیرید. هر دستگاه تلفن بطور متوسط در هر روز کاری ۸ ساعته ۴ تماس برقرار می کند، و زمان متوسط هر تماس ۶ دقیقه است. ده درصد تماسها راه دور هستند (یعنی از ایستگاه بین شهری رد می شوند). حداکثر تعداد شماره هایی که هر ایستگاه پایانی می تواند پشتیبانی کند، چقدر است؟ مدارها را 4-kHz در نظر بگیرید.
۱۹. یک شرکت تلفن منطقه ای ۱۰ میلیون مشترک دارد، که بوسیله زوج-تاییده به ایستگاه مرکزی متصل شده اند. متوسط طول مدارهای پایانی ۱۰ کیلومتر است. ارزش مس موجود در مدارهای پایانی این سیستم چقدر است؟ سطح مقطع سیمها را دایره ای بقطر 1 mm، چگالی مس را 9.0 gr/cm^3 و ارزش هر کیلوگرم مس را 3 دلار در نظر بگیرید.
۲۰. یک خط لوله نفت سیستمی یکطرفه است، یا دو-طرفه ناهمزمان، یا دو-طرفه همزمان، یا هیچکدام؟
۲۱. قیمت میکروپروسسورهای سریع آنقدر کاهش یافته، که می توان در هر دستگاه مودم یک میکروپروسسور قرار دارد. این کار چه تأثیری روی مقابله با خطاهای خطوط تلفن دارد؟
۲۲. دیاگرام فلکی شکل ۲-۲۵ چهار نقطه داده در مختصات $(1, 1)$ ، $(1, -1)$ ، $(-1, 1)$ و $(-1, -1)$ دارد. مودمی با این پارامترها در 1200 baud به چه سرعتی (bps) می تواند دست یابد؟
۲۳. مودمی با دیاگرام فلکی شبیه شکل ۲-۲۵ دارای نقاط داده ای در مختصات $(0, 1)$ و $(0, 2)$ است. این مودم

- از مدولاسیون فاز استفاده می‌کند، یا مدولاسیون دامنه؟
۲۴. در یک دیاگرام فلکی تمام نقاط روی دایره‌ای به مرکز مبدأ مختصات واقع شده‌اند. مدولاسیون این مودم چیست؟
۲۵. یک مودم دو-طرفه همزمان QAM-64 از چند فرکانس استفاده می‌کند؟
۲۶. در یک سیستم ADSL که از DMT استفاده می‌کند، 3/4 کانالهای موجود به لینک دریافت اختصاص داده شده است. هر کانال از مدولاسیون QAM-64 استفاده می‌کند. ظرفیت لینک دریافت چقدر است؟
۲۷. در سیستم LMSD چهار قطاعی شکل ۲-۳، هر قطاع یک کانال اختصاصی 36-Mbps دارد. طبق تئوری صف، اگر 50% کانال پر باشد، زمان انتظار در صف معادل زمان بار شدن است. در چنین شرایطی، بار شدن یک صفحه وب 5-KB چقدر طول خواهد کشید؟ بار شدن این صفحه روی یک خط ADSL با سرعت 1 Mbps چقدر طول می‌کشد؟ یا یک مودم 56-kbps چقدر؟
۲۸. ده سیگنال، که هر کدام به پهنای باند 4000 Hz نیاز دارند، با استفاده از FDM روی یک کانال مالتی پلکس شده‌اند. حداقل پهنای باند مورد نیاز این کانال چقدر است؟ پهنای باندهای محافظ را 400 Hz در نظر بگیرید.
۲۹. چرا زمان نمونه برداری PCM در 125 μsec ثابت شده است؟
۳۰. درصد سرآیند یک کاربر T1 (درصدی از 1.544 Mbps که بکار داده‌های کاربر نمی‌آید) چقدر است؟
۳۱. حداکثر نرخ داده یک کانال بدون نویز 4-kHz را با استفاده از تکنیکهای زیر مقایسه کنید:
(الف) کدگذاری آنالوگ (مثلاً، QPSK) با 2 bits/sample.
(ب) سیستم T1 PCM.
۳۲. اگر یک سیستم T1 دچار لغزش شود، برای سنکرون شدن مجدد از اولین بیت هر فریم استفاده می‌کند. برای سنکرون شدن مجدد با احتمال خطای 0.001، چند فریم باید بررسی شود؟
۳۳. فرق بخش دمودلاتور یک مودم با بخش دکودر یک کدک چیست (و آیا اساساً فرقی دارند)؟ توجه داشته باشید که هر دوی اینها سیگنالهای آنالوگ را به دیجیتال تبدیل می‌کنند.
۳۴. سیگنالی بصورت دیجیتال روی یک کانال بدون نویز 4-kHz (با یک نمونه در هر 125 μsec) فرستاده می‌شود. با هر یک از روشهای کدگذاری زیر چند بیت در ثانیه ارسال می‌شود:
(الف) استاندارد CCITT 2.048 Mbps.
(ب) سیستم DPCM با مقدار نسبی سیگنال 4-bit.
(ج) مدولاسیون دلتا.
۳۵. یک سیگنال سینوسی کامل با دامنه A با استفاده از مدولاسیون دلتا (با x samples/sec) کد شده است. خروجی 1+ معادل $A/8$ تغییر در سیگنال ورودی، و خروجی 1- معادل $A/8$ تغییر در سیگنال ورودی است. بیشترین فرکانسی که این سیستم می‌تواند بدون خطای تجمعی تعقیب کند، چقدر است؟
۳۶. ساعت‌های SONET خطایی معادل 1 در 10^9 دارند. چه مدت طول می‌کشد، تا این اختلاف باندازه 1 بیت شود؟ عوارض جانبی این پدیده چیست؟
۳۷. در شکل ۲-۳۷، نرخ داده کاربر OC-3 از 148.608 Mbps شروع شده است. نشان دهید این عدد چگونه از پارامترهای SONET OC-3 بدست آمده است.
۳۸. سیستم SONET برای انطباق با سرعت‌های پائین‌تر از STS-1 از روشی بنام انشعابات مجازی (Virtual Tributaries - VT) استفاده می‌کند. یک VT عبارتست از یک بار جزئی، که می‌توان آنرا به همراه بارهای

- جزئی دیگر در یک فریم STS-1 قرار داد. VT1.5 از ۳ ستون، VT2 از ۴ ستون، VT3 از ۶ ستون و VT6 از ۱۲ ستون فریم STS-1 استفاده می کنند. کدام VT با هر یک از سرویسهای زیر منطبق است؟
- (الف) سرویس DS-1 (1.544 Mbps)
- (ب) سرویس European CEPT-1 (2.048 Mbps)
- (ج) سرویس DS-2 (6.312 Mbps)
۳۹. تفاوت بنیادی سونیچینگ مداری یا سونیچینگ بسته ای چیست؟
۴۰. پهنای باند کاربر یک اتصال OC-12c چقدر است؟
۴۱. سه شبکه سونیچینگ بسته ای هر کدام ۳ گره دارند. شبکه اول دارای توپولوژی ستاره (با سونیچ مرکزی) است، شبکه دوم حلقه (دوطرفه) است، و شبکه سوم اتصالات کامل داخلی دارد (یعنی هر گره مستقیماً به تمام گره های دیگر متصل است). بهترین، بدترین و متوسط پرش (hop) در ارتباط از یک نقطه به نقطه دیگر در هر یک از این شبکه ها چیست؟
۴۲. زمان تأخیر ارسال یک پیام x -bit در مسیری با k پرش در یک شبکه سونیچینگ مداری و یک شبکه سونیچینگ بسته ای (با بار کم) را با یکدیگر مقایسه کنید. زمان برقراری مدار را d ثانیه، زمان تأخیر در هر پرش را d ثانیه، اندازه هر بسته را p بیت، و نرخ انتقال داده را b pbs در نظر بگیرید. در چه شرایطی تأخیر ارسال شبکه سونیچینگ بسته ای کمتر است؟
۴۳. فرض کنید می خواهیم x بیت اطلاعات را در یک شبکه سونیچینگ بسته ای با k پرش، بصورت بسته هایی با p بیت داده و h بیت سرآیند (با این فرض که $x \gg p + h$) منتقل کنیم. نرخ انتقال داده خطوط b pbs، و زمان تأخیر انتشار در آنها قابل صرف نظر کردن است. چه مقداری از p تأخیر کلی را به حداقل می رساند؟
۴۴. در یک سیستم تلفن همراه با سلولهای شش ضلعی، استفاده از باندهای فرکانسی مشابه در سلولهای مجاور ممنوع است. اگر ۸۴۰ باند فرکانسی داشته باشیم، در هر سلول از چند فرکانس می توان استفاده کرد؟
۴۵. طرح کلی سلولهای یک شبکه تلفن همراه بندرت مانند شکل ۲-۴۱ منظم است؛ حتی شکل هر سلول نیز منظم نیست. یک علت برای این وضعیت بیاورید.
۴۶. برای پوشش دادن به شهری با مساحت 120 km^2 ، به چه تعداد سلول PCS با قطر 100 m نیاز داریم؟ (تخمین بزنید.)
۴۷. گاهی هنگام عبور یک کاربر تلفن همراه از سلولی به سلول دیگر، با وجود اینکه تمام فرستنده ها و گیرنده ها بخوبی کار می کنند، ارتباط ناگهان قطع می شود. چرا؟
۴۸. کیفیت صدای D-AMPS بسیار پایتتر از GSM است. آیا این بخاطر اجبار D-AMPS در حفظ سازگاری با AMPS است (در حالیکه GSM چنین محدودیتی ندارد)؟ یا علت دیگری دارد؟
۴۹. حداکثر تعداد کاربران همزمان در یک سلول D-AMPS را محاسبه کنید. آیا چنین محاسبه ای برای GSM هم امکان دارد؟ علت را توضیح دهید.
۵۰. فرض کنید سه ایستگاه A ، B و C در یک سیستم CDMA (با توابعهای چپب شکل ۲-۴۵ ب) همزمان اقدام به ارسال بیت های ۰ می کنند. توالی چپب حاصله چیست؟
۵۱. در بحث متعامد بودن بردارهای توالی چپب CDMA، گفتیم که اگر $S \cdot T = 0$ ، آنگاه $S \cdot \bar{T} = 0$ ثابت کنید.
۵۲. اجازه دهید متعامد بودن بردارهای توالی چپب CDMA را به روشی دیگر بیان کنیم: هر بیت در یک جفت توالی یا یکسان هستند، یا نیستند. متعامد بودن بردارها را با استفاده از اصطلاحات یکسان بودن و یکسان نبودن توضیح دهید.

۵۳. یک گیرنده CDMA توالی $(+1 +1 -3 +1 -1 -3 +1 +1)$ را دریافت می کند. با فرض توابعهای چپ شکل ۲-۴۵ (ب)، تعیین کنید کدام ایستگاه ها، چه بیت هایی را ارسال کرده اند؟
۵۴. شبکه تلفن در بخش انتهایی دارای توپولوژی ستاره است، که در آن تمام انشعابات به ایستگاه پایانی ختم می شوند. در حالیکه در تلویزیون کابلی، یک کابل مشترک مانند ماری بین مشترکین مختلف خزیده است. فرض کنید در آینده در شبکه های کابلی بجای کابل های ماری از فیبر نوری 10 Gbps استفاده شود. آیا با چنین سیستمی می توان مدل شبکه تلفن (یک خط مستقل از هر مشترک به ایستگاه مرکزی) را شبیه سازی کرد؟ اگر پاسخ مثبت است، هر فیبر چند کاربر تلفن می تواند داشته باشد؟
۵۵. سیستم های تلویزیون کابلی معمولاً دارای ۱۰۰ کانال تجاری هستند، که بطور متناوب برنامه و آگهی پخش می کنند. این سیستم بیشتر شبیه TDM است یا FDM؟
۵۶. یک شرکت کابلی تصمیم می گیرد به ۵۰۰۰ مشترک خود سرویس اینترنت ارائه دهد. این شرکت از کابل های کواکسیال استفاده می کند، که هر کابل می تواند تا 100 Mbps روی کانال دریافت از اینترنت ظرفیت داشته باشد. شرکت مزبور برای جذب مشتریان تصمیم می گیرد که تا دریافت 2 Mbps را برای هر مشترک تضمین کند. توضیح دهید این شرکت برای رسیدن به هدف فوق چه کاری باید انجام دهد؟
۵۷. با توجه به تخصیص فرکانس نشان داده شده در شکل ۲-۴۸ و اطلاعات داده شده در متن کتاب، یک سیستم کابلی چه مقدار (Mbps) از ظرفیت را به ارسال به اینترنت و چه مقدار را به دریافت از اینترنت اختصاص می دهد؟
۵۸. اگر سیستم کابلی کاملاً بیکار باشد، کاربر با چه سرعتی می تواند اطلاعات را دریافت کند؟
۵۹. مالتی پلکس کردن استریم های متعدد STS-1 (که به آنها انشعاب گفته می شود) نقش مهمی در سیستم SONET بازی می کند. یک مالتی پلکسر 3:1 سه ورودی STS-1 را در یک خروجی STS-3 مالتی پلکس می کند. اینکار بصورت بایت به بایت انجام می شود، یعنی سه بایت اول خروجی بترتیب بایتهای اول انشعابهای 1، 2 و 3 هستند؛ سه بایت دوم خروجی بترتیب بایتهای دوم انشعابهای 1، 2 و 3؛ و الی آخر. برنامه ای بنویسید که این مالتی پلکسر 3:1 را شبیه سازی کند. برنامه شما باید پنج روال داشته باشد: روال اصلی (که چهار روال دیگر را اجرا می کند)، یک روال برای هر یک از انشعابهای STS-1 (مجموعاً سه روال)، و یکی برای مالتی پلکسر. هر روال انشعاب یک فریم STS-1 را از فایلی بطول ۸۱۰ بایت خوانده، و این فریمها را (بایت به بایت) به روال مالتی پلکسر می فرستد. روال مالتی پلکسر این بایتها را خوانده، و یک فریم STS-3 را (بایت به بایت) روی خروجی استاندارد (stdout) می نویسد. برای ارتباط بین پردازشها از پایپ (pipe) استفاده کنید.

لایه پیوند داده



در این فصل اصول طراحی لایه دوم، لایه پیوند داده (data link layer)، را بررسی خواهیم کرد، و طی آن با الگوریتمهای لازم برای دستیابی به یک ارتباط قابل اطمینان و کارا بین دو کامپیوتر همسایه (در لایه پیوند داده) آشنا خواهیم شد. منظور از دو کامپیوتر همسایه، کامپیوترهایی هستند که یک کانال ارتباطی سیم-مانند (کابل کواکسیال، خط تلفن، و یا ارتباط بیسیم) بین آنها برقرار است. خصلت بنیادی یک کانال «سیم-مانند» اینست که بیت‌ها دقیقاً با همان نظمی که فرستاده می‌شوند، در گیرنده دریافت شوند.

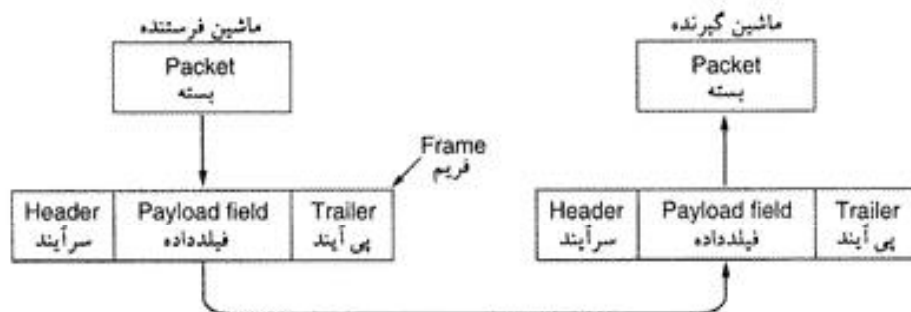
شاید در نگاه اول این خصلت آنقدر ساده و ابتدایی بنظر برسد، که فکر کنید چه نیازی به الگوریتم و نرم‌افزار هست: ماشین A بیت‌ها را می‌فرستد، و ماشین B آنها را می‌گیرد. متأسفانه، مسئله بهمین سادگی نیست، چون مدارهای مخابراتی پر از نویز و خطا هستند. علاوه بر آن ظرفیت کانالهای مخابراتی نامحدود نیست، و بین ارسال و دریافت بیت‌ها یک تأخیر زمانی نیز وجود دارد. این محدودیت‌ها تأثیر جدی روی کارایی سیستمهای انتقال داده می‌گذارند. پروتکل‌های مخابراتی (که موضوع اصلی این فصل هستند) باید تمام این ملاحظات را در نظر بگیرند. بعد از آشنایی با نکات کلیدی در طراحی لایه پیوند داده، پروتکل‌های این لایه را بررسی خواهیم کرد. برای شروع خصلت خطاهای کانالهای مخابراتی، منشأ آنها و نحوه کشف و رفع این خطاها را بررسی کرده، و سپس پروتکل‌های لازم برای حل آنها را مورد مطالعه قرار می‌دهیم. در پایان، صحت این مدل‌ها را بررسی کرده، و چند نمونه از پروتکل‌های واقعی لایه پیوند داده ارائه خواهیم کرد.

۱-۳ ملاحظات طراحی لایه پیوند داده

لایه پیوند داده وظایف خاصی دارد که باید انجام دهد. این وظایف عبارتند از:

۱. ارائه سرویسهای مشخص به لایه شبکه.
۲. مدیریت خطاهای انتقال.
۳. تنظیم جریان داده‌ها (بگونه‌ایکه گیرنده‌های کُند زیر بمباران فرستنده‌های سریع غرق نشوند).

برای رسیدن به این اهداف، لایه پیوند داده بسته‌های رسیده از لایه شبکه را گرفته و آنها بصورت فریم (frame) در می‌آورد. هر فریم سه قسمت دارد: سرآیند (header)، داده اصلی، و پی‌آیند (trailer)؛ شکل ۱-۳ را ببینید. مدیریت فریم‌ها کلیدی‌ترین وظیفه لایه پیوند داده است، که در بخشهای آینده بتفصیل درباره آن صحبت خواهیم کرد

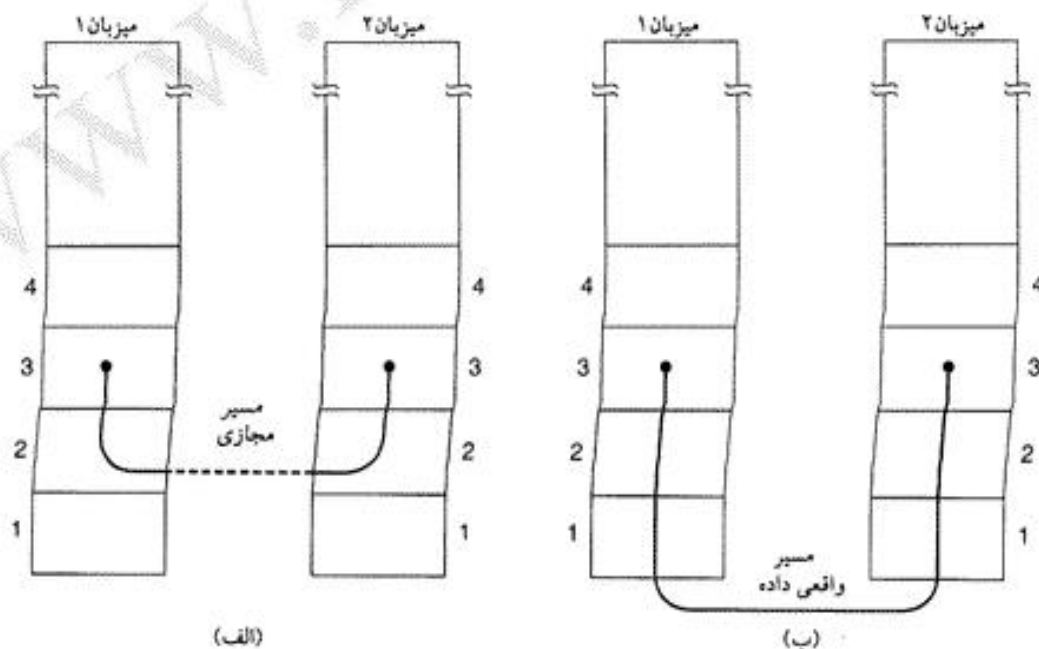


شکل ۳-۱. رابطه بسته و فریم.

با اینکه این فصل منحصرأ درباره لایه پیوند داده و پروتکل های آن است، اصولی که در اینجا خواهید دید (مانند کنترل خطا و کنترل جریان داده) در لایه های دیگر (مانند لایه انتقال) نیز کاربرد دارند. در حقیقت، در بسیاری از شبکه ها این کارکردها را فقط در لایه های بالاتر (نه در لایه پیوند داده) می توانید پیدا کنید. اما صرف نظر از اینکه آنها را کجا می توان پیدا کرد، اصول کار یکسان است و اهمیتی ندارد که در این فصل آنها را بررسی کنیم یا فصلهای دیگر. تنها مزیت لایه پیوند داده آنست که این تکنیکها در این لایه ساده تر و واضح ترند، بنابراین بخوبی می توان آنها را مطالعه کرد.

۱-۱-۳ سرویسهایی که به لایه شبکه داده می شود

وظیفه لایه پیوند داده ارائه سرویس به لایه شبکه است. مهمترین این وظایف عبارتست از انتقال داده ها از لایه شبکه ماشین مبدأ به لایه شبکه ماشین مقصد. در لایه شبکه ماشین مبدأ چیزی هست بنام پروسس، که تعدادی بیت را به لایه پیوند داده می دهد تا به مقصدی خاص منتقل کند. وظیفه لایه پیوند داده ماشین مبدأ انتقال این بیت ها به ماشین مقصد، و رساندن آنها بدست لایه شبکه مقابل است؛ شکل ۳-۲ (الف) را ببینید. البته مسیری که این بیت ها



شکل ۳-۲. (الف) ارتباط مجازی. (ب) ارتباط واقعی.

واقعاً طی می کنند، مانند شکل ۳-۲ (ب) است، ولی ساده ترست تصور کنیم دو پروتکل در لایه پیوند داده آنها را بین خود رد و بدل می کنند. بهمین دلیل در این فصل همه جا از مدل شکل ۳-۲ (الف) استفاده خواهیم کرد. لایه پیوند داده را می توان بگونه ای طراحی کرد که سرویسهای مختلفی ارائه کند، که این سرویسها از سیستمی به سیستم دیگر متفاوت است. معقولترین این سرویسها عبارتند از:

۱. سرویس غیرمتصل بدون تصدیق دریافت (unacknowledged connectionless).
۲. سرویس غیرمتصل با تصدیق دریافت (acknowledged connectionless).
۳. سرویس اتصال-گرا با تصدیق دریافت (acknowledged connection-oriented).

اجازه دهید این سرویسها را یکی یکی بررسی کنیم.

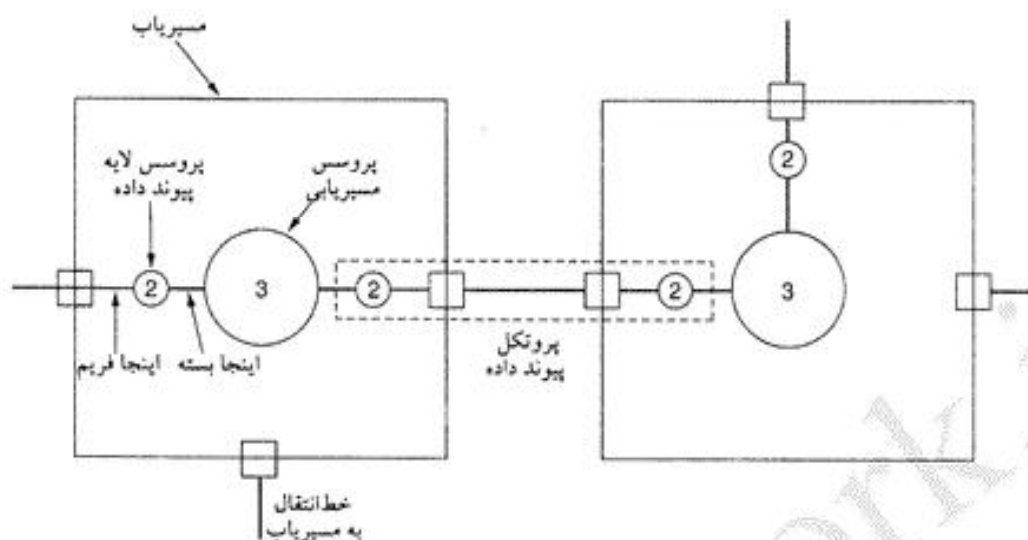
در سرویس غیرمتصل بدون تصدیق دریافت ماشین مبدأ فریمهای مستقلی را به ماشین مقصد می فرستد، بدون اینکه منتظر تصدیق دریافت آنها از طرف ماشین مقصد بماند. هیچ اتصال منطقی بین دو ماشین برقرار نمی شود، پس نیازی به قطع اتصال هم نیست. اگر فریمی در اثر نویز خط از بین برود، هیچ کوششی برای تشخیص این موضوع و مقابله با آن در لایه پیوند داده صورت نمی گیرد. این سرویس برای مواقعی مناسب است که نرخ خطا بسیار پائین باشد، و در این حالت مقابله با خطا به لایه های بالاتر واگذار می شود. این سرویس برای ترافیک زمان-واقعی (مانند سرویس صدا)، که در آن دیر رسیدن بدتر از نرسیدن است، نیز مناسب است. در اغلب LAN ها نیز لایه پیوند داده از سرویسهای غیرمتصل بدون تصدیق دریافت استفاده می کند.

سرویس بعدی که قابلیت اعتماد بیشتری دارد، سرویس غیرمتصل با تصدیق دریافت است. در این سرویس نیز هیچ اتصال منطقی بین مبدأ و مقصد وجود ندارد، ولی دریافت فریمها از سوی ماشین مقصد تصدیق می شود. بدین ترتیب، فرستنده می تواند پی ببرد که آیا فریمها بدرستی دریافت شده اند یا خیر. اگر فریمی در مدت زمان معین به مقصد نرسد، می توان آنرا دوباره ارسال کرد. این سرویس برای کانالهای غیر قابل اعتماد (مانند سیستمهای بیسیم) مناسب است.

لازمست تأکید کنیم که توجه به تصدیق دریافت در لایه پیوند داده فقط برای بهینه سازی سیستم است و هیچ الزامی در آن نیست، چون این کار را همیشه می توان در لایه شبکه انجام داد. اگر تصدیق دریافت در زمان مشخص از راه نرسد، فرستنده می تواند بسته را دوباره ارسال کند. مشکل اینجااست که فریمها معمولاً طول مشخصی دارند، در حالیکه بسته ای که لایه شبکه می فرستد چنین نیست. اگر بسته ای به، مثلاً، ۱۰ فریم شکسته شود، و ۲۰ درصد این فریمها در راه گم شوند، زمان ارسال بسته بسیار طولانی خواهد شد. اما اگر برای هر فریم تصدیق دریافت درخواست شود، این کار سریعتر می شود. در کانالهای قابل اعتماد مانند فیبر نوری، بار اضافی چنین پروتکل های سختگیرانه ای در لایه پیوند داده غیر ضروری است، اما در محیطهای ذاتاً پرنویز مانند بیسیم ارزشش را دارد.

بهترین سرویسی که لایه پیوند داده می تواند به لایه شبکه بدهد، سرویس اتصال-گرا (connection-oriented) است. در این سرویس قبل از شروع ارسال داده از مبدأ به مقصد، یک اتصال بین آنها برقرار می شود. هر فریمی که روی این اتصال فرستاده می شود شماره گذاری شده است، و لایه پیوند داده دریافت آنها را نیز تضمین می کند. همچنین تضمین می شود که هر فریم فقط یک بار (و به همان ترتیب ارسال) دریافت شود. اما در سرویسهای غیرمتصل، می توان انتظار داشت که بسته ای چندین بار ارسال (و در نتیجه چندین بار هم دریافت) شود. سرویس اتصال-گرا استریم قابل اعتمادی از بیت ها را در اختیار لایه شبکه می گذارد.

ارسال داده ها در سرویس اتصال-گرا سه مرحله دارد. در مرحله اول اتصال برقرار شده، و متغیرهای لازم (برای شمارش فریمها، و اینکه کدام فریمها دریافت شده اند و کدامها خیر) ست می شوند. در مرحله دوم، فریمها منتقل می شوند. و در مرحله آخر، اتصال قطع شده و منابع آن (متغیرها و بافرها) آزاد می شود.



شکل ۳-۳. محل فعالیت لایه پیوند داده.

اجازه دهید یک مثال بزنیم: یک زیرشبکه WAN متشکل از چند مسیریاب که با خطوط نقطه-به-نقطه تلفن به یکدیگر متصل شده اند، را در نظر بگیرید. وقتی یک فریم به مسیریاب می رسد، سخت افزار (با استفاده از تکنیکهایی که در همین فصل خواهید دید) آنرا از نظر خطا چک می کند، و سپس به نرم افزار لایه پیوند داده (که می تواند روی چیپهای کارت شبکه قرار داشته باشد) تحویل می دهد. نرم افزار لایه پیوند داده فریم را چک می کند تا مطمئن شود همان چیزیست که باید باشد، و اگر چنین بود، قسمت داده اصلی آنرا به نرم افزار مسیریابی می دهد. نرم افزار مسیریابی مسیر خروجی مناسب را تعیین کرده، و بسته را به لایه پیوند داده پس می دهد تا ارسال شود (شکل ۳-۳ را ببینید).

نرم افزارهای مسیریابی معمولاً حوصله بسته هایی که مدام گم می شوند را ندارند، و دوست دارند بسته ها درست و مرتب روی خطوط نقطه-به-نقطه تحویل شوند. این دیگر بر عهده پروتکل لایه پیوند داده است که خطوط پرنویز و غیر قابل اعتماد را بصورتی مطمئن (یا نسبتاً مطمئن) در آورد. با اینکه در شکل ۳-۳ نرم افزار لایه پیوند داده (در هر مسیریاب) در دو نقطه دیده می شود، اما این در واقع یک پروکس واحد است که (به کمک جدول ها و ساختمان داده های مختلف) تمام کارها را انجام می دهد و تمام خطوط را کنترل می کند.

۲-۱-۳ فریم بندی

لایه پیوند داده به لایه شبکه سرویس می دهد، و خود نیز از سرویسهای لایه فیزیکی استفاده می کند. چیزی که لایه فیزیکی می گیرد، استریمی است از بیت ها که باید آنرا به طرف مقابل تحویل دهد. هیچ تضمینی وجود ندارد که این استریم سالم و عاری از خطایه مقصد برسد. تعداد بیهیهای رسیده می تواند کمتر، مساوی یا بیشتر از بیت های ارسال شده باشد، و یا حتی مقدار برخی از آنها تغییر کرده باشد. این بر عهده لایه پیوند داده است که خطاها را کشف کرده، و در صورت لزوم آنها را برطرف کند.

یکی از روشهای متداول اینست که استریم بیت ها در لایه پیوند داده به چند فریم شکسته شده، و برای هر فریم جمع تطبیقی (checksum) محاسبه شود. (الگوریتمهای جمع تطبیقی را در همین فصل خواهید دید). وقتی فریمها به مقصد می رسند، جمع تطبیقی آنها مجدداً محاسبه شده و با جمع تطبیقی مبدأ (که به انتهای فریم ضمیمه شده) مقایسه می شود. اگر این دو یکی نباشند، لایه پیوند داده متوجه می شود که خطایی در فریم رخ داده، و به

سراغ روشهای مقابله با خطا می رود (که یکی از این روشها می تواند دور انداختن فریم، و درخواست ارسال مجدد آن باشد).

شکستن استریم بیت ها به فریم (که به آن فریم بندی - framing - گفته می شود) از آنچه در نگاه اول بنظر می رسد، مشکلتر است. یکی از روشهای فریم بندی می تواند انداختن فاصله زمانی در نقاطی از استریم بیت ها باشد (مانند فاصله انداختن بین کلمات متن). ولی در شبکه ها پندرت زمانبندی وجود دارد، و امکان دارد این فاصله ها از بین بروند و یا بیشتر شوند.

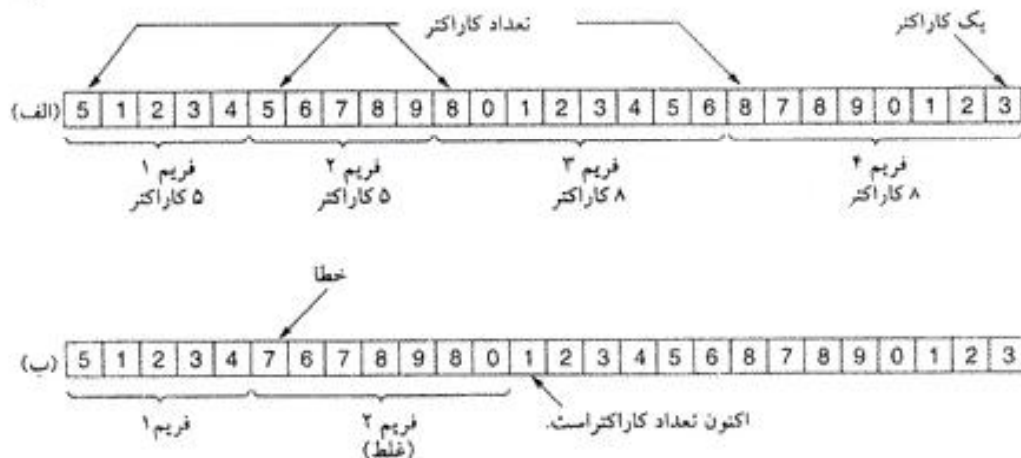
از آنجائیکه تکیه بر زمانبندی برای تعیین ابتدا و انتهای فریمها بسیار خطرناک است، روشهای دیگری برای اینکار ابداع شده، که در این قسمت با چهار تا از آنها آشنا می شوید:

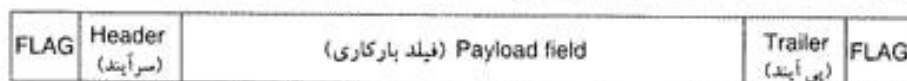
۱. شمارش کاراکترها.
۲. بایت های پرچم، با لاگذاری بایت.
۳. پرچمهای شروع و پایان، با لاگذاری بیت.
۴. حالت های غیرمجاز گذگذاری لایه فیزیکی.

در اولین روش فریم بندی تعداد کاراکترهای فریم در یکی از فیلدهای سرآیند آن نوشته می شود. وقتی این فریم به مقصد می رسد، لایه پیوند داده می تواند به کمک این فیلد ابتدا و انتهای فریم را مشخص کند. در شکل ۳-۴ (الف) چهار فریم با تعداد کاراکترهای ۵، ۵، ۸ و ۸ را می بینید.

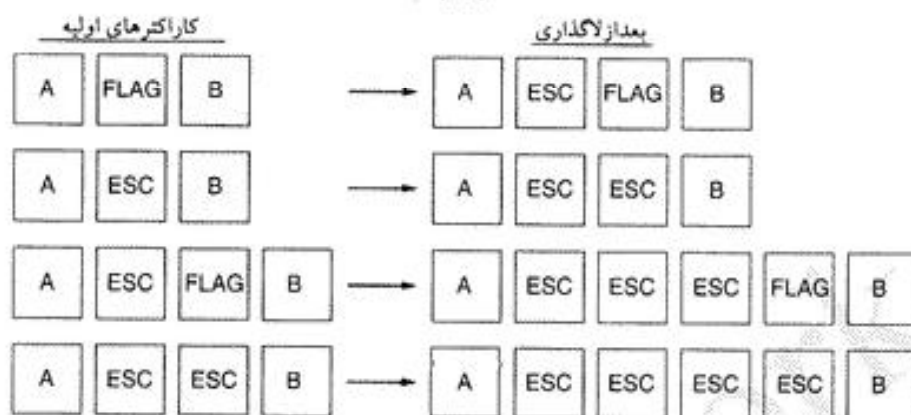
اشکال این روش آنست که فیلد تعداد کاراکترها نیز می تواند دچار خطا شود. برای مثال در شکل ۳-۴ (ب)، فیلد تعداد کاراکترها در فریم دوم از ۵ به ۷ تبدیل شده است، و ماشین مقصد دیگر قادر نیست فریمهای بعدی را بدرستی بخواند (چون قادر نیست ابتدای آنها را تشخیص دهد). حتی اگر جمع تطبیقی اشتباه باشد و ماشین مقصد متوجه باشد که خطایی رخ داده، باز هم تشخیص نقطه شروع بعدی برای آن غیرممکن است. درخواست ارسال مجدد نیز کمکی نمی کند، چون ماشین مقصد نمی داند فریمها تا کجا درست بوده، و اشتباه از کجا رخ داده است و نیاز به ارسال مجدد دارد.

مشکل سنکرون شدن مجدد مبدأ و مقصد بعد از بروز خطا در روش دوم فریم بندی (بایت های پرچم، با لاگذاری بایت) حل شده است، بدین ترتیب که هر فریم با تعدادی بایت خاص شروع و پایان می یابد. در گذشته، بایتهای شروع و پایان متفاوت بودند، ولی در سالهای اخیر از بایتهای یکسانی بعنوان بایت پرچم (flag byte) در





(الف)



(ب)

شکل ۵-۳. (الف) تعیین ابتدا و انتهای فریم با استفاده از بایت پرچم. (ب) چهار توالی

بایت قبل و بعد از لاگذاری بایت.

شروع و پایان فریم استفاده می شود. این بایتها را در شکل ۵-۳ (الف) با عنوان FLAG ملاحظه می کنید. در این روش اگر گیرنده همزمانی خود با فرستنده را از دست بدهد، فقط کافیسیت با جستجوی بایت پرچم انتهای فریم فعلی را پیدا کند. دو بایت پرچم که پشت سر هم بیایند، بمعنای پایان یک فریم و شروع فریم بعدی هستند. یکی از مشکلات جدی این روش آنست که طرح بیت بایت پرچم می تواند در داده های اصلی نیز وجود داشته باشد (بویژه اگر اطلاعات از نوع برنامه های اجرایی یا اعداد اعشاری باشد). این وضعیت گیرنده را به اشتباه خواهد انداخت. یکی از راه های حل این وضعیت آنست که پروتکل لایه پیوند داده در سمت فرستنده قبل از هر توالی بیت پرچم که در داده اصلی ظاهر می شود، یک بایت گریز (escape byte) خاص قرار دهد. لایه پیوند داده مقصد این بایت ها را حذف کرده، و داده های اصلی را به لایه شبکه تحویل می دهد. به این تکنیک لاگذاری بایت (byte stuffing) یا لاگذاری کاراکتر (character stuffing) گفته می شود. با این روش بایت پرچم بآسانی قابل تشخیص است، چون قبل از آن بایت گریز وجود ندارد.

اما حالا سؤال دیگری پیش می آید: اگر در وسط داده اصلی طرحی مشابه بایت گریز وجود داشت، چه اتفاقی می افتد؟ جواب اینست که قبل از این بایت هم یک بایت گریز قرار داده می شود؛ بعبارت دیگر دو بایت گریز پشت سر هم یعنی یک بایت گریز در داده اصلی. در شکل ۵-۳ (ب) چند نمونه از حالت هایی که می تواند پیش آید، آورده شده است. در هر مورد آن چیزی که گیرنده می گیرد، دقیقاً مشابه آن چیزیست که فرستنده ارسال کرده است. تکنیک لاگذاری بایت که در شکل ۵-۳ نشان داده شده، شکل ساده شده آن چیزیست که در پروتکل PPP (یکی از مهمترین پروتکل های ارتباط با اینترنت در کامپیوترهای شخصی) مورد استفاده قرار می گیرد. (بعدها در همین فصل درباره PPP صحبت خواهیم کرد.)

یکی از معایب بزرگ فریم بندی بایت های پرچم با لاگذاری بایت وابستگی شدید آن به کاراکترهای ۸-بیتی است، و همانطور که می دانید تمام گدها ۸-بیتی نیستند (برای مثال، در استاندارد یونی کد از کاراکترهای ۱۶-بیتی استفاده می شود). فرض ۸-بیتی بودن کاراکترها در مکانیزم فریم بندی یکی از مشکلات جدی آن محسوب می شود، به همین دلیل روش جدیدی که در آن طول کاراکتر می تواند متغیر باشد، ابداع شده است.

در این روش جدید طول کاراکترها اهمیتی ندارد، و فریمها می توانند تعداد بیتهای دلخواه داشته باشند. طرز کار این تکنیک جدید چنین است: هر فریم با طرح بیت خاصی (01111110 - که در واقع یک بایت پرچم است) شروع می شود. هر گاه لایه پیوند داده در سمت فرستنده پنج 1 پشت سر هم در داده اصلی دید، بطور خودکار یک 0 بعد از آن قرار می دهد. این روش، که به آن لاگذاری بیت (bit stuffing) گفته می شود، بسیار شبیه لاگذاری بایت است. وقتی گیرنده پنج 1 متوالی ببیند که یک 0 پشت سر آنها آمده، بطور خودکار این 0 را حذف می کند. لاگذاری بیت نیز مانند لاگذاری بایت یکلی از دید لایه شبکه در هر دو کامپیوتر پنهان (شفاف) است. اگر در داده کاربر طرح بیت 01111110 وجود داشته باشد، لایه پیوند داده فرستنده آنرا به 011111010 تبدیل می کند، و در سمت گیرنده این 0 اضافی حذف شده و طرح بیت 01111110 به لایه بالاتر تحویل داده می شود. به یک مثال در شکل ۳-۶ توجه کنید.

(الف) 011011111111111111110010

(ب) 011011111011111011111010010

بیت های لاگذاری

(ج) 011011111111111111110010

شکل ۳-۶. لاگذاری بیت. (الف) داده اولیه. (ب) داده ها بصورتی که روی خط فیزیکی

ارسال می شود. (ج) داده ها بصورتی که در گیرنده دریافت می شود.

در روش لاگذاری بیت نیز محدوده فریم با استفاده از پرچمهای شروع و پایان مشخص می شود، و گیرنده می تواند از آنها برای سنکرون شدن با فرستنده استفاده کند.

آخرین روش فریم بندی فقط در شبکه هایی قابل بکارگیری است که در کدگذاری لایه فیزیکی آنها نوعی افزونگی (redundancy) وجود داشته باشد. برای مثال در برخی از شبکه های LAN هر بیت داده با دو بیت فیزیکی نمایش داده می شود: بیت 1 با زوج بالا-پائین، و بیت 0 با زوج پائین-بالا. بدین ترتیب هر بیت داده دارای نوعی تغییر و تناژ است، که تشخیص آنرا برای گیرنده ساده تر می کند. در چنین شبکه هایی زوج بالا-بالا و پائین-پائین برای داده ها استفاده نمی شود، و می توان از آنها برای مشخص کردن محدوده فریمها سود برد.

لازم به ذکر است که در بسیاری از پروتکل های لینک داده برای اطمینان بیشتر از ترکیب روش شمارش کاراکترها با یکی دیگر از تکنیکهای گفته شده استفاده می شود. در این روش، انتهای فریم با استفاده از فیلد تعداد کاراکترها مشخص می شود، ولی فقط زمانی مورد قبول قرار می گیرد که جمع تطبیقی فریم نیز معتبر بوده و در این نقطه طرح بیت پایان فریم وجود داشته باشد. اگر چنین نباشد، گیرنده طرح بیت پایان فریم را در نقاط دیگر جستجو خواهد کرد.

۳-۱-۳ کنترل خطا

بعد از حل مسئله ابتدا و انتهای فریمها، نوبت به مسئله بعدی می رسد: چگونه می توان تمام فریمها را سالم و با ترتیب صحیح به مقصد رساند؟ فرض کنید فرستنده فقط فریمها را می فرستد و کاری ندارد که آنها به مقصد می رسند یا خیر. این وضعیت برای سرویسهای غیر متصل بدون تصدیق دریافت خوب است، ولی برای سرویسهای قابل اعتماد (مانند سرویس اتصال-گرا با تصدیق دریافت) مسلماً خوب نیست.

یک سرویس قابل اعتماد باید پنحوی از رسیدن بسته ها به مقصد و آنچه در آنجا اتفاق می افتد، مطلع شود. معمولاً در این موارد پروتکل درخواست می کند که یک فریم کنترلی خاص (که محتوی تصدیق یا عدم تصدیق

دریافت صحیح فریمهاست) به فرستنده باز پس فرستاده شود. اگر فرستنده تصدیق مثبت دریافت کند، مطمئن می شود که فریم به سلامت به مقصد رسیده است. اما تصدیق منفی نشان می دهد که اوضاع رو به راه نیست، و فریم باید مجدداً فرستاده شود.

مشکل دیگر اینجاست که گاهی (در اثر اشکالات سخت افزاری) یک فریم یکبارگی گم و ناپدید می شود. در این حالت گیرنده هیچ عکس العملی نشان نمی دهد، چون اساساً چیزی نگرفته که عکس العمل نشان دهد. بروشنی پیدا است که در این حالت پروتکل سمت فرستنده نا اید منتظر دریافت تصدیق از گیرنده می شود، تصدیقی که هرگز نخواهد رسید.

این مشکل را می توان با تعبیه یک تایمر در لایه پیوند داده حل کرد. وقتی فرستنده فریمی را می فرستد، تایمر را هم راه اندازی می کند. زمانی که این تایمر اندازه می گیرد آنقدر طولانی هست که بتوان با اطمینان گفت «فریم باید به مقصد رسیده، و تصدیق دریافت آن برگشته باشد». اگر همه چیز خوب پیش رفته باشد، معمولاً قبل از اینکه زمان تایمر به انتها برسد، تصدیق دریافت فریم به فرستنده برمی گردد (و تایمر ریست می شود).

اگر فریم با پاسخ آن در راه گم شوند، تایمر در انتهای زمان مقرر اخطار می دهد؛ و ساده ترین راه حل همانا ارسال مجدد فریم است. اما ارسال چندبارۀ فریمها این خطر را در بر دارد که چند تا از این فریمهای یکسان به مقصد برسند و به لایه شبکه تحویل داده شوند، برای اجتناب از این وضعیت، فرستنده به هر فریم یک شماره ترتیبی می دهد تا گیرنده بتواند فریمهای مشابه و تکراری را تشخیص دهد.

با تمام این تمهیدات (تایمر و شماره ترتیبی فریمها) می توان مطمئن بود که از هر فریم یک (و فقط یک) نسخه به لایه شبکه می رسد - و این یکی از مهمترین وظایف لایه پیوند داده است. در ادامه این فصل خواهید دید که لایه پیوند داده چگونه این وظیفه را انجام می دهد.

۴-۳ کنترل جریان

یکی دیگر از مسائل مهم در طراحی لایه پیوند داده (و لایه های بالاتر) اینست که با فرستنده هایی که سریعتر از توان دریافت گیرنده مبادرت به ارسال اطلاعات می کنند، چه باید کرد؟ اگر کامپیوتر طرف فرستنده قویتر از گیرنده باشد (و یا بار کاری کمتری داشته باشد)، این وضعیت براحتی می تواند پیش بیاید. در این حالت گیرنده در سیلاب فریمهای ارسالی از فرستنده غرق می شود. حتی اگر کانال ارتباطی کاملاً عاری از خطا باشد، لحظه ای می رسد که گیرنده دیگر قادر به پردازش فریمهای ارسال شده نیست، و برخی از آنها را از دست می دهد. روشن است که باید کاری برای جلوگیری از این وضعیت کرد.

دو رهیافت برای مقابله با این وضعیت بکار گرفته می شود. در رهیافت اول، که کنترل جریان بر اساس بازخور (feedback-based flow control) نام دارد، این گیرنده است که آمادگی خود را برای دریافت اطلاعات بیشتر به فرستنده اعلام می کند (یا حداقل اعلام می کند در چه وضعیتی است). در رهیافت دوم، کنترل جریان بر اساس نرخ (rate-based flow control)، پروتکل مکانیزمی دارد که بدون استفاده از بازخور گیرنده نرخ ارسال اطلاعات را محدود می کند. در این فصل با روشهای کنترل جریان بر اساس بازخور آشنا می شوید، ولی از آنجائیکه رهیافت دوم هرگز در لایه پیوند داده کاربرد ندارد، توضیح درباره آنرا به فصل ۵ موقوف می کنیم.

انواع مختلفی از کنترل جریان بر اساس بازخور وجود دارد، ولی همه آنها اصول مشترکی دارند: پروتکل قواعد تعریف شده ای دارد که زمان ارسال فریم بعدی را مشخص می کنند. طبق این قواعد فرستنده نمی تواند فریم بعدی را بفرستد، مگر اینکه (بطور صریح یا ضمنی) اجازه گیرنده را دریافت کرده باشد. مثلاً، وقتی اتصال برقرار می شود، گیرنده می تواند به فرستنده بگوید: «اکنون می توانی ۱۱ فریم بفرستی، ولی بعد از آن تا اجازه نداده ام چیزی نفرست».

۲-۳ کشف و تصحیح خطا

همانطور که در فصل ۲ دیدید، سیستم تلفن سه بخش عمده دارد: سوئیچها، ترانکها، و حلقه های محلی. در اکثر کشورهای توسعه یافته دو بخش اول تماماً دیجیتال هستند. اما قسمت اعظم حلقه های محلی کماکان آنالوگ است، که بایستی با صرف هزینه های هنگفت در آینده به دیجیتال تبدیل شود. با اینکه در بخش دیجیتال خطا بندرت روی می دهد، نرخ آن در حلقه های محلی آنالوگ همچنان بالاست. علاوه بر آن، مخابرات بیسیم نیز سرعت گسترش می یابد، که نرخ خطا در این قبیل سیستمها چندین برابر کانالهای فیبر نوری است. نتیجه اخلاقی: فعلاً تا مدتها باید با خطاهای انتقال در سیستمهای مخابراتی بسازیم. در این قسمت خواهید دید چگونه.

خصلت خطا به منبع آن بستگی دارد؛ برای مثال، در سیستمهای راداری خطا بصورت فورانی (burst) رخ می دهد، نه تکی. این نوع خطا مزایا و معایبی دارد، توجه داشته باشید که کامپیوترها اطلاعات خود را بصورت بسته ای ارسال می کنند. اگر هر بسته داده ۱۰۰۰ بیت و نرخ خطا نیز ۱ در ۱۰۰۰ باشد، می توان انتظار داشت که (در حالت غیرفوزانی) تقریباً تمام بسته ها با خطا به مقصد برسند. اما اگر خطا بصورت فورانهای ۱۰۰ بیتی رخ دهد، بطور متوسط فقط یک یا دو بسته را خراب خواهد کرد. عیب بزرگ خطاهای فورانی آنست که کشف و تصحیح خطا در آنها بسیار دشوارتر است.

۱-۲-۳ گداهای تصحیح خطا

طراحان شبکه دو استراتژی کلی برای مقابله با خطاها توسعه داده اند. یک راه اضافه کردن اطلاعات پراکنده به هر بلوک از داده هاست، بطوریکه گیرنده بتواند داده واقعی را از آن استنتاج کند. در روش دیگر فقط آنقدر اطلاعات اضافی به داده اصلی اضافه می شود که گیرنده از وقوع یا عدم وقوع خطا آگاهی یابد، و در صورت لزوم تکرار ارسال را خواستار شود. استراتژی اول گداهای تصحیح خطا (error-correcting codes) و استراتژی دوم گداهای کشف خطا (error-detecting codes) نام دارند. به کاربرد گداهای تصحیح خطا اغلب تصحیح پیشگیرانه خطا نیز گفته می شود.

هر یک از این تکنیکها جایگاه خاص خود را دارند. در کانالهای قابل اطمینان، مانند فیبر نوری، مقرون بصرفه تر است که از گداهای کشف خطا استفاده کرده و بسته های معدودی را که خراب می شوند، دوباره ارسال کنیم. اما در کانالهایی مانند لینکهای بیسیم که پر از خطا هستند، بهتر است از تکنیکهای تصحیح خطا استفاده کرده و اجازه دهیم گیرنده خود داده واقعی را بدست آورد (چون با احتمال زیاد ارسال مجدد بسته ها هم عاری از خطا نخواهد بود). برای مقابله با خطاها، ابتدا باید بدانیم خطا واقعاً چیست. معمولاً، یک فریم m بیت داده اصلی (یعنی، پیام) و r بیت داده پراکنده (یا اطلاعات چک کننده) دارد، که در مجموع n بیت می شود ($n = m + r$). به این واحد n بیتی (داده های اصلی و پراکنده) اغلب کلمه کُد n بیتی گفته می شود.

دو کلمه کُد 10001001 و 10110001 را در نظر بگیرید: براحتی می توان مشخص کرد که این دو کلمه چند اختلاف دارند. در این مورد ۳ بیت اختلاف وجود دارد. برای تعیین تعداد اختلاف ها می توان دو کلمه کُد را با هم XOR (OR انحصاری) کرد، و تعداد 1 ها را شمرد:

```
10001001
10110001
00111000
```

به تعداد اختلافهای دو کلمه کُد فاصله همینگ (Hamming distance) گفته می شود (Hamming, 1950). اهمیت این فاصله در آنجاست که می توان ثابت کرد برای تبدیل شدن اتفاقی دو کلمه با فاصله d ، بایستی d خطای تکبیتی روی دهد.

در اکثر سیستمهای انتقال، تمامی 2^m حالت ممکنه داده اصلی مجاز است، ولی بدلیل روش محاسبه بیت های افزونگی، تمام 2^n حالت کلمه کُد مجاز نیست. با توجه به الگوریتم محاسبه بیت های افزونگی، می توان لیستی از تمام حالت های مجاز کلمه کُد بدست آورد، و از این لیست دو کلمه ای که کمترین فاصله همینگ را دارند، پیدا کرد. این فاصله فاصله همینگ الگوریتم یا کُد مورد نظر است.

خصوصیات تصحیح خطا یا کشف خطای یک کُد به فاصله همینگ آن بستگی دارد. برای کشف d خطا، به کُدی با فاصله همینگ $d + 1$ نیاز داریم، چون با چنین کُدی هیچ d خطای تک بیتی وجود ندارد که بتواند یک کلمه کُد مجاز را به کلمه کُد مجاز دیگر تبدیل کند. اگر گیرنده کلمه کُد غیر مجازی دریافت کرد، می تواند با اطمینان بگوید که خطایی رخ داده است. بهمین ترتیب، برای تصحیح d خطا، به کُدی با فاصله $2d + 1$ نیاز داریم، چون در این حالت کلمات کُد چنان از هم فاصله دارند که حتی با بروز d خطا، کلمه کُد خراب شده هنوز نزدیکترین فاصله را با کلمه کُد اصلی دارد، و تشخیص آن براحتی ممکن است.

بعنوان نمونه ای از کُد های کشف خطا، کُدی با یک بیت توازن (parity bit) را در نظر بگیرید. این بیت توازن بگونه ای انتخاب می شود که تعداد بیت های 1 کلمه کُد همواره زوج (یا فرد) شود. برای مثال، اگر بخواهیم کلمه 1011010 را با توازن زوج (even parity) ارسال کنیم، یک بیت 0 به انتهای آن اضافه می کنیم (10110100)؛ اما اگر بخواهیم همین کلمه را با توازن فرد (odd parity) ارسال کنیم، باید یک بیت 1 به انتهای آن اضافه کنیم (10110101). کُدی با یک بیت توازن دارای فاصله همینگ 2 است، چون هر خطای تک بیتی کلمه کُدی با توازن اشتباه تولید می کند. این کُد می توان یک خطا در هر کلمه را آشکار کند.

بعنوان یک نمونه ساده از کُد های تصحیح خطا، کُدی را در نظر بگیرید که فقط چهار کلمه کُد مجاز دارد:

0000000000, 0000011111, 1111100000, 1111111111

فاصله همینگ این کُد 5 است، بنابراین می تواند دو خطا را تصحیح کند. اگر گیرنده کلمه کُدی بصورت 0000000111 دریافت کند، می داند که کلمه اصلی باید 0000011111 بوده باشد. اما اگر سه خطا کلمه 0000000000 را به 0000000111 تبدیل کرده باشد، دیگر نمی توان آنرا بدرستی تصحیح کرد.

فرض کنید می خواهیم کُدی با m بیت داده اصلی و r بیت افزونگی طراحی کنیم که بتواند تمام خطاهای تک بیتی را تصحیح کند. هر یک از 2^m پیام مجاز دارای n کلمه کُد غیر مجاز است که با آن 1 فاصله دارد (این را می توان پسادگی از معکوس کردن هر یک از بیت های کلمه کُد n بیتی فهمید). بنابراین هر یک از 2^m پیام مجاز به طرحی اختصاصی با $n + 1$ بیت نیاز دارد. از آنجائیکه تعداد ترکیبات ممکنه کلمه کُد 2^n است، بایستی داشته باشیم: $2^n \leq (n + 1)2^m$. با قرار دادن $n = m + r$ در این رابطه، داریم: $(m + r + 1) \leq 2^r$. با داشتن m ، از این رابطه حداقل بیت های افزونگی لازم (r) برای تصحیح خطاهای تک بیتی بدست می آید.

همینگ در یکی از مقالات خود (1950) روشی برای بدست آوردن این حداقل معرفی کرد. وی بیت های کلمه کُد را از چپ براسست شماره گذاری کرد. بیت هایی که توانایی از 2 هستند (1، 2، 4، 8، 16، و غیره)، بیت های چک کننده اند؛ سایر بیت ها (3، 5، 6، 7، 9، و غیره) بیت های پیام (m) هستند. هر بیت چک کننده توازن مجموعه ای از بیت (از جمله خودش) را زوج (یا فرد) می کند. هر بیت می تواند در بیش از یک مجموعه توازن محاسبه شود. برای دیدن اینکه کدام بیت های چک کننده در محاسبه توازن بیت داده ای در موقعیت k دخالت دارند، k را بصورت مجموع توان های 2 می نویسیم. برای مثال، $11 = 1 + 2 + 8$ ، و $29 = 1 + 4 + 8 + 16$. هر بیت فقط با بیت های چک کننده ای که در موقعیت های بدست آمده از مجموع توان های 2 قرار دارند، چک می شود (مثلاً، بیت موقعیت 11 فقط با بیت های چک کننده 1، 2 و 8 چک می شود).

وقتی یک کلمه کُد به گیرنده می رسد، گیرنده یک شمارنده را 0 می کند. سپس تمام بیت های چک کننده (k) را

کاراکتر	ASCII	بیت های چک کننده
H	1001000	00110010000
a	1100001	10111001001
m	1101101	11101010101
m	1101101	11101010101
i	1101001	01101011001
n	1101110	01101010110
g	1100111	01111001111
	0100000	10011000000
c	1100011	11111000011
o	1101111	10101011111
d	1100100	11111001100
e	1100101	00111000101

ترتیب انتقال بیت ها

شکل ۳-۷. استفاده از کد همینگ برای تصحیح خطاهای فورانی.

از نظر توازن چک می کند ($k = 1, 2, 4, 8, \dots$). اگر توازن k درست نباشد، گیرنده k را به شمارنده اضافه می کند. اگر پس از پایان این عملیات شمارنده همچنان 0 باشد، کلمه کد صحیح تلقی و قبول می شود. اگر شمارنده 0 نباشد، حتماً شماره بیت خطا را نشان می دهد. برای مثال، اگر توازن بیت های چک کننده 1، 2 و 8 اشتباه باشد، بیت 11 غلط است، چون این تنها بیتی است که با بیت های چک کننده 1، 2 و 8 چک می شود. در شکل ۳-۷ چند کاراکتر آسکی ۷-بیتی را که با کد همینگ ۱۱-بیتی کد شده اند، می بینید. فراموش نکنید که داده های اصلی در موقعیتهای 3، 5، 6، 7، 9، 10 و 11 قرار دارند.

کدهای همینگ فقط می توانند خطاهای تک بیتی را تصحیح کنند. با این حال روشی وجود دارد که اجازه می دهد تا این کد خطاهای فورانی را نیز تصحیح کند. در این روش k کلمه کد متوالی بصورت ماتریس (یک کلمه کد در هر سطر) چیده می شوند. معمولاً، این کلمات تک به تک (از چپ بر راست) ارسال می شوند. برای تصحیح خطاهای فورانی، پایستی داده ها را بصورت ستونی (باز هم از چپ بر راست) ارسال کرد. وقتی k بیت اول (ستون اول) ارسال شد، نوبت به ستون دوم (و سپس ستونهای بعدی) می رسد (شکل ۳-۷ را ببینید). وقتی این فریم به گیرنده رسید، ماتریس از نو (ستون به ستون) ساخته می شود. اگر یک خطای فورانی به طول k رخ داده باشد، حداکثر یک بیت در هر کلمه کد تغییر خواهد کرد، و از آنجائیکه کد همینگ می تواند یک خطا را تصحیح کند، تمام بلوک قابل تصحیح خواهد بود. در این روش برای مصون کردن k بیت داده در مقابل خطاهای فورانی با طول k (یا کمتر)، از k^2 بیت چک کننده استفاده شده است.

۲-۲-۳ کدهای کشف خطا

کدهای کشف خطا در لینکهای بیسیم، که در مقایسه با سیم مسی و فیبر نوری بطور وحشتناکی نویزی هستند، کاربرد گسترده ای دارد. بدون این کدها شاید اساساً نتوان چیزی روی این لینکها رد و بدل کرد. اما در سیمهای مسی و فیبرهای نوری نرخ خطا بسیار کمتر است، و تشخیص خطا و ارسال مجدد بسته هایی که (ندرتاً) خراب می شوند، کاملاً کفایت می کند.

به عنوان مثال، کانالی را در نظر بگیرید که نرخ خطا در آن 1 در 10^6 و خطاها غیر فورانی هستند؛ اندازه هر بلوک را هم 1000 بیت فرض می کنیم. برای داشتن ویژگی تصحیح خطا، هر بلوک ۱۰۰۰ بیتی به ۱۰ بیت چک کننده نیاز دارد، بعبارت دیگر برای ارسال 1 Mb داده باید 10 kb اطلاعات افزونگی (بیت های چک کننده) را نیز به همراه آن

بفرستیم. اما برای کشف خطا فقط یک بیت توازن در هر بلوک کافیتست. در این روش بار اضافی کشف خطا + ارسال مجدد یک بلوک خراب برای 1 Mb داده فقط 2001 بیت است، که در مقایسه با 10,000 بیت کد همینگ بسیار کمتر است.

اگر در هر بلوک از یک بیت توازن برای کشف خطا استفاده کنیم و یک خطای فورانی رخ دهد، احتمال اینکه بتوانیم خطا را کشف کنیم فقط ۵۰٪ است، که بهیچوجه قابل قبول نیست. اما با تشکیل ماتریسی با n ستون و k سطر (که در بالا توضیح دادیم) اوضاع بنحو قابل توجهی بهتر خواهد شد. در این روش برای هر ستون یک بیت توازن محاسبه، و در آخرین سطر ماتریس نوشته می شود. هنگام ارسال نیز این ماتریس بصورت ستونی فرستاده می شود. گیرنده بعد از دریافت کل ماتریس، تمام بیت های توازن را چک می کند؛ و اگر هر یک از این بیت ها غلط باشد، ارسال مجدد ماتریس را درخواست می کند. این کار تا زمانی که ماتریس بطور کامل و بدون خطای توازن به دست گیرنده برسد، تکرار خواهد شد.

روش فوق می تواند خطاهای فورانی با طول حداکثر n بیت را آشکار کند، چون در این حالت فقط یک بیت در هر ستون تغییر خواهد کرد. اما اگر یک خطای فورانی با طول $1 + n$ رخ دهد بگونه ای که فقط بیت اول و آخر را تغییر دهد (و سایر بیت ها تغییر نکنند)، نمی توان آنرا کشف کرد، زیرا بیت اول و آخر در یک سطر قرار می گیرند و توازن این سطر بدون تغییر خواهد ماند. (یک خطای فورانی الزاماً بمعنای معکوس شدن تمام بیت ها نیست: فقط می توان از معکوس شدن بیت اول و آخر مطمئن بود). اگر طول خطای فورانی خیلی زیاد باشد یا تعدادی خطای فورانی کوتاه و پشت سر هم رخ دهد، احتمال اینکه یکی از ستونها تصادفاً صاحب توازن درست شود، ۵۰٪ است، بنابراین احتمال اینکه چنین بلوکی (به اشتباه) صحیح تلقی شود، 2^{-n} خواهد بود.

با اینکه روش فوق در مواردی کفایت می کند، اما در عمل از روش دیگری استفاده می شود: کد چندجمله ای (polynomial code)، که به CRC (چک افزونگی چرخه ای - Cyclic Redundancy Check) نیز معروفست. در کدهای چندجمله ای مبنای این است که هر رشته یک چندجمله ایست با ضرایب 0 و 1. با این فرض، یک فریم k -بیتی معادلست با عبارتی k جمله ای، با ضرایب x^{k-1} تا x^0 . این چندجمله ای از درجه $k-1$ است. با ارزشترین بیت (منتهی الیه سمت چپ) ضریب x^{k-1} است، بیت بعدی ضریب x^{k-2} ، و الی آخر. برای مثال، رشته 110001 دارای 6 بیت است بنابراین نشاندهنده یک شش جمله ایست با ضرایب 1، 0، 0، 0، 1، و 1، که می توان آنرا چنین نوشت: $x^5 + x^4 + x^0$.

محاسبات چندجمله ایها در مدول 2 (و طبق قوانین جبر میدان) انجام می شود. در جمع و تفریق 2 بر 1 نادیده گرفته می شود، بعبارت دیگر شبه XOR است. برای مثال،

$$\begin{array}{r} 10011011 \\ + 11001010 \\ \hline 01010001 \end{array} \quad \begin{array}{r} 00110011 \\ + 11001101 \\ \hline 11111110 \end{array} \quad \begin{array}{r} 11110000 \\ - 10100110 \\ \hline 01010110 \end{array} \quad \begin{array}{r} 01010101 \\ - 10101111 \\ \hline 11111010 \end{array}$$

تقسیم درست مانند تقسیم باینری است، با این تفاوت که تفریق ها در مدول 2 (مانند بالا) انجام می شود. در هنگام استفاده از روش کد چندجمله ای، فرستنده و گیرنده بایستی از قبل بر سر یک چندجمله ای مولد (generator polynomial)، که آنرا $G(x)$ می نامیم، توافق کنند. با ارزشترین (چپ ترین) و کم ارزشترین (راست ترین) بیت های چندجمله ای مولد باید 1 باشد. برای محاسبه مجموع چک (checksum) یک فریم m -بیتی (که چندجمله ای متناظر با آن $M(x)$ است)، این فریم باید طولانیتر از چندجمله ای مولد باشد. ایده آنست که یک مجموع چک به انتهای فریم اصلی چسبانده شود، بگونه ای که فریم حاصله بر $G(x)$ قابل تقسیم باشد. اگر

تقسیم این فریم بر $G(x)$ در سمت گیرنده باقیمانده آورد، معلوم می شود که خطایی رخ داده است.

الگوریتم محاسبه مجموع چک چنین است:

۱. فرض می کنیم چندجمله ای $G(x)$ از درجه r است. r بیت ۰ به سمت راست فریم اضافه می کنیم تا تعداد بیت های آن به $m + r$ برسد. این چندجمله ای معادل $x^r M(x)$ خواهد شد.

۲. رشته $x^r M(x)$ را (در مدول ۲) بر $G(x)$ تقسیم می کنیم.

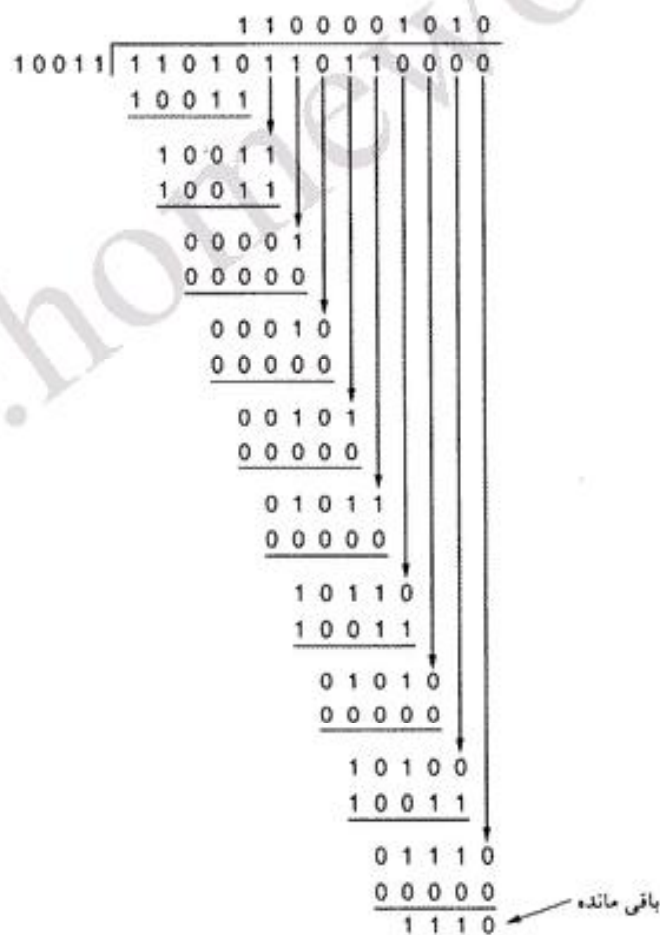
۳. باقیمانده را (که همیشه r بیت یا کمتر دارد) از $x^r M(x)$ کم می کنیم (این تفریق هم در مدول ۲ انجام می شود). حاصل تفریق همان فریم مورد نظر (فریم اولیه + مجموع چک) است، که آنرا $T(x)$ می نامیم.

در شکل ۸-۳ طرز محاسبه مجموع چک برای فریم ۱۱۰۱۰۱۱۰۱۱ را با مولد $G(x) = x^4 + x + 1$ ملاحظه می کنید.

فریم: ۱۱۰۱۰۱۱۰۱۱

مولد: ۱۰۰۱۱

پیام بعد از اضافه شدن ۴ بیت: ۱۱۰۱۰۱۱۰۱۱۰۰۰۰



فریم اضافه شده: ۱۱۰۱۰۱۱۰۱۱۱۱۱۰

شکل ۸-۳ محاسبه مجموع چک کد چندجمله ای.

همانطور که براحتی معلوم می شود، $T(x)$ بر $G(x)$ (در مدول 2) بخش پذیر است (چون وقتی باقیمانده تقسیم را از مقسوم کم کنیم، عدد حاصله بطور حتم بر مقسوم علیه بخش پذیر خواهد بود). بطور مثال، اگر 210,278 را (در مبنای 10) بر 10,941 تقسیم کنیم، باقیمانده 2399 می شود که اگر آنرا از 210,278 کم کنیم، آنچه باقی می ماند (207,879) بر 10,941 بخش پذیر خواهد بود.

حال اجازه دهید قدرت این روش را بررسی کنیم. این روش چه نوع خطاهایی را می تواند کشف کند؟ فرض کنید خطایی رخ داده، و بجای $T(x)$ رشته $T(x) + E(x)$ به گیرنده رسیده است، بطوریکه هر بیت 1 در $E(x)$ متناظر با یک بیت تغییر یافته است (بعبارت دیگر، اگر در این عبارت، $E(x)$ ، k بیت 1 وجود داشته باشد، k خطای تکبیتی رخ داده است). خطای فورانی نیز عبارتست از خطایی که با یک بیت 1 شروع و ختم شود، و بین آنها هر ترکیبی از 0 و 1 می تواند وجود داشته باشد.

وقتی این فریم به مقصد می رسد، گیرنده آنرا بر $G(x)$ تقسیم می کند (بعبارت دیگر $[T(x)+E(x)]/G(x)$ را محاسبه می کند). از آنجائیکه $T(x)/G(x) = 0$ ، این تقسیم معادل $E(x)/G(x)$ است. همانطور که می بینید، اگر خطای رخ داده دقیقاً طرحی شبیه $G(x)$ نداشته باشد، بطور مسلم آشکار خواهد شد. فرض کنید یک خطای تکبیتی رخ داده است، یعنی، $E(x) = x^i$ (که i بیت خطاست). اگر $G(x)$ بیش از دو جمله داشته باشد، $E(x)$ هرگز بر آن بخش پذیر نخواهد بود - پس، این روش می تواند تمام خطاهای تکبیتی را آشکار کند.

اگر دو خطای تکبیتی جدا از هم رخ دهد، بطوریکه $E(x) = x^i + x^j$ (که در آن $i > j$)، می توان $E(x)$ را به صورت $(x^{j+1} + 1)x^j$ تجزیه کرد. اگر $G(x)$ بر x بخش پذیر نباشد، شرط کافی برای اینکه تمام خطاهای دوبیتی قابل کشف باشد آن است که $E(x)$ عبارت $x^k + 1$ را (برای تمام k های کوچکتر از $i - j$) بخش نکند. چندجمله ای های ساده و از درجه پائینی می شناسیم که می توان با آنها فریمهای نسبتاً طولیل را محافظت کرد. مثلاً، چندجمله ای $x^{15} + x^{14} + 1$ هیچ عبارت $x^k + 1$ را برای تمام k های کوچکتر از 32,768 بخش نمی کند.

اگر تعداد خطاهای رخ داده عددی فرد باشد، تعداد جملات $E(x)$ نیز فرد خواهد بود (برای مثال، تعداد جملات $x^2 + 1$ فرد است، ولی $x^2 + 1$ چنین نیست). جالبست بدانید که هیچ چندجمله ای با تعداد جملات فرد وجود ندارد که (در مدول 2) بر $x + 1$ بخش پذیر باشد. بدین ترتیب اگر $G(x)$ را طوری انتخاب کنیم که بر $x + 1$ بخش پذیر باشد، می توانیم هر خطایی که تعداد بیتهای تغییر کرده فرد باشد را کشف کنیم.

برای اثبات اینکه هیچ چندجمله ای فرد وجود ندارد که بر $x + 1$ بخش پذیر باشد، فرض کنید $E(x)$ چندجمله ای فردیست که چنین خاصیتی دارد (بر $x + 1$ بخش پذیر است). اگر از $x + 1$ فاکتور بگیریم، $E(x)$ بصورت $Q(x)(x + 1)$ در می آید. حال $E(1) = (1 + 1)Q(1)$ را محاسبه می کنیم. از آنجائیکه (در مدول 2) $1 + 1 = 0$ ، $E(1) = 0$ باید باشد. اما اگر تعداد جملات $E(x)$ فرد باشد، قرار دادن 1 بجای x در آن همیشه نتیجه 1 می دهد. بنابراین فرض ما نمی تواند درست باشد، و هیچ چندجمله ای فرد بر $x + 1$ بخش پذیر نیست.

بالاخره، و از همه مهمتر، یک کد چندجمله ای با r بیت چک کننده تمام خطاهای فورانی با طول کمتر یا مساوی r را آشکار می کند. یک خطای فورانی با طول k را می توان با $(x^{k-1} + \dots + 1)x^i$ نشان داد، که در آن i نقطه شروع خطای فورانی از سمت راست فریم است. اگر مولد $G(x)$ دارای جمله x^0 باشد، بر x^k بخش پذیر نخواهد بود؛ بنابراین اگر درجه عبارت داخل پراتنز از درجه $G(x)$ کمتر باشد، باقیمانده تقسیم هرگز نمی تواند 0 شود.

اگر طول خطای فورانی $r + 1$ باشد، باقیمانده تقسیم بر $G(x)$ صفر می شود فقط و فقط اگر طرح بیت خطا با $G(x)$ یکسان باشد. طبق تعریف بیت های اول و آخر خطای فورانی باید 1 باشند، بنابراین یکسان بودن آنها به $r - 1$ بیت میانی بستگی دارد. اگر تمام ترکیبات این $r - 1$ بیت را یکسان فرض کنیم، احتمال بروز این وضعیت $\frac{1}{2^{r-1}}$ خواهد بود.

همچنین می توان نشان داد که اگر طول خطای فورانی از $1 + 2^r$ بزرگتر باشد یا چند خطای فورانی کوتاهتر رخ دهد، احتمال کشف نشدن خطا (با فرض یکسان بودن تمام ترکیبات ممکنه) $\frac{1}{2^r}$ است.

برخی از چندجمله ایها بصورت استاندارد بین المللی در آمده اند، که از میان آنها می توان به چندجمله ای زیر (که در IEEE 802 از آن استفاده می شود) اشاره کرد:

$$1 + x^1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^{10} + x^{11} + x^{12} + x^{16} + x^{22} + x^{23} + x^{26} + x^{32}$$

از ویژگیهای جالب این چندجمله ای آن است که هر نوع خطای فورانی با طول 32 یا کمتر، و خطاهای فورانی که تعداد پشتهای تغییر کرده فرد باشد، را آشکار می کند.

با اینکه بنظر می رسد محاسبه مجموع چک و تست آن کار پیچیده ای باشد، پترسون و براون (1961) نشان دادند که می توان این کار را با یک مدار شیفت رجیستر (shift register) ساده بصورت سخت افزاری انجام داد. در واقع، این مدار در تمام کارتهای شبکه تعبیه شده است، و بسیاری از خطوط نقطه به نقطه هم از آن استفاده می کنند.

برای مدتهای مدید تصور بر آن بود که فریمهایی که مجموع چک آنها محاسبه می شود دارای طرح بیت تصادفی هستند، و تمام الگوریتمهای محاسبه مجموع چک نیز فرض را بر این می گذاشتند. اما بررسی دقیق داده های واقعی نشان داده که این فرض بکلی اشتباه است. در نتیجه، خطاهایی که (تحت شرایط خاص) کشف نشده می مانند شایعتر از آن چیز است که قبلاً تصور می شد (Partridge et al., 1995).

۳-۳ چند پروتکل ساده لینک داده

برای آشنایی با پروتکل های لایه پیوند داده، در این قسمت سه پروتکل را (که بتدریج پیچیده تر می شوند) بررسی خواهیم کرد. برای خوانندگان علاقمند، شبیه ساز این پروتکلها (و پروتکلهایی که در آینده خواهید دید) را در سایت وب کتاب قرار داده ایم (<http://www.prenhall.com/tanenbaum>). اما قبل از اینکه سراغ این پروتکلها برویم، اجازه دهید چند تا از فرضهایی را که درباره مدل ارتباطی زیربنایی داشته ایم، توضیح دهیم. اول اینکه فرض کرده ایم در لایه های فیزیکی، لایه پیوند داده و شبکه پروسسهای هستند که مستقل از یکدیگرند، و ارتباط آنها از طریق رد و بدل کردن پیام صورت می گیرد. در بسیاری از موارد، پروسسهای لایه فیزیکی و لینک داده در پردازنده کارت شبکه اجرا می شوند، و پروسسهای لایه شبکه در CPU اصلی کامپیوتر. البته پیاده سازیهای دیگری نیز محتمل است (مثلاً، تمام پروسسهای لایه فیزیکی، لینک داده و شبکه در پردازنده کارت شبکه اجرا شوند، یا همگی آنها را CPU اصلی اجرا کند). در هر حال، مستقل دانستن این پروسسها بحث درباره آنها را بسیار ساده تر می کند، و تأکید است بر مستقل بودن لایه ها.

فرض کلیدی دیگر اینست که، ماشین A با استفاده از یک سرویس اتصال-گرای قابل اعتماد استریم طولی از داده ها را به ماشین B می فرستد. بعدها این حالت که B هم همزمان به A داده بفرستد، را بررسی خواهیم کرد. علاوه بر آن، فرض کرده ایم ماشین A منبع بی پایانی از داده ها دارد که ارسال کند، و هرگز منتظر آمدن داده ها نخواهد شد. عبارت دیگر، هر گاه لایه پیوند داده A درخواست داده کند، لایه شبکه بلافاصله اجابت می کند. (بعدها این فرض را هم کنار خواهیم گذاشت.)

فرض دیگر ما اینست که این کامپیوترها هرگز از کار نمی افتند؛ یعنی، پروتکلهای ما فقط با خطاهای مخابراتی سروکار دارند، نه هنگ کردن کامپیوتر و مسائلی از این قبیل.

دیگر اینکه، تا آنجا که به لایه پیوند داده مربوط است، بسته ای که به لایه شبکه داده می شود، داده خالص است، و باید تا آخرین بیت به آن تحویل شود. این که بخشی از این داده ها سرآیند بسته هستند و لایه شبکه آنها را دور می ریزد، به خودش مربوط است نه به لایه پیوند داده.

وقتی لایه پیوند داده بسته ای از لایه شبکه می گیرد تا ارسال کند، آنرا فریم بندی کرده و سرآیند (header) و پی آیند (trailer) های لازم را به آن می چسباند (شکل ۳-۱ را ببینید). بنابراین هر فریم از سه بخش تشکیل می شود: قسمتی از بسته ای که لایه شبکه فرستاده، یک سرآیند شامل اطلاعات کنترلی، و یک پی آیند شامل مجموع چک فریم. سپس این فریم به لایه پیوند داده ماشین مقصد فرستاده می شود. فرض ما بر این است که روال های کتابخانه ای *to_physical_layer* (برای ارسال) و *from_physical_layer* (برای دریافت) از قبل وجود دارند. مجموع چک نیز (مثلاً، با استفاده از کدهای چند جمله ای) توسط سخت افزار محاسبه (و به انتهای فریم اضافه) می شود، بنابراین لازم نیست لایه پیوند داده نگران آن باشد.

در ابتدا، لازم نیست گیرنده کاری انجام دهد؛ فقط منتظر می ماند تا اتفاقی بیفتد. در مثالهای این فصل، فرض کرده ایم که لایه پیوند داده این کار را با روالی بنام *wait_for_event(&event)* انجام می دهد. این روال فقط وقتی به پایان می رسد (و کنترل را به برنامه اصلی برمی گرداند) که اتفاقی افتاده باشد (یعنی، یک فریم دریافت شده باشد). اینکه چه اتفاقی افتاده است، را متغیر *event* مشخص می کند؛ و این که چه اتفاقی می تواند بیفتد، به تعریف پروتکل بستگی دارد. توجه داشته باشید که در دنیای واقعی لایه پیوند داده (مانند این مثالها) در یک حلقه بی انتها منتظر رسیدن فریمها نمی ماند، بلکه با استفاده از وقفه (*interrupt*) به آنها رسیدگی می کند. با این حال برای اجتناب از پیچیدگی مطلب، فرض کرده ایم که لایه پیوند داده هیچ کار دیگری جز رسیدگی به کانال ارتباطی ما ندارد.

وقتی یک فریم به گیرنده می رسد، سخت افزار مجموع چک آنرا محاسبه می کند. اگر این مجموع چک اشتباه باشد (یعنی خطایی رخ داده)، به لایه پیوند داده اطلاع داده می شود (*event = cksum_err*). اگر فریم بدرستی دریافت شده باشد، باز هم به اطلاع لایه پیوند داده می رسد (*event = frame_arrival*). در این حالت لایه پیوند داده با استفاده از تابع *from_physical_layer* فریم را گرفته، اطلاعات کنترلی موجود در سرآیند آنرا چک می کند، و اگر همه چیز مرتب باشد، سرآیند را جدا کرده و بخش اصلی داده را به لایه شبکه تحویل می دهد.

تحت هیچ شرایطی سرآیند فریم به لایه شبکه تحویل نمی شود، و برای این کار دلیل خوبی وجود دارد. پروتکل های لینک داده و شبکه باید کاملاً از یکدیگر مستقل باشند. مستقل بودن پروتکل های این دو لایه باعث می شود که بتوان هر کدام از این پروتکلها را تغییر داد، بدون اینکه نیاز باشد پروتکل های لایه دیگر تغییر کند (البته تحت هر شرایطی نحوه تعامل و ارتباط لایه ها نباید تغییر کند). جدا و مستقل بودن لایه ها تا حد زیادی طراحی آنها را ساده می کند، چون می توان بدون نگرانی از اتفاقاتی که در لایه های دیگر می افتد، روی طراحی عملکردهای همان لایه تمرکز کرد.

در شکل ۳-۹ مقداری تعریف (به زبان C) می بینید، که در طراحی پروتکل های این قسمت به آنها نیاز داریم. در اینجا پنج ساختار داده تعریف شده است: *frame*، *frame_kind*، *packet*، *seq_nr*، *boolean*. ساختار *boolean* از نوع شمارشی (*enum*) است، و می تواند دو مقدار *true* و *false* بگیرد. ساختار *seq_nr* از نوع عدد صحیح بدون علامت (*unsigned int*) تعریف شده، و برای شماره گذاری فریمها از آن استفاده خواهیم کرد. شماره گذاری فریمها از 0 تا *MAX_SEQ* (که بسته به نیاز هر پروتکل تعریف می شود) انجام می گیرد. *packet* (بسته) واحدی از اطلاعات است که بین لایه شبکه و لایه پیوند داده (روی یک ماشین، یا روی ماشین های جداگانه) رد و بدل می شود. در مدل ما هر بسته همیشه حاوی *MAX_PKT* بایت داده است، ولی به واقعیت نزدیکتر است که طول بسته را متغیر در نظر بگیریم.

هر *frame* از چهار فیلد تشکیل شده: *ack*، *seq*، *kind* و *info* - که سه تای اول اطلاعات کنترلی هستند، و آخری همان داده هائیکه باید منتقل شود. به مجموعه فیلدهای کنترلی سرآیند فریم (*frame header*) گفته می شود.


```

#define MAX_PKT 1024 /* determines packet size in bytes */

typedef enum {false, true} boolean; /* boolean type */
typedef unsigned int seq_nr; /* sequence or ack numbers */
typedef struct {unsigned char data[MAX_PKT];} packet; /* packet definition */
typedef enum {data, ack, nak} frame_kind; /* frame_kind definition */

typedef struct { /* frames are transported in this layer */
    frame_kind kind; /* what kind of a frame is it? */
    seq_nr seq; /* sequence number */
    seq_nr ack; /* _acknowledgement number */
    packet info; /* the network layer packet */
} frame;

/* Wait for an event to happen; return its type in event. */
void wait_for_event(event_type *event);

/* Fetch a packet from the network layer for transmission on the channel. */
void from_network_layer(packet *p);

/* Deliver information from an inbound frame to the network layer. */
void to_network_layer(packet *p);

/* Go get an inbound frame from the physical layer and copy it to r. */
void from_physical_layer(frame *r);

/* Pass the frame to the physical layer for transmission. */
void to_physical_layer(frame *s);

/* Start the clock running and enable the timeout event. */
void start_timer(seq_nr k);

/* Stop the clock and disable the timeout event. */
void stop_timer(seq_nr k);

/* Start an auxiliary timer and enable the ack_timeout event. */
void start_ack_timer(void);

/* Stop the auxiliary timer and disable the ack_timeout event. */
void stop_ack_timer(void);

/* Allow the network layer to cause a network_layer_ready event. */

```

```
void enable_network_layer(void);
```

```
/* Forbid the network layer from causing a network_layer_ready event. */
```

```
void disable_network_layer(void);
```

```
/* Macro inc is expanded in-line: Increment k circularly. */
```

```
#define inc(k) if (k < MAX_SEQ) k = k + 1; else k = 0
```

شکل ۳-۹. تعریف های مورد نیاز برای پروتکل های که در این فصل می نویسیم. این

تعریف ها در فایل بنام *protocol.h* قرار داده می شوند.

فیلد *kind* می گوید که آیا داده ای در فریم وجود دارد یا خیر، چون برخی از پروتکل های فریم های بدون داده را از فریم هایی که داده دارند، تمیز می دهند. فیلدهای *seq* و *ack* به ترتیب برای شماره ترتیبی فریم و تصدیق دریافت مورد استفاده قرار می گیرند؛ بعداً در این باره بیشتر توضیح خواهیم داد. داده اصلی (بسته) در فیلد *info* فریم قرار دارد؛ فریم های کترلی نیز وجود دارند که اساساً در آنها فیلد *info* وجود ندارد. در پروتکل های واقعی که طول فیلد *info* می تواند متغیر باشد، نیازی به تمایز بین فریم های داده و فریم های کترلی نیست (چون فریم کترلی فریمی است که طول فیلد *info* در آن ۰ است).

در اینجا لازم است تفاوت فریم (*frame*) و بسته (*packet*) را مجدداً یادآور شویم. لایه شبکه با گرفتن پیام از لایه انتقال و اضافه کردن سرآیند، آنرا بصورت بسته در می آورد. سپس این بسته به لایه پیوند داده تحویل می شود، که در آنجا در فیلد *info* یک فریم قرار داده شده و برای ارسال آماده می شود. وقتی این فریم به لایه پیوند داده ماشین مقصد رسید، بسته از فیلد *info* استخراج شده و به لایه شبکه تحویل می شود. این فرآیند بکلی شفاف است، و لایه های شبکه در دو ماشین متقابل تصور می کنند که مستقیماً در حال تبادل بسته اند.

در برنامه شکل ۳-۹ چند روال (تابع) نیز تعریف شده است. اینها روال های کتابخانه ای هستند، که فقط کاری که انجام می دهند برای ما مهم است (و اصلاً اهمیتی ندارد این کار را چگونه انجام می دهند). روال *wait_for_event* (همانطور که قبلاً گفتیم) در یک حلقه بی انتها به انتظار می ماند تا اتفاقی بیفتد. روال های *to_network_layer* و *from_network_layer* به ترتیب برای ارسال بسته به لایه شبکه و برای گرفتن بسته از این لایه بکار می روند (اینها واسطه لایه های ۲ و ۳ هستند). برای تبادل اطلاعات با لایه فیزیکی نیز از روال های *to_physical_layer* و *from_physical_layer* استفاده می شود (اینها واسطه لایه های ۱ و ۲ هستند).

در پروتکل های واقعی فرض بر اینست که کانال ارتباطی نامطمئن است، و احتمال این هست که فریم ها در راه از بین بروند. برای مقابله با چنین وضعیتی، لایه پیوند داده همزمان با ارسال هر فریم، باید یک تایمر را راه اندازی کند. اگر بعد از مدتی معین پاسخی از طرف مقابل نرسید، تایمر مزبور لایه پیوند داده را (با استفاده از یک وقفه) مطلع می کند.

در پروتکل های ما، در چنین وضعیتی روال *wait_for_event* مقدار *event = timeout* برمی گرداند. روال های *start_timer* و *stop_timer* نیز به ترتیب تایمر را روشن و خاموش می کنند (البته سپری شدن زمان مقتضی - *timeout* - فقط زمانی اتفاق می افتد که تایمر روشن باشد). یک تایمر را می توان (قبل از منقضی شدن آن) با اجرای مجدد روال *start_timer* ریست کرد.

روال های *start_ack_timer* و *stop_ack_timer* تایمر دیگری را (برای ایجاد فریم های تصدیق دریافت در شرایطی خاص) کنترل می کنند.

از روالهای `enable_network_layer` و `disable_network_layer` در پروتکل‌های پیچیده‌تر استفاده می‌شود (ما در پروتکل‌های ساده این قسمت از این روالها استفاده نخواهیم کرد، چون فرض کرده‌ایم که لایه شبکه همیشه می‌تواند به لایه پیوند داده بسته تحویل دهد). وقتی لایه پیوند داده لایه شبکه را فعال می‌کند (`enable_network_layer`)، لایه شبکه اجازه دارد آماده شدن بسته داده را با یک وقفه به لایه پیوند داده اطلاع دهد (ایسن کار با `event = network_layer_ready` انجام می‌شود). اگر لایه شبکه غیرفعال باشد (`disable_network_layer`)، اجازه چنین کاری را ندارد. با استفاده دقیق و بسجا از روالهای `enable_network_layer` و `disable_network_layer`، لایه پیوند داده می‌تواند مطمئن شود که (هنگام پر شدن بافر) در سیلاب بسته‌های ارسالی از لایه شبکه غرق نخواهد شد.

شماره ترتیبی فریم همیشه بین 0 تا `MAX_SEQ` (و از جمله خود این دو عدد) است، که البته `MAX_SEQ` در پروتکل‌های مختلف می‌تواند متفاوت باشد. شماره ترتیبی فریم‌ها معمولاً یکی یکی اضافه می‌شود، و وقتی به `MAX_SEQ` رسید، دوباره 0 خواهد شد - این کار بر عهده ماکرو `inc` گذاشته شده است. برای سرعت بخشیدن به اجرای این عملیات، `inc` بصورت ماکرو (`macro`) تعریف شده است. [کامپایلر با دیدن یک ماکرو، دستور معادل را جایگزین آن می‌کند، و مانند تابع آنرا فراخوانی نمی‌کند. م.] همانطور که بعداً خواهیم دید، سرعت اجرای پروتکل‌ها یکی از عوامل کلیدی در کارایی شبکه است، و استفاده از ماکرو (بجای تابع) تأثیر زیادی بر بهبود این کارایی دارد. همچنین، از آنجائیکه `MAX_SEQ` در پروتکل‌های مختلف مقادیر متفاوتی دارد، تعریف `inc` بصورت ماکرو امکان می‌دهد تا بدون هیچ مشکلی از آن در پروتکل‌های مختلف استفاده کنیم.

تعاریف شکل ۳-۹ بخشی از پروتکل‌هاییست که در قسمتهای آینده خواهیم نوشت. البته می‌توانستیم آنها را در ابتدای هر پروتکل نیز بیاوریم، ولی با جمع کردن آنها در یک فایل کدهای آینده بسیار ساده‌تر خواهند شد. در زبان C، این کار با استفاده از دستور `#include` و نوشتن نام فایل تعاریف (در اینجا `protocol.h`) انجام می‌شود.

۱-۳-۳ پروتکل یکطرفه نامقید

در اولین مثال یک پروتکل بسیار ساده را در نظر می‌گیریم، که در آن داده‌ها فقط در یک جهت منتقل می‌شوند. لایه شبکه در فرستنده و گیرنده آماده کار هستند، زمان پردازش را می‌توان نادیده گرفت، از نظر بافر هیچ کمبودی وجود ندارد، و مهمتر از همه اینکه کانال ارتباطی بین دو لایه پیوند داده کامل و بدون نقص است، و هیچ خطایی در آن رخ نمی‌دهد. این پروتکل غیرواقعی (که شاید «توپیا» - پروتکل آرمانی - مناسبترین نام برای آن باشد) را در شکل ۳-۱۰ ملاحظه می‌کنید.

```
/* Protocol 1 (utopia) provides for data transmission in one direction only, from
sender to receiver. The communication channel is assumed to be error free
and the receiver is assumed to be able to process all the input infinitely quickly.
Consequently, the sender just sits in a loop pumping data out onto the line as
fast as it can. */
```

```
typedef enum {frame_arrival} event_type;
#include "protocol.h"
```

```
void sender1(void)
{
```

```

frame s; /* buffer for an outbound frame */
packet buffer; /* buffer for an outbound packet */

while (true) {
    from_network_layer(&buffer); /* go get something to send */
    s.info = buffer; /* copy it into s for transmission */
    to_physical_layer(&s); /* send it on its way */
} /* Tomorrow, and tomorrow, and tomorrow,
    Creeps in this petty pace from day to day
    To the last syllable of recorded time.
    - Macbeth, V, v */

}

void receiver1(void)
{
    frame r;
    event_type event; /* filled in by wait, but not used here */

    while (true) {
        wait_for_event(&event); /* only possibility is frame_arrival */
        from_physical_layer(&r); /* go get the inbound frame */
        to_network_layer(&r.info); /* pass the data to the network layer */
    }
}

```

شکل ۳-۱۰. پروتکل یکطرفه نامقید.

این پروتکل دارای دو روال مجزا است: فرستنده و گیرنده. فرستنده در لایه پیوند داده ماشین مبدأ، و گیرنده در لایه پیوند داده ماشین مقصد اجرا می شود. در اینجا از شماره ترتیبی فریمها و تصدیق دریافت استفاده ای نمی شود، بنابراین به MAX_SEQ هم نیازی نیست. تنها رویداد قابل انتظار $frame_arrival$ (رسیدن صحیح و سالم فریم) است.

فرستنده یک حلقه بی انتهای $while$ است که داده ها را با حداکثر توان به بیرون پمپ می کند. بدنه این حلقه سه کار انجام می دهد: آوردن یک بسته از لایه شبکه (که همیشه آماده خدمت است)، ایجاد فریم خروجی با استفاده از متغیر s ، و فرستادن فریم. این پروتکل فقط از فیلد $info$ فریم استفاده می کند، چون فیلدهای دیگر مربوط به کنترل جریان و خطا هستند، که طبق فرض ما چنین محدودیتهایی در اینجا وجود ندارد.

گیرنده هم بهمان اندازه ساده است: انتظار بی پایان برای دریافت فریمی که همیشه سالم و بی نقص است. وقتی یک فریم از راه رسید، روال $wait_for_event$ کنترل را به برنامه اصلی برمی گرداند، و متغیر $event$ را به $frame_arrival$ ست می کند (که بهر حال استفاده ای از آن نمی شود). با فراخوانی روال $from_physical_layer$ ، فریم تازه از راه رسیده از بافر سخت افزاری برداشته شده و در متغیر r قرار داده می شود (تا گیرنده می تواند آنرا بردارد). در پایان، بخش داده این فریم (فیلد $info$) به لایه شبکه فرستاده شده، و لایه پیوند داده به انتظار فریم بعدی می نشیند.

۲-۳-۳ پروتکل توقف-انتظار یکطرفه

حال اجازه دهید غیر واقعی ترین بخش از پروتکل ۱ را کنار بگذاریم: نامحدود بودن توانایی این پروتکل در دریافت بسته ها از لایه شبکه، و پردازش فریمهای ورودی (که به معنای نامحدود بودن بافر لایه پیوند داده در سمت گیرنده است). اما کانال ارتباطی را همچنان بدون خطا، و ارتباط را یکطرفه فرض کرده ایم.

مهمترین مشکلی که با آن روبرو هستیم، این است که چگونه از غرق شدن گیرنده در سیلاب فریمهایی (که پردازش آنها از توان وی خارج است) جلوگیری کنیم. واضحست که، اگر گیرنده برای اجرای روالهای *from_physical_layer* و *to_network_layer* به زمان Δt نیاز داشته باشد، سرعت متوسط ارسال فرستنده بایستی از یک فریم بر Δt کمتر باشد. همچنین اگر فرض کنیم که سخت افزار گیرنده بطور خودکار عمل بافر کردن فریمها را انجام نمی دهد، فرستنده نباید قبل از برداشته شدن یک فریم از بافر لایه فیزیکی (که توسط روال *from_physical_layer* انجام می شود)، فریم بعدی را ارسال کند چون در غیر اینصورت فریم قبلی از بین می رود (به این حالت روهم نویسی - overrun - می گویند).

در شرایط خاصی (مانند ارتباط سنکرون، و گیرندهای که تنها وظیفه آن گرفتن اطلاعات از خط ورودی است)، با ایجاد تأخیر در قسمت فرستنده پروتکل ۱ و کند کردن آن می توان به اهداف فوق دست یافت. اما بسیار محتملتر است که یک لایه پیوند داده مجبور باشد چندین خط ورودی را پردازش کند، که در این حالت فاصله زمانی دریافت فریمها و پردازش آنها می تواند بسیار متغیر باشد. اگر طراحان شبکه بتوانند بدترین حالت گیرنده را محاسبه کنند، می توانند فرستنده را آنقدر کند کنند که روهم نویسی هرگز اتفاق نیفتد. اما این روش بسیار محافظه کارانه است و بنحو بسیار بدی پهنای باند را تلف می کند، مگر اینکه تفاوت بهترین و بدترین حالت چندان زیاد نباشد (یعنی تفاوت پاسخهای لایه پیوند داده ناچیز باشد).

راه حل بهتر این معضل، برگرداندن بازخور (feedback) از گیرنده به فرستنده است. بعد از تحویل بسته به لایه شبکه، گیرنده یک فریم کوچک (که لازم نیست معنی خاصی هم داشته باشد) به فرستنده می فرستد، که در واقع مجوز ارسال فریم بعدی محسوب می شود. فرستنده بعد از ارسال یک فریم، آنقدر منتظر می ماند تا این فریم کوچک (که در واقع همان تصدیق دریافت - acknowledgement - است) از راه برسد. استفاده از بازخور گیرنده برای اطلاع به فرستنده (و دادن مجوز ارسال فریمهای بعدی) یکی از نمونه های کنترل جریان (flow control)، که قبلاً به آن اشاره کردیم، است.

پروتکلهایی که در آنها فرستنده قبل از ارسال فریم بعدی منتظر تصدیق دریافت فریم قبلی از گیرنده می ماند، به پروتکلهای توقف-انتظار (stop-and-wait) معروفند. در شکل ۳-۱۱ یک نمونه از پروتکلهای توقف-انتظار را ملاحظه می کنید.

/* Protocol 2 (stop-and-wait) also provides for a one-directional flow of data from sender to receiver. The communication channel is once again assumed to be error free, as in protocol 1. However, this time, the receiver has only a finite buffer capacity and a finite processing speed, so the protocol must explicitly prevent the sender from flooding the receiver with data faster than it can be handled. */

```
typedef enum {frame_arrival} event_type;
"h.locotorp" edulcni#
```

```

void sender2(void)
{
    frame s;                /* buffer for an outbound frame */
    packet buffer;          /* buffer for an outbound packet */
    event_type event;       /* frame_arrival is the only possibility */

    while (true) {
        from_network_layer(&buffer); /* go get something to send */
        s.info = buffer;             /* copy it into s for transmission */
        to_physical_layer(&s);       /* bye-bye little frame */
        wait_for_event(&event);     /* do not proceed until given the go ahead */
    }
}

void receiver2(void)
{
    frame r, s;             /* buffers for frames */
    event_type event;       /* frame_arrival is the only possibility */
    while (true) {
        wait_for_event(&event); /* only possibility is frame_arrival */
        from_physical_layer(&r); /* go get the inbound frame */
        to_network_layer(&r.info); /* pass the data to the network layer */
        to_physical_layer(&s);     /* send a dummy frame to awaken sender */
    }
}

```

شکل ۳-۱۱. پروتکل توقف-انتظار یکطرفه.

با اینکه این پروتکل یکطرفه (simplex) است (یعنی ما فقط در یک جهت ارسال می‌کنیم)، اما فریمها می‌توانند در هر دو جهت رفت و آمد کنند. برای این منظور لازم است کانال ارتباطی ما چنین قابلیت داشته باشد؛ البته یک کانال دوطرفه ناهمزمان (half-duplex) هم کفایت می‌کند، چون فرستنده و گیرنده در آن واحد اقدام به فرستادن فریمها نمی‌کنند: فرستنده یک فریم می‌فرستد، گیرنده پاسخ می‌دهد، فرستنده فریم بعدی را می‌فرستد، گیرنده پاسخ می‌دهد، و الی آخر.

در اینجا هم (مانند پروتکل ۱) فرستنده همان سه کار قبلی را انجام می‌دهد: آوردن یک بسته از لایه شبکه، ایجاد فریم خروجی، و فرستادن آن. اما برخلاف پروتکل ۱، قبل از ادامه کار (آوردن بسته بعدی و ارسال آن در قالب یک فریم) باید منتظر رسیدن فریم تصدیق دریافت از گیرنده بماند. نیازی نیست که لایه پیوند داده فرستنده فریم دریافتی را بررسی کند، چون فقط یک احتمال وجود دارد: این فریم همیشه تصدیق دریافت گیرنده است. تنها تفاوت receiver2 با receiver1 این است که بعد از تحویل بسته به لایه شبکه و قبل از ورود به حالت انتظار، receiver2 یک فریم تصدیق دریافت به فرستنده باز پس می‌فرستد. از آنجائیکه فقط خود فریم مهم است و نه محتویات آن، گیرنده هیچ داده‌ای در فیلد info این فریم قرار نمی‌دهد.

۳-۳-۳ پروتکل یکطرفه برای کانالهای نوبزدار

اکنون به یک حالت واقعی‌تر می‌پردازیم: کانالهایی که نویز دارند. فریمها می‌توانند با خطا به مقصد برسند، و یا بکلی گم شده و اصلاً به مقصد نرسند. با این حال، فرض می‌کنیم که اگر فریمی با خطا به مقصد رسید، سخت‌افزار لایه فیزیکی با محاسبه جمع تطبیقی متوجه خطا می‌شود. پروتکل ما در یک حالت به اشتباه عمل خواهد کرد: خطای رخ داده آنقدر شدید باشد که جمع تطبیقی تصادفاً درست از کار در آید (اتفاقی که بسیار نامحتمل است). در نگاه اول با یک تغییر کوچک در پروتکل ۲ (اضافه کردن یک تایمر) می‌توان آن را با وضعیت جدید تطبیق داد. فرستنده می‌تواند در هر زمانی یک فریم بفرستد، ولی گیرنده فقط وقتی فریم تصدیق دریافت را برمی‌گرداند که این فریم را بدستی دریافت و پردازش کرده باشد. اگر فریم ناقص به مقصد برسد، دور انداخته خواهد شد. بعد از مدتی تایمر فرستنده به انتها می‌رسد، و چون هنوز تصدیق دریافت گیرنده را نگرفته، مجدداً اقدام به ارسال فریم می‌کند. این ماجرا تا زمانی که فریم به سلامت به مقصد برسد، تکرار خواهد شد.

اما طرح بالا یک مشکل اساسی دارد. قبل از خواندن ادامه کتاب، کمی فکر کنید و ببینید می‌توانید متوجه اشکال آن شوید.

برای درک مشکل، بیاد بیاورید که وظیفه برقراری یک کانال ارتباطی عاری از خطا بین لایه‌های شبکه بر عهده لایه پیوند داده است. لایه شبکه ماشین A یک سری بسته به لایه پیوند داده می‌دهد، و باید مطمئن باشد که این بسته‌ها به همان ترتیبی که ارسال شده‌اند بدست لایه شبکه ماشین B خواهند رسید. بویژه، لایه شبکه ماشین B هیچ راهی ندارد تا بفهمد که یک بسته گم شده یا تکراریست. به همین دلیل لایه پیوند داده ماشین B باید تضمین کند که هیچ بسته‌ای گم نمی‌شود، و یا تکراری نیست.

سناریوی زیر را در نظر بگیرید:

۱. لایه شبکه ماشین A بسته ۱ را به لایه پیوند داده می‌دهد. این بسته صحیح و سالم به لایه پیوند داده ماشین B می‌رسد، و تحویل لایه شبکه می‌شود. ماشین B یک فریم تصدیق دریافت به A می‌فرستد.
۲. فریم تصدیق دریافت ماشین B در راه از بین می‌رود، و هرگز به A نمی‌رسد. اگر فقط فریمهای داده گم می‌شدند و این اتفاق برای فریمهای کنترلی نمی‌افتاد، زندگی چقدر شیرین‌تر بود! ولی متأسفانه کانالهای مخابراتی اهل تبعیض نیستند!
۳. تایمر لایه پیوند داده A به انتها می‌رسد، و چون هیچ فریم تصدیق دریافتی بدستش نرسیده، (باشتاباه) تصور می‌کند که فریم به مقصد نرسیده (یا خراب شده)، پس آنرا دوباره می‌فرستد.
۴. فریم تکراری (در کمال صحت و سلامت) به لایه پیوند داده B می‌رسد، و این لایه هم (بی‌خبر از همه جا) آنرا به لایه شبکه تحویل می‌دهد. تصور کنید که اگر A در حال ارسال یک فایل به B باشد، تکراری بودن بخشی از آن چه فاجعه‌ای پیاور خواهد آورد. همانطور که می‌بینید، پروتکل ما یک شکست کامل است.

چیزی که ما به آن احتیاج داریم، وسیله‌ایست که بتوان فریمهای تکراری را از فریمهایی که برای اولین بار دریافت می‌شوند، تشخیص داد. راه حل واضح این مشکل آن است که فرستنده یک شماره ترتیبی (sequence number) در سرآیند فریمهایی که می‌فرستد، قرار دهد. گیرنده می‌تواند این شماره را چک کرده، و فریمهای تکراری را دور بیندازد.

از آنجائیکه سرآیند یک فریم باید حتی‌الامکان کوچک باشد، سؤالی که پیش می‌آید اینست که: حداقل تعداد بیت‌های لازم برای فیلد شماره ترتیبی چقدر است؟ در پروتکل ما تنها ابهام در فریم m و فریم بعدی آن یعنی $m + 1$ است. اگر فریم m خراب شود یا از بین برود، گیرنده فریم تصدیق دریافت آنرا بر نمی‌گرداند، پس فرستنده سعی می‌کند آنرا دوباره بفرستد. همین که این فریم سالم به مقصد رسید، گیرنده فریم تصدیق دریافت را به فرستنده پس می‌فرستد. همین جاست که مشکل بروز می‌کند: اگر فریم تصدیق دریافت صحیح و سالم به فرستنده بفرستد،

فرستنده فریم بعدی (یعنی $m + 1$) را می فرستد، در غیر اینصورت فریم m را خواهد فرستاد. برای ارسال فریم $m + 2$ ، فرستنده باید قبلاً تصدیق دریافت فریم $m + 1$ را گرفته باشد. اما این بدان معناست که فریم m به سلامت به مقصد رسیده و تصدیق دریافت آن هم پدرستی به فرستنده برگشت داده شده است (چون در غیر اینصورت فرستنده فریم $m + 1$ را هم نمی فرستاد، چه رسد به فریم $m + 2$). بنابراین، تنها ابهامی که می تواند وجود داشته باشد، بین یک فریم و فریم بعدی آن است. برای تشخیص این دو هم یک شماره ترتیبی یک بیتی (0 یا 1) کافیت. عبارت دیگر، در هر لحظه گیرنده باید بدنبال شماره بعدی باشد. اگر فریمی با شماره اشتباه دریافت شد، گیرنده آنرا تکراری تلقی کرده و دور می اندازد. اما اگر شماره ترتیبی فریم درست بود، به لایه شبکه تحویل داده می شود. با این توصیف فیلد شماره ترتیبی باید در مدول 2 افزایش داده شود (بعبارت دیگر، 1 به 0 تبدیل می شود، و 0 به 1).

پروتکلی با این مشخصات را در شکل ۳-۱۲ ملاحظه می کنید. به پروتکل هایی که فرستنده برای ارسال فریم بایستی منتظر یک تصدیق دریافت مثبت بماند، PAR (تصدیق دریافت مثبت با ارسال مجدد - Positive Acknowledgement with Retransmission) یا ARQ (درخواست تکرار خودکار - Automatic Repeat reQuest) نیز گفته می شود. این پروتکل هم، مانند پروتکل ۲، فقط در یک جهت داده می فرستد. تفاوت پروتکل ۳ با دو نای قبلی اینست که، روالهای فرستنده و گیرنده متغیری دارند که مقدار آن حتی در زمانی که لایه پیوند داده به حالت انتظار می رود، دست نخورده باقی می ماند. فرستنده باید شماره ترتیبی فریم بعدی که می خواهد بفرستد، را بداند: `next_frame_to_send`؛ و گیرنده هم باید شماره ترتیبی فریم بعدی که باید منتظر آن باشد، را بداند: `frame_expected`.

```
/* Protocol 3 (par) allows unidirectional data flow over an unreliable channel. */
#define MAX_SEQ 1 /* must be 1 for protocol 3 */
typedef enum {frame_arrival, cksum_err, timeout} event_type;
#include "protocol.h"
void sender3(void)
{
    seq_nr next_frame_to_send; /* seq number of next outgoing frame */
    /*
    frame s; /* scratch variable */
    packet buffer; /* buffer for an outbound packet */
    event_type event;
    next_frame_to_send = 0; /* initialize outbound sequence numbers */
    from_network_layer(&buffer); /* fetch first packet */
    while (true) {
        s.info = buffer; /* construct a frame for transmission */
        /*
        s.seq = next_frame_to_send; /* insert sequence number in frame */
        /*
        to_physical_layer(&s); /* send it on its way */
        start_timer(s.seq); /* if answer takes too long, time out */
    }
}
```



```

*/
    wait_for_event(&event);          /* frame_arrival, cksum_err, timeout
*/
    if (event == frame_arrival) {
        from_physical_layer(&s);      /* get the acknowledgement */
        if (s.ack == next_frame_to_send) {
            stop_timer(s.ack);        /* turn the timer off */
            from_network_layer(&buffer); /* get the next one to send */
            inc(next_frame_to_send);   /* invert next_frame_to_send */
        }
    }
}

void receiver3(void)
{
    seq_nr frame_expected;
    frame r, s;
    event_type event;
    frame_expected = 0;
    while (true) {
        wait_for_event(&event);          /* possibilities: frame_arrival,
cksum_err */
        if (event == frame_arrival) {    /* a valid frame has arrived. */
            from_physical_layer(&r);      /* go get the newly arrived frame
*/
            if (r.seq == frame_expected) { /* this is what we have been
waiting for. */
                to_network_layer(&r.info); /* pass the data to the network
layer */
                inc(frame_expected);      /* next time expect the other
sequence nr */
            }
            s.ack = 1 - frame_expected;    /* tell which frame is being acked
*/
            to_physical_layer(&s);        /* send acknowledgement */
        }
    }
}

```

فرستنده، بعد از ارسال فریم، تایمر را راه می اندازد. اگر تایمر از قبل در حال اجرا باشد، این کار آنرا ریست کرده و آماده کار بعدی می کند. فاصله زمانی تایمر باید بگونه ای انتخاب شود که وقت کافی برای سه رویداد بدست دهد: رسیدن فریم به گیرنده، پردازش آن در گیرنده (در بدترین حالت)، و برگشت فریم تصدیق دریافت به فرستنده. فقط پس از سپری شدن این زمان است که می توان مطمئن شد فریم (یا تصدیق دریافت آن) به مقصد نرسیده، و فرستنده باید دوباره آنرا بفرستد. اگر فاصله زمانی تایمر کم انتخاب شود، تعداد دفعاتی که فرستنده فریم تکراری می فرستد افزایش یافته، و (با اینکه این کار تأثیر منفی روی گیرنده ندارد) به کارایی سیستم لطمه می زند. بعد از ارسال فریم و راه انداختن تایمر، فرستنده منتظر یک اتفاق هیجان انگیز می ماند. البته فقط سه احتمال برای چنین اتفاقی وجود دارد: فریم تصدیق دریافت صحیح و سالم از راه برسد، فریم تصدیق دریافت با خطا وارد شود، تایمر منقضی شود (زمان مشخص شده به انتها برسد). در حالت اول، فرستنده بسته دیگری از لایه شبکه گرفته، و در بافر خود (buffer) قرار می دهد. سپس، شماره ترتیبی فریم (next_frame_to_send) را بالا می برد. اما در دو حالت دیگر (خراب شدن فریم تصدیق دریافت یا نرسیدن آن)، بافر و شماره ترتیبی هیچکدام تغییری نمی کند، بنابراین در هیچ حالتی فریم تکراری فرستاده نخواهد شد. وقتی یک فریم به گیرنده می رسد، گیرنده شماره ترتیبی آن را چک می کند، تا از تکراری نبودن آن مطمئن شود. این فریم فقط در صورت تکراری نبودن به لایه شبکه تحویل داده می شود. بدین ترتیب، فریمهای تکراری و خراب به لایه شبکه نخواهند رسید.

۴-۳ پروتکل های پنجره لغزنده

در پروتکل های قبل، فریمهای داده فقط در یک جهت ارسال می شدند. اما در عمل باید بتوانیم در هر دو جهت انتقال داده داشته باشیم. یکی از راههای داشتن یک کانال دوطرفه همزمان (full-duplex) استفاده از دو کانال یکطرفه (simplex) در دو جهت مخالف است، که هر کدام فقط در یک جهت داده می فرستند (و البته فریمهای تصدیق دریافت می کنند). اما این روش چیزی جز اتلاف پهنای باند (و پول) نیست. روش بهتر استفاده از یک کانال واحد برای ارسال داده در دو جهت است (مگر نه اینکه در پروتکل های ۲ و ۳ در هر دو جهت فریم ارسال کردیم). از آنجائیکه در این مدل، فریمهای داده و تصدیق دریافت در هر دو جهت می توانند فرستاده شوند، باید کاری کنیم که گیرنده بتواند آنها را از یکدیگر تشخیص دهد. برای این منظور می توانیم از فیلد info در سرآیند فریمها استفاده کنیم. با اینکه ترکیب فریمهای داده و تصدیق دریافت روی یک مدار واحد (بجای دو مدار جداگانه) یک قدم به جلو محسوب می شود، اما باز هم می توان کارایی سیستم را بهبود بخشید. وقتی گیرنده یک فریم داده دریافت می کند، بجای اینکه بلافاصله یک فرم کنترلی پس بفرستد، منتظر می ماند تا بسته بعدی را برای ارسال از لایه شبکه بگیرد. تصدیق دریافت فریم قبلی در فیلد ack فریم داده ای که اکنون می خواهد فرستاده شود، قرار داده می شود، و در واقع فریم تصدیق دریافت از فریم داده سواری مجانی (piggyback) می گیرد (و به همین نام هم خوانده می شود). یکی از مزایای تکنیک سواری مجانی نسبت به ارسال مستقل فریمهای تصدیق دریافت، استفاده بهینه تر از پهنای باند موجود است: فیلد ack فقط چند بیت از سرآیند را اشغال می کند، در حالیکه یک فریم مستقل برای خود سرآیند، جمع تطبیقی، و تصدیق دریافت دارد. علاوه بر آن، هر چه تعداد فریمهایی که در یک جهت فرستاده می شوند کمتر باشد، گیرنده بهتر می تواند به کارهای دیگرش (پردازش فریمهای رسیده و خالی کرن بافرها) برسد، و این هم به بهبود کارایی سیستم کمک می کند. در پروتکلی که در این قسمت می نویسیم، فیلد سواری مجانی فقط یک بیت به سرآیند فریم اضافه می کند (و بندرت پیش می آید که مقدار آن از چند بیت بیشتر شود).

اما سواری مجانی هم خالی از اشکال نیست. برای مثال، در این حالت لایه پیوند داده گیرنده چقدر باید منتظر بسته از لایه شبکه خود شود؟ اگر خیلی منتظر بماند، تایمر فرستنده به انتها رسیده و فریم را تکرار می کند، که این نقض غرض (از ارسال فریم تصدیق دریافت) است. اگر لایه پیوند داده قدرت پیشگویی داشت، می توانست زمان دریافت بسته بعدی از لایه شبکه را پیشگویی کند، و آنوقت می توانست تصمیم بگیرد که منتظر این بسته بماند یا بلافاصله فریم تصدیق دریافت را به فرستنده بفرستد. اما متأسفانه لایه پیوند داده نمی تواند آینده را پیشگویی کند، پس باید روش ساده تری (انتظار بمدت ثابت، مثلاً چند میلی ثانیه) پیدا کنیم. اگر در این فاصله بسته ای از لایه شبکه رسید، لایه پیوند داده تصدیق دریافت را سوار آن می کند؛ در غیر اینصورت یک فریم مستقل تصدیق دریافت به سمت فرستنده ارسال می کند.

پروتکل هایی که در این قسمت خواهید دید، به کلاس پروتکل های پنجره لغزنده (sliding window) تعلق دارند، که فقط از نظر کارایی، پیچیدگی و بافر با هم تفاوت دارند. در این پروتکل ها، مانند سایر پروتکل های پنجره لغزنده، هر فریم خروجی یک شماره ترتیبی (از 0 تا یک حداکثر) دارد. حداکثر شماره فریم ها معمولاً $2^n - 1$ است، بنابراین در یک فیلد n -بیتی بخوبی جا می شود. پروتکل پنجره لغزنده توقف-انتظار از $n = 1$ استفاده می کرد، اما در پروتکل های دیگر این عدد می تواند بیشتر باشد.

ایده اصلی در تمام پروتکل های پنجره لغزنده این است که، فرستنده در هر لحظه از زمان لیستی از شماره های ترتیبی متناظر با فریم هایی که می تواند ارسال کند، در اختیار دارد. اصطلاحاً گفته می شود که این فریم ها در پنجره ارسال (sending window) قرار دارند. گیرنده هم یک پنجره دریافت (receiving window) دارد که متناظر است با فریم هایی که مجاز به دریافت آنهاست. الزامی نیست که پنجره ارسال و پنجره دریافت حد پائین و بالای مشابه داشته باشند، یا حتی هم اندازه باشند. در برخی پروتکل ها اندازه این پنجره ها ثابت است، ولی در پروتکل های دیگر می تواند کوچک یا بزرگ شود.

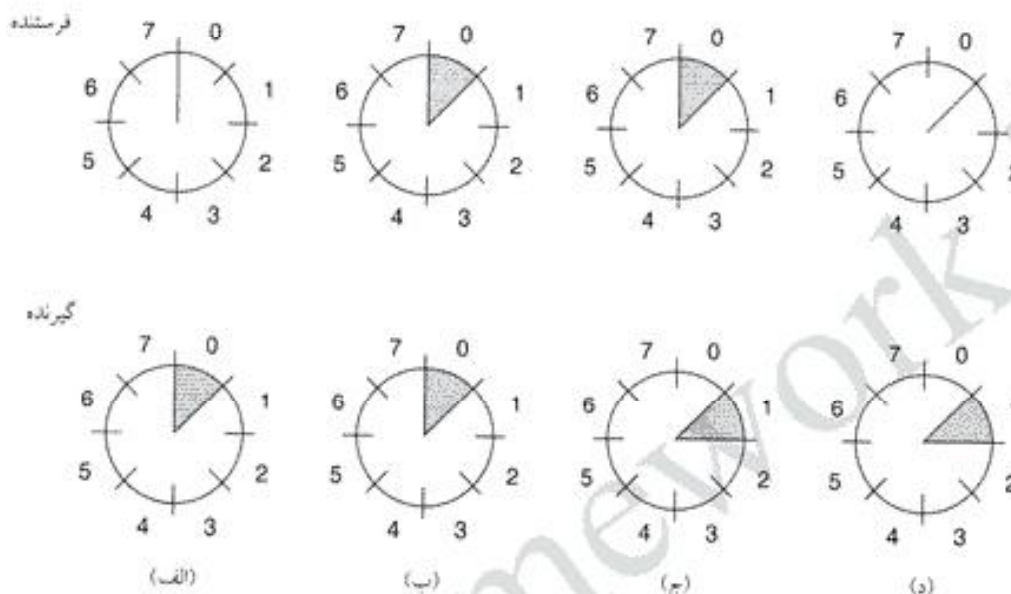
با اینکه این پروتکل ها آزادی عمل بیشتری به لایه پیوند داده در ارسال و دریافت فریم ها می دهند، لازم است مجدداً تأکید کنیم که تحویل بسته ها به لایه شبکه مقصد باید با همان ترتیبی صورت گیرد که در ماشین مبدأ تحویل لایه پیوند داده شده اند. (و یک بار دیگر خاطر نشان می کنیم که، لایه فیزیکی یک کانال ارتباطی ساده است که فریم ها را به همان ترتیبی که به آن داده شده، منتقل می کند.)

شماره های موجود در پنجره ارسال شماره فریم هاییست که باید ارسال شوند، و یا ارسال شده اند ولی هنوز تصدیق دریافت آنها برگشته است. وقتی یک بسته جدید از لایه شبکه می رسد، لایه پیوند داده بالاترین شماره ترتیبی موجود را به آن می دهد، و لبه بالایی پنجره ارسال را یکی زیاد می کند. وقتی یک تصدیق دریافت وارد شد، لبه پائینی پنجره ارسال نیز بالا برده می شود. بدین ترتیب پنجره ارسال همیشه شامل لیست فریم هاییست که دریافت آنها هنوز توسط گیرنده تصدیق نشده است. در شکل ۳-۱۳ یک نمونه را ملاحظه می کنید.

از آنجائیکه امکان کم یا خراب شدن فریم هایی که در حال حاضر در پنجره ارسال قرار دارند، همیشه وجود دارد، فرستنده باید آنها را برای ارسال مجدد (احتمالی) در حافظه نگه دارد. بنابراین اگر حداکثر اندازه این پنجره n باشد، فرستنده باید بافری باندازه n فریم تصدیق نشده داشته باشد. اگر پنجره ارسال از حداکثر پیش بینی شده بزرگتر شود، لایه پیوند داده باید گرفتن بسته از لایه شبکه را تا زمان آزاد شدن بافر متوقف کند.

پنجره دریافت در گیرنده متناظر است با فریم هایی که گیرنده مجاز به دریافت آنهاست. هر فریمی که خارج از این پنجره قرار گیرد، بدون هیچ توضیحی دور انداخته خواهد شد. وقتی فریمی که شماره ترتیبی آن معادل لبه پائین پنجره دریافت است، از راه می رسد، به لایه شبکه تحویل شده و پس از ایجاد تصدیق دریافت آن، پنجره دریافت یک واحد می چرخد. برخلاف پنجره فرستنده، پنجره گیرنده همیشه به همان اندازه اولیه می ماند. توجه

کنید که پنجره دریافت ۱ بمعنای اینست که لایه پیوند داده فریمها را فقط به ترتیب می پذیرد، ولی در پنجره های بزرگتر الزاماً چنین نیست (با این حال، لایه شبکه همیشه داده ها را به ترتیب صحیح تحویل لایه پیوند داده می دهد).



شکل ۳-۱۳. یک پنجره لغزنده یک واحدی، با شعاعه ترتیبی ۳-بینی. (الف) در شروع کار. (ب) بعد از ارسال اولین فریم. (ج) بعد از آنکه اولین فریم دریافت شد. (د) بعد از آنکه فرستنده اولین تصدیق دریافت را گرفت.

در شکل ۳-۱۳ حداکثر اندازه پنجره ۱ است. در لحظه اول لبه های پائین و بالای پنجره ارسال یکی هستند، ولی با گذشت زمان موقعیت آنها طبق شکل تغییر می کند.

۱-۴-۳ پروتکل پنجره لغزنده ۱-بیتی

قبل از پرداختن به حالت کلی، اجازه دهید ابتدا پروتکلی با پنجره لغزنده ۱-بیتی را مورد بررسی قرار دهیم. در واقع این پروتکل نوعی پروتکل توقف-انتظار است، چون فرستنده قبل از گرفتن تصدیق دریافت فریم فرستاده شده، فریم بعدی را ارسال نخواهد کرد.

در شکل ۳-۱۴ پروتکل پنجره لغزنده ۱-بیتی را ملاحظه می کنید. مانند پروتکل های قبلی، این پروتکل هم با تعریف متغیرها شروع می شود. متغیر `next_frame_to_send` فریم بعدیست که فرستنده باید بفرستد. در طرف مقابل هم، متغیر `frame_expected` فریمی را نشان می دهد که گیرنده منتظر آن است. در هر دو طرف، تنها حالت های مجاز فقط ۰ یا ۱ است.

```
/* Protocol 4 (sliding window) is bidirectional. */
#define MAX_SEQ 1 /* must be 1 for protocol 4 */
typedef enum {frame_arrival, cksum_err, timeout} event_type;
#include "protocol.h"
void protocol4 (void)
```



```

{
    seq_nr next_frame_to_send;           /* 0 or 1 only */
    seq_nr frame_expected;               /* 0 or 1 only */
    frame r, s;                         /* scratch variables */
    packet buffer;                       /* current packet being sent */
    event_type event;
    next_frame_to_send = 0;              /* next frame on the outbound
stream */
    frame_expected = 0;                  /* frame expected next */
    from_network_layer(&buffer);         /* fetch a packet from the network
layer */
    s.info = buffer;                     /* prepare to send the initial frame */
    s.seq = next_frame_to_send;          /* insert sequence number into
frame */
    s.ack = 1 - frame_expected;          /* piggybacked ack */
    to_physical_layer(&s);               /* transmit the frame */
    start_timer(s.seq);                  /* start the timer running */
    while (true) {
        wait_for_event(&event);          /* frame_arrival, cksum_err, or
timeout */
        if (event == frame_arrival) {    /* a frame has arrived
undamaged. */
            from_physical_layer(&r);      /* go get it */
            if (r.seq == frame_expected) { /* handle inbound frame
stream. */
                to_network_layer(&r.info); /* pass packet to network
layer */
                inc(frame_expected);      /* invert seq number expected
next */
            }
            if (r.ack == next_frame_to_send) { /* handle outbound frame
stream. */
                stop_timer(r.ack);        /* turn the timer off */
                from_network_layer(&buffer); /* fetch new pkt from network
layer */
                inc(next_frame_to_send);  /* invert sender's sequence
number */
            }
        }
        s.info = buffer;                 /* construct outbound frame */
        s.seq = next_frame_to_send;      /* insert sequence number into it

```

```

*/
s.ack = 1 - frame_expected;          /* seq number of last received
frame */
to_physical_layer(&s);                /* transmit a frame */
start_timer(s.seq);                  /* start the timer running */
}
}

```

شکل ۳-۱۲. پروتکل پنجره لغزنده ۱-بیتی.

در شرایط عادی، یکی از دو طرف پیش دستی کرده و اولین فریم را می فرستد. بعبارت دیگر، فقط یکی از دو لایه پیوند داده روالهای `to_physical_layer` و `start_timer` را خارج از حلقه اصلی برنامه اجرا می کند. اگر در پیشامدی نادر هر دو طرف بطور همزمان شروع به ارسال اولین فریم کنند، وضعیت عجیبی پیش می آید، که بعداً آنرا توضیح خواهیم داد. ماشین شروع کننده اولین بسته را از لایه شبکه گرفته، یک فریم از آن می سازد، و سپس ارسال می کند. وقتی این فریم (یا هر فریم دیگری) به طرف مقابل رسید، لایه پیوند داده گیرنده چک می کند که تکراری نباشد (درست مثل پروتکل ۳). اگر این همان فریم مورد نظر باشد، به لایه شبکه تحویل شده و پنجره دریافت یک واحد به جلو لغزانده می شود.

فیلد تصدیق دریافت حاوی شماره آخرین فریمیست که بدون خطا دریافت شده است. اگر این شماره با شماره فریمی که فرستنده در صدد ارسال آن است یکی باشد، فرستنده می فهمد که دیگر نیازی به فریم داخل بافر ندارد و می تواند بسته بعدی را از لایه شبکه بگیرد. اگر شماره ها یکی نباشند، فرستنده باید به ارسال همان فریم قبلی ادامه دهد. وقتی یک فریم دریافت شود، فریمی نیز پس فرستاده می شود.

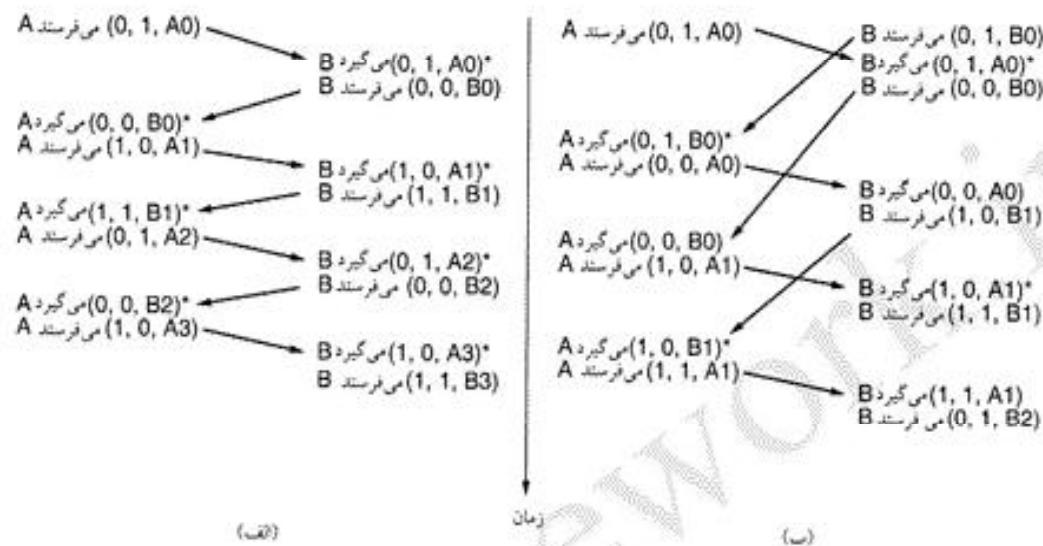
حال اجازه دهید ببینیم پروتکل ۴ در شرایط غیر عادی چگونه رفتار می کند. فرض کنید ماشین A در صدد ارسال فریم ۰ به کامپیوتر B است، و در همان زمان B نیز تصمیم می گیرد فریم ۰ خود را به A بفرستد. در ضمن فرض می کنیم که A مسابقه را زودتر شروع می کند، ولی فاصله زمانی تایمر آن بسیار کوتاه است. در نتیجه، ماشین A پشت سر هم فریمهای یکسان، با $seq = 0$ و $ack = 1$ ، به B می فرستد.

وقتی اولین فریم به کامپیوتر B رسید، پذیرفته شده و `frame_expected` به ۱ ست می شود؛ تمام فریمهای بعدی رد خواهند شد، چون اینک B در انتظار فریمی با شماره ترتیبی ۱ است نه ۰. علاوه بر آن، چون تمام در فریمهای تکراری $ack = 1$ و B همچنان منتظر تصدیق دریافت فریم ۰ است، ماشین B گرفتن بسته از لایه شبکه خود را متوقف خواهد کرد.

بعد از آن که تمام فریمهای تکراری به B رسیدند، B یک فریم با $seq = 0$ و $ack = 0$ به A می فرستد. بالاخره یکی از این فریمها سالم به A می رسد، و باعث می شود تا A ارسال فریم بعدی را شروع کند. همانطور که می بینید، در هیچ شرایطی بسته تکراری به لایه شبکه نمی رسد، بسته ای گم نمی شود، و سیستم قفل نمی کند.

اما اگر هر دو طرف در یک لحظه شروع به ارسال اولین فریم کنند، وضعیت عجیبی پیش می آید. این مشکل سنکرون شدن را در شکل ۳-۱۵ (ب) مشاهده می کنید. (در شکل ۳-۱۵ الف برای مقایسه عملکرد عادی پروتکل ۴ نشان داده شده است.) اگر B قبل از ارسال فریمهای خود منتظر دریافت اولین فریم A بماند، هیچ مشکلی پیش نمی آید و تمام بسته ها براحتی پذیرفته می شوند (شکل الف). ولی اگر هر دو با هم مخابره اولین فریم را شروع کنند، این فریمها با یکدیگر تصادم کرده، و حالت (ب) پیش می آید. در (الف) دریافت هر فریم باعث گرفتن یک بسته از لایه شبکه می شود، و هیچ فریم تکراری هم وجود ندارد. اما در (ب) با وجود اینکه هیچ خطایی هم در

کانال وجود ندارد، نصف فریمها تکراری هستند. این وضعیت هنگامی که تایمر یکی از دو طرف بیش از حد کوتاه باشد، نیز پیش می آید (حتی اگر دو طرف همزمان شروع به مخابره فریمهای خود نکرده باشند). در چنین وضعیتی حتی امکان دارد برخی از فریمها سه یا چهار بار تکرار شوند.



شکل ۳-۱۵. دو سناریوی پروتکل ۴. (الف) حالت عادی. (ب) حالت غیرعادی. اعداد داخل پرانتز از چپ به راست عبارتند از: seq، شماره بسته. بسته‌هایی که پذیرفته و به لایه شبکه تحویل می‌شوند، با * مشخص شده‌اند.

۲-۴-۳ پروتکل «N تا به عقب برگرد»

تا اینجا بطور ضمنی فرض کرده بودیم که زمان رسیدن فریم به مقصد بعلاوه زمان برگشت فریم تصدیق ناچیز است. اما گاهی این فرض آشکارا نادرست است. در چنین مواردی زمان طولانی رفت و برگشت فریم می‌تواند تأثیر چشمگیری روی کارایی مصرف پهنای باند داشته باشد. بعنوان مثال، یک کانال ماهواره‌ای با پهنای باند 50 kbps و زمان تأخیر رفت و برگشت 500 msec را در نظر بگیرید. فرض کنید می‌خواهیم با این لینک ماهواره‌ای و با استفاده از پروتکل ۴ فریمهای 1000 بیتی ارسال کنیم. در لحظه $t = 0$ فرستنده مخابره اولین فریم را شروع می‌کند. و در $t = 20$ msec کار ارسال فریم به پایان می‌رسد. اما در بهترین شرایط (و با فرض اینکه در گیرنده نیز هیچ تأخیری وجود ندارد)، تا $t = 270$ msec این فریم هنوز به گیرنده نرسیده، و تا $t = 520$ msec نیز مسلماً فریم تصدیق دریافت به دست فرستنده نخواهد رسید. این بدان معناست که فرستنده در 500/520 یا 96% زمان باید متوقف بماند، و نمی‌تواند چیزی بفرستد؛ عبارت دیگر، از فقط 4% پهنای باند استفاده می‌کند. پیداست که ترکیب تأخیر طولانی در کانال، پهنای باند زیاد، و فریمهای کوچک چیزی جز اتلاف وحشتناک منابع نیست.

مشکلی که در بالا دیدید از آنجا ناشی می‌شد که فرستنده برای ارسال یک فریم باید منتظر تصدیق دریافت فریم قبلی بماند. اما اگر این قید را برداریم، می‌توانیم به کارایی بهتری دست پیدا کنیم. در واقع بجای 1 فریم، معمولاً فرستنده می‌تواند w فریم بفرستد و پس از آن منتظر رسیدن فریمهای تصدیق بماند. اگر w طوری انتخاب شود که فرستنده در تمام مدت زمان تأخیر رفت و برگشت در حال ارسال فریم باشد، می‌تواند بدون مشکل از تمام پهنای باند استفاده کند. در مثال بالا w باید حداقل 26 باشد. فرستنده مانند قبل ارسال اولین فریم را در $t = 0$ شروع می‌کند، و زمانی که فریم 26 را می‌فرستد ($t = 520$ msec)، تصدیق فریم 0 را دریافت خواهد کرد. از آن به

بعد نیز فریمهای تصدیق دریافت هر 20 msec از راه می رستند، و فرستنده می تواند بطور پیوسته به ارسال فریمها ادامه دهد. در تمام زمانها فرستنده 25 یا 26 فریم در بافر خود دارد که هنوز تصدیق دریافت آنها را نگرفته است. به بیان دیگر، اندازه پنجره ارسال حداکثر 26 است.

پنجره ارسال بزرگ فقط زمانی لازم می شود که حاصلضرب پهنای باند $\times$ تأخیر رفت و برگشت عددی بزرگ باشد. اگر پهنای باند زیاد باشد، حتی با تأخیر کم نیز فرستنده سرعت پنجره ارسال را پُر می کند. اگر تأخیر رفت و برگشت زیاد باشد (مانند کانالهای ماهواره ای GEO)، حتی در پهنای باند متوسط نیز پنجره ارسال بزودی پُر می شود. ظرفیت یک کانال اساساً با حاصلضرب این دو عامل (پهنای باند و تأخیر رفت و برگشت) تعیین می شود، و فرستنده برای رسیدن به حداکثر کارایی باید بتواند کانال را بدون وقفه پُر کند.

به این تکنیک لوله کشی (pipelining) گفته می شود. اگر ظرفیت کانال b bits/sec، اندازه فریم l بیت، و تأخیر رفت و برگشت R sec باشد، زمان لازم برای ارسال هر فریم l/b sec خواهد بود. بعد از ارسال آخرین بیت یک فریم، و قبل از رسیدن این بیت به گیرنده، تأخیری به میزان $R/2$ وجود دارد؛ بازگشت فریم تصدیق دریافت نیز با همین مقدار تأخیر همراه است (که کل تأخیر به R می رسد). با پروتکل توقف-انتظار، خط بمدت l/b کار کرده و سپس بمدت R بیکار می ماند، که در نتیجه

$$\text{بهره خط} = l / (l + bR)$$

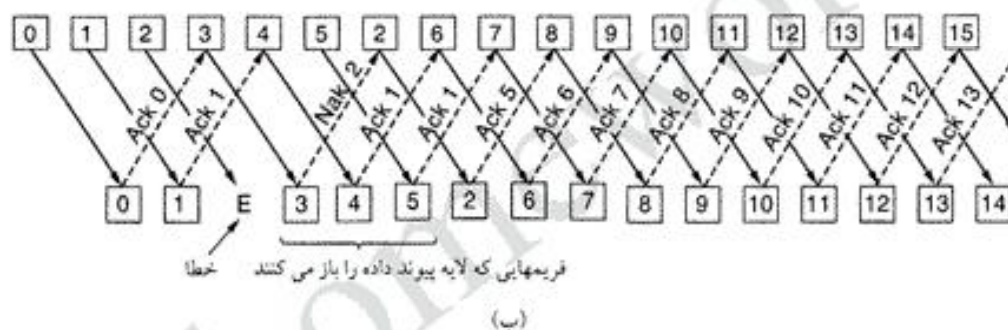
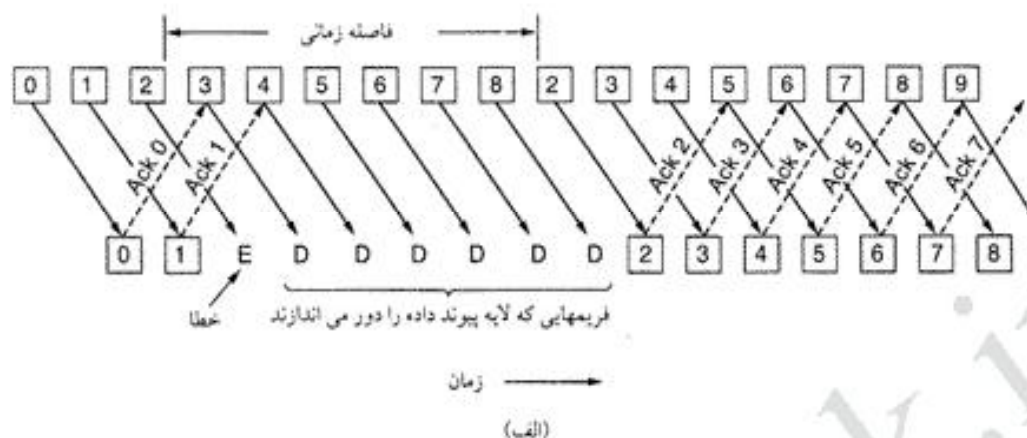
اگر $bR < l$ ، کارایی خط زیر 50% خواهد بود. از آنجائیکه تأخیر رفت و برگشت خطوط انتقال هرگز صفر نیست، با تکنیک لوله کشی می توان کاری کرد که خط همیشه مشغول باشد - اما اگر مقدار این تأخیر ناچیز باشد، ارزش پیچیدگی بیشتر پروتکل را ندارد.

لوله کشی فریمها در کانالهای غیر قابل اطمینان می تواند منجر به مشکلات جدی شود. اول اینکه، اگر یکی از فریمهای این صف طولی ناپدید یا خراب شود، چه خواهد شد؟ قبل از آنکه حتی فرستنده متوجه این خطا شود، تعداد زیادی از فریمهای بعدی به گیرنده رسیده اند. با رسیدن فریم خراب، گیرنده مسلماً آنرا دور می اندازد، اما با فریمهای سالم بعدی چه باید بکند؟ بیاد داشته باشید که لایه پیوند داده باید بسته ها را با ترتیب صحیح به لایه شبکه تحویل دهد. در شکل ۱۶-۳ این وضعیت (بروز خطا در خط لوله - pipeline) را ملاحظه می کنید. اجازه دهید آنرا دقیقتر بررسی کنیم.

برای مقابله با خطا در تکنیک لوله کشی دو رهیافت کلی وجود دارد. در رهیافت اول، که به N تابه عقب برگرد معروف است، گیرنده تمام فریمهای بعد از فریم خراب را دور می اندازد و هیچ تصدیقی برای آنها برنمی گرداند. این استراتژی معادل است با پنجره دریافتی باندازه 1. بعبارت دیگر، لایه پیوند داده در گیرنده هیچ فریمی غیر از آن فریمی که باید به لایه شبکه تحویل دهد، را قبول نمی کند. اگر پنجره ارسال فرستنده قبل از انقضای تایمر پُر شود، خط لوله شروع به خالی شدن خواهد کرد. پس از آن تایمر فرستنده به انتها رسیده و تمام فریمهای باقی مانده (از فریمی که خراب یا گم شده) را دوباره ارسال خواهد کرد. اگر نرخ خطا در خط زیاد باشد (مانند کانالهای بیسیم)، این رهیافت باعث اتلاف شدید پهنای باند خواهد شد.

در شکل ۱۶-۳ (الف) این حالت را ملاحظه می کنید. در این مثال فریمهای 0 و 1 سالم به مقصد رسیده اند، ولی فریم 2 خراب شده است. فرستنده تا زمانی که تایمر این فریم منقضی نشود، از خراب شدن آن مطلع نخواهد شد. پس از آنکه فرستنده فهمید فریم 2 سالم به مقصد نرسیده، برمی گردد و ارسال فریمها را از این فریم از سر می گیرد.

رهیافت دوم مقابله با خطا در تکنیک لوله کشی تکرار انتخابی (selective repeat) نام دارد. در این رهیافت، فریم خراب در گیرنده دور انداخته شده، ولی فریمهای سالم بعدی بافر می شوند. وقتی تایمر فریم معیوب منقضی



شکل ۳-۱۶. مقابله با خطا در خط لوله. تأثیر خطا وقتی که (الف) اندازه پنجره دریافت گیرنده ۱ است، و (ب) پنجره دریافت بزرگ است.

شد، فرستنده فقط همین فریم (که قدیمی ترین فریم تصدیق نشده است) را مجدداً ارسال می کند. اگر این فریم سالم به مقصد رسید، گیرنده این فریم و فریمهای بافر شده را بترتیب به لایه شبکه تحویل می دهد. در رهیافت تکرار انتخابی، معمولاً گیرنده برای فریمهای خراب (فریمهایی که جمع تطبیقی آنها اشتباه است، یا خارج از نظم وارد می شوند) نیز یک فریم تصدیق دریافت منفی (negative acknowledgement) - که به فریم NAK معروف است - به فرستنده برمی گرداند. فریم NAK کارایی سیستم را به مقدار زیادی بهبود می بخشد، چون باعث می شود که فرستنده قبل از انقضای تایمر کار ارسال مجدد فریمهای از دست رفته را شروع کند.

در شکل ۳-۱۶ (ب) نیز فریمهای ۰ و ۱ سالم به مقصد رسیده اند، ولی فریم ۲ خراب شده است. وقتی فریم ۳ به گیرنده می رسد، لایه پیوند داده متوجه می شود که یک فریم جا افتاده است، پس یک NAK برای فریم ۲ به فرستنده فرستاده، و فریم ۳ را در بافری که برای این منظور اختصاص یافته، ذخیره می کند. فریمهای ۴ و ۵ نیز پس از رسیدن به گیرنده، بجای تحویل به لایه شبکه، بافر می شوند. با رسیدن NAK فریم ۲، فرستنده بلافاصله این فریم را ارسال می کند، که با رسیدن آن به مقصد، گیرنده می تواند فریم ۲ و فریمهای پس از آن (تا فریم ۵) را به لایه شبکه تحویل دهد (و تصدیق دریافت آنها را به فرستنده برگرداند). حتی اگر NAK فریم ۲ در راه گم شود و به دست فرستنده نرسد، باز هم پس از انقضای تایمر، فرستنده نسبت به ارسال مجدد آن (و فقط همین یک فریم) اقدام خواهد کرد؛ که البته در این میان فقط وقت بیشتری تلف خواهد شد. در واقع، NAK فقط ارسال مجدد

فریمهای معیوب را تسریع می کند.

رهیافت تکرار انتخابی خاص پنجره های دریافت بزرگتر از 1 است. هر فریمی در داخل پنجره دریافت قرار داشته باشد، می تواند بافر شود تا فریمهای قبل از آن دریافت و به لایه شبکه تحویل شوند. البته اگر پنجره دریافت خیلی بزرگ باشد، لایه پیوند داده به حافظه زیادی برای بافر کردن فریمها نیاز دارد.

این دو رهیافت داد و ستدی هستند بین پهنای باند و فضای بافر لایه پیوند داده، که بسته به اهمیت هر یک از این منابع می توان یکی از رهیافت های (N تا به عقب برگرد) یا «تکرار انتخابی» را انتخاب کرد. در شکل ۳-۱۷ پروتکل لوله کشی با پنجره دریافت 1 را ملاحظه می کنید؛ در این پروتکل تمام فریمهای بعد از یک فریم معیوب دور انداخته می شوند. در این پروتکل برای اولین بار فرض نامحدود بودن بسته هایی که لایه شبکه می تواند ارائه کند، را نیز کنار گذاشته ایم. در اینجا، وقتی لایه شبکه آماده ارسال یک بسته است، رویداد `network_layer_ready` را تحریک می کند. با این حال، برای آنکه هیچوقت بیش از `MAX_SEQ` فریم وارد صف ارسال لایه پیوند داده نشود، باید کاری کنیم که این لایه بتواند بطریقی مانع ارسال بسته های اضافی از لایه شبکه شود - برای این منظور از روالهای کتابخانه ای `enable_network_layer` و `disable_network_layer` استفاده کرده ایم.

```
/* Protocol 5 (go back n) allows multiple outstanding frames. The sender may transmit up
   to MAX_SEQ frames without waiting for an ack. In addition, unlike in the previous
   protocols, the network layer is not assumed to have a new packet all the time. Instead,
   the network layer causes a network_layer_ready event when there is a packet to send.
*/
```

```
#define MAX_SEQ 7 /* should be 2^n - 1 */
typedef enum {frame_arrival, cksum_err, timeout, network_layer_ready} event_type;
#include "protocol.h"
```

```
static boolean between(seq_nr a, seq_nr b, seq_nr c)
{
    /* Return true if a <= b < c circularly; false otherwise. */
    if (((a <= b) && (b < c)) || ((c < a) && (a <= b)) || ((b < c) && (c < a)))
        return(true);
    else
        return(false);
}
```

```
static void send_data(seq_nr frame_nr, seq_nr frame_expected, packet buffer[])
```

```
{
    /* Construct and send a data frame. */
    frame s; /* scratch variable */

    s.info = buffer[frame_nr]; /* insert packet into frame */
}
```



```

s.seq = frame_nr; /* insert sequence number into frame */
s.ack = (frame_expected + MAX_SEQ) % (MAX_SEQ + 1); /* piggyback ack */
to_physical_layer(&s); /* transmit the frame */
start_timer(frame_nr); /* start the timer running */
}

void protocol5(void)
{
    seq_nr next_frame_to_send; /* MAX_SEQ > 1; used for outbound
stream */
    seq_nr ack_expected; /* oldest frame as yet unacknowledged */
    seq_nr frame_expected; /* next frame expected on inbound
stream */
    frame r; /* scratch variable */
    packet buffer[MAX_SEQ + 1]; /* buffers for the outbound stream */
    seq_nr nbuffered; /* # output buffers currently in use */
    seq_nr i; /* used to index into the buffer array */
    event_type event;

    enable_network_layer(); /* allow network_layer_ready events */
    ack_expected = 0; /* next ack expected inbound */
    next_frame_to_send = 0; /* next frame going out */
    frame_expected = 0; /* number of frame expected inbound */
    nbuffered = 0; /* initially no packets are buffered */
    while (true) {
        wait_for_event(&event); /* four possibilities: see event_type
above */

        switch(event) {
            case network_layer_ready: /* the network layer has a packet to send
*/

                /* Accept, save, and transmit a new frame. */
                from_network_layer(&buffer[next_frame_to_send]); /* fetch new packet */
                nbuffered = nbuffered + 1; /* expand the sender's window */
                send_data(next_frame_to_send, frame_expected, buffer); /* transmit the
frame */

                inc(next_frame_to_send); /* advance sender's upper window
edge */

                break;

            case frame_arrival: /* a data or control frame has arrived */

```

```

        from_physical_layer(&r);          /* get incoming frame from physical
layer */

        if (r.seq == frame_expected) {
            /* Frames are accepted only in order. */
            to_network_layer(&r.info);      /* pass packet to network layer */
            inc(frame_expected);            /* advance lower edge of receiver's
window */
        }

        /* Ack n implies n - 1, n - 2, etc. Check for this. */
        while (between(ack_expected, r.ack, next_frame_to_send)) {
            /* Handle piggybacked ack. */
            nbuffered = nbuffered - 1; /* one frame fewer buffered */
            stop_timer(ack_expected); /* frame arrived intact; stop timer */

            inc(ack_expected);          /* contract sender's window */
        }
        break;

        case cksum_err: break;          /* just ignore bad frames */

        case timeout:                   /* trouble; retransmit all outstanding
frames */
            next_frame_to_send = ack_expected; /* start retransmitting here */
            for (i = 1; i <= nbuffered; i++) {
                send_data(next_frame_to_send, frame_expected, buffer); /* resend
1 frame */

                inc(next_frame_to_send); /* prepare to send the next one */
            }

        }

        if (nbuffered < MAX_SEQ)
            ;(reyal_krowten_elbane
        else
            disable_network_layer();
    }
}

```


توجه کنید که در هر لحظه نباید بیش از MAX_SEQ فریم (و نه $MAX_SEQ + 1$) در صف ارسال وجود داشته باشد - حتی با وجود اینکه تعداد شماره‌های ترتیبی، در $MAX_SEQ + 1$ است: $0, 1, 2, \dots, MAX_SEQ$ برای پی بردن به علت این محدودیت، سناریوی زیر را که در آن $MAX_SEQ = 7$ نظر بگیرید:

۱. فرستنده فریمهای 0 تا 7 (هشت فریم) را ارسال می‌کند.
۲. پس از مدتی، تصدیق دریافت فریم 7 (با استفاده از تکنیک سواری مجانی) به فرستنده برمی‌گردد.
۳. فرستنده 8 فریم بعدی را با همان شماره‌های ترتیبی 0 تا 7 ارسال می‌کند.
۴. حال فریم تصدیق دریافت دیگری برای فریم شماره 7 (به همان صورت سواری مجانی) از راه می‌رسد.

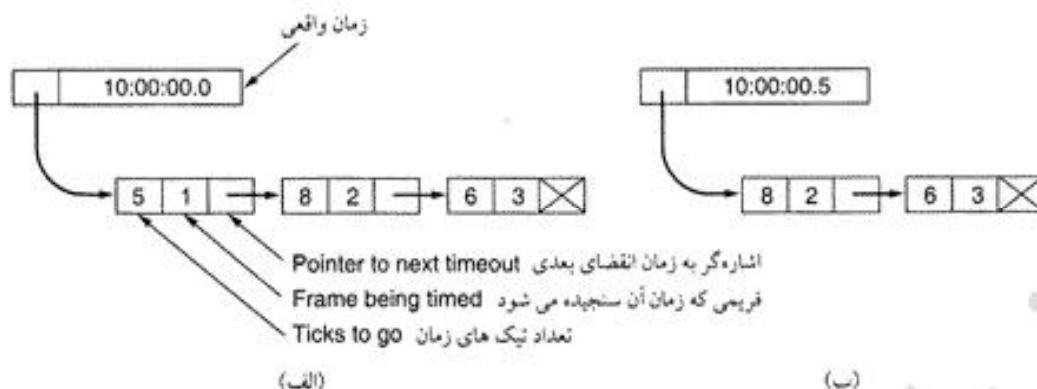
سؤال این است: آیا تمام 8 فریمی که متعلق به دسته دوم بودند، سالم به مقصد رسیده‌اند، یا (با توجه به اینکه تمام فریمهای پس از یک فریم خراب دور انداخته می‌شوند) همگی از بین رفته‌اند؟ اگر دقت کرده باشید، در هر دو حالت گیرنده تصدیق دریافت فریم 7 را پس می‌فرستد، و فرستنده هم راهی برای تشخیص این موضوع ندارد. به همین دلیل حداکثر فریمهایی که در صف ارسال می‌ایستند، باید MAX_SEQ باشد.

با اینکه پروتکل 5 فریمهای پس از یک فریم معیوب را بافر نمی‌کند، اما هنوز به مقداری بافر در سمت فرستنده نیاز داریم. از آنجائیکه فرستنده باید تمام فریمهای داخل پنجره ارسال را تا رسیدن تصدیق دریافت آنها نگه دارد (چون ممکنست لازم شود آنها را دوباره بفرستد)، باید فضای کافی برای بافر کردن این فریمها داشته باشد. در این پروتکل وقتی تصدیق دریافت فریم n از راه برسد، فریمهای $n-1$ ، $n-2$ و ... نیز بطور خودکار تصدیق شده محسوب می‌شوند. این ویژگی بویژه اگر فریمهای تصدیق قبلی در راه برگشت به فرستنده از بین بروند، اهمیت می‌یابد. با رسیدن هر فریم تصدیق دریافت، لایه پیوند داده چک می‌کند که کدام بافرها را می‌تواند آزاد کند. وقتی بافر آزاد (و جایی در پنجره ارسال باز) شد، لایه شبکه (که قبلاً متوقف شده) دوباره با رویداد `enable_network_layer` فعال می‌شود و می‌تواند بسته‌های بعدی را به لایه پیوند داده بفرستد.

در پروتکل 5 فرض کرده‌ایم که همیشه ترافیک برگشتی کافی برای سواری مجانی وجود دارد. اگر چنین نباشد، فریمهای تصدیق دریافت را هم نمی‌توان ارسال کرد. در پروتکل 4 چنین فرضی وجود نداشت، و برای هر فریم یک فریم تصدیق دریافت مستقل پس فرستاده می‌شد. در پروتکل آینده این مشکل را هم بنحو جالبی حل خواهیم کرد.

دیدید که در پروتکل 5، فرستنده می‌تواند در هر لحظه تعداد زیادی فریم ارسال شده (ولی هنوز تصدیق نشده) در بافر خود داشته باشد: هر یک از این فریمها یک تایمر مستقل می‌خواهند. این تایمرها را می‌توان بصورت نرم‌افزاری و با استفاده از وقفه‌های ساعت سخت‌افزاری، ایجاد کرد. این تایمرها تشکیل یک لیست پیوندی (linked list) می‌دهند، که هر گره آن سه بخش دارد: زمان باقی مانده تایمر، فریمی که به این تایمر مربوط است، و یک اشاره گر به گره بعدی.

در شکل 3-18 (الف) طرز پیاده‌سازی این تایمرها را می‌بینید. فرض کنید که ساعت سیستم هر 100 msec یک تیک (وقفه سخت‌افزاری) می‌فرستد. در لحظه شروع، که زمان واقعی 10:00:00.0 است، سه تایمر برای زمانهای 10:00:00.5، 10:00:01.3 و 10:00:01.9 ست می‌شوند. با هر تیک ساعت سخت‌افزاری، شمارنده تایمری که در رأس لیست قرار دارد، کاهش می‌یابد. وقتی این شمارنده 0 شد، گره مربوطه از لیست حذف می‌شود؛ شکل 3-18 (ب) را ببینید. سازماندهی تایمرها به صورت فوق باعث می‌شود که در هر بار فراخوانی روالهای `start_timer` و `stop_timer` کل لیست اسکن شود، ولی از سوی دیگر کار لازم برای به روز کردن تایمرها در هر تیک بسیار ناچیز است. همانطور که می‌بینید، در پروتکل 5 روالهای `start_timer` و `stop_timer` پارامتری می‌گیرند، که نشان می‌دهد زمان کدام فریم بایستی سنجیده شود.



شکل ۳-۱۸. شبیه سازی چند تایمر بوسیله نرم افزار.

۳-۴-۳ پروتکل تکرار انتخابی

اگر نرخ خطا ناچیز باشد، پروتکل ۵ بخوبی کار خواهد کرد، اما در خطوط پرنویز این پروتکل مقدار زیادی از پهنای باند را (برای ارسال مجدد فریمها) به هدر می دهد. استراتژی دیگر مقابله با خطاهای خط اینست که به گیرنده اجازه دهیم فریمهای سالمی که بعد از یک (یا چند) فریم معیوب از راه می رسند، را بافر کند. چنین پروتکلی دیگر فریمها را به صرف اینکه فریم قبلی آنها خراب بوده یا گم شده، دور نمی اندازد. در این پروتکل، فرستنده و گیرنده هر دو پنجره ای از شماره های قابل قبول دارند. پنجره ارسال در فرستنده از 0 شروع شده، و می تواند تا MAX_SEQ بالا برود؛ اما اندازه پنجره دریافت همیشه ثابت، و معادل MAX_SEQ است. گیرنده برای هر یک از فریمهایی که در پنجره دریافت قرار می گیرند، یک بافر ثابت دارد، که هر یک از این بافرها دارای بیتی هستند (بنام بیت *arrived*) که مشخص می کند بافر پر است یا خالی. وقتی یک فریم بدست گیرنده می رسد، شماره آن را با تابع *between* چک می کند تا ببیند در داخل پنجره دریافت است یا خارج آن. اگر فریم داخل پنجره باشد و تکراری هم نباشد، پذیرفته شده و در بافر ذخیره می شود (خواه این فریم محتوی بسته ای باشد که باید در نوبت بعد به لایه شبکه داده شود، یا نه). البته این فریم تا زمانی که تمام فریمهای قبل از آن به لایه شبکه تحویل نشده اند، در لایه پیوند داده می ماند. در شکل ۳-۱۹ پروتکلی که بر اساس این الگوریتم نوشته شده، را ملاحظه می کنید.

/* Protocol 6 (selective repeat) accepts frames out of order but passes packets to the network layer in order. Associated with each outstanding frame is a timer. When the timer

expires, only that frame is retransmitted, not all the outstanding frames, as in protocol 5. */

```
#define MAX_SEQ 7 /* should be 2^n - 1 */
#define NR_BUFS ((MAX_SEQ + 1)/2)
typedef enum {frame_arrival, cksum_err, timeout, network_layer_ready, ack_timeout}
event_type;
#include "protocol.h"
boolean no_nak = true; /* no nak has been sent yet */
```



```

seq_nr oldest_frame = MAX_SEQ + 1;      /* initial value is only for the simulator
*/
static boolean between(seq_nr a, seq_nr b, seq_nr c)
{
    /* Same as between in protocol5, but shorter and more obscure. */
    return ((a <= b) && (b < c)) || ((c < a) && (a <= b)) || ((b < c) && (c < a));
}
static void send_frame(frame_kind fk, seq_nr frame_nr, seq_nr frame_expected,
packet buffer[])
{
    /* Construct and send a data, ack, or nak frame. */
    frame s;                          /* scratch variable */

    s.kind = fk;                      /* kind == data, ack, or nak */
    if (fk == data) s.info = buffer[frame_nr % NR_BUFS];
    s.seq = frame_nr;                 /* only meaningful for data frames */
    s.ack = (frame_expected + MAX_SEQ) % (MAX_SEQ + 1);
    if (fk == nak) no_nak = false;    /* one nak per frame, please */
    to_physical_layer(&s);            /* transmit the frame */
    if (fk == data) start_timer(frame_nr % NR_BUFS);
    stop_ack_timer();                 /* no need for separate ack frame */
}
void protocol6(void)
{
    seq_nr ack_expected;              /* lower edge of sender's window */
    seq_nr next_frame_to_send;        /* upper edge of sender's window +
1 */
    seq_nr frame_expected;            /* lower edge of receiver's window */
    seq_nr too_far;                   /* upper edge of receiver's window + 1
*/

    int i;                            /* index into buffer pool */
    frame r;                          /* scratch variable */
    packet out_buf[NR_BUFS];          /* buffers for the outbound stream */
    packet in_buf[NR_BUFS];           /* buffers for the inbound stream */
    boolean arrived[NR_BUFS];         /* inbound bit map */
    seq_nr nbuffered;                 /* how many output buffers currently
used */
    event_type event;

    enable_network_layer();           /* initialize */
    ack_expected = 0;                 /* next ack expected on the inbound

```

```

stream */
    next_frame_to_send = 0;                /* number of next outgoing frame */
    frame_expected = 0;
    too_far = NR_BUFS;
    nbuffered = 0;                        /* initially no packets are buffered */
    for (i = 0; i < NR_BUFS; i++) arrived[i] = false;
    while (true) {
        wait_for_event(&event);            /* five possibilities: see event_type
above */
        switch(event) {
            case network_layer_ready:        /* accept, save, and transmit a new
frame */
                nbuffered = nbuffered + 1;    /* expand the window */
                from_network_layer(&out_buf[next_frame_to_send % NR_BUFS]); /* fetch
new packet */
                send_frame(data, next_frame_to_send, frame_expected, out_buf); /*
transmit the frame */
                inc(next_frame_to_send);      /* advance upper window edge */
                break;

            case frame_arrival:             /* a data or control frame has arrived
*/
                from_physical_layer(&r);      /* fetch incoming frame from
physical layer */
                if (r.kind == data) {
                    /* An undamaged frame has arrived. */
                    if ((r.seq != frame_expected) && no_nak)
                        send_frame(nak, 0, frame_expected, out_buf); else
                            start_ack_timer();
                    if (between(frame_expected, r.seq, too_far) &&
(arrived[r.seq%NR_BUFS] == false)) {
                        /* Frames may be accepted in any order. */
                        arrived[r.seq % NR_BUFS] = true; /* mark buffer as
full */
                        in_buf[r.seq % NR_BUFS] = r.info; /* insert data into
buffer */

                        while (arrived[frame_expected % NR_BUFS]) {
                            /* Pass frames and advance window. */
                            to_network_layer(&in_buf[frame_expected % NR_BUFS]);
                            no_nak = true;
                            arrived[frame_expected % NR_BUFS] = false;

```



```

    inc(frame_expected); /* advance lower edge of receiver's
window */

    inc(too_far); /* advance upper edge of
receiver's window */

    start_ack_timer(); /* to see if a separate ack is
needed */

}

}

if((r.kind==nak) &&
between(ack_expected,(r.ack+1)%(MAX_SEQ+1),next_frame_to_send))
    send_frame(data, (r.ack+1) % (MAX_SEQ + 1), frame_expected,
out_buf);

while (between(ack_expected, r.ack, next_frame_to_send)) {
    nbuffered = nbuffered - 1; /* handle piggybacked ack */
    stop_timer(ack_expected % NR_BUFS); /* frame arrived intact
*/

    inc(ack_expected); /* advance lower edge of sender's
window */

}
break;

case cksum_err:
    if (no_nak) send_frame(nak, 0, frame_expected, out_buf); /* damaged
frame */
    break;

case timeout:
    send_frame(data, oldest_frame, frame_expected, out_buf); /* we timed
out */
    break;

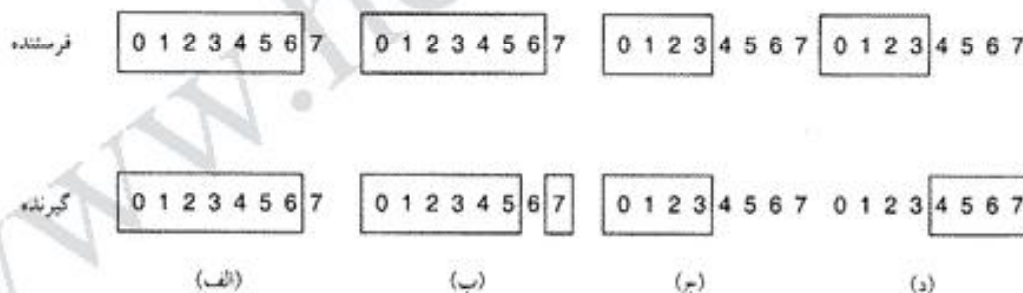
case ack_timeout:
    send_frame(ack,0,frame_expected, out_buf); /* ack timer expired;
send ack */
}
):(reyal_krowten_elbasid esle :(reyal_krowten_elbane )SFUB_RN < dereffubn( fi
}
}
}

```

دریافت نامنظم فریمها مسائلی را بوجود می آورد که در پروتکل های قبلی (که فریمها را فقط بترتیب شماره قبول می کنند) وجود نداشت. با یک مثال بهتر می توان این مشکل را نشان داد. فرض کنید از شماره های ترتیبی سه بیتی استفاده می کنیم، بنابراین فرستنده قبل از توقف برای رسیدن اولین فریم تصدیق دریافت می تواند حداکثر هفت فریم ارسال کند. در لحظه شروع، پنجره های ارسال و دریافت مانند شکل ۳-۲۰ (الف) هستند. فرستنده فریمهای 0 تا 6 را می فرستد. پنجره دریافت گیرنده فقط اجازه پذیرش فریمهایی را می دهد که شماره آنها (منحصراً) بین 0 تا 6 باشد. تمام هفت فریم اول سالم به مقصد می رسند، بنابراین گیرنده دریافت آنها را تصدیق کرده، پنجره دریافت را برای دریافت سری بعدی فریمها (0، 1، 2، 3، 4، 5) بجلو می برد، و تمام بافرها را هم با علامت «خالی» نشانه گذاری می کند؛ شکل ۳-۲۰ (ب) را ببینید.

درست در همین لحظه یک صاعقه به خط تلفن اصابت کرده، و تمام فریمهای تصدیق دریافت را (در راه بازگشت به فرستنده) از بین می برد. پس از مدتی تایمر فریم 0 به انتها رسیده، و فرستنده این فریم را مجدداً ارسال می کند. وقتی این فریم تکراری به گیرنده رسید، گیرنده چک می کند که آیا در داخل پنجره دریافت هست یا خیر. متأسفانه همانطور که در شکل ۳-۲۰ (ب) می بینید، فریم 0 هنوز در داخل پنجره دریافت قرار دارد، بنابراین گیرنده آنرا قبول می کند. از آنجائیکه فریمهای 0 تا 6 دریافت شده اند، گیرنده تصدیق دریافت فریم 6 را سوار فریم بعدی کرده و به فرستنده برمی گرداند.

فرستنده هم خوشحال از اینکه تمام فریمهای فرستاده شده سالم به مقصد رسیده اند، پنجره ارسال را بجلو برده و بلافاصله فریمهای 0، 1، 2، 3، 4 و 5 را می فرستد. وقتی فریم 7 به مقصد رسید، لایه پیوند داده گیرنده آنرا تحویل لایه شبکه می دهد. بلافاصله پس از آن لایه پیوند داده چک می کند که آیا بسته 0 معتبری وجود دارد یا خیر، و چون چنین بسته ای در بافرهای آن موجود است، آنرا به لایه شبکه می دهد؛ در حالیکه می دانیم این همان بسته 0 اول است و نباید دوباره به لایه شبکه داده شود - پس پروتکل ما مرتکب خطا شده است.



شکل ۳-۲۰. (الف) پنجره های ارسال و دریافت هفت تایی در لحظه شروع. (ب) بعد از رسیدن هفت فریم به مقصد، و قبل از بازگشت تصدیق دریافت به فرستنده. (ج) پنجره های ارسال و دریافت چهار تایی در لحظه شروع. (د) بعد از رسیدن چهار فریم به مقصد، و قبل از بازگشت تصدیق دریافت به فرستنده.

منشأ این مشکل آنجاست که بعد از جلو رفتن پنجره دریافت در گیرنده، شماره های معتبر جدید با شماره های قدیمی همپوشانی (overlap) دارد. در نتیجه، فریمهای بعدی می توانند تکراری باشند (اگر تمام فریمهای تصدیق دریافت از بین برود) یا نباشند (اگر تمام فریمهای تصدیق دریافت سالم به فرستنده برسند). گیرنده بیچاره هیچ راهی برای تشخیص این وضعیت ندارد.

راه چاره این معضل آن است که مطمئن شویم بعد از جلو رفتن پنجره گیرنده، با پنجره اصلی همپوشانی نداشته باشد. برای رسیدن به این هدف پنجره دریافت باید از نصف تعداد شماره های ترتیبی تجاوز نکند؛ شکل

۳-۲۰ (پ) و (ت) را ببینید. برای مثال، اگر از شماره‌های ترتیبی ۴-بیتی استفاده کنیم، محدوده ما 0 تا 15 خواهد بود، و فرستنده نباید در هر لحظه بیش از هشت فریم تصدیق نشده در بافر خود داشته باشد. بدین ترتیب، وقتی گیرنده فریمهای 0 تا 7 را گرفته و پنجره دریافت را جلو ببرد، به سری 8 تا 15 می‌رسد که آشکار است دیگر با مشکل قبلی مواجه نخواهد شد. در حالت کلی، اندازه پنجره در پروتکل ۶-بیتی حداکثر $(MAX_SEQ + 1)/2$ باشد. در مثال قبل، پنجره ارسال و دریافت را باید $(7 + 1)/2 = 4$ انتخاب کنیم.

و حالا یک سؤال جالب دیگر: گیرنده چند بافر باید داشته باشد؟ گیرنده در هیچ شرایطی فریمی که شماره آن کمتر از لبه پایین پنجره دریافت یا بیشتر از لبه بالای آن باشد، را قبول نخواهد کرد. در نتیجه، تعداد بافرهای گیرنده باید معادل اندازه پنجره دریافت باشد (نه تعداد شماره‌های ترتیبی). با شماره‌های ترتیبی ۴-بیتی، گیرنده به حداکثر هشت بافر نیاز دارد. وقتی فریم i بدست گیرنده می‌رسد، آنرا در بافری بشماره $i \text{ MOD } 8$ قرار می‌دهد. شاید حدس زده باشید که فریمهای i و $i + 8$ هر دو در یک بافر قرار می‌گیرند، ولی توجه کنید که این دو فریم هرگز در یک پنجره واقع نمی‌شوند (چون برای آنکه چنین اتفاقی بیفتد، اندازه پنجره باید حداقل 9 باشد).

به دلیل مشابه، تعداد تایمرهای فرستنده نیز باید معادل پنجره دریافت باشد، نه تعداد شماره‌های ترتیبی (چون هر تایمر به یک بافر وابسته است و وقتی تایمر به انتها می‌رسد، بافر دوباره ارسال می‌شود).

در پروتکل ۵ این پیش‌فرض ضمنی را داشتیم که بار کانال بسیار زیاد است: وقتی یک فریم از راه می‌رسد، تصدیق آن بلافاصله برگردانده نمی‌شود تا سوار فریم بعدی (که از گیرنده به فرستنده می‌رود) شود. اما اگر ترافیک در جهت مخالف کم باشد، فریمهای تصدیق دریافت خیلی معطل خواهند شد. در این روش، وقتی فرستنده MAX_SEQ بسته فرستاد، منتظر می‌ماند و از آنجائیکه ترافیک جهت مقابل کم است، مدت زیادی بیکار خواهد ماند؛ علت فرض زیاد بودن بار کانال نیز همین موضوع است.

در پروتکل ۶ این مشکل نیز برطرف شده است. وقتی یک فریم (که طبق ترتیب مورد انتظار فرستاده شده) از راه رسید، گیرنده یک تایمر کمکی را (با تابع $start_ack_timer$) راه می‌اندازد. اگر در مدتی که این تایمر منقضی می‌شود، فریمی برای ارسال تحویل لایه پیوند داده نشد، یک فریم تصدیق دریافت مستقل برگردانده خواهد شد (رویداد این تایمر $ack_timeout$ نام دارد). با این تمهید دیگر نیازی نیست متکی به ترافیک سنگین دوطرفه باشیم، و پروتکل ۶ حتی می‌تواند بصورت کاملاً یکطرفه هم کار کند. از این تایمر کمکی فقط یکی وجود دارد، و اجرای تابع $start_ack_timer$ آنرا ریست می‌کند. البته ضروری است که فاصله زمانی تایمر کمکی بسیار کوتاهتر از فاصله زمانی تایمرهای فریمهای داده باشد، تا این تایمرها قبل از رسیدن تصدیق دریافت منقضی نشوند.

استراتژی مقابله با خطا در پروتکل ۶ بسیار کارآمدتر از پروتکل ۵ است. اگر گیرنده به هر دلیلی ظن خطا ببرد، یک فریم تصدیق دریافت منفی (NAK) به فرستنده پس می‌فرستد. این NAK صریحاً از فرستنده می‌خواهد که فریم مشخص شده را دوباره ارسال کند. دو حالت وجود دارد که گیرنده باید به بروز خطا مشکوک شود: دریافت یک فریم معیوب، یا دریافت فریمی که انتظار آنرا ندارد. برای اجتناب از تکرار این درخواستها، گیرنده لیستی از فریمهایی که برای آنها NAK فرستاده را نگه می‌دارد. اگر متغیر no_nak در پروتکل ۶ مقدار $true$ داشته باشد، نشان می‌دهد که هنوز برای فریم $frame_expected$ هیچ NAK فرستاده نشده است. خراب یا گم شدن NAK ها مشکلی بوجود نخواهد آورد، فقط زمان ارسال مجدد فریمها را کمی به تأخیر می‌اندازد (چون تایمر فرستنده بهر حال منقضی خواهد شد). اگر پس از NAK فریم خواسته شده از راه برسد، گیرنده مقدار no_nak را به $true$ ست کرده و تایمر کمکی را راه می‌اندازد ($start_ack_timer$). پس از انقضای این تایمر، گیرنده یک فریم ACK به فرستنده پس می‌فرستد، و بدین ترتیب خود را با آن سنکرون می‌کند.

در برخی شرایط، زمان لازم برای سفر فریمهای داده به مقصد، پردازش در آنجا، و بازگشت فریم تصدیق

دریافت (تقریباً) ثابت است. در چنین شرایطی فرستنده می‌تواند تایمرهای خود را کمی بالاتر از این مقدار تنظیم کند. اما اگر این زمان در حد وسیعی متغیر باشد، فرستنده دو راه در پیش رو دارد: فاصله زمانی تایمرهای خود را خیلی کوچک بگیرد (و ریسک ارسالهای تکراری را بپذیرد)، یا آنرا بسیار بزرگ بگیرد (و بعد از هر خطا مدت زیادی بیکار بماند). هر دوی این گزینه‌ها تلف کردن پهنای باند است.

اگر ترافیک جهت مخالف کم باشد، زمان برگشت فریمهای تصدیق دریافت نیز نامنظم خواهد بود (گاهی کم و گاهی زیاد). تغییر زمان پردازش فریم در گیرنده هم می‌تواند مزید علت باشد. در کل، اگر انحراف معیار فاصله زمانی فریم تصدیق دریافت (در مقایسه با کل فاصله زمانی) کوچک باشد، می‌توان فاصله زمانی تایمرها را «کوچک» در نظر گرفت، که در این حالت NAK سودمندی خود را از دست می‌دهد. در غیر اینصورت، برای اجتناب از تکرار در ارسال فریمها باید فاصله زمانی تایمرها را «بزرگ» گرفت؛ در این حالت استفاده از NAK می‌تواند ارسال مجدد فریمهای معیوب و گم شده را تسریع کند.

سؤال دیگری که در همین زمینه پیش می‌آید این است که: کدام فریم باعث انقضای تایمر شده است؟ پروتکل 5 همیشه *ack_expected* است (یعنی فقط انتظار دریافت تصدیق را دارد)، چون همیشه قدیمی‌ترین باعث انقضای تایمر می‌شود. اما در پروتکل 6 راه ساده‌ای برای تعیین این موضوع وجود ندارد. فرض کنید فریمهای 0 تا 4 فرستاده شده‌اند، و لیست فریمهای بافر شده در فرستنده از قدیم‌ترین به جدیدترین (از چپ بر راست) عبارتند از: 01234. حال وضعیت زیر را در نظر بگیرید: فریم 0 منقضی می‌شود، فرستنده فریم (جدید) 5 را می‌فرستد، فریم 1 منقضی می‌شود، و فرستنده فریم (جدید) 6 را می‌فرستد. در این لحظه، بافر فرستنده حاوی فریمهای 3405126 (از قدیم به جدید) است. اگر در این لحظه تمام فریمهای برگشتی (که فرض می‌کنیم حامل فریمهای تصدیق دریافت هستند) از بین بروند، هفت فریمی که در بافر قرار دارند، به همین ترتیب منقضی می‌شوند. برای اجتناب از پیچیدگی بیشتر (که تا همین جا هم باز اندازه کافی پیچیده هست)، به مدیریت تایمرها نپرداختیم. بجای آن فرض کردیم که متغیر *oldest_frame* در لحظه انقضای تایمر نشان می‌دهد که کدام فریم منقضی شده است.

۵-۳ ارزیابی پروتکل‌ها

پروتکل‌های واقعی (و کُد نرم‌افزاری آنها) اغلب بسیار پیچیده‌اند، به همین دلیل تحقیقات زیادی انجام شده تا روشهای ریاضی و مشخصی برای ارزیابی آنها ابداع شود. در این قسمت برخی از این تکنیکها و مدلها را بررسی خواهیم کرد. با اینکه در اینجا برای ارزیابی پروتکل‌های لایه پیوند داده از این تکنیکها استفاده کرده‌ایم، ولی آنها را می‌توان برای لایه‌های دیگر نیز بکار گرفت.

۱-۵-۳ مدل ماشین حالت محدود

یکی از مفاهیم کلیدی در مدلسازی پروتکلها، ماشین حالت محدود (finite state machine) است. در این تکنیک، هر ماشین پروتکل (protocol machine) - یعنی، فرستنده یا گیرنده - در هر لحظه از زمان در حالتی خاص قرار دارد. این حالت (state) عبارتست از مقدار تمام متغیرها، و شمارنده برنامه (program counter). در اکثر مواقع می‌توان تعداد زیادی از حالت‌ها را برای آنالیز دسته‌بندی کرد. برای مثال، گیرنده پروتکل 3 را در نظر بگیرید؛ تمام حالت‌های ممکن این گیرنده را می‌توان به دو دسته مهم تقسیم کرد: انتظار برای فریم 0، و انتظار برای فریم 1. تمام حالت‌های دیگر را می‌توان مراحل گذار از یکی از این دو حالت به حالت دیگر دانست. معمولاً حالت‌ها را بگونه‌ای انتخاب می‌کنند که در آن لحظه ماشین پروتکل در انتظار وقوع رویداد بعدی است (در مثال ما، اجرای روال *wait(event)*). در این لحظه، حالت پروتکل را می‌توان بطور کامل با دانستن مقدار متغیرهای آن

تعیین کرد. تعدادی حالت های چنین ماشینی 2^n است، که در آن n تعداد بیت های لازم برای نمایش تمام ترکیبات ممکنه متغیرهای آن است.

حالت کلی سیستم نیز عبارتست از ترکیب حالت های دو ماشین پروتکل (فرستنده و گیرنده) و حالت کانال. حالت کانال با محتویات آن تعیین می شود. اگر باز هم از پروتکل ۳ کمک بگیریم، کانال ما می تواند حالت های زیر را داشته باشد: فریم 0 یا 1 از فرستنده به گیرنده می رود، فریم تصدیق دریافت از گیرنده به فرستنده برمی گردد، و یا کانال خالیست. اگر فرض کنیم گیرنده و فرستنده نیز هر کدام فقط دو حالت دارند، کل سیستم دارای ۱۶ حالت مجزا خواهد بود.

همین جا باید نکته کوچکی را درباره کانال توضیح دهیم. وقتی می گوئیم «فریم در کانال است»، البته از یک مفهوم مجرد حرف می زنیم. آنچه واقعاً منظور ماست، اینست که «فریم احتمالاً به مقصد رسیده، ولی هنوز پردازش نشده است». این فریم تا زمانی که ماشین پروتکل روال *FromPhysicalLayer* را اجرا نکرده و آنرا پردازش نکند، «در کانال می ماند».

هر حالت دارای تعدادی گذار (transition) به حالت های دیگر است. گذار زمانی روی می دهد که رویدادی رخ دهد. در یک ماشین پروتکل، ارسال یک فریم، دریافت یک فریم، انقضای یک تایمر و مانند آن، همگی نمونه هایی از گذار هستند. رویدادهایی که می تواند در یک کانال رخ دهد، نیز عبارتند از: وارد شدن یک فریم جدید به کانال توسط ماشین پروتکل، برداشته شدن فریم از کانال توسط ماشین پروتکل، و یا گم شدن فریم در اثر نویز. با در دست داشتن توصیف کاملی از ماشین های پروتکل و مشخصات کانال، می توان سیستم را بصورت نموداری از گره ها (حالت ها) و خطوط متصل کننده (گذارها) نمایش داد.

یکی از حالت های مهم در هر سیستم، حالت اولیه (initial state) است. این حالت متناظر است با وضعیت سیستم در لحظه شروع به کار، یا (اگر مناسبتر باشد) کمی پس از آن. از این حالت اولیه می توان به کمک توالی گذارها به تمام (یا برخی از) حالت های دیگر رسید. با کمک تکنیک های نظریه گراف (graph theory) می توان مشخص کرد که کدام حالت ها قابل دسترسی اند، و کدامها نیستند. با این تکنیک، که به آنالیز دسترسی (reachability analysis) معروفست (Lin et al., 1987)، می توان دریافت که آیا یک پروتکل درست عمل می کند یا خیر.

مدل ماشین حالت محدود یک پروتکل را می توان بردار چهار عضوی (S, M, I, T) دانست، که در آن:

۱. S عبارتست از مجموعه حالت هایی که پروسس ها و کانال می توانند در آن باشند.
۲. M عبارتست از مجموعه فریم هایی که می توان روی کانال مبادله کرد.
۳. I عبارتست از مجموعه حالت های اولیه پروسس ها.
۴. T عبارتست از مجموعه گذارهای بین حالت ها.

در لحظه شروع، تمام پروسسها در حالت اولیه شان هستند. سپس رویدادها شروع به رخ دادن می کنند: فریمی برای ارسال آماده می شود، تایمرها خاموش می شوند، و مانند آن. هر رویداد می تواند باعث شود که یک پروسس یا کانال عملی را انجام داده و به حالت دیگر برود. با تعیین دقیق پیامدهای هر حالت، می توان گراف دسترسی را رسم و پروتکل را آنالیز کرد.

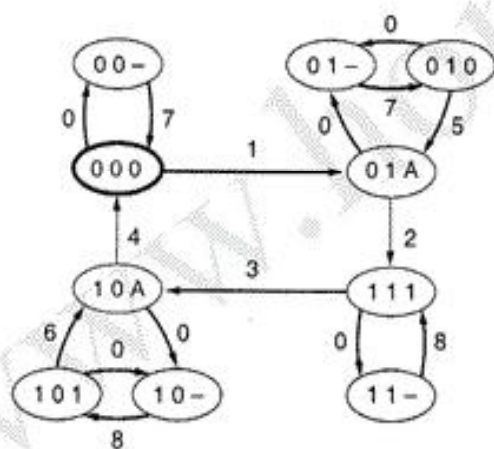
آنالیز دسترسی می تواند خطاهای مختلفی را در طراحی پروتکل روشن کند. برای مثال، اگر یک فریم خاص بتواند در حالت خاصی وجود داشته باشد و ماشین حالت محدود نتواند بگوید در این موقعیت چه باید کرد، طراحی ما مشکل دارد (ناقص است). اگر حالت یا حالت هایی وجود داشته باشد که نتوان از آن خارج شد (بعبارت دیگر، دریافت فریم سالم دیگر امکانپذیر نباشد)، باز هم پروتکل ما مشکل دارد (بن بست). موقعیت دیگری (که

در واقع مشکل چندان بزرگی نیست) آنست که پروتکل ما برای حالتی تجهیز شده باشد که امکان زوی دادن آنها وجود ندارد (گذارهای نامربوط)، گراف دسترسی خطاهای دیگر را هم می تواند کشف کند.

در شکل ۳-۲۱ (الف) نمونه ای از یک مدل ماشین حالت محدود را ملاحظه می کنید. این گراف معادل پروتکل ۳ است، که در بالا توضیح دادیم: هر ماشین پروتکل دارای دو حالت، و کانال دارای چهار حالت است. در کل ۱۶ حالت ممکن وجود دارد، که البته از حالت اولیه نمی توان به همه آنها رسید. در شکل این حالت های غیر قابل دسترسی را نشان نداده ایم، و برای سادگی کار از خطاهای جمع تطبیقی (checksum) هم چشم پوشیده ایم.

هر حالت با سه حرف SRC مشخص می شود، که در آن S یا 0 است یا 1 (متناظر با فریمی که فرستنده می خواهد بفرستد)؛ R نیز یا 0 است یا 1 (متناظر با فریمی که گیرنده منتظر دریافت آن است)؛ و C می تواند چهار مقدار 0 (فریم 0)، 1 (فریم 1)، A (فریم تصدیق دریافت) یا خالی (-) بگردد (متناظر با حالت های چهارگانه کانال). در مثال بالا، حالت اولیه با (000) نشان داده شده است: یعنی فرستنده فریم 0 را فرستاده، گیرنده منتظر دریافت فریم 0 است، و فریم 0 اکنون در کانال است.

در شکل ۳-۲۱ (ب) حالت گذار مختلف نشان داده شده است. در گذار 0 کانال محتویات خود را از دست می دهد. در گذار 1 کانال محتویات خود را به گیرنده تحویل می دهد، و گیرنده نیز حالت خود را به 1 (انتظار برای فریم 1) تغییر داده و یک فریم تصدیق دریافت به فرستنده پس می فرستد. گذار 1 همچنین متناظرست با تحویل بسته 0 به لایه شبکه در گیرنده. گذارهای دیگر را در شکل ۳-۲۱ (ب) ملاحظه می کنید. رسیدن فریمی با خطای جمع تطبیقی در اینجا نشان داده نشده، چون این اتفاق در پروتکل ۳ باعث تغییر حالت نمی شود.



(الف)

به لایه شبکه	فریم ارسال شده	فریم پذیرفته شده	نوبت چیست ؟	انتقال
-	-	(فریم گم شده)	-	0
Yes	A	0	R	1
-	1	A	S	2
Yes	A	1	R	3
-	0	A	S	4
No	A	0	R	5
No	A	1	R	6
-	0	(انقضای زمان)	S	7
-	1	(انقضای زمان)	S	8

(ب)

شکل ۳-۲۱. (الف) دیاگرام حالت پروتکل ۳. (ب) گذارها.

در حالت عادی، گذارهای 1، 2، 3 و 4 پشت سرهم و بارها و بارها تکرار می شوند. در هر سیکل دو بسته منتقل می شود، و با این کار فرستنده دوباره به حالت اولیه (ارسال فریم 0) برمی گردد. اگر فریم 0 در کانال از بین برود، سیستم از حالت (000) به حالت (00-) می رود (گذار 0). پس از مدتی تاخیر فرستنده به انتها رسیده (گذار 7)، و سیستم به حالت (000) باز می گردد. از بین رفتن فریم A (تصدیق دریافت) پیچیده تر است، و برای جبران آن به دو گذار نیاز داریم: 7 و 5، یا 6 و 8.

یکی از ویژگی هایی که یک پروتکل با شماره های ترتیبی 1-بیتی باید داشته باشد اینست که هرگز نباید دو فریم فرد متوالی (یا دو فریم زوج متوالی) به گیرنده برسد. در شکل ۳-۲۱ می توان این ویژگی را چنین نشان داد: هیچ

مسیری از حالت اولیه وجود ندارد که طی آن دو گذار متوالی ۱ رخ دهد، بدون اینکه بین آنها یک گذار ۳ وجود داشته باشد، و بالعکس. از شکل می توان دید که پروتکل ۳ این ویژگی را دارد.

الزام دیگر اینست که هیچ مسیری نباید وجود داشته باشد که طی آن فرستنده دو بار تغییر حالت دهد (از ۰ به ۱ و برگشت دوباره به ۰) در حالیکه گیرنده ثابت مانده است. اگر چنین مسیری وجود داشته باشد، بمعنای آن است که دو فریم از بین رفته بدون اینکه گیرنده متوجه شده باشد.

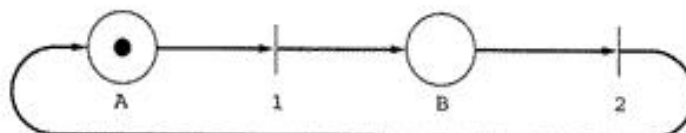
ویژگی مهمتر یک پروتکل عدم وجود بن بست (deadlock) در آن است. بن بست حالتی است که پروتکل (تحت هیچ شرایطی) دیگر قادر به جلو رفتن نباشد (یعنی نتواند بسته ها را لایه شبکه تحویل دهد). در مدل گراف، بن بست به زیرمجموعه ای از حالتها گفته می شود که بتوان از شرایط اولیه به آن رسید و دو ویژگی زیر را داشته باشد:

۱. هیچ گذاری برای خروج از این زیرمجموعه وجود نداشته باشد.
۲. هیچ گذاری در داخل زیرمجموعه وجود نداشته باشد که باعث پیشرفت کار شود.

وقتی یک پروتکل وارد بن بست شود، دیگر برای همیشه آنجا می ماند. باز هم از گراف شکل ۳-۲۱ می توان دید که (خوشبختانه) پروتکل ۳ هیچ بن بستنی ندارد.

۲-۵-۳ مدل شبکه پتری

ماشین حالت محدود تنها مدل برای ارزیابی پروتکلها نیست. در این قسمت به بررسی تکنیکی کاملاً متفاوت بنام شبکه پتری (Petri net) می پردازیم (Danthine, 1980). یک شبکه پتری دارای چهار عنصر اساسی است: مکان (place)، گذار (transition)، کمان (arc)، و نشانه (token). مکان حالتیست که سیستم (یا بخشی از آن) می تواند در آن باشد. در شکل ۳-۲۲ یک شبکه پتری را با دو مکان A و B می بینید، که با دایره مشخص شده اند. سیستم در حال حاضر در حالت A قرار دارد، که این موضوع با نشانه (نقطه سیاه) در مکان A مشخص شده است. گذار با یک خط افقی یا عمودی مشخص می شود. هر گذار می تواند دارای تعدادی کمان ورودی (input arc) - که از مکانهای ورودی آن می آیند و تعدادی کمان خروجی (output arc) - که به مکانهای خروجی آن می روند باشد. گذار فعال به گذاری گفته می شود که حداقل یک نشانه در یکی از ورودیهای آن وجود داشته باشد. گذار فعال می تواند هر گاه که اراده کند، آتش (fire) کرده و یک نشانه را از یکی از ورودیها برداشته و در یکی از خروجیهای خود قرار دهد. اگر تعداد کمانهای ورودی با کمانهای خروجی مساوی نباشد، نشانه ابقا (conserve) نخواهد شد.

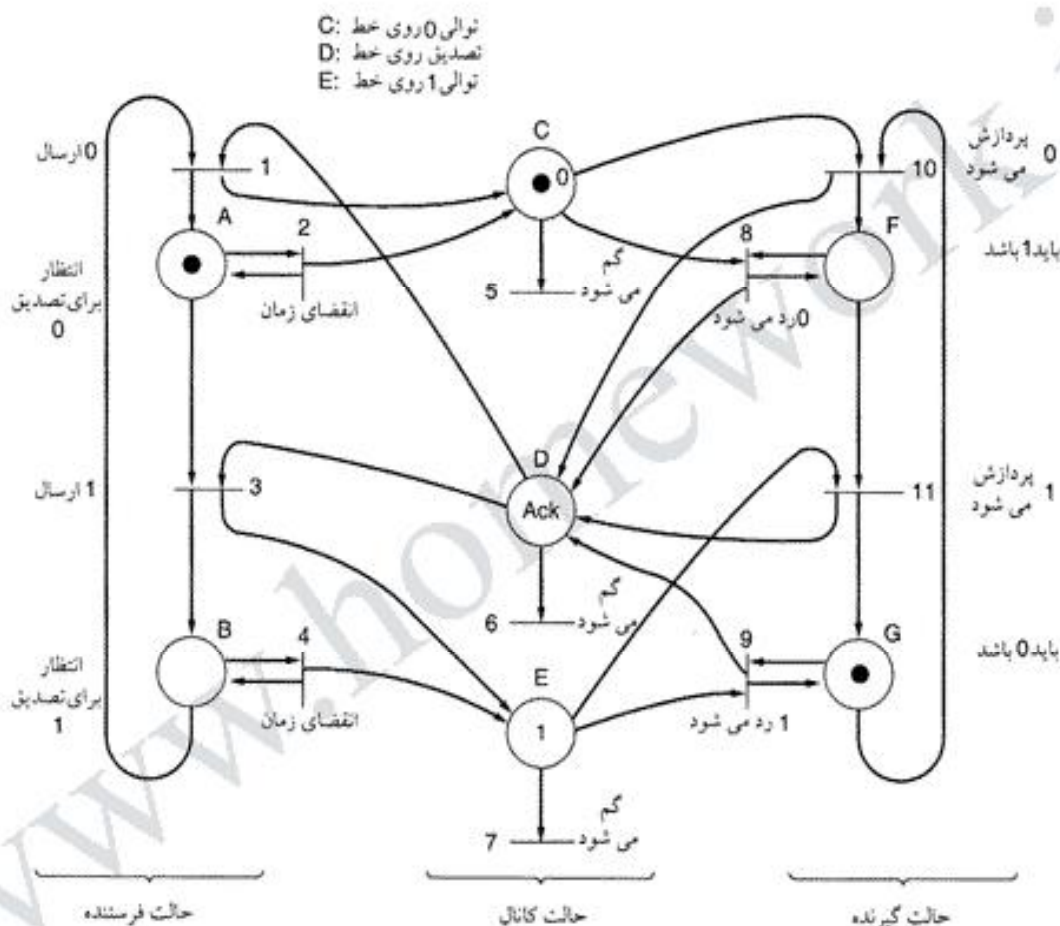


شکل ۳-۲۲. یک شبکه پتری با دو مکان و دو گذار.

اگر دو یا چند گذار فعال وجود داشته باشد، هر کدام از آنها می توانند آتش کنند. اینکه کدام گذار آتش می کند نامشخص است، و همین ویژگیست که شبکه پتری را برای مدلسازی پروتکل ها سودمند کرده است. شبکه پتری شکل ۳-۲۲ کاملاً مشخص است و از آن فقط برای مدلسازی پروسسهایی که دو فاز بیشتر ندارند، می توان استفاده کرد (مانند رفتار یک نوزاد: خوردن، خوابیدن، خوردن، خوابیدن، و الی آخر). در اینجا هم مانند سایر تکنیکهای مدلسازی جزئیات زائد حذف می شوند.

در شکل ۳-۲۳ مدل شبکه پتری شکل ۳-۱۲ (پروتکل ۳) را ملاحظه می کنید. بر خلاف مدل ماشین حالت

محدود، در اینجا حالت های ترکیبی وجود ندارد؛ حالت فرستنده، گیرنده و کانال بطور مجزا و مستقل نمایش داده می شوند. گذارهای ۱ و ۲ بترتیب عبارتند از ارسال فریم ۰ توسط فرستنده در حالت عادی، و بعد از انقضای تایمر. گذارهای ۳ و ۴ متناظر با ارسال فریم ۱ در این دو موقعیت هستند. گذارهای ۵، ۶ و ۷ نیز بترتیب از بین رفتن فریمهای ۰، ACK، ۱ را نشان می دهند. گذارهای ۸ و ۹ نشان می دهند که فریمی با شماره ترتیبی اشتباه (بترتیب ۰ و ۱) به گیرنده رسیده است. گذارهای ۱۰ و ۱۱ نیز بترتیب حاکی از رسیدن صحیح و سالم فریمهای ۰ و ۱ به گیرنده، و تحویل آنها به لایه شبکه هستند.



شکل ۳-۲۳. مدل شبکه پتری پروتکل ۳.

با شبکه پتری نیز می توان مانند ماشین حالت محدود مشکلات یک پروتکل را تشخیص داد. برای مثال، اگر یک توالی آتش وجود داشته باشد که در آن دو گذار ۱۰ (بدون یک گذار ۱۱ بین آنها) رخ دهد، پروتکل ما مشکل دارد. مفهوم بن بست در شبکه پتری نیز کاملاً شبیه همین مفهوم در ماشین حالت محدود است.

شبکه پتری را بصورت جبری نیز می توان نوشت: هر گذار یک جمله جبری است، که در یک دستور (که مکانهای ورودی و خروجی گذار را مشخص می کند) بکار می رود. از آنجائیکه شکل ۳-۲۳ دارای ۱۱ گذار است، برای نمایش جبری آن به ۱۱ دستور (که آنها را متناظر با گذارها شماره گذاری می کنیم) نیاز داریم. معادل جبری شبکه پتری شکل ۳-۲۳ چنین است:

- 1: BD → AC
- 2: A → A
- 3: AD → BE
- 4: B → B
- 5: C →
- 6: D →
- 7: E →
- 8: CF → DF
- 9: EG → DG
- 10: CG → DF
- 11: EF → DG

همانطور که می بینید، یک پروتکل نسبتاً پیچیده به ۱۱ جمله جبری ساده تبدیل شده که به آسانی می توان آنرا با کامپیوتر تحلیل کرد.

حالت فعلی شبکه پتری با مجموعه نامنظمی از مکانها (که هر مکان به تعداد نشانه هایی که دارد، ظاهر می شود) نشان داده می شود. در هر دستور، مکانهایی که سمت چپ جمله قرار دارند می توانند آتش کرده، و (بعد از حذف خود از حالت فعلی) خروجی خود را به حالت سیستم اضافه کنند. شکل ۳-۲۳ را می توان با علامت ACG نشان داد (یعنی، مکانهای A، C و G هر کدام یک نشانه دارند). در نتیجه، دستورات ۲، ۳ و ۱۰ فعال هستند، و هر کدام از آنها می توانند اجرا شوند و حالت سیستم را عوض کنند (که البته حالت قبلی هم جزء حالت های ممکنه است). توجه داشته باشید که، برای مثال، در این لحظه دستور ۳ (AD → BE) نمی تواند اجرا شود، چون D در علامت سیستم وجود ندارد.

۶-۳ چند نمونه از پروتکل های لینک داده

در این قسمت چند نمونه از پروتکل های لینک داده را که کاربرد وسیعی دارند، مورد بررسی قرار خواهیم داد. اولین آنها، HDLC، یک پروتکل بیت-گرا (bit-oriented) است که سالهاست در برنامه های بسیاری از ویرایش های مختلف آن استفاده می شود. دومی، PPP، یک پروتکل لینک داده است که برای اتصال کامپیوترهای خانگی به اینترنت بکار می رود.

۱۶-۳ HDLC - کنترل سطح بالای لینک داده

در این قسمت گروهی از پروتکل های نزدیک به هم را بررسی می کنیم که با وجود قدیمی بودن، همچنان کاربرد گسترده ای دارند. همه این پروتکلها از اولین پروتکل لینک داده که برای کامپیوترهای بزرگ IBM توسعه داده شد، مشتق شده اند: پروتکل SDLC (کنترل لینک داده سنکرون - Synchronous Data Link Control). بعد از توسعه این پروتکل، IBM آنرا برای پذیرش بعنوان استاندارد آمریکایی و بین المللی به ANSI و ISO فرستاد. ANSI و ISO هر کدام تغییراتی در این پروتکل دادند، و بترتیب پروتکل های ADCCP (روش پیشرفته کنترل مخابرات داده - Advanced Data Communication Control Procedure) و HDLC (کنترل سطح بالای لینک داده - High-Level Data Link Control) را از آن مشتق کردند. بعد از مدتی، CCITT تغییراتی در پروتکل HDLC داده، و پروتکل جدید را که LAP (روش دسترسی لینک - Link Access Procedure) نام گرفت، بعنوان قسمتی از استاندارد شبکه های X.25 معرفی کرد (این پروتکل بعدها برای سازگاری بهتر با ویرایش های جدید HDLC باز هم اصلاح و LAPB نامیده شد). خوبی استانداردها همین تنوع زیاد آنهاست، و بالاخره می توانید یکی را مطابق سلیقه تان پیدا کنید؛ اگر هم پیدا نکردید، زیاد جای نگرانی نیست: سال آینده

مدلهای جدیدتری به بازار خواهد آمد!

تمام این پروتکلها مبنای واحدی دارند: همه آنها بیت-گرا هستند، و از تکنیکهای لاگذاری بیت (bit stuffing) برای افزونگی داده ها استفاده می کنند. اختلاف آنها کوچک (ولی بهر حال، ناراحت کننده) است. برای اطلاع از مشخصات دقیق هر پروتکل می توانید به منابع مربوطه مراجعه کنید.

تمام پروتکلهای بیت-گرا از فریمهایی با ساختار شکل ۳-۲۴ استفاده می کنند. هر فریم با یک توالی پرچم (01111110) شروع می شود. فیلد آدرس (Address) در خطوطی اهمیت می یابد که ترمینالهای متعددی دارند، و از این فیلد برای مشخص کردن ترمینال مقصد استفاده می شود. در خطوط نقطه-به-نقطه (point-to-point) گاهی از این فیلد برای تشخیص فرمان (command) از پاسخ (response) استفاده می شود.

فیلد کنترل (Control) برای شماره ترتیبی فریم، تصدیق دریافت، و مقاصد دیگر بکار می رود (در ادامه این فیلد را بیشتر توضیح خواهیم داد).

Bits	8	8	8	≥ 0	16	8
بیت ها	0 1 1 1 1 1 0	Address (آدرس)	Control (کنترل)	Data (داده)	Checksum (جمع تطبیقی)	0 1 1 1 1 1 0

شکل ۳-۲۴. فرمت فریم در پروتکلهای بیت-گرا.

فیلد داده (Data) محتوی اطلاعاتیست که فریم باید منتقل کند. طول این فیلد می تواند هر اندازه ای باشد، اگر چه با زیاد شدن آن کارایی تکنیک جمع تطبیقی (بدلیل بالا رفتن احتمال بروز خطاهای فورانی) کاهش خواهد یافت.

فیلد جمع تطبیقی (Checksum) یک کد افزونگی چرخه ای (cyclic redundancy) است، که در قسمت ۲-۳ توضیح دادیم.

در انتها، فریم به یک توالی پرچم دیگر (01111110) ختم می شود. وقتی یک خط نقطه-به-نقطه بیکار است، بطور پیوسته توالیهای پرچم را ارسال می کند. هر فریم باید حداقل سه فیلد (مجموعاً ۳۲ بیت) داشته باشد (البته منهای پرچمهای ابتدا و انتها).

فریمها بر سه نوعند: اطلاعاتی (Information)، سرپرستی (Supervisory)، و بدون شماره (Unnumbered). در شکل ۳-۲۵ فیلد کنترل هر یک از این سه نوع فریم را ملاحظه می کنید. این پروتکل از تکنیک پنجره لغزنده، با شماره های ترتیبی ۳-بیتی، استفاده می کند. در هر لحظه تا هفت فریم تصدیق نشده می تواند در بافر فرستنده وجود داشته باشد. فیلد Seq در شکل ۳-۲۵ (الف) شماره ترتیبی فریم را نشان می دهد. فیلد Next نیز فیلد سواری مجانی برای تصدیق دریافت است. با این حال، در تمام انواع پروتکلهای HDLC مرسوم است که بجای سوار کردن شماره آخرین فریم دریافت شده، شماره اولین فریمی که هنوز دریافت نشده (فریمی که گیرنده منتظر آن است) برگردانده شود. این دو روش هیچ مزیتی بر یکدیگر ندارند، و انتخاب یکی از آنها به طراح پروتکل بستگی دارد؛ البته مشروط باینکه همواره از یک روش استفاده کند.

فیلد P/F مخفف Poll/Final (سرکشی/پایان) است. این فیلد در کامپیوترها یا مودمهایی بکار می رود که به چندین ترمینال سرکشی (polling) می کنند. اگر این فیلد محتوی P باشد، کامپیوتر (یا مودم) ترمینال را دعوت به ارسال داده می کند. در تمام فریمهایی که ترمینال می فرستد، فیلد P/F مقدار P دارد (بجز در آخرین فریم، که مقدار آن F است).

در برخی از پروتکلها بیت P/F باعث می شود تا ماشین طرف مقابل بلافاصله فریم سرپرستی را بفرستد، و منتظر سواری مجانی نشود. در ارتباطاتی که از فریمهای بدون شماره سود می برند، نیز این بیت کاربرد دارد.

Bits	1	3	1	3
(الف)	0	Seq (توالی)	P/F	Next (بعدي)
(ب)	1	0	Type (نوع)	P/F
(ج)	1	1	Type (نوع)	Modifier (تغییر دهنده)

شکل ۳-۲۵. فیلد کنترل در یک (الف) فریم اطلاعاتی، (ب) فریم سرپرستی، (ج) فریم بدون شماره.

انواع مختلف فریمهای سرپرستی با فیلد *Type* مشخص می شود. نوع 0 فریم تصدیق دریافت (که رسماً *RECEIVE READY* نامیده می شود) است، و مشخص می کند که گیرنده آماده دریافت فریم بعدیست. این فریم وقتی ارسال می شود که ترافیک جهت برگشت برای اجرای تکنیک سواری مجانی وجود نداشته باشد. نوع 1 فریم تصدیق دریافت منفی (که رسماً *REJECT* نامیده می شود) است، از آن برای تصدیق دریافت فریم با خطا استفاده می شود. در این حالت فیلد *Next* حاوی اولین فریمیست که درست دریافت نشده است (یعنی فریمی که باید دوباره ارسال شود). در اینجا فرستنده باید تمام فریمهای بعد از فریم *Next* را مجدداً بفرستد، و از این نظر شبیه پروتکل 5 است تا پروتکل 6.

نوع 2 فریم *RECEIVE NOT READY* است؛ این فریم اعلام می کند که تمام فریمهای قبل از *Next* درست دریافت شده اند، ولی خود *Next* خیر (و از این نظر شبیه *RECEIVE READY* است، با این تفاوت که جلوی ادامه ارسال را می گیرد). فریم *RECEIVE NOT READY* در واقع نوعی اعلام مشکل است از سوی گیرنده، مثلاً مشکل پُر شدن بافرها. بعد از برطرف شدن این وضعیت، گیرنده یکی از فریمهای کنترلی دیگر (مانند *RECEIVE READY* یا *REJECT*) را می فرستد.

نوع 3 فریم *SELECTIVE REJECT* است. این فریم فقط زمانی فرستاده می شود که گیرنده خواستار ارسال مجدد یک فریم خاص باشد؛ HDLC از این نظر شبیه پروتکل 6 است تا پروتکل 5، و برای مواقعی مفید است که اندازه پنجره ارسال نصف شماره ترتیبی باشد. بنابراین اگر گیرنده بخواهد فریمهای نامنظم را بافر کرده و فقط ارسال مجدد برخی از فریمهای معیوب را طلب کند، می تواند از *SELECTIVE REJECT* استفاده کند. این فریم کنترلی فقط در HDLC و ADCCP وجود دارد، ولی در SDLC و LAPB تعریف نشده است.

سومین نوع از فریمها، فریمهای بدون شماره (*Unnumbered*) است. این نوع گاهی کاربردهای کنترلی دارد، ولی در سرویسهای غیرقابل اعتماد غیر متصل نیز می توان از آن برای انتقال داده استفاده کرد. برخلاف دو نوع دیگر، این نوع فریم عملکردهای متفاوتی در انواع پروتکلهای بیت-سگرا است. در این حالت پنج بیت برای تعیین کارکرد فریم وجود دارد، ولی تمام ۳۲ حالت آن استفاده نمی شود.

در تمام پروتکلها فرمانی وجود دارد بنام *DISC* (قطع ارتباط) که اجازه می دهد یک ماشین خاموش شدن خود را به ماشینهای دیگر اعلام کند. فرمان دیگری وجود دارد که به یک ماشین اجازه می دهد تا بازگشت خود را اعلام کرده، و تمام شماره های ترتیبی را به 0 برگرداند؛ این فرمان *SNRM* (ست شدن حالت پاسخ عادی - *Set Normal Response Mode*) نام دارد. متأسفانه، «حالت پاسخ عادی» هر چیزی هست جز عادی. این حالت عبارتست از یک رابطه نامتقارر، که در آن یک سر خط «ارباب» و سر دیگر خط «رعیت» است. این فرمان از زمانی

به جای مانده که یک کامپیوتر بزرگ مرکزی وجود داشت و تعداد زیادی ترمینال به آن متصل می شد، که در این حالت مسلماً رابطه نامتقارن ارباب و رعیتی «عادی» محسوب می شود. برای آن که این پروتکلها بتوانند پاسخگوی نیازهای جدید (رابطه متقارن) باشند، در HDLC و LAPB فرمان دیگری وجود دارد بنام SABM (ست شدن حالت آسنکرون متعادل - Set Asynchronous Balanced Mode)، که خط را ریست کرده و دو سر آنرا به حالت هم ارز و متعادل درمی آورد. در این پروتکلها دو فرمان دیگر وجود دارد بنامهای SABME و SNRME که برترتیب با SABM و SNRM متناظر هستند، و فقط در آنها شماره ترتیبی فریمها (بجای ۳-بیتی) ۷-بیتی است. فرمان سومی که در تمام پروتکلها وجود دارد، FRMR (Frame Reject) است که نشان می دهد جمع تطبیقی فریم صحیح است ولی از نظر شکلی درست نیست. از میان چنین شکلهای نادرستی می توان به فریم سرپرستی نوع 3 در LAPB، فریمی که کوتاهتر از ۳۲ بیت است، و تصدیق دریافت فریمی که خارج از پنجره دریافت است، اشاره کرد. در فریم FRMR یک فیلد داده ۲۴ بیتی وجود دارد که علت خطا را توضیح می دهد. اطلاعاتی که در این فیلد مخابره می شود، عبارتند از: فیلد کنترل فریم معیوب، پارامترهای پنجره دریافت، و اطلاعاتی درباره طبیعت خطا.

احتمال خراب یا گم شدن فریمهای تصدیق دریافت نیز مانند سایر فریمها وجود دارد، بنابراین برای آنها نیز باید نوعی تصدیق دریافت پس فرستاد. برای این منظور فریم کنترلی خاصی بنام UA (تصدیق دریافت بدون شماره - Unnumbered Acknowledgement) در نظر گرفته شده است. از آنجائیکه همیشه فقط یک فریم تصدیق دریافت تصدیق نشده در گیرنده وجود دارد، هیچ ابهامی وجود ندارد که منظور از یک فریم UA کدام فریم تصدیق دریافت است.

سایر فریمهای کنترلی کارهایی مانند آماده سازی (initialization)، سرکشی (polling) و گزارش وضعیت (status report) انجام می دهند. فریم کنترلی دیگری نیز وجود دارد بنام UI (Unnumbered Information)، که می توان با آن اطلاعات دلخواه ارسال کرد؛ این اطلاعات برای لایه پیوند داده هستند، و به لایه شبکه داده نمی شوند.

پروتکل HDLC علیرغم کاربرد وسیع آن، بهیچوجه کامل نیست. در (Fiorini et al., 1994) می توانید بحثی درباره مشکلات این پروتکل را ببینید.

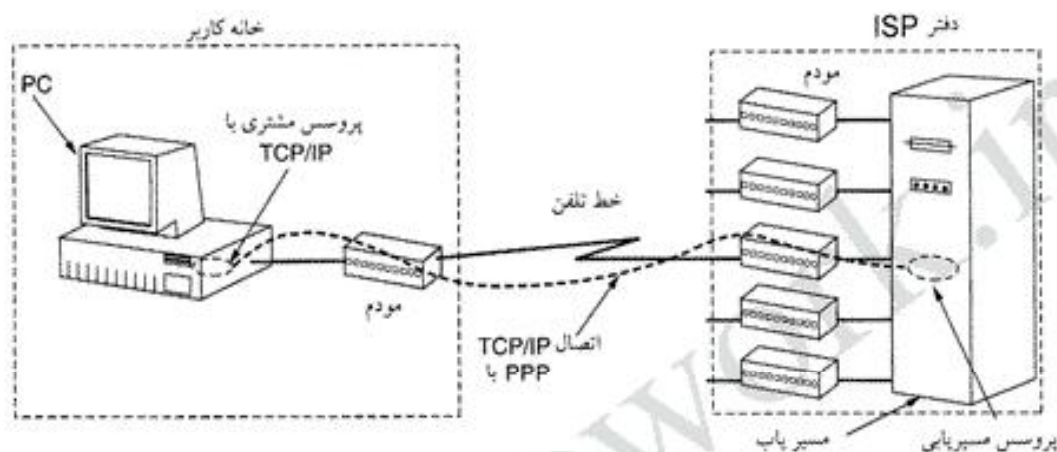
۲-۶-۳ لایه پیوند داده در اینترنت

اینترنت شامل ماشینهای متعددی (میزبان و مسیریاب) است، که توسط یک ستون فقرات به یکدیگر متصل می شوند. در یک ساختمان کوچک می توان از تکنیکهای LAN برای ارتباط استفاده کرد، ولی در اینترنت اغلب ارتباطات از نوع نقطه-به-نقطه (point-to-point) است. در فصل ۴ درباره لایه پیوند داده در LAN صحبت خواهیم کرد؛ در این قسمت به لایه پیوند داده در خطوط نقطه-به-نقطه می پردازیم.

در عمل، ارتباط نقطه-به-نقطه در دو حالت بکار برده می شود. اول، هزاران شرکت و مؤسسه دارای شبکه های محلی با تعداد زیادی ماشین میزبان (کامپیوترهای رومیزی، ایستگاههای کاری، کامپیوترهای سرور و غیره) و یک مسیریاب (یا یک پل - bridge) که در عمل همان وظیفه را انجام می دهد) هستند، و این مسیریابها در یک شبکه بزرگتر یکدیگر متصل شده اند. معمولاً این مسیریابها بوسیله ارتباط نقطه-به-نقطه و از طریق خطوط اجاره ای (leased line) به یک (یا دو) مسیریاب دیگر متصل می شوند. همین مسیریابها و خطوط اجاره ای هستند که زیر شبکه اینترنت را می سازند.

دومین حالتی که ارتباط نقطه-به-نقطه بکار برده می شود، میلیونها کاربر اینترنتی هستند که از منزل و با یک مودم (از طریق خط تلفن) به اینترنت متصل می شوند. اتفاقی که معمولاً می افتد اینست که کامپیوتر کاربر به

مسیریاب سرویس دهنده اینترنت (ISP) زنگ می زند، و از آن طریق (درست مثل یک میزبان معمولی) به اینترنت متصل می شود. در این روش فرقی نمی کند که خط تلفن معمولی است یا اجاره ای، فقط بعد از اینکه کاربر دیگر نیازی به آن نداشت، ارتباط قطع می شود. در شکل ۳-۲۶ این نوع ارتباط نقطه به نقطه را ملاحظه می کنید. مودمی که در این شکل نشان داده ایم یک مودم خارجی است، اما مودمهای داخلی نیز دقیقاً همان کار را انجام می دهند.



شکل ۳-۲۶. یک کامپیوتر خانگی می تواند نقش میزبان اینترنت را بازی کند.

در هر دو حالت (ارتباط مسیریاب-مسیریاب یا میزبان-مسیریاب) به یک پروتکل لینک داده نقطه به نقطه نیاز داریم تا وظایفی از قبیل فریم بندی، کنترل خطا و مانند آن را انجام دهد. پروتکلی که در اینترنت از آن استفاده می شود (و در این قسمت آنرا بررسی خواهیم کرد)، PPP نام دارد.

PPP - پروتکل نقطه به نقطه

اینترنت در موارد مختلفی، از قبیل ترافیک مسیریاب به مسیریاب یا ترافیک کاربر به ISP، به پروتکل نقطه به نقطه نیاز دارد. این پروتکل PPP (پروتکل نقطه به نقطه - Point-to-Point Protocol) نام دارد، که در RFC 1661 تعریف شده و در چند RFC دیگر (از قبیل RFC 1662 و RFC 1663) مشخصات آن بهبود یافته است. PPP ویژگیهای کنترل خطا دارد، از پروتکلهای مختلف پشتیبانی می کند، اجازه می دهد تا آدرس IP در زمان اتصال به طرف مقابل درخواست شود، تعیین هویت (authentication) انجام می دهد، و دهها ویژگی دیگر. مهمترین بخشهای PPP عبارتند از:

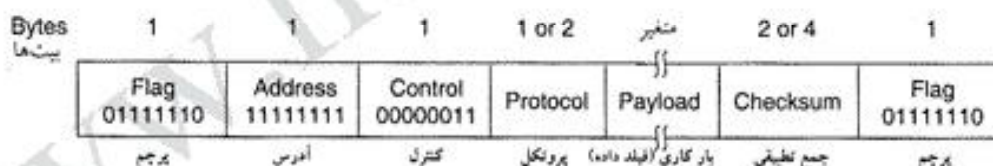
۱. یک روش فریم بندی، که ابتدا و انتهای فریمها را بوضوح مشخص می کند. فرمت فریم در PPP تشخیص خطا را نیز انجام می دهد.
۲. یک پروتکل کنترل لینک برای برقراری ارتباط، تست آن، مذاکره برای سایر گزینه ها، و در پایان قطع ارتباط بصورتی آپرومندان. این پروتکل LCP (پروتکل کنترل لینک - Link Control Protocol) نام دارد. LCP از هر دو ارتباط سنکرون و آسنکرون، و بیت-گرا یا بایت-گرا پشتیبانی می کند.
۳. روشی برای ارتباط و مذاکره درباره گزینه های لایه شبکه، که از طرز کار این لایه مستقل است. در این روش برای هر نوع لایه شبکه یک NCP (پروتکل کنترل شبکه - Network Control Protocol) وجود دارد.

برای اینکه ببینید این قطعات چگونه با یکدیگر جفت می شوند، سناریوی ارتباط کاربر به ISP را در نظر می گیریم. ابتدا PC کاربر از طریق مودم خود به مسیریاب ISP زنگ می زند. بعد از اینکه مودم مسیریاب گوشی را

برداشت و ارتباط برقرار شد، PC از طریق فیلد داده یک یا چند فریم PPP چند بسته LCP به مسیریاب می‌فرستد. پارامترهای PPP از طریق همین بسته‌ها و پاسخ آنها انتخاب می‌شود. بعد از آنکه بر سر این پارامترها توافق حاصل شد، یک سری بسته NCP برای پیکربندی لایه شبکه رد و بدل می‌شود. معمولاً PC هایی که به اینترنت متصل می‌شوند از TCP/IP استفاده می‌کنند، پس PC ما به یک آدرس IP نیاز دارد. تعداد آدرسهای IP آنقدر زیاد نیست که بتوان به همه آدرس ثابت اختصاص داد، به همین دلیل ISP تعدادی آدرس IP را بصورت دینامیک به PC هایی که به آن وصل می‌شوند، اختصاص می‌دهد. اگر یک ISP دارای n آدرس IP باشد، می‌تواند در هر لحظه (حداکثر) n کاربر متصل به اینترنت داشته باشد (البته تعداد کل مشترکان آن می‌تواند خیلی بیشتر باشد). یکی از بسته‌های NCP که درخواست IP می‌کند، این آدرس را به PC اختصاص می‌دهد.

از این لحظه به بعد PC ما درست مثل یک میزبان معمولی اینترنت است، و می‌تواند بسته‌های IP رد و بدل کند. وقتی کاربر ارتباط را قطع کند، NCP نیز ارتباط لایه شبکه را قطع کرده و آدرس IP را آزاد می‌کند. پس از آن LCP ارتباط لایه پیوند داده را قطع می‌کند، و کامپیوتر نیز به مودم دستور می‌دهد که گوشی را بگذارد و به ارتباط فیزیکی پایان دهد.

فرمت فریمهای PPP بسیار شبیه HDLC است (همیشه که نباید چرخ را از نو اختراع کرد). تفاوت اصلی PPP و HDLC اینست که، PPP کاراکتر-گرا (character-oriented) است نه بیت-گرا. بویژه، PPP از تکنیک لاگذاری بایت (byte stuffing) استفاده می‌کند، بنابراین تعداد بایتهای یک فریم همیشه عددی صحیح است. در PPP (برخلاف HDLC) نمی‌توان فریمی فرستاد که مثلاً 30.25 بایت داشته باشد. البته PPP می‌تواند (علاوه بر خطوط تلفن معمولی) روی SONET یا خطوط بیت-گرای HDLC نیز کار کند. فرمت فریم PPP را در شکل ۳-۲۷ ملاحظه می‌کنید.



شکل ۳-۲۷. فرمت فریم کامل PPP برای حالت بدون شماره.

تمام فریمهای PPP با بایت پرچم استاندارد HDLC (یعنی 01111110)، شروع می‌شوند، که اگر این بایت در داخل داده‌ها وجود داشته باشد از تکنیک لاگذاری بایت برای متمایز کردن آن استفاده می‌شود. بعد از آن فیلد Address می‌آید، که همیشه 11111111 است و مشخص می‌کند که تمام گیرنده‌ها باید این فریم را قبول کنند. با استفاده از این آدرس مشکل تخصیص آدرس‌های لینک داده نیز هم حل می‌شود.

بدنبال آدرس فیلد Control قرار دارد، که مقدار پیش‌فرض آن 00000011 است، و نشان می‌دهد که فریمها بدون شماره (unnumbered) هستند. بعبارت دیگر، در PPP فریمها شماره ترتیبی ندارند، و از فریم تصدیق دریافت نیز خبری نیست. البته در محیط‌های پرنویز (مانند لینکهای بیسیم) می‌توان از فریمهای شماره‌دار استفاده کرد. جزئیات دقیق این حالت در استاندارد RFC 1663 مشخص شده، ولی در عمل بندرت از آن استفاده می‌شود.

از آنجائیکه در پیکربندی پیش‌فرض فیلدهای Address و Control همواره ثابت هستند، LCP مکانیزم خاصی را بین دو سر خط پیاده می‌کند که این دو بایت بکلی حذف و فریمها کوتاهتر شوند.

فیلد چهارم PPP فیلد Protocol است، و مشخص می‌کند که داده موجود در فریم (قسمت Payload) از چه

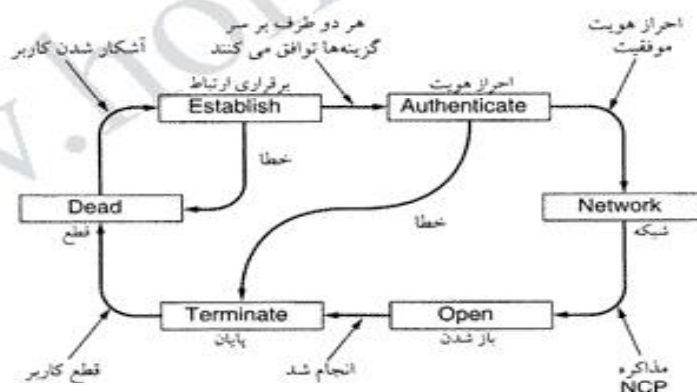
نوعیست. برای پروتکل‌های مختلف از قبیل LCP، NCP، IP، IPX و Apple Talk و پروتکل‌های دیگر کدهایی تعریف شده است. پروتکل‌هایی که با 0 شروع می‌شوند، پروتکل‌های لایه شبکه (مانند IP، IPX، OSI CLNP و XNS) هستند؛ آنهایی که با 1 شروع می‌شوند، برای مذاکره در باره پروتکل‌های دیگر بکار می‌روند (از جمله LCP، و یک NCP خاص برای هر یک از انواع لایه‌های شبکه). اندازه پیش‌فرض این فیلد 2 بایت است، ولی دو طرف می‌توانند از طریق LCP مذاکره کرده و اندازه آنرا به 1 بایت تقلیل دهند.

طول فیلد Payload متغیر است، و حداکثر مقدار آن در مذاکره اولیه مشخص می‌شود. اگر دو طرف در مذاکره اولیه (هنگام برقراری ارتباط) بر سر این عدد به توافق نرسند، از عدد پیش‌فرض 1500 بایت استفاده خواهد شد. اگر مقدار داده ارسالی به این حد نرسد، لایه پیوند داده بقیه را با کاراکتر خاصی پُر خواهد کرد.

بدنبال Payload فیلد Checksum می‌آید، که مقدار آن معمولاً 2 بایت است، ولی دو طرف می‌توانند بر سر جمع تطبیقی 2 بایتی هم توافق کنند.

بطور خلاصه، PPP یک مکانیزم فریم‌بندی چندپروتکلی است، که می‌توان از آن روی مودم، خطوط بیت-گرای HDLC، SONET و سایر لایه‌های فیزیکی استفاده کرد. این پروتکل از کشف خطا، مذاکره برای گزینه‌های مطلوب، فشرده‌سازی سرآیند (header compression) (و در صورت نیاز، از ارتباط قابل اعتماد با فرمت HDLC) پشتیبانی می‌کند.

اکنون اجازه دهید ببینیم در PPP برقراری خط و قطع ارتباط چگونه انجام می‌شود. در شکل ۳-۲۸ مراحل (ساده شده) برقراری خط، بکارگیری، و قطع آن نشان داده شده است. این مراحل برای ارتباط مودمی و یا مسیریاب-به-مسیریاب هر دو صادق است.



شکل ۳-۲۸. مراحل ساده شده برقراری و قطع خط در پروتکل PPP.

در شروع کار پروتکل خط در وضعیت DEAD است، و این به معنای آنست که هیچگونه کاربر یا ارتباطی در لایه فیزیکی وجود ندارد. بعد از برقراری ارتباط در لایه فیزیکی، خط به وضعیت ESTABLISH می‌رود. در این لحظه مذاکره بر سر گزینه‌های LCP شروع می‌شود، که اگر موفقیت‌آمیز باشد، به وضعیت AUTHENTICATE منجر می‌شود. حال دو طرف می‌توانند در صورت تمایل هویت یکدیگر را چک کنند. بعد از ورود به مرحله NETWORK، لایه شبکه با اجرای پروتکل NCP مناسب پیکربندی می‌شود. اگر این پیکربندی موفقیت‌آمیز

باشد، مرحله *OPEN* فرا رسیده و تبادل اطلاعات می تواند شروع شود. وقتی تبادل اطلاعات انجام شد، خط به مرحله *TERMINATE* رفته، و از آنجا دوباره به وضعیت *DEAD* برمی گردد و کاربر قطع می شود. در مرحله *ESTABLISH*، مذاکره بر سر گزینه های پروتکل لینک داده توسط LCP انجام می شود. البته خود LCP هیچ علاقایی به این گزینه ها ندارد، و فقط مکانیزمی برای مذاکره فراهم می آورد (بعبارت دیگر، پیشنهادی را فرستاده و بعد از دریافت پاسخ آنرا پذیرفته یا رد می کند). بررسی کیفیت خط (و اینکه آیا برای برقراری ارتباط باندازه کافی خوب هست یا نه) نیز بر عهده LCP است. قطع خط (بعد از پایان یافتن کار) یکی دیگر از وظایف پروتکل LCP است.

در RFC 1661 یازده نوع فریم LCP تعریف شده است، که آنها را در شکل ۳-۲۹ ملاحظه می کنید. چهار فریم *Configure-* به آغازکننده (I) اجازه می دهند تا گزینه ای را پیشنهاد کنند، و پاسخ دهنده (R) می تواند آنها را پذیرفته یا رد کند. اگر پاسخ دهنده گزینه ای را رد کند، می تواند پیشنهاد مورد نظر خود را ارائه کرده، و یا اعلام کند که اساساً مایل نیست راجع به آن مذاکره کند. گزینه ها و پاسخ آنها جزئی از فریمهای LCP هستند.

نام	جهت	توضیح
Configure-request	I → R	لیست گزینه ها و مقادیر پیشنهادی
Configure-ack	I ← R	تمام گزینه ها قبول می شوند
Configure-nak	I ← R	بعضی از گزینه ها قبول نمی شوند
Configure-reject	I ← R	بعضی از گزینه ها مذاکره نمی شوند
Terminate-request	I → R	تقاضای قطع خط
Terminate-ack	I ← R	قبول، خط قطع شد
Code-reject	I ← R	دریافت درخواست نامعلوم
Protocol-reject	I ← R	درخواست پروتکل نامعلوم
Echo-request	I → R	لطفاً این فریم را پس بفرست
Echo-reply	I ← R	فریم پس فرستاده شد
Discard-request	I → R	این فریم را نادیده بگیر (برای تست بود)

شکل ۳-۲۹. انواع فریمهای LCP.

کدهای *Terminate-* برای قطع کردن خط (وقتی که دیگر نیازی به آن نیست) هستند. کدهای *Code-reject* و *Protocol-reject* مشخص می کنند که پاسخ دهنده چیزی را دریافت کرده که معنی آنرا نمی فهمد. یکی از دلایل این وضعیت می تواند رخ دادن خطاهای کشف نشده روی خط باشد، ولی با احتمال بیشتر دلیل آن یکی نبودن ویرایش پروتکل LCP در دو سمت خط است. فریمهای *Echo-* برای تست کیفیت خط بکار برده می شوند. و بالاخره، فریم *Discard-request* برای دیباگ کردن بکار می آید (و نویسنده پروتکل می تواند از آن برای تست پروتکل خود استفاده کند). گیرنده این قبیل فریمها را نادیده می گیرد، و هیچ عکس العملی به آنها نشان نمی دهد.

برخی از مهمترین گزینه هایی که می توان درباره آنها مذاکره کرد، عبارتند از: حداکثر اندازه قسمت *Payload* فریم، فعال کردن احراز هویت و انتخاب پروتکل آن، فعال کردن ماینورینگ کیفیت خط در طول عملیات، و انتخاب گزینه های فشرده سازی سرآیند.

درباره پروتکلهای NCP حرف کلی چندانی نمی توان زد. هر پروتکل NCP خاص یک نوع لایه شبکه است، و بر سر گزینه های پیکربندی آن مذاکره می کند. برای مثال، در یک لایه شبکه IP احتمالاً مهمترین گزینه تخصیص آدرس IP است.

۷-۳ خلاصه

وظیفه لایه پیوند داده تبدیل استریم خام بیت های لایه فیزیکی به استریمی از فریمها، و هدایت این استریم به لایه شبکه است. روشهای مختلفی برای فریم بندی وجود دارد، که شمارش کاراکتر، لاگذاری بایت، و لاگذاری بیت از آن نمونه است. پروتکل های لینک داده می توانند خطاها را کنترل کرده، و در صورت نیاز فریمهای معیوب را مجدداً ارسال کنند. برای جلوگیری از غرق شدن گیرنده های گنبد در سیلاب داده های فرستنده های پرسرعت، لایه پیوند داده جریان داده ها را نیز کنترل می کند. یکی از پروتکل هایی که کنترل خطا و کنترل جریان در آن لحاظ شده، پروتکل پنجره لغزنده است.

پروتکل های پنجره لغزنده را می توان بر حسب اندازه پنجره ارسال و دریافت دسته بندی کرد. وقتی اندازه این پنجره ها هر دو 1 باشد، پروتکل توقف-انتظار نام دارد. اگر پنجره ارسال و دریافت (برای اجتناب از قفل شدن فرستنده در محیطهای با تأخیر زیاد) بزرگتر از 1 باشند، گیرنده می تواند فریمهای خراج از نظم را به کلی دور انداخته و یا آنها را بافر کند.

در این فصل چندین پروتکل را مورد بررسی قرار دادیم. پروتکل 1 برای محیطهای عاری از خطا طراحی شده، و می تواند انتقال اطلاعات را با هر سرعتی انجام دهد. در پروتکل 2 محیط همچنان بدون خطا فرض شده، ولی قادر است جریان داده ها را کنترل کند. در پروتکل 3 برای کنترل خطا از شماره ترتیبی فریمها و الگوریتم توقف-انتظار استفاده شده است. ترافیک در پروتکل 4 می تواند دوطرفه باشد، و در آن تکنیکی بنام سواری مجانی معرفی شده است. پروتکل 5 از تکنیک پنجره لغزنده «N تا به عقب برگرد» استفاده می کند. و بالاخره، در پروتکل 6 از تکنیکهای تکرار انتخابی و تصدیق دریافت منفی استفاده کردیم.

برای تست صحت و کارایی پروتکلها، روشهای مختلف مدلسازی (از جمله مدل ماشین حالت محدود و مدل شبکه پتری) را معرفی کردیم.

شبکه های بسیاری در لایه پیوند داده از پروتکل های بیت-گرا استفاده می کنند، که از آن میان می توان به SDLC، HDLC، ADCCP، یا LAPB اشاره کرد. در تمام این پروتکلها ابتدا و انتهای فریم با یک بایت پرچم مشخص می شود، و اگر این بایت در داده ها موجود باشد، از تکنیک لاگذاری بیت برای مشخص کردن آن استفاده می شود. همه این پروتکلها از یک پنجره لغزنده برای کنترل جریان سود می برند. در اینترنت نیز PPP بعنوان پروتکل اصلی لایه پیوند داده در خطوط نقطه-به-نقطه کاربرد گسترده ای دارد.

مسائل

- یک بسته از لایه بالاتر به ۱۰ فریم تقسیم شده، و احتمال اینکه هر یک از آنها سالم به مقصد برسد، ۸۰٪ است. اگر در لایه پیوند داده کنترل خطا انجام نشود، این پیام چند بار باید فرستاده شود تا تمام آن صحیح و سالم به مقصد برسد؟
- در لایه پیوند داده از گدگذاری زیر استفاده شده است:

A: 01000111; B: 11100011; FLAG: 01111110; ESC: 11100000

در هر یک از حالت های زیر، توالی بیت (باینری) فریم چهار کاراکتری A B ESC FLAG را نشان دهید: (الف) شمارش کاراکتر.

(ب) بایت پرچم با لاگذاری بایت.

(ج) بایت پرچم در ابتدا و انتها، با لاگذاری بیت.

- با تکنیکی که در کتاب شرح داده شد، قطعه داده FLAG D ESC FLAG C ESC B A در وسط یک استریم ظاهر شده است. خروجی لاگذاری این قطعه چیست؟

۴. یکی از همکلامی های شما عقیده دارد که قرار دادن یک بایت پرچم در ابتدای فریم و یکی در انتهای آن کار بیهوده ایست، و یک بایت کاملاً کفایت می کند (این بایت انتهای یک فریم، و ابتدای فریم بعدی محسوب خواهد شد). نظر شما چیست؟
۵. می خواهیم رشته 011110111110111110 را از طریق لایه پیوند داده ارسال کنیم. این رشته بعد از لاگذاری بیت به چه شکلی در می آید؟
۶. آیا در هنگام استفاده از لاگذاری بیت احتمال دارد که خطایی (از قبیل اضافه، کم شدن و یا تغییر یک بیت) رخ دهد، ولی جمع تطبیقی آنرا کشف نکند؟ اگر خیر، چرا؟ اگر بلی، چگونه؟ آیا طول جمع تطبیقی در اینجا نقشی دارد؟
۷. آیا فکر می کنید حالتی وجود دارد که یک پروتکل حلقه باز (مانند کد همینگ) بر پروتکل های مبتنی بر بازخور (که در این فصل دیدید) ارجحیت داشته باشد؟
۸. برای اطمینان بیشتر، بجای یک بیت توازن از کدی استفاده می کنیم که یک بیت توازن برای بیت های فرد و یک بیت توازن دیگر برای بیت های زوج دارد. فاصله همینگ این کد چقدر است؟
۹. برای ارسال یک پیام ۱۶ بیتی از کد همینگ استفاده کرده ایم. برای آنکه گیرنده بتواند تمام خطاهای تک بیتی را کشف و تصحیح کند، به چند بیت چک کننده نیاز داریم. روش کار را برای پیام 1101001100110101 نشان دهید. فرض کنید در این کد همینگ از توازن زوج استفاده شده است.
۱۰. می خواهیم یک بایت ۸ بیتی با مقدار باینری 10101111 را با استفاده از روش همینگ توازن زوج کد کنیم. خروجی چیست؟
۱۱. یک کد همینگ ۱۲ بیتی با مقدار هگزادسیمال 0xE4F به گیرنده می رسد. مقدار هگزادسیمال اولیه چه بوده است؟ فرض کنید بیش از یک بیت خطا رخ نداده است.
۱۲. یکی از تکنیک های تشخیص خطا، ارسال داده ها بصورت ماتریسی از n سطر و k ستون است که هر سطر و ستون دارای بیت های توازن خاص خود است. آخرین بیت در منتهی الیه سمت راست بیتی است که سطر و ستون مربوطه را چک می کند. آیا این روش می تواند خطاهای تک بیتی را کشف کند؟ خطاهای دوبیتی را چطور؟ خطاهای سه بیتی را چطور؟
۱۳. ماتریسی با n سطر و k ستون دارای بیت های توازن افقی و عمودی است. فرض کنید در هنگام انتقال اطلاعات ۴ بیت تغییر کرده است. احتمال کشف نشدن این خطا را بصورت یک عبارت ریاضی استخراج کنید.
۱۴. حاصل تقسیم $x^7 + x^5 + 1$ بر چند جمله ای مولد $x^3 + 1$ چیست؟
۱۵. می خواهیم با استفاده از تکنیک CRC استریم 10011101 را ارسال کنیم. چند جمله ای مولد $x^3 + 1$ است. استریم فرستاده شده چیست؟ فرض کنید بیت سوم از سمت چپ در حین ارسال معکوس می شود. نشان دهید که گیرنده می تواند این خطا را کشف کند.
۱۶. پروتکل های لینک داده همیشه CRC را بجای سرآیند در پی آیند پیام قرار می دهند. چرا؟
۱۷. یک کانال 4-kbps دارای تأخیر انتشار 20 msec است. تا چه اندازه فریمی کارایی پروتکل توقف-انتظار بیش از ۵۰٪ است؟
۱۸. یک ترانک T1 بطول 3000 km از فریم های 64 بیتی و پروتکل 5 استفاده می کند. اگر تأخیر انتشار 6 $\mu\text{sec}/\text{km}$ باشد، تعداد بیت های شماره ترتیبی چقدر باید باشد؟
۱۹. آیا در پروتکل 3 فرستنده می تواند تایمری را که در حال کار است، از نو شروع کند؟ اگر بلی، در چه موقعیتی؟ اگر خیر، چرا؟
۲۰. فرض کنید در یک پروتکل پنجره لغزنده تعداد بیت های شماره ترتیبی آنقدر زیاد است که «برگشت»

- هرگز رخ نمی دهد. چه رابطه ای باید بین لایه های پنجره ها و اندازه پنجره (که ثابت، و در فرستنده و گیرنده یکی است) برقرار باشد؟
۲۱. اگر روال *between* در پروتکل 5 بجای $a \leq b < c$ شرط $a \leq b \leq c$ را چک کند، چه تأثیری روی درستی یا کارایی پروتکل خواهد گذاشت؟ توضیح دهید.
۲۲. در پروتکل 6 وقتی یک فریم به گیرنده می رسد، چک می کند که آیا شماره ترتیبی آن همانی است که باید باشد، و آیا *no_nak* مقدار *true* دارد یا خیر. اگر هر دو شرط *true* باشند، یک *NAK* فرستاده می شود؛ در غیر اینصورت، تایمر کمکی راه اندازی می شود. اگر قسمت *else* را حذف کنیم، چه تأثیری روی درستی پروتکل می گذارد؟
۲۳. فرض کنید حلقه *while* سه دستوری نزدیک به انتهای پروتکل 6 را حذف کرده ایم. آیا این کار بر درستی پروتکل اثر می گذارد، یا فقط کارایی آنرا تحت تأثیر قرار می دهد؟ توضیح دهید.
۲۴. فرض کنید بخش *case* مربوط به خطاهای جمع تطبیقی را از دستور *switch* پروتکل 6 حذف کرده ایم. این کار چه تأثیری بر عملکرد پروتکل خواهد گذاشت؟
۲۵. کد *frame_arrival* در پروتکل 6 بخشی برای *NAK* ها دارد. این بخش زمانی اجرا می شود که فریم ورودی یک *NAK* باشد و شرط دیگری وجود داشته باشد. سناریویی را شرح دهید که در آن وجود این شرط دوم الزامی باشد.
۲۶. فرض کنید می خواهید برای لایه پیوند داده خطی برنامه بنویسید که در آن داده ها فقط به سمت شما می آیند، و شما هیچ چیزی نمی فرستید. سمت مقابل از پروتکل HDLC با شماره ترتیبی 3 بیتی، و پنجره 7 فریمی استفاده می کند. برای بالا بردن کارایی سیستم، تصمیم گرفته اید حداکثر فریمهای خارج از نظم ممکنه را بافر کنید، ولی مجاز به دستکاری در نرم افزار سمت فرستنده نیستید. آیا می توان پنجره دریافتی بزرگتر از 1 داشت، و تضمین کرد که این پروتکل هرگز با شکست مواجه نشود؟ اگر بلی، بزرگترین پنجره ای که می توان با اطمینان بکار برد، چقدر است؟
۲۷. پروتکل 6 را روی یک خط 1-Mbps عاری از خطا در نظر بگیرید. حداکثر اندازه فریم 1000 بیت است. بسته ها با فواصل یک ثانیه ای تولید می شوند. فاصله زمانی انقضای تایمر 10 msec است. اگر تایمر مخصوص تصدیق دریافت را حذف کنیم، انقضاهای غیر لازم رخ خواهد داد. یک پیام با طول متوسط چند بار باید مجدداً ارسال شود؟
۲۸. در پروتکل 6 داریم: $MAX_SEQ = 2^n - 1$. با اینکه این شرط برای استفاده بهینه از بیت های سرآیند آشکارا مناسب است، نشان ندادیم که الزامی هم هست. اگر، برای مثال $MAX_SEQ = 4$ ، آیا باز هم این پروتکل بدرستی کار می کند؟
۲۹. با استفاده از یک کانال ماهواره ای 1-Mbps که زمان رسیدن سیگنال از زمین به ماهواره 270 msec است، فریمهای 1000 بیتی ارسال می کنیم. فریمهای تصدیق دریافت همیشه با سواری مجانی برمی گردند، و سرآیند فریمها بسیار کوتاه است. حداکثر نرخ مصرف قابل دستیابی در پروتکل های زیر چقدر است؟
- (الف) پروتکل توقف-انتظار.
- (ب) پروتکل 5.
- (ج) پروتکل 6.
۳۰. مقدار انلااف پهنای باند پروتکل 6 را (مربوط به سرآیند و ارسال مجدد) در یک کانال ماهواره ای 50-kbps پرتراфик، با فریمهایی متشکل از 40 بیت سرآیند و 3960 بیت داده، محاسبه کنید. فرض کنید زمان رسیدن سیگنال از زمین به ماهواره 270 msec است؛ فریمهای ACK هرگز ارسال نمی شوند؛

فریمهای NAK دارای طولی معادل 40 بیت هستند؛ نرخ خطا در فریمهای داده ۱٪، و در فریمهای NAK قابل چشم پوشی است؛ و شماره های ترتیبی 8 بیتی هستند.

۳۱. می خواهیم روی یک کانال ماهواره ای عاری از خطا با ظرفیت 64 kbps فریمهای 512 بیتی (در یک جهت)، با فریمهای تصدیق دریافت بسیار کوتاه (در جهت دیگر)، بفرستیم. اگر اندازه پنجره 1، 7، 15 و 127 باشد، حداکثر ظرفیت خط چقدر خواهد بود؟ زمان ارسال سیگنال از زمین به ماهواره را 270 msec بگیرد.

۳۲. یک کابل T1 بطول 100 km را در نظر بگیرید. سرعت انتشار امواج در این کابل 2/3 سرعت نور در خلاء است. چند بیت در این کابل جا می شود؟

۳۳. فرض کنید پروتکل 4 را با ماشین حالت محدود مدل کرده ایم. هر ماشین چند حالت دارد؟ کانال مخابراتی چند حالت دارد؟ سیستم کامل (دو ماشین و یک کانال مخابراتی) چند حالت دارد؟ از خطاهای جمع تطبیقی صرف نظر کنید.

۳۴. توالی آتش شبکه پتری شکل ۳-۲۳ که با توالی حالت (01A)، (010)، (01-)، (01A)، (000) در شکل ۳-۲۱ متناظر است، را در نظر بگیرید. توضیح دهید که این توالی چه چیزی را نشان می دهد.

۳۵. شبکه پتری دستورات گذار $AC \rightarrow B$ ، $AC \rightarrow B$ ، $CD \rightarrow E$ ، $CD \rightarrow E$ و $E \rightarrow CD$ را رسم کنید. از این شبکه پتری گراف دسترسی حالت محدود برای رسیدن به حالت ACD را رسم کنید. این گراف چه مفهوم شناخته شده ای را مدل می کند؟

۳۶. پروتکل PPP از HDLC مشتق شده، که در آن برای جلوگیری از تفسیر اشتباه بابت پرچم در داده ها از تکنیک لاگذاری بیت استفاده می شود. یک دلیل بیاورید که چرا PPP از لاگذاری بابت استفاده می کند، نه لاگذاری بیت.

۳۷. حداقل سرپاره یک بسته IP که با PPP فرستاده شده، چقدر است؟ فقط سرپاره PPP را در نظر بگیرید، نه سرپاره سرآیند IP را.

۳۸. هدف از این تمرین آزمایشگاهی پیاده سازی یک مکانیزم کشف خطا با استفاده از الگوریتم CRC (که در متن کتاب توضیح دادیم) است. دو برنامه بنویسید: مولد (generator) و تست کننده (verifier). برنامه مولد یک رشته n بیتی از 0 ها و 1 ها را از ورودی استاندارد (یا فرمت ASCII) می خواند. خط دوم یک چند جمله ای k بیتی است، که آن هم بصورت ASCII از ورودی خوانده می شود. برنامه خروجی خود را، که تشکیل شده از $n + k$ بیت 0 و 1 (و در واقع همان ورودی گد شده است)، روی خروجی استاندارد می نویسد. (برنامه مولد این خروجی را هم با فرمت ASCII بیرون می دهد). برنامه تست کننده خروجی مولد را گرفته، و با پیامی نشان می دهد که آیا این رشته درست است یا خیر. برای تست این دو برنامه، یک برنامه کمکی دیگر (بنام alter) بنویسید که یکی از بیت های خط اول را معکوس کند و سایر بیت ها را بهمان صورت باقی بگذارد (شماره بیتی که باید معکوس شود را - با آغاز شمارش از سمت چپ - بصورت آرگومان ورودی به این برنامه بدهید). با نوشتن

generator < file | verifier

برنامه باید پاسخ «درست است» بدهد، اما با دستور

generator < file | alter arg | verifier

باید پیام «درست نیست» بگیرد.

۳۹. برنامه ای بنویسید که رفتار یک شبکه پتری را شبیه سازی کند. این برنامه باید دستورات گذار، و لیستی از حالت های لایه شبکه (هنگام ارسال و دریافت یک بسته جدید) را بعنوان ورودی بخواند. برنامه باید از حالت اولیه (که آنرا هم از ورودی می خواند) یکی از گذارهای فعال را بصورت تصادفی آتش کند، و سپس چک کند که آیا هر یک از ماشینها ۲ بسته قبول می کند، بدون این که بین آنها یک بسته جدید بفرستد.

زیر لایه نظارت بر دسترسی به رسانه انتقال

به گونه ای که در فصل اول اشاره کردیم، شبکه ها را می توان به دو رده تقسیم بندی کرد: آنهایی که از اتصالات نقطه به نقطه استفاده می کنند و آنهایی که از کانالهای پخش فراگیر (Broadcast) بهره می گیرند. در این فصل به شبکه های پخش فراگیر و پروتکل های آنها خواهیم پرداخت.

در هر شبکه فراگیر مسئله اصلی آنست که وقتی برای دسترسی به کانال انتقال، رقابت وجود دارد چگونه می توان تعیین کرد که چه کسی باید از کانال استفاده کند. برای روشن تر شدن قضیه یک کنفرانس تلفنی را با حضور شش نفر، در نظر بگیرید که از طریق شش خط تلفن بهم متصل شده و نشستی را ترتیب داده اند و هر کدام قادرند با دیگران گفت و شنود داشته باشند. در چنین وضعیتی کاملاً طبیعی است که وقتی یک نفر صحبت خود را قطع می کند دو یا چند نفر به طور همزمان شروع به حرف زدن نمایند و برای لحظاتی بی نظمی و اغتشاش بر جلسه حاکم شود. در ملاقاتهای رو در رو یکمک اشاره و علامت، از بروزی نظمی احتراز می شود؛ مثلاً شخص مایل به گفتگو، دست خود را به علامت اجازه خواستن برای شروع سخن، بالا می برد. وقتی فقط یک کانال منفرد در اختیار است، تعیین نفر بعدی برای ارسال، دشوارتر خواهد بود. برای حل این مسئله پروتکل های متعددی عرضه شده است که معرفی آنها شاکله این فصل را تشکیل می دهد. در ادبیات شبکه، کانالهای فراگیر گاهی با عناوین «کانالهای با دسترسی چندگانه» (Multiaccess Channel) یا «کانالهای با دسترسی تصادفی» (Random Access Channel) معرفی می شوند.

پروتکل هایی که برای تعیین نفر بعدی در استفاده از کانال مشترک کاربرد دارند متعلق به زیر لایه ای از لایه پیوند داده ها هستند که اصطلاحاً زیر لایه MAC (Medium Access Control) نامیده می شود. زیر لایه MAC در شبکه های محلی که اغلب آنها از کانالهای مشترک به عنوان زیربنای ارتباط استفاده می کنند، از اهمیت ویژه ای برخوردار است. در مقابل، به غیر از شبکه های ماهواره ای، تمام شبکه های WAN از خطوط نقطه به نقطه بهره گرفته اند [و در آنها زیر لایه MAC جایگاهی ندارد]. از آنجایی که کانالهای با دسترسی چندگانه و شبکه های LAN کاملاً به یکدیگر مرتبط هستند لذا در این فصل بطور عام شبکه های LAN را بررسی می کنیم و موارد معدودی را نیز که مستقیماً مربوط به زیر لایه MAC نمی شوند، بررسی خواهیم نمود.

از دیدگاه فنی، زیر لایه MAC بخش زیرین لایه پیوند داده ها محسوب می شود و منطقاً می بایست قبل از آنکه در فصل سوم به پروتکل های نقطه به نقطه بپردازیم ابتدا این زیر لایه را بررسی می کردیم. ولیکن برای اغلب افراد، فهم پروتکل های دوطرفه ساده تر از پروتکل های چند طرفه است. [منظور از پروتکل دوطرفه پروتکلیست که در آن فقط و فقط دو ماشین درگیر مبادله داده هستند.] به همین دلیل در روند تشریح مطالب (که از لایه های پائین به

سمت بالا خواهد بود)، اندکی از این ترتیب تخطی کردیم.

۱.۴ مسئله تخصیص کانال

مضمون اصلی این فصل آنست که چگونه یک کانال مشترک و فراگیر را بین کاربران رقیب تقسیم نماییم. در ابتدا به الگوهای تخصیص «ایستا» و «پویا» نگاهی می اندازیم؛ سپس به تشریح تعدادی از الگوریتمهای خاص در این زمینه خواهیم پرداخت.

۱.۴.۱ تخصیص ایستای کانال در شبکه های LAN و MAN

روش سنتی در تخصیص یک کانال منفرد، (مثل خطوط اصلی تلفن - Telephone Trunk)، بین چندین کاربر که برای استفاده از آن رقابت می کنند، روش FDM (تسهیم در حوزه فرکانس) است. اگر N کاربر حضور داشته باشند، پهنای باند کانال به N بخش مساوی تقسیم می شود و به هر کاربر یکی از این بخشها اختصاص داده می شود (شکل ۳۱-۲ را ببینید). از آنجایی که هر کاربر دارای یک باند فرکانس اختصاصی است لذا کاربران هیچگونه تداخل و مزاحمتی برای یکدیگر ندارند. وقتی تعداد کاربران ثابت و کم باشد و هر کدام نیز دارای بار سنگین (و بافر شده) ترافیک باشند (همانند مراکز سوئیچ تلفن)، روش FDM مکانیزمی ساده و کارآمد برای تخصیص کانال خواهد بود. ولیکن وقتی تعداد ارسال کنندگان زیاد و دائماً در حال تغییر باشد، یا ترافیک ارسالی آنها به صورت لحظه ای و انفجاری تولید شود، FDM مشکلات متعددی را بروز خواهد داد: اگر طیف فرکانسی کانال به N بخش مجزا تقسیم شود ولی تعداد کاربرانی که تمایل به ارسال و مخابره داده دارند کمتر از N باشد بخش ارزشمندی از این طیف فرکانسی تلف خواهد شد. اگر تعداد کاربران بیش از N باشد، برخی از آنها بدلیل کمبود پهنای باند، مجوز ارسال نخواهند داشت؛ حتی اگر برخی از کاربرانی که بدانها باند فرکانسی تخصیص داده شده به ندرت بخواهند چیزی ارسال یا دریافت کنند.

با این وجود، حتی اگر بتوان تعداد کاربران را عدد ثابت N فرض کرد، باز هم تقسیم ایستای طیف فرکانسی کانال منفرد به تعدادی زیرکانال، ذاتاً روشی ناکارآمد و بی کفایت تلقی می شود؛ مشکل اساسی آنست که وقتی برخی از کاربران، تقاضای ارسال نداشته باشند پهنای باند آنها هدر می رود. گذشته از آن در بسیاری از سیستم های کامپیوتری، ترافیک داده ها بشدت انفجاری است (نسبت حداکثر ترافیک به متوسط ترافیک در حدود 1000:1 است). در نتیجه اکثر اوقات کانالها خالی و بلااستفاده باقی می ماند.

کارآئی بسیار ضعیف روش FDM را می توان با یک محاسبه ساده در «نظریه صف» (Queuing Theory) اثبات کرد. برای شروع، فرض کنید بخواهیم زمان متوسط تاخیر T را برای کانالی محاسبه کنیم که در آن نرخ ارسال C بیت بر ثانیه، نرخ دریافت فریمها λ فریم بر ثانیه و طول هر فریم تصادفی است و از تابع چگالی احتمالی نمایی با میانگین $1/\mu$ بیت بر فریم تبعیت می کند.

با این پارامترها، نرخ دریافت فریمها λ فریم بر ثانیه و نرخ سرویس دهی $\mu.C$ فریم بر ثانیه خواهد بود. با استفاده از نظریه صف، می توان نشان داد که اگر نرخ دریافت و زمان سرویس دهی به فریمها از تابع توزیع پواسون تبعیت کند خواهیم داشت:

$$T = \frac{1}{\mu.C - \lambda}$$

بعنوان مثال اگر C معادل 100 مگابیت بر ثانیه، متوسط طول فریمها (یعنی $1/\mu$) معادل 10000 بیت و نرخ دریافت فریمها (یعنی λ) معادل 5000 فریم بر ثانیه باشد، متوسط تاخیر هر فریم $T = 200 \mu s$ خواهد بود. دقت کنید که اگر از تاخیر صف (یعنی تاخیر انتظار فریم) صرف نظر می کردیم و زمان ارسال 10000 بیت را بر روی یک شبکه

100Mbps محاسبه می نمودیم به پاسخ اشتباه 100 میکروثانیه می رسیدیم. این نتیجه فقط زمانی صدق می کند که هیچ رقابتی در بدست گرفتن کانال وجود نداشته باشد. حال بیایید این کانال منفرد را به N زیرکانال مستقل با ظرفیت C/N بیت بر ثانیه تقسیم کنیم. در این حالت نرخ متوسط ورودی به هر یک از این زیرکانالها λ/N خواهد بود. با محاسبه مجدد T به دست می آوریم:

$$T_{FDM} = \frac{1}{\mu(C/N) - (\lambda/N)} = \frac{N}{\mu.C - \lambda} = N.T \quad \text{رابطه (۴-۱)}$$

متوسط تاخیر در FDM، N برابر بیشتر از زمانی است که تمام فریمها به ترتیب در یک صف مرکزی طولانی و مرتب شده، پشت سرهم ارسال شوند.

تمام استدلالات و مباحثاتی که در مورد روش FDM اعمال کردیم دقیقاً در مورد TDM نیز صدق می کند. به هر کاربر بصورت ثابت یکی از N برش زمانی (slot time) اختصاص داده می شود. اگر یک کاربر از برش زمانی تخصیص یافته به خود استفاده نکند، آن زمان بلااستفاده مانده و هدر خواهد رفت. با استفاده مجدد از مثال قبلی، اگر یک شبکه 100Mbps را به ده شبکه 10Mbps تقسیم کرده و هر کاربر از یکی از آنها استفاده کند، میانگین تاخیر از 200 میکروثانیه به 2 میلی ثانیه افزایش خواهد یافت.

از آنجاکه هیچیک از روشهای معمول و ایستای تخصیص کانال در محیطهای با ترافیک انفجاری کار نخواهد کرد لذا در ادامه به بررسی روشهای پویا می پردازیم.

۲-۱-۴ تخصیص پویای کانال در LAN و MAN

قبل از آنکه به اولین روش از روشهای متعددی تخصیص کانال پیردازیم دسته بندی و فرموله کردن مسائل و مشکلات تخصیص کانال مفید خواهد بود. کل کاری که باید برای تخصیص کانال انجام شود مبتنی بر پنج فرض اساسی است که در زیر تشریح شده اند:

۱. مدل ایستگاه (Station Model): این مدل شامل N ایستگاه مستقل (مثل کامپیوتر، تلفن یا دستگاههای مخابراتی شخصی) است که در هر کدام از آنها یک برنامه یا کاربر، فریمهایی را برای ارسال تولید می کند. برخی از اوقات به ایستگاهها، «پایانه» (یا ترمینال) نیز گفته می شود. احتمال آنکه در بازه زمانی ΔT فریمی تولید شود، $\lambda \Delta T$ است که در آن λ یک مقدار ثابت است (در حقیقت λ میانگین نرخ تولید فریمهای جدید است). به محض آنکه فریمی تولید گردد، ایستگاه متوقف شده و تا زمانی که آن فریم به صورت موفقیت آمیز ارسال نشود کاری انجام نمی دهد.

۲. فرض کانال منفرد (Single Channel Assumption): در این حالت تنها یک کانال منفرد و مشترک برای مخابرات داده در اختیار ایستگاهها است. تمام ایستگاهها می توانند اطلاعات خود را بر روی این کانال بفرستند یا از آن دریافت کنند. از دیدگاه سخت افزاری تمام ایستگاهها هم ارز و معادل یکدیگرند؛ اگرچه ممکن است در نرم افزار پروتکل به هر ایستگاه، اولویتی خاص داده شود.

۳. فرض تصادم (Collision Assumption): هرگاه دو فریم بطور همزمان [بر روی کانال مشترک] ارسال شوند یا یکدیگر تداخل کرده و سیگنال حاصل بی ارزش و نامعتبر خواهد بود. این رخداد اصطلاحاً «تصادم» (Collision) نامیده می شود. هر ایستگاه می تواند از وقوع تصادم آگاه گردد. فریمی که در حین ارسال آن تصادم رخ داده است باید از نو فرستاده شود. در این مدل هیچ خطائی به غیر از خرابی فریم در اثر تصادم لحاظ نمی شود.

۴. الف: مدل زمان پیوسته (Continuous Time): در این مدل، ارسال فریمها می تواند در هر لحظه از زمان

شروع شود و هیچگونه سیگنال ساعت مرکزی (سراسری) که زمان را به برشهای گسسته و مجزا تقسیم کند، وجود ندارد.

۴. ب: «مدل زمان گسسته» (Slotted time): در این مدل، زمان به برشهای گسسته و مستقلی تقسیم می شود و ارسال فریم همیشه باید در ابتدای یکی از این برشهای زمانی انجام گیرد. در هر برش زمان ممکن است صفر، یک، یا چند فریم ارسال شود که به ترتیب: صفر فریم به معنای بیکار و بلااستفاده ماندن آن برش زمانی، یک فریم به معنای ارسال موفق و چند فریم معادل تصادم خواهد بود.

۵. الف: شنود سیگنال حامل (Carrier Sense): در این مدل، ایستگاهها قبل از شروع به ارسال فریم خود، قادرند تشخیص بدهند که آیا کانال مشغول است یا آزاد؟ اگر ایستگاهی احساس کند که کانال مشغول است هیچگاه سعی در استفاده از آن نخواهد کرد مگر آنکه مجدداً کانال بیکار شود.

۵. ب: عدم شنود سیگنال حامل (No Carrier Sense): در این مدل، ایستگاهها قادر نیستند قبل از استفاده از کانال، آنرا بشنوند و سیگنال روی آنرا احساس کنند؛ لذا فقط پس از فرستادن فریم می توان تعیین کرد که آیا ارسال موفق بوده یا تصادم پدیده آمده است.

اندکی توضیح در خصوص برخی از فرضیات فوق مفید خواهد بود: اولین بند اذغان می دارد که ایستگاهها مستقل هستند و فریمها را با نرخ ثابتی تولید می کنند. [بهراتی نرخ میانگین تولید فریمها ثابت است. -م] همچنین در این بند تلویحاً فرض شده که هر ایستگاه تنها یک کاربر یا برنامه فعال دارد و بدین ترتیب هرگاه یک ایستگاه، متوقف (بلوکه) شود هیچ فریم جدیدی تولید نخواهد شد. مدلهای پیچیده دیگری نیز وجود دارد که در آنها ایستگاهها اجازه دارند به صورت چندبرنامه ای (Multiprogrammed) تولید فریمها را ادامه بدهند (حتی وقتی که ایستگاه در انتظار ارسال فریم قبلی بلوکه شده است)؛ ولیکن تحلیل چنین ایستگاههایی بسیار دشوار است. فرض کانال منفرد، هسته اصلی این مدل است. ایستگاهها بجز یک کانال واحد و مشترک راهی برای مبادله اطلاعات ندارند. ایستگاهها نمی توانند همانند یک کلاس درس یا بالا بردن دست خود، از معلم کلاس برای صحبت کردن اجازه بگیرند!

فرض تصادم نیز محوری است، اگر چه برخی از سیستمها (بویژه سیستمهای مبتنی بر «طیف گسترده» - Spread Spectrum) از این فرض مستثنی بوده و نتایج شگفت آوری نیز به همراه دارند؛ همچنین در شبکه های توکن رینگ (Token Ring) یک نشانه خاص (توکن) ایستگاه به ایستگاه می چرخد و هر ایستگاه که آنرا در اختیار بگیرد اجازه ارسال فریم خود را خواهد داشت ولیکن در بخشهای بعدی به کانالهای منفردی خواهیم پرداخت که برای استفاده از آنها رقابت وجود دارد و تصادم پدید می آید.

در خصوص زمان ارسال فریمها، دو فرض (۱) زمان پیوسته (۲) زمان گسسته قابل اعمال است. برخی از سیستمها از مدل اول پیروی می کنند و برخی دیگر از مدل دوم؛ بنابراین ما هر دو مدل را تحلیل خواهیم کرد. در هر سیستم فقط یکی از این مدلها قابل اعمال است.

همچنین در یک شبکه ممکن است سیگنال روی کانال احساس شود (فرض ۵-الف) یا شنود سیگنال میسر نباشد (فرض ۵-ب). شبکه های محلی (LAN) عموماً قادر به احساس سیگنال روی کانال هستند ولیکن در شبکه های بی سیم این مدل قابل استفاده نخواهد بود چراکه برخی از ایستگاهها در محدوده شنود سیگنال ایستگاههای دیگر نیستند. ایستگاههایی که به کانالهای سیمی متصل هستند در مدل «شنود سیگنال حامل» (Carrier Sense) قرار می گیرند و قادرند به محض کشف پدیده تصادم، ارسال فریم را خاتمه بدهند. کشف تصادم در شبکه های بی سیم به دلایل فنی مهندسی به ندرت قابل انجام است. دقت کنید که کلمه «حامل»

(Carrier) در اینجا اشاره به یک سیگنال الکتریکی بر روی کابل دارد.

۲-۴ پروتکل های دسترسی چندگانه

الگوریتم های پیشماری در مورد تخصیص کانال های با دسترسی چندگانه [کانال های مشترک] معرفی شده اند. در بخش های آتی نمونه هایی از جالبترین این پروتکل ها را بررسی کرده و مثالهایی از کاربرد آنها ارائه خواهیم نمود.

۱-۲-۴ ALOHA

در آوان ۱۹۷۰، نورمن آبرامسون و همکاران او در دانشگاه هاوانی روشی جدید و جالب برای حل مسئله تخصیص و دسترسی به کانال های اشتراکی ابداع کردند. بعداً، کار آنها توسط بسیاری از پژوهشگران ادامه یافت و تکمیل شد. (مرجع (Abramson, 1985) اگر چه کار آقای آبرامسون که سیستم ALOHA نامیده شد براساس پخش امواج رادیویی زمینی بود، ولیکن نظریه آنها در هر سیستمی که در آن کاربران برای استفاده از یک کانال مشترک، به صورت ناهماهنگ رقابت می کنند، قابل اعمال و پیاده سازی است.

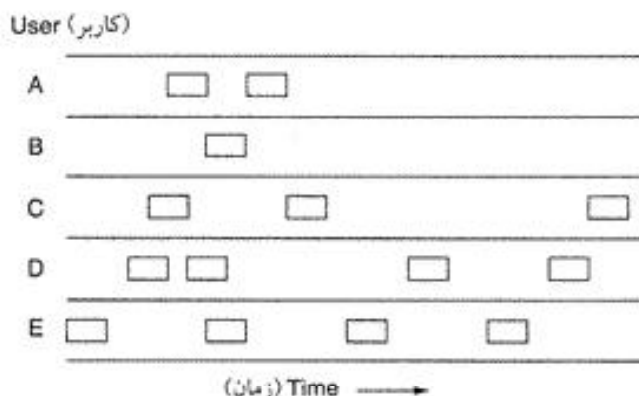
در اینجا دو نسخه متفاوت از ALOHA یعنی Pure ALOHA و Slotted ALOHA را تشریح خواهیم کرد. تفاوت این دو نسخه در تقسیم بندی زمان به برشهایی است که فریمها بتوانند در خلال یکی از آنها ارسال شوند. Pure ALOHA نیازی به هماهنگی زمانی (Time Synchronization) ندارد در حالیکه Slotted ALOHA نیازمند این هماهنگی است.

Pure ALOHA

ایده اصلی در سیستم ALOHA بسیار ساده است: کاربران اجازه دارند هر زمان که داده ای برای ارسال داشتند آنرا بفرستند. البته تصادمهایی رخ خواهد داد و فریمهایی که تصادم کنند از بین خواهند رفت. با این وجود در ALOHA، یک «کانال بازگشت سیگنال» (Feedback) وجود دارد که فرستنده با گوش دادن به این کانال، می تواند متوجه بروز تصادم و خرابی فریم شود. در شبکه های محلی LAN بروز تصادم به صورت سریع و آبی کشف می شود در حالیکه در شبکه های ماهواره ای یک ایستگاه پس از گذشت ۲۷ میلی ثانیه می تواند متوجه شود که آیا ارسال او موفق بوده یا نه! هرگاه به هر دلیلی امکان نشود سیگنال بازگشتی در حین ارسال وجود نداشته باشد بایستی دریافت فریمها توسط گیرنده تایید گردد. [با ارسال فریمهای مستقلی به نام Ack] هرگاه فریمی خراب شود، فرستنده آن، به اندازه یک زمان تصادفی صبر خواهد کرد و آن را مجدداً ارسال می کند. زمان انتظار قطعاً باید تصادفی باشد وگرنه تصادمها در یک دور نامتناهی تکرار خواهند شد. سیستمهایی که در آنها چندین کاربر از یک کانال مشترک به نحوی استفاده کنند که احتمال تصادم و تلاقی وجود دارد اصطلاحاً «سیستم های رقابتی» نامیده می شوند. در شکل ۱-۴ نمایشی از تولید فریم در سیستم ALOHA نشان داده شده است. در این شکل طول تمام فریمها را یکسان در نظر گرفته ایم زیرا اگر طول فریمها، اندازه ثابتی داشته باشند کارآیی و توان خروجی ALOHA حداکثر خواهد بود.

هرگاه دو فریم بطور همزمان بر روی کانال ارسال شوند، تصادم رخ داده و هر دو خراب خواهند شد. حتی اگر اولین بیت از یک فریم جدید با آخرین بیت از فریم قبلی تداخل کند هر دو فریم بطور کامل خراب شده و بعداً باید از نو ارسال شوند زیرا کدهای کشف خطا نمی توانند (و نباید هم بتوانند) تشخیص بدهند خطا در کجا اتفاق افتاده و بدین ترتیب کل فریم بلااستفاده خواهد بود.

سوال جالبی که پیش می آید آنست که کارآیی کانال در ALOHA چقدر است؟ عبارت دیگر می خواهیم بدانیم در این محیط نامنظم و هرج و مرج، چند درصد از کل فریمهای ارسالی از تصادم جان سالم به در می برند؟



شکل ۴-۱. در Pure ALOHA فریمها در زمانهای کاملاً دلخواه ارسال می شوند.

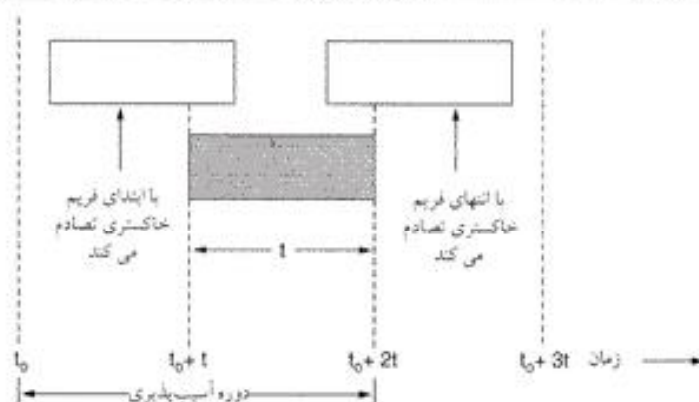
در ابتدا فرض می کنیم که تعدادی نامتناهی از کاربران در شبکه وجود دارند که پشت کامپیوترهای خود نشسته اند؛ هر کاربر در یکی از دو وضعیت «تایپ» یا «انتظار» قرار دارد. هرگاه تایپ یک خط به اتمام رسید و کلید Enter فشار داده شد، کاربر از تایپ دست می کشد و در انتظار پاسخ باقی می ماند. ایستگاه، فریم حاوی این خط را بلافاصله بر روی کانال می فرستد و بررسی می کند که آیا ارسال موفقیت آمیز بوده است؟ اگر فرایند ارسال موفق باشد کاربر پاسخ خود را دریافت کرده و عمل تایپ را از سر می گیرد و در غیر اینصورت، کاربر باز هم منتظر می ماند تا ارسال فریم آنقدر تکرار شود تا بالاخره یکی از آنها به سلامت به مقصد برسد.

«زمان فریم» (Frame Time)، مقدار زمانی است که طول می کشد تا یک فریم با طول ثابت و استاندارد ارسال شود. (به عبارت دیگر این زمان معادل با طول فریم تقسیم بر نرخ ارسال خواهد بود.) در اینجا فرض را بر آن می گذاریم که تعداد نامحدودی کاربر، مبتنی بر تابع توزیع پواسون و با میانگین N فریم در واحد زمان، فریم های جدید تولید می کنند. (واحد زمان در اینجا زمان لازم برای ارسال یک فریم است). فرض بی نهایت بودن کاربران از آن جهت لازم است که مطمئن باشیم وقتی یک کاربر متوقف و منتظر می شود از تعداد کاربران کاسته نخواهد شد. اگر $N > 1$ باشد مجموع کاربران با نرخ بیشتر از ظرفیت کانال، فریم تولید کرده اند و تقریباً تمام فریمها در اثر تصادم نابود خواهند شد. برای آنکه کارآئی قابل ملاحظه ای داشته باشیم انتظار می رود که $0 < N < 1$ باشد. هر ایستگاه علاوه بر فریمهای جدید خود، فریمهایی را که قبلاً در اثر تصادم خراب شده اند، نیز ارسال می کند.

اجازه بدهید فرض کنیم که احتمال «تلاش برای ارسال k فریم در واحد زمان» نیز از تابع توزیع پواسون با متوسط G تبعیت کند. (به خاطر داشته باشید که واحد زمان در اینجا زمان لازم جهت ارسال یک فریم -Frame Time- است). بدیهی است که $G \geq N$. [زیرا N متوسط تولید فریمهای جدید است در حالیکه G متوسط تولید فریمهای جدید و فریمهای قبلی خراب شده می باشد.]

در بار پائین (یعنی $N \approx 0$) تعداد تصادمها نیز اندک بوده و طبعاً ارسال مجدد فریمها ناچیز خواهد بود لذا $(G \approx N)$ است. در بار بالا تصادمهای زیادی رخ می دهد یعنی $G > N$ است؛ در هر شرایط بار، بازده مفید کانال (یعنی S) مساوی است با حاصل ضرب میزان بار (یعنی G) در احتمال موفقیت در ارسال (یعنی P_0) بنابراین داریم: $S = G \times P_0$ که در آن P_0 احتمال عدم خرابی یک فریم در اثر تصادم است.

فقط وقتی یک فریم در اثر تصادم خراب نخواهد شد که در زمان ارسال آن هیچ فریم دیگری ارسال نشود؛ به شکل ۴-۲ دقت کنید. تحت چه شرایطی فریمی که در شکل بصورت سایه دار نشان داده شده است سالم به مقصد خواهد رسید؟ t را زمان لازم برای ارسال یک فریم در نظر بگیرید. هرگاه کاربر دیگری در فاصله زمانی t_0 تا $t_0 + t$ فریمی را تولید و ارسال کرده باشد انتهای فریم او با ابتدای فریم سایه دار تصادم خواهد کرد. در حقیقت سرنوشت



شکل ۴-۲. دوره آسیب پذیری برای فریم خاکستری.

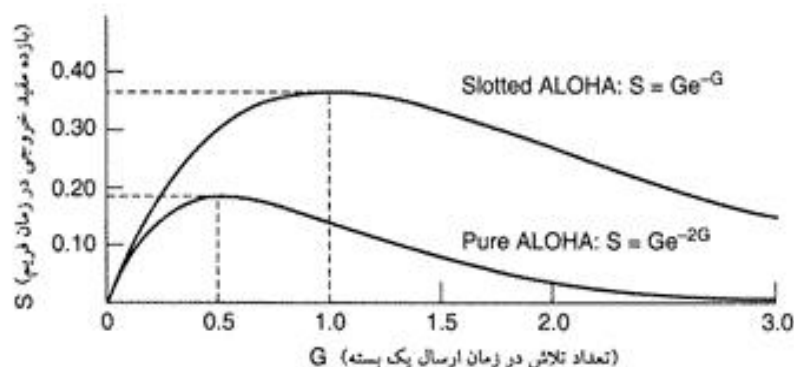
فریم سایه دار به گذشته نیز بستگی دارد، حتی قبل از آنکه اولین بیت آن ارسال شود؛ چراکه در سیستم Pure ALOHA، ایستگاه قبل از شروع به ارسال یک فریم قادر نیست به کانال گوش داده و متوجه شود که فریم دیگری در حال ارسال است. بدلیل مشابه اگر فریم جدیدی در فاصله زمان $t_0 + t$ تا $t_0 + 2t$ ارسال شود با انتهای فریم سایه دار در شکل تصادم خواهد کرد. احتمال آنکه در زمان ارسال یک فریم [یعنی در زمان t] تعداد K فریم تولید شود از تابع توزیع پواسون تبعیت می کند یعنی:

$$\text{رابطه (۴-۲)} \quad \Pr[k] = \frac{G^k \cdot e^{-G}}{k!}$$

بدین ترتیب احتمال ارسال صفر فریم معادل با e^{-G} است. در فاصله زمان ارسال دو فریم [یعنی $2t$] میانگین فریم تولید شده معادل با $2G$ است. احتمال آنکه در طول «زمان آسیب پذیری» یک فریم، هیچ ترافیک دیگری تولید و ارسال نشود مساوی با $P_0 = e^{-2G}$ است. با اعمال رابطه $S = G \times P_0$ به دست خواهیم آورد:

$$S = G \times e^{-2G}$$

در شکل ۴-۳، رابطه بین ترافیک و بازده مفید کانال (Throughput) نشان داده شده است. بیشترین بازده به ازای $G = 0.5$ بدست می آید و در این حالت بازده کانال معادل $S = \frac{1}{2e}$ ، یعنی چیزی حدود ۰.۱۸۴ خواهد بود. به عبارت دیگر، بیشترین بهره مورد انتظار کانال (Channel Utilization) چیزی حدود ۱۸ درصد است. این مقدار بهره کانال چندان جالب نیست ولی در سیستمی که ایستگاهها در زمان دلخواه فریم خود را ارسال می کنند نمی توان انتظار داشت بهره کانال صد درصد باشد. [یعنی هیچ تصادفی اتفاق نیفتد.]



شکل ۴-۳. بازده مفید کانال برحسب ترافیک عرضه شده در سیستم ALOHA.

Slotted ALOHA

در سال ۱۹۷۲، شخصی به نام روبرتز، روشی برای دو برابر کردن ظرفیت مفید سیستم ALOHA ارائه کرد. (مرجع Roberts, 1972) پیشنهاد وی مبنی بر آن بود که زمان به برشهای گسسته‌ای تقسیم شود و هر برش زمان معادل با زمان لازم برای ارسال یک فریم باشد. این روش مستلزم آن بود که کاربران محدوده این برشهای زمانی (Time Slots) را به درستی بدانند. برای رسیدن به چنین هماهنگی می‌توان یک ایستگاه خاص را به خدمت گرفت تا در ابتدای هر برش زمانی سیگنالی همانند سیگنال ساعت منتشر نماید.

در روش روبرتز که امروزه به نام Slotted ALOHA مشهور شده است، برخلاف روش Pure ALOHA هیچ کامپیوتری مجاز نیست وقتی که کلید Enter (Carriage Return) فشار داده شد، داده‌ها را بر روی کانال بفرستد؛ در عوض باید آنقدر منتظر بماند تا به آغاز برش زمان بعدی برسد. بنابراین روش پیوسته Pure ALOHA به روش گسسته Slotted ALOHA تبدیل شده است. از آنجایی که «دوره آسیب‌پذیری» (یعنی زمانی که در خلال آن نباید فریم دیگری ارسال شود) نصف شده است لذا احتمال آنکه هیچ داده دیگری در خلال یک برش (اسلات) تولید نشود e^{-G} خواهد بود؛ بدین ترتیب بدست می‌آوریم:

$$S = G \times e^{-G} \quad \text{رابطه (۳-۴)}$$

به گونه‌ای که از شکل ۳-۴ مشهود است، در سیستم Slotted ALOHA، بهره کانال در $G=1$ به مقدار حداکثر خود می‌رسد و در این نقطه بهره کانال معادل با $S = \frac{1}{e}$ (یعنی حدود 0.368) خواهد بود. اگر این سیستم در شرایط $G=1$ عمل کند احتمال آنکه یک برش زمانی خالی باشد [و بتوان ارسال موفق داشت] حدود 0.368 خواهد بود. (طبق رابطه ۳-۴) بیشترین موفقیتی که می‌توان از Slotted ALOHA انتظار داشت عبارتست از: ۳۷ درصد برای خالی ماندن یک اسلات، ۳۷ درصد برای ارسال موفق و ۲۶ درصد برای تصادم خواهد بود. اگر این سیستم با مقدار بیشتر G کار کند، تعداد برشهای خالی کاهش یافته و میزان تصادمها به صورت نمایی افزایش خواهد داشت. [G: تلاش برای ارسال G عدد فریم در واحد زمان است و واحد زمان نیز زمان لازم برای ارسال یک فریم می‌باشد.] برای آنکه بررسی کنیم افزایش سریع تعداد تصادمها با افزایش G ، از کجا منشأ می‌گیرد ارسال یک فریم آزمایشی را مد نظر قرار بدهید: احتمال آنکه این فریم از تصادم جان سالم به در ببرد (یعنی تمام ایستگاههای دیگر در این برش زمانی ساکت باشند) e^{-G} است و بالطبع احتمال تصادم معادل با $1 - e^{-G}$ خواهد بود. احتمال ارسال موفق فریم، منوط به k تلاش پیاپی خواهد بود (یعنی $k-1$ تصادم متوالی و نهایتاً یک ارسال موفق) یعنی:

$$P_k = e^{-G} \cdot (1 - e^{-G})^{k-1}$$

پس از فشار داده شدن کلید Enter، میانگین دفعات ارسال یعنی E ، معادل است با:

$$E = \sum_{k=1}^{\infty} k \cdot P_k = \sum_{k=1}^{\infty} k \cdot e^{-G} (1 - e^{-G})^{k-1} = e^{-G}$$

در نتیجه، از آنجایی که E به صورت نمایی با G ارتباط دارد اندکی افزایش در بار کانال، بهره کانال را بشدت کاهش خواهد داد.

روش Slotted ALOHA بدلائلی که در بدو امر چندان مشهود و روشن به نظر نمی‌رسد، از اهمیت ویژه‌ای برخوردار است. این روش در دهه ۱۹۷۰ ابداع گردید، در چند سیستم آزمایشی و ابتدائی به کار گرفته شد و پس از آن تقریباً به دست فراموشی سپرده شد ولیکن وقتی دسترسی به اینترنت از طریق کابل اختراع شد، ناگاه این مشکل بزرگ به میان آمد که چگونه می‌توان کانالی مشترک را به کاربران رقیب اختصاص داد. اینجا بود که Slotted ALOHA بار دیگر به صحنه آمد. پروتکل‌هایی بوده‌اند که اگرچه درست و موثر کار می‌کرده‌اند ولی بدلائل سیاسی [غیر علمی] کنار گذاشته شده‌اند (مثلاً برخی از شرکت‌های بزرگ علاقمندند که همه دنباله‌روی عملکرد آنها باشند).

ولیکن سالها بعد اشخاص زیرک و با هوش بدین حقیقت می‌رسند که این پروتکل‌های فراموش شده می‌توانند مشکلات فعلی آنها را حل کنند. به همین دلیل در این فصل به پروتکل‌های زیبا و جالبی خواهیم پرداخت که در حال حاضر کاربرد گسترده‌ای ندارند ولیکن ممکن است در کاربردهای آتی مفید واقع شوند؛ بشرط آنکه طراحان شبکه نسبت به آنها آگاهی داشته باشند. البته ما به بررسی پروتکل‌های متعددی نیز پرداخته‌ایم که در حال حاضر کاربرد بسیار گسترده‌ای دارند.

۲-۲-۴ پروتکل‌های دسترسی چندگانه با قابلیت شنود سیگنال حامل (CSMA)

در روش Slotted ALOHA بیشترین بهره مفیدی که می‌توان بدست آورد $1/c$ [معادل 0.368] است. این بهره چندان جالب نیست چراکه در این روش هر ایستگاه بدون اعتنا به وضعیت بقیه ایستگاه‌ها و به دلخواه ارسال خود را انجام می‌دهد، لذا در این سیستم تعداد تصادمها زیاد خواهد بود. لیکن در شبکه‌های محلی امکان آن وجود دارد که هر ایستگاه بتواند تشخیص بدهد دیگر ایستگاه‌ها چه می‌کنند و بر اساس این تشخیص عملکرد خود را تنظیم نماید. در چنین شبکه‌هایی می‌توان به بهره کانال بسیار بالاتر از $1/c$ دست یافت. در این بخش چندین پروتکل برای افزایش کارایی و بهره کانال معرفی می‌کنیم.

پروتکل‌هایی که در آنها هر ایستگاه به سیگنال حامل روی کانال گوش داده و براساس وضعیت کانال عمل می‌کنند اصطلاحاً «پروتکل‌های شنود حامل» (Carrier Sense Protocols) نامیده می‌شوند. تاکنون تعداد بی‌شماری از این پروتکل‌ها معرفی شده‌اند. کلین‌زاک و توباک (۱۹۷۵) معدودی از این پروتکل‌ها را تحلیل کرده‌اند. در ذیل چندگونه از پروتکل‌های مبتنی بر شنود سیگنال حامل را بررسی خواهیم کرد.

Persistent and Nonpersistent CSMA

اولین پروتکل مبتنی بر شنود سیگنال حامل که در اینجا بررسی خواهیم کرد، روش 1-Persistent CSMA است. در این روش هرگاه یک ایستگاه، داده‌ای برای ارسال داشته باشد ابتدا به کانال گوش می‌دهد تا ببیند آیا در این لحظه کسی دیگری در حال ارسال هست یا خیر. اگر کانال مشغول باشد ایستگاه آنقدر منتظر می‌ماند تا کانال آزاد شود؛ ولی اگر ایستگاه، کانال را آزاد تشخیص بدهد فریم خود را ارسال می‌کند. اگر تصادمی رخ بدهد ایستگاه به اندازه یک زمان تصادفی صبر کرده و تمام مراحل را از نو آغاز می‌نماید. این پروتکل، اصطلاحاً 1-Persistent (پروتکل پافشاری بر ارسال) نامیده می‌شود چراکه وقتی یک ایستگاه کانال را آزاد تشخیص بدهد با احتمال ۱ ارسال خود را آغاز می‌کند. [یعنی اگر ایستگاه کانال را آزاد تشخیص بدهد یقیناً و به صورت غیرمشرط ارسال خود را آغاز می‌نماید. -م]

«تاخیر انتشار» (Propagation Delay) تأثیر بسزایی در کارایی این پروتکل دارد. احتمال ناچیزی وجود دارد که دقیقاً پس از شروع ارسال فریم توسط یک ایستگاه، ایستگاه دیگری نیز آماده ارسال شده و کانال را بررسی و شنود نماید. اگر سیگنال ایستگاه اول هنوز به ایستگاه دوم نرسیده باشد [بدلیل تاخیر انتشار]، دومی نیز کانال را آزاد تشخیص داده و ارسال خود را آغاز می‌کند و طبعاً منجر به تصادم (Collision) خواهد شد. هر چه تاخیر انتشار بیشتر باشد تأثیر مخرب آن بیشتر و منجر به کارایی بدتر پروتکل خواهد شد.

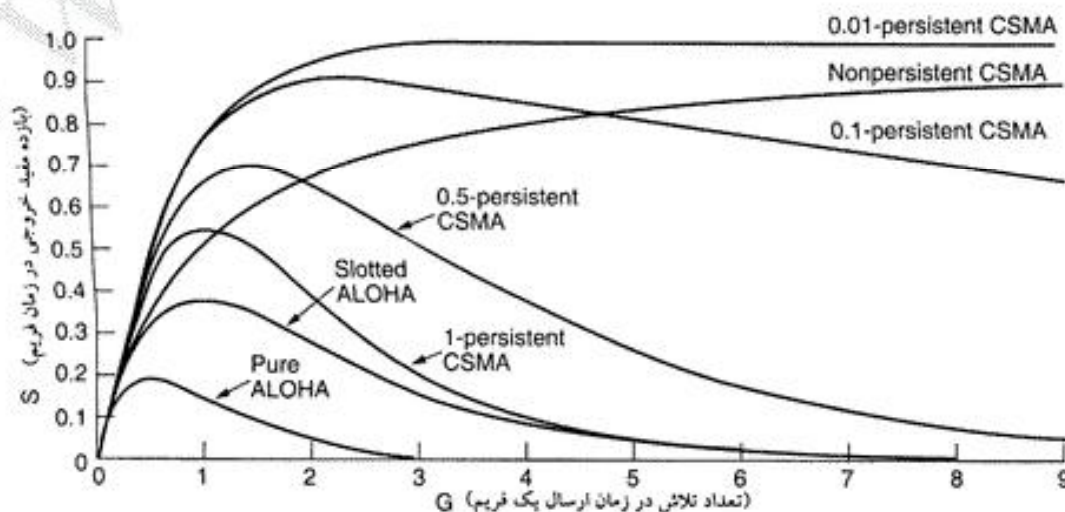
حتی اگر تاخیر انتشار صفر باشد باز هم «تصادم» وجود خواهد داشت: هرگاه دو ایستگاه در خلال ارسال ایستگاه ثالثی آماده ارسال فریم شوند هر دوی آنها مودبانه منتظر خاتمه ارسال فریم جاری می‌شوند و به محض آزاد شدن کانال بطور همزمان ارسال فریم خود را آغاز می‌نمایند که منجر به تصادم خواهد شد. اگر این دو ایستگاه در ارسال فریم خود عجل و مضر نبودند تصادم‌های کمتری رخ می‌داد؛ با این حال این پروتکل بسیار بهتر از Pure ALOHA عمل می‌کند زیرا در این پروتکل دو ایستگاه [یا شنود کانال] از تلاقی با ارسال فریم ایستگاه در

حال ارسال اجتناب می کنند. می توان به صورت ذهنی کارآئی این روش را بسیار بیشتر از Pure ALOHA ارزیابی کرد. بدلیل مشابه، کارآئی این روش از Slotted ALOHA نیز بیشتر است.

دومین پروتکل که آن نیز مبتنی بر شنود سیگنال است Nonpersistent CSMA نام دارد. در این پروتکل، تلاش آگاهانه جهت ارسال بر روی کانال با اصرار کمتری نسبت به پروتکل قبلی انجام می گیرد: هر ایستگاه قبل از ارسال، کانال را بررسی (شنود) می کند؛ اگر کسی دیگر در حال ارسال نباشد ایستگاه، فریم خودش را می فرستد ولیکن هرگاه کانال از قبل در اختیار دیگری باشد، ایستگاه بطور دائم به شنود کانال نخواهد پرداخت. در عوض [هرگاه ایستگاه کانال را مشغول تشخیص بدهد] به اندازه یک زمان تصادفی صبر کرده و پس از آن الگوریتم فوق را تکرار می کند. [تنها تفاوت این روش با روش قبلی در آنست که هرگاه کانال مشغول باشد ایستگاه مترصد آزاد شدن آن باقی نمی ماند و به اندازه یک زمان تصادفی کانال را به حال خود رها می کند و بدان گوش نمی دهد. -م] این الگوریتم طبعاً بهره کانال بهتری را ارائه می دهد [چرا که تصادمها در لحظه آزاد شدن کانال کاهش می یابد] ولیکن تاخیر بیشتری نسبت به روش 1-Persistent CSMA دارد. [تاخیر ارسال]

آخرین پروتکل، p-Persistent CSMA نام دارد. این روش فقط بر روی کانالهای زمان بندی شده (Slotted Time Channels) قابل اعمال است و بدین ترتیب عمل می کند: هرگاه ایستگاهی آماده ارسال شود ابتدا کانال را شنود می نماید؛ اگر کانال آزاد باشد فریم خود را با احتمال p ارسال می کند و یا به احتمال $q=1-p$ ارسال خود را تا فرا رسیدن برش بعدی زمان [اسلات بعدی] به تعویق می اندازد. یعنی حتی اگر یک اسلات خالی باشد ممکن است با احتمال p فریم خود را بفرستد یا با احتمال q به تعویق بیندازد. این فرآیند آنقدر تکرار می شود تا آنکه یا فریم ارسال شود یا آنکه ایستگاهی دیگر ارسال خود را آغاز نماید. در حالت دوم [یعنی ایستگاهی دیگر موفق به ارسال شود] ایستگاه ناموفق، همانند وقتی که تصادم رخ داده عمل می کند یعنی به اندازه یک زمان تصادفی صبر کرده و از نو شروع می نماید. اگر ایستگاه در همان ابتدا کانال را مشغول تشخیص بدهد تا اسلات بعدی صبر می کند و الگوریتم فوق را به اجرا می گذارد.

شکل ۴-۴ منحنی ظرفیت مفید (Throughput) کانال را بر مبنای حجم ترافیک تولید شده، برای سه پروتکل فوق و پروتکل های Pure ALOHA و Slotted ALOHA نشان می دهد.

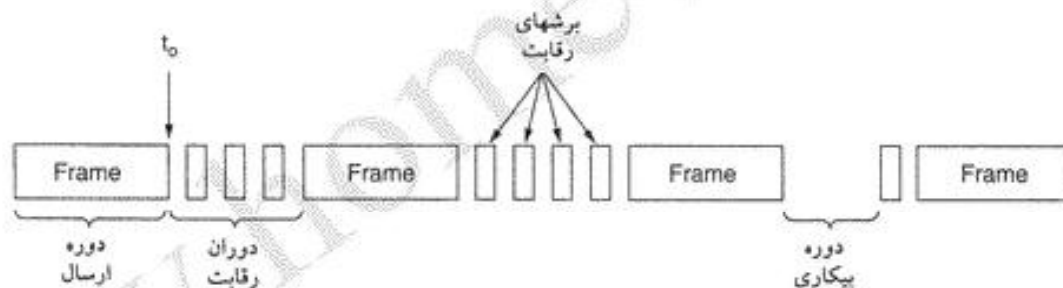


شکل ۴-۴. مقایسه بهره وری کانال (ظرفیت مفید) بر حسب بار برای پروتکل های گوناگون دسترسی تصادفی به کانال.

پروتکل CSMA با تشخیص تصادم

پروتکل‌های Persistent & Nonpersistent CSMA به روشنی بهینه‌تر از ALOHA هستند زیرا مطمئناً اگر ایستگاهی کانال را مشغول تشخیص بدهد ارسال خود را آغاز نخواهد کرد. بهبود دیگر این روشها آنست که ایستگاهها به محض آنکه از وقوع تصادم آگاه شدند ارسال خود را نیمه کاره رها کنند. به عبارت دیگر هرگاه دو ایستگاه، کانال را آزاد احساس کرده و همزمان شروع به ارسال نمایند، تقریباً هر دوی آنها بلافاصله از وقوع تصادم مطلع خواهند شد. در چنین حالتی به محض کشف پدیده تصادم، ایستگاهها بجای ارسال کامل فریمهای آسیب دیده، به ارسال خود خاتمه می‌دهند. قطع سریع ارسال فریمهای آسیب دیده، در زمان و پهنای باند صرفه‌جویی خواهد کرد. چنین پروتکلی که اصطلاحاً CSMA/CD نام دارد بطور گسترده در شبکه‌های محلی به کار گرفته شده است. به ویژه، این پروتکل مبنای شبکه محلی و شناخته شده اترنت است، لذا ارزش آنرا دارد که اندک زمان بیشتری برای بررسی جزئیات آن صرف کنیم.

CSMA/CD نظیر بسیاری از پروتکل‌های دیگر LAN از مدل مفهومی نشان داده شده در شکل ۴-۵ تبعیت می‌کند. در لحظه‌ای که با نماد t_0 مشخص شده، ایستگاهی ارسال فریم خود را به پایان رسانده است. در این لحظه ایستگاههایی که فریمی برای ارسال دارند ممکن است برای ارسال آن تلاش کنند. اگر دو یا چند ایستگاه بطور همزمان تصمیم به ارسال بگیرند تصادم رخ خواهد داد. وقوع تصادم را می‌توان با بررسی توان مصرفی یا اندازه‌گیری و مقایسه پهنای پالس سیگنال دریافتی از کانال و مقایسه آن با سیگنال ارسالی تشخیص داد.



شکل ۴-۵. CSMA/CD می‌تواند در یکی از سه وضعیت: «رقابت»، «ارسال» یا «بیکار» قرار داشته باشد.

پس از آنکه یک ایستگاه متوجه وقوع تصادم شد، ارسال خود را ناتمام رها کرده و از نو شروع می‌کند؛ (البته با این فرض که ایستگاه دیگری در این بینابین ارسال خود را آغاز ننماید). بدین ترتیب مدل ارائه شده برای CSMA/CD، شامل: (۱) چندین مرحله متناوب «رقابت» (Contention) (۲) بازه‌های ارسال و (۳) بازه‌های بیکاری خواهد بود. (بازه‌های بیکاری بدین معناست که تمام ایستگاهها بدلیل عدم نیاز به ارسال ساکت بوده‌اند). حال جزئیات الگوریتم رقابت را دقیقتر بررسی می‌نماییم: فرض کنید دو ایستگاه بطور همزمان ارسال فریم خود را در زمان t_0 آغاز نمایند. چقدر طول می‌کشد تا متوجه شوند تصادم اتفاق افتاده است؟ پاسخ این سوال برای تعیین طول زمان رقابت و همچنین محاسبه تاخیر و ظرفیت مفید کانال حیاتی است.

حداقل زمان لازم برای تشخیص وقوع تصادم، معادل زمانی است که طول می‌کشد تا سیگنال ارسالی از یک ایستگاه به ایستگاه دیگر منتشر شود. براساس این استدلال شاید شما تصور کنید که یک ایستگاه تا پس از زمانی معادل با زمان انتشار سیگنال بر روی کل کابل از وقوع تصادم مطلع نشده و از تصرف کانال، مطمئن نخواهد بود. (منظورمان از «تصرف» کانال آنست که ایستگاههای دیگر متوجه شوند که او در حال ارسال است و تداخلی پیش نیاید). این نتیجه‌گیری صحیح نیست. به سناریوی زیر که در بدترین شرایط در نظر گرفته شده، دقت نمایید: زمان

لازم برای انتشار سیگنال بین دورترین ایستگاه ها از یکدیگر را τ فرض کرده ایم؛ در لحظه t_1 یک ایستگاه ارسال خود را آغاز می کند. در لحظه $t = t_1 + \tau$ یعنی دقیقاً قبل از لحظه ای که سیگنال به دورترین ایستگاه می رسد، آن ایستگاه شروع به ارسال می کند. مشخصاً این ایستگاه وقوع تصادم را کشف می کند ولیکن سیگنال نویزی که در اثر تصادم تولید می شود تا زمان $t = t_1 + 2\tau$ به ایستگاه مبداء باز نخواهد گشت. به عبارت دیگر در بدترین حالت، تا انتقضای زمان 2τ پس از شروع ارسال، ایستگاه مبداء نمی تواند از تصرف کانال مطمئن باشد. بهمین دلیل ما بازه رقابت را همانند روش Slotted ALOHA مدل کرده ایم که در آن پهنای هر برش زمانی (اسلات) معادل 2τ می باشد. بر روی یک کابل کوآکسیال به طول ۱ کیلومتر، 2τ حدوداً معادل ۵ میکروثانیه است. برای سادگی فرض را بر آن خواهیم گذاشت که هر برش زمان در برگیرنده تنها یک بیت [به طول 2τ] است و به محض آنکه کانال تصرف شد، ایستگاه می تواند با هر سرعت دلخواه ارسال خود را انجام بدهد ولیکن بدیهی است که با ارسال با نرخ $1 \text{ bit}/2\tau$ نخواهد بود!

درک این موضوع که فرآیند کشف تصادم به صورت آنالوگ انجام می شود اهمیت دارد. سخت افزار هر ایستگاه باید در حین ارسال به کابل گوش بدهد. اگر آنچه را که از کابل بازخوانی می کند با آنچه بر روی کابل قرار داده، متفاوت باشد متوجه وقوع تصادم می شود. کشف تصادم مستلزم آنست که روش کدینگ سیگنال این امکان را فراهم بیاورد (زیرا مثلاً تصادم دو سیگنال صفر ولت قابل کشف نیست). به همین دلیل معمولاً از روشهای کدینگ خاص [مثل منچستر] استفاده می شود.

بدیهی است که ایستگاه فرستنده باید بطور مدام بر کانال نظارت کند و منتظر شنیدن سیگنال نویزی که مشخص کننده وقوع تصادم است باقی بماند. بهمین دلیل CSMA/CD با یک کانال منفرد ذاتاً سیستمی Half Duplex (دو طرفه غیرهمزمان) است. در چنین سیستمی برای ایستگاه این امکان وجود ندارد که ارسال و دریافت فریمها را بطور همزمان انجام بدهد زیرا بخش گیرنده آن حتی در خلال ارسال مشغول و در حال پیگیری بروز تصادم بر روی کانال است. البته در حین ارسال، گیرنده فقط آماده کشف تصادم به صورت آنالوگ است و داده های روی کانال، دریافت یا ذخیره نمی شوند.

لازم است بدین نکته بدیهی اشاره کنیم که پروتکل زیرلایه MAC دریافت مطمئن فریمها را تضمین نمی کند چراکه اگر حتی تصادمی رخ ندهد گیرنده فریم ممکن است بدلائل گوناگون نتواند فریم را به درستی دریافت کند. (دلائلی مثل فقدان فضای بافر یا وقفه های گمشده)

۳-۲-۴ پروتکل های بدون تصادم

اگرچه در روش CSMA/CD پس از آنکه یک ایستگاه بطور مطمئن کانال را تصرف کرد دیگر وقوع تصادم متفی است ولیکن وقوع تصادمهای مکرر در دوره رقابت^۱ کاملاً طبیعی است. این تصادمها تاثیر زیانباری بر روی کارآئی سیستم خواهد داشت؛ بالاخص زمانی که کابل، طولانی (و در نتیجه τ یا همان تاخیر انتشار بالاست) و یا فریمها کوتاه هستند، این تاثیر بسیار مخرب است. در چنین حالاتی روش CSMA/CD قابل استفاده نخواهد بود. در این بخش چند پروتکل را بررسی می کنیم که در آنها مسئله رقابت حل شده و بروز تصادم حتی در زمان رقابت متفی است. اگرچه بسیاری از این روشها در حال حاضر بر روی سیستمهای شناخته شده به کار گرفته نشده است ولیکن در اختیار داشتن پروتکل های با ویژگیهای عالی، برای به کارگیری در سیستمهای پیشرفته آینده یک حسن محسوب می شود!

در پروتکل های که بررسی می شوند فرض را بر آن گذاشته ایم که دقیقاً N ایستگاه به کانال مشترک متصلند و

^۱ Contention Period

هر ایستگاه آدرسی بین 0 تا $N-1$ دارد که درون سخت افزار ایستگاه حک شده است. این موضوع که ممکن است برخی از ایستگاهها در برخی از مواقع غیر فعال و خاموش باشند چندان اهمیت ندارد. همچنین فرض کرده ایم که تاخیر انتشار قابل صرف نظر باشد. مسئله بنیادی هنوز باقی است: «پس از آنکه ارسال فعلی یک ایستگاه به پایان رسید کدامین ایستگاه حق دارد کانال را در اختیار بگیرد و فریم خود را ارسال نماید؟» در این بخش نیز از مدل شکل ۴-۵ با برشهای گسسته رقابت^۱ استفاده می کنیم.

یک پروتکل مبتنی بر نشانه های بیتی (Bit Map)

در اولین پروتکل بدون تصادم که Basic Bit-Map Method نام دارد بازه رقابت متشکل از دقیقاً N برش زمانی است. هرگاه ایستگاه شماره صفر، فریمی آماده ارسال داشته باشد در برش (اسلات) شماره صفر، بیت ۱ را بر روی کانال می گذارد؛ هیچ ایستگاه دیگری حق ندارد در این برش چیزی بر روی کانال بگذارد. فارغ از آنکه ایستگاه صفر چه کاری می کند ایستگاه شماره ۱ نیز این فرصت را دارد که در برش شماره ۱ با قرار دادن بیت ۱ بر روی کانال تقاضای ارسال فریم بدهد. (البته اگر فریمی برای ارسال آماده داشته باشد). بطور کلی ایستگاه شماره i حق دارد در صورتی که فریمی جهت ارسال داشته باشد در برش شماره i با گذاشتن بیت ۱ بر روی کانال، تقاضای خود را اعلام نماید. پس از آنکه کل N برش رقابت سپری شد تمام ایستگاهها فهرست کاملی از ایستگاههای متقاضی ارسال، در اختیار دارند. پس از این لحظه، ایستگاههای متقاضی ارسال، به ترتیب شماره، ارسال فریمهای خود را آغاز می کنند. (شکل ۴-۶ را ملاحظه کنید)



شکل ۴-۶. پروتکل پایه مبتنی بر نشانه های بیتی (Basic Bitmap Protocol).

از آنجایی که تمام ایستگاهها در هر لحظه می دانند که چه ایستگاهی حق ارسال دارد به هیچ وجه تصادم رخ نخواهد داد. پس از آنکه آخرین ایستگاه، فریم خود را ارسال کرد، (رخدادی که همه ایستگاهها پسادگی می توانند متوجه آن شوند) مجدداً دوره رقابت، شامل N برش (با عبارتی N بیت) شروع می شود. هرگاه ایستگاهی پس از گذشت برش (اسلات) متعلق به او، آماده ارسال شود فرصت را از دست داده و بایستی خاموش بماند تا دیگر ایستگاهها شانس خود را آزموده و کارشان را انجام بدهند تا دور بعدی رقابت فرا برسد. پروتکلهایی همانند این روش که در آنها هر ایستگاه تقاضای خود را قبل از ارسال و به صورت فراگیر به اطلاع همه می رساند اصطلاحاً پروتکل های رزرو سازی (Reservation) نامیده می شوند.

اجازه بدهید مختصراً به کارآئی این پروتکل بپردازیم. برای سادگی، واحد زمان را معادل طول بیت هر برش رقابت در نظر گرفته ایم و طول هر فریم داده را معادل d واحد زمان فرض کرده ایم. [یعنی هر یک از برشهای رقابت را یک واحد و بر این اساس، طول فریم را d واحد زمان، تلقی کرده ایم.] در شرایطی که بار ایستگاهها پائین است، برشهای رقابت بطور متوالی تکرار می شوند و بدلیل عدم تقاضا برای ارسال فریم مکرراً خالی می مانند. حال وضعیت را از دیدگاه یک ایستگاه با شماره پائینی مثل صفر یا یک بررسی می کنیم. [بدون آنکه به عمومیت استدلال لطمه ای بخورد] وقتی چنین ایستگاهی آماده ارسال می شود باید تا رسیدن برش متناظر با شماره خودش صبر کند

در حالیکه ممکن است شماره برش فعلی یکی از شماره های میانی باشد. بطور «میانگین» ایستگاه مجبور است به اندازه $N/2$ صبر کند تا دور فعلی خاتمه یافته و در برش متعلق به خودش تقاضا بدهد؛ سپس باید اندازه N برش صبر کند تا تمام برشها به ترتیب بگذرند و او بتواند ارسال خود را آغاز کند. [چون نوبت او گذشته بطور میانگین باید $N/2$ برش دیگر صبر کند تا زمان ارسال او فرا برسد. بنابراین بطور میانگین زمانی معادل $1.5N$ معطل می شود.] چشم انداز شانس ایستگاه های با شماره بالا روشتر است. عموماً چنین ایستگاه هایی بطور میانگین مجبورند قبل از شروع به ارسال، $N/2$ برش صبر کنند تا نوبت به اعلام تقاضا و سپس ارسال آنها برسد. ایستگاه های با شماره بالا به ندرت مجبورند که یک دور کامل صبر کنند تا دور بعدی فرا برسد. از آنجایی که ایستگاه های با شماره پائین باید بطور میانگین $1.5N$ و ایستگاه های با شماره بالا باید $0.5N$ منتظر بمانند لذا میانگین این دو زمان N خواهد شد. با این استدلال، محاسبه کارآئی کانال در بار پائین ساده است: میزان سرباری که باید برای هر فریم d بیت متحمل شد N بیت است بنابراین این کارآئی کانال در بار پائین عبارتست از: $d/(N + d)$

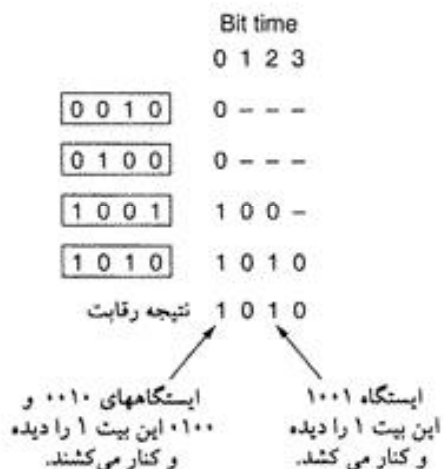
در بار بالا، یعنی وقتی تمام ایستگاه ها چیزی برای ارسال داشته باشند، N برش رقابت برای ارسال N تا فریم متوالی صرف می شود و این سربار به ازای هر فریم ۱ بیت خواهد بود [یعنی N بیت سربار بر روی N فریم d بیتی سرشکن می شود لذا به ازای هر فریم d بیت فقط یک بیت سربار تحمیل می شود]. بنابراین کارآئی کانال در بار بالا عبارتست از: $d/(d+1)$. میانگین تاخیر ارسال هر فریم نیز، معادل با زمان تاخیر در صف داخلی هر ایستگاه به اضافه مقدار $N.(d + 1)/2$ (معادل تاخیر ارسال پس از آنکه فریم به سر صف داخلی ایستگاه می رسد) خواهد بود.

روش شمارش دودویی معکوس (Binary Countdown)

مشکل پروتکل Bitmap آنست که به ازای هر ایستگاه یک بیت سربار تحمیل می شود فلذا شبکه قابل گسترش به مقیاس هزاران ایستگاه، نخواهد بود. می توان براساس آدرس دودویی ایستگاه ها، فرآیند رزرو سازی را به روش بهتری انجام داد: ایستگاهی که می خواهد کانال را به خدمت بگیرد شماره آدرس خود را به صورت دنباله ای از بیتها که از پرارزشترین آن شروع می شود بر روی کانال منتشر می کند. فرض بر آنست که تمام آدرسها دارای طول یکسانی هستند. هر یک بیتهای آدرس پس از انتشار بر روی کانال بطور منطقی با یکدیگر (Wired OR) OR می شوند. این روش که «پروتکل شمارش دودویی معکوس» نام دارد برای اولین بار در سیستم Datakit (Fraser, 1987) به کار گرفته شد. در این روش به صراحت فرض شده که تاخیر انتشار خط ناچیز است و تمام ایستگاه ها می توانند بطور همزمان بیتهای ارسالی را بر روی کانال را بشنوند.

برای پیشگیری از هرگونه برخورد، باید یک «قاعده داوری» (Arbitration Rule) اعمال شود: به محض آنکه ایستگاهی متوجه شود که بیت پرارزش صفر او که بر روی کانال قرار گرفته با بیت ۱ بازنویسی شده از دور رقابت کنار می کشد. به عنوان مثال اگر ایستگاه های 0100، 0010، 1001 و 1010 بخواهند کانال را بدست بیاورند، همگی در اولین برش زمانی، بیت پرارزش خود را بر روی کانال می گذارند، یعنی به ترتیب بیتهای 0، 0، 1 و 1 بر روی کانال ظاهر می شود. این بیتها با یکدیگر Wired OR شده و نتیجه آن ۱ خواهد بود. ایستگاه های 0010 و 0100 متوجه ظهور ۱ بر روی خط شده و نتیجه می گیرند که ایستگاهی با شماره بالاتر برای تصرف کانال رقابت می کند، لذا از دور فعلی خارج می شوند. در ادامه فقط ایستگاه های 1001 و 1010 به رقابت می پردازند.

بیت بعدی هر دو صفر است فلذا این دو ایستگاه باز هم ادامه می دهند. بیت بعدی روی کانال ۱ خواهد بود لذا ایستگاه 1001 کنار می کشد. در نهایت ایستگاه 1010 برنده این رقابت خواهد بود چرا که دارای بالاترین شماره آدرس است. پس از پیروزی در این رقابت (که به صورت مزایده برگزار شده)، ایستگاه برنده می تواند فریم خود را ارسال کند و پس از آن دور بعدی «مزایده» آغاز می شود. عملکرد این پروتکل در شکل ۴-۷ به تصویر کشیده شده



شکل ۴-۷. پروتکل شمارش دودویی معکوس. خط تیره علامت سکوت ایستگاه است.

است. پروتکل فوق دارای این ویژگی است که ایستگاه‌های با شماره بالاتر اولویت بیشتری نسبت به ایستگاه‌های با شماره پائینتر دارند؛ این ویژگی براساس زمینه و نوع کار می‌تواند خوب یا بد باشد. کارآئی کانال در این روش معادل $d/(d + \log_2 N)$ است. با اینحال اگر قالب فریم بگونه‌ای زیرکانه انتخاب شود که آدرس فرستنده فریم، همان اولین فیلد باشد مقدار سریار $\log_2 N$ بیت نیز تلف نخواند شد و کارآئی کانال صددرصد است!

دو پژوهشگر به نامهای «مارک» و «وارد» (۱۹۷۹)، گونه‌ای از روش «شمارش دودویی معکوس» را با استفاده از واسطه‌های موازی (Parallel Interfaces) به جای واسطه‌های سریال معرفی و تشریح کردند. [یعنی ارسال اطلاعات به جای سریال به صورت موازی انجام می‌شود.] آنها پیشنهاد کردند که برای آدرس دهی ایستگاه‌ها، [به جای شماره‌های ثابت] از شماره‌های مجازی استفاده شود. پس از آنکه ایستگاهی موفق به ارسال شد، به شماره تمام ایستگاه‌های قبل از آن، تا شماره صفر، یک واحد اضافه می‌شود تا در مرحله بعد، آنهایی که موفق به ارسال نشده‌اند اولویت بالاتری داشته باشند. بدین ترتیب مسئله قبضه شدن کانال توسط یک ایستگاه حل خواهد شد. -م]

بعنوان مثال فرض کنید ایستگاه‌های C, D, H, A, G, B, E و F به ترتیب دارای اولویتهای ۷, ۶, ۵, ۴, ۳, ۲, ۱ و ۰ هستند و [از بین ایستگاه‌های متقاضی مثل A, B و F]، D موفق به ارسال فریم خود شده است. پس از ارسال، D به انتهای این فهرست رفته و ترتیب اولویتها به صورت C, D, H, A, G, B, E و F تغییر می‌کند. بدین ترتیب C به صورت مجازی ایستگاه شماره ۷ باقی می‌ماند ولی شماره A از ۴ به ۵ صعود و شماره D، از ۵ به صفر سقوط می‌کند. حال ایستگاه D فقط در صورتی قادر به در اختیار گرفتن کانال خواهد شد که هیچ ایستگاه دیگری بدان احتیاج نداشته باشد.

روش شمارش دودویی معکوس نمونه‌ای است از یک پروتکل ساده، جالب و کارآمد که در انتظار کشف مجدد، روزگار می‌گذراند و در آینده، منزلتی را برای خود خواهد یافت!!

۴-۲-۴ پروتکل‌های با رقابت محدود

تاکنون دو استراتژی بنیادی را برای دستیابی به کانال در شبکه‌های مبتنی بر کابل، بررسی کرده‌ایم: روشهای رقابت و تصادم مثل CSMA و روشهای بدون تصادم. هر کدام از این استراتژیها را می‌توان بر اساس دو معیار مهم رده‌بندی کرد: (۱) میزان تاخیر در بار پائین (۲) کارآئی و بهره کانال در بار بالا.

در بار پایین روشهای مبتنی بر رقابت (مثل ALOHA یا CSMA) ارجحتر است چراکه تاخیر ناچیزی دارند؛ ولیکن وقتی بار افزایش می یابد روشهای مبتنی بر رقابت ارزش خود را از دست می دهند زیرا سر بار تحمیل شده در اثر مبارزه بر سر بدست آوردن کانال، به شدت افزایش می یابد. دقیقاً عکس این موضوع در مورد پروتکل های بدون تصادم صادق است: در بار پائین تاخیر بالایی دارند ولی وقتی بار شبکه افزایش می یابد (برعکس پروتکل های مبتنی بر رقابت)، کارآئی کانال رو به افزایش می گذارد.

روشن است که اگر بتوانیم دو ویژگی ممتاز این روشها را با هم ترکیب کنیم به یک پروتکل مطلوب خواهیم رسید که در بار پائین بروش رقابت عمل می کند (تا تاخیر کمی داشته باشد) ولی در بار بالا از روش بدون تصادم بهره می گیرد (تا کارآئی کانال افزایش یابد). این پروتکلها به نام «پروتکل های با رقابت محدود» مشهور هستند و بررسی آنها، مطالعات ما را در مورد شبکه های مبتنی بر شنود سیگنال، به نتیجه نهانی می رساند.

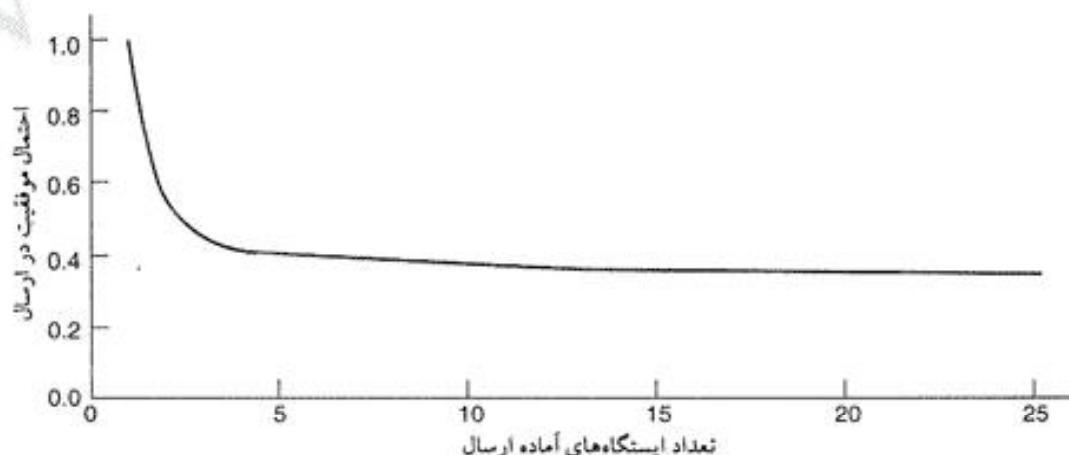
پروتکل های مبتنی بر رقابت که تاکنون بررسی کرده ایم «متقارن» (Symmetric) هستند بدین معنا که هر کدام از ایستگاه ها، با احتمال p تلاش می کند کانال را در اختیار بگیرد و برای تمام ایستگاه ها، این احتمال یکسان و مساوی p فرض شده است. می توان کارآئی کل سیستم را با انتساب احتمالات مختلف به ایستگاه های متفاوت، بهبود داد. قبل از آنکه به پروتکل های نامتقارن (Asymmetric) بپردازیم اجازه بدهید تا مروری اجمالی بر کارآئی شبکه در حالت متقارن داشته باشیم. فرض کنید تعداد ایستگاه برای دسترسی به کانال رقابت می کنند و هر کدام در ابتدای یک برش زمان، با احتمال p اقدام به ارسال می نمایند. احتمال آنکه یکی از ایستگاه ها موفق به در اختیار گرفتن کانال در یکی از برشهای زمانی شود عبارتست از:

$$k.p.(1-p)^{k-1}$$

به منظور پیدا کردن مقداری بهینه برای p [به نحوی که احتمال فوق به حداکثر برسد] از رابطه فوق بر حسب p مشتق گرفته و حاصل را مساوی صفر قرار می دهیم؛ سپس p را محاسبه می نماییم. پس از محاسبه، بهترین مقدار p معادل $1/k$ بدست می آید. با قرار دادن $1/k$ به جای p در رابطه فوق خواهیم داشت:

$$P_s[\text{موفقیت در ارسال با بالاترین احتمال}] = \left[\frac{k-1}{k}\right]^{k-1} \quad \text{رابطه (۴-۴)}$$

منحنی این رابطه در شکل ۴-۸ ترسیم شده است. اگر تعداد ایستگاه ها کم باشد، احتمال موفقیت بالاست ولی به محض آنکه تعداد ایستگاه ها به ۵ یا بیشتر می رسد، احتمال موفقیت در ارسال، به مقدار تقریباً ثابت $1/e$ میل می کند.



شکل ۴-۸ منحنی احتمال موفقیت در تصرف کانال برای یک کانال متقارن.

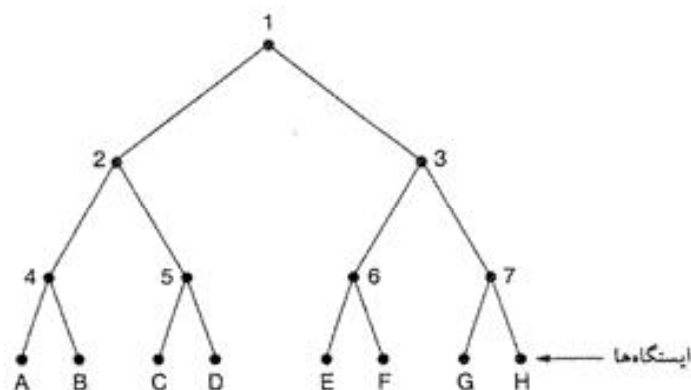
از شکل ۴-۸، بخوبی مشهود است که می توان احتمال موفقیت ایستگاه ها را با کاهش دادن حجم رقابت، افزایش داد؛ «پروتکل های با رقابت محدود» سعی می کنند دقیقاً همین کار را انجام بدهند. این پروتکلها ابتدا ایستگاه ها را به چند گروه تقسیم می نمایند. (لزومی به از هم جدا بودن گروه ها نیست.) در برش زمانی شماره صفر، فقط اعضای گروه شماره صفر حق شرکت در رقابت را دارند. اگر یکی از آنها موفق شد، کانال را در اختیار می گیرد و فریم خود را ارسال می نماید. اگر برش زمانی شماره صفر خالی ماند یا تصادم رخ داد، اعضای گروه شماره ۱ در برش شماره ۱ رقابت می کنند و کار به همین ترتیب ادامه می یابد. با تقسیم بندی صحیح و مناسب ایستگاه ها به چند گروه، میزان رقابت در هر برش زمانی کاهش یافته و طبق منحنی شکل ۴-۸، شبکه در نزدیکی سمت چپ نمودار کار می کند.

نکته ظریف در اینجا است که ایستگاه ها را چگونه به برش های زمانی مستقل منتسب نماییم. [یعنی تعیین آنکه چه ایستگاه هایی در چه برش زمانی رقابت کنند. -م] یکی از حالات ویژه آنست که هر گروه صرفاً یک عضو بیشتر نداشته باشد. چنین انتسابی تضمین کننده آنست که هیچ تصادمی بوجود نخواهد آمد چرا که برای تصاحب هر برش زمانی فقط یک ایستگاه رقابت می کند. با چنین پروتکل هایی قبلاً برخورد کرده ایم (مثل روش شمارش دودویی معکوس). حالت خاص دیگر آنست که در هر گروه دو ایستگاه قرار بگیرد. احتمال آنکه هر دوی آنها بطور همزمان و در یک برش زمانی بخواهند ارسال داشته باشند معادل p^2 است که برای مقادیر کوچک p ، بسیار ناچیز خواهد بود. هر چه تعداد ایستگاه ها در هر گروه بیشتر باشد احتمال تصادم در برش های زمانی متعلق به آن گروه افزایش خواهد یافت ولی در عوض تعداد برش های زمانی لازم برای رقابت گروه ها کاهش می یابد. در محدودترین حالت تمام ایستگاه ها در یک گروه واحد قرار می گیرند؛ در این حالت عملکرد آن همانند روش Slotted ALOHA خواهد بود. بالطبع به روشی نیاز داریم که برش های زمانی (Time Slots) را بطور پویا به ایستگاه ها اختصاص بدهد یعنی وقتی بار کم است تعداد زیادی ایستگاه در یک برش رقابت کنند ولیکن وقتی بار بالاست تعداد کمی ایستگاه (حتی یک ایستگاه) در هر گروه رقابت نمایند.

پروتکل پیمایش و فقی درخت (Adaptive Tree Walk)

یکی از روش های بسیار ساده برای انجام عمل انتساب فوق الذکر، الگوریتمی است که توسط ارتش ایالات متحده در جنگ جهانی دوم برای آزمایش سربازان و تشخیص بیماری سیفلیس ابداع شد. (Dorfman, 1943) ارتش، نمونه خون N سرباز را می گرفت و بخشی از این نمونه های خون، درون یک لوله آزمایش واحد ریخته و با هم مخلوط می شد. سپس این نمونه مخلوط شده، مورد آزمایش قرار می گرفت و اگر هیچ آنتی بادی، در آن پیدا نمی شد تمام سربازان سالم تشخیص داده می شدند. در صورت تشخیص آلودگی در نمونه خون، دو نمونه دیگر از خونها تهیه می شد: نمونه اول شامل مخلوطی از نمونه خون سربازان ۱ تا $N/2$ و نمونه دومی از خون مابقی افراد. این فرآیند آنقدر تکرار می شد تا سربازان آلوده مشخص شوند.

برای نسخه کامپیوتری این الگوریتم (Capetanakis, 1979)، ساده تر آنست که ایستگاه ها را به عنوان برگ های یک درخت دودویی فرض کنید (همانند آنچه که در شکل ۴-۹ می بینید). در اولین برش رقابت، (یعنی پس از خاتمه ارسال یک ایستگاه و آزاد شدن کانال) تمام ایستگاه ها مجازند برای در اختیار گرفتن کانال رقابت کنند. اگر تصادمی به وقوع پیوست، در برش شماره ۱، فقط ایستگاه هایی که در پوشش گره ۲ از درخت واقعتاً به رقابت می پردازند. اگر یکی از آنها کانال را تصاحب و فریم خود را ارسال کرد، برش زمان بعدی، برای رقابت ایستگاه های تحت پوشش گره ۳ در نظر گرفته می شود ولیکن اگر بیش از یکی از ایستگاه های تحت پوشش گره ۲، بخواهند ارسال داشته باشند در همان برش شماره ۱ تصادم رخ می دهد و در برش شماره ۲ نوبت به رقابت ایستگاه های گره ۴ خواهد بود.



شکل ۹-۴. یک درخت برای هشت ایستگاه.

هرگاه در برش شماره تصادمی رخ بدهد، کل درخت به صورت «عمقی»^۱ (Depth First) پیمایش می شود تا ایستگاه‌ها به ترتیب شناسایی و تعیین موقعیت شده و ارسال خود را انجام بدهند. هر برش رقابت، به یک گره خاص در درخت تعلق دارد. اگر تصادمها تکرار شود، پیمایش و جستجو از چپ به راست و به صورت بازگشتی (Recursive) ادامه می یابد. اگر یکی از برشها خالی بماند یا فقط یک ایستگاه، بدون تصادم کانال را صاحب شود جستجو در آن گره خاتمه می یابد، چراکه تمام ایستگاه‌های آماده ارسال در آن گره، مشخص شده اند. (اگر بیش از یک ایستگاه در آن گره تمایل به ارسال می داشتند تصادم رخ می داد).

وقتی بار شبکه بالا است به ندرت اتفاق می افتد که مسئله تخصیص کانال در همان برش رقابت شماره صفر که متعلق به گره ۱ است حل شود زیرا به احتمال زیاد بیش از یک ایستگاه آماده ارسال هستند. بدلیل مشابه این مسئله در برش ۲ و ۳ نیز حل نخواهد شد و رقابت به مراحل بعدی خواهد کشید. سوال آنست که بطور کلی، جستجو از چه سطحی آغاز شود؟ روشن است که هر چه بار سنگینتر باشد، جستجو باید از سطوح پائین تر شروع شود. فرض را بر آن خواهیم داشت که هر ایستگاه تخمین خوبی از تعداد کل ایستگاه‌های آماده ارسال (که آنرا q می نامیم) در اختیار دارد و این تخمین را به روشی مثل نظارت بر ترافیک جاری شبکه بدست آورده است.

در ادامه اجازه بدهید سطوح درخت را شماره گذاری کرده و رأس آن یعنی گره شماره ۱ را سطح صفر بنامیم. بدین ترتیب، گره ۲ و ۳ در سطح یک قرار می گیرند و شماره گذاری سطوح بهمین نحو ادامه می یابد. دقت کنید که هر گره در سطح i ، کسر $\frac{1}{2^i}$ از کل ایستگاه‌های شبکه را در بر می گیرد. اگر q ایستگاه آماده ارسال، بطور یکنواخت در گره‌ها توزیع شده باشند، میانگین تعداد ایستگاه‌های آماده ارسال که تحت پوشش گرهی خاص در سطح i قرار دارند، معادل $2^{-i}q$ خواهد بود. بطور حسی می توان انتظار داشت که سطح بهینه ای که جستجو باید از آن سطح آغاز شود همان سطحی است، تعداد متوسط ایستگاه‌های آماده در زیر هر گره ۱ باشد، یعنی همان سطحی که در آن $2^{-i}q = 1$ است. با حل این معادله بدست می آوریم: $i = \log_2 q$.

دو پژوهشگر بنامهای Gallager و Bertsekas (۱۹۹۲)، نسخه های متعدد و پیشرفته تری از الگوریتم فوق را ابداع و جزئیات آنها را تشریح کرده اند. به عنوان مثال، حالتی را در نظر بگیرید که در آن ایستگاه‌های G و H [تحت پوشش گره ۷] تنها ایستگاه‌های آماده ارسال هستند. در گره ۱ تصادم رخ خواهد داد، در حالیکه وقتی در برش شماره ۲ نوبت به رقابت اعضای گره ۲ می رسد، آن برش خالی خواهد ماند. مجدداً در حین آزمایش گره ۳ تصادم پیش می آید؛ (تا اینجا می دانیم که دو یا چند ایستگاه در گره ۱ تمایل به ارسال دارند ولی قطعاً در گره ۲ واقع

۱. در مبحث ساختمان داده و الگوریتمها، پیمایش عمقی روشی برای پیمایش درخت محسوب می شود. م

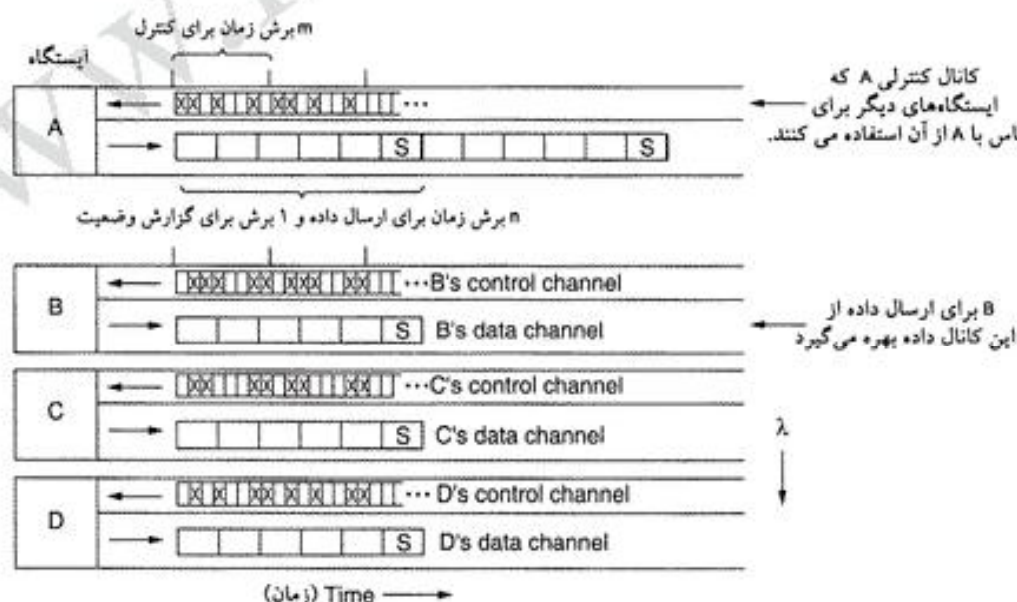
نشده‌اند). آزمایش گره ۳ ناکام مانده و به آزمون گره ۶ می‌انجامد. باز هم در آزمایش گره ۷ تصادم رخ می‌دهد و عاقبت در آخرین تلاش G موفق به ارسال خواهد شد. دو پژوهشگر فوق برای حل مشکلات این چنینی، راهکارهایی را معرفی کرده‌اند.

۵-۲-۴ پروتکل‌های دسترسی چندگانه مبتنی بر تقسیم طول موج^۱

یک راهکار متفاوت برای تخصیص کانال آنست که آثرا بروشی مثل FDM یا TDM (یا تلفیقی از هر دو) به چندین «زیرکانال» تقسیم کرده و آنها را در صورت نیاز به صورت پویا به ایستگاه‌ها تخصیص بدهیم. چنین الگویی، به طور رایج بر روی شبکه‌های محلی با فیبرنوری کاربرد دارد تا با استفاده از طول موجهای مختلف (یعنی فرکانسهای متفاوت)، امکان محاوره و ارتباط همزمان ایستگاه‌ها فراهم آید. در این بخش یکی از این پروتکل‌ها را بررسی خواهیم کرد. (Humblot et al., 1992)

ساده‌ترین راه برای ساختن یک شبکه محلی کاملاً نوری، آنست که شبیه به شکل ۱۰-۲، از ترویج کننده غیرفعال با توپولوژی ستاره استفاده نمائیم. در حقیقت دو فیبرنوری منشعب شده از هر ایستگاه به یک استوانه شیشه‌ای وارد شده‌اند. [در این استوانه تمام پرتوهای نوری با هم ممزوج شده، در هم می‌آمیزند. م] یکی از تارهای فیبر نوری بعنوان خروجی و دیگری بعنوان ورودی ایستگاه از این استوانه منشعب می‌شوند. پرتوی نور خارج شده از یک ایستگاه، در درون استوانه منتشر شده و در ورودی تمام ایستگاهها قابل دریافت و تشخیص است. شبکه‌های نوری غیر فعال با توپولوژی ستاره می‌توانند صدها ایستگاه داشته باشند.

برای آنکه ارسال همزمان چندین ایستگاه میسر باشد. طیف نوری به چندین کانال (باند‌های مختلف با طول موج متفاوت) تقسیم می‌گردد. (شکل ۳۱-۲ را ببینید). در این پروتکل که WDMA (دسترسی چندگانه مبتنی بر تقسیم طول موج) نامیده شده به هر ایستگاه دو کانال متناسب می‌شود: یک کانال با پهنای باند باریک که از آن به عنوان کانال کنترل، جهت هماهنگی (سیگنالینگ) با ایستگاه استفاده می‌شود و یک کانال با پهنای باند وسیع که برای ارسال فریم، در اختیار ایستگاه قرار می‌گیرد.



شکل ۱۰-۴. دسترسی چندگانه مبتنی بر تقسیم طول موج.

به نحوی که در شکل ۴-۱۰ دیده می شود هر کانال به چندین گروه برش زمانی مستقل (اسلات) تقسیم شده است. اجازه بدهید تعداد برشهای زمانی در کانال کنترل را m و تعداد برشهای زمانی کانال داده را $n+1$ فرض کنیم؛ n تا از برشهای کانال داده، برای ارسال فریم و آخرین آنها برای گزارش وضعیت خود ایستگاه، کاربرد دارد (در حقیقت، این برش برای گزارش برشهای آزاد در دو کانال به کار می آید). در هر دو کانال [داده و کنترل] دنباله برشهای زمانی بطور دائم و متناوب تکرار می شوند و از برش شماره صفر آغاز می گردد. برش شماره صفر بگونه ای نشانه گذاری شده که دیگر ایستگاه ها بسادگی قادر به تشخیص آن هستند. تمام کانالها با استفاده از یک سیگنال ساعت سراسری سنکرون می شوند.

این پروتکل از سه رده متفاوت ترافیک حمایت می کند: (۱) ترافیک اتصال گرا با نرخ ثابت برای عملیاتی نظیر ارسال تصاویر ویدیویی غیر فشرده (۲) ترافیک اتصال گرا با نرخ متغیر برای عملیاتی نظیر انتقال فایل (۳) ترافیک دیتاگرام مثل ارسال بسته های UDP.

در دو پروتکل اتصال گرا، نظریه آن بوده که وقتی A بخواهد با B ارتباط برقرار کند ابتدا یک فریم کنترلی خاص به نام «فریم تقاضای اتصال» (CONNECTION REQUEST) در یک برش خالی از کانال کنترلی متعلق به B قرار می دهد. اگر B این تقاضا را پذیرفت مبادله، از طریق کانال داده انجام می شود. هر ایستگاه دارای دو فرستنده و دو گیرنده به ترتیب زیر است:

۱. یک گیرنده با طول موج ثابت برای گوش دادن به کانال کنترلی خودش
 ۲. یک فرستنده قابل تنظیم برای ارسال بر روی کانالهای کنترلی دیگر ایستگاهها
 ۳. یک فرستنده با طول موج ثابت برای انتقال فریمهای داده بر روی خروجی خودش
 ۴. یک گیرنده با طول موج قابل تنظیم برای انتخاب یکی از فرستنده های داده و گوش دادن به آن
- به عبارت دیگر هر ایستگاه مدام به کانال کنترلی خودش گوش فرا می دهد تا تقاضاهای ارتباط را دریافت کند ولیکن برای دریافت داده های دیگران، باید خود را با طول موج فرستنده تنظیم کرده و تطبیق بدهد. تنظیم طول موج به کمک ابزاری به نام «ایتر فیر و متر فابری-پرو یا ماک-زندر»^۱ انجام می شود که در حقیقت نوعی فیلتر نوری است که در آن به غیر از یک باند خاص از طول موجها، بقیه حذف می شوند.
- حال ببینیم که ایستگاه A، چگونه یک کانال رده ۲ [یعنی ارسال اتصال گرا با نرخ متغیر] با B، مثلاً برای انتقال فایل ایجاد می کند. ابتدا A، گیرنده داده خود را بر روی طول موج ایستگاه B تنظیم می نماید و انتظار می کشد تا به یک «برش گزارش وضعیت»^۲ (Status Slot) برسد. این برش زمانی، مشخص می کند که کدامیک از برشها در کانال کنترلی، خالی و کدام تخصیص داده شده اند. بعنوان مثال، در شکل ۴-۱۰ می بینیم که از میان هشت برش کنترلی متعلق به ایستگاه B، شماره های ۰ و ۴ و ۵ آزاد و بقیه اشغال هستند. (علامت «x» معنای پر بودن برش است). ایستگاه A یکی از برشهای آزاد کانال کنترل (مثلاً ۴) را انتخاب کرده و پیام «تقاضای اتصال» (CONNECTION REQUEST) خود را در آن می گذارد. از آنجایی که B بطور مدام بر کانال کنترل خود نظارت دارد این تقاضا را می بیند و برش ۴ را به A انتساب می دهد. این انتساب در «برش گزارش وضعیت» از کانال داده متعلق به ایستگاه B، اعلام می شود. وقتی A این اعلام را دریافت می کند متوجه می شود که اتصالی یک طرفه برایش مهیا است. اگر A تقاضای اتصالی دوطرفه داشته باشد، ایستگاه B نیز باید همین روال را انجام بدهد. ممکن است درست در زمانی که A سعی می کند برش شماره ۴ از کانال کنترل B را برای خود تصرف نماید، C

۱. Fabry-Perot or Mach-Zehnder Interferometer

۲. «برش گزارش وضعیت» (Status Slot)، آخرین برش در هر گروه از برشهای زمانی است.

نیز همین کار را بکند. در این حالت هیچکدام موفق نخواهند شد و با نظارت بر «برش گزارش وضعیت» متوجه این شکست می‌شوند. [چون پاسخی در «برش گزارش وضعیت» دریافت نمی‌کنند.] در این حالت هر یک به اندازه یک مقدار زمان تصادفی صبر کرده و از نو شروع می‌نمایند.

در این لحظه، هر یک از طرفین روشی بدون اشکال برای ارسال پیامهای کنترلی کوتاه خود به طرف دیگر در اختیار دارند. برای انجام عملیات انتقال فایل، اکنون A می‌تواند پیغام کنترلی کوتاهی برای B مثلاً بدین مضمون بفرستد: «لطفاً داده‌های من را در برش داده شماره ۳ دریافت کنید؛ در این برش یک فریم داده برای شما ارسال شده است.» وقتی B این پیام کنترلی را دریافت می‌کند گیرنده خود را با طول موج کانال خروجی A تنظیم می‌کند. بسته به پروتکل لایه بالاتر، B نیز می‌تواند در صورت تمایل از چنین مکانیزمی برای برگرداندن پیام‌های تصدیق (ACK) استفاده نماید.

دقت کنید که مسئله دیگری نیز ممکن است بوجود بیاید و آن هم اینکه اگر A و C اتصال با B داشته باشند و هر کدام بطور همزمان به B اعلام کند که به برش شماره ۳ مراجعه کند. B تقاضای یکی از آنها را بطور تصادفی انتخاب کرده و دیگری قادر به ارسال نخواهد شد.

برای ترافیک با نرخ ثابت، گونه دیگری از این پروتکل به کار گرفته می‌شود: وقتی A تقاضای ایجاد یک اتصال می‌کند، بطور لحظه‌ای فریمی را بدین مضمون برای B می‌فرستد: «آیا مجاز به ارسال دائم در برش شماره ۳ برای شما هستم؟» اگر B قادر به پذیرش چنین تقاضایی باشد (یعنی هیچ قرار قبلی برای این برش نگذاشته باشد)، یک اتصال با پهنای باند تضمین شده، ایجاد خواهد شد. در غیر این صورت A می‌تواند پیشنهاد دیگری را براساس خالی بودن برشهای دیگر ارائه بدهد. برای ترافیک رده سوّم (یعنی ترافیک دیتاگرام) نیز از گونه متفاوتی استفاده می‌شود: به جای نوشتن پیغام «تقاضای اتصال» در یک برش کنترلی، پیام DATA FOR YOU IN SLOT 3 را بر روی کانال کنترل قرار می‌دهد (به مضمون آنکه در برش ۳ داده‌ای بر شما وجود دارد). اگر B در خلال برش شماره ۳ از کانال داده آزاد باشد، انتقال انجام می‌شود و در غیر این صورت فریم از دست خواهد رفت. بدین ترتیب نیاز به برقراری هیچ اتصالی نخواهد بود.

گونه‌های متفاوتی از این پروتکل قابل اجرا است. مثلاً به جای آنکه هر ایستگاه بطور مستقل برای خودش کانال کنترل داشته باشد، یک کانال کنترلی واحد بین همه ایستگاه‌ها مشترک باشد. به هر ایستگاه نیز مجموعه‌ای از برشهای هر گروه متناسب می‌شود تا بتوان چندین کانال مجازی را بر روی یک کانال فیزیکی واحد مالتی پلکس کرد. همچنین این امکان وجود دارد که فقط از یک فرستنده قابل تنظیم و یک گیرنده قابل تنظیم در هر ایستگاه استفاده شود و کانال واحد هر ایستگاه به m برش کنترلی و بدنبال آن n+1، برش داده تقسیم گردد. اشکال این روش آنست که ایستگاه‌های فرستنده مجبورند برای در اختیار گرفتن کانال، زمان بیشتری را منتظر بمانند و از طرفی دنباله فریمهای داده با فاصله بیشتری از هم ارسال می‌شوند چراکه فریم کنترلی مابین آنها قرار گرفته است.

تاکنون پروتکل‌های WDM بی‌شماری پیشنهاد و پیاده‌سازی شده که در بسیاری از جزئیات با هم متفاوت هستند. برخی از آنها دارای یک کانال کنترل واحد هستند در حالیکه برخی دیگر چندین کانال کنترلی دارند. بعضی از آنها تاخیر انتشار را به حساب آورده‌اند در حالیکه بعضی از آن چشمپوشی کرده‌اند. برخی از آنها زمان تنظیم [فیلتر گیرنده یا فرستنده] را به عنوان بخشی مشخص از مدل خود در نظر گرفته‌اند در حالیکه برخی از آن صرفنظر کرده‌اند. این پروتکلها از دیدگاه پیچیدگی پردازش، ظرفیت و بهره مفید و قابلیت گسترش نیز با یکدیگر متفاوتند. به سیستمی که از تعداد زیادی فرکانس استفاده می‌کند، اصطلاحاً DWDM (Dense Wavelength Division Multiplexing) اطلاق می‌شود. برای آگاهی بیشتر به مراجع ذیل مراجعه کنید:

(Bogineni et al., 1993; Chen, 1994; Goralski, 2001; Kartalopoulos, 1999; Levine & Akyidiz, 1995)

۶-۲-۴ پروتکل‌های بی‌سیم برای شبکه محلی

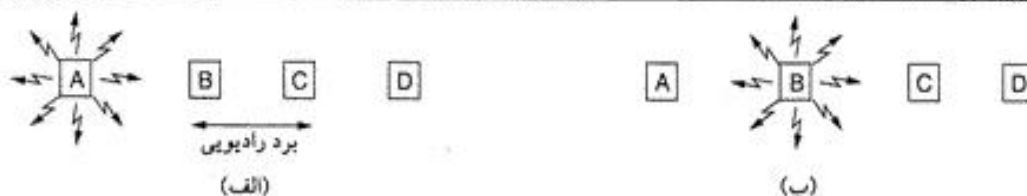
به موازات رشد تعداد دستگاه‌های محاسباتی و مخابراتی قابل حمل و نقل (همراه)، تقاضا برای اتصال آنها به دنیای خارج نیز افزایش یافته است. اگرچه حتی نخستین سری تلفنهای همراه امکان اتصال به تلفن دیگر را داشتند ولیکن کامپیوترهای کیفی، فاقد چنین قابلیت بودند. چیزی نگذشت که مودمها به عنوان یک ابزار معمولی بر روی کامپیوترهای همراه جا خوش کردند ولیکن برای برقراری ارتباط، می‌بایست سیم این کامپیوترها را به پریزهای دیواری تلفن وصل کرد. نیاز به سیم جهت اتصال به یک شبکه ثابت، بدین معناست که کامپیوترها قابل حمل و نقل هستند ولی «همراه» (Mobile) تلقی نمی‌شوند.

برای تحقق معنای واقعی «همراه بودن»، کامپیوترهای کیفی نیاز به ارتباط از طریق سیگنالهای رادیویی (یا مادون قرمز) داشتند. در چنین شرایطی، کاربران می‌توانند در حین گردش یا قایقرانی، نامه‌های خود را بخوانند یا نامه بفرستند. به گونه‌ای که در بخش ۱-۵-۴ اشاره شد، کامپیوترهای کیفی که از طریق سیگنالهای رادیویی با یکدیگر مخابرات اطلاعات می‌کنند، شبکه محلی بی‌سیم تلقی می‌شوند. اینگونه شبکه‌های محلی تفاوت‌های عمده‌ای با شبکه‌های محلی رایج دارند و به پروتکل‌های خاصی در زیرلایه MAC نیازمندند. در این بخش برخی از این پروتکلها را بررسی خواهیم کرد. برای آگاهی بیشتر در مورد شبکه‌های محلی بی‌سیم مراجع (Geier, 2002; O'Hara and Petrick, 1999) مفیدند.

پیکربندی رایج برای شبکه‌های محلی بی‌سیم بدین نحو است که در یک ساختمان اداری تعدادی ایستگاه ثابت (که نقاط دسترسی -Access Point- نامیده می‌شوند) نصب می‌شود. تمام این ایستگاه‌های ثابت از طریق فیبرنوری یا سیم مسی به یکدیگر متصل هستند. اگر توان انتقال رادیویی ایستگاه‌های ثابت و کامپیوترهای کیفی بگونه‌ای تنظیم شود که محدوده‌ای حدود ۳ تا ۴ متر را پوشش بدهد هر اتاق در ساختمان نقش یک «سلول» را ایفاء خواهد کرد و کل ساختمان همانند سیستمهای معمولی تلفن سلولی (یعنی شبکه تلفن همراه که در فصل ۲ بدان پرداختیم) عمل می‌کند ولی برخلاف سیستم تلفن سلولی، هر سلول تنها دارای یک کانال واحد است که کل پهنای باند موجود را در برگرفته و تمام ایستگاه‌های درون آن سلول را پوشش خواهد داد. عموماً پهنای باند این شبکه بین ۱۱ تا ۴۵ مگاهرتز بر ثانیه است.

در توضیحات زیر برای ساده‌تر شدن بحث، فرض را بر آن گذاشته‌ایم که تمام فرستنده‌های رادیویی، دارای برد ثابت و محدودی هستند. وقتی گیرنده‌ای در برد دو ایستگاه فعال و در حال ارسال قرار بگیرد سیگنال دریافتی او، مصدوم و بلااستفاده خواهد بود؛ درک این حقیقت که در اغلب شبکه‌های محلی بی‌سیم، تمام ایستگاه‌ها الزاماً در برد یکدیگر قرار ندارند، بسیار مهم است و منجر به پیچیدگی‌هایی خواهد شد. به علاوه هرگاه شبکه محلی بی‌سیم در داخل ساختمان قرار گرفته باشد وجود دیوار بین ایستگاه‌ها می‌تواند تاثیر مخربی بر روی برد هر ایستگاه داشته باشد.

یک روش ناشیانه برای تخصیص کانال در شبکه محلی بی‌سیم، بکارگیری روش CSMA است: ایستگاه‌ها به کانال گوش بدهند و در صورتی که هیچ ایستگاه دیگری در حال ارسال نبود، انتقال فریم انجام شود. مشکلی که در این پروتکل وجود دارد [در حالیکه در شبکه‌های مبتنی بر کابل وجود ندارد] آنست که در محیط بی‌سیم، تداخل امواج و تصادم فقط در گیرنده اهمیت دارد نه در فرستنده! برای آنکه به ماهیت این مشکل پی ببریم به شکل ۴-۱۱ که در آن چهار ایستگاه بی‌سیم ترسیم شده نگاهی بیندازید. اینکه کدام ایستگاه ثابت و کدامیک کامپیوتر کیفی [همراه] هستند، برای هدفی که در پیش داریم اهمیتی ندارد. برد رادیویی ایستگاه‌ها به گونه‌ای است که A و B در



شکل ۱۱-۴. یک شبکه محلی بی سیم (الف) A در حال ارسال (ب) B در حال ارسال.

برد یکدیگر هستند و سیگنال آنها می توانند با یکدیگر تداخل کرده، خراب شود. C نیز می تواند با B و D تداخل سیگنال داشته باشد ولی با A تداخل ندارد [چون در برد او نیست].

ابتدا فرض کنید که A در حال ارسال برای B، است: (شکل ۱۱-۴-الف) اگر C کانال را شنود کند هیچ چیزی از A نمی شنود چون در برد A نیست و بدین ترتیب به اشتباه نتیجه می گیرد که می تواند برای B ارسال داشته باشد. اگر C ارسال خود را آغاز نماید در ایستگاه B [با سیگنال ارسالی از A] تداخل کرده و فریم رسیده از A را نابود خواهد کرد. این مشکل که یک ایستگاه قادر نیست حضور یک رقیب را بر روی کانال تشخیص بدهد، «مشکل ایستگاه پنهان» (Hidden Station Problem) نامیده می شود و از آنجا ناشی می شود که ایستگاه ها از هم دور بوده و سیگنال های یکدیگر را نمی شنوند.

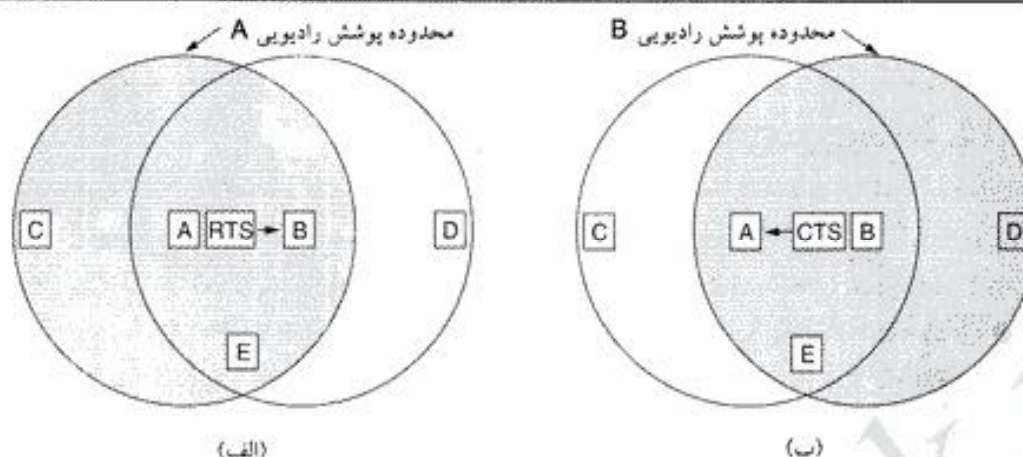
اکنون حالت برعکس شرایط فوق را بررسی می کنیم: به نحوی که در شکل ۱۱-۴-ب می بینید، B در حال ارسال برای A است. اگر C به شنود کانال بپردازد متوجه می شود که کانال اشغال است و به غلط نتیجه می گیرد که نباید برای D ارسال داشته باشد در حالیکه ارسال برای D هیچ اشکالی ندارد چرا که A و D در ناحیه ای دور از یکدیگر قرار گرفته اند و تصادمی پدید نخواهد آمد. این مشکل به نام «مشکل ایستگاه آشکار» (Exposed Station Problem) نامیده می شود.

مسئله اصلی آنست که یک دستگاه قبل از شروع به ارسال می خواهد بداند که آیا در پیرامون گیرنده، فعالیتی وجود دارد یا خیر؟ روش CSMA صرفاً می تواند در مورد فعالیت پیرامون فرستنده تشخیص خود را ارائه بدهد. به خاطر داشته باشید که بر روی سیم، سیگنالها به تمامی ایستگاه ها منتشر می شوند و طبعاً در آن واحد، فقط یک ایستگاه می تواند ارسال داشته باشد. در سیستمی که مبتنی بر امواج رادیویی یا برد کوتاه است چندین ایستگاه می توانند بطور همزمان ارسال داشته باشند البته مشروط به آنکه ایستگاه های مقصد، متفاوت و در محدوده برد یکدیگر نباشند.

روش دیگری جهت اندیشیدن به این مشکل، تجسم اداره ای است که کارمندان آن کامپیوترهای کیفی به همراه کارت شبکه بی سیم در اختیار دارند. فرض کنید کارمندی مثل لیندا می خواهد پیامی را برای میلتنون بفرستد. کامپیوتر لیندا محیط پیرامون خود را شنود کرده و چون فعالیتی را تشخیص نمی دهد ارسال را شروع می کند. غافل از آنکه در محل دفتر کار میلتنون، تصادم رخ خواهد داد زیرا شخص ثالثی هم اکنون در حال ارسال برای اوست و از آنجا که دفتر کار آن شخص از لیندا دور بوده، فعالیت او تشخیص داده نشده است.

MACAW, MACA

یکی از اولین پروتکل های طراحی شده برای شبکه های محلی بی سیم، پروتکل MACA (پروتکل دسترسی چندگانه با اجتناب از تصادم) است. (Karn, 1990) ایده اصلی در این روش آنست که فرستنده به نحوی گیرنده را تحریک به ارسال یک فریم کوتاه برای ایستگاه های پیرامون خود کند تا آنها بی که در برد او هستند و این فریم کوتاه را می شنوند، از ارسال اطلاعات در خلال زمان دریافت فریم، خودداری کنند. روش MACA در شکل ۱۲-۴ نشان داده شده است.



شکل ۱۲-۴. پروتکل MACA. (الف) در حال ارسال یک فریم RTS به B. (ب) در حال ارسال فریم پاسخ CTS به A.

حال بررسی کنیم که چگونه A فریمی را برای B می فرستد. ایستگاه A کارش را با ارسال یک فریم کوتاه به نام RTS^۱ برای B شروع می کند. (شکل ۱۲-۴-الف) درون این فریم کوتاه ۳۰ بیتی، طول کل فریم داده ای که قرار است در آینده ارسال گردد مشخص شده است. B در پاسخ، فریمی به نام CTS^۲ ارسال می نماید. فریم CTS نیز طول فریم داده ای را که قرار است ارسال شود، مشخص می کند؛ (شکل ۱۲-۴-ب). پس از دریافت فریم CTS، ایستگاه A می تواند ارسال خود را آغاز نماید.

حال ببینیم ایستگاه هائی که این فریمها را می شنوند چه عکس العملی نشان می دهند. هر ایستگاهی که RTS را می شنود به A نزدیک است و باید به اندازه ای صبر کند تا پیام CTS، بدون تداخل و تصادم باز گردد. هر ایستگاه که CTS را می شنود، نزدیک به B (گیرنده فریم داده) است و باید در خلال ارسال فریم کامل توسط A، ساکت بماند. از آنجا که طول فریم داده ای که در آینده قرار است ارسال شود، در فریم CTS مشخص شده، ایستگاه های شنونده CTS نیز می توانند زمان سکوت و انتظار، را تخمین بزنند.

در شکل ۱۲-۴ ایستگاه C در برد A است ولی در برد B نیست لذا RTS منتشره از A را می شنود ولی CTS ارسالی از B را نمی شنود. این ایستگاه مجاز است پس از آنکه به اندازه زمان ارسال فریم CTS منتظر ماند، همزمان با ارسال فریم داده A، او هم به ارسال خود مشغول باشد. در سمت مقابل، D در برد B هست ولی در برد A قرار ندارد، فلذا RTS را نمی شنود در حالیکه CTS را می شنود. شنیدن CTS بدین معناست که ایستگاه در نزدیکی ایستگاه گیرنده واقع شده و ایستگاه های شنونده CTS باید ارسال خود را آنقدر به تعویق بیندازند تا ارسال فریم مورد انتظار به پایان برسد. ایستگاه E هر دو فریم کنترل RTS و CTS را می شنود و همانند D بایستی منتظر بماند تا ارسال فریم خاتمه یابد.

با تمام این تمهیدات، باز هم وقوع تصادم محتمل است. به عنوان مثال اگر B و C هر دو بطور همزمان فریم RTS برای A بفرستند، تصادم رخ می دهد و فریم از بین خواهد رفت. در صورت بروز تصادم، فرستنده ناموفق (یعنی فرستنده ای که به مدت زمان مشخصی پس از ارسال RTS، فریم CTS دریافت نکند) به اندازه یک زمان تصادفی صبر کرده و از نو تلاش می کند. الگوریتم انتظار و تکرار مجدد به نام «الگوریتم عقبگرد نمایی» شهرت دارد و در مطالعه شبکه اتترنت آنرا تشریح خواهیم کرد.

به منظور افزایش کارایی پروتکل MACA، پژوهشگری به نام Bharghavan و گروه همکار او (۱۹۹۴) با

بررسی نتایج مطالعات شبیه سازی، اصلاحاتی را پیشنهاد کرده و پروتکل جدید را MACAW^۱ نامیدند. در بدو کار آنها متوجه شدند که اگر در لایه پیوند داده، پس از دریافت یک فریم سالم، پیام اعلام وصول (ACK) آن ارسال نگردد، طبعاً فریمهای از بین رفته، ارسال مجدد نخواهد شد تا آنکه مدتی بعد و در لایه انتقال، عدم وجود آنها کشف شود؛ این زمان انتظار بسیار طولانی و وقتگیر است و کارایی پروتکل را بشدت کاهش خواهد داد. آنها این مشکل را با معرفی یک فریم جدید ACK که پس از دریافت موفق یک فریم داده ارسال می شود، حل کردند. همچنین آنها متوجه شدند که برخی از ویژگیهای CSMA باز هم در این شبکه قابل استفاده است؛ مثلاً برای آنکه یک ایستگاه، همزمان با ایستگاه دیگری برای یک مقصد مشابه، فریم RTS نفرستد، بهتر است قبل از ارسال RTS کانال شنود شود؛ بدین ترتیب قابلیت شنود سیگنال حامل نیز به پروتکل اضافه شد. به علاوه آنها تصمیم گرفتند که الگوریتم «عقبگرد نمایی» را به جای آنکه برای هر ایستگاه به کار ببرند بطور مستقل برای یک استریم داده (که بین یک زوج مبدا و مقصد جریان دارد) اعمال نمایند. در آخر آنها به این پروتکل مکانیزمهای افزودند تا ایستگاهها با یکدیگر اطلاعاتی در خصوص ازدحام (Congestion) رد و بدل کنند. همچنین الگوریتم «عقبگرد نمایی» را بگونه ای تغییر دادند تا در خصوص مشکلات موقت، عکس العمل شدیدی از خودش نشان ندهد و بدین ترتیب کارایی سیستم بهبود یابد.

۳-۴ اترنت

تا اینجا پروتکل های تخصیص کانال را بصورت کلی و اجمالی مطالعه کردیم؛ اکنون زمان آن فرا رسیده تا این قواعد را بر روی سیستم های واقعی و خصوصاً شبکه های LAN، اعمال نماییم. همانگونه که در بخش ۳.۵.۱ اشاره شد، IEEE تعدادی از شبکه های محلی و بین شهری را با نام IEEE 802 استاندارد سازی کرده است. در فهرست شکل ۳.۸.۱ دیدیم که برخی از این شبکه ها هنوز وجود دارند و برخی دیگر به تاریخ پیوسته اند. شاید آنهایی که به «نظریه تناسخ روح» معتقدند بتوانند اینگونه بیندیشند که آقای چارلز داروین برگشته و به عنوان یکی از اعضای کمیته IEEE در حال حذف، پالایش و تکمیل زنجیره استانداردهاست!!! از بین استانداردهای باقیمانده از گذشته، می توان به 802.3 (اترنت) و 802.11 (شبکه محلی بی سیم) اشاره کرد. هنوز زود است که در خصوص 802.15 (Blue tooth) و 802.16 (شبکه بین شهری بی سیم) قضاوت کنیم؛ در این مورد می توانید به ویرایش پنجم این کتاب مراجعه کنید؛ لایه فیزیکی و زیرلایه MAC در شبکه های 802.3 و 802.11 تفاوت بنیانی دارند ولی این تفاوتها در زیرلایه منطقی (تعریف شده در 802.2 LLC)، همگرا خواهد شد و بدین ترتیب هر دوی آنها، واسطه (Interface) مشابهی با لایه شبکه دارند. [LLC فوقانی ترین زیرلایه از لایه پیوند داده هاست و وظیفه دارد تفاوتها و ناهمگونیهای اجتناب ناپذیر زیرلایه های پایین را از دید لایه شبکه مخفی نگاه دارد تا سرویسهایی که به این لایه ارائه می شود استاندارد و یکسان باشد. -م]

در خصوص اترنت در بخش ۱.۵.۳ توضیحاتی داده شده است؛ لذا آن مفاد را در اینجا تکرار نخواهیم کرد. در عوض، بر روی جزئیات تخصصی اترنت، پروتکل های مرتبط با آن و پیشرفتهای اخیر در زمینه اترنت سرعت بالا (گیگابیت اترنت) متمرکز خواهیم شد. از آنجایی که «اترنت» و IEEE 802.3 به غیر از دو تفاوت جزئی که به آنها اشاره خواهیم کرد، از بقیه جهات یکسان هستند، بسیاری از افراد این دو شبکه را معادل هم در نظر گرفته و ما نیز این دو را یکی فرض خواهیم کرد. برای اطلاعات بیشتر در خصوص اترنت از مراجع زیر استفاده کنید:

Breyer and Riley, 1999; Scifert, 1998; and Spurgeon 2000)

^۱. Multiple Access with Collision Avoidance for Wireless networks

۱-۳-۴ کابل کشی اترنت

از آنجایی که نام اترنت در اصل برگرفته از واژه «اتر» است که به کابل اشاره دارد ما نیز توضیحات خود را از کابل شروع خواهیم کرد. در این شبکه چهار نوع کابل که فهرست آنها در شکل ۱۳-۴ آمده، رایج هستند:

مزایا	تعداد کمره در هر قطعه	حداکثر طول قطعه	نوع کابل	نام کابل
کابل اصلی و اولیه (از رده خارج)	100	500 m	Thick coax	10Base5
به هاب نیازی نیست	30	185 m	Thin coax	10Base2
ارزان ترین سیستم	1024	100 m	Twisted pair	10Base-T
بهترین انتخاب برای مابین ساختمانها	1024	2000 m	Fiber optics	10Base-F

شکل ۱۳-۴. رایجترین انواع کابل کشی اترنت.

از منظر تاریخ، نخستین نوع کابل در اترنت، 10Base5 بود که عموماً به «اترنت ضخیم» شهرت داشت. این کابل به شیلنگهای زرد رنگ باغبانی شبیه است و هر ۲/۵ متر بر روی آن علامتی گذاشته شده تا محل انشعاب تزریقی مشخص باشد. (البته طبق استاندارد 802.3، زرد بودن کابل الزامی نیست ولی پیشنهاد شده است.) اتصال به کابل عموماً از طریق «انشعاب تزریقی» (Vampire Tap) انجام می شود که در آن یک سوزن به دقت در مرکز کابل کوآکسیال فرو می رود. نماد 10Base5 بدین معنی است که شبکه با نرخ 10Mbps کار می کند، از سیگنالینگ باند پایه (Baseband) بهره گرفته و طول حداکثر یک قطعه کابل ۵۰۰ متر است. گونه دیگری کابل با نام 10Broad36، در باند وسیع (Broadband) طراحی شد ولی هیچگاه به بازار نیامد و عملاً از صحنه محو گردید. در صورتی که کانال از نوع کابل کوآکس باشد، عددی که بعد از کلمه Base ظاهر می شود، حداکثر طول کابل را بر مبنای ۱۰۰ متر مشخص می کند.

نوع دوم کابل کشی اترنت 10Base2 یا «اترنت نازک» نام داشت که برخلاف کابل قبلی (که شبیه به شیلنگ باغبانی و غیر قابل انعطاف بود) به راحتی خم می شد. برای اتصال به این نوع کابل، به جای استفاده از انشعاب تزریقی، می توان از کانکتورهای BNC معمولی و ایجاد یک اتصال بشکل T، بهره گرفت. کانکتورهای BNC بسیار قابل اعتماد و کاربرد آنها ساده تر است. کابلهای اترنت نازک نیز، بسیار ارزان و نصب آن راحت است ولیکن حداکثر طول کانال به ۱۸۵ متر کاهش یافته و به هر قطعه کابل حداکثر می توان ۳۰ ایستگاه متصل کرد.

تشخیص طول بیش از اندازه کابل، انشعابات بد، شل شدن اتصالات یا هرگونه پارگی در جایی از آن، از اساسی ترین مشکلات این دو نوع کابل محسوب می شوند [زیرا بروز یکی از این اشکالات کل شبکه از کار خواهد انداخت]. به همین دلیل تکنیکی برای تشخیص این معایب ابداع شده است: یک پالس الکتریکی با شکل معمولی به درون کابل تزریق می شود. اگر این پالس به یک مانع [پارگی] یا به انتهای کابل برخورد کند، یک سیگنال بازتاب (Echo) تولید شده و باز خواهد گشت. با اندازه گیری دقیق زمان بین ارسال پالس و زمان دریافت بازتاب آن، پیدا کردن محل تقریبی عامل بازتاب [محل خرابی] کشف خواهد شد. به این روش اصطلاحاً «بازتاب سنجی در حوزه زمان» (Time Domain Reflectometry) گفته می شود.

مشکلات تشخیص محل پارگی در کابل، باعث شد که الگوی متفاوتی در سیم کشی این نوع شبکه به کار گرفته شود؛ در روش جدید هر ایستگاه یک کابل اختصاصی دارد که آنرا به یک هاب مرکزی متصل می کند. این هاب اتصال الکتریکی تمام ایستگاهها را از درون، برقرار می سازد. معمولاً این سیمها، از نوع زوج سیمهای معمولی خطوط تلفن هستند زیرا در ساختمانهای اداری این سیم کشی از قبل وجود دارد و تعداد زیادی از این زوج سیمها بلااستفاده رها شده اند. به این روش سیم کشی اصطلاحاً 10BaseT گفته می شود. هابها ترافیک داده های ورودی را

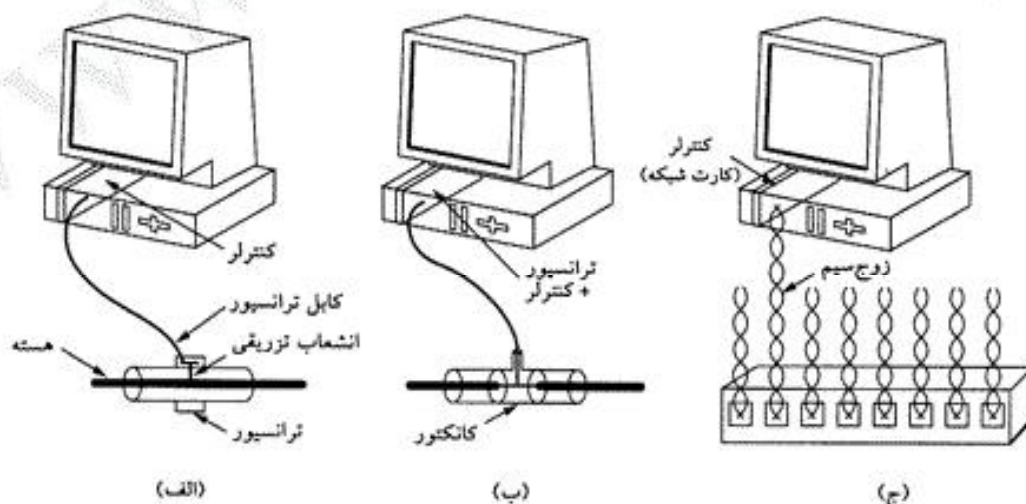
بافر نمی کنند [بلکه فقط ایستگاهها را به کانال مشترک متصل می نمایند]. در ادامه همین فصل نسخه پیشرفته تر این ایده (یعنی سوئیچها) را تشریح خواهیم کرد که قادرند ترافیک ورودی را بافر کنند.

این سه روش سیم کشی در شکل ۴-۱۴ به تصویر کشیده شده است. در سیم کشی 10Base5، مدار ترانسیور^۱، باید بدقت و با اطمینان به دور کابل اصلی، محکم شود تا اتصال آن با هسته مرکزی کابل برقرار گردد. مدار ترانسیور دارای مدار الکترونیکی کوچکی است که حضور سیگنال حامل و بروز تصادم را تشخیص می دهد و در صورتی که متوجه بروز تصادم شود یک سیگنال خاص و غیر معتبر بر روی کابل می گذارد تا مطمئن شود بقیه نیز از تصادم مطلع شده اند.

در 10Base5، یک کابل بنام ترانسیور (کابل اتصال) ارتباط کارت واسط ایستگاه و کابل اصلی را برقرار می کند. طول این قطعه کابل، می تواند تا ۵۰ متر باشد و درون آن ۵ جفت سیم زره دار (Shielded) وجود دارد. دو جفت از آنها برای داده های ورودی و خروجی از ایستگاه است. دو جفت نیز برای سیگنالهای کنترلی ورودی/خروجی به کار گرفته می شوند. زوج پنجم که ممکن است از آن استفاده نشود برای تغذیه مدار الکترونیکی ترانسیور کاربرد دارد. همچنین، بعضی از ترانسیورها اجازه می دهند تا حداکثر هشت کامپیوتر نزدیک به هم، بدانها متصل شود تا تعداد ترانسیورهای مورد نیاز کاهش یابد.

کابل ترانسیور از یک طرف به کارت شبکه در درون کامپیوتر متصل است؛ کارت شبکه شامل یک تراشه کنترلر است که ارسال یا دریافت فریم به ترانسیور را بر عهده دارد. این کنترلر وظیفه دارد داده ها را در قالب یک فریم مناسب سازماندهی کند. همچنین محاسبه کدهای کشف خطا برای فریمهای خروجی و ارزیابی صحت فریمهای ورودی بر عهده همین کنترلر است. برخی از این تراشه ها دارای مقداری بافر هستند تا فریم های ورودی را جهت ارسال به صف نمایند؛ همچنین قادرند داده ها را به روش DMA به حافظه اصلی کامپیوتر منتقل کنند و در ضمن برخی از عملیات مدیریت شبکه را نیز انجام می دهند.

در سیم کشی 10Base2، اتصال کامپیوتر با کابل یکمک یک کانکتور BNC معمولی (از نوع T-شکل) انجام می شود. در این نوع سیم کشی، بخش الکترونیکی ترانسیور بر روی برد کنترلر قرار گرفته است و بدین ترتیب هر ایستگاه ترانسیور خود را دارد.



شکل ۴-۱۴. سه روش کابل کشی اترنت (الف) 10Base5 (ب) 10Base2 (ج) 10Base-T.

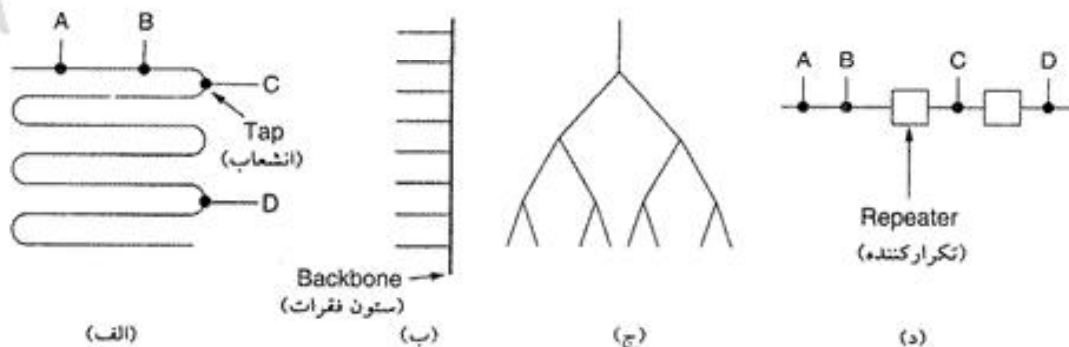
۱. ترانسیور (Tranciever) قطعه ای الکترونیکی است که اتصال کابل نازک متصل به ایستگاه و کابل ضخیم اصلی را برقرار می کند. س

در ساختار 10Base-T، هیچ کابلی مشترکی وجود ندارد بلکه فقط یک هاب (جعبه ای پر از مدارات الکترونیک) وجود دارد که تمام ایستگاه ها با یک کابل اختصاصی (غیرمشترک) بدان متصل می شوند. حذف یا اضافه یک ایستگاه بدین ساختار بسیار ساده و هرگونه پارگی در کانال به راحتی قابل کشف است. اشکال ساختار 10Base-T آنست که طول کابل متصل به هاب، حداکثر می تواند صد متر و در صورت استفاده از کابل باکیفیت و گرانی مثل Cat 5، حداکثر دویست متر باشد. علیرغم این محدودیت، ساختار 10Base-T بدلیل سادگی نصب و پشتیبانی و همچنین استفاده از سیم کشی موجود ساختمانها، بسیار رایج شده است. نسخه سریعتر 10Base-T یعنی 100Base-T در ادامه همین فصل تشریح خواهد شد.

روش چهارم کابل کشی، 10Base-F است که در آن از فیبر نوری بهره گرفته شده است. این گزینه بدلیل هزینه بالای اتصالات و پایانه های^۱ مورد نیاز، بسیار گران است ولی در عوض ایمنی بسیار بالایی در مقابل نویز دارد و انتخاب مناسبی برای کابل کشی بین ساختمانها یا هابهای دور از هم، به شمار می رود. در این روش، رشته هایی با طول کیلومتر نیز مجاز هستند. همچنین در این روش ضریب امنیت اطلاعات بسیار بالاست چراکه انشعاب گرفتن از این کانال [واستراق سمع داده ها] بسیار دشوارتر سیم های مسی است.

شکل ۴-۱۵ روشهای مختلف کابل کشی ساختمان را نشان می دهد. در شکل ۴-۱۵-الف یک کابل واحد، اطاق به اطاق کشیده شده و هر ایستگاه در نزدیکترین نقطه بدان متصل شده است. در شکل ۴-۱۵-ب یک رشته عمودی در نقش ستون فقرات از طبقه همکف تا بام کشیده شده و در هر طبقه کابل های افقی از طریق تقویت کننده های خاص (بنام تکرارکننده) به این ستون فقرات متصل شده اند. در برخی از ساختمانها کابل های افقی، نازک و کابل ستون فقرات، از نوع ضخیم انتخاب می شود. رایجترین ساختار، توپولوژی درختی است که در شکل ۴-۱۵-ج دیده می شود.

برای هر یک از روشهای کابل کشی اترنت، طول هر قطعه کابل نباید از یک مقدار حداکثر تجاوز کند. برای شبکه های وسیع می توان شبیه به شکل ۴-۱۵-د از چندین قطعه کابل که توسط «تکرارکننده» بهم متصل شده اند، استفاده کرد. تکرارکننده یک ابزار در لایه فیزیکی است داده ها را دریافت، تقویت (بازتولید) و مجدداً ارسال می کند. از دیدگاه نرم افزار، مجموعه ای از قطعات کابل که از طریق تکرارکننده به هم متصل شده اند هیچ تفاوتی با یک شبکه با کابل یکپارچه ندارد (مگر اندکی تاخیر اضافی که توسط تکرارکننده ها تحمیل می شود). یک سیستم ممکن است از چندین قطعه کابل و چند تکرارکننده تشکیل شده باشد ولیکن فاصله بین هر دو ترانسیور نیایستی از ۲/۵ کیلومتر تجاوز کند و در مسیر بین هر دو ترانسیور نباید بیش از ۴ تکرارکننده موجود داشته باشد.



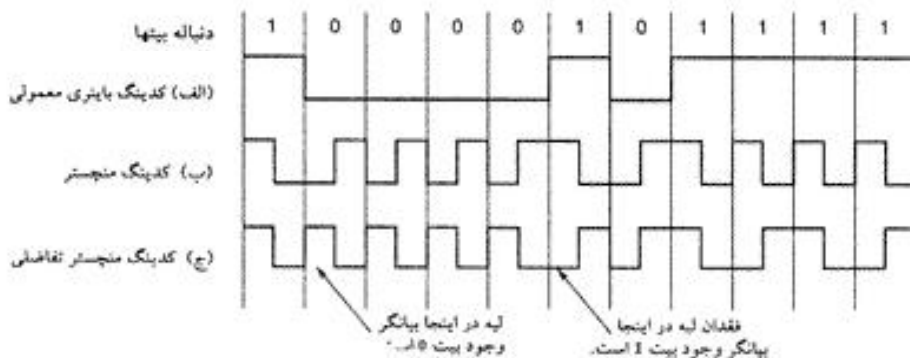
شکل ۴-۱۵ م. توپولوژی های مختلف کابل (الف) خطی (ب) ستون فقرات (ج) درختی (د) چندبخشی.

۲-۳-۴ کدینگ منچستر

در هیچیک از نسخه های مختلف اترنت، از روش معمولی کدینگ (یعنی صفر ولت برای بیت 0 و ۵ ولت برای بیت 1) استفاده نشده است چراکه منجر به برخی اشکالات و ابهام [در دریافت بیت ها] خواهد شد: مثلاً اگر ایستگاهی رشته بیت 0001000 را ارسال کند، ایستگاه های دیگر، ممکن است به غلط آنرا 10000000 یا 01000000 تشخیص بدهد زیرا ایستگاه نمی تواند تفاوت بین آزاد بودن خط (معادل صفر ولت) و بیت صفر (باز هم معادل صفر ولت) را تشخیص بدهد. این مسئله را می توان با در نظر گرفتن $+1$ ولت برای بیت یک و -1 ولت برای بیت صفر، حل کرد ولیکن باز هم مشکل اختلاف فرکانس نمونه برداری از سیگنال (نسبت به فرکانس اصلی) باقی می ماند. تفاوت در سرعت سیگنال ساعت فرستنده و گیرنده باعث خواهد شد که این دو از حالت سنکرون خارج شوند و در اینصورت محدوده هر بیت قابل تشخیص نخواهد بود؛ بالاخص وقتی یک دنباله طولانی پی در پی صفر یا دنباله یک پشت سرهم ارسال شود. روش معمولی کدینگ را در شکل ۴-۱۶ الف می بینید.

به روشی نیاز است تا گیرنده بتواند به درستی شروع، پایان یا وسط هر بیت را بدون نیاز به یک سیگنال ساعت خارجی تشخیص بدهد. دو روش به نامهای «کدینگ منچستر» و «کدینگ منچستر تفاضلی» دارای چنین قابلیتی هستند. در روش منچستر هر بیت از لحاظ زمانی به دو نیم بیت تقسیم می شود: برای ارسال بیت 1، در نیم بیت اول، ولتاژ بالا (High) و در نیمه دوم ولتاژ پائین (Low) قرار داده می شود. برای بیت صفر برعکس عمل می شود: در نیمه اول ولتاژ پائین و در نیمه دوم ولتاژ بالا ارسال می گردد. این روش اطمینان می دهد که در وسط هر بیت یک لبه (Transition) وجود دارد. اشکال روش منچستر آنست که در مقایسه با روش معمولی، به پهنای باند دو برابر نیاز دارد چراکه طول هر پالس نصف طول یک بیت است. [در حقیقت برای هر بیت ۲ پالس ارسال می شود]. به عنوان مثال برای ارسال داده با سرعت ده مگابیت بر ثانیه، سیگنال تولیدی باید با نرخ ۲۰ میلیون بار در ثانیه، تغییر سطح ولتاژ داشته باشد. روش کدینگ منچستر در شکل ۴-۱۶ ب نشان داده شده است.

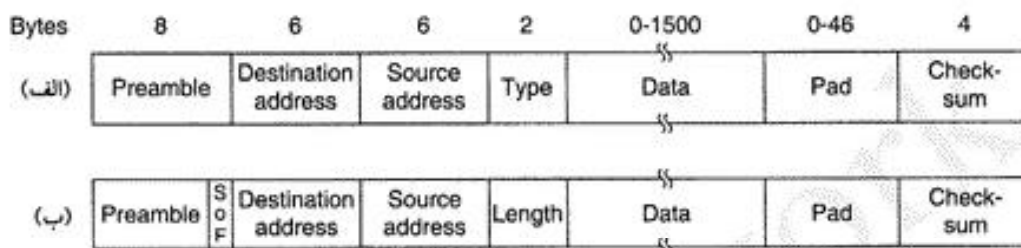
به گونه ای که در شکل ۴-۱۶ ج می بینید روش منچستر دیفرانسیلی نیز گونه ای از روش منچستر معمولی است. عدم وجود هرگونه لبه در ابتدای هر بیت (ابتدای Bit Time) نشان دهنده بیت 1 است و وجود لبه در ابتدای هر بیت، بیت صفر را مشخص می کند. در هر دو حالت، قطعاً یک لبه در میانه هر بیت وجود دارد. روش منچستر تفاضلی به ابزارهای پیچیده تری نیازمند است ولیکن در عوض ایمنی بیشتری در مقابل نویز از خود نشان می دهد. تمام سیستمهای اترنت برای سادگی از روش کدینگ منچستر معمولی استفاده کرده اند و در آنها سطح بالای سیگنال (حالت High) مقدار $+0.85$ ولت و سطح پایین سیگنال (حالت Low) مقدار -0.85 ولت و مقدار DC سیگنال نیز صفر است. در اترنت از روش منچستر تفاضلی استفاده نمی شود ولیکن در برخی از شبکه های محلی دیگر (مثل IEEE 802.5 Token Ring) از آن استفاده شده است.



شکل ۴-۱۶. الف) کدینگ باپتری معمولی (ب) کدینگ منچستر (ج) کدینگ منچستر تفاضلی.

۳-۳-۴ پروتکل زیرلایه MAC در اترنت

قالب اصلی فریم DIX (پیشنهاد شرکت های Xerox, Intel, DEC) در شکل ۴-۱۷ الف نشان داده شده است. هر فریم با هشت بایت Preamble (دبیاچه) شروع می شود که تمام بایتها دارای الگوی 10101010 هستند. کدینگ منچستر این الگوی هشت بایتی، بمدت ۶/۴ میکروثانیه یک سیگنال ساعت مربعی ده مگاهرتز تولید می کند تا بکمک آن گیرنده بتواند سیگنال ساعت خود را با سیگنال ساعت فرستنده سنکرون کند. گیرندگان موظفند با بهره گیری از ویژگی کدینگ منچستر تا انتهای فریم، سنکرون باقی بمانند و محدوده بیتها را به درستی تشخیص بدهند.



شکل ۴-۱۷. قالب فریم (الف) اترنت DIX (ب) اترنت IEEE 802.3

در این فریم دو آدرس تعریف شده است: یکی برای آدرس مقصد و یکی برای آدرس مبدا. طبق استاندارد، استفاده از آدرسهای ۲ یا ۶ بایتی مجاز است ولیکن در مشخصات معرفی شده برای استاندارد 10Mbps، آدرسها صرفاً شش بایتی هستند.^۱ اگر مقصد فریم یک ایستگاه واحد باشد، پرازشترین بیت آدرس مقصد، صفر و اگر مقصد فریم یک گروه از ایستگاهها باشد، ۱ است. آدرسهای گروهی این امکان را فراهم می آورند تا چندین ایستگاه بتوانند به یک آدرس واحد گوش بدهند. وقتی فریمی به یک آدرس گروهی ارسال می شود تمام ایستگاههای گروه قادرند آنرا دریافت نمایند. ارسال فریم برای یک گروه، اصطلاحاً چندپخش (Multicast) نامیده می شود. هرگاه تمام بیتهای آدرس مقصد در یک فریم، ۱ باشند به این آدرس «پخش فراگیر» (Broadcast) گفته می شود. فریمی که تمام بیتهای فیلد آدرس مقصد آن ۱ است توسط همه ایستگاههای شبکه دریافت خواهد شد. تفاوت بین ارسال «چندپخش» و «فراگیر» آنقدر مهم است که ارزش تکرار مجدد را دارد: یک فریم چندپخش برای یک گروه انتخابی از ایستگاهها در شبکه اترنت ارسال می شود در حالیکه فریم فراگیر، برای تمام ایستگاههای شبکه ارسال می گردد. ارسال چندپخش، انتخابی و منعطف است ولیکن نیاز به مدیریت گروهها دارد. در عوض ارسال فراگیر غیرمنعطف و غیرانتخابی است ولیکن به مدیریت گروه نیازی ندارد.

یکی دیگر از ویژگیهای جالب آدرس دهی در اترنت آنست که بیت چهل و ششم (مجاور بیت پرازش)، سراسری یا محلی بودن آدرسها را مشخص می کند. آدرسهای محلی، آدرسهای هستند که توسط مسئول شبکه تعیین شده و در خارج از شبکه محلی هیچ معنا و ارزشی ندارند. در مقابل آدرسهای سراسری بطور خاص و مرکزی توسط IEEE اختصاص داده می شوند تا این اطمینان حاصل شود که هیچ دو ایستگاهی در کل دنیا دارای آدرس سراسری یکسان نیستند. ۴۶ بیت باقیمانده (۴۶ = ۲ - ۴۸) فضائی معادل 2^{46} آدرس سراسری ایجاد می کند. ایده اصلی آنست که هر ایستگاه به صورت یکتا و منحصر به فرد آدرس دهی شود. تعیین موقعیت و مشخص کردن ایستگاه مقصد، برعهده لایه شبکه گذاشته شده است.

در ادامه، فیلد Type تعریف شده که به گیرنده فریم تفهیم می کند که با این فریم چه کاری بکند. وقتی که

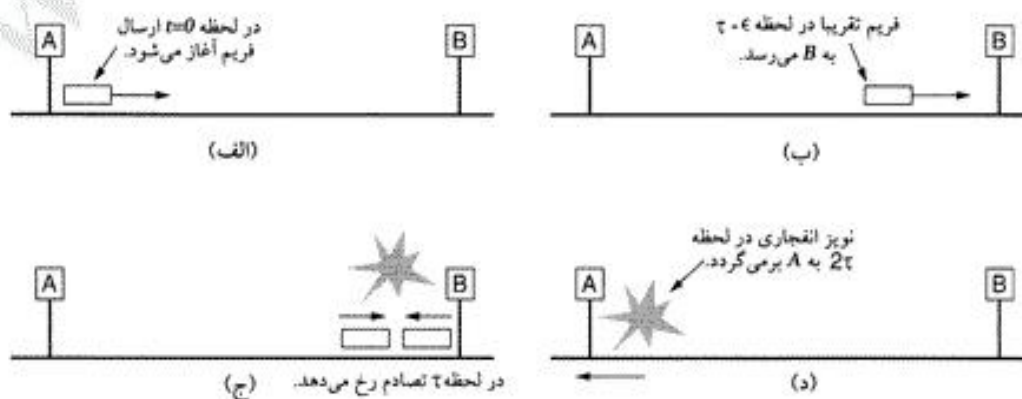
۱. امروزه در هیچیک از گونه های مختلف اترنت آدرسهای دوبایتی وجود خارجی ندارند. -م

چندین پروتکل لایه شبکه، بطور همزمان بر روی یک ماشین واحد اجرا شده باشند، هسته سیستم عامل پس از دریافت فریم، براساس این فیلد تصمیم می گیرد که آنرا به دست چه پروسه ای بسپارد. محتوای فیلد Type، پروسه ای که باید این فریم را تحویل بگیرد، مشخص می کند.

در ادامه، «فیلد داده» قرار می گیرد که گنجایش آن حداکثر ۱۵۰۰ بایت است. محدودیت ۱۵۰۰ بایتی، در زمان ارائه استاندارد DIX، بدلیخواه انتخاب شد و انگیزه اصلی طراحان از این انتخاب، صرفه جویی در میزان حافظه لازم برای کارتهای شبکه بوده است زیرا در آن سالها (۱۹۷۸) حافظه RAM گران تمام می شد.

گذشته از آنکه طول فریم دارای سقف حداکثر است، حداقل طول فریم نیز محدودیت دارد. گاهی ارسال فریمهای بدون داده، مفید خواهد بود ولی استفاده از چنین فریمهایی می تواند مشکل آفرین باشد. وقتی تصادم کشف می شود، فرستنده از ارسال باقیمانده فریم جاری صرف نظر می کند و بدین نحو دنباله ای از بیتها (که در حقیقت قطعه ابتدایی فریم است) روی کانال رها می شود. برای تشخیص فریمهای معتبر از فریمهای آشغال، در اترنت نیاز است که طول فریمهای معتبر حداقل ۶۴ بایت باشد که این ۶۴ بایت از ابتدای آدرس مقصد تا انتهای فیلد Checksum در نظر گرفته می شود. اگر بخش داده فریم از ۴۶ بایت (۱۸-۶۴) کمتر باشد در فیلد Pad آنقدر صفر اضافه می شود تا اندازه فریم به حداقل اندازه مجاز برساند.

دلیل دیگر (و بسیار مهمتر) در محدودیت حداقل اندازه برای هر فریم، آنست که مبادا ایستگاه قبل از رسیدن اولین بیت از فریمش به انتهای کابل، ارسال خود را به پایان برساند زیرا ممکن است تصادم بوجود بیاید (و چون ارسال فریم خاتمه یافته، ایستگاه از این موضوع باخبر نخواهد شد. -م) این مشکل در شکل ۴-۱۸ نشان داده شده است. در زمان $t=0$ ایستگاه A در یک طرف شبکه شروع به ارسال فریم روی کابل می کند. فرض کنید زمان تاخیر رسیدن فریم به انتهای کابل τ ثانیه باشد. دقیقاً قبل از رسیدن فریم به انتهای کابل (یعنی در لحظه $t = \tau$) آخرین ایستگاه متصل به کابل یعنی B شروع به ارسال فریم خود می نماید. حال وقتی B تشخیص می دهد که توان دریافتی از کانال از توان ارسالی بیشتر است متوجه بروز تصادم می شود و با قطع ارسال خود و تولید یک نویز ۴۸ بیتی قوی، بروز تصادم را به اطلاع بقیه ایستگاهها می رساند. بعبارت دیگر با ارسال یک نویز قوی این اطمینان حاصل می شود که هیچ ایستگاهی از پدیده تصادم بی خبر نخواهد ماند. در زمان حدود 2τ ، فرستنده متوجه نویز شده و از ارسال دست می کشد. سپس قبل از تلاش مجدد برای ارسال، به اندازه یک عدد تصادفی صبر می کند.



شکل ۴-۱۸. کشف تصادم می تواند تا زمان 2τ طول بکشد.

اگر ایستگاه تلاش کند فریمی کوتاه ارسال نماید و تصادم پدید بیاید، اگر چه ممکن است از تصادم مطلع شود ولی چون قبل از زمان بازگشت نویز (2τ)، ارسال فریم به پایان رسیده، فرستنده به غلط نتیجه می گیرد که فریم او

با موفقیت ارسال شده است و تکرار ارسال ممکن نخواهد بود. برای اجتناب از چنین وضعیتی، تمام فریمها بایستی از لحاظ زمانی از (2t) بیشتر باشند تا اطمینان حاصل شود که قبل از خاتمه ارسال فریم، تصادم کشف خواهد شد. برای شبکه اترنت 10Mbps با کابلی به طول حداکثر ۲۵۰۰ متر و چهار تکرارکننده (طبق تعریف 802.3)، زمان رفت و برگشت سیگنال (با احتساب تاخیر انتشار چهار تکرارکننده) در بدترین حالت تقریباً پنجاه میکروثانیه محاسبه شده که قطعاً قابل صرفنظر نیست. بدین ترتیب حداقل طول فریم باید به اندازه ای باشد که از لحاظ زمانی حداقل ۵۰ میکروثانیه باشد. در سرعت 10Mbps هر بیت ۱۰۰ نانوثانیه طول می کشد لذا کوچکترین طول فریم که عملکرد صحیح شبکه را تضمین می کند ۵۰۰ بیت خواهد بود. برای افزودن به حاشیه امنیت این مقدار، ۵۱۲ بیت معادل ۶۴ بایت در نظر گرفته شده است. در فیلد Pad از فریمهای کوچکتر از ۶۴ بایت، باید آنقدر داده زائد اضافه شود تا طول آن به ۶۴ بایت برسد.

به تناسب بالا رفتن سرعت شبکه یا باید طول کانال را کاهش داد یا آنکه به طول حداقل فریمها افزود. در یک شبکه LAN با کابلی به طول ۲۵۰۰ متر و نرخ ارسال 1 Gbps، طول حداقل هر فریم، ۶۴۰۰ بایت خواهد بود! در عوض اگر بیشترین فاصله دو ایستگاه به ۲۵۰ متر کاهش یابد طول حداقل فریم به ۶۴۰ بایت کاهش خواهد یافت. این محدودیت زمانی که به سمت شبکه های چند گیگابیتی حرکت کنیم بسیار مشکل آفرین خواهد بود. آخرین فیلد از فریم اترنت فیلد Checksum (کد کشف خطا) است. در حقیقت این فیلد یک کد ۳۲ بیتی استخراج شده از داده هاست. هرگاه تعدادی از بیت های فریم (در اثر نویز روی کانال) اشتباه دریافت شوند، کد Checksum آن نادرست خواهد بود و بدین ترتیب خطا کشف می شود. کد کشف خطای به کار رفته در اترنت از نوع CRC است که آنرا در فصل سوم تشریح کردیم. این کدها فقط برای کشف خطا کاربرد دارند و قادر به تصحیح خطا نیستند.

زمانی که IEEE اترنت را استاندارد کرد، دو تغییر کوچک در قالب فریم DIX ایجاد کرد؛ این تغییرات در شکل ۴-۱۷-ب مشهود است. اولین تغییر آن بود که طول Preamble (دبیاچه) را به ۷ بایت کاهش داد و بایت هشتم را به عنوان «بایت مشخص کننده ابتدای فریم» (Start Of Frame Delimiter) به کار برد تا با استانداردهای 802.4 و 802.5 سازگار باشد. تغییر دوم آن بود که فیلد Type را به فیلد Length تغییر کاربری داد. البته در اینصورت گیرنده نمی تواند تشخیص بدهد که با فریم دریافتی چه کند ولی این مسئله با اضافه کردن یک سرآیند اضافی به قسمت داده حل شده است. قالب قسمت داده را در همین فصل و در توضیح «کنترل منطقی لینک» تشریح خواهیم کرد. متأسفانه در زمان ارائه 802.3 سخت افزار و نرم افزار زیادی بر اساس اترنت DIX در حال استفاده بود و شرکت های سازنده و کاربران، چندان علاقمند به تبدیل فیلد Type به Length نبودند! در سال ۱۹۹۷، IEEE کوتاه آمد و اذعان کرد که هر دو روش قابل قبول است. خوشبختانه، تمام مقادیری که در فیلد Type درج می شد بالاتر از عدد ۱۵۰۰ بودند. بدین ترتیب قرار بر آن شد که اگر عدد درون این فیلد از ۱۵۰۰ بیشتر باشد به عنوان فیلد Type فرض شود و در غیر اینصورت به عنوان فیلد Length تعبیر گردد. حال دیگر IEEE می توانست همه را راضی نگه دارد؛ چه آنهایی که از استاندارد او استفاده می کردند و چه آنهایی که می خواستند کار با استاندارد DIX را بدون احساس کمبود ادامه بدهند!!

۴-۳-۴ الگوریتم عقب گرد نمایی

در این بخش بررسی خواهیم کرد که وقتی تصادم به وقوع می پیوندد روال تولید عدد تصادفی [به منظور انتظار و تلاش مجدد] چگونه است. الگوی ما کماکان مدل شکل ۴-۵ می باشد. پس از بروز تصادم، زمان به تعدادی برش مجزا تقسیم می شود؛ طول این برشها معادل با بیشترین زمان رفت و برگشت سیگنال بر روی کانال (یعنی 2t) است. یا در نظر گرفتن بیشترین طول کابل در اترنت، هر یک از این برشهای زمانی معادل ۵۱۲ بیت یعنی ۵۱/۲

میکروثانیه خواهد بود.

پس از اولین تصادم هر ایستگاه قبل از تلاش مجدد، به صورت تصادفی صفر یا یک برش زمانی منتظر می ماند. (یعنی بصورت تصادفی یکی از اعداد ۰ یا ۱ را تولید می کند). اگر دو ایستگاه با هم تصادم کنند و اعداد تصادفی تولید شده مشابه باشند، تصادم تکرار خواهد شد. لذا در دومین تصادم متوالی، یکی از اعداد ۰، ۱، ۲، ۳ انتخاب و بهمان تعداد برش زمانی انتظار می کشد. [یعنی اگر عدد ۲ باشد $2 \times 51/2$ میکروثانیه منتظر می ماند.] اگر سومین تصادم متوالی رخ بدهد (که احتمال چنین رخدادی ۰/۲۵ است) تعداد برشهای زمانی انتظار، عددی تصادفی بین صفر تا ۱ - 2^3 (۷-۰) خواهد بود.

بطور عام در تصادم پیاپی i ام، عددی تصادفی بین صفر تا $2^i - 1$ انتخاب می شود و متناسب با عدد انتخابی، بر مبنای برشهای $51/2$ میکروثانیه ای منتظر می ماند. با این حال پس از دهمین تصادم متوالی، بازه تولید اعداد تصادفی بین صفر تا 1024 ثابت خواهد ماند. پس از شانزدهمین تصادم پیاپی، کنترلر کارت شبکه، دیگر ادامه نداده و پیغامی مبنی بر وجود اشکال جدی در شبکه، به کامپیوتر گزارش خواهد کرد. تشخیص بیشتر، بر عهده لایه های بالاتر است.

این الگوریتم که اصطلاحاً «عقب گرد نمایی دودویی» (Binary Exponential Backoff) نامیده می شود، بدین دلیل انتخاب شده تا بتواند خود را بصورت پویا با هر تعداد ایستگاه که در تلاش برای ارسال هستند، تطبیق بدهد. اگر در هر تصادم، زمان تصادفی انتظار به صورت ثابت و در محدوده صفر تا 1023 انتخاب می شد اگر چه احتمال دو تصادم متوالی بسیار ناچیز بود ولی در عوض زمانی که ایستگاه های تصادم کننده باید صبر می کردند، بالغ بر صدها بازه زمانی می شد و تاخیر بالا می رفت. برعکس، اگر بطور دائم یکی از اعداد صفر یا یک انتخاب شود آنگاه اگر به فرض صد ایستگاه سعی در ارسال کنند تصادم های پیاپی آنقدر تکرار می شود تا زمانی که ۹۹ ایستگاه تصادفاً ۱ و یکی از آنها ۰ را انتخاب کند. این اتفاق ممکن است سالها طول بکشد! اگر الگوریتمی داشته باشیم که در آن بازه های زمانی انتظار در پی تصادمات متوالی، بطور نمایی رشد کند، این اطمینان حاصل می شود وقتی تعداد ایستگاه های آماده ارسال کم است تاخیر کمی بوجود بیاید و وقتی تعداد ایستگاه های آماده ارسال زیاد است در مدت زمان معقولی مسئله تصادم حل شود. گذاشتن سقف 1023 ، اجازه رشد بیش از اندازه زمان انتظار را نخواهد داد.

یگونه ای که قبلاً اشاره کردیم در CSMA/CD دریافت فریم نباید نمی شود [یعنی پس از ارسال فریم سالم، دریافت آن گزارش نخواهد شد]. از آنجایی که جان به در بردن از تصادم تضمین کننده عدم خرابی بیتها در اثر نویز کانال نیست، لذا در مقصد هر فریم باید بررسی و در صورت صحت، پیغام تصدیق (ACK) برگردانده شود. طبعاً پیغام نباید وصول، خودش یک فریم معمولی است که برای ارسال آن نیز همانند فریم داده [برای در اختیار گرفتن کانال] باید مبارزه شود. با این وجود با یک تغییر ساده در الگوریتم رقابت، می توان به دریافت پیغام ACK سرعت بخشید. (Tokoro and Tamaru, 1977) تمام کاری که باید انجام شود آنست که پس از خاتمه ارسال فریم، اولین بازه زمانی به ایستگاه مقصد اختصاص داده شود تا پیغام ACK خود را ارسال نماید. [فریم ACK بسیار کوتاه است]. متناسفانه در استاندارد اترنت، چنین قابلیتی گنجانیده نشده است.

۵-۳-۴ کارائی (بازده) اترنت

در اینجا اجازه بدهید کارائی اترنت را در شرایط بار سنگین و ثابت، یعنی شرایطی که در آن همیشه k ایستگاه آماده ارسال هستند، بررسی نماییم. تحلیل دقیق الگوریتم عقب گرد نمایی پیچیده است. به جای آن روش «متکالف» و «باگز» را دنبال کرده و فرض می کنیم که احتمال تکرار ارسال فریم در هر برش رقابت، ثابت باشد. اگر هر ایستگاه در خلال برش رقابت با احتمال p اقدام به ارسال فریم نماید احتمال آنکه یک ایستگاه در همان برش موفق به

ارسال شود مساویست با:

$$A = k.p.(1 - p)^{k-1}$$

A زمانی حداکثر خواهد شد که $p=1/k$ باشد؛ با فرض $p=1/k$ ، وقتی k به سمت بی نهایت میل می کند مقدار A، $1/e$ خواهد بود. احتمال آنکه دوره رقابت دقیقاً از مرحله متوالی ادامه یابد، مساوی با $A.(1-A)^{k-1}$ است فلذا میانگین تعداد دفعات تصادم در هر ارسال طبق رابطه زیر بدست می آید:

$$\sum_{j=0}^{\infty} j.A.(1-A)^{j-1} = 1/A$$

از آنجایی که هر برش زمانی حدوداً 2τ طول می کشد، میانگین زمان رقابت $W = 2\tau/A$ خواهد بود. با فرض مقدار بهینه برای p (یعنی $1/e$) میانگین تعداد برشهای رقابت هرگز از $2\tau.e$ بیشتر نخواهد شد [زیرا مقدار A در رابطه $W=2\tau/A$ حداکثر $1/e$ است] بدین ترتیب W معادل $2\tau.e$ یعنی حدود 5.45 خواهد بود.

با فرض آنکه تعداد ایستگاه های آماده ارسال، زیاد باشد و ارسال فریم، P ثانیه به طول بینجامد داریم:

$$\text{رابطه ۶-۴} \quad \text{کارآئی کانال} = \frac{P}{P + 2\tau/A}$$

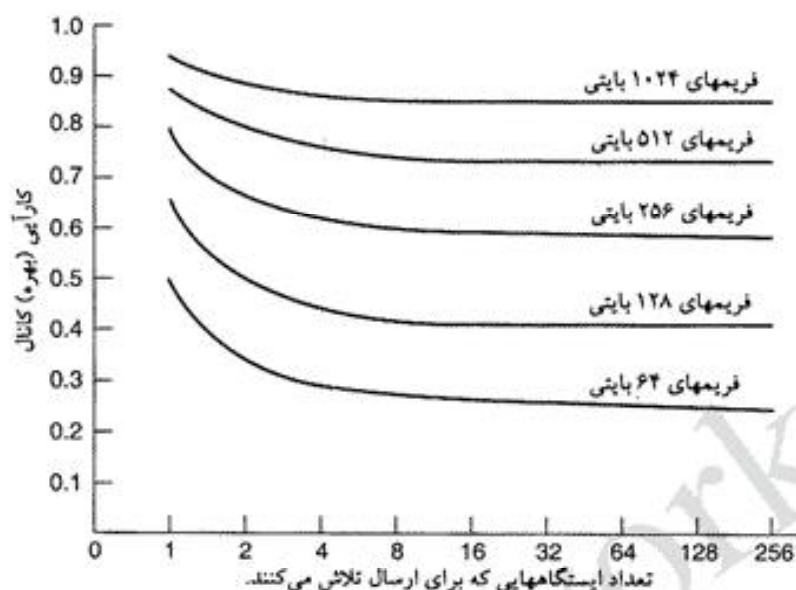
در اینجا می بینیم که فاصله حداکثر کابل بین دو ایستگاه در کارآئی این شبکه وارد می شود (زیرا مقدار τ به طول کابل بستگی دارد). این موضوع نشان می دهد که برای فواصل طولانی باید به سراغ یک توپولوژی بغیر از آنچه که در شکل ۴-۱۵-الف دیده می شود، رفت. هر چه طول کانال زیادتر باشد طول دوره رقابت بیشتر خواهد شد. این نتیجه بروشنی مشخص می کند که چرا در استاندارد اترنت، طول کابل محدودیت دارد. بهتر آنست که رابطه ۶-۴ را به صورت واضحتر و بر حسب طول فریم، F، پهنای باند شبکه، B، طول کابل، L، و سرعت انتشار سیگنال بر روی کانال، C و مقدار بهینه A یعنی $1/e$ بنویسیم. با قرار دادن $P = F/B$ معادله ۶-۴ به صورت زیر در می آید:

$$\text{معادله ۷-۴} \quad \text{کارآئی کانال} = \frac{1}{1 + 2BLE/cF}$$

وقتی دومین عبارت یعنی $(2B.L.e/c.F)$ بزرگ باشد کارآئی کانال پائین خواهد بود. بالاخص با افزایش پهنای باند یا طول کانال شبکه (یا بطور کلی با افزایش حاصل ضرب $B.L$)، کارآئی شبکه (برای فریمهای با طول مشخص) کاهش خواهد یافت. متأسفانه بیشتر پژوهشهایی که برای بالا بردن سرعت سخت افزار شبکه انجام می شود، این حاصل ضرب را افزایش می دهد زیرا امروزه عموم مردم به پهنای باند بالا و کانالهای طولانی نیاز دارند (مثلاً برای شبکه فیبر نوری MAN) و محاسبه فوق نشان می دهد که شبکه اترنت برای چنین محیط هائی مناسب نیست. در ادامه همین فصل خواهیم دید که روش دیگری برای پیاده سازی اترنت پیشنهاد شده که در مبحث اترنت مبتنی بر سوئیچ، بدان خواهیم پرداخت.

در شکل ۴-۱۹، منحنی کارآئی کانال بر حسب تعداد ایستگاه های آماده جهت ارسال و با در نظر گرفتن $2\tau = 51.2\mu s$ و نرخ ارسال $B=10Mbps$ ترسیم شده است. [هر یک از منحنی ها به ازای یک F یعنی طول فریم مشخص ترسیم شده اند.] برای فریمهای ۶۴ بیتی کارآئی کانال چندان جالب نیست. در طرف مقابل، با نظر گرفتن ۶۴ بایت برای هر برش رقابت، میانگین زمان رقابت برای فریمهای ۱۰۲۴ بیتی، معادل ۱۷۴ بایت بوده و رانده مان کانال تقریباً ۸۵٪ است.

برای آنکه تعداد متوسط ایستگاه های آماده ارسال را در بار سنگین محاسبه نماییم می توان از روش زیر بهره گرفت: هر فریم برای مدت زمانی معادل: «طول دوره رقابت به اضافه زمان ارسال فریم» یعنی به اندازه $P+W$ ثانیه، کانال را درگیر خواهد کرد. بدین ترتیب تعداد فریمها در هر ثانیه $1/(P+W)$ خواهد بود. هرگاه ایستگاهها با نرخ



شکل ۴-۱۹. منحنی کارایی کانال در اینترنت 10 Mbps با فرض برشهای رقابت ۵۱۲ بیتی.

میانگین k فریم بر ثانیه به تولید فریم مشغول باشند و سیستم «در حالت k »^۱ باشد، مجموع نرخ تولید فریم در ایستگاههای فعال k فریم بر ثانیه خواهد بود. از آنجایی که در حالت «آرامش» (Equilibrium) بایستی نرخ ورودی و خروجی یکسان باشد لذا می توانیم این دو عبارت را معادل هم قرار داده و آنرا بر حسب k حل کنیم. (دقت کنید که w خود تابعی از k است.) روش تحلیل پیچیده و دقیقتری توسط (Bertsekas and Gallager, 1992) ارائه شده است.

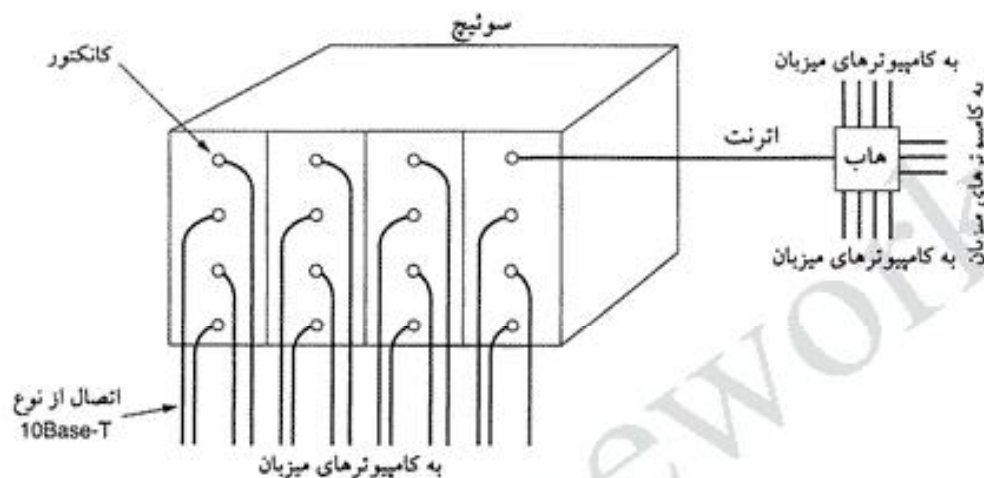
اشاره به این نکته خالی از لطف نیست که تاکنون حجم بسیار گسترده‌ای از مطالعات در خصوص تحلیل کارایی شبکه اینترنت (و شبکه‌های دیگر) انجام شده است؛ در تمام آنها مجازاً فرض شده که ترافیک ایستگاه‌ها مبتنی بر تابع توزیع پواسون تولید می‌شود. وقتی پژوهشگران مطالعات خود را به مدل حقیقی تولید ترافیک، (در عمل) معطوف کردند، مشخص شد که ترافیک شبکه به ندرت پواسون است و «ترافیک هر ایستگاه خاص فقط شبیه به خودش است»! (Paxon and Floyd, 1994; and Willinger et al., 1995) این بدین معناست که میانگین‌گیری در یکدوره طولانی از زمان، «میانگین همواره‌شده» (Smoothed Average) ترافیک را محاسبه نخواهد کرد؛ یعنی متوسط تعداد فریمها در «هر دقیقه از ساعت» دارای واریانس (پراکندگی) بیشتری نسبت به متوسط تعداد فریمها، در «یک ثانیه از هر دقیقه» خواهد داشت. این نتیجه‌گیری بدین معناست که اغلب مدل‌های عرضه شده برای ترافیک شبکه قابل اعمال در محیط‌های واقعی نیست.

۴-۳-۴ اینترنت مبتنی بر سوئیچ

هر چه بر تعداد ایستگاه‌های شبکه اینترنت افزوده شود، ترافیک شبکه افزایش یافته و سرانجام شبکه LAN اشباع خواهد شد. یکی از روشهای کاهش این مشکل آنست که از اینترنت 10Mbps به سوی اینترنت 100Mbps حرکت کنیم ولیکن با رشد کاربردهای چندرسانه‌ای، اینترنت 100Mbps یا حتی 1-Gbps نیز، اشباع خواهد شد.

۱. «حالت k » و «حالت آرامش» از اصطلاحات نظریه صف است. برای درک این مبحث باید اندکی در خصوص نظریه صف (Queuing Theory) مطالعه قبلی داشته باشید. -م

خوشبختانه راه دیگری برای مواجهه با بار سنگین وجود دارد: «اترنت مبتنی بر سوئیچ». این مدل در شکل ۴-۲۰ به تصویر کشیده شده است. در مرکز این سیستم «سوئیچ» قرار گرفته که از یک Backplane پرسرعت و فضائی برای نصب ۴ تا ۳۲ «کارت اتصال خط»^۱ تشکیل شده است و هر کارت ۱ تا ۸ کانکتور دارد. بطور معمول، کانکتورها از نوع 10Base-T هستند که از طریق سیمهای زوجی بهم تابیده، مستقیماً به یک کامپیوتر میزبان متصل می شوند.



شکل ۴-۲۰. مثالی ساده از اترنت مبتنی بر سوئیچ.

هرگاه ایستگاهی بخواهد یک فریم اترنت را بفرستد، آن فریم را در قالب عادی و استاندارد سازماندهی کرده و آنرا برای سوئیچ می فرستد. کارت واسط دریافت کننده فریم، ابتدا آدرسهای آنرا بررسی می کند تا ببیند آیا گیرنده فریم (مقصد) به همان کارت متصل است؛ اگر مقصد به همان کارت واسط متصل نبود، فریم بر روی Backplane پرسرعت ارسال می شود تا کارت واسط مقصد آنرا دریافت کند. Backplane با سرعتی حدود چندین گیگابیت در ثانیه و با پروتکلی اختصاصی کار می کند.

حال ببینیم وقتی دو ایستگاه متصل به یک کارت مشابه، به بطور همزمان فریمی را ارسال کنند چه اتفاقی می افتد؟ این مسئله به ساختار طراحی آن کارت بستگی دارد. امکان دارد که تمام پورتهای یک کارت از طریق یک سیم، مستقیماً به هم وصل شده و یک شبکه محلی (باس) تشکیل شده باشد. آنگاه، تصادماتی که ممکن است بر روی این LAN داخلی رخ بدهد بروش معمولی CSMA/CD حل و فصل شده و ارسال مجدد نیز طبق الگوریتم «عقب گرد دودویی نمایی» (Binary Exponential Backoff) انجام می گیرد. در چنین کارتهائی، از بین ایستگاههای متصل به هر کارت، در آن واحد فقط یک ایستگاه می تواند ارسال کند در حالیکه ایستگاههای متصل به کارتهای متفاوت می توانند بطور موازی [همزمان] ارسال داشته باشند. در این طرح هر کارت برای خودش یک «حوزه تصادم»^۲ (Collision Domain) تشکیل داده و این حوزه، مستقل از حوزه تصادم دیگر کارتهاست. هرگاه فقط یک ایستگاه در هر حوزه وجود داشته باشد [یعنی فقط یک ایستگاه در هر کارت] آنگاه تصادم متفی است و کارائی افزایش خواهد داشت.

در انواع دیگر کارتها، هر یک از پورتهای ورودی دارای بافری اختصاصی هستند و بدین ترتیب فریمهای

۱. Plug-in Line Card.

۲. ایستگاههایی که مستقیماً به یک کانال مشترک متصلند و برای ارسال بروش CSMA/CD رقابت می کنند اصطلاحاً در یک

«حوزه تصادم» قرار دارند. -م

ورودی در درون حافظه RAM کارت، ذخیره می شوند. در این ساختار تمام ایستگاه ها می توانند بطور همزمان و دوطرفه (Full Duplex)، ارسال و دریافت داشته باشند؛ عملی که در ساختار معمولی CSMA/CD، با یک کانال واحد ممکن نیست. هنگامی که یک فریم بطور کامل دریافت شود، کارت مربوطه می تواند بررسی کند که آیا ماشین مقصد به پورتی بر روی همان کارت متصل است یا باید به سمت پورتی بر روی کارت دیگر هدایت شود. در حالت اول، فریم مستقیماً به سمت مقصد ارسال می شود. در حالت دوم، فریم از طریق Backplane به سمت کارت مناسب هدایت می گردد. در این طراحی، هر پورت برای خود «حوزه تصادم» کاملاً مستقل دارد، یعنی هرگز تصادم رخ نمی دهد. در این ساختار، ظرفیت کل سیستم در مقایسه با 10Base5 (که در آن برای کل سیستم یک حوزه تصادم واحد وجود دارد) چندین برابر افزایش خواهد داشت.

از آنجایی که سوئیچ، فریمهای استاندارد اترنت را از پورتهای ورودی خود می پذیرد لذا می توان از برخی پورتهای در نقش «متمرکز کننده» (Concentrator) استفاده کرد. در شکل ۴-۲۰ پورت سمت چپ/بالایی سوئیچ، بطور مستقیم به یک ایستگاه واحد متصل نیست بلکه مثلاً به یک هاب ۱۲ پورت متصل است. وقتی فریمی به هاب وارد می شود، بروش معمول برای بدست آوردن کانال رقابت می کند؛ در حالیکه ممکن است تصادم رخ داده و از الگوریتم عقب گرد نمایی استفاده شود. فریمی که موفق به ارسال روی کانال شده و به سوئیچ وارد گردد، همانند یک فریم معمولی با آن برخورد می شود یعنی برای رسیدن به خط خروجی مناسب از طریق Backplane ارسال و به مقصد هدایت خواهد شد.

اگر چه هابها ارزانتر از سوئیچها هستند ولیکن با سقوط قیمت سوئیچها، هاب به سرعت از صحنه خارج می شود، ولی هنوز میراث هابها باقی است!

۴-۳-۷ اترنت سریع

در بدو آن ایام، سرعت 10Mbps برای اترنت بسیار شگفت انگیز و رویایی به نظر می رسید؛ همانگونه که ظهور مودمهای ۱۲۰۰bps در دورانی که فقط مودمهای آکوستیکی و ابتدائی ۳۰۰bps رایج بود، پدیده ای اعجاب انگیز محسوب می شد؛ ولیکن هر پدیده نوظهوری به سرعت فرسوده و محو می شود. به عنوان نتیجه قانون پارکینسون می توان منتظر زمانی بود که ارسال داده ها با نرخی معادل با پهنای باند طبیعی هر کانال ارسال شوند. (قانون پارکینسون بیان می کند که: «یک کار تا زمانی که وقت برای تکمیل شدن داشته باشد، ادامه می یابد.») برای افزایش سرعت شبکه ها، گروه های صنعتی مختلف دو طرح جدید مبتنی بر شبکه حلقه (Ring) و فیبرنوری ارائه کردند: یکی از آنها طرح شبکه FDDI^۱ و دیگری طرح Fibre Channel^۲ نام داشت. برای کوتاه کردن داستان مفصل آنها، باید اشاره کنیم که اگر چه از هر دو به عنوان ستون فقرات شبکه استفاده شد ولیکن هیچیک از آنها نتوانست برای اتصال کامپیوترهای رومیزی به کار برود، چراکه در این دو شبکه، مدیریت ایستگاه ها بسیار پیچیده بود و نیاز به تراشه های پیشرفته داشت که به گران شدن قیمت آنها می انجامید؛ سرانجام پرونده آنها به تاریخ پیوست. درسی که باید از سرنوشت اینگونه شبکه ها گرفت آنست که باید هر سیستمی را ساده و سهل طراحی کرد.

بهر تقدیر، ناکامی و شکست شبکه های LAN مبتنی بر فیبرنوری [مثل FDDI]، کمبود و خلاء اترنت سریعتر از 10Mbps را هر چه بیشتر نمایان می کرد. در بسیاری از محیطها، افراد به پهنای باند بیشتری نیاز داشتند و برای رفع این مشکل به ناچار چندین شبکه محلی 10Mbps را نصب و از طریق تکرارکننده (Repeater)، پل

۱. Fiber Distributed Data Interface

۲. نام این طرح Fibre Channel است نه Fiber Channel، زیرا ویرایشگر این طرح بریتانیایی بوده نه آمریکایی! - اندی ناننم

(Bridge) یا مسیریاب (دروازه - Gateway) به هم متصل کرده بودند. این ساختار بهم پیچیده، گاهی چنان سردرگم و پراشکال می شد که مسئول شبکه احساس می کرد این شبکه ها به وسیله آدامس بادکنکی یا نخ به هم کوک خورده اند!

چنین فضا و شرایطی، IEEE را بر آن داشت تا در سال ۱۹۹۲ کمیته 802.3 را دور هم جمع کند تا در خصوص شبکه محلی سریعتر از ده مگابیت تصمیم بگیرند. یک پیشنهاد آن بود که 802.3 بهمان شکل اصلی حفظ شده و فقط سرعت آن افزایش یابد. پیشنهاد دیگر آن بود که 802.3 بطور کل از نو تدوین شده و ویژگی های جدیدی مثل ارسال ترافیک بی درنگ و مبادله دیجیتالی سیگنال صوت به آن اضافه شود ولی در عین حال (بدلیل مسائل بازار و فروش) نام قبلی آن حفظ شود. پس از بحث و مناقشه فراوان، کمیته تصمیم گرفت که ساختار شبکه 802.3 را به همان شکل اصلی حفظ کند و فقط به سرعت آن بیفزاید.

گروهی که پیشنهاد آنها در خصوص طراحی مجدد 802.3 پذیرفته نشد برای خودشان کمیته تشکیل دادند و شبکه محلی پیشنهادی خود را با نامی دیگر استاندارد کردند که نهایتاً شبکه 802.12 پدید آمد. (کاری که معمولاً هر شخصی با دید صنعتی یا اقتصادی در چنین شرایطی انجام خواهد داد انجام طرحی است که درست و منطقی به نظر می رسد. ولی بازار چیز دیگری می گوید! تاریخ نشان داد که شبکه 802.12 که شرکت هیولت پاکارد بر روی آن سرمایه گذاری کرد اقبالی بدست نیاورد!)

به سه دلیل زیر، کمیته 802.3 تصمیم گرفت اترنت را بدون تغییر در بنیان آن، فقط توسعه بدهد:

۱. نیاز به سازگاری شبکه جدید با شبکه های اترنت موجود
 ۲. نگرانی از آنکه پروتکل جدید مشکلات پیش بینی نشده داشته باشد!
 ۳. تمایل به آنکه قبل از تغییر تکنولوژی بتوانند کار را به اتمام برسانند و مشمول زمان نشود.
- کار به سرعت انجام گرفت (البته با سرعتی که در عرف کمیته استاندارد است) و حاصل کار، استاندارد 802.3u بود که به صورت رسمی در ژوئن ۱۹۹۵ توسط IEEE تائید و معرفی شد. از دیدگاه فنی، 802.3u استاندارد جدیدی محسوب نمی شود بلکه یک ضمیمه مکمل برای استاندارد موجود 802.3 است. (تا بر سازگاری با استاندارد قبل تاکید شده باشد). از آنجایی که عموم افراد به استاندارد جدید به جای 802.3u، «اترنت سریع» (Fast Ethernet) می گویند ما نیز همین اصطلاح را به کار خواهیم گرفت.
- ایده اصلی در اترنت سریع بسیار ساده بود: تمام ویژگیها مثل قالب فریم، واسطها، قواعد و الگوریتمها را بدون تغییر نگاه داشته و فقط «زمان یک بیت» را از 100 نانوثانیه به 10 نانوثانیه کاهش بدهیم. از دیدگاه فنی، این کار در شبکه اترنت با کابل 10Base5 یا 10Base2 بسادگی میسر است و به شرط آنکه طول حداکثر کانال با ضریب ده کاهش یابد، تصادمها نیز بموقع کشف خواهد شد. با این وجود محاسن سیم کشی مبتنی بر 10Base-T (یعنی استفاده از زوج سیم های به هم تابیده) بقدری مفید و چشمگیر بود که اترنت سریع کلاً بر اساس این نوع کابل طراحی شد. لذا کل سیستم اترنت سریع، از هاب یا سوئیچ بهره می گیرد و استفاده از کابل های چندانشعابی، کانکتورهای BNC یا انشعاب های تزریقی (Vampire Tap) مجاز نیست.

ولیکن هنوز چند گزینه دیگر باقی مانده بود که بایستی برای آنها نیز تصمیم گرفته می شد؛ مهمترین آنها، انتخاب انواع سیم زوجی بود که اترنت سریع باید از آنها پشتیبانی می کرد. یکی از نامزدها کابل زوجی رده ۳ (Cat 3) بود. [کابل Cat 3 برای سیم کشی تلفن در ساختمانها به کار می رود]. استدلال این انتخاب آن بود که در دنیای غرب در هر دفتر از ساختمانهای اداری حداقل چهار جفت سیم Cat 3 (یا حتی بهتر از Cat 3) از قبل وجود دارد که در فاصله ای حدود صد متر به جعبه تقسیم تلفن کشیده شده است. بنابراین با استفاده از کابل زوجی Cat 3 این امکان وجود داشت که بدون نیاز به سیم کشی مجدد ساختمان، بتوان کامپیوترهای رومیزی را از طریق اترنت

سریع به هم متصل کرد؛ این ویژگی، امتیاز بزرگی برای اغلب سازمانها و موسسات به حساب می‌آید. بزرگترین اشکال سیم‌های زوجی Cat 3 آنست که قادر نیستند سیگنالی با نرخ تغییر 200Mbaud/sec را در فاصله صد متر حمل کنند (100Mbps در روش منچستر معادل 200Mbaud/sec است)؛ در استاندارد 10Base-T، فاصله صد متر، بیشترین فاصله مجاز کامپیوتر از هاب است. از طرفی کابل‌های رده ۵ (یعنی Cat 5) می‌توانند چنین سیگنالی را به راحتی در فاصله صد متری منتقل کنند؛ فیبر نوری حتی قادر است با نرخ بسیار بالاتر، این کار را انجام بدهد.

تصمیم نهایی بر آن شد که مطابق با مشخصات شکل ۴-۲۱، هر سه گزینه در استاندارد جدید مجاز شمرده شود، ولیکن به گزینه استفاده از سیم‌های Cat 3 بهبودهایی داده شد تا بتواند سیگنال با نرخ مورد نیاز را حمل کند.

مزایا	حداکثر طول هر قطعه	نوع کابل	نام کابل
از کابل‌های معمولی تلفن (UTP Cat 3) استفاده می‌کند.	100 m	Twisted pair	100Base-T4
ارسال دوطرفه کامل با نرخ 100Mbps و کابل UTP Cat 5	100 m	Twisted pair	100Base-TX
ارسال دوطرفه کامل با نرخ 100Mbps در فواصل طولانی	2000 m	Fiber optics	100Base-FX

شکل ۴-۲۱. کابل‌کشی اترنت سریع.

در الگوی سیم‌کشی با کابل‌های معمولی Cat 3 (از نوع UTP) که 100Base-T4 نامیده می‌شود از نرخ سیگنالینگ ۲۵ مگاهرتز استفاده شده است فلذا فقط ۲۵ درصد سریعتر از استاندارد ۲۰ مگاهرتز در اترنت استاندارد است (با مرور شکل ۴-۱۶ به یاد بیاورید که در کدینگ منچستر، هر یک از ده میلیون بیت در ثانیه نیاز به دو پالس ساعت دارد). بهر حال برای تأمین پهنای باند مورد نیاز در 100Base-T4 به چهار زوج سیم Cat 3 نیاز است. [۴×۲۵MHz] از آنجایی که سازه‌هاست در استاندارد سیم‌کشی تلفن ساختمانها، از کابل‌هایی استفاده می‌شود که دارای چهار جفت سیم هستند لذا بسیاری از ادارات چنین امکانی را در اختیار داشتند. البته این موضوع بدین معناست که مرکز تلفن را کنار بگذارید ولی در عوض هزینه بسیار کمتری برای داشتن سیستم پست الکترونیکی سریعتر صرف خواهد شد!

از بین چهار زوج سیم، از یکی به عنوان خط ورودی دائم به هاب، از یکی به عنوان خط خروجی دائم از هاب و از دوتای دیگر بصورت قابل تنظیم در جهت فعلی ارسال [از هاب به کامپیوتر یا بالعکس] استفاده می‌شود. برای صرفه‌جویی در پهنای باند مورد نیاز، از روش کدینگ منچستر استفاده نشده است چراکه با ابداع مولدهای مدرن سیگنال ساعت و در فاصله‌ای بدین کونا، اصولاً نیازی هم به کدینگ منچستر نیست. بجای آن از سیگنال‌های Ternary استفاده شده که در این کدینگ، در هر سیکل سیگنال ساعت یکی از مقادیر ۰، ۱ یا ۲ ارسال می‌شود. (یعنی پالس‌ها سه سطحی هستند). با داشتن سه زوج سیم در جهت ارسال و روش سیگنالینگ سه سطحی، ۲۷ (۳×۳×۳) سمبل مختلف قابل ارسال است؛ بدین ترتیب با ارسال هر سمبل، می‌توان ۴ بیت را (به همراه مقداری افزونگی) انتقال داد. [۴ بیت معادل ۱۶ سمبل و ۱۱ حالت افزونگی] ارسال چهار بیت در هر سیکل از سیگنال‌های ۲۵ مگاهرتزی، نرخ معادل 100 مگابیت بر ثانیه را فراهم می‌آورد. همچنین، در جفت سیم باقیمانده (زوج چهارم) یک کانال معکوس ۳۳/۳ مگابیت بر ثانیه ایجاد می‌شود. این ساختار که اصطلاحاً 8B/6T نامیده می‌شود (بمعنای آنکه هشت بیت در چهار تریٹ Trit- نگاشته می‌شود) روشی جالب و جذابی نیست ولی بهر حال با ساختار موجود سیم‌کشی، کار می‌کند.

الگوی سیم‌کشی با سیم‌های زوجی Cat 5 (که اصطلاحاً 100Base-TX نامیده می‌شود) ساده‌تر است زیرا این سیم‌ها قادر به حمل سیگنال‌هایی با نرخ ۱۲۵ مگاهرتز هستند. بدین ترتیب برای هر ایستگاه فقط به دو زوج سیم

نیاز است: یکی ورودی به هاب و دیگری خروجی از هاب. در این الگو نیز از روش کدینگ معمولی استفاده نشده است: در عوض از روش 4B/5B بهره گرفته شده که مشابه و سازگار با FDDI است. در روش 4B/5B، هر گروه متشکل از پنج کلاک متوالی (معادل پنج بیت) و شامل ۳۲ حالت مختلف است. ۱۶ تا از این ترکیبات برای ارسال ۴ بیت داده به کار می‌رود. [عبارت ساده‌تر در روش 4B/5B به ازای هر چهار بیت، پنج بیت ارسال می‌شود.] برخی از ۱۶ حالت باقیمانده برای عملیات کنترلی نظیر مشخص کردن ابتدا و انتهای فریم کاربرد دارد. ترکیبات شانزده گانه داده به نحوی انتخاب شده‌اند که در الگوی سیگنال تولیدی، لبه لازم برای سنکرون ماندن سیگنال ساعت وجود داشته باشد. سیستم 100Base-TX دو طرفه همزمان است: ایستگاه می‌تواند داده‌ها را با نرخ ۱۰۰ مگابیت در ثانیه ارسال و بطور همزمان دریافت نماید. اغلب به روشهای 100Base-TX و 100Base-T4 به اختصار 100Base-T گفته می‌شود.

آخرین گزینه، 100Base-FX است که در آن از دو رشته فیبرنوری مالتی مود (Multimode) استفاده می‌شود؛ هر یک از تارهای نوری قادر به حمل ۱۰۰ مگابیت در ثانیه به صورت همزمان هستند. مضاف بر این، فاصله بین ایستگاه و هاب تا ۲ کیلومتر قابل افزایش است.

در پاسخ به نیاز عمومی، کمیته IEEE 802 در سال ۱۹۹۷ روش کابل‌کشی جدیدی به نام 100Base-T2 به استاندارد اضافه کرد که اجازه می‌داد اترنت سریع با دو زوج سیم معمولی Cat 3 کار کند ولیکن بدلیل استفاده از روش کدینگ خاص، به یک پردازنده سیگنال دیجیتال (DSP Processor) پیچیده نیاز است که این انتخاب را اندکی گران قیمت می‌کند. اکنون از این ساختار به دلانلی مثل پیچیدگی و قیمت بالا و این حقیقت که بسیاری از ساختمان‌های اداری به سیم‌کشی با کابل Cat 5 تن داده‌اند، به ندرت استفاده می‌شود.

در 100Base-T، جهت اتصال ایستگاه‌ها به یکدیگر می‌توان از دو نوع ابزار استفاده کرد: هاب یا سوئیچ. (شکل ۴-۲) در هاب تمام خطوط ورودی (یا حداقل تمام ورودی‌هایی که به یک کارت واحد وارد می‌شوند) به صورت منطقی به هم متصل هستند و یک حوزه تصادم واحد را ایجاد می‌کنند. در این ابزار دقیقاً مثل اترنت، تمام قواعد استاندارد (نظیر الگوریتم عقب‌گرد نمایی) اعمال می‌شود، خصوصاً آنکه در هر لحظه تنها یک ایستگاه می‌تواند ارسال داشته باشد؛ بعبارت دیگر، در هاب ماهیت ارتباط، «دوطرفه غیرهمزمان» (Half Duplex) است. در یک سوئیچ هر یک از فریمهای ورودی بر روی حافظه کارت متصل به خط، بافر می‌شود و در صورت نیاز از طریق یک Backplane بسیار سریع، از کارت مبدا به کارت مقصد هدایت می‌گردد. ساختار Backplane استانداردسازی نشده و نیازی هم بدین امر نبوده است چرا که به عنوان بخشی پنهان در درون سوئیچ انجام وظیفه می‌کند. با تکیه بر تجارب گذشته می‌توان پیش‌بینی کرد که تولیدکنندگان سوئیچ به شدت در حال رقابت هستند تا Backplane سوئیچها سریعتر شده و کارآئی کل سیستم افزایش یابد. از آنجایی که کابل‌های 100Base-FX برای کشف بموقع تصادم در شبکه اترنت بیش از اندازه طولانی هستند، فلذا این کابلها لزوماً باید، به سوئیچها متصل شوند؛ بدین ترتیب هر ایستگاه برای خود یک «حوزه تصادم» مستقل پدید آورده و تصادم متفی خواهد بود. استفاده از هاب در 100Base-FX مجاز نیست.

به عنوان آخرین نکته باید اشاره کرد که تمام سوئیچها (مجازاً) می‌توانند با تلفیقی از ایستگاه‌های 10Mbps و 100Mbps کار کنند تا ارتقاء سیستمهای موجود به سیستمهای سریعتر ساده‌تر شود. در صورت افزایش نیاز یک سایت به تعداد بیشتری ایستگاه 100Mbps، تنها کافی است به تعداد لازم کارت‌های ورودی خریداری شده و درون سوئیچ نصب شود. [به شکل ۴-۲۰ نگاه کنید.] در حقیقت در خود استاندارد روشی پیش‌بینی شده تا دو ایستگاه بتوانند بر سر نرخ ارسال (10Mbps یا 100Mbps) با هم توافق کرده و ماهیت ارتباط (Full Duplex یا Half Duplex بودن ارتباط) را انتخاب نمایند. بسیاری از محصولات اترنت از این ویژگی برخوردار هستند تا

بتوانند به صورت خودکار خودشان را پیکربندی نمایند.

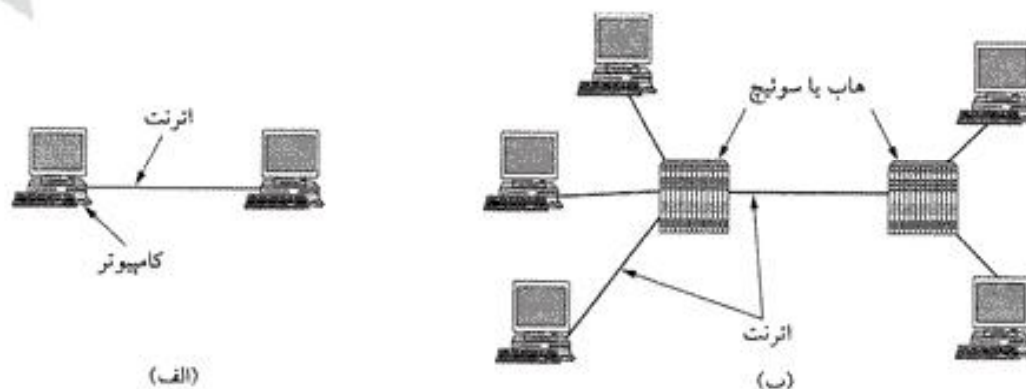
۸-۳-۴ اینترنت گیگابیت (Gigabit Ethernet)

هنوز جوهر مستندات استاندارد اینترنت سریع، خشک نشده بود که کمیته 802 کار را بر روی اینترنت سریعتر از آن آغاز کرد (۱۹۹۵). این تحقیقات سریعاً تحت عنوان اینترنت گیگابیت به ثمر رسید و در سال ۱۹۹۸ توسط IEEE با نام 802.3z تصویب شد. پیشنهاد حرف z در نام این استاندارد بدین مضمون بود که اینترنت گیگابیت آخرین حلقه در مسیر تکاملی اینترنت خواهد بود مگر آنکه کسی بتواند حرف جدیدی پس از z ابداع کند!! در ادامه برخی از ویژگی های اساسی اینترنت گیگابیت را تشریح خواهیم کرد. برای آگاهی بیشتر می توان به مقاله (Seifert, 1998) مراجعه کرد.

اهداف کمیته 802.3z مشابه با اهداف کمیته 802.3u بود: اینترنت ده برابر سریعتر شود و همچنان با استاندارد موجود اینترنت سازگاری داشته باشد. بویژه اینترنت گیگابیت باید از خدمات «انتقال دیتاگرام بدون تصدیق دریافت فریم»^۱ به صورت تک بخشی و چند بخشی^۲ و با استفاده از همان ساختار ۴۸ بیتی آدرس موجود، پشتیبانی می کرد. همچنین قالب فریمها و طول حداقل و حداکثر هر فریم باید مشابه با قبل انتخاب می شد. استاندارد نهائی، تمام این اهداف را برآورده کرده است.

کل پیکربندی شبکه اینترنت گیگابیت، به جای آنکه همانند اینترنت کلاسیک دارای ساختار باس چندانتهالی (Multidrop) باشد صرفاً نقطه به نقطه (Point To Point) است. [یعنی به جای آنکه همه ایستگاهها به یک کابل مشترک متصل باشند بطور مستقیم به یک سوئیچ متصل می شوند. -م] ساده ترین الگوی پیکربندی اینترنت گیگابیت، در شکل ۴-۲۲-الف نشان داده است؛ در این شکل دو کامپیوتر به صورت مستقیم به هم متصل شده اند. الگوی رایج دیگر که در شکل ۴-۲۲-ب دیده می شود شامل یک سوئیچ یا هاب است که به چندین کامپیوتر یا چند سوئیچ یا هاب دیگر متصل هستند. در هر دو الگو، به هر کابل اینترنت صرفاً دو ابراز متصل است، نه کمتر نه بیشتر [یعنی دو کارت اینترنت گیگابیت در دو سر کابل].

اینترنت گیگابیت می تواند به دو روش عمل کند: «حالت دوطرفه همزمان» (Full Duplex) و «حالت دوطرفه غیرهمزمان» (Half Duplex). عملکرد طبیعی و پیش فرض سیستم، حالت دو طرفه همزمان است و بدین ترتیب این امکان وجود دارد که ترافیک داده ها بتواند به صورت همزمان در دو جهت جریان داشته باشد. از این حالت زمانی استفاده می شود که یک سوئیچ مرکزی وجود داشته و به کامپیوتر دیگر (یا سوئیچهای دیگر) در پیرامون



شکل ۴-۲۲. (الف) اینترنت گیگابیت با دو ایستگاه. (ب) اینترنت گیگابیت با چند ایستگاه.

خود متصل باشد. در این پیکربندی تمام خطوط بافر شده‌اند^۱ و طبعاً هر کامپیوتر یا سوئیچ اجازه دارد هر زمان که تمایل داشت فریم خود را ارسال کند. فرستنده مجبور نیست کانال را شنود کند و حضور دیگران بر روی کانال را تشخیص بدهد چرا که در این ساختار اصولاً هرگونه رقابتی منتفی است و هرگز تصادم رخ نخواهد داد. از آنجایی که بین سوئیچ و کامپیوتر دو کابل مستقل وجود دارد، ارسال همزمان از کامپیوتر به سوئیچ (حتی اگر سوئیچ در حال ارسال فریم برای همان کامپیوتر باشد)، میسر است. از آنجایی که هیچ رقابتی مطرح نیست لذا پروتکل CSMA/CD کاربردی ندارد و حداکثر طول کانال فقط بر اساس توان سیگنال‌رسانی [و میزان تضعیف آن روی کابل] تعیین می‌شود و ربطی به مدت زمان بازگشت نویز حاصل از تصادم [و تاخیر انتشار کابل] ندارد. سوئیچها آزادند که با نرخ مختلف ارسال و به صورت تطبیقی عمل کنند. در اترنت گیگابیت همانند اترنت سریع، از پیکربندی خودکار حمایت می‌شود.

حالت دیگر عملکرد اترنت گیگابیت، حالت دو طرفه غیر همزمان (Half Duplex) است و زمانی کاربرد دارد که کامپیوترها به جای وصل به سوئیچ به هاب متصل باشند. یک هاب فریمهای ورودی را بافر نخواهد کرد و در عوض به صورت داخلی تمام خطوط را به صورت الکتریکی به هم متصل کرده و همانند اترنت کلاسیک کابلی چنداتصال را شبیه‌سازی می‌کند. در این حالت وقوع تصادم محتمل بوده و به پروتکل CSMA/CD نیاز خواهد بود. از آنجایی که کوتاه‌ترین فریم مجاز (یعنی فریم ۶۴ بایتی)، در اترنت گیگابیت صدبرابر سریعتر از اترنت کلاسیک ارسال می‌شود لذا حداکثر طول کابل بایستی ۱۰۰ برابر کاهش یابد تا آنکه ویژگی بنیادی مورد نیاز در CSMA/CD یعنی «عدم خاتمه ارسال فریم قبل از بازگشت نویز حاصل از تصادم» برآورده شود. (یعنی طول کابل باید حداکثر ۲۵ متر باشد.) با کابلی به طول ۲۵۰۰ متر و سرعت 1Gbps، قبل از آنکه یک فریم ۶۴ بایتی بتواند حتی یکدهم از مسیر خود را طی کند، فرستنده آن ارسال خود را به پایان رسانده و از تصادم احتمالی مطلع نخواهد شد!

کمیت 802.3z بدین نتیجه رسید که شعاع ۲۵ متر برای اترنت گیگابیت قابل قبول و مناسب نیست، فلذا دو ویژگی جدید به استاندارد افزود تا شعاع شبکه افزایش یابد. اولین ویژگی که «توسیع حامل» (Carrier Extension) نامیده شده سخت‌افزار را وادار می‌کند تا آنقدر اطلاعات زائد در انتهای فریم معمولی اضافه کند تا طول فریم حداقل ۵۱۲ بایت شود. از آنجایی که این اطلاعات زائد در سخت‌افزار مبداء اضافه و در سخت‌افزار مقصد حذف می‌شوند لذا نرم‌افزار از انجام چنین عملی بی‌اطلاع خواهد بود و هیچ تغییری در ساختار نرم‌افزار موجود نیاز نخواهد بود. البته طبیعی است که فرستادن فریمی ۵۱۲ بایتی برای ارسال ۴۶ بایت داده خام کاربر، بهره‌مندی معادل ۹ درصد دارد.

دومین ویژگی که اصطلاحاً Frame Bursting نامیده شده اجازه می‌دهد که فرستنده دنباله‌ای از چندین فریم را در یک بار ارسال بفرستد. اگر مجموع دنباله فریمها کمتر از ۵۱۲ بایت شود باز هم سخت‌افزار، داده‌های زائد [به انتهای آخرین فریم] اضافه می‌کند. هرگاه تعداد فریمهای منتظر ارسال کافی باشد این روش بسیار کارآمد و نسبت به روش «توسیع حامل» ارجح‌تر است. این ویژگی جدید، شعاع شبکه را به ۲۰۰ متر افزایش می‌دهد که برای بسیاری ادارات و موسسات کافی است. اگرچه در اترنت گیگابیت از CSMA/CD پشتیبانی می‌شود ولیکن تصور آن بسیار دشوار است که سازمانی کارتهای اترنت گیگابیت خریداری و نصب کند ولی آنها را به یک هاب متصل و اترنت معمولی را شبیه‌سازی نماید! [بهره واقعی اترنت گیگابیت با سوئیچها بدست می‌آید نه با هاب] اگر چه هابها از سوئیچها بسیار ارزانتر هستند ولی هنوز قیمت کارت‌های اترنت گیگابیت نسبتاً گران است.

۱. یعنی داده‌های ارسالی مستقیماً به درون حافظه موقت منتقل می‌شوند. -م

خرید هابهای ارزان قیمت [برای کارتهای گرانقیمت گیگابیتی] و کاستن از بهره واقعی سیستم، کوته فکری به نظر می رسد! بهرحال ویژگی «سازگاری با قبل» یکی از نگرانیهای صنایع کامپیوتریست و طبعاً نیاز بوده که کمیته 802.3z آنرا در استاندارد خود لحاظ کند. ولیکن دلیلی ندارد از ویژگیهای جدید آن به بهای سازگاری با قبل صرف نظر شود!

اترنت گیگابیت مطابق با فهرست ۴-۲۳ از کابلهای مسی و فیبرهای نوری پشتیبانی می کند. تولید سیگنال با نرخ نزدیک به 1-Gbps بدان معناست که منبع مولد نور باید در زمانی زیر یک نانوثانیه خاموش و روشن شود. مولد نور معمولی یعنی LED نمی تواند با این سرعت عمل کند و طبعاً به لیزر نیاز است. استفاده از پرتوهای لیزر با طول موج ۸۵۰/۰ میکرون (طول موج کوتاه) و ۱۳۰/۰ میکرون (طول موج بلند) مجاز است. لیزرهای 0.85 میکرونی ارزانتر هستند ولیکن بر روی فیبرنوری تک مود (Single Mode) کار نمی کنند.

مزایا	حداکثر طول هر قطعه	نوع کابل	نام کابل
از فیبرهای چندموده ۵۰ و ۶۲/۵ میکرون استفاده می کند.	550 m	Fiber optics	1000Base-SX
با فیبرهای تک موده ۱۰ میکرون یا چندموده ۵۰ و ۶۲/۵ میکرون	5000 m	Fiber optics	1000Base-LX
از کابلهای زوجی زره دار (STP) استفاده می کند.	25 m	2 Pairs of STP	1000Base-CX
از کابلهای استاندارد 5 Cat UTP استفاده می کند.	100 m	4 Pairs of UTP	1000Base-T

شکل ۴-۲۳. کابل کشی اترنت گیگابیت.

در اترنت گیگابیت، استفاده از فیبرهای نوری با قطر ۱۰، ۵۰ و ۶۲/۵ میکرون مجاز است. مورد اول برای فیبرنوری تک مود (Single Mode) و دو مورد بعدی برای فیبرهای چندموده هستند. تمام شش ترکیب مختلف [یعنی ترکیبات مختلف دو نوع لیزر و سه قطر] مجاز نیست ولیکن طول حداکثر کابل به ترکیب مورد استفاده وابسته است. اعدادی که در شکل ۴-۲۳ ارائه شده اند برای بهترین حالت ممکن هستند. رسیدن به طول کابل ۵۰۰۰ متری فقط با لیزر ۱۳۰/۰ میکرون و صرفاً با کابل تک موده به قطر ۱۰ میکرون امکان پذیر است ولیکن این انتخاب علیرغم قیمت گران آن بهترین گزینه برای پیاده سازی ستون فقرات شبکه های ناحیه ای (Campus) محسوب می شود.

در الگوی 1000Base-CX، از کابلهای مسی زره دار کوتاه (STP) استفاده شده است. این انتخاب از یک طرف با فیبرنوری با کارآئی بالا و از طرف دیگر با سیم های ارزان قیمت UTP در رقابت است زیرا نه به ارزان UTP است و نه به کارآمدی فیبر نوری! لذا احتمالاً از آن استقبال نخواهد شد.

آخرین انتخاب، استفاده از چهار زوج سیم 5 Cat است که همزمان با یکدیگر کار می کنند. بدلیل آنکه حجم زیادی از این نوع کابل از قبل نصب شده لذا می توان از آن به عنوان گونه فقیرانه و کم خرج اترنت گیگابیت یاد کرد! اترنت گیگابیت برای کدینگ سیگنال روی فیبرنوری از روشی جدید استفاده می کند. بهره گیری از روش منچستر با سرعت 1-Gbps، نیاز به تغییر در سطح سیگنال با نرخ معادل 2-G baud/sec خواهد داشت که رسیدن به آن بسیار دشوار بوده و نیاز به پهنای باند بسیار زیادی دارد. در عوض از روشی جدید به نام 8B/10B استفاده شده است. در این روش هر بایت هشت بیتی قبل از ارسال بر روی فیبرنوری به یک الگوی ده بیتی نگاشته می شود؛ به همین دلیل نام آن 8B/10B انتخاب شده است. از آنجایی که کلمه کد ده بیتی خروجی (که به ازای هر بایت ورودی تولید می شود) دارای ۱۰۲۴ حالت مختلف است لذا برای هر یک از ۲۵۶ حالت مختلف ورودی می توان یک کلمه کد، متناسب با شرایط کانال انتخاب کرد. برای انتخاب کلمه کد، دو قاعده زیر به کار گرفته می شود:

۱. هیچ کلمه کدی نباید بیش از ۴ بیت مشابه و پشت سرهم داشته باشد.

۲. هیچ کلمه کدی نباید جمعاً بیش از ۶ بیت صفر یا ۶ بیت یک داشته باشد.

این معیارها باعث خواهد شد که در جریان سیگنال تولید شده در خروجی بقدر کافی لبه (Transition) وجود داشته باشد تا این اطمینان حاصل شود که گیرنده با فرستنده هماهنگ و سنکرون باقی مانده و تعداد صفرها و یکهای ارسالی بر روی فیبر، حتی الامکان مساوی یا نزدیک به هم باشد. مضاف بر این بسیاری از بایتهای ورودی می توانند بیش از یک کلمه کد هم ارز داشته باشند. وقتی که کدکننده، در انتخاب کلمه کد آزادی انتخاب داشته باشد می تواند کلمه کدی را انتخاب کند که برآیند تعداد صفرها و یکهایی که تاکنون ارسال شده اند حتی الامکان معادل باشد. تاکید می شود که بر روی معادل بودن تعداد صفرها و یکها وجود دارد از آن جهت است که مولفه DC سیگنال حتی الامکان پائین بوده و در صورت عبور از مبدلها (Transformers) تغییری در شکل سیگنال ایجاد نشود. اگر چه دانشمندان کامپیوتر تمایلی ندارند که خصوصیات یک ترانسفورمر، نوع کدینگ آنها را تعیین کند ولیکن بهر حال نکته ای است که باید رعایت شود.

در اترنت گیگابیت برای الگوی سیم کشی 1000Base-T، از روش کدینگ متفاوتی بهره گرفته شده است چراکه ارسال پالسهای داده بر روی سیم مسی در زمان یک نانو ثانیه بسیار دشوار است. در کدینگ جدید از چهار زوج سیم Cat 5 استفاده شده تا چهار سمبل به صورت همزمان و موازی ارسال شوند. هر سمبول با پنج سطح ولتاژ متفاوت کد می شود. این الگو اجازه می دهد تا یک سمبول بتواند یکی از حالات 00، 01، 10، یا 11 و یک حالت کنترل خاص را کد نماید. بنابراین در هر سیکل سیگنال ساعت، بر روی هر زوج سیم ۲ بیت و بر روی کل سیمها ۸ بیت ارسال می شود. سیگنال ساعت در فرکانس ۱۲۵ مگاهرتز کار می کند که اجازه ارسال یک گیگابیت در هر ثانیه را خواهد داد. دلیل آنکه بجای استفاده از چهار سطح ولتاژ از پنج سطح استفاده شده آنست که برخی از ترکیبات آن برای عملیات کنترلی و فریمینگ باقی بماند.

سرعت 1 Gbps بسیار سریع و بالاست: به عنوان مثال هرگاه یک گیرنده، فقط برای یک میلی ثانیه به کاری دیگر مشغول باشد و نتواند بافر ورودی یکی از خطوط را خالی کند، در این وقفه یک میلی ثانیه ای، ۱۹۵۳ فریم در بافر جمع خواهد شد!! همچنین اگر کامپیوتری در شبکه اترنت گیگابیت، داده های را برای کامپیوتری در شبکه اترنت کلاسیک بفرستد به احتمال بسیار زیاد داده ها در بافر روی هم نوشته شده و از دست خواهد رفت. پیامد این دو پدیده آن بود که در اترنت گیگابیت از کنترل جریان (Flow Control) پشتیبانی شود. (همچنان که در اترنت سریع نیز کنترل جریان وجود دارد ولیکن بروشی متفاوت)

برای عملیات کنترل جریان یکی از طرفین با ارسال یک فریم کنترلی خاص، از طرف مقابل خود می خواهد که برای مدتی ارسال داده را متوقف نماید. فریمهای کنترلی، فریمهای معمولی اترنت هستند که در فیلد Type آنها عدد 0x8808 قرار گرفته است. در این صورت، دو بایت ابتدائی از فیلد داده نوع فرمان را مشخص می کند و بایتهای بعدی به عنوان پارامتر آن فرمان تلقی می شوند (در صورت وجود). برای کنترل جریان، فریم PAUSE به کار می رود و پارامتر آن مدت زمان توقف را (بر مبنای طول حداقل فریم) مشخص می نماید. در اترنت گیگابیت واحد زمان ۵۱۲ نانو ثانیه است و فریم PAUSE می تواند ایستگاه را تا ۳۳/۶ میلی ثانیه متوقف کند.

پس از آنکه اترنت گیگابیت استاندارد شد کمیته ۸۰۲ که خسته شده بود می خواست پی کار خود برود ولی IEEE از آنها خواست که کار را بر روی اترنت ده گیگابیت شروع کنند. پس از جستجوی سخت برای حرفی که بتواند جایگزین z [در 802.3z] شود آنها به این نتیجه رسیدند که از پسوند دو حرفی استفاده کنند!! کار آنها نتیجه داد و استاندارد مربوطه با نام 802.3ae به تایید IEEE رسید. آیا اترنت صد گیگابیت بر ثانیه نیز می تواند محقق شود؟

۹-۳-۴ IEEE 802.2: کنترل منطقی لینک

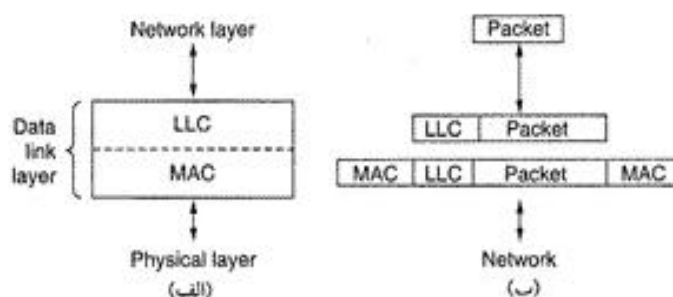
شاید زمان آن فرا رسیده باشد تا اندکی به عقب بازگردیم و برآنچه که در این فصل و ماقبل آن آموختیم، مروری تطبیقی و مقایسه‌ای داشته باشیم. در فصل ۳ آموختیم که چگونه دو ماشین می‌توانند با استفاده از پروتکل‌های پیوند داده، بر روی خطی غیرقابل اعتماد، به صورت مطمئن مبادله اطلاعات کنند. این پروتکلها امکان نظارت برخطا (با استفاده از تصدیق دریافت فریم - Ack) و کنترل جریان (با استفاده از پنجره لغزان) را فراهم آورده‌اند.

برعکس، در این فصل در خصوص مبادله مطمئن داده‌ها سخنی به میان نیاوردیم. تمام گونه‌های اترنت و دیگر پروتکل‌های ۸۰۲ تلاش می‌کنند سرویس دیتاگرام عرضه کنند. [یعنی ارسال داده‌ها بدون تصدیق دریافت آنها -Ack]. در اغلب موارد، این سرویس کافی به نظر می‌رسد. به عنوان مثال برای انتقال بسته‌های IP، به هیچ تضمینی در رسیدن بسته‌ها نیازی نیست. یک بسته IP را می‌توان بسادگی درون فیلد حمل داده از فریم ۸۰۲ قرار داد و آنرا ارسال کرد. اگر فریم از بین برود مهم نیست.

علیرغم این، سیستم‌هایی وجود دارند که به پروتکلی با قابلیت‌های نظارت برخطا و کنترل جریان نیاز دارند. IEEE پروتکلی تعریف کرده که می‌تواند بر روی پروتکل اترنت یا هر پروتکل سری ۸۰۲ قرار گرفته و اجرا شود. این پروتکل که LLC (Logical Link Control) نامیده می‌شود، قادر است تفاوت‌های انواع مختلف شبکه‌های ۸۰۲ را از طریق تعریف یک قالب و واسطه (Interface) واحد و مشترک مخفی کند. این پروتکل بسیار شبیه به پروتکل HDLC است که در فصل ۳ آنرا بررسی کردیم. LLC نیمه بالایی لایه پیوند داده‌ها را تشکیل می‌دهد، در حالیکه زیرلایه MAC، نیمه پایینی این لایه محسوب می‌شود. (شکل ۴-۲۴)

استفاده رایج از LLC بدین ترتیب است: لایه شبکه در ماشین فرستنده، بسته‌ای را یکمک توابع پایه و بنیادی لایه LLC، بدان تحویل می‌دهد. زیرلایه LLC سرآیند لازم را به بسته می‌افزاید؛ این سرآیند شامل شماره ترتیب (Seq No) و شماره تصدیق (Ack No) است. بسته حاصل، درون فیلد حمل داده از فریم ۸۰۲ قرار گرفته و ارسال می‌شود. در گیرنده نیز عکس این فرآیند انجام می‌شود.

LLC از سه رده خدمات حمایت می‌کند: (۱) خدمات ارسال نامطمئن دیتاگرام، (۲) خدمات دیتاگرام بانصدیق وصول و (۳) خدمات ارسال اتصال‌گرای مطمئن. سرآیند LLC سه فیلد را در برمی‌گیرد: «نقطه دسترسی در مقصد»، «نقطه دسترسی در مبدا» و «فیلد کنترل». نقطه دسترسی مشخص می‌کند که این فریم از چه پروسه‌ای آمده و به کدام پروسه باید تحویل شود که در حقیقت نقش همان فیلد Type در فریم DIX را ایفاء می‌کند. فیلد کنترل نیز در برگیرنده شماره ترتیب و شماره تصدیق است که شباهت فراوانی با ساختار HDLC دارد (شکل ۳-۲۴)، ولی کاملاً با آن یکسان نیست. از این فیلدها زمانی استفاده می‌شود که در سطح لایه پیوند داده‌ها به یک اتصال مطمئن (Reliable Connection) نیاز باشد که در این حالت شبیه به روشی عمل می‌شود که در فصل سوم تشریح شد. برای شبکه اترنت، صرف تلاش در تحویل بسته IP کفایت می‌کند و به هیچ بیغام اعلام وصول فریم در سطح LLC نیازی نیست.



شکل ۴-۲۴. (الف) موقعیت LLC. (ب) پشته پروتکلی (ب) قالب‌های پروتکل.

۴-۳-۱۰ نگاهی به گذشته اینترنت

اینترنت برای حدود ۲۰ سال در صحنه بوده و تقریباً هیچ رقیب جدی نداشته است و به نظر می‌رسد در سالهای آتی نیز همچنان یک‌تاز باشد. تعداد بسیار کمی سیستم عامل، زبان برنامه نویسی یا معماری CPU وجود داشته که بتواند برای دو دهه متوالی بر قله افتخار بایستد و در حال ورود به دهه سوم افتخار خود باشد. روشن است که اینترنت حرفه‌ای برای گفتن داشته که بدین گونه دوام آورده است؛ اینها چه بوده‌اند؟

شاید دلیل اصلی بقای اینترنت «سادگی» و «قابلیت اعتماد» آن بوده است. در عمل، معیار «سادگی» به «قابلیت اعتماد»، «ارزانی»، «سهولت نصب و نگهداری» تعبیر می‌شود. وقتی در اینترنت انشعابات تزیینی (Vampire Tap) با کانکتورهای BNC عوض شد خرابیها به شدت کاهش یافت. بطور معمول، عموم افراد حاضر نیستند از ابزارهایی که بخوبی کار می‌کنند رو بگردانند و این پافشاری از آنجاست که هر همه روشن شده بسیاری از محصولات بنجل در صنعت کامپیوتر باهیا هو می‌آیند ولی بسیار ضعیف عمل می‌کنند، حتی گاهی محصولاتی که با عنوان «ارتقاء» (Upgrade) معرفی می‌شوند بسیار بدتر و ناسازگارتر از وقتی عمل می‌کنند که بطور کامل عوض شوند! ولیکن نسخه‌های ارتقایافته اینترنت، اعتماد عمومی را جلب کرد.

«سادگی» [در مورد اینترنت] به ارزان بودن نیز تعبیر می‌شود. اینترنت نازک و سیم‌های زوجی نسبتاً کم بها و ارزان هستند. کارت‌های واسطه اینترنت نیز بسیار ارزانند. فقط در دورانی که هاب و سوئیچ معرفی شد، به مقداری سرمایه‌گذاری نیاز داشت ولیکن در زمان عرضه آنها، شبکه اینترنت بقدر کافی جا افتاده بود.

اینترنت از لحاظ نصب و نگهداری ساده است. هیچ نرم‌افزار اضافی نباید نصب شود (مگر نرم‌افزارهای بسیار کوچک راه‌انداز آنها) و به هیچ جدول یا تنظیمات پیکربندی خاصی نیاز ندارد که مدیریت آن (یا هرگونه اشتباه در تنظیم آن) کار را دشوار کند. در ضمن اضافه کردن یک ماشین جدید به شبکه به سادگی وصل آن به هاب یا سوئیچ است و کار چندانی ندارد.

امتیاز دیگر اینترنت آن بود که بسادگی با TCP/IP که پروتکل غالب دنیاست، کار می‌کند. IP پروتکلی بدون اتصال (Connectionless) و دقیقاً متناسب با اینترنت است که آن هم بدون اتصال عمل می‌کند. در مقابل، IP سازگاری بسیار کمی با ATM (که اتصال‌گراست) داشت و این عدم سازگاری به موقعیت ATM لطمه بسیار فراوانی زد.

در آخر آنکه اینترنت ظرفیت پیشرفت و بهبود فراوانی از خود نشان داد. سرعت آن چند ده برابر شد، هاب و سوئیچ عرضه گردید ولی با تمام این تغییرات نیازی به تغییر در نرم‌افزار نبود. تصور کنید وقتی یک عرضه‌کننده محصولات شبکه، تجهیزات مفصلی را ارائه کرده و می‌گوید: «این شبکه جدید و عالی را برای شما تدارک دیده‌ایم. برای استفاده از آن باید زحمت بکشید و سخت‌افزارهای قبلی خود را دور ریخته و تمام نرم‌افزارهای خود را از نو بنویسید!! او در فروش شبکه خود قطعاً به مشکل برخورد خواهد خورد!!! اینترنت چنین مشکلی نداشت. شبکه‌هایی مثل FDDI، Fibre Channel و ATM در زمان عرضه بسیار سریعتر از اینترنت بودند ولیکن هیچکدام از آنها با اینترنت سازگار نبودند، بسیار پیچیده‌تر و مدیریت آنها دشوارتر از اینترنت بود. سرانجام وقتی سرعت اینترنت بهبود یافت این شبکه‌ها دیگر هیچ مزیتی نداشتند و سریعاً از میان رفتند؛ البته بجز ATM که آن هم در هسته سیستم‌های تلفنی به کار گرفته شده بود.

۴-۴ شبکه‌های محلی بی‌سیم

اگر چه اینترنت در سطح گسترده‌ای رایج است ولی رقیب جدیدی برای آن در حال ظهور است. شبکه‌های بی‌سیم

در حال رواج هستند و بطور فزاینده‌ای در دفاتر اداری، فروشگاه‌ها و دیگر مکانهای عمومی به کار گرفته می‌شوند. شبکه‌های بی‌سیم به نحوی که در شکل ۱-۳۵ دیدیم به دو روش پیکربندی می‌شوند: «در کنار یک ایستگاه ثابت» و «بدون ایستگاه ثابت». استاندارد 802.11 هر دوی این پیکربندیها را مد نظر قرار داده و برای هر دو، تدارک لازم را دیده است.

در بخش ۱-۵-۴ پیش‌زمینه‌ای از 802.11 ارائه نمودیم. حال زمان آن رسیده تا نگاهی دقیقتر به این تکنولوژی بیندازیم. در بخشهای آتی نگاهی به پشته پروتکلی، تکنیکهای ارسال رادیویی در لایه فیزیکی، پروتکل زیرلایه MAC، ساختار فریم و خدمات ارائه شده در این شبکه خواهیم انداخت. برای کسب آگاهی بیشتر در خصوص 802.11 به مراجع زیر مراجعه نمایید:

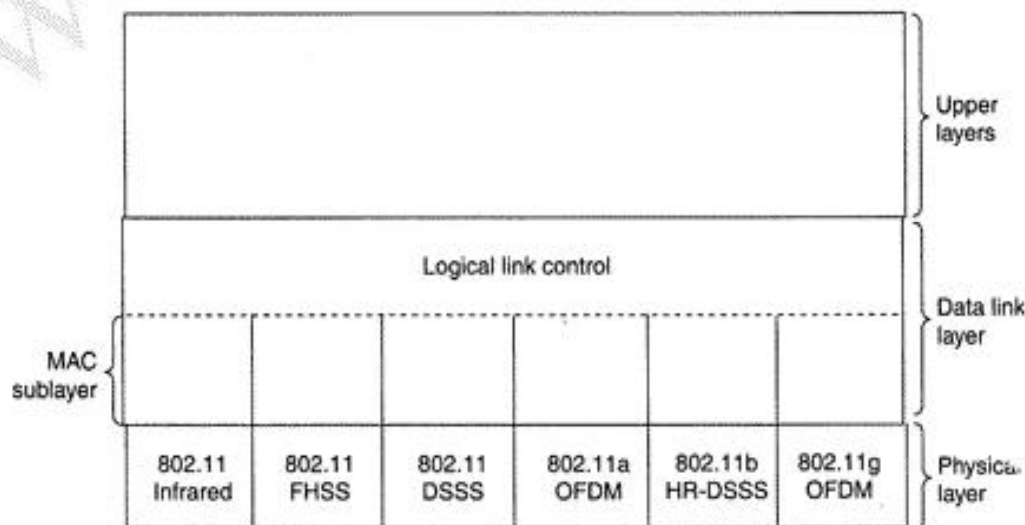
(Crow et al., 1997; Geier, 2002; Heegard et al., 2001; Kapp, 2002; O'Hara & Petric, 1999; and Severance, 1999)

برای شنیدن حقیقت درست و مسلم این شبکه، باید مستقیماً به استاندارد IEEE 802.11 مراجعه نمایید.

۱-۴-۴ پشته پروتکلی 802.11

پروتکل‌هایی که در استانداردهای سری ۸۰۲ و از جمله اترنت به کار گرفته شده مشترکات ساختاری فراوان دارند. شمای کلی پشته پروتکلی 802.11، در شکل ۴-۲۵ نمایش داده شده است. لایه فیزیکی به خوبی هم تراز با لایه فیزیکی از مدل OSI است در حالیکه لایه پیوند داده از دو لایه مستقل تشکیل شده است. در 802.11 زیرلایه MAC (زیرلایه کنترل دسترسی به کانال)، چگونگی دسترسی و تخصیص کانال، یعنی ایستگاهی را که باید در ادامه ارسال داشته باشد، مشخص می‌کند. بر روی آن، زیرلایه LLC قرار می‌گیرد که وظیفه اصلی آن مخفی کردن تفاوت‌های موجود در گونه‌های مختلف ۸۰۲ است، بگونه‌ای که این تفاوتها برای لایه شبکه [لایه سوم در مدل OSI] مخفی و غیر قابل تشخیص باشد. در همین فصل وقتی اترنت را بررسی می‌کردیم، LLC را نیز مطالعه نمودیم و نیازی به تکرار آنها نیست.

در سال ۱۹۹۷ استاندارد 802.11 سه تکنیک مختلف انتقال رادیویی، برای به کارگیری در لایه فیزیکی معرفی کرد. مثلاً روش مبتنی بر امواج مادون قرمز، بسیار شبیه به روشی است که در کنترل از راه دور تلویزیونها به کاررفته



شکل ۴-۲۵. بخشی از پشته پروتکلی 802.11.

است. در روش دیگر از امواج رادیویی برد کوتاه و از تکنیک هایی به نام FHSS و DSSS بهره گرفته شده است. هر دوی این روشها از محدوده ای در طیف فرکانس استفاده می کنند که نیازی به اخذ مجوز از دولت ندارد (باند 2.4 GHz). به عنوان مثال دریاکن های کنترل از راه دور نیز از همین باند فرکانسی استفاده می کنند لذا کامپیوتر کیفی شما ممکن است در حین استفاده از کانال، خودش را رقیب درب گاراژ شما ببیند!!! تلفن های بی سیم و اجاق های مایکروویو نیز از همین باند فرکانسی بهره گرفته اند.

تکنیک های ارسال رادیویی با نرخ ۱ تا ۲ مگابیت در ثانیه و با توان بسیار کمی عمل می کنند تا تداخل رادیویی این ابزارها با یکدیگر حداقل باشد. در سال ۱۹۹۹ دو تکنیک جدید معرفی شد تا پهنای باند (نرخ ارسال) آن افزایش یابد. این دو تکنیک جدید OFDM و HR-DSSS نامیده شده اند و به ترتیب با سرعت های ۵۴ و ۱۱ مگابیت بر ثانیه عمل می کنند. در سال ۲۰۰۱ گونه دومی از مدولاسیون OFDM ولی در باند فرکانسی متفاوت نسبت به OFDM اولیه، معرفی شد. در ادامه بطور مختصر آنها را بررسی خواهیم کرد. از دیدگاه فنی این تکنیکها به لایه فیزیکی تعلق دارند و باید در فصل دوم بررسی می شدند ولی از آنجایی که این تکنیکها به LAN و خصوصاً زیرلایه MAC 802.11 وابسته هستند، در این بخش بدانها پرداخته ایم.

۲-۴-۴ لایه فیزیکی در 802.11

تمام پنج تکنیک انتقال رادیویی، این امکان را فراهم کرده اند که یک فریم MAC از ایستگاهی به ایستگاه دیگر منتقل شود. تفاوت های آنها در تکنولوژی به کاررفته و سرعت قابل حصول آنهاست. پرداختن به جزئیات این روشها از حوصله این کتاب خارج است ولیکن چند کلمه صحبت در مورد آنها و بخصوص معرفی کلمات کلیدی آن می تواند به علاقمندان کمک کند تا بتوانند برای کسب آگاهی بیشتر، در اینترنت یا مراجع دیگر جستجو کنند. در گزینه «امواج مادون قرمز» از امواج پخش (Diffused) با طول موج ۸۵۰/۹۵۰ میکرون بهره گرفته شده است. در این روش سرعت های ۱ و ۲ مگابیت در ثانیه مجاز می باشد. در نرخ 1 Mbps از روش کدینگ خاصی به نام Gray Code استفاده شده که در آن گروه های ۴ بیتی به یک کلمه کد ۱۶ بیتی تبدیل می شوند، به نحوی که در این کلمه ۱۶ بیتی تنها یک بیت ۱ و پانزده بیت ۰ وجود دارد. این کد دارای این ویژگی اساسی است که خطائی کوچک در سنکرونیزاسیون زمان، تنها یک بیت خطا در خروجی ایجاد خواهد کرد. در سرعت 2 Mbps، دو بیت اخذ و یک کد چهار بیتی تولید می شود که در آن تنها یک بیت ۱ وجود دارد. (یعنی هر دو بیت به یکی از چهار حالت 0001, 0100, 1000 نگاشته می شود). سیگنال های مادون قرمز نمی توانند در موانعی مثل دیوار نفوذ کنند لذا سلول هایی که در اتاقهای مختلف ایجاد می شوند کاملاً از هم جدا و تفکیک شده هستند. علیرغم این، بدلیل نرخ ارسال پائین (و این حقیقت که نور خورشید، امواج مادون قرمز را در خود غرق و محو می کند) از این گزینه استقبال چندانی نشد.

در FHSS (Frequency Hopping Spread Spectrum) از ۷۹ کانال مستقل استفاده شده که هر یک از این کانالها 1 MHz پهنای باند دارند و از پائین ترین فرکانس باند 2.4 GHz ISM شروع می شوند. برای مشخص کردن دنباله فرکانسهایی که باید بدانها پرش شود از یک مولد اعداد شبه تصادفی استفاده شده است. مادامیکه تمام ایستگاهها در الگوریتم مولد اعداد از نقطه شروع (Seed) یکسانی استفاده کنند و از لحاظ زمانی با هم سنکرون باشند همگی بطور همزمان به فرکانسهای یکسانی پرش خواهند کرد. مدت زمانی که ایستگاهها در یک فرکانس خاص باقی می مانند، اصطلاحاً dwell time نامیده می شود و پارامتری قابل تنظیم است ولیکن باید کمتر از ۴۰۰ میلی ثانیه باشد. استفاده تصادفی از باندهای فرکانسی، روش مناسبی برای تخصیص کانال بروشی غیر معمول در باند ISM است. این ویژگی کم و بیش به امنیت اطلاعات کمک خواهد کرد چرا که اگر یک اخلالگر ترتیب پرشهای فرکانس یا پارامتر Dwell time را نداند، نمی تواند اطلاعات کانال را استراق سمع کند. در فواصل دور.

محوشدگی سیگنال بدلیل «چندمسیره شدن سیگنال» (Multipath Fading) اشکال عمده‌ای به حساب می‌آید و خوشبختانه FHSS در مقابله با این مشکل، موفق عمل می‌کند. همچنین این روش نسبت به تداخل رادیویی، نسبتاً حساس نیست و برای ایجاد لینک بین ساختمانها بسیار مناسب خواهد بود. بزرگترین اشکال این روش پهنای باند کم آنست. (1 Mbps)

سومین روش مدولاسیون یعنی DSSS (Direct Sequence Spread Spectrum) نیز به یکی از نرخ‌های ۱ یا ۲ مگابیت بر ثانیه محدود شده است. تکنیک به کار رفته در DSSS تا حدودی مشابه به سیستم CDMA است که در بخش ۲.۶.۲ بررسی شد ولیکن از برخی جهات با آن تفاوتی دارد. هر بیت در قالب یازده Chips ارسال می‌شود که به دنباله بارکر (Barker Sequence) مشهور است. در این روش از مدولاسیون تغییر فاز (Phase Shift) با نرخ تغییر 1 Mbaud استفاده شده که وقتی در نرخ 1 Mbps عمل می‌کند در هر تغییر فاز یک بیت ولی وقتی در نرخ 2 Mbps عمل می‌کند در هر تغییر فاز، دو بیت منتقل می‌گردد. برای سالیهای متمادی سازمان [FCC سازمان تخصیص فرکانس] فقط اجازه می‌داد که تجهیزات مخابرات بی‌سیم در ایالات متحده، صرفاً از باند ISM و طیف گسترده (Spread Spectrum) استفاده کنند ولی ظهور تکنولوژیهای جدید، به لغو این قانون در سال ۲۰۰۲ انجامید.

اولین شبکه محلی بی‌سیم پرسرعت یعنی 802.11a، با بهره‌گیری از مدولاسیون OFDM^۱ در باند فرکانسی 5-GHz عمل می‌کرد تا به سرعت 54 Mbps دست یابد. اگر بخواهیم با استعارات FDM سخن بگوئیم، در OFDM از ۵۲ زیرکانال فرکانسی استفاده شده که ۴۸ تا از آنها برای داده و ۴ تا برای سنکرونیزاسیون کاربرد دارد و بی‌شبهت به ADSL نیست. از آنجایی که ارسال، بطور همزمان بر روی فرکانسهای متفاوتی انجام می‌شود لذا این روش گونه‌ای از روشهای مبتنی بر «طیف گسترده» محسوب می‌شود ولیکن با روشهای CDMA یا FHSS کاملاً متفاوت است. تقسیم سیگنال به تعداد بسیار زیادی باند باریک در مقایسه با استفاده از یک باند عریض و واحد، مزایای بسیار مهمی در بردارد که از جمله می‌توان به ایمنی بیشتر در مقابل تداخل امواج باند باریک و امکان استفاده از باندهای غیر مجاور (noncontiguous band) اشاره کرد. در این روش از سیستم کدینگ پیچیده‌ای مبتنی بر مدولاسیون تغییر فاز برای سرعت زیر 18 Mbps و مدولاسیون QAM برای سرعت‌های بالاتر استفاده شده است. در سرعت 54 Mbps، ۲۱۶ بیت داده به سمبول‌های ۲۸۸ بیتی کد می‌شود. بخشی از انگیزه‌های خلق OFDM سازگاری آن با سیستم اروپائی HiperLAN/2 بوده است. (Doufexi et al., 2002) این روش دارای کارائی بسیار بالائی در استفاده از طیف فرکانسی (برحسب bits/HZ) بوده و ایمنی خوبی در مقابل پدیده «محوشدگی ناشی از مسیرهای چندگانه» دارد.

نهایتاً به HR-DSSS (High Rate Direct Spread Spectrum) می‌رسیم که روشی دیگر مبتنی بر تکنیک طیف گسترده است و با به کارگیری 11 million chips/sec در باند 2.4 GHz، به نرخ ارسال یازده مگابیت در ثانیه رسیده است. این روش 802.11b نامیده شده ولیکن دنباله روی 802.11a نبوده است. در این روش از نرخهای ۱، ۲، ۵/۵ و ۱۱ مگابیت در ثانیه حمایت می‌شود. دو نرخ ارسال پائین [یعنی ۱ و ۲ مگابیت بر ثانیه] به ترتیب از سیگنالی با نرخ تغییر 1 Mbaud/sec بهره می‌گیرد (یعنی در هر تغییر فاز، ۱ یا ۲ بیت کد می‌شود). در HR-DSSS به منظور سازگاری با DSSS، از روش مدولاسیون تغییر فاز بهره گرفته شده است. برای نرخ ارسال سریعتر (۵/۵ و ۱۱ مگابیت بر ثانیه) از سیگنالی با نرخ تغییر 1.375 Mbaud/sec استفاده شده و در هر تغییر به ترتیب ۴ یا ۸ بیت کد می‌شود؛ کدها از نوع Walsh/Hadamard هستند. نرخ ارسال به صورت پویا و در خلال عملیات شبکه تعیین

می شود تا سرعت بهینه بر اساس شرایط فعلی حاکم بر شبکه (شامل نویز محیط و بار) تنظیم گردد. در عمل، سرعت شبکه 802.11b تقریباً همیشه 11 Mbps است. اگر چه سرعت 802.11b از سرعت 802.11a کمتر می باشد ولیکن بُرد این شبکه حدوداً هفت برابر بیشتر است که این ویژگی در بسیاری از محیطها اهمیت بسزائی دارد.

نسخه بهبود یافته 802.11b یعنی 802.11g، در نوامبر سال ۲۰۰۱ (پس از کشمکش فراوان بر سر آنکه از کدام تکنولوژی استفاده شود) به تائید IEEE رسید. این استاندارد از مدولاسیون OFDM (به کار رفته در 802.11a) بهره می گیرد ولیکن مثل 802.11b در باند 2.4 GHz عمل می کند. از نظر تنوری این سیستم می تواند در سرعت 54 Mbps عمل کند، ولیکن هنوز روشن نیست که آیا این سرعت در عمل نیز محقق خواهد شد یا خیر. بدین ترتیب کمیته 802.11 سه شبکه محلی پرسرعت بی سیم معرفی کرده است. 802.11a، 802.11b و 802.11g (به سه شبکه کدتر فعال کاری نداریم) شاید این سوال درست به ذهن شما خطور کند که آیا تعریف سه استاندارد متفاوت کار درستی است؟ شاید عدد ۳، عدد طلایی شانس باشد!

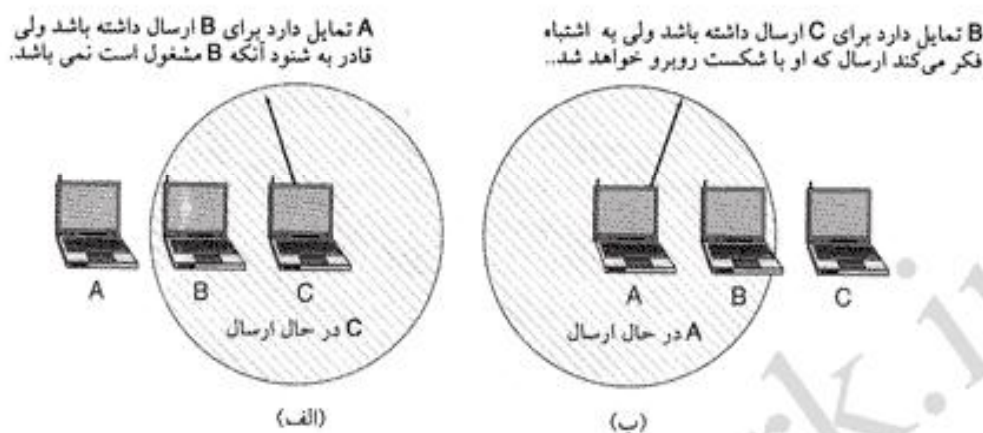
۳-۴-۴ پروتکل زیرلایه MAC در 802.11

حال اجازه بدهید از فضای مهندسی برق به سرزمین مهندسی کامپیوتر برگردیم. پروتکل زیرلایه MAC در 802.11 کاملاً با اترنت تفاوت دارد زیرا شرایط حاکم بر محیطهای بی سیم در مقایسه با سیستمهای سیم دار، دارای پیچیدگیهای ذاتی است. در اترنت یک ایستگاه منتظر می ماند تا کانال آزاد شود؛ سپس ارسال خود را شروع می کند. اگر در خلال ارسال ۶۴ بایت اول فریم، هیچ نویز شدیدی برنگشت، می توان اعتقاد داشت که فریم به درستی تحویل مقصد شده است. در شبکه بی سیم چنین وضعیتی حاکم نیست.

برای شروع باز هم یادآور می شویم که مشکل ایستگاه مخفی (که قبلاً تشریح و مجدداً در شکل ۴-۲۶ به تصویر کشیده شده است) ایجاد اشکال خواهد کرد. از آنجایی که تمام ایستگاهها در برد رادیویی یکدیگر نیستند فلذا ارسال سیگنال در بخشی از یک سلول، ممکن است در ناحیه دیگری از همان سلول قابل دریافت و شنود نباشد. در این مثال ایستگاه C در حال ارسال به ایستگاه B است ولی اگر A کانال را بررسی و شنود کند هیچ چیزی نمی شنود و به غلط نتیجه می گیرد که باید ارسال برای B را آغاز کند. [شکل ۴-۲۶-الف]

همچنین عکس این مشکل نیز وجود دارد. در شکل ۴-۲۶-ب ایستگاه B می خواهد فریمی را برای C بفرستد و به همین دلیل به کانال گوش می دهد. وقتی سیگنال در حال انتقال را شنود می کند به غلط نتیجه می گیرد که نباید برای ایستگاه C چیزی بفرستد ولی علیرغم آنکه A در حال ارسال برای D است (D در شکل نشان داده نشده) ایستگاه B می تواند برای C ارسال داشته باشد. مضاف براین، ارتباط بی سیم اغلب ماهیت «دوطرفه غیرهمزمان» (Half Duplex) دارد بدین معنا که ایستگاهها نمی توانند در حین ارسال و بطور همزمان، کانال را برای آگاهی از وضعیت تصادم بر روی همان باند فرکانسی شنود کنند. [این کار در کانالهای سیمی به راحتی امکان پذیر است] در نتیجه 802.11، برخلاف اترنت از CSMA/CD استفاده نمی کند.

برای رفع این مشکلات، استاندارد 802.11 از دو روش عملکرد پشتیبانی می کند: در اولین روش که DCF نامیده می شود (Distributed Coordination Function) هیچ گونه کنترلی مرکزی وجود ندارد (و از این دیدگاه مشابه با اترنت است). در روش دیگر که PCF (Point Coordination Function) نامیده می شود، برای کنترل و نظارت بر کلیه فعالیتهای درون هر سلول، از یک ایستگاه ثابت استفاده شده است. در پیاده سازی استاندارد 802.11، باید از DCF پشتیبانی شود در حالیکه حمایت از PCF اختیاری است. به ترتیب این دو روش را تشریح خواهیم کرد.



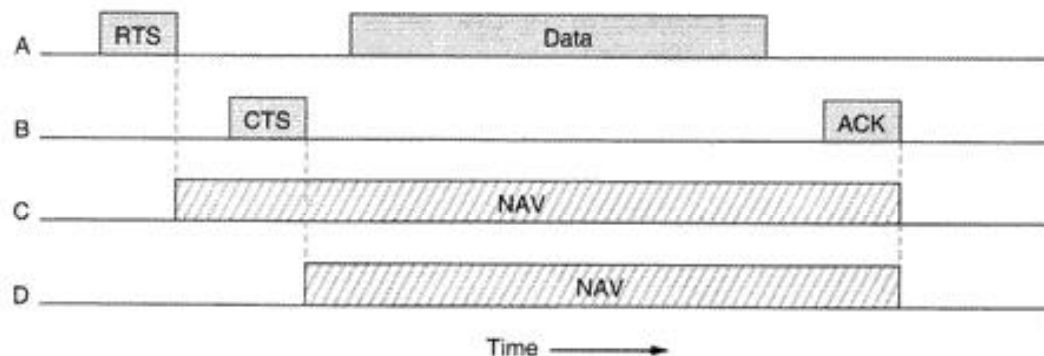
شکل ۴-۲۶. (الف) مشکل ایستگاه مخفی (ب) مشکل ایستگاه آشکار.

وقتی از حالت DCF استفاده می شود، 802.11 از پروتکلی به نام CSMA/CA^۱ بهره می گیرد. در این پروتکل هم کانال فیزیکی و هم کانال مجازی شنود می شوند. CSMA/CA از دو عملکرد متفاوت پشتیبانی می کند: در روش اول وقتی یک ایستگاه می خواهد فریمی را ارسال کند، ابتدا به شنود می پردازد و اگر کانال آزاد بود ارسال خود را آغاز می کند. در حین ارسال فریم، کانال شنود نمی شود و کل فریم منتقل می گردد، در حالیکه ممکن است این فریم، بدلیل تداخل رادیویی در گیرنده از بین برود. برعکس، اگر کانال اشغال باشد فرستنده ارسال خود را تا زمان آزاد شدن کانال به تعویق انداخته و سپس شروع می کند. در صورت بروز تصادم، ایستگاه های تصادم کننده به اندازه یک زمان تصادفی (که بر اساس الگوریتم عقب گرد نمایی تعیین می شود) منتظر مانده و از نو تلاش می کنند. [تا اینجا همه چیز شبیه به پروتکل CSMA/CD است.]

روش دیگر به کاررفته در CSMA/CA مبتنی بر MACAW است که از روش «شنود کانال مجازی» بهره می گیرد. در مثال شکل ۴-۲۷، ایستگاه A می خواهد فریمی را برای B بفرستد. C، ایستگاهی در برد ایستگاه A است (شاید C هم در برد ایستگاه B باشد ولی مهم نیست). D، ایستگاهی در برد B است ولی در برد A قرار ندارد. پروتکل زمانی آغاز می شود که A تصمیم به ارسال داده برای B می گیرد. او کارش را با ارسال یک فریم کوتاه RTS برای B آغاز و تقاضای مجوز ارسال فریم می نماید. هرگاه B این تقاضا را دریافت کند (احتمالاً) تصمیم به صدور مجوز می گیرد؛ در این حالت فریم CTS را برمی گرداند. پس از دریافت فریم CTS، ایستگاه A ارسال فریم خود را شروع کرده و یک زمان سنج خاص به نام ACK-Timer را روشن می کند. پس از دریافت فریم داده، ایستگاه B با ارسال فریم ACK به مبادله داده خاتمه می دهد. اگر زمان سنج، قبل از آنکه فریم ACK باز گردد، منقضی شود [یعنی فریم ACK در زمان معقولی برنگردد] کل این فرآیند باید از نو اجرا شود.

حال بیایید از دیدگاه ایستگاه های C و D به این فرآیند مبادله فریم، نگاه کنیم. C ایستگاهی است که در برد A است و طبقاً RTS را دریافت می کند. اگر اینگونه باشد متوجه خواهد شد که شخص دیگری بزودی ارسال خود را آغاز خواهد کرد، فلذا برای احتیاط کامل، تا تکمیل عملیات مبادله داده، از ارسال هر چیزی اجتناب می کند. C می تواند از طریق اطلاعاتی که از فریم RTS بدست می آید، کل زمان ارسال را تخمین بزند (با احتساب زمان برگشت فریم ACK)، لذا برای خودش فرض می کند که در خلال زمانی که در شکل ۴-۲۷ با عنوان NAV

۱. CSMA with Collision Avoidance



شکل ۲۷-۴. کاربرد کانال مجازی در روش CSMA/CA.

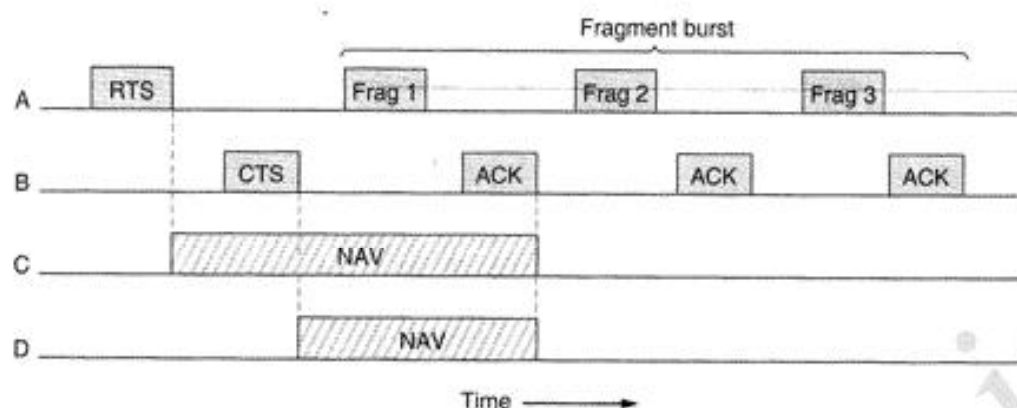
(Network Allocation Vector) مشخص شده، کانال مجازی مشغول است. ایستگاه D فریم RTS را نمی شنود ولیکن CTS را خواهد شنید لذا او هم برای خودش به اندازه زمان NAV کانال را مشغول فرض می کند. دقت کنید که سیگنال NAV به هیچ وجه ارسال نمی شود بلکه فقط یک یادداشت داخلی است که ایستگاه را در مدت زمان معینی ساکت نگه می دارد.

برخلاف شبکه های سیمی، شبکه های بی سیم غیر قابل اعتماد و نویزی هستند چراکه دستگاه های مختلف مثل اجاق های مایکروویو (که آنها نیز در باند ISM کار می کنند) نویز قوی و مخرب تولید می نمایند. در نتیجه، احتمال انتقال موفق فریم با افزایش طول فریم کاهش خواهد یافت. هرگاه احتمال خرابی یک بیت p باشد احتمال آنکه یک فریم n بیتی، کامل و سالم دریافت شود $(1-p)^n$ خواهد بود. برای مثال اگر $p=10^{-4}$ باشد احتمال سالم رسیدن یک فریم کامل اترنت (۱۵۱۴ بیتی) کمتر از ۳۰ درصد است. برای $p=10^{-5}$ ، از هر ۹ فریم یکی خراب خواهد شد. برای $p=10^{-6}$ حدود یک درصد از کل فریم ها آسیب خواهد دید که تقریباً معادل یک دوجین فریم در ثانیه خواهد بود. بطور خلاصه اگر یک فریم بزرگ باشد شانس کمتری در سالم رسیدن آن به مقصد وجود دارد و احتمالاً باید مجدداً ارسال شود.

برای کاهش مشکل کانال های نویزی، 802.11 اجازه داده که هر فریم به قطعات کوچکتری تقسیم شده و هر کدام کد کشف خطای خود را داشته باشند. قطعات بطور مجزا شماره گذاری شده و دریافت آن به روش «توقف و انتظار» (Stop & Wait) تائید می شود. (به عبارت دیگر فرستنده قطعه شماره $k+1$ را نخواهد فرستاد مگر آنکه پیام ACK قطعه k مبنی بر دریافت صحیح آن- دریافت شود.) پس از آنکه به کمک RTS و CTS، کانال در اختیار ایستگاهی قرار گرفت، طبق شکل ۲۸-۴، آن ایستگاه می تواند متوالیاً چندین قطعه مستقل، ارسال کند. دنباله قطعات متوالی، اصطلاحاً Fragment Burst نامیده می شود.

عملیات قطعه سازی فریمها، کارانی مفید شبکه را افزایش خواهد داد چرا که به جای ارسال مجدد کل فریم فقط قطعات کوچکی که در اثر خطای کانال خراب شده اند از نو ارسال خواهند شد. طول هر قطعه به صورت قطعی و ثابت در استاندارد تعیین نشده است بلکه جزو پارامترهای قابل تنظیم هر سلول محسوب و توسط ایستگاه ثابت (Base Station) تنظیم می شود. مکانیزم NAV ایستگاهها را آنقدر ساکت نگاه می دارد تا زمانی که دریافت کل فریم تایید شود، ولیکن برای آنکه ایستگاهها آنقدر تامل کنند تا دنباله کل قطعات فریم (Fragment Burst) بدون تداخل ارسال شود، مکانیزم دیگری که در زیر تشریح شده، بکار می رود.

تمام توضیحات فوق در حالت DCF از استاندارد 802.11 قابل اعمال و صائب است. یادآوری می کنیم که در حالت DCF هیچ کنترل و نظارت مرکزی وجود ندارد و تمام ایستگاهها مشابه با آنچه که در اترنت اتفاق می افتد برای بدست آوردن کانال رادیویی رقابت می کنند. در حالت دیگر یعنی PCF، یک ایستگاه ثابت یکی یکی به



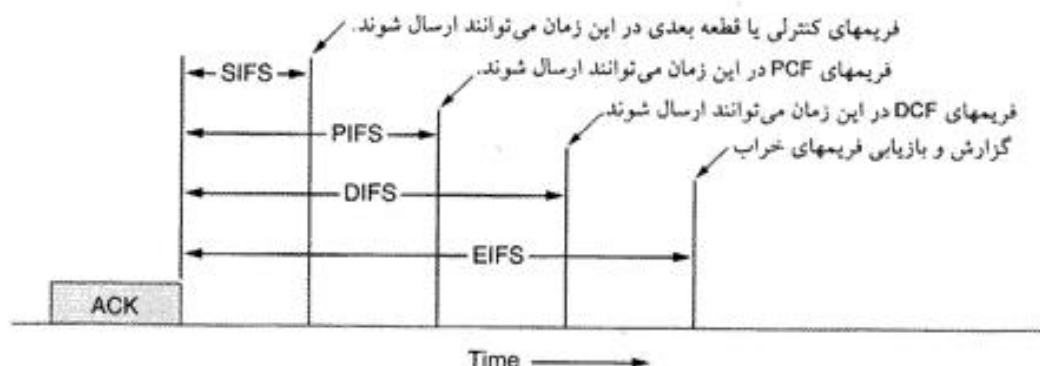
شکل ۲۸۴. ارسال انفجاری چند قطعه.

ایستگاه‌ها سرکشی کرده و از آنها سوال می‌کند که آیا فریمی جهت ارسال دارند یا خیر. از آنجایی که در حالت PCF بر تمام تقاضاهای ارسال فریم به صورت مرکزی نظارت می‌شود لذا هیچگونه تصادم اتفاق نخواهد افتاد. استاندارد 802.11، مکانیزم سرکشی به ایستگاه‌ها را تعیین کرده است ولیکن دفعات سرکشی، ترتیب سرکشی یا حتی تصمیم‌گیری در خصوص آنکه آیا همه ایستگاه‌ها باید به یک اندازه از شبکه سرویس بگیرند، در استاندارد تعریف و تعیین نشده است.

مکانیزم اصلی در سرکشی به ایستگاه بدین نحو است که یک ایستگاه فریم خاصی به نام Beacon Frame (فریم فانوس دریائی) را بطور متناوب در فضای پیرامون خود منتشر می‌کند. (ده تا صد بار در هر ثانیه) «فریم فانوس دریائی» شامل پارامترهای مختلف سیستم مثل: «ترتیب پرش فرکانسی» (Hopping Sequence) و پارامتر dwell time (برای مدولاسیون FHSS) و پارامتر سنکرون‌سازی سیگنال ساعت و نظایر آن، می‌باشد. همچنین توسط این فریم از ایستگاه‌های جدید دعوت می‌شود تا به منظور سرکشی شدن ثبت‌نام کنند. پس از آنکه ایستگاهی برای دریافت خدمات سرکشی با نرخ معین، ثبت‌نام کرد، این تضمین را دارد که بخش معینی از پهنای باند شبکه به او اختصاص داده خواهد شد و بدین ترتیب آن ایستگاه می‌تواند «کیفیت خدمات» (Quality of Service) خود را تضمین کند. [برای مطالعه در خصوص کیفیت خدمات به بخش ۱-۳-۳ و فصل ۵ مراجعه کنید.]

طول عمر باتری یکی از مسائل مهم در ابزارهای همراه و بی‌سیم بوده و هست؛ به همین دلیل در استاندارد 802.11 به مسئله مدیریت توان مصرفی، توجه ویژه‌ای شده است. خصوصاً در حالت PCF ایستگاه ثابت می‌تواند ایستگاه همراه را به «حالت استراحت» (Sleep State) ببرد؛ تا زمانی که بطور مشخص ایستگاه ثابت یا کاربری دیگر آن را از این حالت بیرون آورده و بتواند فعالیت عادی خود را از سر بگیرد. مجبور کردن یک ایستگاه به استراحت، بدین معناست که ایستگاه ثابت مسئولیت دارد تمام فریم‌هایی را که برای ایستگاه غیرفعال (در حال استراحت) ارسال می‌شود، دریافت و بافر کند. بعداً این فریم‌ها به صورت یکجا تحویل خواهد شد.

در درون یک سلول می‌توان بطور همزمان هم حالت PCF و هم حالت DCF را به کار گرفت. در نگاه اول ممکن است به نظر برسد که نظارت مرکزی و نظارت توزیع شده بطور همزمان میسر نباشد ولی در استاندارد 802.11 برای رسیدن به چنین هدفی راهکاری مناسب اندیشیده شده است. برای این کار، بازه‌های زمانی بین فریم‌ها بدقت تعریف می‌شود. پس از آنکه یک فریم ارسال شد و قبل از آنکه ایستگاهی بتواند فریم بعدی را ارسال نماید به مدتی «زمان مرده» نیاز است. در این زمان مرده، چهار بازه زمانی مجزا با اهداف خاص، تعریف



شکل ۲۹-۴. فاصله زمانی بین فریمها در 802.11.

شده است. این چهار بازه زمانی در شکل ۲۹-۴ نشان داده شده است.

کوتاه ترین بازه زمانی، بازه SIFS (Short InterFrame Spacing) است. این بازه زمانی به ایستگاهها فرصت می دهد تا به ارسال فریمهای کنترل خاص بپردازند. در این زمان ایستگاهها اجازه می یابند عملیاتی مثل ارسال CTS (در پاسخ به RTS)، ارسال فریم ACK در پاسخ به یک فریم کامل یا یک قطعه از فریم، ارسال یک قطعه از دنباله قطعات (بدون ارسال RTS مجدد) یا نظائر این را انجام بدهند.

همیشه فقط یک ایستگاه است که پس از زمان SIFS، به منظور پاسخ دهی [و ارسال فریم کنترل مناسب] حق ارسال دارد. اگر ایستگاه مربوط نتواند از این فرصت استفاده کند و زمان PIFS (PCF InterFrame Spacing) منقضی شود، ایستگاه ثابت می تواند «فریم فانوس دریایی» (Beacon) یا «فریم سرکشی» ارسال نماید. این مکانیزم اجازه می دهد که ایستگاه در حال انتقال فریم یا دنباله قطعات، بدون آنکه ایستگاه دیگری در این میان مداخله کند ارسال فریم خود را به پایان برساند در حالیکه ایستگاه ثابت نیز این فرصت را خواهد داشت که وقتی ایستگاه قبلی کارش را به اتمام رساند کانال را بدون رقابت با کاربران متعادل به ارسال تصرف نماید.

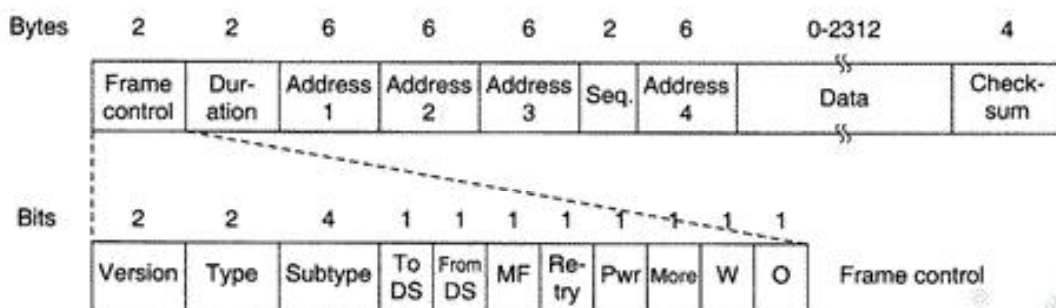
هرگاه ایستگاه ثابت چیزی برای ارسال نداشته باشد و زمان DIFS (DCF InterFrame Spacing) منقضی شود هر ایستگاه می تواند بخت خود را در تصرف کانال و ارسال فریم بیازماید. در این لحظه، برای در اختیار گرفتن کانال، روش معمولی رقابت و در صورت تصادم الگوریتم عقب گرد نمایی اعمال می شود.

آخرین بازه زمانی یعنی EIFS (Extended InterFrame Spacing) مورد استفاده ایستگاهی قرار می گیرد که یک فریم خراب یا فریمی ناشناس دریافت کند تا بتواند این مسئله را گزارش بدهد. دلیل اصلی آنکه به این بازه زمانی کمترین اولویت داده شده [آخرین بازه زمانی است] آن بوده که چون گیرنده نمی داند چه اتفاقی در جریان است لذا باید مدت زمان قابل توجهی صبر کند تا از هرگونه مداخله در گفتگوی دو ایستگاه اجتناب نماید.

۴-۴ ساختار فریم 802.11

استاندارد 802.11 سه رده مختلف فریم برای ارسال بر روی کانال تعریف کرده است: «فریم داده»، «فریم کنترل» و «فریم مدیریتی». هر یک از این فریمها سرآیند (Header) خاص خود را دارند که در هر سرآیند، فیلدهایی جهت استفاده در زیرلایه MAC تعریف شده است. مضاف بر این سرآیندهایی جهت به کارگیری در لایه فیزیکی تعریف گردیده که اغلب تکنیک های مدولاسیون [و پارامترهای مخابراتی] را مشخص می کنند، لذا در اینجا بدانها نخواهیم پرداخت.

قالب فریم داده، در شکل ۳-۴ نشان داده شده است. در ابتدا «فیلد کنترل» ظاهر شده که این فیلد خودش دارای یازده فیلد فرعی است. اولین زیرفیلد، شماره نسخه پروتکل (Protocol Version) را مشخص می کند؛



شکل ۳-۴. فریمهای داده در 802.11.

بدین ترتیب در آن واحد و در یک سلول مشابه، به کارگیری دو پروتکل متفاوت ممکن خواهد بود. در ادامه، زیرفیلد دو بیتی Type آمده که نوع فریم را [اعم از فریم داده، کنترلی و مدیریتی] مشخص می‌نماید؛ پس از آن زیرفیلد Subtype قرار گرفته که مشخصات دقیقتر فریم (مثل RTS و CTS) را تعریف می‌کند. سپس بیهیهای ToDS و FromDS آمده که مشخص می‌کنند که آیا فریم از یک «سیستم توزیع درون‌سلولی» (مثلاً شبکه اینترنت) بیرون آمده یا بدانجا رهسپار است. بیت MF نشان می‌دهد که هنوز قطعاتی از فریم در پیش رو هستند [و هنوز قطعاتی از فریم باقیست]. بیت Retry نشانگر آنست که فریم جاری قبلاً یکبار ارسال شده است. ایستگاه ثابت بکمک بیت Power Management (که در شکل با نماد Pwr نشان داده شده است)، ایستگاه متحرک را به حالت استراحت (Sleep) برده یا آنرا از حالت استراحت بیرون می‌آورد. بیت More نشان می‌دهد که فرستنده باز هم فریمهایی برای ارسال به گیرنده آماده دارد. بیت W مشخص می‌کند که بدنه فریم با استفاده از الگوریتم WEP (Wire Equivalent Privacy) (فصل نهم) رمزنگاری شده است. نهایتاً بیت O به گیرنده تفهیم می‌کند دنباله‌ای از فریمها که این بیت در آنها ۱ است باید الزاماً به ترتیب (و پشت سرهم) پردازش شوند.

فیلد دوم از فریم داده یعنی فیلد Duration، مشخص‌کننده آنست که ارسال فریم جاری و دریافت ACK آن، جمعاً چه زمانی کانال را به حالت اشغال در خواهد آورد. این فیلد که در فریمهای کنترلی نیز وجود دارد مشخص می‌نماید که ایستگاه‌های دیگر به چه نحوی باید مکانیزم NAV خود را مدیریت کنند. سرآیند فریم حاوی چهار آدرس است که همگی منطبق با قالب استاندارد IEEE 802 هستند که توسط IEEE تعریف شده است. بدیهی است که به دو فیلد آدرس مبدا و مقصد نیاز بوده ولیکن دوتای دیگر چه کاربردی دارند؟ به خاطر داشته باشید که یک فریم ممکن است از طریق ایستگاه ثابت به یک سلول وارد یا از سلول خارج شود. دو آدرس اضافی، برای تعیین ایستگاه ثابت مبدا و مقصد بکار می‌رود. (وقتی که ترافیک داده‌ها بین چند سلول در حال جریان است).

فیلد Sequence اجازه می‌دهد تا قطعات یک فریم شماره گذاری شوند. از شانزده بیت موجود در این فیلد ۱۲ بیت، هویت فریم را و ۴ بیت شماره قطعه را مشخص می‌کند. در فیلد Data داده‌های خام قرار می‌گیرد که می‌تواند حداکثر ۲۳۱۲ بایت باشد. در آخر نیز فیلد Checksum قرار گرفته که به منظور کشف خطا کاربرد دارد.

«فریم مدیریتی» قالبی مشابه با «فریم داده» دارد با این تفاوت که این فریم یکی از آدرسهای ایستگاه ثابت را ندارد، زیرا فریمهای مدیریتی فقط محدود به یک سلول خاص هستند [و بین سلولهای متفاوت مبادله نخواهند شد]. فریم کنترلی بازهم کوتاهتر هستند و فقط یک یا حداکثر دو فیلد آدرس دارند. این فریمها، فاقد فیلد داده و فیلد Sequence هستند. اطلاعات اساسی اینگونه فریمها، درون فیلد Subtype نهفته است که عموماً نوع RTS، CTS یا ACK را مشخص می‌کند.

۵-۴-۴ خدمات

استاندارد 802.11 بیان داشته که هر شبکه محلی بی سیم باید ۹ نوع خدمات عرضه نماید. این خدمات به دو رده تقسیم بندی شده اند. پنج نوع خدمات «توزیعی» و چهار نوع خدمات «ایستگاهی». خدمات توزیعی در خصوص مدیریت بر عضویت ایستگاههای درون سلول و تعامل با ایستگاههای خارج از سلول است. در مقابل، خدمات ایستگاهی صرفاً در خصوص فعالیتهای درون یک سلول واحد ارائه می شود.

پنج نوع خدمات توزیعی که توسط ایستگاههای ثابت عرضه می شوند، در خصوص قابلیت تحرک ایستگاههای همراه، ورود و خروج آنها به سلولها و اتصال یا انفصال از ایستگاه ثابت کاربرد دارند. این خدمات عبارتند از:

۱. Association (پیوستن به شبکه): ایستگاههای متحرک از این سرویس بهره می گیرند تا خود را به ایستگاه ثابت متصل نمایند. بطور معمول زمانی که یک ایستگاه متحرک وارد محدوده رادیویی یک ایستگاه ثابت می شود با استفاده از این سرویس، هویت و قابلیتهای خود را معرفی می کند. قابلیتها عبارتند از نرخ ارسال داده ها (نرخهای متعددی که از آن حمایت می شود)، نیاز به خدمات متعدد PCF (مثل سرکشی) و نیازمندیهای آن در خصوص مدیریت توان مصرفی. ایستگاه ثابت می تواند حضور ایستگاه متحرک را بپذیرد یا رد کند. اگر ایستگاه متحرک پذیرفته شد بایستی هویت خود را اثبات کند. [روشهای احراز هویت را در فصل نهم مطالعه کنید].

۲. Disassociation (ترک شبکه): ممکن است ایستگاه متحرک یا ایستگاه ثابت، در هر زمان اراده کنند از یکدیگر جدا شوند و ارتباط خود را قطع نمایند. وقتی ایستگاهی بخواهد شبکه را ترک کند یا خاموش شود، از این سرویس بهره می گیرد اما ایستگاه ثابت نیز قبل از قطع موقت ارتباط ممکن است از آن استفاده کند.

۳. Reassociation (پیوستن مجدد): یک ایستگاه متحرک می تواند با استفاده از این سرویس، ایستگاه ثابت خود را تغییر بدهد. این قابلیت زمانی کاربرد دارد که ایستگاههای متحرک بخواهند از یک سلول به سمت سلول دیگر حرکت نمایند. اگر از این امکان به موقع و صحیح استفاده شود در اثر این جابجائی، هیچ داده ای از دست نخواهد رفت.

۴. Distribution (توزیع): این سرویس مسیر ارسال فریمهای ارسالی به سوی ایستگاه ثابت را تعیین می نماید. اگر مقصد فریمها در همان محل ایستگاه ثابت واقع شده باشد فریم می تواند مستقیماً از طریق هوا ارسال شود. در غیر این صورت فریمها باید از طریق شبکه سیمی (ارتباط سیمی بین ایستگاههای ثابت) به مقصد هدایت شود.

۵. Integration (یکپارچگی): اگر نیاز باشد فریمی به شبکه ای غیر از 802.11 (با ساختار آدرس و ساختار فریم متفاوت) ارسال شود، این سرویس می تواند وظیفه ترجمه و تبدیل قالب فریم (متناسب با شبکه مقصد) را برعهده بگیرد.

چهار سرویس باقیمانده در درون یک سلول بکار می آیند. (بعبارت دیگر در خصوص عملیات درون یک سلول واحد کاربرد دارند.) این سرویسها پس از پیوستن ایستگاهها به یک ایستگاه ثابت [به عنوان عضوی از شبکه] مورد استفاده قرار گرفته و به عبارت زیر هستند:

۱. Authentication (احراز هویت): از آنجایی که در مخابرات بی سیم ایستگاههای غیرمجاز و بیگانه نیز می توانند ارسال و دریافت داشته باشند فلذا هر ایستگاه باید قبل از دریافت مجوز ارسال، هویت خود را اثبات کند. پس از آنکه یک ایستگاه متحرک به عنوان عضو، به یک ایستگاه ثابت پیوست (یا بعبارت دیگر درون

سلول پذیرفته شد)، ایستگاه ثابت یک فریم خاص به نام «فریم چالش» (Challenge) برای او می‌فرستد تا ببیند آیا آن ایستگاه کلید سری (کلمه عبور) خود را می‌داند یا نه؟ ایستگاه، آگاهی خود از کلید سری را با رمز کردن فریم و بازگرداندن آن، اثبات می‌نماید. اگر نتیجه درست باشد، ایستگاه متحرک در درون سلول ثبت‌نام و عضو می‌شود. در استاندارد اولیه لازم نبود که ایستگاه ثابت نیز عضویت خود را احراز کند در حالیکه اصلاح این اشکال بزرگ در دست اجرا است.

۲. Deauthentication (لغو حضور در شبکه): وقتی یک ایستگاه که قبلاً احراز هویت (و عضو) شده بخوهد شبکه را ترک کند باید لغو حضور خود در شبکه را به اطلاع ایستگاه ثابت برساند. پس از لغو حضور و ترک شبکه، ایستگاه دیگر نمی‌تواند از شبکه بهره بگیرد.

۳. Privacy (محرمانه نگاه داشتن اطلاعات): برای محرمانه ماندن اطلاعاتی که از طریق شبکه بی‌سیم مبادله می‌شود، بایستی رمز شوند. این کار از طریق رمزنگاری و رمزگشایی میسر است. الگوریتم رمزنگاری مورد استفاده روش RC4 می‌باشد که توسط رونالد ری‌وست در دانشگاه MIT ابداع شده است.

۴. Delivery (تحویل): انتقال داده‌ها کل آن هدفی است که در پی آن هستیم لذا در 802.11 روشی برای ارسال و دریافت داده‌ها ارائه شده است. از آنجایی که 802.11 مبتنی بر مدل شبکه اترنت است و مبادله اطلاعات در اترنت تضمین صددرد ندارد، 802.11 نیز مبادله مطمئن داده‌ها را تضمین نمی‌کند. لایه‌های بالاتر باید در خصوص کشف و تصحیح خطا کاری انجام بدهند.

در استاندارد 802.11، هر سلول پارامترهایی دارد که می‌توان آنها را بررسی و در صورت نیاز تنظیم کرد. این پارامترها در خصوص عملیات رمزنگاری، نرخ ارسال، بازه‌های زمانی [زمان‌های انقضای مهلت یا Timeouts]، تناوب ارسال فریمهای Beacon و نظایر اینها، تعریف شده‌اند.

شبکه‌های محلی بی‌سیم 802.11، در حال ورود به عرصه ادارات، فرودگاه‌ها، هتلها، رستورانها و محوطه‌های دانشگاهی هستند و انتظار می‌رود رشد سریع و چشمگیری داشته باشند. برای آنکه در خصوص استفاده گسترده از 802.11، اطلاعات و تجاربی بدست بیاورید به مرجع (Hills, 2001) مراجعه کنید.

۵-۴ بی‌سیم با باند گسترده

بیش از اندازه به درون (شبکه‌های LAN) پرداخته‌ایم. حال اجازه بدهید پا را فراتر گذاشته و ببینیم آیا در خارج از دنیای LAN هم شبکه‌های جالبی وجود دارند. با رفع موانع قانونی از مقررات حاکم بر عرضه خدمات تلفن، در بسیاری از کشورها شرکتهای مخابراتی رقیب اجازه یافتند تا به ارائه خدمات صوت و خدمات اینترنت پرسرعت بپردازند. امروزه نیاز عمومی در این خصوص بسیار بالا و پررونق است. برای چنین شرکتهایی مسئله اساسی آنست که کشیدن کابلهای فیبرنوری، کابل کوآکسیال یا سیمهای زوجی (از نوع CAT 5) تا درب در میلیونها خانه و محل کار، بسیار گران و نامعقول به نظر می‌رسد. پس این رقباى تجارى برای عرضه خدمات ارزان، چه باید می‌کردند؟

پاسخ به این نیاز «شبکه بی‌سیم باند گسترده» است. برافراشتن یک آنتن بزرگ بر روی قله یا یک تپه و نصب آنتنهای بر روی پشت بام مشتریان که به سمت آنتن اصلی جهت‌گیری شده، بسیار ارزان‌تر از حفر زمین و کابل‌کشی است. بهمین دلیل شرکتهای مخابرات راه دور بیشتر گرایش دارند که برای عرضه خدمات اینترنت، ارسال صدا و ارائه نمایشهای ویدیویی راه دور، از سیستمهای چندگیکایی بی‌سیم بهره بگیرند. همانگونه که در شکل ۳-۲ ملاحظه نمودید، LMDS به همین منظور ابداع شد ولیکن تاکنون هر یک از شرکت‌های عرضه‌کننده

LMDS، به سلیقه خود سیستمی را طراحی و نصب کرده است. فقدان یک استاندارد واحد، بدین معناست که سخت‌افزار و نرم‌افزار آنها نمی‌تواند بطور انبوه تولید شود و طبعاً قیمت‌ها بالا و استقبال عمومی از آنها کم خواهد بود.

نهایتاً بسیاری از دست‌اندرکاران صنعت بدین نتیجه رسیدند که داشتن یک استاندارد برای بی‌سیم باند گسترده [با نرخ ارسال بسیار بالا] یکی از مولفه‌های اصلی و حلقه مفقوده در کار آنهاست، لذا از IEEE خواستند تا برای تدوین چنین استانداردی، کمیته‌ای با حضور دست‌اندرکاران شرکت‌های مهم و مراکز دانشگاهی، تشکیل بدهد. اولین عدد اختصاصی داده نشده در ردیف شماره‌های 802.x، ۱۶ بود و به همین دلیل استاندارد، با عنوان IEEE 802.16 معرفی شد. کار کمیته در ژولای ۱۹۹۹ شروع و استاندارد نهانی در آوریل ۲۰۰۲ به تائید رسید. نام رسمی استاندارد «واسط هوایی برای سیستم‌های بی‌سیم غیرمتحرک با پهنای باند وسیع»^۱ انتخاب شده ولیکن برخی از افراد ترجیح می‌دهند آنرا «شبکه بین شهری بی‌سیم» یا «حلقه بی‌سیم» بنامند. ما تمام این واژه‌ها را معادل یکدیگر فرض خواهیم کرد.

همانند بسیاری از استانداردهای دیگر سری ۸۰۲، استاندارد 802.16 نیز شدیداً تحت تأثیر مدل OSI بوده و این تأثیر در (زیر) لایه‌ها، اصطلاحات، سرویس‌های پایه و موارد دیگر بخوبی محسوس است. متأسفانه این استاندارد نیز شبیه به OSI پیچیده شده است. در بخش بعدی بطور مختصر نکات و ویژگیهای برجسته 802.16 را برخواهیم شمرد ولی این توضیحات به هیچ وجه کامل نبوده و جزئیات آن ناگفته خواهد ماند.

۱.۵.۴ مقایسه 802.11 با 802.16

در همان ابتدا ممکن است بدین نکته بیندیشید که چرا استاندارد جدیدی ابداع شد؟ چرا از 802.11 استفاده نشد؟ دلایل متعدد و محکمی برای عدم استفاده از 802.11 وجود دارد. در اصل 802.11 و 802.16 نیازهای متفاوتی را برآورده می‌کنند. قبل از پرداختن به فناوری 802.16 شاید چند کلمه‌ای بحث در خصوص دلایل نیاز به استاندارد جدید، خالی از لطف نباشد.

محیطی که 802.11 و 802.16 در آن، عمل می‌کنند از چند منظر شبیه به هم هستند: در اصل هر دوی این استانداردها بدن جهت طراحی شده‌اند که ارتباط بی‌سیم با پهنای باند بسیار بالا را میسر نمایند. ولیکن این دو شبکه از جهات بسیار مهمی با هم متفاوتند. اصلی‌ترین تفاوت آنست که 802.16 برای ارائه خدمات به ساختمانها طراحی شده است و طبعاً ساختمانها حرکت نمی‌کنند!!! ساختمانها تغییر سلول نمی‌دهند! بخش اعظم عملیات 802.11 با مسائل ناشی از متحرک بودن ایستگاه‌ها سروکار دارد و چنین مسائلی در 802.16، محلی از اعراب ندارد. گذشته از آن، در ساختمانها ممکن است بیش از یک کامپیوتر وجود داشته باشد و پیچیدگیهای چنین محیطی در جانی که ایستگاه نهانی یک کامپیوتر کیفی واحد است، بروز نخواهد کرد. از آنجایی که مالکین ساختمان معمولاً آمادگی پرداخت پول بیشتری در مقایسه با صاحب یک کامپیوتر کیفی دارند لذا می‌توان خدمات ارتباط رادیویی بهتری در اختیارشان گذاشت. این تفاوت بدین معناست که در 802.16 می‌توان ارتباط دوطرفه همزمان (Full Duplex) داشت در حالیکه برای پائین نگاه داشتن هزینه ارتباط رادیویی در 802.11 از آن اجتناب شده است. [ارتباط 802.11 دو طرفه غیر همزمان - Half Duplex - و با برد بسیار کم می‌باشد.]

از آنجایی که 802.16 در محدوده بخشی از یک شهر به اجرا در می‌آید، فواصل [بین نقاط] می‌تواند تا چندین کیلومتر باشد و این موضوع بدن معناست که توان مورد نیاز ایستگاه ثابت می‌تواند بسته به موقعیت و فاصله‌ها متغیر باشد. این تفاوتها و تغییرها بر روی نسبت سیگنال به نویز (SNR) تأثیر گذاشته و در نتیجه، باید به اجبار از

چندین روش مدولاسیون استفاده شود. همچنین مخابره آزادانه در سطح یک شهر مبین آنست که تمهیدات امنیتی و حفظ حریم افراد، نیازی بنیادی و اجتناب ناپذیر است.

به علاوه، در مقایسه با سلول های معمول 802.11، در اینجا سلولها می توانند کاربران بسیار زیادتری را در بر بگیرند و این کاربران نیز توقع پهنای باند بیشتری نسبت به کاربران 802.11 دارند. گذشته از آن به ندرت اتفاق می افتد که شرکتی از پنجاه کارمند خود دعوت کند تا برای مشاهده اشباع شدن [و از کار افتادن 802.11] دور هم جمع شده و پنجاه فیلم مجزا تماشا کنند!! [در حالیکه در یک ساختمان مسکونی احتمال دارد پنجاه نفر بدیدن فیلم از طریق شبکه مشغول باشند]. به همین دلیل به پهنای وسیعتری از طیف فرکانسی موجود در باند ISM نیاز است و طبعاً 802.11 مجبور است در محدوده فرکانسی ۱۰ تا ۶۶ گیگاهرتزی عمل کند؛ یعنی تنها محدوده استفاده نشده از طیف فرکانس که هنوز موجود است.

ولیکن این امواج با طول موج میلی متری نسبت به امواج با طول موج بیشتر در باند ISM، ویژگی های فیزیکی کاملاً متفاوتی دارند و در نتیجه به لایه فیزیکی کاملاً متفاوتی نیاز است. یکی از ویژگی های امواج میلی متری آنست که توسط آب شدیداً جذب می شوند. (بالاخص باران، در برخی از مواقع برف، تگرگ و متاسفانه مه غلیظ) در نتیجه مدیریت خطا بسیار با اهمیت تر از محیط های داخلی [مثل محیط LAN] است. امواج میلی متری می توانند به خط مستقیم متمرکز و منتشر شوند (در حالیکه در 802.11 انتشار امواج در همه جهات است) فلذا گزینه ها و تمهیدات پیش بینی شده در ارتباط با انتشار چندمسیره (Multipath) در 802.16 محلی از اعراب ندارد.

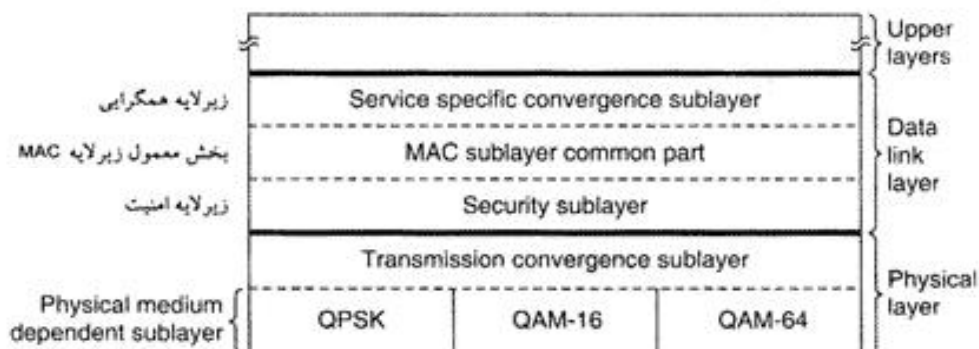
مورد دیگر مربوط به «کیفیت خدمات» (QoS) است. اگر چه 802.11 برای ترافیک بی درنگ پشتیبانی به عمل آورده (در حالت PCF) ولیکن حقیقتاً برای کاربردهای تلفنی و عملیات چندرسانه ای سنگین و دائم طراحی نشده است. در مقابل، از 802.16 انتظار می رود که کاملاً از چنین کاربردهائی پشتیبانی نماید زیرا برای استفاده در محیط های مسکونی و اداری مد نظر بوده است.

کوتاه سخن آنکه 802.11 طراحی شده تا یک اترنت متحرک باشد در حالیکه 802.16 طراحی شده تا شبیه به تلویزیون کابلی، ثابت ولی بی سیم عمل نماید. این تفاوتها آنقدر بنیادینست که استانداردهای حاصل اختلاف فراوانی دارند و در هر یک سعی در بهینه سازی و حل و فصل نیازهای متفاوتی شده است.

مقایسه ای بسیار کوتاه با سیستم تلفن سلولی [تلفن همراه] نیز خالی از فایده نیست. وقتی در مورد تلفن های همراه صحبت می کنیم روی سخن با ایستگاه هائی همراه با محوریت ارسال صوت، توان مصرفی پائین و پهنای باند باریک است که از امواج مایکروویو با طول موج متوسط بهره گرفته اند. هیچ کس یک فیلم دو ساعته و با کیفیت بالا را بر روی گوشی موبایل GSM خود تماشا نمی کند (البته فعلاً!!) حتی UMTS نیز آمیدی به تغییر چنین وضعیتی ندارد. [امید به افزایش چشمگیر پهنای باند در GSM] در عبارتی کوتاه در دنیای شبکه های بین شهری بی سیم (Wireless MAN) تقاضا برای پهنای باند بسیار بیشتر از دنیای تلفن همراه است و طبعاً به سیستمهائی کاملاً متفاوت نیاز می باشد. اینکه آیا 802.16 می تواند در آینده برای ابزارهای متحرک و همراه نیز به کار گرفته شود سوال جالبی است: این شبکه برای چنین محیطی بهینه نشده ولیکن شاید بشود! فعلاً تمرکز آن بر روی شبکه داده بی سیم و غیر متحرک است.

۲-۵-۴ پشته پروتکلی 802.16

شکل ۴-۳۱ پشته پروتکلی 802.16 را نشان می دهد. ساختار کلی این پشته، شبیه به شبکه های دیگر 802 است ولیکن تعداد بیشتری زیرلایه دارد. پائینترین زیرلایه با انتقال فیزیکی بیتها بر روی کانال سروکار دارد و در آن از رادیوی باند باریک و روش های معمول مدولاسیون، بهره گرفته شده است. بر روی «لایه فیزیکی انتقال»، زیرلایه «همگرانی» (Convergence Sublayer) قرار گرفته تا تکنولوژیهای متفاوت به کار رفته در زیر را از دید لایه پیوند



شکل ۳-۴. پشته پروتکلی 802.16.

داده ها مخفی نگاه دارد. در حقیقت، 802.11 نیز چیزی شبیه به همین زیرلایه را دارد ولی کمیته مربوطه، آنرا با استفاده از اسامی رایج در مدل OSI تعریف نکرده است.

اگرچه در شکل ۳-۴ نشان نداده ایم ولی در آینده دو پروتکل جدید به لایه فیزیکی اضافه می شود: (۱) استاندارد 802.16a که از روش مدولاسیون OFDM در باند ۲ تا ۱۱ گیگاهرتز پشتیبانی می نماید. (۲) استاندارد 802.16b که در باند ۵ گیگاهرتز ISM عمل می کند. در هر دوی اینها تلاش شده تا 802.16 به 802.11 نزدیکتر شود.

لایه پیوند داده متشکل از سه زیرلایه است: زیرلایه پائینی با امنیت و محرمانه نگاه داشتن اطلاعات سروکار دارد که برای شبکه های عمومی در محیط های باز بسیار حیاتی تر از شبکه های خصوصی بسته مثل اترنت است. این زیرلایه عملیات رمزنگاری، رمزگشایی و مدیریت کلیدها را برعهده دارد.

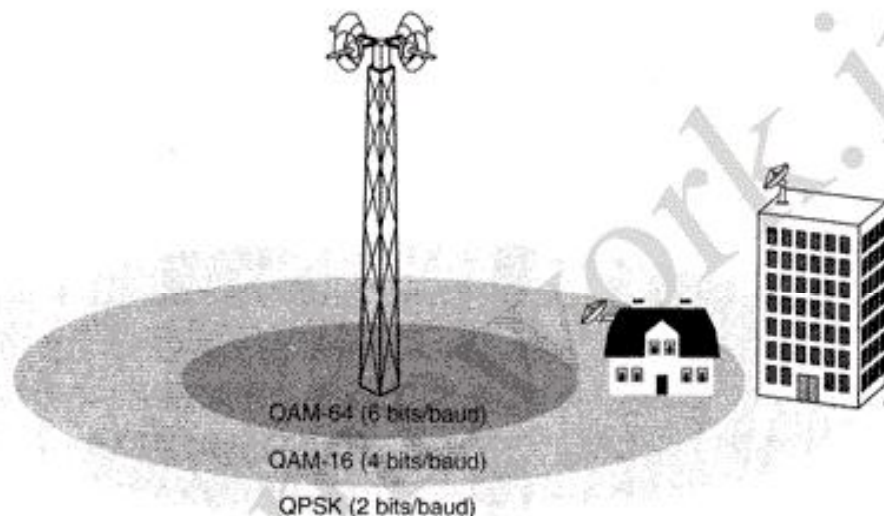
سپس بخش مشترک از زیرلایه MAC قرار می گیرد. این لایه همان نقطه ای است که پروتکل های اساسی مثل پروتکل های مدیریت کانال در بر می گیرد. در این مدل مبنا آنست که ایستگاه ثابت، سیستم را کنترل می کند. این ایستگاه می تواند «جریان اطلاعات از ایستگاه ثابت به مشترکین» (که اصطلاحاً downstream نام دارد) را به صورت کاملاً متفاوت و نمایز زمان بندی نماید و نیز نقش بسیار مهمی در مدیریت «جریان اطلاعات از مشترکین به ایستگاه ثابت» (Upstream) ایفا می کند. یکی از ویژگی های نامتعارف در زیرلایه MAC آنست که برخلاف شبکه های دیگر 802، این استاندارد کلاً اتصال گرا (Connection Oriented) است، تا بتواند «کیفیت خدمات» (QoS) را برای ارتباطات تلفنی و سایر کاربردهای چندرسانه ای تضمین نماید.

زیرلایه «همگرایی خاص سرویس دهی» (Service Specific Convergence Sublayer) به جای «زیرلایه لینک منطقی» (یعنی زیرلایه معمول LLC) قرار گرفته است. عملکرد این زیرلایه ایجاد واسطی متناسب با لایه شبکه است. پیچیدگی این زیرلایه از آنجا نشأت می گیرد که 802.16 به نحوی طراحی شده تا بتواند با پروتکل های دیتاگرام (مثل PPP، IP و اترنت) و همچنین شبکه اتصال گرای ATM قابل جمع باشد و با آنها کار کند. مشکل آنجاست که پروتکل های «مبتنی بر بسته» بدون اتصال (Connectionless) هستند در حالیکه ATM اتصال گراست. این بدین معناست که هر اتصال ATM بایستی به یک اتصال در 802.16 نگاشته شود که اصولاً کار ساده و سراسری است. ولی سوال این است که یک بسته IP ورودی باید بر روی کدامین اتصال 802.16 نگاشته شود؟ این مشکل در همین زیرلایه حل خواهد شد.

۳-۵-۴ لایه فیزیکی در 802.16

بالا اشاره شد که در بی سیم با بهنای باند وسیع، به حس گسترده تری از طیف فرکانسی نیاز است و تنها محل

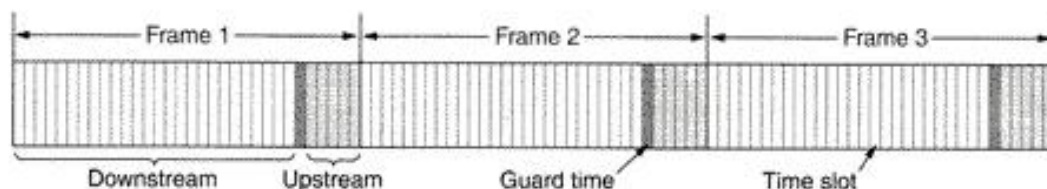
دسترسی به چنین وسعتی محدوده ۱۰ تا ۶۶ گیگاهرتزی است. این امواج با طول موج میلی متری ویژگیهای جالبی دارند که امواج مایکروویو یا طول موج بلندتر ندارند: آنها برخلاف صوت و مشابه با نور به خط مستقیم سیر می کنند. در نتیجه، ایستگاه ثابت بایستی چندین آنتن داشته باشد و هر یک از آنها بسوی قطاع خاصی از مناطق پیرامون خود نشانه رفته باشد. (شکل ۴-۳۲) هر قطاع، کاربران خاص خود را دارد و مستقل از قطاعهای همجوار خود است؛ چنین ساختاری برای رادیوی سلولی صادق نیست چرا که آنتنهای آنها «همه جهته» (Omnidirectional) هستند.



شکل ۴-۳۲. محیط انتقال در 802.16

از آنجایی که در باند امواج میلی متری توان سیگنال براساس فاصله از ایستگاه ثابت، شدیداً کاهش می یابد، نسبت سیگنال به نویز (SNR) نیز برحسب فاصله افت خواهد داشت. به همین دلیل 802.16 بسته به فاصله یک مشترک از ایستگاه ثابت، سه روش مدولاسیون متفاوت را به کار گرفته است. برای مشترکین نزدیک، از روش QAM-64 با مشخصه 6 bits/baud استفاده شده است. برای مشترکین با فاصله متوسط از QPSK با مشخصه 2 bits/baud استفاده می شود. برای مشترکین دور دست، روش QPSK با مشخصه 2 bits/baud به کار می رود. برای مثال به ازای پهنای معمول ۲۵ مگاهرتزی از طیف فرکانس، پهنای باند روش QAM-64 نرخ ۱۵۰ مگابیت بر ثانیه، روش QAM-16 نرخ ۱۰۰ مگابیت بر ثانیه و روش QPSK نرخ ۵۰ مگابیت بر ثانیه را در اختیار قرار می دهد. به عبارت دیگر هر چه مشترکین از ایستگاه ثابت دورتر باشند، نرخ ارسال پائینتر خواهد بود (شبیه به آنچه که در خصوص ADSL در شکل ۲-۲۷ دیدید) در شکل ۲-۲۵ نمودار فضایی این سه مدولاسیون، نمایش داده شده است.

با توجه به آنکه هدف اصلی ایجاد سیستمی با باند وسیع بوده و با در نظر داشتن محدودیتهای فیزیکی، طراحان 802.16 کوشیدند تا از طیف موجود به نحو بهینه استفاده کنند. آنها به روش به کاررفته در GSM و DAMPS علاقه ای نداشتند. هر دوی این سیستمها از فرکانسهای متفاوت ولی در باند مشابهی برای ارسال جریان ترافیک رو به بالا و رو به پائین (Upstream/Downstream) بهره می گیرند. برای صوت شاید ترافیک در اکثر بخشها متقارن باشد [عبارت دیگر برای صوت میزان ارسال و دریافت حدوداً به یک اندازه است] ولیکن برای دسترسی به اینترنت غالباً حجم ترافیک دریافتی بیشتر از ترافیک ارسالی است. در نتیجه 802.16 روشی منعطفتر برای تخصیص پهنای باند ارائه کرده و از دو روش FDD (Frequency Division Duplexing) و TDD (Time Division Duplexing) بهره گرفته است. روش دوم یعنی TDD، در شکل ۴-۳۳ نشان داده شده.



شکل ۳۳-۴. فریمها و برشهای زمانی در روش TDD (Time Division Duplexing).

است. در اینجا ایستگاه ثابت بطور متناوب فریمهایی را منتشر می کند. هر فریم شامل تعدادی برش زمانی مستقل (Time Slot) است. برشهای ابتدایی هر فریم، برای ارسال ترافیک روبه پائین [یعنی از ایستگاه ثابت به کاربر] در نظر گرفته شده است. پس از آن «زمان مراقبت» (Guard Time) فرا می رسد که به ایستگاه ها مهلت می دهد تا جهت ارسال و دریافت خود را تغییر بدهند. در آخر برشهایی را برای ارسال ترافیک رو به بالا [از کاربر به ایستگاه ثابت] خواهیم داشت. تعداد برشهای زمانی را که به ارسال در هر یک از جهات، اختصاص می یابد، می توان به صورت پویا تغییر داد تا پهنای باند در هر جهت، با حجم ترافیک تطبیق داشته باشد.

ترافیک روبه پائین توسط ایستگاه ثابت در درون برشهای زمانی نگاشته می شود. ایستگاه ثابت بطور کامل بر این جهت از جریان، کنترل دارد. ترافیک روبه بالا [که توسط کاربران تولید می شود] با پیچیدگی بیشتری مواجه بوده و میزان آن به کیفیت خدمات (QoS) مورد نیاز بستگی دارد. در ادامه وقتی به تشریح زیرلایه MAC پرداختیم به روش تخصیص برشهای زمانی هم خواهیم رسید.

ویژگی جالب دیگر در لایه فیزیکی، توانایی آن در ارسال متوالی و پشت سرهم فریمهای MAC، در قالب یک انتقال فیزیکی واحد است. این ویژگی سریار ناشی از پشتهای آغازین (Preamble) و سرآیند لازم در لایه فیزیکی را کاهش داده و در نتیجه بازده مفید طیف را افزایش خواهد داد.

نکته قابل توجه دیگر، استفاده از کدهای همینگ (Hamming) به منظور تصحیح مستقیم خطا در لایه فیزیکی است.^۱ تقریباً در تمام شبکه ها، به کدهای کشف خطا بسنده می شود و هرگاه فریم دریافتی دارای خطا باشد ارسال مجدد صورت می گیرد، ولی از آنجاییکه در محیط های باز و در نرخ ارسال بالا، احتمال بروز خطا در حین انتقال، خیلی بیشتر است لذا گذشته از عملیات کشف خطا (که در لایه های بالاتر انجام می شود) در لایه فیزیکی نیز عملیات تصحیح خطا صورت می گیرد. تاثیر نهانی تصحیح خطا آنست که کانال بهتر از آنی که هست به نظر می رسد. (بدلیل مشابه اگر چه CD-ROM ها قابل اعتماد به نظر می رسند ولیکن این اعتماد، حاصل از تخصیص بیش از نیمی از بیتها به تصحیح خطا در لایه فیزیکی است.) [به عبارت دیگر در هر CD، بیش از نیمی از بیتهای سطح دیسک فقط به منظور عملیات کشف و تصحیح خطا ذخیره شده اند!]

۴.۵.۴ پروتکل زیرلایه MAC در 802.16

همانگونه که در شکل ۴-۳۱ دیدیم، در 802.16، لایه پیوند داده ها به سه زیرلایه تقسیم می شود. از آنجایی که ما تا فصل هشتم به مطالعه رمزنگاری نخواهیم پرداخت لذا تشریح عملکرد «زیرلایه امنیت» در اینجا سخت است. به همین مقدار بسنده می کنیم که برای سری نگاه داشتن داده های ارسالی از رمزنگاری در سطح لایه فیزیکی و به صورت بی درنگ، بهره گرفته شده است. صرفاً بخش داده هر فریم رمزنگاری می شود و سرآیند آن (Header) رمز نخواهد شد. این خصوصیت بدین معناست که یک جاسوس قادر است بفهمد چه کسی با چه کسی محاوره می کند ولی قادر به فهم آنچه با یکدیگر می گویند نیست.

۱. Forward Error Correction

اگر با رمزنگاری آشنائی قبلی داشته باشید در حدّ یک پارگراف، زیرلایه امنیت را بررسی می‌کنیم ولیکن در صورت عدم آشنائی با رمزنگاری، پارگراف بعدی چیزی به دانش شما نخواهد افزود. (می‌توانید این پارگراف را پس از اتمام فصل ۸ مجدداً مطالعه نمایید.)

زمانی که یک مشترک به ایستگاه ثابت متصل می‌شود، دویه‌دو یکدیگر را با استفاده از روش «رمزنگاری RSA» و مبتنی بر «گرویتامه‌های X.509» احراز هویت می‌کنند. بخش داده فریمهای آنها با استفاده از یک سیستم رمزنگاری متقارن رمز می‌شود که این سیستم یا مبتنی بر «DES» یا «نجمیه‌سازی بلوکها» است و یا از روش «DES سه گانه با دو کلید» استفاده می‌شود. احتمالاً به زودی روش AES (Rijndael) نیز به آن اضافه می‌شود. عملیات بررسی صحت داده‌ها نیز با استفاده از SHA-1 انجام می‌گیرد. تا اینجا چندان بد نبود، نظر شما چیست؟!؟

حال اجازه بدهید نگاهی به بخش مشترک از زیرلایه MAC بیندازیم. فریمهای MAC، تعداد مشخصی از برشهای زمانی لایه فیزیکی را اشغال می‌کنند. هر فریم از چندین فریم کوچکتر (Subframe) تشکیل شده که دو تایی ابتدائی آن به ترتیب برای نگاشت و حمل ترافیک روبه بالا (Upstream) و روبه پائین (Downstream) در نظر گرفته شده است. ساختار این نگاشت (یعنی درج داده‌ها درون برشهای زمانی) بگونه‌ای است که مشخص می‌کند چه چیزی در یک برش زمانی درج شده و کدامیک از برشهای زمانی آزاد هستند. نگاشت ترافیک روبه پائین نیز شامل پارامترهای مختلف سیستمی است تا شروع فعالیت یک ایستگاه جدید را به اطلاع ایستگاه ثابت برساند. عملکرد کانال روبه پائین نسبتاً ساده و سر راست است. ایستگاه ثابت بسادگی تصمیم می‌گیرد که چه چیزی را در کدام فریم کوچک (Subframe) قرار بدهد. عملکرد کانال رو به بالا پیچیده‌تر است زیرا در اینجا مشترکین رقیب و ناهماهنگ، برای در اختیار گرفتن آن با یکدیگر رقابت می‌کنند. روش تخصیص کانال به مواردی در خصوص «کیفیت خدمات» (QoS) بستگی دارد. در 802.16 چهار رده از خدمات، به ترتیب ذیل تعریف شده‌اند:

۱. خدمات با نرخ ارسال ثابت (Constant Bit Rate Service)
۲. خدمات بی‌درنگ با نرخ ارسال متغیر (Real-time Variable Bit Rate)
۳. خدمات غیر بی‌درنگ با نرخ ارسال متغیر (NonReal-time Variable bit Rate)
۴. خدمات مبتنی بر «حداکثر تلاش» (Best Effort Service)

تمام خدمات عرضه شده در 802.16 اتصال‌گرا هستند و به هر اتصال^۱ فقط یکی از رده‌های خدمات فوق تعلق می‌گیرد و نوع خدمات نیز در زمان برقراری اتصال تعیین می‌شود. این طراحی بسیار متفاوت از 802.11 یا اترنت است که در آنها هیچ اتصالی در زیرلایه MAC ایجاد نمی‌شود.

«خدمات با نرخ ارسال ثابت» برای انتقال سیگنال صوتی غیرفشرده همانند یک کانال TI مدّ نظر بوده است. به این خدمات از آن جهت نیاز است که بتوان حجم معینی از داده‌ها را در زمان مشخصی ارسال کرد. برای رسیدن به این هدف، به هر اتصال از این نوع، تعداد ثابت و معینی برش زمانی (در هر فریم^۲) اختصاص داده می‌شود. هرگاه پهنای باند لازم اختصاص داده شود این برشهای زمانی بطور خودکار و بدون نیاز به درخواست بعدی در اختیار خواهد بود.

«خدمات بی‌درنگ با نرخ ارسال متغیر» برای کاربردهای چندرسانه‌ای فشرده‌شده و هرگونه عملیات بی‌درنگ که در آنها مقدار پهنای باند مورد نیاز، متغیر است سودمند خواهد بود. این کار بدین نحو انجام می‌شود که ایستگاه ثابت در فواصل زمانی مشخص به مشترک خود سرکشی کرده و از او در خصوص میزان پهنای باند مورد نیازش

۱. اتصال را یک ارتباط منطقی و هماهنگ شده بین دو نقطه در شبکه در نظر بگیرید. -م

۲. معنای واژه فریم در اینجا با معنای متعارف آن تفاوت دارد. معنای فریم 802.16 در شکل ۴-۳۳ مشخص است. -م

سوال می کند.

«خدمات غیربی درنگ با نرخ ارسال متغیر» برای حجم ترافیک سنگینی که بی درنگ نیست (همانند انتقال فایل های طولانی) در نظر گرفته شده است. برای ارائه چنین خدماتی، اغلب ایستگاه ثابت به مشترک خود سرکشی می کند ولیکن فواصل زمانی سرکشی به مشترک، قطعی و منظم نیست. یک مشترک که از نرخ ثابت بهره می گیرد می تواند یک بیت خاص را در یکی از فریمهای خود تنظیم کرده و تقاضای سرکشی بدهد تا بتواند ترافیک اضافی (با نرخ متغیر) ارسال نماید.

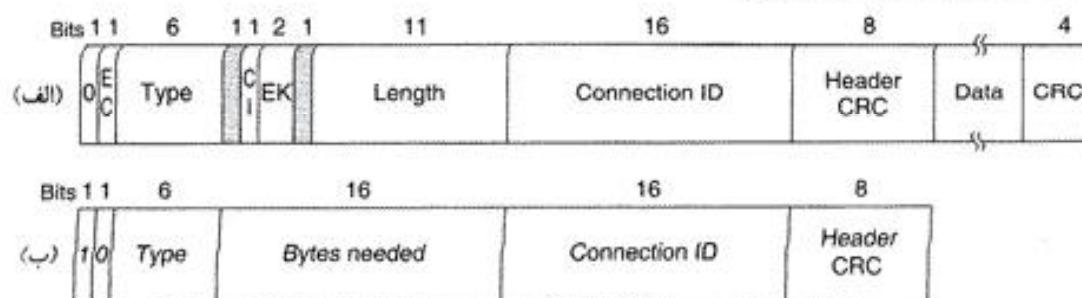
اگر ایستگاهی k بار متوالی سرکشی شود و پاسخی ندهد، ایستگاه ثابت او را در یک «گروه چندپخشی» (Multicast Group) قرار داده و از آن به بعد بصورت اختصاصی سرکشی نخواهد شد. در عوض وقتی به یک «گروه چندپخشی» سرکشی می شود هر کدام از ایستگاه های گروه می توانند پاسخ بدهند و برای دریافت خدمات رقابت کنند. بدین ترتیب ایستگاه هایی که ترافیک ناچیزی دارند زمان باارزش سرکشی را هدر نخواهند داد.

خدمات مبتنی بر «بهترین تلاش» در موارد متفاوتی کاربرد دارد. در اینجا هیچ سرکشی انجام نمی گیرد و هر مشترک برای دریافت این خدمات باید با مشترکین دیگر (که آنها نیز در پی خدمات مبتنی بر بهترین تلاش هستند) رقابت کند. تقاضای پهنای باند با علامتگذاری در یکی از برشهای زمانی ترافیک روبه بالا که برای رقابت تدارک دیده شده انجام می گیرد. اگر تقاضا به صورت موفقیت آمیز اعلان شود، این موفقیت در نگاشت ترافیک روبه پائین [فریم مربوطه به Downstream] مشخص خواهد شد. اگر تقاضا موفق نبود، مشترک بایستی مجدداً تلاش کند. برای آنکه تصادمها حداقل شود از الگوریتم عقب گرد نمایی در اترنت بهره گرفته شده است.

استاندارد 802.16، دو نوع تخصیص پهنای باند تدارک دیده است: تخصیص پهنای باند به ازای هر ایستگاه و تخصیص پهنای باند به ازای هر اتصال (per-connection). در روش اول، ایستگاه نصب شده در یک ساختمان، کلیه تقاضاهای کاربران را به صورت یکجا جمع کرده و به نیابت از همه آنها تقاضای پهنای باند می کند و اگر توانست پهنای باند درخواستی را بدست بیاورد، این پهنای باند را به تناسب بین کاربران تقسیم می کند. در روش دوم، ایستگاه ثابت هر اتصال را مستقلاً مدیریت می کند.

۵-۵-۴ ساختار فریم در 802.16

تمام فریمهای MAC با یک سرآیند عمومی شروع می شوند. به نحوی که در شکل ۴-۳۴ می بینید پس از سرآیند، بخشی اختیاری جهت حمل داده و کد اختیاری کشف خطا (CRC) قرار گرفته است. در فریمهای کترلی به بخش داده (Payload) نیازی نیست (مثلاً در فریمهایی که تقاضای برش زمانی می دهند). کد کشف خطا (Checksum) نیز اختیاری است زیرا در لایه فیزیکی، عملیات تصحیح خطا انجام می گیرد و همچنین قرار نیست فریمهای بی درنگ (در صورت خراب شدن) از نو ارسال شوند. اگر قرار نباشد که فریمی از نو ارسال شود چرا با افزودن کد کشف خطا، خود را به زحمت بیندازیم.



شکل ۴-۳۴. (الف) قالب عمومی فریم (ب) فریم تقاضای پهنای باند.

فیلدهای سرآیند شکل ۴-۳۴ الف را به اختصار معرفی می‌کنیم: بیت EC مشخص می‌کند که آیا بخش داده، رمزنگاری شده است؟ فیلد Type، نوع فریم را تعیین می‌کند، بالاخص آنکه آیا داده‌ها به صورت مجموعه‌ای از قطعات یا آنکه به صورت یکجا ارسال می‌شود. فیلد CI مشخص می‌کند که آیا فیلد کشف خطا (Checksum) در پایان فریم وجود دارد یا خیر. فیلد EK مشخص‌کننده آنست که کدامیک از کلیدهای رمزنگاری به کارگرفته شده است. (بشرط آنکه رمزنگاری انجام و چندین کلید تعریف شده باشد). فیلد Length طول کل فریم را با احتساب سرآیند تعیین می‌کند. فیلد Connection Identifier مشخص می‌کند که فریم متعلق به کدام «اتصال» است. در آخر، فیلد Header CRC در برگزیده کد کشف خطائی است که فقط از بخش سرآیند و با استفاده از چندجمله‌ای $x^8 + x^2 + x + 1$ استخراج می‌شود.

نوع دوم سرآیند که فقط برای فریمهایی که پهنای باند تقاضا می‌دهند تعریف شده، در شکل ۴-۳۴ ب نشان داده شده است. این سرآیند به جای بیت صفر یا بیت ۱ شروع شده ولیکن ترکیب کلی آن با ترکیب فریم عمومی [شکل ۴-۳۴ الف] مشابه است، با این تفاوت که بابت دوم و سوم این فریم یک عدد ۱۶ بیتی را تشکیل داده و محتوای آن مشخص می‌کند که برای مبادله تعداد مشخص بایت، به چه اندازه پهنای باند نیاز است. فریم تقاضای پهنای باند دارای بخش حمل داده (Payload) و کد کشف خطا برای کل فریم نیست. اگر چه می‌توان مفصلاً در خصوص 802.16 شرح داد ولی در اینجا به بحث خاتمه می‌دهیم. برای آگاهی بیشتر مستقیماً به استاندارد آن مراجعه کنید.

۶-۲ بلوتوث (Bluetooth)

در سال ۱۹۹۸ شرکت ال.ام. اریکسون علاقمند شد تا گوشی تلفنهای همراه تولیدی او بتوانند به صورت بی‌سیم به ابزارهای دیگر (مثل PDA) وصل شوند. اریکسون و چهار شرکت دیگر (آی‌بی‌ام، ایتل، نوکیا و توشیبا) یک «گروه SIG»^۱ تشکیل دادند تا استاندارد بی‌سیم برای اتصال ابزارهای مخابراتی رایانه‌ای و ابزارهای جانبی آنها طراحی کنند که بردی کوتاه، مصرف توان پائین و قیمتی ارزان داشته باشد. نام این پروژه، بلوتوث انتخاب شد که برگرفته از نام «هرالد بلاتاند دوم» (مشهور به Bluetooth)، یکی از پادشاهان وایکینگ است (۹۸۱-۹۴۰) که دانمارک و نروژ را با هم متحد کرد (البته با زور و بدون کابل!!!)

اگر چه تفکر اصلی، رهایی از سُرکابلهای مابین دستگاه‌های دیجیتالی بود ولی به سرعت در حوزه‌های دیگر نیز گسترش یافت و به تدریج در حیطه شبکه‌های محلی بی‌سیم نیز وارد شد. اگرچه گسترش و رشد این استاندارد، روز بروز کاربرد آنرا بیشتر می‌کرد ولی در عوض چالشهایی بین این استاندارد و 802.11 پدید آورد. نقطه شدت این چالش آنجاست که این دو سیستم از لحاظ الکتریکی با یکدیگر تداخل فرکانسی دارند. اشاره به این نکته مهم است که شرکت هیولت پاکارد چندین سال قبل از آن شبکه‌ای مبتنی بر نور مادون قرمز برای وصل بی‌سیم دستگاه‌های جانبی کامپیوتر عرضه کرده بود، ولی استقبال چندانی از آن نشد.

فارغ از همه اینها، در ژولای ۱۹۹۹ گروه طراح بلوتوث، مشخصات هزار و پانصد صفحه‌ای از نسخه ۱ آن یعنی V1.0 را منتشر نمود. به فاصله کوتاهی، گروه استانداردسازی IEEE که در اندیشه تدوین استاندارد 802.15 برای «شبکه‌های شخصی بی‌سیم» بودند مستندات استاندارد بلوتوث را به عنوان مبنای کار خود برگزیدند و شروع به پالایش و تکمیل آن نمودند. اگر چه استانداردسازی چیزی که مشخصات تفصیلی و مشروح آن در اختیار است و پیاده‌سازیهای متنوع و ناسازگار ندارد (که نیاز به یکنواخت‌سازی و هماهنگی داشته باشد) عجیب به نظر می‌رسد

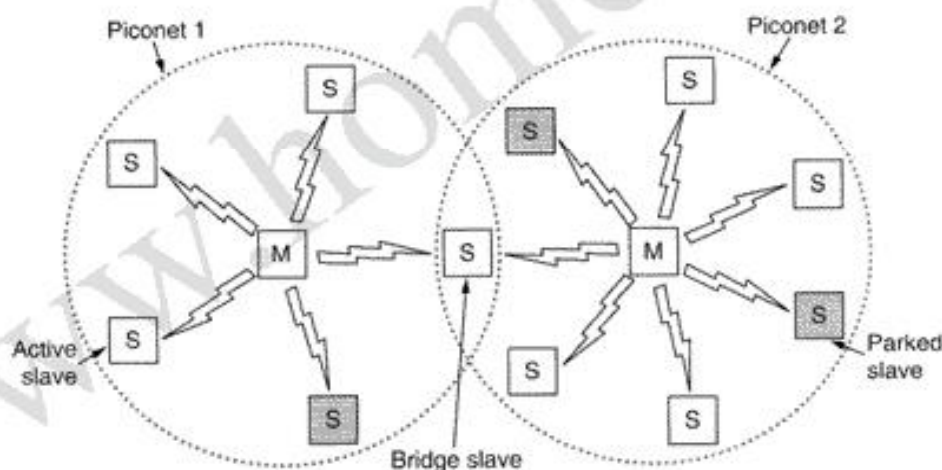
۱. SIG به معنای گروهی با گرایش خاص یا همان کنسرسیوم است.

ولی تاریخ نشان داده که وجود یک «استاندارد باز» که توسط سازمانی بی طرف مثل IEEE تدوین و مدیریت می شود عموماً کاربری یک تکنولوژی را ترویج و ترغیب خواهد کرد. اگر بخواهیم اندکی دقیقتر سخن بگوئیم باید اشاره کنیم که توصیف استاندارد بلوتوث برای سیستمی کامل تدوین شده که از لایه فیزیکی تا لایه کاربرد را در بر می گیرد در حالیکه کمیته IEEE 802.15 فقط لایه های فیزیکی و پیوند داده را استاندارد سازی کرده و باقیمانده پشته پروتکلی خارج از برنامه این استاندارد است.

هرچند IEEE اولین «استاندارد شبکه شخصی»^۱ (PAN) را در سال ۲۰۰۲ با عنوان 802.15.1 به تصویب رساند ولی هنوز کنسرسیوم بلوتوث فعال و سرگرم بهبود و توسعه آنست. اگر چه نسخه استاندارد عرضه شده توسط کنسرسیوم بلوتوث و IEEE یکی نیستند ولی انتظار می رود بزودی به یک استاندارد واحد همگرا شوند.

۱-۶-۴ معماری بلوتوث

اجازه بدهید بررسی سیستم بلوتوث را با مروری سریع بر دستاوردها و اهداف آن آغاز نمائیم. واحد پایه در سیستم بلوتوث یک «پیکونت» (Piconet) است که از یک «گره اصلی» (Master Node) و حداکثر هفت «گره پیرو فعال» (Active Slave Node) به فاصله حداکثر ده متر، تشکیل شده است. در یک فضای بزرگ و واحد می توان چندین پیکونت داشت و حتی می توان آنها را از طریق یک گره که نقش پل (Bride) ایفاء می کند به هم متصل کرد (به شکل ۳۵-۴ نگاه کنید). به مجموعه ای از پیکونتهای متصل بهم اصطلاحاً Scatternet (شبکه متفرق/پراکنده) گفته می شود.



شکل ۳۵-۴. دو پیکونت می توانند با اتصال بهم یک Scatternet تشکیل بدهند.

در یک پیکونت علاوه بر هفت گره فعال پیرو، می تواند تا ۲۵۵ گره غیرفعال وجود داشته باشد. اینها دستگاه هایی هستند که گره اصلی آنها را در حالت استراحت و کم توان وارد کرده تا مصرف باتری آن کاهش یابد. یک ایستگاه در حالت غیر فعال هیچ کاری نمی تواند انجام بدهد به جز آنکه به سیگنال فعال سازی خود یا سیگنال Beacon که از گره اصلی می رسد، پاسخ بدهد. به غیر از این حالات، دو حالت میانی در مصرف توان به نامهای حالت Hold و Sniff نیز وجود دارد که در اینجا بدان نخواهیم پرداخت.

دلیل طراحی Master/Slave (اصلی/پیرو) آن بود که طراحان آن در نظر داشتند قیمت کل سیستم بلوتوث پیاده سازی شده بر روی تراشه، زیر پنج دلار باشد. نتیجه این تصمیم گیری آنست که گره های پیرو [مثل صفحه

کلیدها، موس، چاپگر] تقریباً غیر هوشمند و ساده هستند و اساساً آنچه را که گره اصلی (Master) به آنها دستور بدهد اجرا می کنند. یک پیکوننت سیستمی مبتنی بر TDM متمرکز (Centralized TDM) است که در آن هسته مرکزی (یعنی گره اصلی یا Master) بر سیگنال ساعت نظارت دارد و تعیین می کند که چه دستگاهی و در کدام برش زمانی (Slot) مخابره داشته باشد. تبادل اطلاعات صرفاً بین گره مرکزی و گره های پیرو انجام می شود و ارتباط مستقیم دو گره پیرو [مثلاً دو صفحه کلید یا دو چاپگر] ممکن نیست.

۲-۶-۴ کاربردهای بلوتوث

بیشتر پروتکل های شبکه فقط کانالی را بین چند مولفه مخابراتی، سازماندهی و ایجاد می کنند و اجازه می دهند طراحان برنامه های کاربردی به پیاده سازی هر آنچه که مورد نیاز است بپردازند. به عنوان مثال 802.11 مشخص نکرده که کاربران باید صرفاً از کامپیوتر کیفی خود برای خواندن ایمیل یا جستجو در وب یا هر چیز دیگری بهره بگیرند. برعکس، در تشریح بلوتوث نسخه ۱.۱ از ۱۳ کاربرد مختلف که باید از آنها پشتیبانی شود، نام برده شده و برای هر یک، پشته پروتکلی متفاوتی ارائه گردیده است. متأسفانه این راهکار به پیچیدگی بسیار زیاد منتهی می شود و ما از آن صرف نظر خواهیم کرد. این سیزده کاربرد که «پروفایل» نام گرفته اند در شکل ۳۶-۴ فهرست شده اند. با نگاهی اجمالی به پروفایلها ممکن است به آنچه که کنسرسیوم بلوتوث در پی انجام آن بوده بیشتر پی ببریم.

نام پروفایل	عملکرد
Generic access	پروسیجرهایی برای مدیریت لینک
Service discovery	پروتکلی برای کشف سرویسهای عرضه شده
Serial port	جایگزینی برای کابل معمولی پورت سریال
Generic object exchange	مدل ارتباطی بین سرویس دهنده و مشتری برای جابجایی (انتقال) اشیاء را تعریف می کند.
LAN access	پروتکل ارتباطی بین کامپیوتر همراه و شبکه محلی ثابت (با کابل سیمی)
Dial-up networking	امکان برقراری تماس یک کامپیوتر کیفی را از طریق تلفن همراه فراهم می آورد.
Fax	امکان ارتباط بین یک دستگاه دورنگار بی سیم و تلفن همراه را فراهم می آورد.
Cordless telephony	ارتباط بین یک دستگاه گوشی تلفن بی سیم و ایستگاه ثابت و محلی آن را برقرار می کند.
Intercom	امکانانی برای واکتی تاکی دیجیتال
Headset	امکان ارتباط از طریق هندزفری (Handsfree) را فراهم می آورد.
Object push	روشی برای مبادله اشیاء ساده
File transfer	عرضه کننده امکانات عمومی بیشتر جهت انتقال فایل
Synchronization	امکان سنکرون سازی داده های یک PDA با کامپیوتری دیگر را فراهم می آورد.

شکل ۳۶-۴. پروفایلهای بلوتوث.

«پروفایل عمومی دسترسی» (Generic Access) حقیقتاً یک برنامه کاربردی نیست بلکه بیشتر یک زیربناست که بر اساس آن برنامه های کاربردی حقیقی ساخته و پیاده می شوند. وظیفه اصلی آن ارائه تمهیداتی است که بتوان بین گره اصلی (Master) و گره های پیرو (Slave) یک کانال مطمئن برقرار و آنرا حفظ کرد. «پروفایل تشخیص خدمات» (Service Discovery) که آنهم تقریباً عمومی و کلی است توسط دستگاهها برای آگاهی از خدماتی که دیگر دستگاهها ارائه می دهند، استفاده می شود. تمام دستگاههای مبتنی بر بلوتوث موظف به پیاده سازی این دو پروفایل هستند. بقیه پروفایلها اختیاریند.

«پروفایل درگاه سریال» (Serial Port) یک پروتکل انتقال است که بقیه پروفایلها از آن بهره می گیرند. این

پرو فایل یک درگاه سریال را شبیه سازی می کند و بطور خاص برای کاربردهای قدیمی که به خط سریال نیاز دارند، سودمند است.

«پرو فایل عمومی مبادله شیء» (Generic Object Exchange) یک ارتباط مبتنی بر مدل مشتری / سرویس دهنده، برای انتقال داده ها تعریف کرده است. اگرچه همیشه مشتری، آغاز کننده عملیات است ولیکن یک گره پیرو می تواند هم سرویس دهنده و هم مشتری باشد. همانند پرو فایل «درگاه سریال» این پرو فایل نیز زیربنای دیگر پرو فایلهاست.

گروه سه تایی پرو فایلهای بعدی به منظور کاربردهای شبکه ای (Networking) تعریف شده اند. «پرو فایل دسترسی به LAN» اجازه می دهد که یک دستگاه مبتنی بر بلوتوث به یک شبکه ثابت متصل شود. این پرو فایل رقیب مستقیم 802.11 است. «پرو فایل شبکه مبتنی بر شماره گیری» (Dialup) انگیزه اصلی کل این پروژه بوده است. این پرو فایل اجازه می دهد که یک کامپیوتر کیفی بتواند به یک تلفن همراه که دارای مودم داخلی بی سیم است متصل شود. «پرو فایل دورنگار» (Fax) شبیه به پرو فایل شماره گیری است با این تفاوت که اجازه می دهد ماشینهای دورنگار بی سیم از طریق یک دستگاه تلفن همراه و بدون نیاز به سیم، اقدام به ارسال یا دریافت دورنگار کنند.

سه پرو فایل بعدی در خصوص تلفن کاربرد دارند. پرو فایل «تلفن بی سیم» (Cordless Telephony) راهی را برای اتصال گوشی یک تلفن بی سیم به ایستگاه ثابت است. در حال حاضر تلفنهای بی سیم خانگی را نمی توان به عنوان تلفن همراه به کار گرفت ولی در آینده شاید تلفن بی سیم و تلفن همراه در هم ادغام شوند. «پرو فایل Intercom» این امکان را فراهم می کند تا دو تلفن، شبیه به «واکی تاکی» (Walkie/Talkie) بهم متصل گردند. نهایتاً «پرو فایل گوشی - Headset» امکان ارتباط بی سیم بین گوشی و ایستگاه ثابت (مثل هندزفری - Hands Free) را فراهم می کند که به عنوان مثال برای صحبت با تلفن در حین رانندگی مفید است.

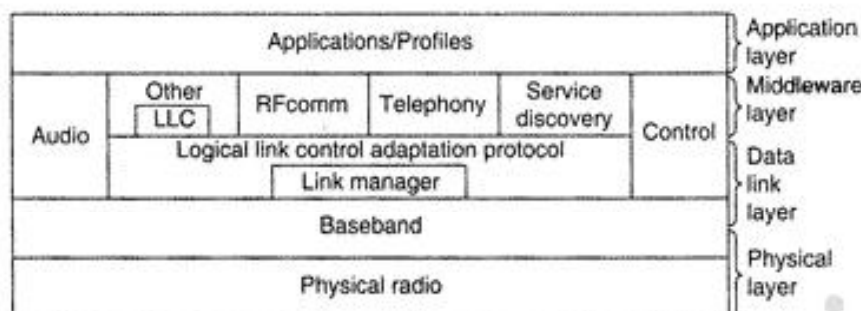
سه پرو فایل باقیمانده برای مبادله اشیاء بین دو ابزار تعریف شده است. اشیاء می توانند فایلهای داده، تصویر یا کارتهای تجاری باشند. «پرو فایل سنکرواسیون»، برای بار کردن داده در درون کامپیوتر کیفی یا PDA (کامپیوترهای دستی) در حین ترک منزل و جمع آوری اطلاعات پس از برگشت، مفید است.

آیا واقعاً نیاز بوده که تمام این کاربردها به تفصیل تحلیل شوند و برای هر کدام پشته پروتکلی متفاوتی تعریف گردد؟ شاید نه! اما بخشهای متفاوت این استاندارد را گروه های کاری مختلف طراحی کرده اند و چون هر یک از این گروه ها بر روی نیازهای خاص خود متمرکز بودند، در نتیجه هر یک پرو فایل مورد نظر خود را پدید آوردند. برای توجیه این موضوع به قانون کانوی (Conway's Law) بپردازیم. (در یکی از شماره های مجله Datamation در آوریل ۱۹۶۸، ملوین کانوی مشاهدات خود را بدین نحو منتشر کرد که اگر n شخص را به نوشتن یک کامپایلر بگمارید، آنچه که بدست خواهید آورد یک کامپایلر n -pass است [یعنی کامپایلری که در آن تعداد مراحل ترجمه یک برنامه، n گذر می باشد]. عبارت عام ساختار نهانی یک نرم افزار آینه تمام نمای ترکیب گروهی است که آن را تولید کرده اند. شاید می شد که به جای سیزده پشته پروتکلی به دو پشته کلی بسنده کرد: یکی برای انتقال فایل و دیگری برای انتقال بی درنگ جریان اطلاعات.

۳-۶-۴ پشته پروتکلی بلوتوث

استاندارد بلوتوث پروتکل های متعددی دارد که بطور ناموزون در چند لایه گروه بندی شده اند. ساختار لایه ها از مدل OSI، مدل TCP/IP، مدل 802 یا هر مدل شناخته شده دیگر تبعیت نمی کند. با این وجود IEEE در حال اصلاح بلوتوث است تا با مدل 802 سازگارتر شود. معماری پروتکل بلوتوث که توسط کمیته 802 اصلاح شده، در شکل ۴-۳۷ مشاهده می شود.

لایه زیرین، «لایه رادیو فیزیکی» است که تقریباً متناظر با لایه فیزیکی از مدل OSI یا مدل 802 می باشد. این



شکل ۴-۳۷. نسخه ۸۰۲.۱۵ از معماری پروتکل بلوتوث.

لایه با انتقال رادیویی و مدولاسیون سرکار دارد. بسیاری از ملاحظات که در طراحی این لایه باید مورد توجه قرار می گرفت آن بود که سیستم ارزان قیمت باشد و بطور انبوه در بازار عرضه شود.

«لایه باند پایه» (Baseband) از جهاتی شبیه به زیرلایه MAC است ولیکن مولفه هایی از لایه فیزیکی را نیز در بر می گیرد. این لایه با مسائلی مثل چگونگی نظارت گره اصلی (Master) بر برشهای زمانی و چگونگی گروه بندی این برشهای زمانی در قالب فریمها، سرکار دارد.

سپس لایه ای شامل یک گروه از پروتکل های مرتبط با هم، تعریف شده است. مدیر لینک (Link Manager) عملیات ایجاد کانال های منطقی بین دستگاه ها، شامل «مدیریت توان مصرفی»، «احراز هویت» (Authentication) و «کیفیت خدمات» (QoS) را بر عهده دارد. «پروتکل تطبیق کنترل لینک منطقی» (که اغلب L2CAP گفته می شود) وظیفه دارد لایه های بالایی را از درگیری با جزئیات ارسال، راحت کند. این لایه مشابه با استاندارد زیرلایه LLC ۸۰۲ است ولی از لحاظ مسائل فنی با آن متفاوت است. دو پروتکل «کنترل» و «صدا» همانگونه که از نامشان بر می آید با مسائل انتقال صدا و عملیات کنترل سروکار دارند. برنامه های کاربردی می توانند بدون نیاز به L2CAP، مستقیماً این دو پروتکل را به خدمت بگیرند.

لایه بعدی یک لایه میانی است (Middleware) و تلفیقی از پروتکل های متفاوت را در بر می گیرد. در این لایه از پروتکل IEEE ۸۰۲ LLC، بمنظور سازگاری با دیگر شبکه های سری ۸۰۲ استفاده شده است. پروتکل های RFcomm، Telephony و Service Discovery صرفاً مرتبط با بلوتوث هستند. RFcomm (Radio Frequency Communication)، پروتکلی جهت شبیه سازی استاندارد درگاه سریال (Serial port) است که در تمام PCها از آن برای اتصال صفحه کلید، موس، مودم و امثال آن استفاده می شود. این پروتکل برای آن طراحی شده تا بتوان از دستگاه های قدیمی به سبب سهولت استفاده کرد. «پروتکل تلفنی» (Telephony) پروتکلی بی درنگ است که برای سه پروفایل مبتنی بر انتقال صدا بکار می آید. این پروتکل همچنین تنظیم و قطع ارتباط را بر عهده دارد. نهایتاً پروتکل «تشخیص خدمات» (Service Discovery) برای کشف و تشخیص انواع خدماتی که درون شبکه عرضه می شود، کاربرد دارد.

بالا ترین لایه، محل قرار گرفتن انواع برنامه های کاربردی و پروفایلها است. این لایه برای انجام کار از خدمات پروتکل های موجود در لایه های زیر بهره می گیرد. هر برنامه کاربردی، زیر مجموعه ای از پروتکل های مختص به خود را به خدمت می گیرد. ابزارهای ویژه ای مثل گوشی بی سیم (Headset) بسته به نوع برنامه کاربردی آنها، فقط به برخی از پروتکل ها نیازمندند.

در بخشهای بعدی سه لایه پائینی از پشته پروتکلی بلوتوث را بررسی خواهیم کرد چرا که تقریباً متناظر با زیرلایه های فیزیکی و MAC است.

۴-۶ لایه رادیویی در بلوتوث

لایه رادیویی بیتها را از گره اصلی به گره پیرو و بالعکس، منتقل می کند. این لایه، سیستمی با توان کم و برد ده متر است که در باند فرکانسی 2.4GHz ISM عمل می کند. این باند به ۷۹ کانال یک مگاهرتزی تقسیم می شود. مدولاسیون به کاررفته FSK (Frequency Shift Keying) و هر هرتز (هر سیکل) معادل یک بیت است که جمعاً نرخ یک مگابیت بر ثانیه را در اختیار می گذارد ولی بیشتر این پهنای باند به دلیل سربار تلف می شود. برای تخصیص مناسب این کانالها از روش پرش فرکانس در طیف گسترده (Spread Spectrum) با نرخ پرش 1600 hops/sec و Dwell time معادل ۶۲۵ میکروثانیه بهره گرفته شده است.

چون بلوتوث و 802.11، هر دو در باند 2.4GHz ISM و دقیقاً در همان ۷۹ کانال کار می کنند لذا با یکدیگر تداخل فرکانسی خواهند داشت. از آنجایی که پرش فرکانس در بلوتوث سریعتر از 802.11 است لذا دستگاه های مبتنی بر بلوتوث به احتمال بیشتری در انتقال 802.11 اختلال خواهد کرد تا 802.11 در بلوتوث! چون 802.11 و 802.15 هر دو استانداردهای IEEE هستند لذا IEEE به دنبال راه حلی برای این مشکل می گردد ولی حل این مشکل چندان ساده نیست چرا که هر دو سیستم، بدلیل مشابهی از این باند فرکانسی بهره گرفته اند؛ زیرا، برای استفاده از این باند فرکانسی، به اخذ هیچ مجوزی نیاز نیست. استاندارد 802.11a از باند دیگر ISM (باند 5GHz) استفاده می کند ولیکن برد کمتری نسبت به 802.11b دارد (بدلیل ماهیت فیزیکی امواج رادیویی این باند) لذا استفاده از 802.11a راه حل مناسبی نخواهد بود. برخی از شرکتها این مشکل را با ممنوعیت استفاده از بلوتوث حل کرده اند. راه حل بازاری این مشکل آنست که صبر کنیم تا عاقبت شبکه ای که از لحاظ اقتصادی و سیاسی جایگاه مستحکم تری در بازار پیدا کرد، از طرف مقابل بخواهد تا استاندارد خود را برای حل مشکل تداخل، اصلاح نماید. در این خصوص مطالبی در مرجع (Lansford et al., 2001) ارائه شده است.

۴-۵ لایه باند پایه در بلوتوث

لایه باند پایه شبیه ترین بخش بلوتوث با زیرلایه MAC است. این لایه، دنباله بیتهای خام را به فریمها تبدیل می کند و بدین منظور چندین قالب مهم فریم تعریف نموده است. در ساده ترین حالت، گره اصلی در هر پیکونت دنباله ای از برشهای زمانی ۶۲۵ میکروثانیه ای (Time Slot) تولید می کند، با این توصیف که ارسال داده های گره اصلی در برشهای زمانی با شماره زوج انجام می شود و گره های پیرو (Slaves) در برشهای زمانی فرد ارسال می نمایند. این روش مشابه با روش تسهیم زمانی (TDM) معمولی است که در آن، گره اصلی نیمی از برشهای زمانی را در اختیار دارد و بقیه گره ها (حداکثر هفت گره) در نیم دیگر سهیم هستند. ارسال هر فریم می تواند ۱، ۳ یا ۵ برش زمانی طول بکشد.

در هر پرش فرکانسی ۲۵۰ تا ۲۶۰ میکروثانیه طول خواهد کشید تا مدار رادیویی بتواند پایدار شود. پایداری سریعتر نیز ممکن است ولی هزینه پیاده سازی بیشتری دارد. برای فریمهای که فقط به یک برش زمانی نیاز دارند پس از هر پرش فرکانسی، ۳۶۶ بیت از کل ۶۲۵ بیت باقی خواهد ماند. از این مقدار ۱۲۶ بیت به «کد دسترسی» (Access Code) و «سرآیند» اختصاص دارد و ۲۴۰ بیت، برای داده ها باقی می ماند. وقتی پنج برش زمانی به هم ملحق می شوند [برای ارسال فریمهایی به طول ۵ برش] تنها به یک زمان پایداری نیاز خواهد بود و طبعاً زمان کوتاهتری برای زمان پایداری، تلف می شود و $5 \times 625 = 3125$ بیت در پنج برش زمانی ارسال می گردد که از این مقدار ۲۷۸۱ بیت برای ارسال داده در اختیار «لایه باند پایه» خواهد بود. بنابراین در بلوتوث فریمهای طولانی کارآمدتر از فریمهای کوچک هستند.

هر فریم بر روی یک کانال منطقی که اصطلاحاً «لینک بین گره اصلی و گره پیرو» نام دارد، ارسال خواهد شد. دو نوع لینک وجود دارد: لینک اول ACL است (Asynchronous Connection-Less) که برای ارسال داده ها در

برشهای زمانی نامنظم کاربرد دارد. این داده‌ها از لایه L2CAP در سمت فرستنده تولید و در سمت گیرنده تحویل لایه L2CAP می‌شوند. ترافیک داده‌های ACL مبتنی بر روش «بیشترین تلاش» (Best Effort) ارسال می‌شود ولی هیچ تضمینی در تحویل آنها نیست. فریم‌ها می‌توانند گم شده یا از بین بروند، بدون آنکه ارسال مجدد شوند. هر گره پیر و فقط می‌تواند یک لینک ACL با گره اصلی داشته باشد.

لینک دیگر، SCO (Synchronous Connection Oriented) نام دارد که برای ارسال داده‌های بی‌درنگ، مثل ارتباط تلفنی کاربرد دارد. این نوع کانال با تخصیص برشهای زمانی مشخص در هر دو جهت، ایجاد می‌شود. بدلیل آنکه لینکهای SCO نسبت به زمان حساس هستند فلذا فریمهای ارسالی بر روی این لینک هرگز ارسال مجدد (Retransmit) نخواهند شد، در عوض از روش «تصحیح مستقیم خطا» استفاده شده تا اطمینان بیشتری داشته باشد. [ارسال مجدد فریمهای خراب بر عهده لایه بعدی است و در این لایه -در صورت امکان- به تصحیح خطا بسنده می‌شود. -م] هر گره پیر و می‌تواند حداکثر سه لینک SCO با گره اصلی داشته باشد. هر لینک SCO می‌تواند یک کانال صدا مبتنی بر PCM با نرخ 64000 بیت بر ثانیه را حمل کند.

۴-۶-۴ لایه L2CAP در بلوتوث

لایه L2CAP سه دسته عملیات مهم را بر عهده دارد: اول آنکه بسته‌هایی با طول حداکثر ۶۴ کیلوبایت را از لایه‌های بالائی پذیرفته و آنها را جهت انتقال، به فریمهای کوچکتری می‌شکند. در سمت مقابل این فریمها مجدداً به بسته اصلی بازسازی خواهند شد.

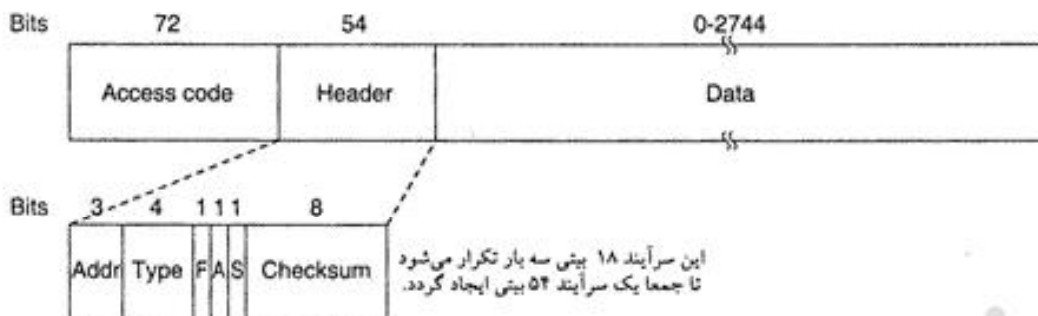
دوم آنکه این لایه عمل جمع‌آوری و توزیع بسته‌هایی که از چندین مبدا آمده [یا به چندین مقصد می‌روند] را بر عهده دارد. وقتی یک بسته بازسازی می‌شود، لایه L2CAP تعیین خواهد کرد که باید به کدام پروتکل در لایه بالاتر (مثلاً RFcomm یا Telephony) تحویل شود.

سوم آنکه این لایه، عملیات تأمین «کیفیت خدمات» (QoS) را بر عهده دارد، چه در هنگام ایجاد لینک و چه در خلال عملکرد طبیعی. همچنین در زمان ایجاد لینک، بر سر اندازه حداکثر و مجاز طول داده، توافق صورت می‌گیرد تا ابرازهایی با طول بسته بزرگ از ارسال چنین بسته‌ای به ابزارهایی با طول بسته کوچک اجتناب کنند. از آنجایی که تمام ابزارهای مبتنی بر بلوتوث نمی‌توانند بسته‌هایی با طول ۶۴ کیلوبایت را بپذیرند فلذا به این ویژگی نیاز است.

۴-۶-۵ ساختار فریم در بلوتوث

چندین نوع قالب فریم در بلوتوث وجود دارد که مهمترین آنها در شکل ۴-۳۸ نشان داده شده است. این فریم با فیلد «کد دسترسی» (Access Code) شروع می‌شود که عموماً هویت یک گره اصلی (Master) را مشخص خواهد کرد تا بدینگونه یک گره پیر و که در برد رادیویی دو گره اصلی قرار دارد، گیرنده حقیقی ترافیک داده‌ها را مشخص نماید. سپس یک سرآیند ۵۴ بیتی آمده که شامل فیلدهای معمولی زیرلایه MAC است. سپس فیلد داده قرار گرفته که حداکثر ۲۷۴۴ بیت را (برای انتقال در پنج برش زمانی) در بر می‌گیرد. در فریمهایی که تنها در یک برش زمانی ارسال می‌شوند، قالب فریم همین است با این تفاوت که فیلد داده آنها حداکثر ۲۴۰ بیت است.

حال اجازه بدهید به فیلدهای سرآیند، نگاهی سریع بیندازیم. «فیلد آدرس» هویت گیرنده فریم را از بین هشت دستگاه فعال در هر پیکونت مشخص می‌کند. «فیلد Type»، اولاً نوع فریم را (از بین انواع ACL، SCO، POLL، یا NUI.L)، ثانیاً روش تصحیح خطای داده‌ها را و ثالثاً تعداد برشهای زمان که ارسال فریم جاری بدان نیاز دارد را مشخص می‌نماید. «بیت Flow» توسط گره‌های پیر و زمانی تنظیم [فعال] می‌شود که بافر آنها پر شده باشد و نتوانند داده بیشتری دریافت کنند. این بیت، شکل ابتدائی کنترل جریان داده‌ها، به حساب می‌آید. «بیت



شکل ۳۸۴. قالب کلی فریم داده در بلوتوث.

Acknowledgement بدین منظور است تا دریافت صحیح یک فریم از طرف مقابل، در فریم ارسالی جاسازی و اعلام شود [یعنی فرآیند Piggybacking]. «بیت Sequence» برای شماره گذاری فریمهاست تا بسته های تکراری کشف شوند. در بلوتوث پروتکل ارسال مجدد، روش «توقف و انتظار» (Stop & wait) است و طبقاً یک بیت برای شماره گذاری فریمها کفایت می کند. در ادامه «فیلد هشتم بیتی Checksum» برای کشف خطای احتمالی در سرآیند تعریف شده است. کل این سرآیند ۱۸ بیتی سه بار تکرار می شود تا سرآیند ۵۴ بیتی نشان داده شده در شکل ۳۸۴-۴ بوجود آید. در سمت گیرنده، با یک مدار ساده سه نسخه تکراری هر بیت بررسی می شود. اگر هر سه بیت مثل هم بودند، آن بیت پذیرفته می شود در غیر اینصورت، بیتی که بیشترین تکرار را دارد قبول می شود. بدین ترتیب، برای ارسال یک سرآیند ۱۰ بیتی ظرفیتی معادل ۵۴ بیت صرف می شود. دلیل آن این بوده که برای ارسال مطمئن داده ها در محیطی سرشار از نویز و با استفاده از ابزاری ارزان قیمت و توانی ناچیز (2.5mW) و قدرت پردازش پائین، به افزودن بسیار زیادی نیاز خواهد بود.

برای فیلد داده در فریمهای ACL، قالبهای متفاوتی تعریف شده است. فریمهای SCO ساده تر هستند: فیلد داده همیشه ۲۴۰ بیتی است. سه گزینه دیگر نیز تعریف شده که در آنها مقدار واقعی داده ها ۸۰، ۱۶۰ یا ۲۴۰ بیتی است و باقیمانده بیتها برای تصحیح خطا به کار می آیند. در مطمئن ترین نسخه (یعنی داده ۸۰ بیتی)، بخش داده همانند سرآیند، سه بار متوالی تکرار می شود.

از آنجایی که گرو پیرو (Slave) تنها می تواند از برشهای زمانی با شماره فرد استفاده نماید فلذا در هر ثانیه ۸۰۰ برش زمانی بدست خواهد آورد. (همینطور گره اصلی) بدین ترتیب با فیلد داده ۸۰ بیتی، ظرفیت کانال از سمت گره پیرو به سمت گره اصلی معادل ۶۴۰۰۰ بیت بر ثانیه خواهد بود (ظرفیت کانال از گره اصلی به گره پیرو نیز ۶۴۰۰۰ بیت است) لذا این کانال دقیقاً برای یک کانال صوتی دو طرفه مبتنی بر PCM کافی است. (دلیل انتخاب نرخ 1600 Hops/sec برای تغییر فرکانس همین بوده است). این اعداد و ارقام بدین معنا هستند که یک کانال صوتی دو طرفه PCM با نرخ ۶۴۰۰۰ بیت بر ثانیه، در حالت مطمئن [یعنی وقتی داده های ۸۰ بیتی با سه بار تکرار ارسال می شوند]، کل پهنای باند موجود در پیکونوت را (علیرغم پهنای باند 1 Mbps آن)، اشباع خواهد کرد. با گزینه نامطمئنتر (یعنی ۲۴۰ بیت در هر برش زمانی بدون هیچگونه افزودنی یا تکرار) می توان از حداکثر سه کانال صوتی همزمان حمایت کرد و به همین دلیل حداکثر سه لینک SCO برای هر گره پیرو مجاز شمرده شده است.

مطالب فراوانتری در خصوص بلوتوث می توان گفت که از حوصله این کتاب خارج است. برای آگاهی بیشتر

به مراجع زیر مراجعه نمائید.

Bhagwat, 2001; Bisdikian, 2001; Bray and Sturman, 2002; Haartsen, 2000; Johansson et al., 2001; Miller and Bisdikian, 2001; Sairam et al., 2002)

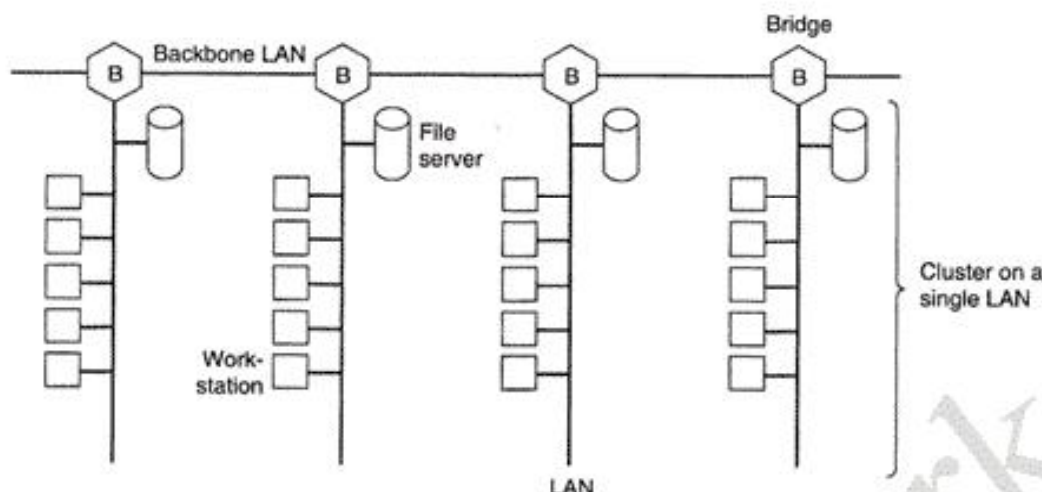
۷-۴ هدایت در سطح لایه پیوند داده ها (Data Link Layer Switching)

بسیاری از سازمانها دارای LANهای متعددی هستند و تمایل دارند آنها را به هم متصل کنند. شبکه های محلی (LAN) را می توان از طریق دستگاه هایی که در لایه پیوند داده ها عمل می کنند و «پل» (Bridge) نامیده می شوند به هم متصل نمود. «پلها» برای مسیریابی و هدایت داده ها، آدرسهای «لایه پیوند داده ها» را بررسی می نمایند. از آنجایی که قرار نیست محتوای فیلد داده (از فریمهایی که باید هدایت شوند) بررسی گردد لذا این فریمها می توانند بسته های IPv4 (که اکنون در اینترنت به کار می رود)، IPv6 (که در آینده در اینترنت به کار گرفته خواهد شد)، بسته های AppleTalk، ATM، OSI یا هر نوع بسته دیگر را در خود حمل کنند. برخلاف پل، «مسیریابها» آدرس درون بسته ها را بررسی کرده و بر این اساس، آنها را هدایت (مسیریابی) می کنند. اگرچه این تعریف تمایز بین «مسیریاب» و «پل» را تعیین می کند ولی پیشرفتهای جدیدی مثل ابداع شبکه LAN مبتنی بر سوئیچ (Switched Ethernet)، آب را گل کرده و به نحوی که در بخشهای آتی بدانها خواهیم پرداخت این تمایز را ابهام آلود کردند! در بخشهای بعدی به پلها و سوئیچها و بالاخص آنهایی که برای اتصال شبکه های محلی 802 به کار می آیند، نگاهی خواهیم انداخت. برای بررسی دقیق پلها، سوئیچها و عناوین مهم در این خصوص، به مراجع (Perlmán, 2000) مراجعه نمایند.

قبل از پرداختن به تکنولوژی پل، بررسی شرایطی که در آن، استفاده از پلها سودمند است، خالی از لطف نخواهد بود. شش دلیل ارائه می کنیم که چرا یک سازمان واحد، ممکن است دارای چندین LAN باشد. اول آنکه بسیاری از دانشگاهها و بخشهای مختلف شرکتها، LAN مختص به خود را دارند تا بتوانند کامپیوترهای شخصی، ایستگاههای کاری (Workstation) و سرورهای دهنده های خاص خود را به هم متصل کنند. از آنجایی که بخشهای مختلف یک موسسه، اهداف متفاوتی را دنبال می کنند لذا در هر بخش، فارغ از آنکه دیگر بخشها چه می کنند، LAN متفاوتی پیاده می شود. دیر یا زود نیاز می شود که این LANها با یکدیگر تعامل و ارتباط داشته باشند. در این مثال، پیدایش LANهای متعدد ناشی از اختیار و آزادی مالکان آن بوده است. دوم آنکه ممکن است سازمانها به صورت جغرافیایی در ساختمانهایی با فاصله قابل توجه، پراکنده باشند. شاید داشتن چندین LAN مجزا در هر ساختمان و وصل آنها از طریق «پلها» و لینکهای لیوری ارزان تر از کشیدن یک کابل واحد بین تمام سایتها تمام شود.

سوم آنکه گاهی برای تنظیم بار و تعدیل ترافیک، لازم است که یک LAN منطقی و واحد به چندین LAN کوچکتر تقسیم شود. به عنوان مثال در بسیاری از دانشگاهها هزاران ایستگاه کاری در اختیار دانشجویان و هیئت علمی قرار گرفته است. عموماً فایلها در ماشینهای سرور دهنده فایل نگهداری می شوند و حسب تقاضای کاربران بر روی ماشینشان منتقل و بارگذاری می شود. مقیاس بسیار بزرگ این سیستم مانع از آن می شود که بتوان تمام ایستگاههای کاری را در یک شبکه محلی واحد قرار داد چرا که پهنای باند مورد نیاز بسیار بالا خواهد بود. در عوض، مشابه با شکل ۴-۳۹ از چندین LAN که توسط «پل» به هم متصل شده، استفاده می شود. هر شبکه LAN گروهی از ایستگاهها و سرور دهنده فایل خاص خود را دربرمی گیرد که بدین ترتیب، بیشتر ترافیک در حوزه یک LAN واحد محدود می شود و بار زیادی به ستون فقرات شبکه اضافه نخواهد شد.

اشاره به این نکته ارزشمند است که اگرچه عموماً شبکه های LAN را به صورت یک کابل چنداتصالی (Multidrop) با ساختار باس ترسیم می کنیم (نمایش کلاسیک) ولیکن امروزه اغلب آنها توسط هاب و خصوصاً سوئیچها پیاده سازی می شوند. با این حال یک کابل طولانی با چندین ماشین متصل به آن، با یک هاب که ماشینها را از درون، بهم متصل می کند، از لحاظ عملکرد یکسان هستند. در هر دو حالت تمام ماشینها به یک «حوزه تصادم» (Collision Domain) یکسان متعلقند و تمام آنها برای ارسال فریم از پروتکل CSMA/CD استفاده می کنند.



شکل ۳۹-۴. چندین LAN از طریق یک ستون فقرات بهم متصل شده اند تا ظرفیت کل حمل بار آن از ظرفیت یک LAN واحد بیشتر شود.

شبکه های مبتنی بر سوئیچ متفاوت هستند و اگر چه قبلاً آنها را بررسی کرده ایم ولی باز هم نگاهی به آنها خواهیم انداخت .

چهارم آنکه در برخی از شرایط اگرچه یک شبکه محلی واحد از نظر حجم بار کفایت می کنند ولیکن فاصله فیزیکی بین ماشینهای دور، بسیار زیاد است. (مثلاً بیش از ۲/۵ کیلومتر در اترنت). حتی اگر عملیات کابل کشی ساده باشد ولیکن شبکه، در اثر تاخیر بسیار زیاد رفت و برگشت سیگنال (Round Trip Delay) کار نخواهد کرد. تنها راه حل آنست که LAN به چند بخش تقسیم شده و بین آنها پل نصب گردد. با استفاده از پل، می توان فاصله فیزیکی کل شبکه را افزایش داد.

پنجم مسئله قابلیت اعتماد است؛ بر روی یک LAN واحد، یک گره خراب که دنباله ای پیوسته از خروجی اشغال تولید می کند قادر است کل شبکه را فلج نماید. پلها را می توان در نقاط حساس قرار داد تا یک گره خراب و مغشوش نتواند کل سیستم را مختل کند. برخلاف یک تکرارکننده (Repeater) که ورودی خود را بی قید و شرط باز تولید می نماید یک پل را می توان به نحوی برنامه ریزی کرد تا در خصوص آنچه که هدایت می کند یا هدایت نمی کند تصمیم آگاهانه بگیرد.

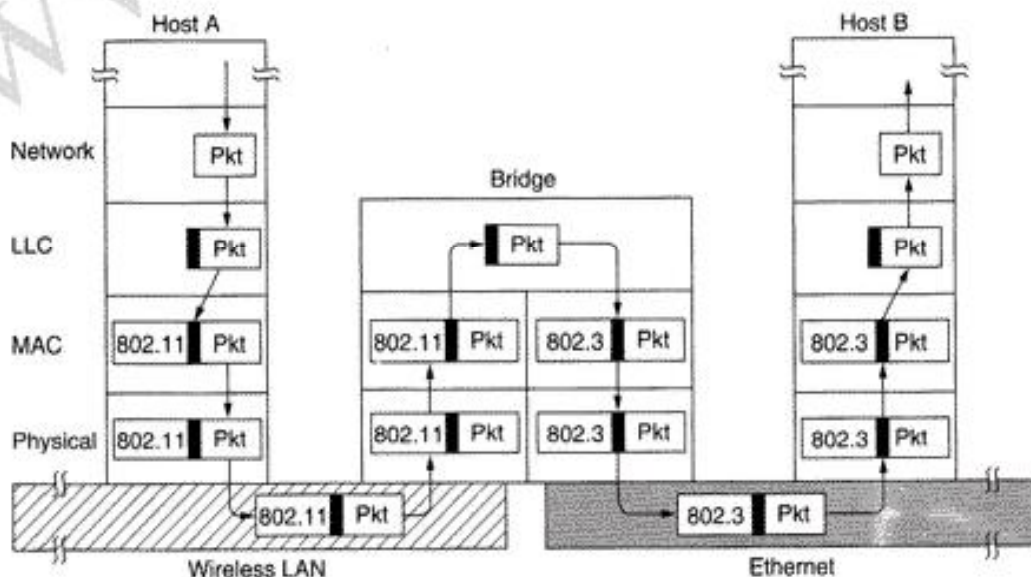
ششم و آخر آنکه پلها می توانند به امنیت اطلاعات در یک سازمان کمک نمایند. بیشتر کارتهای واسط شبکه های LAN دارای حالتی به نام «حالت بی قید» (Promiscuous mode) هستند که در چنین حالتی تمام فریمهای جاری بر روی شبکه تحویل گرفته می شود، نه فریمهایی که دقیقاً به آدرس او ارسال شده اند. جاسوسان و فضولان به این ویژگی علاقمند هستند. با قرار دادن پلها در نقاط مختلف و اطمینان از عدم هدایت اطلاعات حساس به بخش های نامطمئن، مسئول سیستم می تواند بخشهایی از شبکه را از دیگر بخشها جدا کرده تا ترافیک آنها به خارج راه پیدا نکرده و در اختیار افراد نامطمئن قرار نگیرد.

هدف آرمانی آنست که پلها کاملاً نامرئی (شفاف-transparent) باشند، بدین معنا که بتوان ماشین را از یک بخش از شبکه به بخش دیگر منتقل کرد بدون آنکه به هیچگونه تغییری در سخت افزار، نرم افزار یا جداول پیکربندی نیاز باشد. همچنین باید این امکان وجود داشته باشد که تمام ماشینهای یک بخش از شبکه بتوانند فارغ از آنکه نوع LAN آنها چیست با ماشینهای بخش دیگر، مبادله اطلاعات داشته باشند. این هدف گاهی برآورده می شود ولیکن نه همیشه!

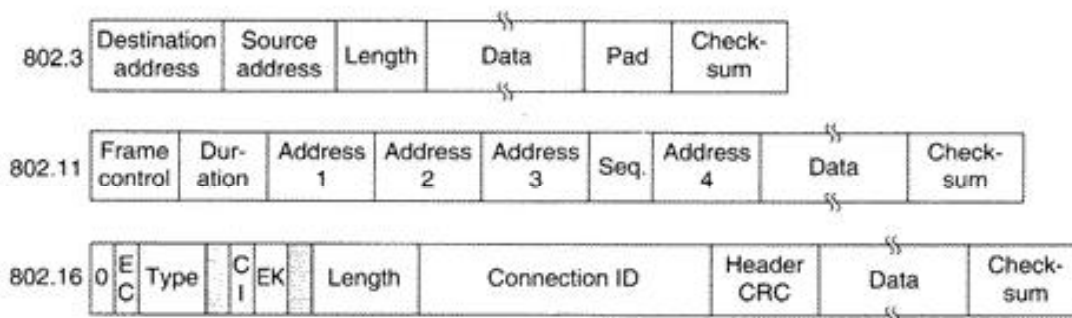
۴-۷-۱ پلهانی از 802.x به 802.y

پس از بررسی آنکه چرا به پلهای نیاز است اجازه بدهید بدین سوال بپردازیم که عملکرد آنها چگونه است؟ شکل ۴-۴۰ عملکرد یک پل ساده با دو درگاه (Port) را به تصویر کشیده است. ماشین میزبان A در یک شبکه محلی بی سیم، بسته‌ای برای ارسال به ماشین میزبان و ثابت B در شبکه اترنت (802.3) (که از طریق پل به شبکه بی سیم متصل شده) آماده کرده است. این بسته از زیرلایه LLC عبور کرده و سرآیند LLC به آن اضافه می‌شود (این سرآیند در شکل به صورت سیاه نشان داده شده است). سپس به زیرلایه MAC تحویل شده و در آنجا نیز سرآیند 802.11 به ابتدای آن افزوده می‌شود. (یک بخش انتهایی نیز به آخر فریم اضافه خواهد شد که در شکل نشان داده نشده است). این واحد داده، در هوا منتشر و توسط ایستگاه ثابت دریافت می‌شود؛ پس از بررسی، ایستگاه ثابت متوجه می‌گردد که باید آنرا به سمت اترنت ثابت هدایت کند. پس از رسیدن آن به پل (که شبکه 802.11 را به شبکه 802.3 متصل کرده)، پل کار دریافت آن از لایه فیزیکی را شروع کرده و آنرا به سمت زیرلایه‌های بالا هدایت می‌کند. در زیرلایه MAC از پل، سرآیند 802.11 حذف می‌شود. بسته اصلی (به همراه سرآیند LLC) تحویل زیرلایه LLC از پل می‌شود. در این مثال، بسته به سمت شبکه محلی 802.3 رهسپار است، لذا بسته از طریق بخش 802.3 در پل به سمت پائین حرکت کرده و نهایتاً بروی شبکه اترنت منتقل می‌شود. دقت کنید که یک پل که k شبکه مختلف را به هم متصل می‌کند دارای k زیرلایه MAC و تعداد k لایه فیزیکی است (به ازای هر نوع یکی). تا اینجا به نظر می‌رسد که انتقال فریم از یک LAN به LAN دیگر ساده است اما واقعیت این چنین نیست. در این بخش، برخی از دشواریهای ساخت یک پل را که انواع مختلف شبکه‌های LAN (و همچنین شبکه MAN) سری 802 را به هم متصل می‌کند، متذکر می‌شویم. ما بروی شبکه‌های 802.3، 802.11 و 802.16 متمرکز خواهیم شد ولی انواع دیگر آن هم وجود دارد که هر کدام مسائل خاص خود را دارند.

برای شروع باید یادآوری کرد که هر LAN قالب فریم خاص خود را دارد. (شکل ۴-۴۱ را ببینید) اگرچه تفاوتی که بین قالب فریم در شبکه‌های اترنت، «توکن رینگ» (Token Ring) و «توکن باس» (Token Bus) وجود دارد بیشتر ناشی از انکاء به نفس شرکتهای بزرگ ابداع‌کننده آنها (یعنی به ترتیب زیراکس، آی‌بی‌ام و جنرال‌موتورز) و همچنین شرایط زمانی آن دوران بوده ولیکن تفاوت شبکه‌های کنونی تقریباً لازم و



شکل ۴-۴۰. عملکرد یک پل از شبکه 802.11 به 802.3.



شکل ۴-۴. انواع قالب فریمهای IEEE 802 (طول هر فریم در شکل، مقیاس اندازه واقعی آن نیست)

واقعی است. به عنوان مثال فیلد Duration (طول زمان) که در 802.11 وجود دارد ناشی از ماهیت پروتکل MACAW و همچنین عدم توانایی شنود کانال در این شبکه می باشد (برخلاف اترنت). در نتیجه، انتقال یک فریم بین دو شبکه متفاوت LAN، مستلزم دگرگونی در قالب فریمهاست که طبعاً نیاز به زمان CPU، محاسبه مجدد کدهای جدید کشف خطا (Checksum) دارد و این احتمال نیز پدید می آید که در اثر خرابی بیتها در حافظه پل، خطاهای غیرقابل کشف، داده‌ها را آلوده کند.

مشکل دوم آنست که LANهای به هم متصل شده الزاماً با سرعت مشابهی کار نمی کنند. وقتی «پل» یک دنباله پی در پی از فریمها را از LAN سریع برای هدایت به LAN کندتر می پذیرد قادر نخواهد بود با همان سرعتی که فریم را دریافت می کند از دست آنها رهائی یابد. به عنوان مثال هرگاه یک شبکه اترنت گیگابیتی با بالاترین سرعت، بیتها را به سوی شبکه یازده مگابیت در ثانیه ای LAN 802.11b روانه کند، پل باید بتواند آنها را موقتاً در حافظه نگهداری کند و این حافظه نباید سرریز شود. [که با چنین سرعتی بعید نیست.] در ضمن، برخی پلها سه یا چندین شبکه LAN را به هم متصل می کنند. در چنین پلهایی اگر چندین شبکه بطور همزمان تلاش کنند فریمهای خود را به یک LAN مشابه بفرستند مشکل سرریز شدن حافظه بوجود خواهد آمد، حتی اگر تمام LANها با سرعت یکسانی کار کنند.

مشکل سوم که خطرترین مشکل بالقوه پلها محسوب می شود آنست که در شبکه های محلی و مختلف سری 802، حداکثر طول فریم، متفاوت است. این مشکل زمانی بروز می کند که یک فریم بزرگ باید به سوی شبکه ای هدایت شود که آن LAN قادر به دریافت آن نیست. شکستن فریم به تعدادی قطعه، خارج از حیطه وظایف این لایه است. در تمام پروتکل های این لایه فرض بر آنست که فریمها یا می رسند یا نمی رسند و هیچ تمهیدی برای شکستن یک فریم به قطعات و بازسازی آن اندیشیده نشده است. نمی توان گفت که چنین پروتکلی ابداع نشده است. چنین پروتکلی ابداع شده و وجود هم دارد ولیکن در حیطه وظایف پروتکل های پیونده داده نیست. پلها نیز نباید به محتوای هر فریم کاری داشته باشند. اساساً (در چنین وضعیتی) راه حلی برای این مشکل وجود ندارد. فریمهای بسیار طولانی باید به جای انتقال، حذف شوند. [جهت اجتناب از هرگونه مداخله در پیگیری سخت افزار یا نرم افزار]

نکته بعدی امنیت داده هاست. شبکه های 802.11 و 802.16، هر دو از رمزنگاری در سطح لایه پیوند داده ها پیشنهادی می کنند در حالیکه اترنت چنین امکانی ندارد. بدین معنا که خدمات متنوع رمزنگاری عرضه شده در شبکه های بی سیم، در خلال ورود ترافیک به شبکه اترنت از دست می رود. بدتر از آن، اینکه اگر یک دستگاه در شبکه بی سیم از رمزنگاری در سطح لایه پیوند داده ها بهره گرفته باشد هیچ راهی برای رمزگشایی آن هنگام

دریافت در یک ایستگاه اترنت وجود ندارد. از طرفی اگر ایستگاه بی سیم از رمزنگاری استفاده نکند ترافیک داده‌ها از طریق لینک هوایی در معرض شنود همگان قرار می‌گیرد.

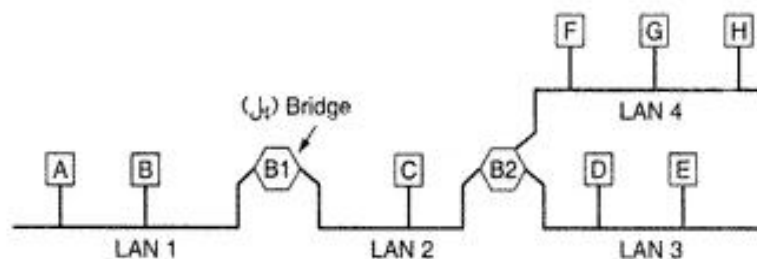
یک راه حل برای مشکل امنیت آنست که رمزنگاری در لایه‌های بالایی انجام شود ولیکن در این روش ایستگاه 802.11 باید بداند که آیا با ایستگاه دیگری در شبکه 802.11 صحبت می‌کند (تا از رمزنگاری در لایه پیوند داده بهره بگیرد) یا نه (تا از چنین امکانی استفاده نکند). وادار کردن ایستگاه به تصمیم‌گیری «شفافیت» را از بین خواهد برد.

نکته آخر مسئله «کیفیت خدمات» است. هر دو شبکه 802.11 و 802.16 این خدمات را به نحو متفاوتی در اختیار می‌گذارند: اولی در «حالت PCF» و دومی با استفاده از «اتصال با نرخ ارسال ثابت» (Constant Bit Rate Connection). در شبکه اترنت چیزی به نام کیفیت خدمات بی مفهوم است و بدین ترتیب ترافیک هر یک از این دو شبکه در حین عبور از اترنت «کیفیت خدمات» را از دست می‌دهند. [به عنوان مثال اگر چه می‌توان میزان حداکثر تاخیر در دو شبکه اول را تضمین کرد ولی در اترنت چنین تضمینی وجود ندارد و با وصل این دو شبکه نمی‌توان تاخیر کل را تضمین نمود و مقدار آن تصادفی خواهد بود. -م]

۲-۷-۴ بهم‌بندی شبکه‌ها به صورت محلی (Local Internetworking)

در بخش قبلی به مشکلاتی که در وصل دو شبکه محلی نوع IEEE 802 بروز می‌کند، پرداختیم. با این وجود در سازمانهای بزرگ با شبکه‌های LAN متعدد، متصل کردن تمام آنها به یکدیگر مشکلات گوناگونی را بوجود می‌آورد، حتی اگر تمام آنها از نوع اترنت باشند. حالت آرمانی آنست که بتوان به راحتی از سازمان بیرون رفت و چند پل مبتنی بر استاندارد IEEE خریداری کرد، سپس با وصل تمام کابلها به پلها، شبکه فوراً و به درستی بکار بیفتد. نباید به هیچ تغییر سخت‌افزاری، نرم‌افزاری، تنظیم سوئیچها، بازگذاری یا تغییر در جداول مسیریابی یا پارامترهای آن یا هر تغییر دیگری نیاز باشد. فقط باید کابلها را وصل کرد و رفت! به علاوه عملکرد طبیعی هیچکدام از شبکه‌های LAN نباید تحت الشعاع قرار بگیرد. به عبارت دیگر، پلها بایستی کاملاً «شفاف» باشند (یعنی از دیدگاه سخت‌افزار و نرم‌افزار غیرقابل رویت باشند). شگفت آنکه این کار عملاً ممکن است. حال اجمالاً بررسی کنیم که این کار جادوئی به چه نحو انجام می‌شود!

در ساده‌ترین حالت، یک «پل شفاف» در «حالت بی‌قید» (Promiscuous Mode) عمل کرده و تمام فریمهای جاری بر روی تمام LANهایی را که بدانها متصل است، می‌پذیرد. به عنوان یک مثال، به پیکربندی شکل ۴-۴۲ توجه نمایید. پل B1 به شبکه‌های محلی ۱ و ۲ متصل شده و پل B2 نیز به شبکه‌های محلی ۲ و ۳ و ۴ متصل است. فریمی که به پل B1 می‌رسد ولی مقصد آن ماشین A است می‌تواند فوراً توسط پل نادیده انگاشته شود. (حذف شود) زیرا این فریم بر روی LAN صحیحی [که به مقصد ختم می‌شود] قرار گرفته است ولی فریمی که از LAN 1 به مقصد C یا F دریافت می‌شود باید منتقل شود.



شکل ۴-۴۲. یک پیکربندی از شبکه‌های متصل بهم با چهار شبکه محلی و دو پل.

وقتی فریمی دریافت می شود، پل در ابتدا باید تصمیم بگیرد که آیا باید آنرا حذف کند یا باید آنرا منتقل نماید؛ سپس در صورت نیاز به انتقال، باید مشخص شود که به کدام LAN هدایت گردد. این تصمیم گیری با جستجوی آدرس مقصد درون یک جدول بزرگ در حافظه پل انجام می گیرد. (به این جدول، Hash Table گفته می شود). این جدول فهرست تمام ماشینهای مقصد را در اختیار دارد و می تواند تعیین کند که این ماشینها به کدامیک از خطوط پل تعلق دارند. مثلاً جدول پل B2 می تواند مشخص کند که ماشین A به LAN 2 تعلق دارد، زیرا تمام آنچه که لازم است B2 بداند آنست که فریمهایی با مقصد A را بر روی چه شبکه ای ارسال کند. در واقع برای پل B2 مهم نیست که این فریم، بعداً چگونه هدایت و منتقل می شود.

وقتی پلها برای اولین بار به کار می افتند تمام جدول Hash خالی هستند. هیچیک از پلها نمی دانند که هر یک از ماشینهای مقصد در کجا قرار گرفته اند، لذا برای انتقال فریم از «الگوریتم سیل آسا» (Flooding Algorithm) استفاده می نمایند یعنی تمام فریمهای ورودی که مقصدشان ناشناخته است بر روی تمام شبکه هائی که پل بدانها متصل است، ارسال می شود (البته به استثنای شبکه ای که فریم از آن دریافت شده است). به مرور زمان، پل متوجه خواهد شد که هر ماشین مقصد، در کجا قرار گرفته است. (طبق الگوریتمی که در زیر بدان اشاره خواهیم کرد). هرگاه یک ماشین مقصد شناسائی شد، فریمهایی که بعداً بدان مقصد روانه می شوند توسط پل بر روی LAN مناسب هدایت خواهد شد و از روش سیل آسا استفاده نمی شود.

الگوریتمی که توسط پلهای شفاف به کار گرفته می شود روش «یادگیری غیرمستقیم» (Backward Learning) است. قبلاً اشاره شد که پل در «حالت بی قید» کار می کند لذا هر فریمی را که بر روی یکی از شبکه های متصل به او مبادله می شود، می بیند (و همچنین دریافت و پردازش می نماید). با نگاهی به آدرس مبدا هر فریم، پل می تواند بفهمد که کدام ماشین از طریق کدام LAN قابل دسترسی است. به عنوان مثال در شکل ۴-۲ فرض کنید که پل B1 فریمی را بر روی LAN 2 می بیند که توسط C تولید شده است؛ بدین ترتیب متوجه می شود که می توان از طریق LAN 2 به C رسید و بدین ترتیب در جدول Hash خود درج می کند که فریمهای روانه به سمت C باید از طریق LAN 2 منتقل و هدایت شوند. بعداً تمام فریمهایی که به مقصد C و از طریق LAN 1 به پل وارد می شوند منتقل می شوند ولی فریمهایی که به مقصد C ولی از طریق LAN 2 به پل می رسند حذف خواهند شد.

با خاموش یا روشن شدن پلها و ماشینها و یا جابجائی آنها در سطح شبکه، توپولوژی شبکه تغییر می کند. برای آنکه توپولوژی شبکه به صورت پویا دنبال شود وقتی یک «درایه» (Entry) در جدول هر پل درج می گردد زمان دریافت فریم نیز در آن «درایه» یادداشت می شود. هرگاه بعداً فریمی که آدرس مبدا آن قبلاً در جدول درج گردیده، دریافت شود زمان درج شده در درایه متناظر آن، با زمان فعلی بهنگام سازی می شود. بدین ترتیب زمان درج شده در هر «درایه» (Entry) آخرین زمانی که فریمی از آن ماشین دریافت شده را تعیین می کند.

بطور متناوب، یک پروسه در درون پل، جدول Hash را جستجو و پویش می کند و تمام درایه هائی را که برای بیش از چند دقیقه به هنگام نشده اند، حذف می نماید. بدین ترتیب اگر یک کامپیوتر از شبکه خود جدا و در ساختمان جابجا شده یا به شبکه دیگری متصل شود پس از گذشت چند دقیقه عملیات عادی خود را از سر می گیرد و نیازی به مداخله و تنظیمات دستی نیست. البته این الگوریتم بدین معنا هم هست که اگر کامپیوتری برای چندین دقیقه ساکت باشد (یعنی هیچ داده ای نفرستد) درایه متناظر با او از جدول پلها حذف شده و از آن به بعد هر ترافیکی که برایش ارسال می شود بروش سیل آسا هدایت خواهد شد مگر آنکه بعداً خودش فریمی ارسال کند [و آدرس او در جدول پلها درج شود. -م]

روند مسیریابی فریمهای ورودی پل، به شبکه LAN مبدا (Source LAN) و شبکه LAN مقصد (Destination LAN) بستگی دارد. این روال به ترتیب ذیل است:

۱. اگر شبکه LAN مبدأ و مقصد یکسان هستند فریم را نادیده بگیر.

۲. اگر شبکه های مبدأ و مقصد متفاوت هستند فریم را منتقل کن.

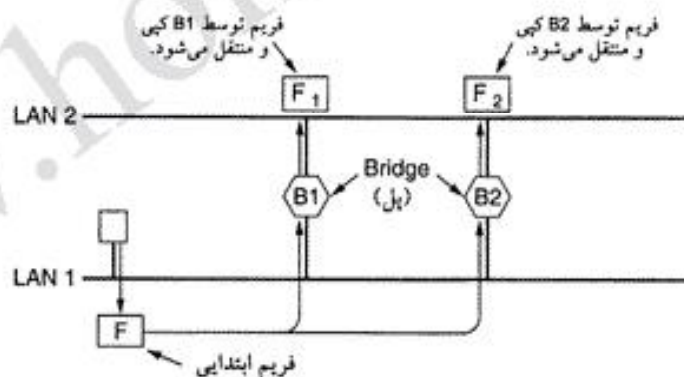
۳. اگر شبکه مقصد ناشناخته است بروش «سیل آسا» عمل کن.

این الگوریتم به ازای ورود هر فریم باید یکبار اعمال گردد. یک تراشه خاص VLSI عملیات جستجو و به هنگام سازی جدول و درایه های (Entries) را در عرض چند میکروثانیه انجام می دهد.

۴-۳-۷ پلهای مبتنی بر درخت پوشا (Spanning Tree)

برخی از سایتها برای افزایش قابلیت اعتماد، بین دو LAN دو یا چند پل موازی نصب می کنند. (به شکل ۴-۲۳ نگاه کنید). ولیکن چنین آرایشی مشکلات دیگری را ایجاد خواهد کرد چراکه در ساختار توپولوژی، حلقه ایجاد می شود.

چنین مشکلاتی را می توان یکمک مثال شکل ۴-۲۳ و با مشاهده روند هدایت فریمی از ماشین F به مقصدی ناشناخته، تحلیل کرد. هر پل طبق قاعده عمومی در مواجهه با فریمهایی که به مقصد ناشناخته روانه هستند از روش «سیل آسا» استفاده می کند یعنی در اینجا فریم بر روی LAN 2 منتقل می شود. [فریم منتقل شده بر روی LAN 2 را F₂ بنامید]. اندکی بعد، پل ۱ فریم F₂ را می بیند که مقصد آن ناشناخته است و آنرا بر روی LAN 1 هدایت می کند و فریم F₃ تولید می شود. (فریم F₃ در شکل نشان داده نشده است). بروش مشابه پل ۲ فریم F₁ را بر روی LAN 1 منتقل کرده و F₄ تولید می شود. (F₄ نیز نشان داده نشده است) [F₄، F₃، F₂، F₁ نسخه های مشابه با فریم F هستند که توسط پل منتقل می شوند]. حال پل ۲ فریم F₄ را منتقل و پل ۲ فریم F₃ را منتقل می نماید و این دور باطل تا ابد ادامه دارد.



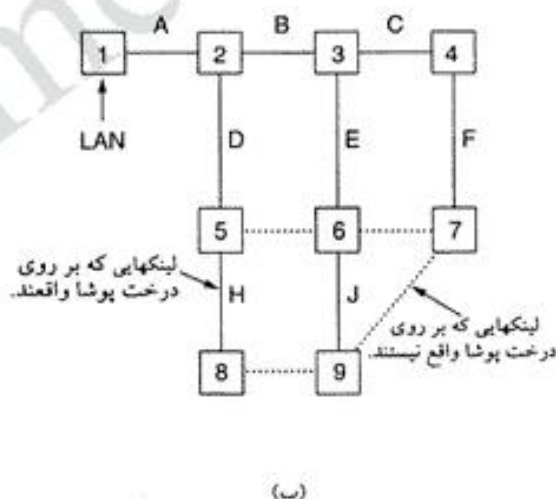
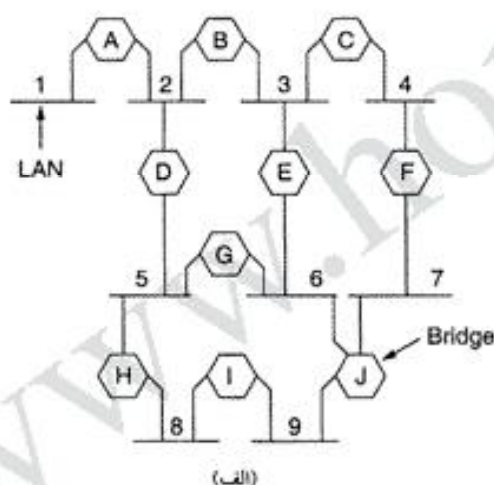
شکل ۴-۲۳. دو پل شفاف (نامرئی) موازی.

راه حل این مشکل آنست که پلهای با یکدیگر ارتباط و محاوره داشته باشند و توپولوژی واقعی را به صورت یک «درخت پوشا» (Spanning Tree) که در آن به تمام LANها مسیر دسترسی وجود دارد، در نظر بگیرند. در نتیجه برخی از اتصالات بالقوه که بین LANها وجود دارد، نادیده گرفته می شود تا مجازاً یک توپولوژی بدون حلقه ایجاد شود. به عنوان مثال در شکل ۴-۲۴-الف، ث شبکه LAN را می بینیم که توسط ده پل متصل شده اند. این یکریبندی را می توان در قالب گرافی در نظر گرفت که در آن، هر کدام از LANها یک «گره» تلقی می شوند. هر «کمان» (Arc) اتصال دو LAN توسط پل را نشان می دهد. با حذف برخی از «کمانها» (که در شکل ۴-۲۴-ب به صورت خط چین نشان داده شده)، این گراف به یک درخت پوشا کاهش می یابد. با استفاده از این درخت، بین دو LAN دقیقاً یک مسیر وجود دارد. پس از آنکه پلهای درخت پوشا را تشکیل دادند، هدایت تمام فریمها (بین

شبکه های محلی) از ساختار درخت پوشا تبعیت خواهد کرد. از آنجایی که بین هر مبدا و مقصد در شبکه فقط و فقط یک مسیر وجود دارد لذا ایجاد حلقه غیرممکن است.

برای ایجاد درخت پوشا، پلها بایستی یک پل را به عنوان ریشه این درخت انتخاب کنند. جهت این انتخاب، پلها شماره سریال خود را که توسط کارخانه سازنده تنظیم و یکتا بودن آن در کل دنیا تضمین شده است، به صورت فراگیر (Broadcast) به اطلاع همه می رسانند. پل که دارای کوچکترین شماره سریال است به عنوان «ریشه» انتخاب می شود. سپس درختی با کوتاهترین مسیر که از ریشه شروع شده و تمام پلها و LANها را در بر می گیرد، ایجاد می شود. این درخت همان «درخت پوشا» است. اگر یک پل یا شبکه LAN از کار بیفتد، این درخت از نو محاسبه می شود.

نتیجه این الگوریتم آنست که بین هر LAN و ریشه و بدین ترتیب از ریشه به بقیه LANها، یک مسیر یکتا ایجاد می شود. اگر چه این درخت تمام LANها را در بر می گیرد ولیکن تمام پلها لزوماً در این درخت قرار نمی گیرند (برای اجتناب از حلقه). حتی پس از ایجاد درخت پوشا و در خلال عملکرد طبیعی، این الگوریتم به صورت متناوب اجرا می شود تا هر گونه تغییر در توپولوژی به صورت خودکار تشخیص داده شده و درخت اصلاح شود. این الگوریتم توزیع شده که برای ایجاد درختهای پوشا مورد استفاده قرار می گیرد توسط خانم رابیدا پرلمن ابداع و به تفصیل در مرجع (Perlman, 2000) تشریح شده است. همچنین این الگوریتم در IEEE 802.1D استاندارد سازی شده است.

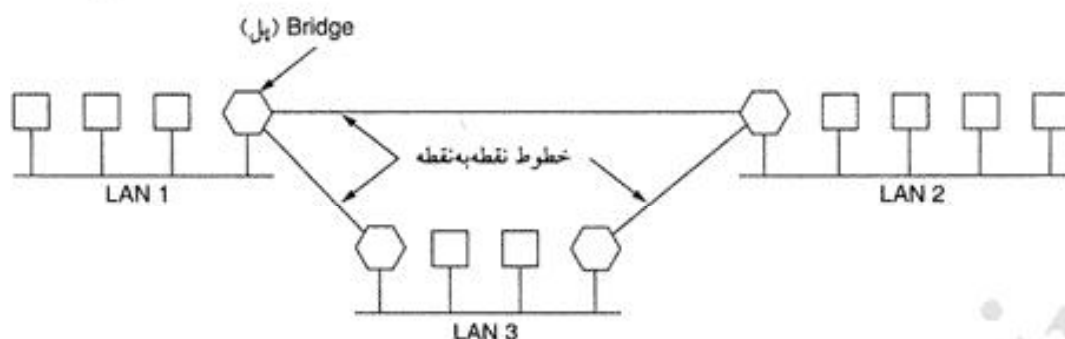


شکل ۴-۴. (الف) چند شبکه LAN بهم متصل (ب) یک درخت پوشا که تمام شبکه های LAN را در بر می گیرد. (خطوط نقطه چین جزو درخت پوشا نیستند).

۴-۷-۲ پلهای راه دور (Remote Bridges)

کاربرد متداول پلها آنست که دو یا چند LAN راه دور را بهم متصل کنند. بعنوان مثال ممکن است یک شرکت دارای کارخانه هایی در چند شهر باشد و هر کارخانه، LAN مختص به خود را داشته باشد. در حالت آرمانی باید تمام LANها به هم متصل شده باشند تا کل سیستم نقش یک LAN عظیم را ایفاء کند.

این هدف با قرار دادن یک پل بین هر دو LAN و وصل کردن پلها به وسیله خطوط نقطه به نقطه (مثلاً بکمک خطوط اجاره ای عرضه شده توسط شرکت های تلفن) برآورده خواهد شد. در شکل ۴-۴، یک سیستم ساده با سه LAN به تصویر کشیده شده است. در اینجا روشهای رایج مسیریابی اعمال می شود. ساده ترین راه برای تحلیل این



شکل ۴-۵۴. برای اتصال شبکه های محلی راه دور می توان از پلهای راه دور بهره گرفت.

ساختار آنست که سه خط نقطه به نقطه را به مثابه یک شبکه LAN بدون هیچ ماشین میزبان (Hostless LAN)، تصور کنید. در این صورت یک سیستم معمولی با شش LAN داریم که با چهار پل به هم متصل شده اند. در هیچ کجا از مطالبی که تاکنون مطالعه کرده ایم گفته نشده که حتماً باید به یک LAN ماشین میزبان متصل شده باشد. برای خطوط نقطه به نقطه می توان از پروتکل های متنوعی بهره گرفت. یک انتخاب آنست که از یکی از استانداردهای موجود برای خطوط نقطه به نقطه مثل PPP استفاده شود و کل فریمهای MAC [شامل سرآیند و فیلدهای پایانی] درون فیلد حمل داده آن (یعنی Payload) قرار بگیرد. این استراتژی بشرط آنکه تمام LAN ها مثل هم باشند، به نحو احسن کار می کند و تنها مسئله باقیمانده آنست که فریمها به LAN صحیح تحویل شوند. انتخاب دیگر آنست که سرآیند و پی آیند^۱ هر فریم MAC در پل مبداء حذف شود و باقیمانده در فیلد حمل داده از فریم مربوط به پروتکل نقطه به نقطه قرار بگیرد. سپس در پل مقصد سرآیند و پی آیند MAC جدیدی برای آن قطعه داده تولید شود. اشکال این روش آنست که کد کشف خطایی که فریم در حین دریافت در ماشین مقصد دارد همانی نیست که توسط ماشین مبداء تولید شده و بدین ترتیب ممکن است خطاهایی که در حافظه پل، داده ها را آلوده می کند کشف نشود.

۵-۷-۴ تکرارکننده^۲، هاب^۳، پل^۴، سوئیچ^۵، مسیریاب^۶ و دروازه^۷

تا اینجا کتاب روشهای گوناگونی را مرور کردیم که وظیفه همگی تحویل فریمها یا بسته ها از یک بخش کابل به بخش دیگر است. همچنین اشاره ای به تکرارکننده ها، پلها، سوئیچها، هابها، مسیریابها و دروازه ها داشتیم. از تمام این ابزارها به یک منظور استفاده می شود ولیکن تفاوت های مشهود و نامشهود زیادی دارند. از آنجایی که این ابزارها بسیار متنوعند، لذا مروری بر همه آنها و بررسی شباهتها و تفاوت های آنها ارزشمند خواهد بود.

برای شروع باید گفت که این ابزارها (به نحوی که در شکل ۴-۴۶ الف دیده می شود) در لایه های متفاوتی عمل می کنند. بسته به آنکه هر یک از این ابزارها در چه لایه ای عمل می کنند مکانیزم هدایت اطلاعات متفاوت است. در قالب یک نمایشنامه عمومی: کاربر مقداری اطلاعات برای ارسال به یک ماشین راه دور تولید می نماید. این داده ها تحویل لایه انتقال (Transport Layer) شده و بدان سرآیند لازم اضافه می گردد (مثلاً سرآیند TCP). سپس، واحد اطلاعاتی حاصل به سمت لایه پائین یعنی لایه شبکه عبور داده می شود. لایه شبکه سرآیند خاص خود را بدان افزوده و یک بسته مخصوص لایه شبکه ساخته می شود (مثلاً یک بسته IP). در شکل ۴-۴۶ ب یک بسته IP را می بینیم که به صورت خاکستری نشان داده شده است. این بسته تحویل لایه پیوند داده (Data Link)

۱. Header and Trailer

۲. Repeater

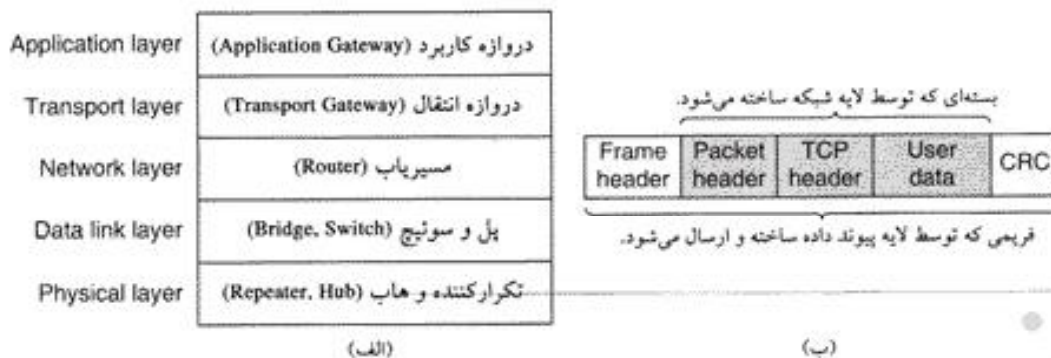
۳. Hub

۴. Bridge

۵. Switch

۶. Router

۷. Gateway

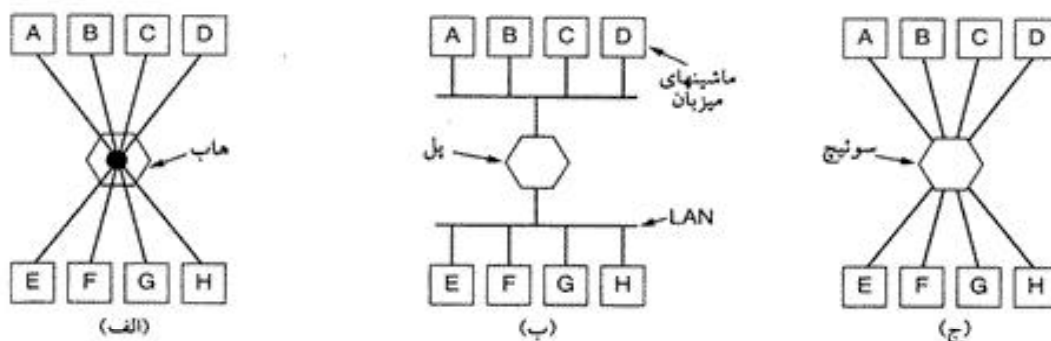


شکل ۴-۴. (الف) جایگاه هر ابزار در پشته پروتکلی (ب) فریمها، بسته ها و سرآیندها.

می شود و آن هم سرآیند خاص خود و کد کشف خطا (CRC) را بدان افزوده و فریم حاصل را جهت ارسال به لایه فیزیکی تسلیم می کند. (مثلاً برای ارسال بر روی LAN)

حال اجازه بدهید نگاهی به ابزارهای هدایت اطلاعات بیندازیم و ببینیم که این ابزارها چه ارتباطی با بسته ها و فریمها دارند. در پائین ترین سطح یعنی در لایه فیزیکی به «تکرارکننده ها» بر می خوریم. تکرارکننده ابزاری است آنالوگ، که دو قطعه کابل را بهم متصل می کند. سیگنالی که بر روی یکی از این قطعات ظاهر گردد، تقویت (بازتولید) شده و بر روی قطعه دیگر قرار داده می شود. تکرارکننده ها هیچگونه درکی از «فریم»، «بسته» یا «سرآیند» ندارند. آنها صرفاً با مفهوم «ولت» آشنا هستند! به عنوان مثال در اترنت کلاسیک اجازه داده شده برای افزایش طول حداکثر کابل از ۵۰۰ متر به ۲۵۰۰ متر از چهار تکرارکننده استفاده شود.

سپس به هاب (Hub) می رسیم: یک هاب معمولی (از نوع غیرفعال)، دارای تعدادی خط ورودی است که این خطوط از لحاظ الکتریکی در درون هاب به هم متصل شده اند. فریمی که از یک خط ورودی دریافت می شود بر روی خطوط دیگر ارسال خواهد شد. هرگاه دو فریم بطور همزمان به هاب ارسال شوند، تصادم رخ خواهد داد؛ دقیقاً همانند اتفاقی که بر روی کابل کواکسیال می افتد. به عبارت دیگر، کابل هاب یک «حوزه تصادم» (Collision Domain) واحد را تشکیل خواهد داد. تمام خطوط ورودی هاب، باید با سرعت یکسانی کار کنند. هابها متفاوت از تکرارکننده ها هستند، از آن جهت که هاب (معمولاً) سیگنالهای ورودی را تقویت نمی کند و طراحی آنها به گونه ای است که چندین کارت واسط خط (Line Card) دارند و هر یک از کارتها خود چندین ورودی دارند ولیکن در مجموع این تفاوتها ناچیز است. همانند تکرارکننده ها، هابها نیز آدرسهای 802 [آدرسهای MAC] را بررسی نکرده و به هیچ وجه از آنها استفاده نمی کنند. در شکل ۴-۴ الف تصویری نمادین از یک هاب نشان داده شده است.



شکل ۴-۴. (الف) یک هاب (ب) یک پل (ج) یک سوئیچ.

حال اجازه بدهید به سمت لایه پیوند داده یعنی لایه‌ای که در آن «پلها» و «سوئیچها» تعریف شده‌اند حرکت کنیم. قبلاً «پلها» را تا حدی مطالعه کردیم. یک پل دو یا چند شبکه LAN را همانند شکل ۴-۴۷-ب به هم متصل می‌کند. وقتی فریمی دریافت می‌شود نرم‌افزار درون پل، آدرس مقصد را از سرآیند فریم استخراج و آنرا درون جدول خود جستجو می‌کند تا محلی را که فریم باید بدانجا ارسال شود، بیابد. در اترنت، این آدرس همان فیلد ۴۸ بیتی آدرس در شکل ۴-۱۷ است. شبیه به هاب، پل‌های پیشرفته نیز دارای «کارت‌های خط» (Line Card) هستند که معمولاً چهار تا هشت خط ورودی از یک نوع شبکه معین در آنها تعبیه شده است. یک «کارت خط اترنت» (Ethernet Line Card) نمی‌تواند مثلاً فریم‌های شبکه توکن رینگ را بپذیرد چرا که نمی‌داند در کجای سرآیند فریم آدرس مقصد را پیدا کند! با این وجود یک «پل» می‌تواند برای انواع شبکه‌های مختلف با سرعت متفاوت، «کارت‌های خط» مجزا داشته باشد. در یک «پل» برخلاف هاب، هر خط «حوزه تصادم» خاص خود را دارد. «سوئیچها» شبیه به پلها هستند چرا که هر دوی آنها براساس آدرسهای درون فریم، آنها را مسیریابی و هدایت می‌کنند. در واقع بسیاری از افراد این دو واژه را به صورت معادل به کار می‌برند. تفاوت اصلی آنها در این است که یک سوئیچ شبیه به شکل ۴-۴۷-ج، برای وصل کردن کامپیوترهای منفرد به یکدیگر، کاربرد دارد. طبعاً وقتی ماشین میزبان A در شکل ۴-۴۷-ب می‌خواهد فریمی را برای ماشین B بفرستد، پل اگر چه فریم را دریافت می‌کند ولی آن را نادیده می‌گیرد. برخلاف آن در شکل ۴-۴۷-ج، سوئیچ باید بلافاصله فریم را از A به سمت B هدایت نماید چرا که هیچ راهی برای تحویل فریم به B [جز از طریق سوئیچ] نیست. از آنجایی که معمولاً هر یک از پورتهای یک سوئیچ به یک کامپیوتر منفرد وصل می‌شود لذا یک سوئیچ باید پورتهای بسیار بیشتری (در مقایسه با پل‌هایی که برای وصل تعداد کمی LAN طراحی شده‌اند) داشته باشد. در ضمن هر یک از «کارت‌های خط» بایستی فضای بافر کافی برای ذخیره فریم‌های دریافتی از هر یک از پورتهای در اختیار داشته باشند. از آنجایی که هر یک از پورتهای «حوزه تصادم» مجزا و متعلق به خود را دارند لذا هیچ فریمی در اثر تصادم از بین نخواهد رفت. با این وجود اگر فریم‌ها با نرخی بیش از ظرفیت سوئیچ، وارد گردند ممکن است فضای بافر پر شده و سوئیچ مجبور به حذف آنها شود.

برای آنکه این مشکل کمی کاهش یابد در سوئیچهای مدرن به محض آنکه فیلد آدرس مقصد از فریم وارد گردید، عمل هدایت و انتقال فریم شروع می‌شود، حتی قبل از آنکه مابقی فریم بطور کامل دریافت شده باشد. البته در صورتی که خط خروجی مورد نظر آزاد باشد. این سوئیچها روش «ذخیره و هدایت» (Store & Forward) را به کار نمی‌برند. [در روش «ذخیره و هدایت» ابتدا کل فریم دریافتی ذخیره شده و سپس عملیات هدایت آغاز می‌شود]. این نوع از سوئیچها که اغلب به نام Cut-Through Switches (سوئیچهای میانبر) مشهورند، کاملاً به صورت سخت‌افزاری پیاده سازی می‌شوند در حالیکه پلها عموماً دارای یک CPU واقعی بوده و عمل «ذخیره و هدایت» را توسط نرم‌افزار انجام می‌دهند. بهر حال چون تمام پلها و سوئیچهای مدرن دارای تراشه مدار مجتمع ویژه‌ای جهت هدایت و انتقال فریمها هستند لذا امروزه تفاوت‌های بین پل و سوئیچ بیشتر به مسائل بازاری بستگی دارد تا مسائل فنی.

تا اینجا تکرارکننده‌ها و هابها را بررسی کرده‌ایم که کاملاً شبیه به هم هستند و هم‌طور به پلها و سوئیچها پرداختیم که آنها نیز شباهت زیادی به یکدیگر دارند. حال به سوی «مسیریاب» (Router) حرکت خواهیم کرد که با تمام ابزارهای فوق تفاوت دارد. وقتی بسته‌ای به یک مسیریاب وارد می‌شود ابتدا سرآیند و فیلدهای انتهایی فریم حذف شده و سپس بسته جاسازی شده در درون فیلد حمل داده فریم (Payload) (که در شکل ۴-۴۶ به صورت خاکستری نشان داده شده)، تحویل نرم‌افزار مسیریابی می‌شود. این نرم‌افزار برای انتخاب خط خروجی از مشخصات واقع در سرآیند بسته استفاده می‌کند. در بسته‌های IP، سرآیند هر بسته شامل یک آدرس ۳۲ بیتی

(در IPv4) یا ۱۲۸ بیتی (در IPv6) است ولی آدرس ۴۸ بیتی 802 ندارد. نرم افزار مسیریابی نمی تواند آدرس فریمها را ببیند و حتی نمی تواند متوجه شود که بسته از طریق یک LAN دریافت شده یا از روی یک خط نقطه به نقطه. در فصل پنجم مسیریابها و فرآیند مسیریابی را تشریح خواهیم کرد.

در لایه بالاتر به «دروازه های انتقال» (Transport Gateway) بر می خوریم. این دروازه ها ارتباط دو کامپیوتر را که از پروتکل های اتصال گرای متفاوتی در لایه انتقال استفاده می کنند، برقرار می نماید. به عنوان مثال فرض کنید کامپیوتری که از پروتکل اتصال گرای TCP/IP استفاده کرده، می خواهد با کامپیوتری که از پروتکل اتصال گرای ATM استفاده می کند، محاوره نماید. یک دروازه انتقال می تواند بسته هایی که از طریق یک «اتصال» دریافت می شوند را پس از تغییرات لازم بر روی «اتصال» دیگر بفرستد.

در نهایت، «دروازه های کاربرد» (Application Gateway) قالب و محتوای داده ها را تشخیص می دهند و یک «پیام» را به پیامی دیگر ترجمه می کنند. بعنوان مثال یک «دروازه پست الکترونیکی» می تواند یک پیام اینترنتی [نامه الکترونیکی] را به پیام SMS برای گوشی های تلفن همراه ترجمه نماید.

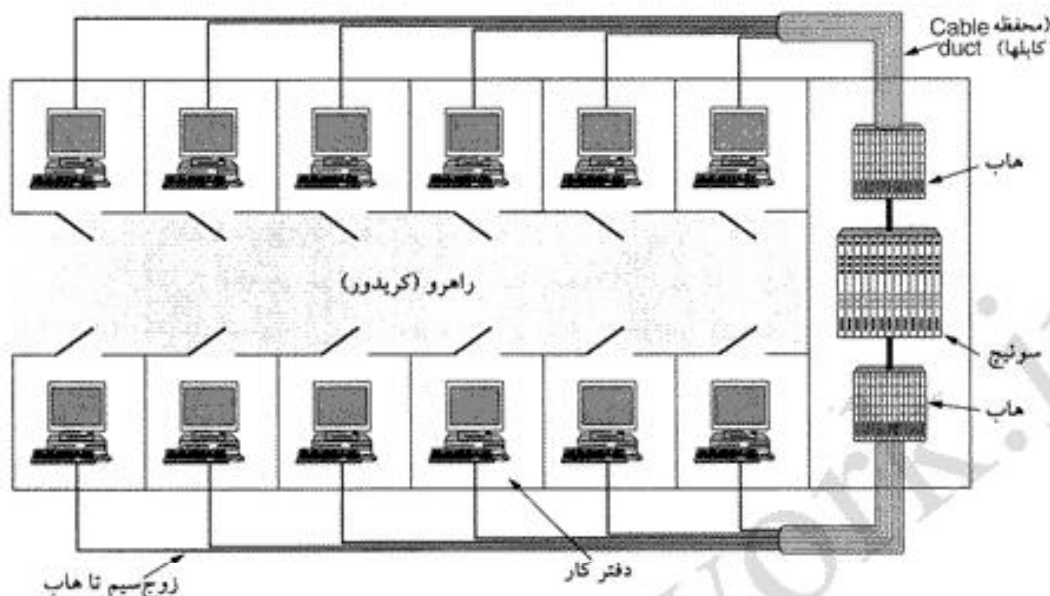
۶-۷-۴ شبکه های محلی مجازی (Virtual LANs)

در دوران اولیه به کارگیری شبکه های محلی، کابل های زرد رنگ ضخیم از طریق مجراهای مخصوص (duct)، بین دفاتر ساختمانها کشیده می شد و هر کامپیوتری که مسئولین امر تصمیم می گرفتند با وصل به این کابل، به شبکه ملحق می شد. اغلب، کابل های بی شماری وجود داشت که به یک «ستون فقرات مرکزی» (Central Backbone) یا به یک هاب مرکزی متصل می شد (شکل ۴-۳۹) و این موضوع که کدام کامپیوتر به کدام LAN متصل می شود چندان مهم نبود. تمام افرادی که در دفاتر مجاور هم بودند بر روی یک LAN مشابه قرار می گرفتند، فارغ از آنکه کارشان ارتباطی با هم داشت یا نه! یعنی «منطق برتر جغرافیایی». [بعبارت دیگر به جای آنکه یک منطق علمی، ساختار و اعضای یک LAN را تبیین کند جغرافیای محیط این ساختار و اعضاء را مشخص می کرد.]

با ابداع 10Base-T و هابها در اوایل ۱۹۹۰ همه چیز عوض شد. همه ساختمانها از نو سیم کشی شدند (البته با صرف هزینه قابل توجه) تا تمام کابل های زرد رنگ با قطر شیلنگ باغبانی برچیده شود و به جای آنها از هر دفتر یک زوج سیم به یک جعبه تقسیم مرکزی واقع در انتهای راهروها یا یک اتاق مرکزی کشیده شود. (شکل ۴-۴۸ را ببینید). اگر معاون رئیس یا مسئول سیم کشی بلند پرواز بود سیم های زوجی رده ۵ (Cat 5) نصب می کرد ولی اگر تنگ نظر بود از سیم کشی موجود خطوط تلفن، استفاده می نمود (که باز هم چند سال بعد و با پدیدار شدن اینترنت سریع باید عوض می شد!)

در اینترنت مبتنی بر هاب (و بعداً مبتنی بر سوئیچ)، اغلب این امکان وجود داشت که LAN ها به جای پیکربندی جغرافیایی به صورت منطقی پیکربندی شوند. اگر شرکتی به k شبکه LAN نیاز داشته باشد، k عدد هاب خریداری می کند. با دقت به آنکه کدام کابل رابط باید به کدام هاب متصل شود اعضای هر LAN را می توان به گونه ای انتخاب کرد که با ساختار سازمانی آنها مطابقت داشته باشد بدون آنکه جغرافیای محل، تاثیر چندانی در این انتخاب بگذارد. البته اگر دو نفر از یک دپارتمان مشابه از سازمان، در ساختمانهای متفاوتی مستقر بودند احتمالاً به دو هاب متفاوت و طبعاً به دو LAN متفاوت متصل می شدند. با این حال، شرایط جدید خیلی بهتر از زمانی است که اعضای یک LAN صرفاً بر اساس جغرافیای محل تعیین شوند.

آیا این مسئله که چه کسی بر روی کدام شبکه LAN قرار گرفته اهمیت دارد؟ مگر نه اینست که سرانجام در هر سازمان تمام LAN ها به هم متصل می شوند؟ کوتاه سخن آنکه جواب مثبت است و مسئله فوق اغلب مواقع اهمیت دارد. مسئولان شبکه بدلائل مختلف علاقمندند که کاربران را به نحوی بر روی LAN گروه بندی کنند که گروه ها به جای آنکه منعکس کننده نقشه فیزیکی ساختمانها باشند، جلوه ای از ساختار سازمانی باشند. یکی از



شکل ۴۸۴. یک ساختمان با سیم‌کشی مرکزی با بهره‌گیری از هاب و سوئیچ.

موارد و دلایل، «امنیت» است. هر کارت شبکه می‌تواند در حالت «بی‌قید» (Promiscuous) قرار بگیرد و تمام ترافیک جاری بر روی کانال را دریافت نماید. بسیاری از دپارتمانها همانند دپارتمان پژوهش، ثبت و حسابداری، اطلاعاتی را در اختیار دارند که نمی‌خواهند به بیرون از دپارتمان خودشان راه پیدا کند. در چنین شرایطی قرار دادن تمام افراد بر روی یک LAN واحد و جلوگیری از خروج ترافیک از آن LAN، معقول به نظر می‌رسد. مدیریت سازمان نمایی به شنیدن آنکه «چنین آرایشی ممکن نیست» ندارند مگر آنکه تمام افراد هر دپارتمان، در دفاتر همجوار جای داده شده باشند و در کار یکدیگر فضولی نکنند!!

مورد دیگر «میزان بار» است: برخی از LANها نسبت به بقیه، زیادتر مورد استفاده قرار می‌گیرند و ممکن است که تکنیک آنها مطلوبتر باشد. به عنوان مثال اگر گروه پژوهش در حال اجرای انواع آزمایشاتی باشند که گاه ترافیکی بیش از اندازه تولید و شبکه LAN را اشباع می‌کند، گروه حسابداری ممکن است علاقمند نباشد که برای کمک به آنان بخشی از ظرفیت [پهنای باند] خود را وقف آنان کند!

مورد سوم، «پخش فراگیر» (Broadcasting) است. اغلب LANها از ارسال فراگیر حمایت می‌کنند و بسیاری از پروتکل‌های لایه‌های بالاتر از این ویژگی در سطح گسترده‌ای استفاده می‌کنند. به عنوان مثال وقتی کاربری می‌خواهد بسته‌ای برای یک ماشین با آدرس IP معادل X بفرستد چگونه آدرس MAC آن ماشین را بدست می‌آورد تا در فریم مربوطه قرار بدهد؟ ما پاسخ این پرسش را در فصل پنجم بررسی خواهیم کرد ولی اگر بخواهیم بطور خلاصه جمع‌بندی نماییم پاسخ آنست که ماشین فریمی را به صورت پخش فراگیر بر روی شبکه قرار می‌دهد که حاوی این سوال است: «چه کسی صاحب آدرس IP معادل X است؟»... سپس منتظر پاسخ باقی می‌ماند. نمونه‌های کاربردی زیادی می‌توان یافت که متکی به پخش فراگیر هستند. هر چه LANهای بیشتری به یکدیگر متصل شوند تعداد فریمهای فراگیر که به هر ماشین وارد می‌شوند به صورت خطی و متناسب با تعداد ماشینها افزایش خواهد یافت.

یکی دیگر از مشکلات پخش فراگیر آنست که اگر زمانی یک کارت شبکه از عملکرد طبیعی خود خارج شده و شروع به تولید جریان بی‌پایانی از فریمهای فراگیر نماید تکلیف چیست. نتیجه به پا شدن این «طوفان فریمهای

فراگیر» آنست که (۱) کل ظرفیت LAN با این فریمها، اشغال و تپاه می شود. (۲) تمام ماشینهای واقع در LANهای متصل به هم، به واسطه صرف زمان جهت پردازش و سپس حذف این فریمهای فراگیر، زمین گیر می شوند. در بدو امر ممکن است به نظر برسد که «طوفان فریمهای فراگیر» را می توان با جدا کردن LANها توسط پل یا سوئیچ محدود کرد ولی اگر هدف نهایی «شفافیت» باشد سوئیچها و پلها موظف به هدایت فریمهای پخش فراگیر هستند. (به عبارتی یک ماشین باید بتواند به یک LAN متفاوت تغییر موقعیت بدهد و هیچکس متوجه این موضوع نشود و در محل جدید نیز قادر به پخش فراگیر باشد).

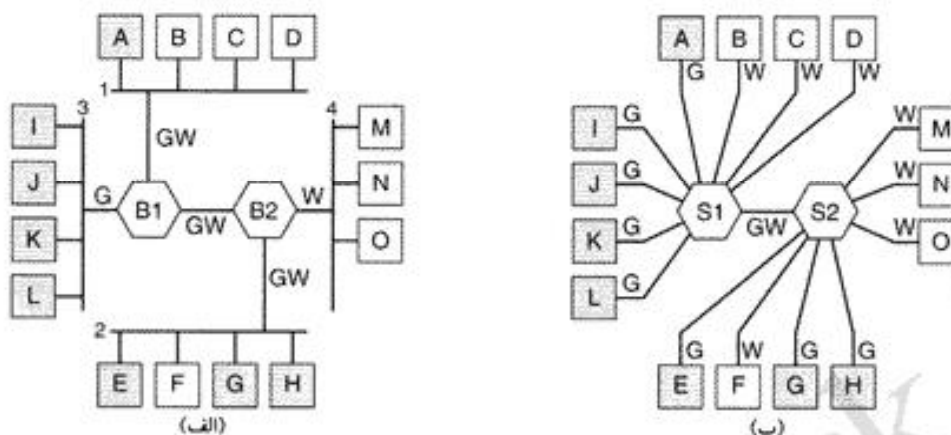
پس از آگاهی از آنکه چرا شرکتها ممکن است بخواهند LANهای متعددی با وسعت محدود داشته باشند، اجازه بدهید به مسئله اصلی یعنی جدا کردن توپولوژی منطقی از توپولوژی فیزیکی پردازیم. فرض کنید که یک کاربر در یک شرکت بدون تغییر محل دفتر کار خود از یک واحد اداری به واحد دیگر منتقل می گردد یا برعکس، دفتر کار خود را بدون جابجایی از واحد قبلی خود تغییر می دهد. در سیم کشی مبتنی بر هاب، تغییر موقعیت به یک LAN جدید مستلزم آنست که مسئول شبکه به سوی جعبه تقسیم (Wiring Closet) رفته و کابل رابط ماشین آن کاربر را از محل فعلی بیرون کشیده و آنرا به هاب جدید متصل کند.

در بسیاری از شرکتها تغییر و تحولات سازمانی یک امر عادی است و این مستلزم آنست که مسئول شبکه وقت بسیار زیادی را صرف جدا کردن کابلهای رابط و قرار دادن آن در محل جدید نماید. البته در برخی از حالات نیز چنین تغییراتی به هیچ وجه ممکن نیست چرا که مثلاً فاصله ماشین کاربر از هاب جدید بسیار دور است.

در پاسخ به نیاز کاربران به قابلیت انعطاف بیشتر، عرضه کنندگان محصولات شبکه کار بر روی طرحی را آغاز کرده اند که بر اساس آن بتوان سیم کشی ساختمانها را به صورت «نرم افزاری» تغییر داد (یعنی پیکربندی ماشینهای هر LAN فارغ از ساختار اتصال فیزیکی آنها ممکن باشد). نتیجه چنین نظریه ای «شبکه محلی مجازی» یا VLAN نامیده می شود و توسط کمیته IEEE استاندارد سازی شده است. اکنون بسیاری از سازمانها آرایشی مبتنی بر VLAN دارند. اجازه بدهید نگاهی به آن بیندازیم. برای کسب آگاهی بیشتر مراجع (Breyer and Riley, 1999, Seifert, 2000) مفیدند.

شبکه های VLAN، یکمک سوئیچهای خاص و سازگار با VLAN، پیاده سازی می شوند، هر چند ممکن است همانند شکل ۴-۴۸ دارای چند هاب جانبی و معمولی نیز باشند. برای پیکربندی شبکه مبتنی بر VLAN، مسئول شبکه تصمیم می گیرد که (۱) چند VLAN باید تعریف شود. (۲) چه کامپیوترهایی بر روی هر VLAN قرار می گیرند. (۳) هر یک از VLANها چگونه نامگذاری می شوند. اغلب، یک VLAN با رنگها نامگذاری می شود، (البته به صورت غیررسمی) زیرا بدین ترتیب امکان آنکه بتوان نقشه فیزیکی ماشینهای شبکه را به صورت رنگی چاپ کرد، وجود دارد. اعضای شبکه LAN قرمز با رنگ قرمز مشخص می شوند، اعضای شبکه LAN سبز با رنگ سبز و به همین ترتیب. با این روش نقشه منطقی و فیزیکی شبکه در یک نمای واحد دیده می شود.

به عنوان یک مثال، چهار LAN نشان داده شده در شکل ۴-۴۹ را مد نظر قرار بدهید که در آن هشت ماشین، متعلق به شبکه VLAN خاکستری (G) هستند، هفت تا متعلق به سفید (W). چهار LAN فیزیکی نیز توسط دو پل B1 و B2 بهم متصل شده اند. اگر در این شکل به جای ساختار یاس از هابهای مرکزی مبتنی بر زوج سیم استفاده می شد ممکن بود که چهار هاب نیز وجود داشته باشد (که در شکل نشان داده نشده است)، ولیکن از دیدگاه منطقی کابلهای چند اتصال و هاب مشابه به هم هستند. اگر می خواستیم شکل را با هاب ترسیم کنیم، گویانی خود را از دست می داد. امروزه واژه پل بیشتر اشاره به ساختاری شبیه به شکل ۴-۴۹-ب دارد که در آن چندین ماشین از طریق پورتهای آن به هم متصل شده اند در حالیکه امروزه «پل» و «سوئیچ» مفهوم معادلی دارند. در شکل



شکل ۴-۴۹. (الف) چهار LAN فیزیکی با استفاده از دو پل، دو VLAN خاکستری و سفید تشکیل داده‌اند. (ب) همان پانزده ماشین یکمک دو سوئیچ، دو VLAN تشکیل داده‌اند.

۴-۴۹-ب همان تعداد ماشین و همان تعداد VLAN [معادل با شکل ۴-۴۹-الف] فقط با استفاده از سوئیچ نشان داده شده است بگونه‌ای که بر روی هر پورت سوئیچ فقط یک ماشین وجود دارد.

برای آنکه VLAN بدرستی کار کند بایستی یک جدول پیکربندی درون هر پل یا سوئیچ تنظیم شود. این جدول مشخص‌کننده آنست که از طریق کدام یک از پورتها می‌توان به کدام VLAN دسترسی داشت. وقتی فریمی از روی یک VLAN مثلاً خاکستری به سوئیچ وارد شود صرفاً باید بر روی تمام پورتهایی که علامت G دارند منتقل شوند. همانند ترافیک معمولی تک‌بخشی (Unicast)، ارسال ترافیک چندبخشی (Multicast) و ارسال ترافیک فراگیر (Broadcast) نیز به همین نحو ممکن خواهد بود.

دقت کنید که ممکن است یک پورت برچسب رنگی چندین VLAN را داشته باشد. در شکل ۴-۴۹-الف چنین ساختاری را مشاهده می‌نماییم. فرض کنید ماشین A یک فریم «فراگیر» (Broadcast) را برای همه اعضای VLAN خود ارسال کند. پل B1 این فریم را دریافت کرده و متوجه می‌شود که توسط ماشینی بر روی VLAN خاکستری تولید شده است لذا آن فریم را بر روی تمام پورتهایی که علامت G دارند (به استثنای پورتی که فریم از روی آن دریافت شده) ارسال می‌نماید. از آنجایی که B1 فقط دو پورت دیگر با برچسب G دارد لذا فریم را بر روی هر دوی آنها ارسال می‌کند.

در B2 داستان به گونه دیگری است: در اینجا پل می‌داند که هیچ ماشینی خاکستری بر روی LAN 4 وجود ندارد لذا فریم بر روی آن هدایت نخواهد شد و فقط به LAN 2 ارسال می‌شود. اگر یکی از کاربران LAN 4 ملزم به تغییر واحد اداری خود شده و به VLAN خاکستری نقل مکان نماید باید جدول B2 بهنگام‌سازی شده و تنها پورت آن که با برچسب W مشخص شده به GW تغییر داده شود. اگر ماشین F به شبکه مجازی خاکستری نقل مکان کند، پورت متصل به LAN 2 باید از GW به G تغییر برچسب داده شود.

حال فرض کنید ماشینهای عضو LAN 2 و LAN 4 همگی به جمع خاکستری‌ها پیوندند، لذا نه تنها پورتهای پل B2 متصل به LAN 2 و LAN 4 هستند علامت G می‌گیرند بلکه پورت اتصال B1 به B2 نیز از GW به G تغییر علامت می‌دهد چراکه لازم نیست فریمهای سفیدی که از LAN 3 یا LAN 1 می‌رسند به B2 هدایت شوند. در شکل ۴-۴۹-ب همین وضعیت حاکم است و تمام پورتهایی که به یک ماشین واحد متصل هستند با یک برچسب تک‌رنگ مشخص می‌شوند چراکه هر ماشین تنها به یک VLAN متعلق است.

تاکنون فرض را بر آن گذاشته بودیم که پلها و سوئیچها به نحوی می‌دانند که رنگ یک فریم ورودی چیست.

آنها چگونه از این موضوع آگاه می شوند؟ برای این کار سه روش زیر کاربرد دارد:

۱. به هر پورت رنگ VLAN انتساب داده شود.
۲. به هر آدرس MAC یک رنگ VLAN منتسب گردد.
۳. به آدرسهای لایه ۲ یا آدرس IP ماشین یک رنگ VLAN منتسب شود.

در روش اول، به هر پورت یک برچسب رنگ داده می شود که این برچسب، VLAN مربوطه را مشخص می کند. با این حال این روش فقط زمانی کار خواهد کرد که تمام ماشینهای متصل به آن پورت، به یک VLAN مشابه متعلق باشند. در شکل ۴-۴۹ الف این ویژگی برای پورت بین پل B1 و LAN 3 صادق است در حالیکه برای پورت متصل به LAN 1 صادق نیست.

در روش دوم، پل یا سوئیچ دارای جدولی است که در آن فهرست آدرسهای ۴۸ بیتی تمام ماشینهای متصل به آن (یعنی آدرس MAC) و مشخصات VLAN هر ماشین، درج می شود. تحت این شرایط، می توان بر روی یک LAN فیزیکی (مثل LAN 1 در شکل ۴-۴۹ الف) چند VLAN مختلف تعریف کرد. وقتی فریمی دریافت می شود، آنچه که هر پل یا سوئیچ باید انجام بدهد آنست که آدرس MAC آن را استخراج و درون جدول به دنبال آن بگردد و ببیند که فریم از کدام VLAN می آید.

در روش سوم، پل یا سوئیچ موظف است تا محتوای فیلد حمل داده (Payload) از هر فریم را بررسی کرده و به عنوان مثال تمام ماشینهای مبتنی بر IP را در یک VLAN و ماشینهای مبتنی بر Apple Talk را در VLAN دیگر دسته بندی کند. در حالت اول می توان برای تشخیص هویت هر ماشین از آدرس IP آن استفاده کرد. این استراتژی زمانی بسیار سودمند خواهد بود که تعداد بی شماری از ماشینهای شبکه کامپیوترهای کیفی هستند و می توانند در هر یک از مکانهای متعدد و مجاز قرار بگیرند. از آنجایی که هر ایستگاه آدرس MAC خاص خود را دارد لذا با توقف ایستگاه در محل جدید، آدرس MAC آن نمی تواند چیزی در مورد آن VLAN که ایستگاه عضو آنست، مشخص کند.

تنها مشکل این روش آنست که یکی از قوانین اساسی در شبکه بندی را نقض می کند: «استقلال لایه ها» تشخیص آنکه چه چیزی در فیلد حمل داده از فریم قرار گرفته برعهده لایه پیوند داده ها نیست. این لایه نباید محتوای داده های درون هر فریم را بررسی کند و با براساس محتوای درون آن تصمیمی بگیرد. یکی از تبعات به کارگیری این روش آنست که هرگاه در پروتکل لایه ۳ تغییری ایجاد شود (مثلاً IPv4 به IPv6 ارتقاء پیدا کند) سوئیچ بلافاصله از کار خواهد افتاد. متأسفانه سوئیچهایی که بدین نحو کار می کنند به وفور در بازار عرضه شده اند. البته تعریف VLAN بر اساس آدرس IP، اشکالی در مسیریابی بسته های IP بوجود نخواهد آورد ولیکن تلفیق وظایف لایه ها سرآغاز بروز مشکلات پیش بینی نشده خواهد بود. (تقریباً کل فصل پنجم به مسیریابی IP اختصاص دارد). شاید عرضه کنندگان سوئیچ چنین استدلالی را به مسخره بگیرند و بگویند سوئیچهای آنه IPv4 و IPv6 را به رسمیت می شناسند و همه چیز به درستی پیش خواهد رفت ولیکن اگر زمانی IPv7 مطرح شد چه اتفاقی می افتد؟ شاید فروشنده بگوید که در آن زمان سوئیچهای جدید بخرید! آیا این کار خیلی شاق است؟

استاندارد IEEE 802.1Q

اندکی اندیشه بیشتر در خصوص VLAN ما را به این نتیجه می رساند که آنچه واقعاً اهمیت دارد آنست که هر فریم ارسال نام VLAN خود را با خود حمل کند نه آنکه VLAN ماشین فرستنده را سوئیچ بروشهای دیگری مشخص کند. اگر روشی برای مشخص کردن VLAN در سرآیند هر فریم وجود داشته باشد دیگر نیازی به بررسی داده های

درونی هر فریم نخواهد بود. برای شبکه‌های جدیدی مثل 802.11 یا 802.16، اضافه کردن فیلد VLAN به سرآیند هر فریم نسبتاً ساده است. در حقیقت فیلد Connection Identifier در 802.16 ذاتاً چیزی شبیه به VLAN Identifier است. ولی در مورد اترنت چه کاری می‌توان کرد؟ شبکه‌ای که رایجترین شبکه دنیاست و هیچ فیلد اضافی برای تعبیه VLAN Identifier در آن، تعریف نشده است.

کمیته IEEE 802 این مسئله را در دستور کار خود قرار داد و پس از مباحثات فراوان کاری غیرقابل تصور انجام داد و سرآیند اترنت را عوض کرد. قالب جدید فریم، در استاندارد IEEE 802.1Q تدوین و در سال ۱۹۹۸ منتشر شد. در قالب جدید، هر فریم دارای یک برچسب VLAN است (با نام VLAN Tag) که مختصراً آن را بررسی خواهیم کرد. متأسفانه تغییر در استاندارد مثل سرآیند اترنت که کاملاً جا افتاده و بطور رایج از آن استفاده می‌شود، چندان ساده نیست. در این خصوص ممکن است سوالات زیر به ذهن خطور کند:

۱. آیا باید چندین میلیون کارت اترنت موجود را دور انداخت؟
۲. اگر نه، چه کسی فیلدهای جدید را تولید نماید؟
۳. برای فریمهایی که اندازه حداکثر دارند چه اتفاقی می‌افتد؟ [چون نمی‌توان به فریمی که بر طول حداکثر آن محدودیت گذاشته شده، داده‌ای افزود.]

البته کمیته 802 از این مشکلات آگاه بود و می‌بایست راه حلی مناسب ارائه می‌کرد، کاری که نهایتاً انجام شد. نکته اصلی در راه حل ارائه شده آنست که فیلدهای VLAN فقط توسط پلها و سوئیچها مورد استفاده قرار می‌گیرد و ماشینهای کاربران بدان نیازی ندارند. بدین ترتیب در شکل ۴-۴۹ وقتی فریمها مستقیماً به سوی یک ایستگاه پایانی ارسال می‌شوند به این فیلدها نیازی نیست و فقط روی خطوطی بین پلها یا سوئیچها، بکار می‌آیند. بنابراین برای بکارگیری VLAN، این پلها یا سوئیچها هستند که باید VLAN را به رسمیت بشناسند؛ این موضوع نیز در پلها و سوئیچها از قبل پیش‌بینی شده و یک نیاز تلقی می‌شود. حال باید نیازمندیهای جدیدی را معرفی نمائیم که 802.1Q بدانها پاسخ داده است.

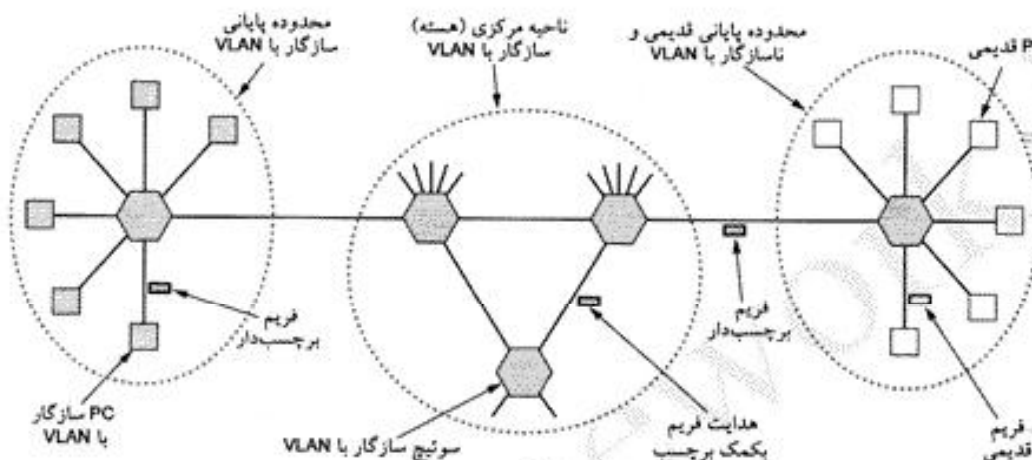
پاسخ به این سوال که «آیا باید تمام کارتهای اترنت موجود را دور انداخت؟» منفی است. به خاطر داشته باشید که کمیته 802.3 حتی نتوانست به افراد بقبولاند که فیلد Type را به فیلد Length تغییر بدهند. حال شما تصور کنید عکس‌العمل مردم در قبال اعلام دور انداختن کارتهای اترنت، چه می‌شد! البته وقتی کارتها جدید به بازار بیایند شاید بتوان امیدوار بود که سازگار با 802.1Q بوده و فیلدهای مربوط به VLAN در آنها تعبیه شده باشند.

به هر حال اگر کارت شبکه تولیدکننده یک فریم، فیلدهای VLAN را بدان اضافه نکند، چه کسی باید این کار را انجام بدهد؟ پاسخ این سوال آنست که اولین پل یا سوئیچ مبتنی بر VLAN که فریم را دریافت می‌کند، این فیلدها را به فریم افزوده و آخرین آنها در مسیر، این فیلدها را حذف می‌نماید. ولیکن یک سوئیچ یا پل از کجا می‌فهمد که یک فریم به کدام VLAN تعلق دارد؟ پاسخ آنست که اولین پل یا سوئیچ می‌تواند به هر یک از پورتهای خود یک شماره VLAN نسبت بدهد یا آنکه به آدرس MAC آن فریم نگاه کند و یا محتوای داده‌های درونی را بررسی کند (کار ممنوع!).

در خصوص کارتهای اترنت که سازگار با 802.1Q هستند، مشکلی وجود ندارد. انتظار قابل تحقق آنست که تمام کارتهای اترنت گیگابیت از همان ابتدا سازگار با 802.1Q باشند و با ارتقاء کارتهای قدیمی به اترنت گیگابیت، 802.1Q نیز به صورت خودکار جای خود را باز کند. برای حل این مشکل که فریمهای اترنت نباید از 1518 بایت بیشتر باشند در 802.1Q طول حداکثر به ۱۵۲۲ بایت افزایش یافته است.

در دوران گذار از اترنت فعلی به اترنت گیگابیت، در بسیاری از شبکه‌ها برخی از ماشینهای قدیمی (عموماً

اترنت کلاسیک و اترنت سریع) با VLAN سازگار نیستند، در حالیکه برخی دیگر (عموماً اترنت گیگابت) از آن پشتیبانی می‌کنند. چنین وضعیتی در شکل ۴-۵۰ به تصویر کشیده شده است و در آن نمادهای خاکستری سازگار با VLAN و نمادهای بی‌رنگ ناسازگار هستند. برای سادگی بحث، فرض را بر آن گذاشته‌ایم که تمام سوئیچها با VLAN سازگار هستند. اگر اینگونه نباشد اولین سوئیچ سازگار با VLAN بکمک آدرس MAC یا آدرس IP درون فریم، برچسب لازم را بدان خواهد افزود.



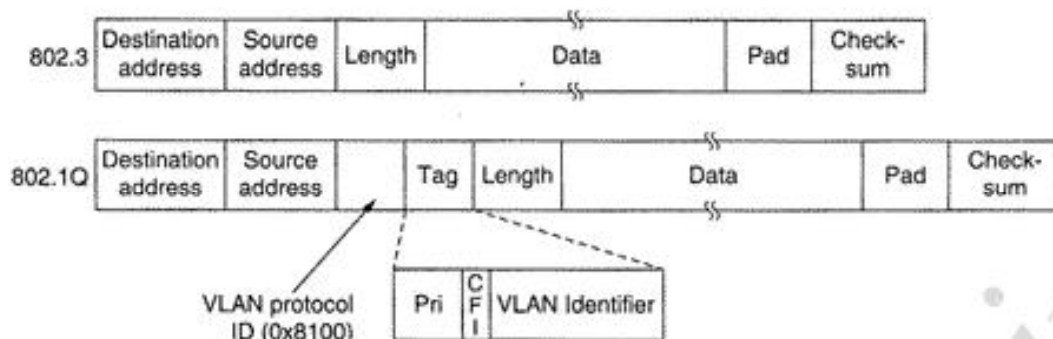
شکل ۴-۵۰. گذر از اترنت قدیمی به اترنت سازگار با VLAN. نمادهای خاکستری با VLAN سازگارند؛ نمادهای بی‌رنگ سازگار نیستند.

در این شکل کارتهای اترنت سازگار با VLAN، مستقیماً فریمهای برچسب‌دار (یعنی فریمهای 802.1Q) را تولید می‌نمایند و سوئیچهای بعدی از این برچسب استفاده خواهند کرد. برای عملیات هدایت فریمها (یعنی عمل سوئیچینگ) هر سوئیچ بایستی بداند که کدام VLAN از طریق کدام پورت در دسترس است. (طبق توضیحات قبلی) دانستن آنکه فریم جاری به VLAN مثلاً خاکستری تعلق دارد فایده‌چندانی نخواهد داشت مگر آنکه سوئیچ بداند کدام پورتهای به ماشینهای عضو VLAN خاکستری متصل هستند. بنابراین سوئیچ نیازمند به داشتن جدولی اندیس‌دار و مرتب شده بر حسب مشخصه VLAN است که تعیین می‌کند از کدامیک از پورتهای باید استفاده شود و آیا ماشینهای متصل بدین پورتهای سازگار با VLAN هستند یا قدیمیند.

وقتی یک PC قدیمی فریمی را برای یک سوئیچ سازگار با VLAN می‌فرستد، سوئیچ مربوطه با استفاده از دانش قبلی^۱ خود در خصوص VLAN متناظر با فرستنده فریم (مثلاً با استناد به شماره پورت فیزیکی، آدرس MAC یا آدرس IP) برچسب لازم را به فریم چسبانده و فریم جدیدی تولید می‌نماید. از این دیدگاه، دیگر قدیمی بودن ماشین فرستنده هیچ اهمیتی ندارد. به روش مشابه هرگاه نیاز باشد فریمی برچسب‌دار (مبتنی بر 802.1Q) تحویل ماشین قدیمی شود قبل از تحویل، ساختار فریم به قالب بدون برچسب تبدیل خواهد شد.

حال بیایید نگاهی به قالب فریم 802.1Q بیندازیم. قالب این نوع فریم در شکل ۴-۵۱ نشان داده شده است. تنها تغییر، اضافه شدن یک جفت فیلد ۲ بیتی (جمعاً ۴ بایت) به قالب قدیمی است. یکی از آنها فیلد مشخصه VLAN Protocol ID است که همیشه مقدار 0x8100 دارد. از آنجایی که این مقدار بیش از ۱۵۰۰ است تمام

۱. به خاطر داشته باشید که سوئیچهای مدرن سازگار با VLAN، بصورت نرم‌افزاری پیکربندی می‌شوند و بعضاً دارای سیستم عامل و فرامین مخصوص هستند. بنابراین در خصوص ماشینهای با کارت اترنت قدیمی تنظیمات VLAN، باید بصورت دستی انجام شود.



شکل ۵۱-۴. قالب فریم قدیمی اترنت 802.3 و فریم 802.1Q.

کارتهای اترنت آنرا به عنوان فیلد «نوع فریم» (Type) تفسیر می کنند و به فیلد طول داده Length تعبیر نخواهد شد. عکس العملی که کارتهای اترنت قدیمی در برخورد با چنین فریمهایی از خود نشان می دهند جای بحث دارد چراکه فرض بر آن است که چنین فریمهایی برای کارتهای قدیمی ارسال نمی شود. فیلد دو بایتی بعدی شامل سه زیرفیلد است که اصلی ترین آنها یعنی VLAN Identifier (شناسه VLAN)، ۱۲ بیت کم ارزش را به خود اختصاص داده است. این فیلد کل آن چیزی است بدان نیازمند بوده ایم: یعنی مشخص می کند که «این فریم به کدام VLAN تعلق دارد». فیلد ۳ بیتی Priority (اولویت) فعالیت کاری در خصوص VLAN بر عهده ندارد و عملاً بلااستفاده است. استدلال کمیته 802.1Q در تعریف این زیرفیلد آن بوده که چون تغییر در سرآیند اترنت فرآیند دشوار و زمانبری است، حالا که این کار خطیر میسر شده چرا چیزهای خوب دیگر نیز بدان اضافه نشود؟ این فیلد ۳ بیتی این امکان را فراهم آورده تا بتوان «ترافیک بی درنگ از نوع سخت و نرم» (Soft & Hard Realtime Traffic) [مثل صدا و تصویر یا نظائر آن] را از ترافیک غیرحساس به زمان [مثل ترافیک ارسال نامه های الکترونیکی] تشخیص داد و امکان ارائه «کیفیت خدمات» (QoS) در اترنت نیز مسیر شود. در آینده به این فیلد جهت ارسال صدا بر روی اترنت نیاز خواهد شد. (هر چند مشابه با همین فیلد در پروتکل IP ربع قرن قبل در نظر گرفته شده بود ولیکن هیچگاه از آن استفاده نشد!!)

آخرین زیرفیلد، بیت CFI (Canonical Format Indicator) است که در حقیقت می بایست بیت CEI (Corporate Ego Indicator) بمعنای مشخصه شخصی شرکت نامیده می شد. اساساً این بیت باید برای مشخص کردن آدرسهای Big Endian از آدرسهای Little Endian بکار می رفت ولی پس از بحث و جدل فراوان از آن صرف نظر شد.^۱ اکنون معنای این بیت آن است که درون فیلد حمل داده از فریم اترنت، یک فریم کامل از نوع 802.5 (فریم شبکه Token Ring) قرار دارد و انتظار می رود به یک شبکه محلی از نوع 802.5 دیگر تحویل شود چراکه اتصال آنها از طریق اترنت برقرار شده است. (به عبارت دیگر این فیلد مشخص می کند که شبکه اترنت باید به عنوان حامل میانی، یک فریم نوع 802.5 را به شبکه ای دیگر از همین نوع تحویل بدهد).^۲ البته این بیت هیچ کاری در مورد شبکه VLAN انجام نمی دهد ولیکن سیاستهای کمیته استانداردسازی IEEE متفاوت با سیاستهای عادی است!!!

با دقت در توضیحات فوق، وقتی یک فریم برچسب دار به یک سوئیچ سازگار با VLAN می رسد آن سوئیچ از

۱. معنای آدرس Big Endian آن است که بایتهای پرارزش آدرس در ابتدا و بایتهای کم ارزش در انتها قرار گرفته و به همان نحو ارسال می شوند. معنای آدرس Little Endian آن است که بایتهای کم ارزش در ابتدا و بایتهای پر ارزش آدرس بعد از آن ارسال

۲. یعنی نوعی از پدیده تونل زنی (Tunneling) در سطح لایه پیوند داده. سم

فیلد VLAN ID به عنوان اندیس (یا عبارتی کلید جستجو) در جدول خود استفاده کرده و به کمک آن پورتهایی که این فریم باید بر روی آنها فرستاده شود را مشخص می کند. ولی سوال اساسی اینجاست که این جدول از کجا بدست می آید؟ اگر این جدول باید به صورت دستی تنظیم شود به پله شماره صفر برگشته ایم یعنی «پیکربندی دستی پلها»؛ زیبایی پلهای شفاف در آن است که به محض وصل، فوراً به کار می افتند (یعنی Plug & Play است) و نیازی به پیکربندی دستی ندارند. از دست دادن این ویژگی بسیار ناگوار است! خوشبختانه پلهای سازگار با VLAN می توانند خود را بر اساس برچسب فریمهای ورودی به صورت خودکار پیکربندی کنند. مثلاً اگر فریمی با برچسب 4 از روی پورت شماره ۳ وارد شود، به روشنی مشخص است که ماشین متصل به پورت ۳ متعلق به VLAN 4 است. استاندارد 802.1Q روش ایجاد پویای این جداول را تشریح کرده ولی در اکثر جاهای این توضیحات خواننده را به بخشهایی خاص از الگوریتم پرلن ارجاع داده است؛ الگوریتم پرلن نیز در 802.1D استاندارد شده است.

قبل از خاتمه بحث مسیریابی در VLAN، اشاره به یک نکته پایانی خالی از لطف نیست. بسیاری از افراد در دنیای اینترنت و ترنت طرفدار ساختار شبکه های «بی اتصال» (Connectionless) هستند و در مقابل هر چیزی که زمینه نیاز به «اتصال» در لایه پیوند داده یا لایه شبکه را فراهم کند به شدت مقاومت می کنند. در VLAN چیزهایی معرفی شده که شبیه به مفهوم «اتصال» هستند. برای عملکرد صحیح VLAN هر فریم با خود یک مشخصه خاص و جدید دارد [یعنی همان VLAN ID] که از آن بعنوان اندیس جستجو در جدول درون سوئیچ، استفاده می شود. این دقیقاً همان اتفاقی است که در شبکه های اتصال گرا می افتد. در شبکه های بدون اتصال به تنها چیزی که برای مسیریابی و هدایت فریم نیاز است آدرس MAC می باشد و به هیچ نوع مشخصه اتصال یا امثال آن نیاز نیست. در فصل پنجم در خصوص مفاهیم اتصال گرایی (Connectionism) بیشتر خواهیم آموخت.

۸-۴ خلاصه

برخی از شبکه ها تنها دارای یک کانال انتقال مشترک هستند که همه ایستگاهها از آن، برای مبادله داده های خود استفاده می کنند. در این شبکه ها، مسئله اصلی در طراحی لایه پیوند داده، تسهیم و تخصیص کانال بین ایستگاه های رقیبی است که علاقمندند از آن استفاده نمایند. الگوریتمهای بی شماری برای مسئله تخصیص کانال ابداع شده است. خلاصه ای از مهمترین روشهای تخصیص کانال در جدول ۴-۵۲ فهرست شده است.

ساده ترین روش تخصیص، FDM و TDM است. این روشها زمانی مفیدند که تعداد ایستگاهها کم و ثابت و ترافیک آنها پیوسته باشد. هر دوی این روشها در محیطهایی با این ویژگی مثلاً برای تقسیم پهنای باند در خطوط اصلی تلفن (شاهراهها)، کاربرد دارند. وقتی تعداد ایستگاهها زیاد و متغیر یا ترافیک آنها نسبتاً انفجاری است روش های TDM و FDM گزینه های مناسبی نیستند. پروتکل ALOHA (چه نوع معمولی و چه نوع Slotted آن) بعنوان روشهای جایگزین TDM و FDM معرفی شدند. گونه های مختلف ALOHA نیز تشریح و تحلیل شده اند. برخی از این گونه ها امروزه در سیستمهای واقعی به کار می روند.

وقتی بتوان حالت فعلی کانال را شنود (احساس) کرد، ایستگاهها قادر خواهند بود مادامیکه ایستگاه دیگر در حال ارسال است از ارسال فریم خود اجتناب کنند. این تکنیک یعنی «شنود سیگنال حامل» منتج به ابداع پروتکلهای متنوعی شد که می تواند در شبکه های LAN و MAN به کار گرفته شود.

رده ای از پروتکلها قادرند مسئله رقابت بر سر تصرف کانال را متفی کنند یا حداقل این رقابت را تا حد قابل توجهی کاهش بدهند. روش «شمارش دودویی معکوس» (Binary Countdown) رقابت را بطور کامل حذف می کند. پروتکل «پیمایش درخت وفقی» (Adaptive Tree Walk) با تقسیم بندی پویای ایستگاهها به گروه های از

روش	توصیف عملکرد
FDM	به هر ایستگاه یک باند فرکانسی تخصیص می‌دهد.
WDM	الگوی پویای FDM برای فیبر نوری
TDM	به هر ایستگاه یک برش زمان تخصیص می‌دهد.
Pure ALOHA	ارسال ناهماهنگ در هر لحظه دلخواه
Slotted ALOHA	ارسال تصادفی در برشهای زمانی مشخص
1-persistent CSMA	استاندارد دسترسی چندگانه مبتنی بر شنود کانال
Nonpersistent CSMA	تاخیر تصادفی در هنگامی که کانال اشغال تشخیص داده می‌شود.
P-persistent CSMA	همان CSMA است با این تفاوت که احتمال اصرار در ارسال فریم P است.
CSMA/CD	همان CSMA است با این تفاوت که بمحض تشخیص تصادم ادامه ارسال را قطع می‌کند.
Bit map	زمان‌بندی به نوبت و چرخشی (Round-Robin) یکمک یک الگوی بیتی (Bitmap)
Binary countdown	ایستگاه با بزرگترین شماره حق ارسال در نوبت بعدی را دارد.
Tree walk	رقابت محدود با فعالیتهای انتخابی (تعریف گروه‌های رقیب در ساختار درختی)
MACA, MACAW	پروتکل‌های شبکه‌های محلی بی‌سیم
Ethernet	همان CSMA/CD با الگوریتم عقبگرد نمایی
FHSS	جهش‌های فرکانسی در طیف گسترده (Frequency Hopping Spread Spectrum)
DSSS	Direct Sequence Spread Spectrum
CSMA/CA	استاندارد دسترسی چندگانه مبتنی بر شنود کانال و اجتناب از تصادم

شکل ۴-۵۲. روشها و سیستمهای تخصیص یک کانال مشترک.

هم جدا، مشکل رقابت را بطور جدی کاهش می‌دهد. تقسیم‌بندی گروه‌ها حتی الامکان به نحوی است که فقط گروههایی که در آنها فقط یک ایستگاه فریمی برای ارسال آماده دارد مجاز به ارسال هستند.

شبکه‌های محلی بی‌سیم مشکلات و راه حل‌های خاص خود را دارند. عمده‌ترین مشکل توسط ایستگاه‌های «پنهان» ایجاد می‌شود و بدین دلیل CSMA جوابگوی چنین شبکه‌هایی نخواهد بود. رده‌ای از این راه‌حلها که در طبقه MACA و MACAW دسته‌بندی شده‌اند سعی دارند ایستگاه‌ها را وادار به ارسال در پیرامون ماشین مقصد نمایند تا بدین نحو CSMA عملکرد بهتری داشته باشد. روش‌های مبتنی بر طیف گسترده از نوع Frequency Hopping و Direct Sequence نیز در این شبکه‌ها به کار گرفته شده است. IEEE 802.11 روش CSMA و MACAW را تلفیق کرد و روش CSMA/CA را معرفی نمود.

اترنت رایجترین نوع شبکه‌های محلی است که برای تخصیص کانال از روش CSMA/CD بهره می‌گیرد. نسخه قدیمی آن از یک کابل واحد که بین تمام ماشینها کشیده می‌شد، استفاده می‌کردند، در حالیکه اکنون استفاده از هاب، سوئیچ و کابل‌های زوجی بسیار رایج است. سرعت آن از ۱۰ مگابیت بر ثانیه تا یک گیگابیت بر ثانیه افزایش یافته و هنوز هم در حال افزایش است.

امروز شبکه‌های محلی بی‌سیم نیز در حال رواج هستند که در این بین استفاده از 802.11 وسیعتر است. لایه فیزیکی در این شبکه، امکان پنج نوع انتقال و مدولاسیون متفاوت را فراهم آورده است که شامل مادون قرمز، گونه‌های متفاوت مبتنی بر طیف گسترده و «سیستم FDM چندکاناله» است. این شبکه می‌تواند یک ایستگاه ثابت در هر سلول داشته باشد ولیکن قادر است بدون ایستگاه ثابت نیز کار کند. این پروتکل گونه‌ای از MACAW و متکی بر شنود مجازی سیگنال حامل است.

شبکه‌های بین شهری بی‌سیم (Wireless MAN) در حال پیدایش و رواج هستند. این گونه شبکه‌ها

سیستمهای پاند گسترده ای هستند که از امواج رادیویی بهره می گیرند تا جایگزین ارتباطات تلفنی شوند. در آنها از تکنیک های مدولاسیون پاند باریک استفاده شده است. کیفیت خدمات نیز از موارد مهمی است که استاندارد 802.16 چهار رده مختلف از این گونه خدمات را تعریف کرده است: ارسال با نرخ ثابت، دو روش ارسال با نرخ متغیر و روش ارسال مبتنی بر بهترین تلاش (Best Effort).

سیستم بلوتوث نیز بی سیم است ولیکن هدف آن وصل ابزارهای رومیزی است؛ مثلاً برای اتصال گوشیه و ابزارهای جانبی کامپیوترهای شخصی بدون نیاز به سیم کاربرد دارد. همچنین به منظور وصل ابزارهایی مثل دورنگار (فکس) و تلفن های همراه به کار می آید. بلوتوث شبیه به 802.11 از تکنیک طیف گسترده مبتنی بر پرس فرکانس بهره گرفته و در باند ISM [یعنی 2.4 Ghz] کار می کند. به خاطر بالا بودن سطح نویز در بسیاری از محیطها و نیاز به فعل و انفعال بسی درنگ، در پروتکل های مختلف آن از روش تصحیح خطای مستقیم (Forward Error Correction) بهره گرفته شده است.

بدلیل وجود انواع متفاوت LAN به روشی جهت اتصال آنها به یکدیگر نیاز است. از پل و سوییچ به همین منظور استفاده می شود. در پلهای نوع Plug & Play از «الگوریتم درخت پوشا» (Spanning Tree) استفاده شده است. پیشرفت جدید در دنیای شبکه ها، VLAN است که بکمک آن توپولوژی منطقی LANها از توپولوژی فیزیکی آن جدا می شود. قالب جدیدی برای فریمهای اترنت معرفی شده است تا راحتتر بتوان VLAN را در سازمان ها پیاده سازی کرد.

مسائل

۱. برای حل این مسئله از رابطه ای که در همین فصل آمده بهره بگیرید ولیکن در ابتدا آن را بیان نمائید. بطور تصادفی فریمهایی برای انتقال بر روی کانالی با نرخ ارسال 100 Mbps تولید می شوند. اگر در زمان تولید یک فریم کانال اشغال بود تا رسیدن نوبت به آن در صف منتظر خواهد ماند. طول فریمها دارای تابع توزیع نمائی با میانگین 10000 bits/Frame است. برای هر یک از نرخهای تولید فریم که در زیر مشخص شده، تاخیری را که هر فریم (با طول متوسط) با آن مواجه می شود (شامل زمان انتظار صف و زمان انتقال) حساب نمائید.

الف) ۹۰ فریم در ثانیه

ب) ۹۰۰ فریم در ثانیه

ج) ۹۰۰۰ فریم در ثانیه

۲. یک گروه N تایی از ایستگاه ها دارای کانالی مشترک با نرخ 56kbps هستند و روش تخصیص کانال نیز Pure ALOHA است. بطور متوسط هر ایستگاه در هر صد ثانیه یک فریم هزار بیتی تولید می کند (حتی اگر فریم قبلی هنوز ارسال نشده باشد چرا که ایستگاه ها می توانند فریمهای خروجی را در بافر خود نگاه دارند). مقدار حداکثر N چقدر می تواند باشد؟

۳. تاخیر Pure ALOHA را در مقایسه با روش Slotted ALOHA و در شرایط بار پائین مد نظر قرار بدهید. تاخیر کدامیک کمتر است؟ پاسخ خود را تشریح کنید.

۴. ده هزار ایستگاه رزرو بلیط هواپیما، برای استفاده از یک کانال واحد، بروش Slotted ALOHA با هم رقابت می کنند. هر ایستگاه بطور متوسط ۱۸ تقاضا در هر ساعت خواهد داشت. برشهای زمانی (Time Slot) ۱۲۵ میکروثانیه ای هستند. مقدار تقریبی بار کل کانال چقدر است؟

۵. جمع کلیری از کاربران سیستم ALOHA در هر ثانیه ۵۰ تقاضا تولد می نمایند (که این مقدار شامل تعداد

- تقاضاهای جدید و تقاضاهای ارسال مجدد فریمهایی است که قبلاً در اثر تصادم خراب شده‌اند). زمان به برشهای متساوی ۴۰ میلی ثانیه‌ای تقسیم شده است:
- الف) احتمال موفقیت ارسال در همان دفعه اول چقدر است؟
- ب) احتمال بروز دقیقاً k تصادم پیاپی و سپس یک موفقیت چقدر است؟
- ج) به طور متوسط برای ارسال یک فریم چند بار تلاش لازم است؟
۶. اندازه‌گیری بار کانال Slotted ALOHA با تعداد نامحدود کاربر، نشان می‌دهد که ده درصد از برشهای زمانی بلااستفاده مانده‌اند:
- الف) بار کانال یعنی G چقدر است؟
- ب) بازده مفید کانال (Throughput) چقدر است؟
- ج) آیا بار کانال بیش از اندازه بالاست یا کمتر از حد متعادل است؟
۷. در یک سیستم Slotted ALOHA با کاربران نامحدود، متوسط تعداد برش زمانی که هر ایستگاه بین تصادم و ارسال مجدد صبر می‌کند، ۴ است. منحنی «تاخیر» را بر حسب «بازده مفید کانال» (Delay Versus throughput) در این سیستم ترسیم نمایید.
۸. یک ایستگاه مثل S را در شبکه‌ای با پروتکل‌های زیر در نظر بگیرید. این ایستگاه قبل از ارسال فریمش بر روی کانال، (در بدترین حالت) چه مدت زمانی باید انتظار بکشد:
- الف) پروتکل Basic Bit Map
- ب) پروتکل‌های Mok و Wark با جایگشت مجازی شماره ایستگاه‌ها
۹. یک LAN از پروتکل «شمارش دودویی معکوس» نسخه Mok و Wark، بهره می‌گیرد. در یک لحظه خاص، ده ایستگاه دارای شماره‌های مجازی ۱، ۲، ۳، ۴، ۵، ۶، ۷، ۸، ۹ و ۱۰ هستند. سه ایستگاه ارسال‌کننده بعدی به ترتیب عبارتند از: ۴، ۳ و ۹. پس از آنکه این ایستگاه‌ها ارسال خود را به اتمام رساندند شماره مجازی ایستگاه‌ها چند خواهد بود؟ [ترتیب شماره‌ها را از راست به چپ در نظر بگیرید.]
۱۰. شانزده ایستگاه که از ۱ تا ۱۶ شماره‌گذاری شده‌اند برای استفاده از یک کانال اشتراکی فروش «پیمایش درخت وقفی» رقابت می‌کنند. اگر تمام ایستگاه‌هایی که آدرس آنها اعداد اول است بطور ناگهانی آماده ارسال شوند برای خاتمه مراحل رقابت چند «برش بیتی» (Bit Slots) نیاز خواهد بود؟
۱۱. مجموعه‌ای از ۲۰ ایستگاه برای دسترسی به یک کابل اشتراکی از روش «پیمایش درخت وقفی» بهره گرفته‌اند. در یک زمان خاص دو ایستگاه آماده ارسال می‌شوند. اگر $1 < 2^n$ باشد مقدار حداقل، حداکثر و میانگین تعداد برشهای زمانی برای پیمایش این درخت [و خاتمه رقابت] چقدر است؟
۱۲. در شبکه محلی بی‌سیم که مطالعه کردیم به جای استفاده از پروتکل CSMA/CD از پروتکلی نظیر MACA استفاده شده بود. تحت چه شرایطی (در صورت امکان) استفاده از روش CSMA/CD میسر خواهد بود؟
۱۳. چه ویژگی‌هایی از پروتکل‌های دسترسی به کانال در GSM و WDMA مشترک هستند؟ GSM را در فصل ۲ مرور نمایید.
۱۴. شش ایستگاه A تا F با بکارگیری پروتکل MACA با یکدیگر در ارتباط و تبادل داده هستند. آیا امکان انتقال همزمان اطلاعات در این شبکه وجود دارد؟ (پاسخ خود را شرح بدهید.)
۱۵. در هر طبقه از یک ساختمان هفت طبقه، ۱۵ دفتر کار همجوار وجود دارد. در هر دفتر یک پریز دیواری (Wall Socket) برای وصل یک پایانه در جلوی آن تعبیه شده است و بدین ترتیب پریزها یک مستطیل در چهارگوش یک صفحه قائم تشکیل داده‌اند که فاصله بین پریزها به صورت عمودی و افقی چهار متر است.

فرض کنید امکان کابل کشی مستقیم بین هر جفت پریز، چه عمودی، افقی یا مورب وجود داشته باشد. برای اتصال این پریزها به هم، به چند متر کابل نیاز است اگر:

الف) پیکربندی مبتنی بر «ستاره» و با یک مسیریاب در وسط، مدنظر باشد.

ب) شبکه محلی 802.3 مدنظر باشد.

۱۶. نرخ تغییرات سیگنال (یعنی Baud Rate) در استاندارد ده‌مگابیتی اترنت چقدر است؟
۱۷. سیگنال حاصل از کدینگ «منچستر» را برای رشته بیت 0001110101 نشان بدهید.
۱۸. سیگنال حاصل از کدینگ «منچستر تفاضلی» را برای رشته بیت مسئله بالا ترسیم نمایید. فرض کنید خط در هنگام شروع به ارسال در حالت LOW قرار داشته باشد.
۱۹. در یک شبکه محلی 10Mbps مبتنی بر CSMA/CD (البته نه 801.3) با طول یک کیلومتر، سرعت انتشار سیگنال ۲۰۰ متر بر میکروثانیه است. طول فریمهای داده ۲۵۶ بیت شامل مجموعاً ۳۲ بیت سرآیند، کد کشف خطا و سربارهای دیگر است. اولین برش بیتی (Bit Slot) پس از ارسال موفق، برای گیرنده فریم در نظر گرفته شده تا بتواند برای ارسال پیام ACK (تصدیق دریافت فریم) کانال را در اختیار بگیرد. نرخ ارسال مفید (بدون در نظر گرفتن سربار و با فرض عدم وجود تصادم) چقدر است؟
۲۰. دو ایستگاه CSMA/CD تلاش می‌کند تا یک فایل بزرگ (شامل چندین فریم) را ارسال نمایند. پس از ارسال هر فریم، آنها برای دسترسی به کانال طبق الگوریتم «عقب‌گرد نمایی» با یکدیگر رقابت می‌کنند. احتمال اینکه هر رقابت در k دور به اتمام برسد و همچنین تعداد متوسط دورها (rounds) در هر رقابت چقدر است؟
۲۱. ساختمانی را در نظر بگیرید که در آن شبکه‌ای CSMA/CD با سرعت 1Gbps و یک کیلومتر کابل، بدون هیچ تکرارکننده (Repeater) کار می‌کند. سرعت انتشار سیگنال روی کابل ۲۰۰۰۰۰ کیلومتر بر ثانیه است. مقدار حداقل طول هر فریم چقدر است؟
۲۲. طول یک بسته IP جهت ارسال بر روی اترنت، ۶۰ بایت (شامل کل سرآیندها) است. اگر از LLC استفاده نشده باشد آیا در فریم اترنت به عمل اضافه کردن داده‌های زائد (Padding) نیاز است و اگر نیاز است چند بایت؟
۲۳. طول فریمهای اترنت باید حداقل ۶۴ بایت باشد تا این اطمینان حاصل شود که در صورت تصادم در انتهای کابل، فرستنده کماکان در حال ارسال است. [تا بتواند قبل از خاتمه فریم تصادم را کشف کند]. حداقل طول فریمها در «اترنت سریع»، ۶۴ بایت است در حالی که ارسال بیتها بر روی خروجی ۱۰ برابر سریعتر شده است. چگونه ممکن بوده که بتوان طول حداقل فریمها را در همان ۶۴ بایت نگه داشت؟
۲۴. در برخی از کتابها طول حداکثر هر فریم اترنت ۱۵۱۸ بایت ذکر شده است. آیا اشتباهی در کار است؟ پاسخ خود را تشریح نمایید.
۲۵. در تشریح مشخصات 1000Base-SX بیان شده که سیگنال ساعت در فرکانس ۱۲۵۰ MHz کار می‌کند، در حالیکه در اترنت گیگابیت فرض بر آن است که نرخ تحویل داده‌ها 1Gbps است. آیا این سرعت بالاتر، برای ایجاد حاشیه اطمینان بیشتر بوده است؟ اگر نه، چه نکته‌ای در کار است؟
۲۶. اترنت گیگابیت از عهده ارسال یا دریافت چند فریم در هر ثانیه برمی‌آید؟ به دقت بیندیشید و تمام حالات را در نظر بگیرید. راهنمایی: واقعیاتی که در شبکه اترنت گیگابیتی وجود دارد را مدّ نظر قرار بدهید.
۲۷. دو شبکه را نام ببرید که اجازه می‌دهند فریمها بطور پشت سرهم و در یک بار ارسال انتقال یابند. به چه دلیل چنین خصوصیاتی ارزشمند است؟

۲۸. در شکل ۴-۲۷ چهار ایستگاه A، B، C و D نشان داده شده‌اند. به نظر شما کدام یک از ایستگاه‌ها به A نزدیکتر هستند؟
۲۹. فرض کنید که در یک شبکه محلی بی‌سیم 802.11b [پس از در اختیار گرفتن کانال] تعدادی فریم پشت سر هم به طول ۶۴ بایت بر روی کانال با نرخ خطای 10^{-7} منتقل گردد. به طور متوسط چند فریم در هر ثانیه در اثر خطا آسیب خواهد دید؟
۳۰. یک شبکه 802.16 دارای کانالی با عرض باند 20MHz است. در هر ثانیه چند بیت برای یک ایستگاه قابل ارسال خواهد بود؟
۳۱. استاندارد IEEE 802.16 از چهار رده خدمات پشتیبانی می‌کند. کدامیک از این رده‌های خدمات، بهترین انتخاب برای ارسال تصاویر ویدیویی فشرده‌نشده می‌باشد؟
۳۲. دو دلیل بیاورید که چرا برخی شبکه‌ها ممکن است از کدهای تصحیح خطا به جای فرآیند کشف خطا و ارسال مجدد استفاده نمایند.
۳۳. در شکل ۴-۳۵ می‌بینیم که یک ابزار مبتنی بر بلوتوث می‌تواند به طور همزمان درون دو پیکونت قرار داشته باشد. آیا دلیلی وجود دارد که یک ابزار نتواند بطور همزمان نقش گره اصلی (Master) را در هر دو پیکونت ایفا کند؟
۳۴. شکل ۴-۲۵، چندین پروتکل لایه فیزیکی را نشان می‌دهد. کدام یک از آنها به پروتکل لایه فیزیکی در بلوتوث نزدیکتر است؟ کدام یک از آنها بیشترین اختلاف را دارد؟
۳۵. بلوتوث از دو نوع لینک بین گره اصلی (Master) و گره‌های پیرو (Slave) پشتیبانی می‌کند. این دو کدام است و هر کدام برای چه کاری در نظر گرفته شده‌اند؟
۳۶. فریمهای Beacon در گونه‌ای از شبکه 802.11 که مبتنی بر روش «طیف گسترده با پرش فرکانسی» (Frequency Hopping Spread Spectrum) است زمانی به نام Dwell time را مشخص می‌کند. آیا به نظر شما در فریمهای مشابه Beacon در بلوتوث نیز زمانی به نام dwell time وجود دارد؟ پاسخ خود را شرح دهید.
۳۷. به شبکه‌های LAN متصل به هم که در شکل ۴-۴۱ نشان داده شده دقت کنید. فرض کنید ماشینهای میزبان a و b بر روی LAN 1 و c بر روی LAN 2 و d بر روی LAN 8 واقع هستند. در ابتدا جداول Hash پلها خالی است و «درخت پوشای» نشان داده شده در شکل ۴-۴۱-ب به کار گرفته شده است. نشان دهید که جداول پلهای مختلف در اثر هر یک از رخدادهای زیر که به ترتیب حروف الفبا اتفاق می‌افتند چگونه تغییر می‌کند.
- الف) برای d ارسال می‌کند.
 - ب) c برای a فریمی می‌فرستد.
 - ج) d برای c فریمی می‌فرستد.
 - د) b به LAN 6 نقل مکان می‌کند.
 - ه) d برای a فریمی می‌فرستد.
۳۸. یکی از تبعات استفاده از «درخت پوشا» برای هدایت فریمها در یک LAN گسترش یافته آن است که برخی از پلها در فرآیند هدایت فریمها نقشی ایفاء نخواهند کرد. در شکل ۴-۴۴ سه تا از اینگونه پلها را مشخص کنید. آیا دلیلی دارد که چنین پلهایی را در شبکه داشته باشیم، حتی وقتی در هدایت فریمها نقشی ایفاء نمی‌کنند؟
۳۹. تصور کنید که یک سوئیچ دارای چند «کارت خط» (Line Card) است و هر یک از کارتها چهار خط

- ورودی دارند. فرض کنید که بطور متناوب فریمهایی که به یکی از خطوط کارت وارد می شوند از خط دیگری بر روی همان کارت خارج می گردند؛ طراح سوئیچ چه راهکاری در مواجهه با این مسئله پیش رو دارد؟
۴۰. یک سوئیچ که برای به کارگیری در اترنت سریع طراحی شده، دارای یک Backplane است که می تواند 10Gbps را منتقل کند. در سنگین ترین بار این سوئیچ قادر به انتقال چند فریم در هر ثانیه است؟
۴۱. به شبکه شکل ۴-۴۹-الف دقت کنید. اگر ماشین L به ناگاه «سفید» شود [یعنی به VLAN سفید بپیوندد] آیا تغییری در برچسبها لازم است؟ اگر این چنین است چگونه؟
۴۲. به طور مختصر تفاوت بین سوئیچهای نوع «ذخیره و هدایت» (Store & Forward) و سوئیچهای Cut Through را توضیح بدهید.
۴۳. سوئیچهای نوع «ذخیره و هدایت» در مقایسه با سوئیچهای Cut Through از دیدگاه آسیب رسیدن به فریمها برتر هستند. توضیح بدهید که این برتری از کجا ناشی می شود؟
۴۴. برای آنکه شبکه های VLAN شروع به کار کنند باید جداول پلها و سوئیچها تنظیم و پیکربندی شوند. اگر در شکل ۴-۴۹-الف به جای استفاده از شبکه مبتنی بر کابل چنداتصال، در VLAN از هاب استفاده شود آیا باز هم به تنظیم این جداول نیاز است؟ آیا هابها نیز نیاز به پیکربندی جدول دارند؟ چرا بله یا چرا نه؟
۴۵. در شکل ۴-۵۰، شبکه موجود و قدیمی سمت راست به یک سوئیچ سازگار با VLAN متصل شده است؛ آیا امکان دارد در اینجا نیز از یک سوئیچ قدیمی استفاده کرد؟ اگر جواب منفی است چرا؟
۴۶. برنامه ای بنویسید تا رفتار پروتکل CSMA/CD را در شبکه اترنت (در شرایطی که پس از ارسال یک فریم، N ایستگاه آماده ارسال هستند) شبیه سازی کند. برنامه شما باید زمانهایی را که هر ایستگاه موفق می شود ارسال فریم خود را آغاز کند گزارش نماید. فرض کنید که به ازای هر برش ۵۱/۲ میکروثانیه ای یک تیک ساعت اتفاق می افتد و کشف تصادم و ارسال سیگنال نویز (دنباله Jam پس از کشف تصادم) جمعاً ۵۱/۲ میکروثانیه طول می کشد. فریمها می توانند بیشترین طول مجاز را داشته باشند.

لایه شبکه

وظیفه لایه شبکه آن است که بسته های داده را به هر طریق از مبدا به مقصد برساند. هر بسته برای رسیدن به مقصد ممکن است از چندین مسیریاب (Router) در میانه راه گذر کند. این عملکرد به وضوح با عملکرد لایه پیوند داده (که هدف آن انتقال فریمها از انتهای یک سیم به نقطه دیگر آن است) تفاوت دارد. بنابراین لایه شبکه تحتانی ترین لایه ای است که با انتقال «انتها به انتها» (End-To-End) سرو کار دارد.

لایه شبکه برای نیل به اهداف خود، موظف است از توپولوژی «زیر شبکه ارتباطی»^۱ (یا به عبارتی مجموعه تمام مسیریابها) اطلاع داشته باشد و با استفاده از این دانش مسیری مناسب را برگزیند. همچنین باید مراقب باشد تا از تحمیل بار اضافی بر روی برخی از خطوط ارتباطی و مسیریابها (در حالی که برخی دیگر آزاد هستند) اجتناب نماید. نهایتاً وقتی که مبدا و مقصد یک بسته در دو شبکه با پروتکل های متفاوت و ناسازگار قرار گرفته اند مشکلات جدیدی بروز می کند که حل آنها بر عهده لایه شبکه گذاشته شده است. در این فصل به مطالعه و تشریح برخی از این موارد خواهیم پرداخت. تمرکز اصلی ما بر اینترنت و لایه شبکه آن یعنی IP است، هر چند شبکه های بی سیم را نیز خاطرنشان خواهیم نمود.

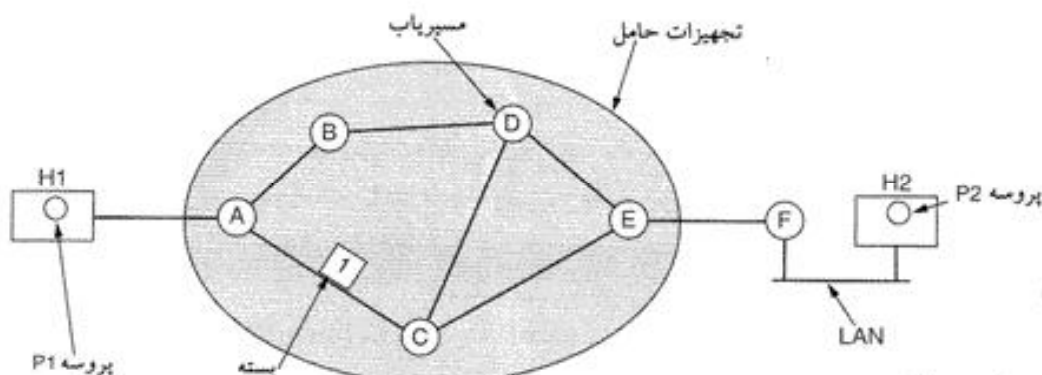
۱-۵ مسائل طراحی لایه شبکه

در بخشهای آتی مقدمه ای خواهیم داشت بر مسائلی که طراحان لایه شبکه با آنها دست به گریبان خواهند بود. این مسائل شامل خدماتی است که به لایه انتقال ارائه می گردد، یا به طراحی داخلی «زیر شبکه» (Subnet) مربوط می شود.

۱-۱-۵ هدایت (سوئیچینگ) بسته به روش «ذخیره و هدایت»

قبل از آنکه به تشریح جزئیات لایه شبکه بپردازیم شاید مروری بر حیطه و زمینه عملکرد پروتکل های لایه شبکه، خالی از لطف نباشد. الگوی کلی چنین محیطی در شکل ۱-۵ دیده می شود. مؤلفه های اصلی این سیستم عبارتند از: تجهیزات حامل (Carrier Equipments) که در این شکل درون بیضی سایه دار قرار گرفته اند و تجهیزات مشتریان (Customer's Equipments) که در خارج از بیضی قرار دارند. ماشین میزبان H1 از طریق یک خط اجاره ای، مستقیماً به یکی از مسیریابهای حامل یعنی A، متصل شده است. در مقابل H2 بر روی یک شبکه LAN

۱. در این فصل به کزات از واژه «زیر شبکه» به معنای Communication Subnet (مجموعه تمام مسیریابهای شبکه و خطوط ارتباطی بین آنها) بهره خواهیم گرفت. -م



شکل ۵-۱. محیطی که پروتکل های لایه شبکه در آن عمل می کنند.

قرار گرفته که دارای یک مسیریاب است و تحت فرمان مشتری عمل می کند. مسیریاب F، دارای یک خط اجاره ای با یکی از «تجهیزات حامل» است. ما در این شکل، مسیریاب F را خارج از بیضی در نظر گرفته ایم چرا که عضو بخش حامل [یعنی زیر شبکه اصلی] نیست ولیکن ممکن است از لحاظ ساختار، نرم افزار و پروتکل تفاوتی با مسیریاب های حامل نداشته باشد. اینکه آیا چنین مسیریابی متعلق به زیر شبکه هست یا نه قابل بحث است ولیکن با اهدافی که در این فصل مدنظر داریم مسیریاب های سمت مشتری را نیز به عنوان بخشی از زیر شبکه در نظر می گیریم چرا که آنها نیز همانند مسیریاب های حامل از الگوریتم های مشابهی استفاده می کنند (و اهم کار ما نیز الگوریتم ها هستند).

از این تجهیزات به نحو زیر استفاده می شود: یکی از ماشین های میزبان که بسته ای را برای ارسال آماده دارد آن را برای نزدیکترین مسیریاب می فرستد، خواه این مسیریاب بر روی LAN خودش واقع باشد و خواه از طریق یک خط اجاره ای مستقیماً به زیر شبکه حامل متصل شده باشد. بسته ارسالی در آن مسیریاب موقتاً ذخیره می شود تا بطور کامل دریافت و کد کشف خطای آن، بررسی گردد؛ سپس به مسیریاب بعدی واقع بر مسیر، هدایت می شود و این روند آنقدر تکرار می شود تا بسته به ماشین مقصد رسیده و تحویل داده شود. این مکانیزم همان روش «ذخیره و هدایت» است که در فصول قبلی بدانها پرداختیم.

۲-۱-۵ خدمات ارائه شده برای لایه انتقال

لایه شبکه خدماتی را برای لایه انتقال تدارک می بیند که توسط واسطه میانی لایه شبکه و انتقال^۱ عرضه می شود. سؤال مهم آن است که لایه شبکه چه نوع خدماتی را برای لایه انتقال تدارک می بیند؟ خدمات لایه شبکه با در نظر داشتن اهداف زیر طراحی شده اند:

۱. این خدمات بایستی مستقل از تکنولوژی بکار رفته در مسیریاب باشد.
۲. لایه انتقال باید از درگیری با جزئیاتی مثل تعداد، نوع و توپولوژی مسیریاب های موجود دور نگه داشته شود.
۳. آدرس های شبکه که در اختیار لایه انتقال قرار می گیرند بایستی از ساختار متحدالشکل و استاندارد برخوردار باشند (خواه در شبکه LAN و خواه در شبکه WAN).

با در نظر داشتن این اهداف، طراحان لایه شبکه در تدوین شرح مبسوط خدماتی که باید به لایه انتقال عرضه شود آزادی عمل دارند. این آزادی عمل اغلب به مناقشات دیرینه دو گروه با طرز فکر مخالف دامن می زند. مناقشه

۱. Network Layer/Transport Layer Interface

بر سر آن است که آیا لایه شبکه باید خدمات اتصال گرا (Connection Oriented) ارائه کند یا خدمات بدون اتصال (Connectionless).

یکی از طرفین (به رهبری دست‌اندرکاران اینترنت) استدلال می‌کند که وظیفه یک مسیر یاب هدایت بسته‌هاست و نه چیز دیگر! از دیدگاه آنها (به پشتوانه سی سال تجربه واقعی با یک شبکه کامپیوتری حقیقی و در حال کار) زیر شبکه ذاتاً غیر قابل اعتماد است و چگونگی طراحی آن اهمیت ندارد. بدین ترتیب ماشینهای میزبان باید این حقیقت را بپذیرند که زیر ساخت ارتباطی شبکه آنها غیر قابل اعتماد است و خودشان موظفند عملیات نظارت بر خطاهای احتمالی (شامل کشف و تصحیح) و همچنین کنترل جریان (Flow Control) را بر عهده بگیرند.

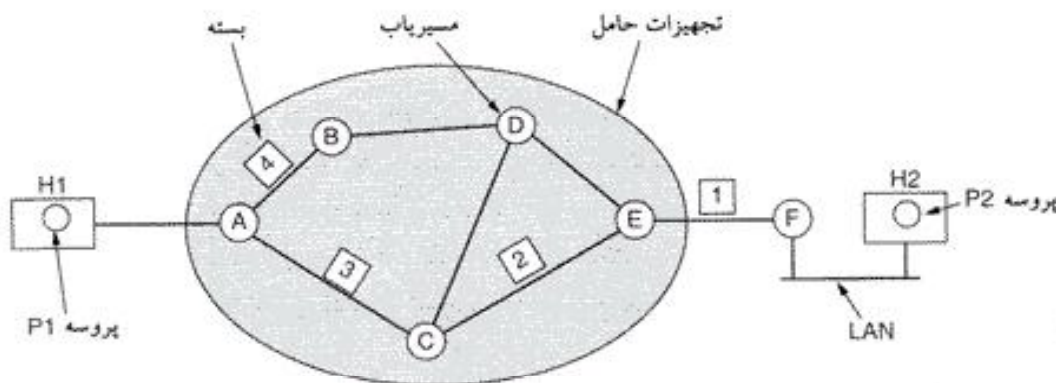
این دیدگاه سریعاً بدانجا منتهی می‌شود که خدمات لایه شبکه باید «بدون اتصال» بوده و فقط عملیات پایه و ابتدایی SEND PACKET و RECEIVE PACKET و تعداد کمی دیگر از همین عملیات را انجام بدهند. خصوصاً نباید عملیات مرتب‌سازی بسته‌ها [از لحاظ ترتیب ارسال] و کنترل جریان انجام شود چرا که ماشینهای میزبان در هر حال چنین کاری را انجام می‌دهند و انجام دوباره یک کار فایده چندانی نخواهد داشت. همچنین هر بسته باید مشخصات کامل آدرس مقصد را با خود داشته باشد چرا که بسته‌های ارسالی باید مستقل از یکدیگر و فارغ از آنکه مسیر یاب قبلی کدام بوده، هدایت شوند.

طرف مقابل این مناقشه (به رهبری شرکت‌های مخابراتی) دلیل می‌آورد که زیر شبکه باید خدماتی اتصال گرا و قابل اعتماد ارائه کند. اینان نیز مدعیند که سابقه درخشان و صد ساله آنها در سیستمهای جهانی تلفن، دلیل عالی و رهنمای راه آنهاست. از دیدگاه این گروه، کیفیت خدمات (QoS) عمده‌ترین مسئله است و اگر زیر شبکه اتصال گرا نباشد دستیابی به کیفیت خدمات (بالاخص برای ترافیک داده‌های بی‌درنگ مثل صدا و تصویر) دشوار خواهد بود. این دو طرف دعوا نمونه‌های خاص خود یعنی اینترنت و ATM را در دست داشتند: اینترنت خدمات لایه شبکه را بدون اتصال عرضه می‌کند در حالی که ATM در لایه شبکه خدماتی اتصال گرا ارائه می‌نماید. ولیکن اشاره به این نکته جالب است که هر چه تضمین کیفیت خدمات بیشتر و بیشتر اهمیت پیدا می‌کند، اینترنت نیز رشد می‌نماید. خصوصاً آنکه در راه کسب ویژگیهایی است که عموماً متناسب به خدمات اتصال گرا هستند؛ بعداً در این خصوص اشاراتی خواهیم داشت. در خلال مطالعه شبکه‌های VLAN در فصل چهارم نمونه‌ای از این رشد و تکامل را ارائه کردیم.

۳-۱-۵ پیاده‌سازی خدمات بی‌اتصال

پس از نگاهی بر دو رده از خدماتی که لایه شبکه می‌تواند به کاربران عرضه نماید وقت آن فرا رسیده که ببینیم در درون این لایه چه می‌گذرد. بسته به نوع خدمات ارائه شده، دو نوع معماری متفاوت ممکن خواهد بود. اگر خدمات بدون اتصال عرضه شود بسته‌ها بطور مجزا و مستقل به درون زیر شبکه تزریق و مستقل از یکدیگر هدایت و مسیر یابی می‌شوند و هیچگونه تنظیمات قبلی نیاز نیست. در چنین حالتی عموماً به بسته‌ها «دیتاگرام» (Datagram) و به زیر شبکه نیز «زیر شبکه دیتاگرام» اطلاق می‌شود. اگر خدمات اتصال گرا عرضه شود قبل از ارسال هر گونه بسته داده بایستی از قبل یک مسیر بین مسیر یاب مبدا و مسیر یاب مقصد تنظیم و ایجاد شود. چنین ارتباطی اصطلاحاً VC (مخفف Virtual Circuit) نامیده می‌شود (مترادف با ایجاد یک مدار فیزیکی در شبکه تلفن) و به زیر شبکه نیز، «زیر شبکه مدار مجازی» (Virtual Circuit Subnet) گفته می‌شود. در این بخش ابتدا به زیر شبکه‌های دیتاگرام می‌پردازیم و در بخش بعدی زیر شبکه‌های مدار مجازی را مطالعه خواهیم نمود.

حال بیایید به بررسی عملکرد زیر شبکه مبتنی بر دیتاگرام بپردازیم. فرض کنید پروسه P1 در شکل ۵-۲ پیامی طولانی برای پروسه P2 دارد. او این پیام را به لایه انتقال سپرده و از او می‌خواهد که پیام را به پروسه P2 بر روی



جدول A

	initially	later
A	-	-
B	B	B
C	C	C
D	B	B
E	C	B
F	C	B

جدول C

A	A
B	A
C	-
D	D
E	E
F	E

جدول E

A	C
B	D
C	C
D	D
E	-
F	F

(خط) Dest. Line (مقصد)

شکل ۵-۲. مسیریابی در یک زیرشبکه دیتاگرام.

ماشین میزبان H2 تحویل بدهد. کد نرم افزار لایه انتقال که عموماً بخشی از سیستم عامل به حساب می آید اجرا شده و سرآیند لایه انتقال در ابتدای پیام درج و نتیجه، تحویل لایه شبکه می شود که آن نیز یک پروتکل اجرایی (روال اجرایی) در سیستم عامل است.

فرض کنیم که پیام چهار برابر طولانی تر از اندازه مجاز بسته های داده باشد؛ لذا لایه شبکه مجبور است آنها را به چهار بسته ۱ و ۲ و ۳ و ۴ بشکند و هر یک از آنها را بطور مجزا و از طریق یک پروتکل نقطه به نقطه (نظیر PPP)^۱ برای مسیریاب A می فرستد. از اینجا به بعد ادامه کار بر عهده زیرشبکه حامل قرار می گیرد. هر مسیریاب دارای جدولی داخلی است که مشخص می کند برای رسیدن به تمام مسیریابهای مقصد در شبکه، بسته باید بر روی کدام خط خروجی [کدام مسیریاب بعدی] ارسال شود. هر یک از «درایه های» این جدول (Table Entry) شامل یک جفت آیتم اطلاعاتی است که یکی «مقصد» و دیگری خط خروجی برای رسیدن بدان مقصد را تعیین می نماید. برای هدایت بسته فقط می توان از خطوطی که مستقیماً به یک مسیریاب مجاور متصلند بهره گرفت. به عنوان مثال در شکل ۵-۲، مسیریاب A فقط دو خط خروجی (به مسیریابهای B و C) دارد و طبعاً هر یک از بسته های ورودی باید برای یکی از این دو مسیریاب ارسال گردد، حتی اگر مقصد نهایی به مسیریابی کاملاً متفاوت متصل باشد. جدول مسیریابی ابتدایی A با عنوان "Initialy"، در شکل ۵-۲ مشخص شده است.

پس از ورود بسته های ۱ و ۲ و ۳ به مسیریاب A، موقتاً در حافظه ذخیره می شوند (تا کدهای کشف خطای آنها بررسی شود). سپس براساس جدول مسیریابی A، این بسته ها به سمت مسیریاب C هدایت می شوند. به همین ترتیب و در ادامه طی مسیر، بسته ۱ از E و نهایتاً F می گذرد. پس از آنکه بسته به F رسید درون فریم لایه پیوند داده

۱. پروتکل PPP یک پروتکل برای خطوط نقطه به نقطه است که در لایه پیوند داده ها عمل می کند یعنی یک فریم را بر روی کانالی واحد و غیر مشترک، از نقطه ای به نقطه دیگر تحویل می دهد. به فصل سوم مراجعه کنید. -م.

جاسازی (کپسوله) شده و از طریق شبکه LAN برای H2 ارسال می‌گردد. بسته‌های ۲ و ۳ نیز به همین طریق مسیریابی و هدایت می‌شوند.

ولیکن (به عنوان مثال) برای بسته ۴ اتفاق متفاوتی می‌افتد. وقتی این بسته به A می‌رسد اگرچه به مقصد F طی مسیر می‌کند ولیکن برخلاف قبل به سوی مسیریاب B ارسال می‌شود. به دلایل متعدد A تصمیم گرفته تا بسته ۴ را از مسیری متفاوت با مسیر سه بسته قبلی، به سمت F بفرستد. شاید این مسیریاب متوجه ازدیاد ترافیک و ازدحام بر روی مسیر ACE شده و جدول مسیریابی خود را به جدولی جدید (که در شکل با عنوان Later مشخص شده) اصلاح و بهنگام‌سازی کرده است. الگوریتمی که مدیریت این جداول را بر عهده دارد و در خصوص مسیریابی تصمیم می‌گیرد اصطلاحاً «الگوریتم مسیریابی» (Routing Algorithm) نامیده می‌شود. الگوریتمهای مسیریابی یکی از اصلیترین مفاد این فصل هستند.

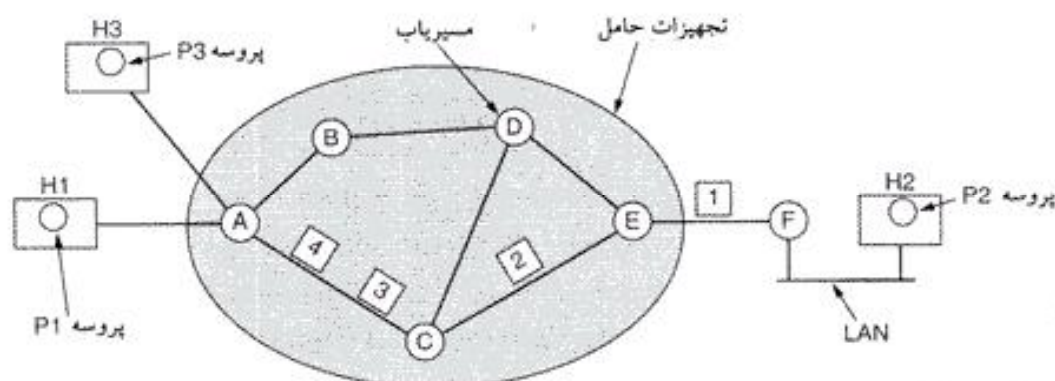
۴-۱-۵ پیاده‌سازی خدمات اتصال‌گرا

رای عرضه خدمات اتصال‌گرا، به زیرشبکه‌های مبتنی بر «مدار مجازی» (Virtual Circuit) نیازمندیم. حال ببینیم که چگونه کار می‌کند: ایده اصلی در مدار مجازی آن است که برخلاف مثال شکل ۵-۲، از انتخاب مسیرهای جدید برای هر بسته اجتناب شود. در عوض وقتی یک «اتصال» ایجاد شد، به عنوان بخشی از عملیات تنظیم اتصال (Connection Setup)، مسیری بین ماشین مبدا و ماشین مقصد انتخاب می‌شود و مشخصات آن در جدول داخلی هر مسیریاب درج می‌گردد. دقیقاً مشابه با سیستم تلفن پس از برقراری اتصال، تمام بسته‌ها از طریق همین مسیر هدایت خواهند شد. پس از آنکه اتصال برقرار شده خاتمه یافت مدار مجازی متناظر نیز پایان یافته و حذف می‌شود. در خدمات اتصال‌گرا هر بسته با خود یک شماره شناسایی حمل می‌کند که مشخص می‌کند به کدام مدار مجازی تعلق دارد.^۱

به عنوان مثال، به وضعیت نشان داده شده در شکل ۵-۳ توجه کنید. در اینجا ماشین میزبان H1 اتصال شماره ۱ را با ماشین میزبان H2 برقرار نموده است. مشخصه این اتصال به عنوان اولین درایه (Entry) در جداول مسیریابی درج شده است. اولین سطر از جدول A بیان می‌کند که اگر بسته‌ای با شماره شناسایی ۱ از H1 دریافت شود باید با همین شماره شناسایی برای مسیریاب C ارسال گردد. به روش مشابه مسیریاب C مطابق با اولین درایه خود، بسته را با شماره شناسایی ۱ به سوی مسیریاب E هدایت می‌نماید.

حال بررسی کنیم که اگر H3 نیز بخواهد اتصالی را با H2 برقرار کند چه اتفاقی می‌افتد. او نیز شماره شناسایی اتصال را ۱ در نظر می‌گیرد (چرا که این اتصال، اولین اتصالی است که او برقرار کرده است) و سپس از زیرشبکه می‌خواهد که برایش مداری مجازی ایجاد کند. انجام این تقاضا منجر به اضافه شدن سطر دوم به جداول است. دقت کنید که در اینجا یک تناقض وجود دارد و آن هم اینکه اگرچه A می‌تواند بسته‌هایی که با شماره شناسایی ۱ از H1 می‌رسند را از بسته‌هایی با همین شماره که از H3 می‌رسند تمیز بدهد ولی مسیریاب C قادر به چنین کاری نیست [چرا که هر دو با یک شماره و از A می‌رسند]. به همین دلیل A برای ترافیک خروجی که در قالب اتصال ۲ خارج می‌شوند شماره شناسایی متفاوتی را در نظر می‌گیرد. برای اجتناب از تناقضاتی از این دست، مسیریابها باید بتوانند شماره شناسایی اتصال هر بسته خروجی را تغییر بدهند.

۱. اتصال را در ذهن خود با یک تماس تلفنی قیاس کنید. ابتدا شماره می‌گیرید، تماس شما برقرار می‌شود، تا زمان دلخواه گفتگو می‌کنید و نهایتاً تماس را قطع می‌نمایید. اتصال در زیرشبکه مدار مجازی بمعنای هماهنگی قبلی بین مبدا، مقصد و مسیریابهای میانی تلفی می‌شود. به مسیری که با هماهنگی قبلی ایجاد می‌شود یک «مدار مجازی» می‌گویند و دارای یک شماره شناسایی است. به این شماره شناسایی اصطلاحاً «شناسه مدار مجازی» یا «شناسه اتصال» (Connection ID) اطلاق می‌شود. -م



جدول A				جدول C				جدول E			
H1	1	C	1	A	1	E	1	C	1	F	1
H3	1	C	2	A	2	E	2	C	2	F	2
In		Out									

شکل ۵-۳. مسیریابی در یک زیر شبکه مدار مجازی.

۵-۱-۵ مقایسه زیر شبکه های مدار مجازی و دیتاگرام

هر یک از روشهای مدار مجازی و دیتاگرام حامیان و منتقدین خود را دارند. حال تلاش می کنیم مباحثات آنها را به طور خلاصه ارائه نماییم. محورهای اصلی این مباحثات در شکل ۵-۴ فهرست شده اند؛ هر چند ممکن است سفسطه کنندگان بتوانند برای هر یک از موارد این جدول، مثال نقضی پیدا کنند.

در هر زیر شبکه می توان اصولی را برای قیاس زیر شبکه های مدار مجازی و دیتاگرام مطرح کرد. یکی از آنها مسئله میزان حافظه مسیریاب در مقایسه با پهنای باند تلفاتی است. روش مدار مجازی اجازه می دهد که بسته ها به جای همراه داشتن آدرسهای کامل فقط شماره مدار مجازی را با خود داشته باشند. هر گاه بسته ها نسبتاً کوتاه باشند وجود آدرس کامل در هر بسته ممکن است حجم سربار زیادی را تحمیل کرده و بدین ترتیب بخشی از پهنای باند مفید هدر می رود. در عوض، هزینه ای که برای بکارگیری مدار مجازی پرداخت می شود فضای حافظه ای است که جدول مشخصات مدارات مجازی در درون مسیریاب، به خود اختصاص می دهد. بسته به قیمت پهنای باند کانالهای مخابراتی در مقایسه با حافظه مسیریاب یکی از این دو ارزانه تر تمام می شود.

یکی دیگر از مسائل قیاسی، «زمان تنظیم و ایجاد مدار مجازی» در مقایسه با زمان جستجو و تحلیل آدرسها است.^۱ استفاده از مدار مجازی منوط به طی مراحل تنظیم مسیر است که فرآیندی زمان بر بوده و منابع زیر شبکه را مصرف می کند. با این وجود در زیر شبکه مبتنی بر مدار مجازی، تشخیص آنکه چه کاری با یک بسته داده باید انجام شود ساده است: مسیریاب از شماره شناسایی مدار مجازی به عنوان اندیس جدول مسیریابی استفاده می کند تا مسیری که بسته باید طی کند مشخص شود. در زیر شبکه دیتاگرام برای جستجو و پیدا کردن درایه متناظر با مقصد، به روالی پیچیده تر نیاز است.

مورد دیگر، فضای مورد نیاز جدول مسیریابی در حافظه مسیریاب است. یک زیر شبکه دیتاگرام نیازمند آن است که به ازای هر مقصد در شبکه، یک درایه (Entry) در جدول مسیریابی خود داشته باشد، در حالی که در

۱. Setup Time versus Address Parsing Time

مورد	زیر شبکه دیتاگرام	زیر شبکه مدار مجازی
تنظیم مدار (Circuit Setup)	نیازی نیست	نیاز است.
آدرس دهی	هر بسته آدرس دقیق و کامل مبدا و مقصد را با خود حمل می کند.	هر بسته فقط یک شماره کوتاه مدار مجازی (VC) را با خود دارد.
اطلاعات وضعیت	مسیر یابی نیازی به نگهداری اطلاعاتی در خصوص وضعیت هر اتصال ندارد.	به ازای هر مدار مجازی تمام مسیر یابی باید اطلاعاتی در خصوص وضعیت آن نگاه دارند.
مسیر یابی	هر بسته بطور مستقل مسیر یابی می شود.	مسیر فقط یکبار و آنهم در هنگام تنظیم مدار مجازی انتخاب می شود و تمام بسته ها از همان مسیر حرکت می کنند.
تأثیر خرابی مسیر یابی	بی تأثیر، مگر در مورد بسته هایی که در حین خرابی از بین رفته اند.	تمام مدارات مجازی که از مسیر یابی خراب شده می گذشته اند قطع می شوند.
تضمین کیفیت خدمات	دشوار	اگر برای هر مدار مجازی منابع لازم از قبل تخصیص یابد، بسیار آسان است.
کنترل ازدحام	دشوار	اگر برای هر مدار مجازی منابع لازم از قبل تخصیص یابد، بسیار آسان است.

شکل ۵-۴. مقایسه زیر شبکه های دیتاگرام و مدار مجازی.

زیر شبکه مدار مجازی، فقط به ازای هر اتصال به یک درایه نیاز است. ولیکن این حسن تا حدودی گمراه کننده و توخالی است چرا که بسته های تنظیم اتصال [که برای اولین بار ارسال می شوند] بهر نحو باید همانند روش دیتاگرام یکبار مسیر یابی شوند و آنها نیز دارای آدرسهای کامل مقصد هستند. [یعنی در زیر شبکه مدار مجازی هم به جدول مسیر یابی و هم به جدول مدارات مجازی نیاز است. -م]

مدارهای مجازی دارای محاسنی در خصوص تضمین «کیفیت خدمات» (QoS) هستند و از بروز ازدحام (congestion) در زیر شبکه جلوگیری می کنند چرا که «منابع» (شامل بافر، پهنای باند و سیکل های CPU) می تواند پیشاپیش و در هنگام ایجاد اتصال، رزرو شود. بعداً، به محض دریافت بسته ها، پهنای باند مورد نیاز و ظرفیت لازم در مسیر یابی، مهیا و آماده است. در زیر شبکه های دیتاگرام، اجتناب از ازدحام دشوارتر است.

در «سیستم های پردازش تراکنشی»^۱، (همانند وقتی که خرید با کارتهای اعتباری بررسی می شوند) مسئله تنظیم یک مدار مجازی و سپس حذف آن ممکن است کاربرد این روش را از اعتبار ساقط کند. [به عبارت بهتر در سیستم هایی که تعامل و مبادله داده با ماشینها، کوتاه و سریع است ایجاد یک مدار مجازی و ختم آن سربار زیادی به شبکه تحمیل می نماید. -م] اگر پخش اعظم ترافیک از این نوع باشد استفاده از مدار مجازی درون یک زیر شبکه چندان مناسب نیست. در طرف مقابل هرگاه حجم داده های ارسالی انبوه باشد، مدارهای مجازی دائم که به صورت دستی تنظیم می شوند و برای ماهها یا سالها تغییر نمی کنند می تواند سودمندتر باشد.

مدارهای مجازی مشکل «آسیب پذیری» دارند. اگر یک مسیر یابی از کار بیفتد و حافظه خود را از دست بدهد (حتی در صورتی که چند ثانیه بعد به زیر شبکه برگردد)، قطعاً تمام مدارهای مجازی که از آن عبور می کنند، از دست خواهند رفت. در مقابل اگر یک مسیر یابی مبتنی بر دیتاگرام از کار بیفتد تنها کاربرانی آسیب می بینند که

بسته های آنها هنگام بروز مشکل، درون حافظه مسیریاب در صف منتظر بوده و بر حسب آنکه آیا دریافت بسته های داده قبلاً اعلام شده باشد یا نه، حتی ممکن است فقط بخشی از آنها با مشکل مواجه شوند. از دست رفتن یک خط مخابراتی برای مدارات مجازی که از آن خط استفاده کرده اند بحران زاست در حالی که اگر از روش دیتاگرام استفاده شده باشد این اشکال براحتی جبران خواهند شد. در روش دیتاگرام مسیریابها مجازند ترافیک را بر روی زیر شبکه توزیع و موازنه کنند لذا ممکن است در حین ارسال یک دنباله طولانی از بسته ها مسیریابها عوض شوند.

۲-۵ الگوریتمهای مسیریابی

وظیفه اصلی لایه شبکه، مسیریابی و هدایت بسته های داده از ماشین مبدا به ماشین مقصد است. در اکثر زیر شبکه ها، بسته ها برای طی مسیر خود بایستی چندین «گام» (HOP) راه پیمایند.^۱ الگوریتمی که این مسیرها را انتخاب می کنند و همچنین ساختمان داده مورد استفاده، یکی از زمینه های مهم در طراحی لایه شبکه محسوب می شود.

«الگوریتمهای مسیریابی» آن بخش از نرم افزار لایه شبکه اند که مسئولیت دارند در خصوص خط خروجی که یک بسته ورودی باید بر روی آن ارسال شود، تصمیم گیری کنند. اگر در داخل زیر شبکه از روش دیتاگرام استفاده شده باشد این تصمیم گیری باید به ازای هر بسته دریافتی از نو تکرار شود چرا که در هر لحظه ممکن است بهترین مسیر تغییر نماید. اگر زیر شبکه از روش مدار مجازی بهره گرفته باشد، تصمیم گیری در خصوص مسیرها فقط یکبار و آنهم در هنگام تنظیم و ایجاد هر مدار مجازی جدید صورت می گیرد و طبعاً بسته های داده، همگی از مسیری که قبلاً ایجاد شده هدایت می شوند. این حالت گاهی روش «مسیریابی مبتنی بر نشست» (Session Routing) نامیده می شود چرا که مسیر ایجاد شده در خلال نشست کاربر برقرار خواهد ماند؛ (مثلاً در حین نشست از راه دور با ترمینال Login session یا نشست انتقال فایل).

گاهی تمایز قائل شدن بین فرآیند «مسیریابی» (که به تصمیم گیری در خصوص مسیرهای بهینه اطلاق می شود) و فرآیند «هدایت» (Forwarding) (آنچه که با ورود بسته اتفاق می افتد) مفید است: می توانید بدینگونه بیندیشید که هر مسیریاب دارای دو پروسه در درون خود است: یکی از آنها بسته ها را به محض ورود پردازش کرده و از طریق جدول مسیریابی، یک خط خروجی مناسب برای آنها انتخاب می نماید. این پروسه همان «عمل هدایت» است. پروسه دیگر موظف به پر کردن و بهنگام سازی محتویات جدول مسیریابی است. این همان جایی که «الگوریتمهای مسیریابی» (Routing Algorithm) به میدان می آیند.

فارغ از آنکه آیا برای هر بسته، مسیرها بطور مستقل و جداگانه انتخاب می شوند یا آنکه فقط یکبار و آن هم در حین ایجاد یک اتصال [مدار مجازی] تعیین می گردد، از یک الگوریتم مسیریابی ویژگیهای خاصی انتظار می رود:

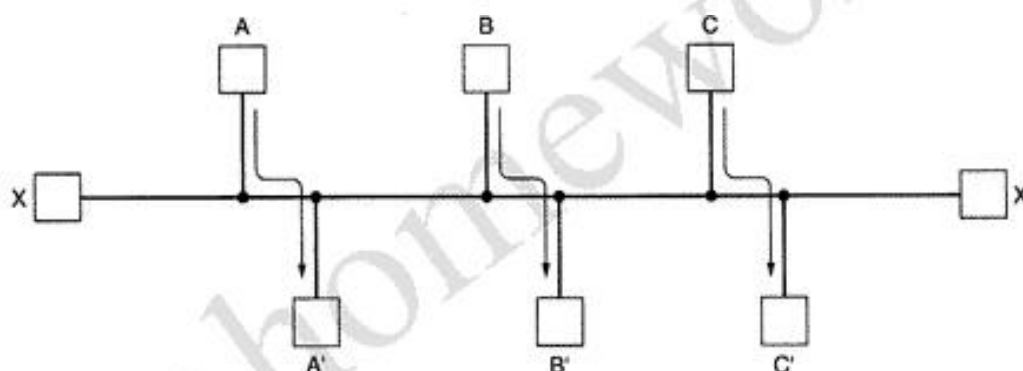
- (۱) صحت عملکرد (correctness) (۲) سادگی (simplicity) (۳) قابلیت تحمل (Robustness) (۴) پایداری (stability) (۵) مساوات (Fairness) (۶) بهینگی (Optimality).

ویژگی «صحت» و «سادگی» نیازی به توضیح ندارد در حالی که نیاز به «قابلیت تحمل» در بدو امر چندان روشن به نظر نمی رسد. با ورود یک شبکه عظیم به صحنه عمل، انتظار آنست که برای سالها بدون بروز هیچگونه خرابی سیستم در سطح وسیع، به کار خود ادامه بدهد. در خلال این دوره زمانی ممکن است انواع سخت افزارها و نرم افزارها از کار بیفتند. ماشینهای میزبان، مسیریابها و خطوط ارتباطی به کرات از کار می افتند و توبولوژی شبکه

۱. به گذر بسته از یک مسیریاب، گام یا Hop گفته می شود

چندین بار تغییر می‌کند. الگوریتم مسیریابی باید بتواند هر گونه تغییر در توپولوژی و ترافیک را بدون ایجاد وقفه در کار ماشینهای میزبان یا نیاز به راه‌اندازی مجدد شبکه (در صورت از کار افتادن برخی از مسیرهایها)، اصلاح و مدیریت نماید.

یکی دیگر از اهداف و ویژگیهای الگوریتم مسیریابی «پایداری» است. برخی از الگوریتمهای مسیریابی هرگز به یک عملکرد ثابت و پایدار همگرا نمی‌شوند بدون آنکه مدت زمان در حال کار و اجرا بودن آنها اهمیت و تأثیری داشته باشد. ویژگیهای «مساوات» و «بهینگی» ممکن است واضح به نظر برسند و مطمئناً هیچ عقل سلیمی با آن مخالف نیست ولی گاهی در مرحله عمل با یکدیگر در تناقض و تضاد هستند. به عنوان یک مثال از این تناقض، به شکل ۵-۵ نگاه کنید. فرض نمائید که ترافیک بین A و A'، بین B و B' و بین C و C' بقدری است که لینک ارتباطی افقی اشباع می‌شود. برای آنکه جریان کلی اطلاعات به حداکثر برسد باید ترافیک بین X و X' قطع گردد؛ [تا بهینگی رعایت شده باشد]. متأسفانه X و X' ممکن است متوجه چنین موضوعی نباشند ولیکن باید بین «کارایی» و «مساوات» حالتی بینابین و متعادل مدنظر قرار بگیرد. اگر ترافیک بین X و X' قطع شود بهینگی بدست می‌آید ولی در عوض مساوات رعایت نخواهد شد!



شکل ۵-۵. تضاد بین «بهینگی» و «مساوات».

قبل از آنکه بتوان بین «مساوات» و «بهینگی» تعادل ایجاد کرد باید ابتدا تصمیم بگیریم که در پی بهینه کردن چه چیزی هستیم! به حداقل رساندن متوسط تأخیر بسته‌ها نامزد واضحی برای بهینه شدن است ولی به حداکثر رساندن بازده مفید شبکه (Throughput) نیز نامزد دیگری است. به علاوه این دو هدف با یکدیگر در تناقض هستند چرا که مثلاً هر گاه یک سیستم صف‌بندی [همانند صفی که در هر مسیر یاب ایجاد می‌شود] در ظرفیت حداکثر خود کار کند تأخیری طولانی را تحمیل خواهد کرد.^۱ به عنوان یک حالت بینابین، در برخی از شبکه‌ها سعی شده تا تعداد «گامهای مسیر» (که یک بسته باید طی کند) به حداقل برسد زیرا کاهش تعداد گامها باعث کاهش تأخیر و پهنای باند مصرفی خواهد شد که نهایتاً به بهبود بازده مفید شبکه می‌انجامد.

الگوریتمهای مسیریابی را می‌توان به دو رده کلی تقسیم‌بندی کرد: «غیر وفقی» (nonadaptive) و «وفقی» (adaptive). روشهای غیر وفقی تصمیم‌گیری خود در خصوص مسیریابی را بر مبنای اندازه‌گیری و تخمین هوشمند ترافیک و توپولوژی فعلی شبکه، قرار نداده‌اند. در عوض برای انتخاب مسیری بین I و I (برای هر I و I مفروض)، یک مسیر از قبل و به صورت offline محاسبه شده و در هنگام راه‌اندازی شبکه درون مسیریابها بارگذاری می‌شود. به این روال اغلب «مسیریابی ایستا» (Static Routing) گفته می‌شود.

۱. در صف نگاه داشتن بسته‌ها می‌تواند به بازده مفید شبکه بیفزاید ولی تأخیر را نیز افزایش خواهد داد. رجوع کنید به نظریه

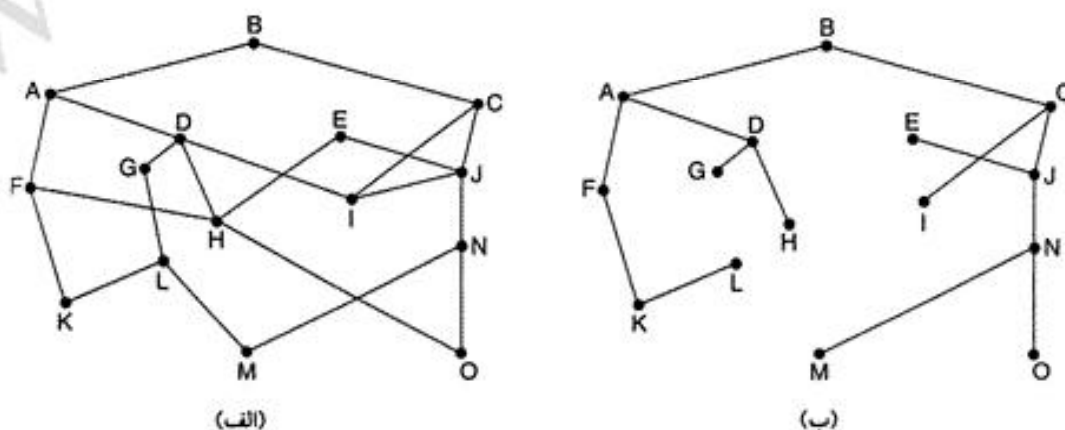
در مقابل، «الگوریتمهای افقی» تصمیم‌گیری در خصوص مسیریابی را برحسب تغییرات توپولوژی و عموماً ترافیک لحظه‌ای، بصورت هوشمند انجام می‌دهند. الگوریتمهای مسیریابی افقی در نحوه گردآوری «اطلاعات مسیر» با یکدیگر متفاوتند (مثلاً اینکه آیا این اطلاعات به صورت محلی گردآوری می‌شود یا از مسیریابها مجاور بدست می‌آید یا از همه مسیریابها می‌رسد). الگوریتمهای افقی همچنین در پارامترهایی نظیر: (۱) زمان بهنگام‌سازی اطلاعات مسیر (مثلاً بطور متناوب هر ΔT ثانیه بهنگام شوند یا فقط زمانی که توپولوژی عوض شود) (۲) معیاری که برای بهینه‌سازی مورد استفاده قرار می‌گیرد (مثل فاصله، تعداد گام (HOP) یا زمان تخمینی انتقال) با یکدیگر متفاوت هستند. در بخش بعدی گونه‌های مختلفی از الگوریتمهای مسیریابی اعم از سویا و ایستا را تشریح خواهیم کرد.

۱۲-۵ اصل بهینگی

قبل از آنکه بر روی یک الگوریتم خاص متمرکز شویم، شاید پرداختن به این نکته مفید باشد که هر کسی می‌تواند بدون توجه به توپولوژی یا ترافیک شبکه یک ارزیابی و تعبیر کلی در خصوص مسیرهای بهینه ارائه بدهد. این تعبیر به نام «اصل بهینگی» مشهور است و بیان می‌کند که اگر مسیریاب J بر روی مسیر بهینه بین مسیریاب I تا مسیریاب K واقع شده باشد بنابراین مسیر بهینه از I تا K نیز بر روی همان مسیر خواهد بود. برای بررسی این موضوع، آن قسمت از مسیر که بین I تا J قرار دارد را r_1 و مابقی مسیر را r_2 بنامید. اگر مسیری بهتر از r_2 بین J تا K وجود داشته باشد می‌توان آن را به r_1 افزود تا مسیر بهتری از I تا K ایجاد شود؛ این موضوع با اصل قضیه که بیان می‌کرد مسیر $r_1 r_2$ بهینه است تناقض دارد.

به عنوان یک نتیجه مستقیم از اصل بهینگی می‌توان اثبات کرد که مجموعه مسیرهای بهینه بین تمام گره‌های مبدا و یک گره مقصد خاص، درختی را تشکیل می‌دهد که ریشه آن بر روی گره مقصد قرار دارد. به چنین درختی اصطلاحاً Sink Tree گفته می‌شود که شمای آن در شکل ۵-۶ به تصویر کشیده شده است. در این شکل معیار فاصله «تعداد گام» در نظر گرفته شده است. دقت کنید که درخت Sink Tree الزاماً یکتا و منحصر به فرد نیست و ممکن است درختی متفاوت با طول مسیرهای مشابه پیدا شود. هدف اصلی الگوریتمهای مسیریابی آن است که برای تمام مسیریابها درخت Sink Tree محاسبه شده و مورد استفاده قرار بگیرد.

بدیهی است که Sink Tree، حداقل یک درخت است، بنابراین بهیچوجه دارای حلقه نیست و بدین ترتیب هر بسته پس از طی چند گام محدود و معین، تحویل مقصد خواهد شد. ولیکن در دنیای عمل همیشه کار بدین سادگی



شکل ۵-۶. (الف) یک زیر شبکه (ب) درخت Sink Tree برای مسیریاب B.

نخواهد بود. لینکها و مسیر یابها می توانند به ناگاه از کار بیفتند و سپس مجدداً عملیات طبیعی خود را از سر بگیرند لذا مسیر یابهای مختلف ممکن است توپولوژی شبکه را به گونه متفاوتی ببینند.^۱ همچنین ممکن است بدین نکته ظریف بیندیشیم که آیا هر مسیر یاب مجبور است مستقلاً اطلاعات لازم در خصوص توپولوژی شبکه را جمع آوری کرده و Sink Tree مربوطه را خودش محاسبه نماید یا آنکه این اطلاعات به روشهای دیگری بدست می آیند. در بخشهای آتی مختصراً بدین موضوعات خواهیم پرداخت. بهر حال، «اصل بهینگی» و «Sink Tree» روشی برای محک زدن و ارزیابی الگوریتمهای مسیر یابی دیگر هستند.

۲-۲-۵ مسیر یابی مبتنی بر کوتاهترین مسیر

اجازه بدهید مطالعات خود پیرامون الگوریتمهای مسیر یابی را با پرداختن به روشی آغاز کنیم که بطور گسترده ای از آن استفاده می شود چرا که ساده و فهم جزئیات آن راحت است. ایده اصلی آن است که گراف زیر شبکه را به گونه ای تشکیل بدهیم که در آن هر گره از گراف یک مسیر یاب را مشخص می کند و هر یک از «کمانها» (Arcs) مشخص کننده یک خط ارتباطی (که اغلب لینک نامیده می شوند) هستند. برای انتخاب بهترین مسیر بین یک زوج مسیر یاب مفروض، یک الگوریتم خاص، «کوتاهترین مسیر» بین آن دو را در این گراف پیدا می کند.

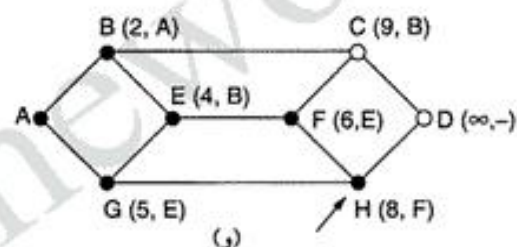
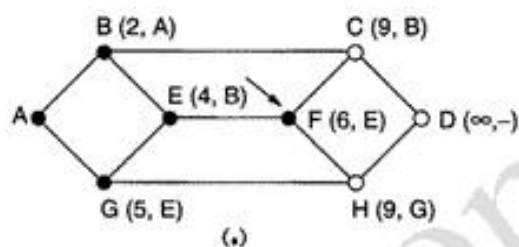
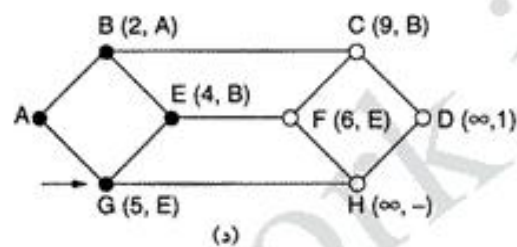
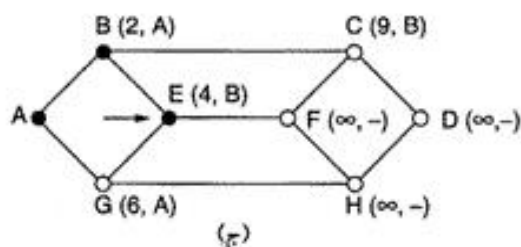
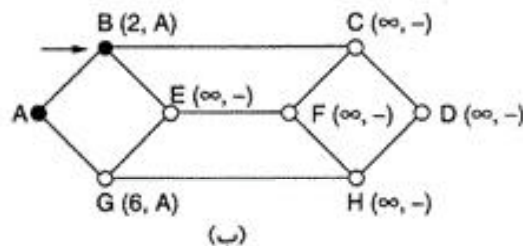
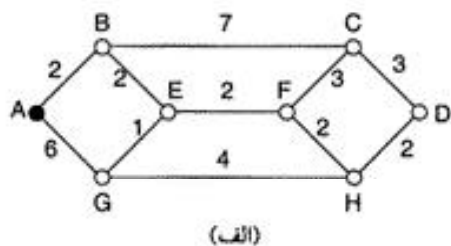
مفهوم «کوتاهترین مسیر» نیاز به اندکی توضیح دارد: یکی از روشهای محاسبه طول مسیر، شمارش تعداد «گام» است. با استناد به این معیار، مسیرهای ABC و ABE در شکل ۷-۵ طول معادلی دارند. یکی دیگر از معیارها، فاصله جغرافیایی گره ها از یکدیگر برحسب کیلومتر است که در این حالت ABC به وضوح از ABE طولانی تر می باشد (با فرض آنکه شکل به گونه ای ترسیم شده که مقیاسی از نقشه واقعی را ارائه می دهد).

با این وجود، به غیر از معیارهای تعداد گام یا فاصله فیزیکی، معیارهای دیگری نیز وجود دارند. به عنوان مثال هر کمان از گراف می تواند برچسبی داشته باشد که مقدار متوسط تأخیر صف [صف درون حافظه مسیر یاب] و تأخیر انتقال را تعیین کند؛ این برچسبها بطور متناوب و با استفاده از برخی بسته های استاندارد محاسبه می شوند. در چنین گراف برچسب داری، کوتاهترین مسیر همانا سریعترین مسیر است (یعنی مسیر با حداقل تأخیر)، نه مسیری که کمترین تعداد کمان یا کوتاهترین فاصله جغرافیایی را دارد.

در حالت کلی، برچسب هر کمان می تواند برحسب تابعی از پارامترهای «فاصله»، «پهنای باند»، «میانگین ترافیک»، «هزینه ارتباط»، «طول متوسط صف»، «تأخیر اندازه گیری شده» یا عوامل دیگر، محاسبه شود. با تغییر در «تابع وزن دهی» (Weighting Function)، این الگوریتم می تواند «کوتاهترین مسیر» را برحسب یکی از این معیارها یا ترکیبی از آنها بدست بیاورد.

تاکنون الگوریتمهای متعددی برای محاسبه کوتاهترین مسیر بین دو گره از گراف، معرفی شده اند. یکی از آنها توسط «دایکسترا» (Dijkstra, 1959) ارائه شد. در الگوریتم دایکسترا، هر گره دارای برچسبی است که فاصله آن گره را تا یک گره مبدا برحسب بهترین مسیر، مشخص می کند. (در شکل ۷-۵ این برچسبها درون پرانتز درج شده اند.) در ابتدا هیچ مسیری مشخص نیست فلذا تمام گره ها در برچسب خود علامت بی نهایت دارند. در حین اجرای الگوریتم و پیدا شدن مسیرها این برچسبها تغییر می کنند تا مسیرهای بهتر را مشخص نمایند. هر برچسب می تواند یکی از دو حالت «موقت» (Tentative) یا «دائم» (Permanent) داشته باشد. در ابتدای کار تمام برچسبها در حالت موقت علامت خورده اند. پس از آن که مشخص شد یک برچسب کوتاهترین مسیر ممکن را از مبدا تا آن گره مشخص کرده، حالت آن گره به حالت «دائم» تغییر کرده و از آن به بعد هیچگاه عوض نخواهد شد.

۱. اگر هر یک از مسیر یابها، مسیرهای بهینه را متفاوت از دیگری تشخیص بدهند و بر اساس اطلاعات ناقص خود مسیر انتخاب کنند، گاهی حلقه بی نهایت ایجاد می شود. -م



شکل ۵-۷. پنج مرحله نخست از روال محاسبه کوتاهترین مسیر از A به D فلشها «گره کار» را مشخص می کنند.

برای آن که ببینیم الگوریتم برچسب گذاری چگونه کار می کند، گراف «بدون جهت» و «وزن دار» شکل ۵-۷-الف را در نظر بگیرید که در آن وزنهای فرضاً فاصله را مشخص کرده است. می خواهیم کوتاهترین مسیر از A به D را پیدا کنیم. کار را با تغییر علامت گره A به حالت «ثابت» (که به صورت یک دایره توپر مشخص شده) شروع می کنیم. سپس به نوبت هر یک از گره های مجاور گره A (که گره کار نام دارد) را آزمایش می کنیم و برچسبشان را برحسب فاصله آنها تا A، تغییر می دهیم. هرگاه برچسب یک گره را تغییر دادیم، در آن برچسب، نام گره ای که آزمایش بر مبنای آن صورت گرفته درج می شود تا بعداً بتوان مسیر نهایی را بدست آورد. [یعنی مثلاً وقتی گره های همسایه A یعنی B و G را بررسی می نماییم و فاصله آنها تا A را در برچسبشان تغییر می دهیم، باید نام گره کار قبلی یعنی A در برچسب آنها درج شود. -م] پس از آزمایش گره های مجاور A، تمام گره هایی که به صورت موقتی علامت خورده اند [یعنی دایره های توخالی] را در «کل گراف» بررسی کرده و گره ای با کوچکترین برچسب را انتخاب و آن را به صورت «دائم» علامت گذاری می کنیم. (شکل ۵-۷-ب) این گره به عنوان «گره کار» جدید انتخاب می شود.

حال از گره B شروع کرده و تمام گره های مجاور آن را آزمایش می نماییم. اگر مجموع برچسب B و فاصله B تا گره مورد نظر از آنچه که درون برچسب B نوشته شده کمتر باشد برچسب آن گره تغییر می کند چرا که مسیری کوتاه تر را پیدا کرده ایم. [به عبارت دیگر وقتی گره های همسایه B را بررسی می نماییم - یعنی E و C را - مجموع فاصله C تا B یعنی ۷ را با فاصله B تا A یعنی ۲ جمع می کنیم و چون ۹ از مقدار بی نهایت در برچسب C کمتر است برچسب آن به صورت (9, B) تغییر می کند یعنی تا رسیدن به گره A فاصله ۹ و گره قبلی B است. -م]

پس از آن که گره‌های مجاور «گره کار» بررسی شدند و در صورت لزوم پرچسبهای موقت تغییر نمودند، برای پیدا کردن گره‌ای با علامت موقت و کمترین مقدار فاصله، کل گراف را جستجو می‌نماییم. این گره به حالت «دائم» تغییر وضعیت داده و گره کار جدید در مرحله بعد خواهد شد. شکل ۵-۷ پنج مرحله اول این الگوریتم را نشان می‌دهد.

برای آنکه ببینیم چرا این الگوریتم بدرستی کار می‌کند به شکل ۵-۷-ج دقت نمایید. در این مرحله، حالت گره E را به حالت «دائم» تغییر داده‌ایم. فرض کنید که مسیری کوتاه‌تر به غیر از ABE مثل AXYZE وجود داشته باشد. دو امکان وجود دارد: یا گره Z قبلاً به حالت «دائم» علامت خورده است یا حالت «موقت» دارد. اگر حالت دائم داشته باشد بنابراین نتیجه می‌گیریم که E قبلاً بررسی شده است (در همان مرحله‌ای که گره Z در آن مرحله علامت دائم خورده است) لذا مسیر AXYZE از چشم ما دور نمانده و قاعدتاً نمی‌توانسته کوتاه‌ترین مسیر باشد. اکنون حالتی را در نظر بگیرید که در آن Z هنوز پرچسب موقت دارد. اگر پرچسب Z بیشتر یا مساوی آنچه که در پرچسب E درج شده، باشد که در این حالت مسیر AXYZE نمی‌تواند از ABE کوتاه‌تر باشد؛ یا آنکه پرچسب Z کوچکتر از پرچسب E است که در این حالت Z بجای E علامت دائم می‌خورد و E باید قبل از Z بررسی شود [یعنی از طریق E باید Z بررسی شود و مسیر AXYZE خواهد بود. -م]

این الگوریتم در شکل ۵-۸ ارائه شده است. متغیرهای سراسری n و dist توصیف‌کننده گراف هستند و قبل از فراخوانی تابع shortest_path مقداردهی می‌شوند. تنها تفاوت بین این برنامه و الگوریتمی که در بالا تشریح شد آن است که در برنامه شکل ۵-۸ محاسبه کوتاه‌ترین مسیر از آخر (یعنی از گره پایانی) شروع شده است. از آنجایی که کوتاه‌ترین مسیر از t به s در یک گراف بی‌جهت هیچ تفاوتی با کوتاه‌ترین مسیر از s به t ندارد لذا آنکه از کدام گره شروع کنیم اهمیتی ندارد. (مگر آن که کوتاه‌ترین مسیرها بیش از یکی باشند و جستجوی معکوس مسیر دیگری را نتیجه بدهد.) دلیل آنکه جستجوی کوتاه‌ترین مسیر برعکس انجام می‌گیرد آن است که پرچسب هر گره با گره قبلی خود (نه گره بعدی) تغییر می‌کند و وقتی مسیر نهایی در متغیر خروجی یعنی path کپی می‌شود، مسیر معکوس خواهد شد [چون عمل کپی از آخر به اول انجام می‌گیرد]. چون از گره آخر به اول شروع کردیم و نهایتاً مسیر بطور معکوس کپی شد این دو، اثر هم را خنثی کرده و جواب نهایی بطور صحیح تولید خواهد شد. [جواب نهایی در متغیر path، بهترین مسیر از گره مبدا به گره مقصد خواهد بود.]

۳-۲-۵ الگوریتم سیل آسا (Flooding)

یکی از الگوریتمهای ایستا در ارسال بسته‌ها، «الگوریتم سیل آسا» است که براساس آن هر بسته ورودی به یک مسیر یاب، بر روی تمام خطوط خروجی (به استثنای خطی که بسته از طریق آن دریافت شده) ارسال می‌شود. این فرآیند سیل آسا، به وضوح منجر به تولید تعداد بسیار زیادی بسته‌های تکراری (در حقیقت بی‌نهایت بسته) خواهد شد مگر آنکه برای خاتمه آن، سنجشی صورت بگیرد. یکی از معیارهای سنجش، آنست که در سرآیند هر بسته یک «شمارنده گام» (Hop Counter) داشته باشیم و در هر گام یک واحد از آن کم شود و هنگامی که مقدار این شمارنده به صفر رسید بسته حذف شود. حالت آرمانی آن است که مقدار اولیه این شمارنده معادل طول مسیر بین مبدا و مقصد در نظر گرفته شود ولیکن اگر فرستنده نداند که طول مسیر چقدر است می‌تواند مقدار این شمارنده را به بدترین حالت یعنی بزرگترین طول مسیر در شبکه تنظیم نماید.^۱

راهکار دیگر برای خاتمه دادن به این فرآیند آن است که فهرست بسته‌هایی که قبلاً یکبار به صورت سیل آسا ارسال شده‌اند را نگاه داریم تا از ارسال مجدد آن اجتناب شود. برای انجام این کار، مسیر یاب مبدا باید در هر بسته

^۱ در ادبیات شبکه به بزرگترین طول مسیر، «قطر شبکه» گفته می‌شود.

```

#define MAX_NODES 1024          /* maximum number of nodes */
#define INFINITY 1000000000     /* a number larger than every maximum path */
int n, dist[MAX_NODES][MAX_NODES]; /* dist[i][j] is the distance from i to j */
void shortest_path(int s, int t, int path[])
{
    struct state {               /* the path being worked on */
        int predecessor;        /* previous node */
        int length;             /* length from source to this node */
        enum {permanent, tentative} label; /* label state */
    } state[MAX_NODES];
    int i, k, min;
    struct state *p;
    for (p = &state[0]; p < &state[n]; p++) { /* initialize state */
        p->predecessor = -1;
        p->length = INFINITY;
        p->label = tentative;
    }
    state[t].length = 0; state[t].label = permanent;
    k = t; /* k is the initial working node */
    do { /* Is there a better path from k? */
        for (i = 0; i < n; i++) /* this graph has n nodes */
            if (dist[k][i] != 0 && state[i].label == tentative) {
                if (state[k].length + dist[k][i] < state[i].length) {
                    state[i].predecessor = k;
                    state[i].length = state[k].length + dist[k][i];
                }
            }
        /* Find the tentatively labeled node with the smallest label. */
        k = 0; min = INFINITY;
        for (i = 0; i < n; i++)
            if (state[i].label == tentative && state[i].length < min) {
                min = state[i].length;
                k = i;
            }
        state[k].label = permanent;
    } while (k != s);

    /* Copy the path into the output array. */
    i = 0; k = s;
    do {path[i++] = k; k = state[k].predecessor; } while (k >= 0);
}

```

شکل ۵-۸ الگوریتم دایجکسترا برای محاسبه کوتاهترین مسیر در یک گراف.

که از یک ماشین میزبان دریافت می کند یک «شماره ترتیب» قرار دهد. هر مسیر یاب نیز باید فهرستی تشکیل بدهد و در آن شماره ترتیب بسته هایی را که قبلاً از یک مسیر یاب دریافت کرده، درج کند. با ورود هر بسته ابتدا شماره ترتیب آن درون این فهرست جستجو می شود؛ اگر شماره آن در این فهرست وجود داشته باشد دیگر بهیچوجه (به روش سیل آسا) ارسال نخواهد شد (چون قبلاً یکبار ارسال شده است).

برای آنکه از رشد بی نهایت این فهرست اجتناب شود در فهرست به ازای هر مسیر یاب مبداء، تنها یک شمارنده نگهداری می شود و مقدار این شمارنده یعنی k نشان می دهد که بسته های تا شماره k قبلاً دریافت شده اند. [یعنی در حقیقت فقط شماره ترتیب آخرین بسته تولید شده توسط یک مسیر یاب مشخص می شود. -م] وقتی بسته ای وارد یک مسیر یاب می شود آزمون تکراری بودن آن بسیار ساده است [اگر شماره آن k یا کمتر از k بود تکراری است]؛ اگر بسته تکراری بود حذف می شود. بدین ترتیب به فهرست بسته های دریافتی با شماره کمتر از k نیازی نیست و فقط در اختیار داشتن آخرین شماره یعنی k کفایت می کند.

گونه دیگری از روش سیل آسا که قابلیت اجرایی بیشتری دارد به نام «روش سیل آسا بصورت انتخابی» (Selective Flooding) مشهور است. در این الگوریتم، مسیر یاب تمام بسته های دریافتی از یک خط را بر روی تمام خطوط خروجی ارسال نمی کند بلکه فقط آنها را بر روی خطوطی ارسال می کند که به صورت تخمینی در مسیر صحیحی قرار دارند. هدایت بسته هایی که قرار است به سمت غرب بروند به سمت شرق، توجه چندانی ندارد مگر آنکه توپولوژی زیر شبکه نوع عجیب و ویژه ای باشد و مسیر یاب از آن به درستی آگاه باشد.

الگوریتم سیل آسا در اغلب کاربردها عملی نیست ولیکن در برخی از موارد خاص کاربردهایی دارد. به عنوان مثال در کاربردهای نظامی که گاهی باید برای تعداد زیادی از مسیر یابها و در لحظه ای کوتاه، حجم زیادی اطلاعات ارسال شود، قدرت روش سیل آسا بسیار مطلوب خواهد بود. در پایگاه داده های توزیع شده (Distributed Database) گاهی لازم است که تمام پایگاههای داده بطور همزمان بهنگام سازی شوند؛ در چنین موردی نیز روش سیل آسا مفید خواهد بود. در شبکه های بی سیم تمام پیامهایی که توسط یک ایستگاه ارسال می شود توسط بقیه ایستگاههایی که در برد رادیویی آن قرار گرفته اند قابل شنود است و در حقیقت نوعی از روش سیل آسا محسوب می شود و برخی از الگوریتمها از این ویژگی در راستای کار خود بهره می گیرند. چهارمین کاربرد روش سیل آسا، آنست که می توان از آن به عنوان معیاری برای مقایسه با الگوریتمهای دیگر مسیر یابی بهره گرفت. در روش سیل آسا همیشه کوتاهترین مسیرها انتخاب می شود چرا که بسته ها از تمام مسیرهای ممکن به صورت موازی ارسال خواهد شد. طبعاً هیچ الگوریتم دیگری نمی تواند تأخیر کمتری نسبت به این روش داشته باشد (به شرط آنکه از سر بار تحمیل شده توسط خود پروسه سیل آسا، قابل چشمپوشی باشد).

۵-۲-۵ مسیر یابی بردار فاصله (Distance Vector Routing)

شبکه های کامپیوتری مدرن، عموماً بجای استفاده از روشهای ایستا که در قبل بدانها اشاره شد از روشهای پویا بهره می گیرند زیرا الگوریتمهای ایستا بار فعلی شبکه را در محاسبات مربوط به بهترین مسیرها دخالت نمی دهند. دو الگوریتم پویا یعنی «مسیر یابی بردار فاصله» (DVR) و «مسیر یابی مبتنی بر حالت لینک» (Link State Routing) از رایج ترین روشهای موجود هستند. در این بخش بطور خاص مروری بر الگوریتم نوع اول (یعنی بردار فاصله) خواهیم داشت و در بخش آتی به مطالعه الگوریتم دوم می پردازیم.

الگوریتم «مسیر یابی بردار فاصله» (DVR) بدین نحو کار می کند: هر مسیر یاب جدولی را در خود نگه می دارد (به عبارتی یک بردار را) که در آن بهترین فاصله تا هر مسیر یاب مقصد [یا به عبارتی کمترین هزینه رسیدن به هر مسیر یاب در زیر شبکه] و خطی که برای رسیدن به آن مقصد باید مورد استفاده قرار بگیرد، درج شده است. این جدول با مبادله اطلاعات بین مسیر یابهای همسایه، بهنگام سازی می شود.

الگوریتمهای مسیریابی بردار فاصله گاهی با نامهای دیگری معرفی می شوند که اهم این اسامی عبارتند از «الگوریتم مسیریابی بلمن-فوردر توزیع شده» (Distributed Bellman-Ford) و «الگوریتم فوردر-فوکرسون» (Ford-Fulkerson) که به افتخار نام پژوهشگرانی است که آن را ابداع کرده اند. (Bellman, 1957; Ford and Fulkerson, 1962) این روش اولین الگوریتم مسیریابی در ARPANET بود و بعداً نیز با نام RIP (Routing Information Protocol) در اینترنت بکار گرفته شد.

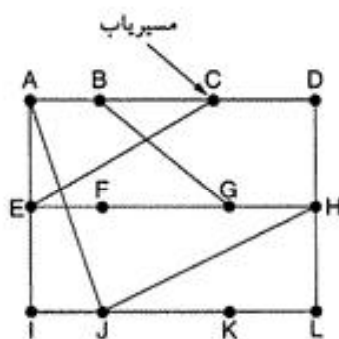
در «مسیریابی بردار فاصله» هر مسیریاب یک جدول مسیریابی دارد که به ازای هر مسیریاب موجود در زیر شبکه، یک «درایه» (Entry) در آن درج شده است. [کل جدول براساس آدرس هر مسیریاب، ایندکس شده است. -م] هر درایه دارای دو بخش است: (۱) خط خروجی مناسب برای رسیدن به مقصد مورد نظر (۲) تخمینی از زمان یا فاصله رسیدن بدان مقصد. معیار هزینه می تواند تعداد گام (Number of Hops)، تأخیر برحسب میلی ثانیه، تعداد کل بسته های به صف شده در آن مسیر یا چیزی شبیه به اینها باشد.

فرض بر آن است که هر مسیریاب «فاصله» خود تا هر یک از همسایه هایش را می داند. اگر معیار بکار رفته تعداد گام باشد فاصله هر مسیریاب از همسایه هایش دقیقاً ۱ است. اگر معیار، طول صف باشد [یعنی تعداد بسته هایی که برای ارسال بر روی یک خط خروجی منتظر هستند] مسیریاب می تواند بسادگی هر یک از صفها را بررسی نماید. اگر معیار، تأخیر باشد مسیریاب می تواند با ارسال بسته های خاصی به نام Echo و دریافت بسته پاسخ (که گیرنده به آن «مهر زمان» (Timestamp) زده و سریعاً برمی گرداند) این تأخیر را مستقیماً اندازه بگیرد.

به عنوان مثال فرض کنید از میزان تأخیر به عنوان معیار بهینگی استفاده شده باشد و مسیریاب تأخیر خود تا هر یک از همسایه ها را بداند. هر T میلی ثانیه یکبار، تمام مسیریابها برای همسایه های خود فهرستی از تأخیر تخمینی در رسیدن به هر یک از مسیریابهای مقصد را ارسال می نمایند. در ضمن فهرست مشابهی را از همسایه های خود دریافت می دارد. تصور کنید که یکی از این جداول از طرف مسیریاب X دریافت شود و نماد X_i مشخص کننده میزان تأخیر برای رسیدن به مسیریاب i باشد. اگر این مسیریاب بداند که تأخیر خودش تا X معادل m میلی ثانیه است می تواند نتیجه بگیرد که با تأخیر $X_i + m$ میلی ثانیه به هر مسیریاب i می رسد. با انجام این محاسبه برای تمام جداولی که از همسایه ها می رسند مسیریاب می تواند نتیجه بگیرد که کدام یک از مسیرها بهتر هستند و مقدار تخمینی تأخیر و همچنین خط متناظر با بهترین مسیر را در جدول جدید خود درج می نماید. پخاطر داشته که جدول قدیمی هر مسیریاب در محاسبات دخالت داده نمی شود.

فرآیند بهنگام سازی، در شکل ۵-۹ به تصویر کشیده شده است. قسمت «الف» از شکل، ساختار یک زیر شبکه را نشان می دهد. چهار ستون سمت چپ در بخش «ب» شکل، «بردار تأخیر» چهار همسایه مسیریاب J است که در لحظه بهنگام سازی دریافت شده است. مسیریاب A ادعا کرده که تا B دوازده میلی ثانیه تأخیر دارد؛ ۲۵ میلی ثانیه تأخیر تا C ، ۴۰ میلی ثانیه تأخیر تا D و به همین ترتیب. فرض کنید مسیریاب J تأخیر خود را تا A و H و K به ترتیب مقادیر ۸، ۱۰، ۱۲ و ۱۶ اندازه گیری کرده یا تخمین زده باشد.

حال ببینیم مسیریاب J چگونه مسیرهای جدید برای رسیدن به مسیریاب G را محاسبه می کند. J می داند که برای رسیدن به A ، ۸ میلی ثانیه تأخیر خواهد داشت و A نیز ادعا کرده که قادر است با تأخیر ۱۸ میلی ثانیه به G برسد، لذا J متوجه می شود که اگر بسته های خود را که به مقصد G روانه هستند برای A بفرستد کل تأخیر حدود ۲۶ میلی ثانیه $[۱۸ + ۸]$ خواهد بود. به همین طریق، J تأخیر خود تا G از طریق H و K را به ترتیب ۴۱ $(۳۱ + ۱۰)$ ، ۱۸ $(۶ + ۱۲)$ و ۳۷ $(۳۱ + ۶)$ میلی ثانیه محاسبه می نماید. از بین این مقادیر، ۱۸ بهترین است لذا J در جدول مسیریابی خود، درایه ای برای G تشکیل می دهد و در آن درج می کند که تأخیر رسیدن به G ، ۱۸ میلی ثانیه است و مسیر مربوطه از طریق H می گذرد. [یعنی از آن به بعد هر بسته ای که به J وارد شود و بخواید به G برود به سمت



تخمین تأخیر دیگر مسیریابها تا J

To	A	I	H	K	Line
A	0	24	20	21	8 A
B	12	36	31	28	20 A
C	25	18	19	36	28 I
D	40	27	8	24	20 H
E	14	7	30	22	17 I
F	23	20	19	40	30 I
G	18	31	6	31	18 H
H	17	20	0	19	12 H
I	21	0	14	22	10 I
J	9	11	7	10	0 -
K	24	22	22	0	6 K
L	29	33	9	9	15 K

تأخیر خط	تأخیر خط	تأخیر خط	تأخیر خط	جدول مسیریابی جدید در J
JA	JI	JH	JK	
8	10	12	6	

بردارهای دریافتی از چهار همسایه J

(الف)

(ب)

شکل ۹-۵. (الف) یک زیر شبکه (ب) جداول دریافتی از A، H، I و K و جدول مسیریابی جدید در J.

مسیریاب H ارسال خواهد شد و تأخیر تخمینی کل مسیر، ۱۸ میلی ثانیه خواهد بود. -م
همین محاسبه برای دیگر مسیریابهای مقصد نیز انجام می شود و نهایتاً جدول مسیریابی جدید (نشان داده شده در سمت راست شکل ۹-۵) بدست خواهد آمد.

مشکل شمارش تا بی نهایت (Count-To-Infinity Problem)

مسیریابی بردار فاصله از دیدگاه تئوری به درستی کار می کند ولیکن در عمل اشکالات جدی خواهد داشت: اگرچه نهایتاً به جواب صحیح همگرا خواهد شد ولی این همگرایی بسیار کند خواهد بود.^۱ بالاخص، این الگوریتم به «خبرهای خوب» واکنش سریع نشان می دهد در حالی که «خبرهای بد» را به آهستگی منتقل می کند.^۲ یک مسیریاب را در نظر بگیرید که در جدول او تأخیر رسیدن به مقصد X، مقدار بسیار بالا، محاسبه و درج شده است. اگر در مرحله بعدی مبادله جداول (یعنی در لحظه بهنگام سازی)، مسیریاب همسایه او (مثلاً A)، هزینه کمتری را برای رسیدن به X اعلام کند، او بلافاصله مسیر قبلی خود را تغییر داده و از آن به بعد مسیر ارسال ترافیک به X را از طریق A انتخاب خواهد کرد. در هر بار مبادله بردارهای هزینه، «خبرهای خوب» به سرعت پردازش می شوند. برای آنکه ببینیم چگونه اخبار خوب منتشر می شوند به زیر شبکه پنج گره ای (خطی) شکل ۱۰-۵ دقت کنید که در آن معیار تأخیر «تعداد گام» است. فرض کنید A در همان ابتدا از کار افتاده و تمام مسیریابها از این موضوع آگاه هستند. به عبارت دیگر تمام مسیریابها تأخیر خود تا A را به مقدار بی نهایت تنظیم کرده اند. وقتی A به کار می افتد دیگر مسیریابها با مبادله بردار [جدول هزینه] بلافاصله از این موضوع آگاه خواهند شد. برای سادگی فرض کرده ایم یک ناقوس بزرگ در جایی وجود دارد که بطور متناوب به صدا در می آید تا همه

۱. منظور از همگرایی رسیدن به یک جدول مسیریابی با مقادیر درست، واقعی و پایدار است. -م

۲. منظور از خبرهای خوب کاهش تأخیر در مسیرها، اضافه شدن لینک یا مسیریاب جدید و منظور از خبرهای بد خرابی یک خط، از کار افتادن یک مسیریاب یا افزایش تأخیر است. -م

A	B	C	D	E		A	B	C	D	E	
•	•	•	•	•	Initially در بدو شروع	•	•	•	•	•	Initially در بدو شروع
	1	•	•	•	پس از اولین مبادله جدول	1	2	3	4	4	پس از اولین مبادله جدول
	1	2	•	•	پس از دومین مبادله جدول	3	4	3	4	4	پس از دومین مبادله جدول
	1	2	3	•	پس از سومین مبادله جدول	5	4	5	4	4	سومین مبادله جدول
	1	2	3	4	پس از چهارمین مبادله جدول	5	6	5	6	6	چهارمین مبادله جدول
						7	6	7	6	6	پنجمین مبادله جدول
						7	8	7	8	8	ششمین مبادله جدول
						•	•	•	•	•	

(الف)

(ب)

شکل ۵-۱۰. مشکل «شمارش نایی نهایی».

مسیریابها بطور همزمان عملیات مبادله بردارها (جداول) را شروع نمایند!! در اولین مبادله، B آگاه می شود که همسایه سمت چپ او تأخیری معادل صفر به A دارد. B در جدول مسیریابی خود یک درایه (Entry) ایجاد و در آن تأخیر رسیدن به A را ۱ درج می نماید. هنوز بقیه مسیریابها گمان می کنند که A غیر فعال است. تأخیر رسیدن به A در درایه های جدول مسیریابی تمام مسیریابها (در این لحظه) در سطر دوم از شکل ۵-۱۰-الف مشاهده می شود. در مبادله بعدی C متوجه می شود که B مسیری به A با طول ۱ دارد و به همین دلیل جدول خود را بهنگام می کند و در آن مسیری به A از طریق B با طول ۲ را مشخص می نماید ولی هنوز D و E از این خبر خوب آگاه نیستند. در یک زیرشبکه، اخبار خوب در هر بار مبادله جداول، یک گام به جلو منتشر می شود. اگر در یک زیرشبکه، طولانی ترین مسیر N گام باشد پس از N بار مبادله، همه مسیریابها از اضافه شدن خطوط یا مسیریابهای جدید به زیرشبکه (خبر خوب)، آگاه خواهند شد.

حال به وضعیت شکل ۵-۱۰-ب نگاه کنید که در آن تمام خطوط و مسیریابها در ابتدا فعال هستند. فاصله مسیریابهای A، B، C، D و E تا A به ترتیب عبارتند از: ۱، ۲، ۳ و ۴. به ناگاه مسیریاب A از کار می افتد یا مثلاً خط بین A و B قطع می شود که هر دوی این حالات از دیدگاه B فرقی نمی کنند.

پس از اولین مبادله بسته، B از A چیزی نمی شنود. خوشبختانه C می گوید: «نگران نباش! من مسیری به A به طول ۲ دارم». B به درستی نمی داند که مسیری که C اعلام کرده از خود او (یعنی B) می گذرد. آنچه که B حدس زده آنست که شاید C ده خط دیگر دارد و می توان از طریق این خطوط با هزینه ۲، به A رسید. در نتیجه، B گمان می کند که قادر است از طریق C با هزینه ۳ به A برسد! D و E درایه های جدول مسیریابی خود را در اولین مبادله تغییر نمی دهند.

در دومین مبادله، C متوجه می شود که هر یک از همسایه های او ادعا کرده اند که مسیری به A با هزینه ۳ دارند لذا یکی از آنها را انتخاب کرده و فاصله جدید رسیدن به A را به ۴ تنظیم می کند. (به گونه ای که در سطر سوم از شکل ۵-۱۰-ب نشان داده شده است). مبادلات بعدی جداول، نتایج نشان داده شده در ادامه شکل ۵-۱۰-ب را تولید خواهد کرد.

از این شکل باید روشن شده باشد که چرا خبرهای بد به کندی منتشر می شوند: همه مسیریابها هزینه جدید را یک واحد بیش از نتیجه مینیمم گیری از هزینه های اعلام شده، در نظر می گیرند. [به عبارت دیگر هزینه های جدید براساس مقادیری محاسبه می شود که قدیمی و اشتباه است. -م] به تدریج مقدار فاصله درج شده در جدول تمام مسیریابها به سمت بی نهایت رشد می کند ولیکن تعداد دفعات مبادله جداول به عددی که برای مقدار «بی نهایت» در

نظر گرفته شده بستگی دارد. به همین دلیل بهترین کار آن است که مقدار بی نهایت، معادل طول بزرگترین مسیر در زیر شبکه به اضافه ۱، در نظر گرفته شود. اگر معیار هزینه، «زمان تأخیر» باشد هیچ حد بالایی را نمی توان تعریف کرد مگر آنکه حد بالا را آنقدر زیاد فرض کنیم که با مسیری که بطور طبیعی تأخیر آن بالاست به عنوان مسیر از کار افتاده و خراب رفتار نشود. مشکل فوق الذکر به نام «مشکل شمارش تا بی نهایت» مشهور است. تلاشهایی برای حل این مشکل انجام گرفته (مثل روش split horizon with poisoned reverse که در RFC 1058 تشریح شده) ولی هیچکدام از آنها موفق عمل نکرده اند. اصل این مشکل از آنجایی ناشی شده که وقتی X به Y می گوید که مسیری به جایی دارد، Y بهیچوجه نمی تواند بفهمد که آیا خودش بر روی این مسیر قرار گرفته یا نه!

۵.۲.۵ مسیریابی حالت لینک (Link State Routing)

از «مسیریابی بردار فاصله» تا سال ۱۹۷۹ در ARPANET استفاده می شد و در همین ایام به «مسیریابی حالت لینک» (Link State) تغییر کرد. دو مشکل اساسی منجر به زوال آن شد: اول آنکه در این الگوریتم معیار تأخیر، طول صف^۱ در نظر گرفته می شد و پهنای باند هر یک از خطوط در محاسبات انتخاب بهترین مسیر، دخالت داده نمی شد. در ابتدا تمام خطوط 56Kbps بودند لذا پهنای باند خط مورد مهمی نبود ولی پس از آنکه برخی از خطوط به سرعت 230 Kbps و برخی دیگر به 44 Mbps ارتقاء یافتند، بحساب نیاوردن پهنای باند، مشکلی عمده به حساب می آمد. البته این امکان وجود داشت که بتوان معیار تأخیر را به پهنای باند خط تغییر داد ولیکن مشکل دیگری ایجاد می شد و آن هم اینکه همگرایی الگوریتم بسیار طولانی می شد (بدلیل مشکل شمارش تا بی نهایت). به همین دلایل با الگوریتم کاملاً جدیدی که اکنون «مسیریابی حالت لینک» (LS) نامیده می شود، تعویض شد. امروزه گونه های متفاوتی از «مسیریابی حالت لینک» مورد استفاده قرار می گیرد. ایده اصلی و زیربنای مسیریابی حالت لینک (LS)، ساده است و می توان آن را در پنج بند بیان کرد. هر مسیریاب باید به ترتیب زیر عمل کند:

۱. همسایه های خود را شناسایی کرده و آدرسهای شبکه آنها را بدست بیاورد.
۲. تأخیر یا هزینه هر یک از همسایه های خود را اندازه گیری نماید.
۳. پستهای بسازد و اطلاعاتی که از همسایه ها کسب کرده، در آن جاسازی کند.
۴. این پسته را برای تمام مسیریابهای دیگر بفرستد.
۵. کوتاهترین مسیر رسیدن به دیگر مسیریابها را محاسبه نماید.

بدین ترتیب، توپولوژی کامل زیر شبکه و تمام تأخیرها (از طریق آزمایش) اندازه گیری شده و بین تمام مسیریابها توزیع می شود. برای پیدا کردن کوتاهترین مسیرها به تمام مسیریابهای زیر شبکه، می توان از «الگوریتم دایکسترا» بهره گرفت. در ادامه هر یک از این پنج مرحله را به تفصیل بررسی خواهیم نمود.

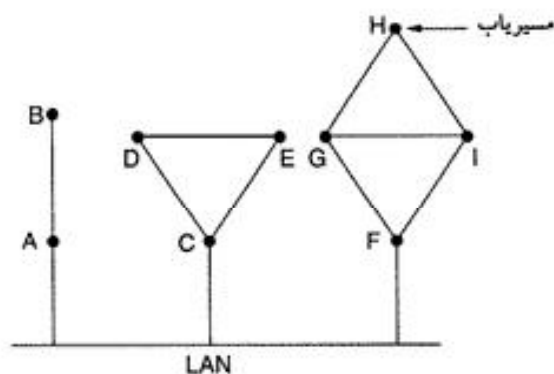
شناسایی همسایه ها

وقتی یک مسیریاب آغاز به کار می کند [بوت می شود] اولین وظیفه او شناسایی همسایه های خودش است. این کار با ارسال یک پسته خاص به نام «پسته سلام» (HELLO Packet) بر روی تمام خطوط نقطه به نقطه انجام می شود. انتظار می رود که مسیریابهایی که در طرف مقابل هر خط هستند پاسخی برگردانده و خود را معرفی کنند. این نامها [یا به عبارت بهتر آدرس مسیریابها] باید جهانی، منحصر به فرد و یکتا باشند چرا که بعداً وقتی یک مسیریاب راه دور می شنود که سه مسیریاب همگی مثلاً به F متصلند، دانستن این موضوع که آیا هر سه F یکی هستند بسیار

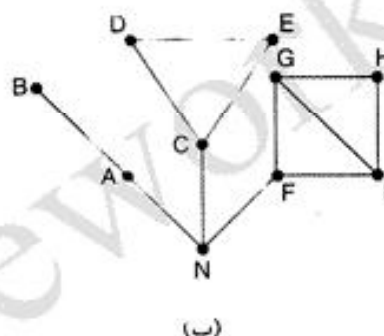
۱. تعداد پسته های به صف شده منتظر ارسال بر روی یکی از خطوط خروجی مسیریاب، به طول صف تعبیر می شود. -م

حیاتی است.

وقتی دو یا چند مسیر یاب از طریق LAN به هم متصل شده باشند وضعیت اندکی پیچیده تر است. شکل ۵-۱۱ الف یک شبکه LAN را نشان می دهد که سه مسیر یاب A، C و F مستقیماً بدان متصل شده اند. بگونه ای که نشان داده شده این مسیر یابها به یک یا چند مسیر یاب دیگر نیز متصلند. یک روش برای مدلسازی LAN آنست که همانند شکل ۵-۱۱ ب، آن را به عنوان یک گره در نظر بگیریم. در اینجا یک گره مصنوعی جدید به نام N معرفی کرده ایم که A و C و F بدانها متصل هستند. این واقعیت که رسیدن از A به C فقط از طریق LAN ممکن است با مسیر ANC بیان می شود.



(الف)



(ب)

شکل ۵-۱۱. (الف) نه مسیر یاب و یک شبکه LAN (ب) مدل گراف از شکل الف.

اندازه گیری هزینه خط

«الگوریتم حالت لینک» نیاز مند آن است که هر مسیر یاب از تأخیر هر یک از همسایه های خود آگاه باشد (یا حداقل تخمینی معقول از تأخیر آنها داشته باشد). راه مستقیم اندازه گیری این تأخیر آن است که یک بسته خاص به نام Echo بر روی خط مورد نظر ارسال شده و طرف مقابل موظف به برگرداندن فوری آن بسته باشد. با اندازه گیری زمان رفت و برگشت این بسته و تقسیم آن بر ۲، مسیر یاب فرستنده می تواند تخمینی معقول از تأخیر بدست بیاورد. برای بدست آوردن نتایج بهتر می توان این آزمون را چندین بار تکرار کرد و میانگین مقادیر اندازه گیری شده را در نظر گرفت. البته در این روش تلویحاً فرض بر آن گذاشته شده که تأخیرها متقارن^۱ است در حالی که ممکن است همیشه اینگونه نباشد.

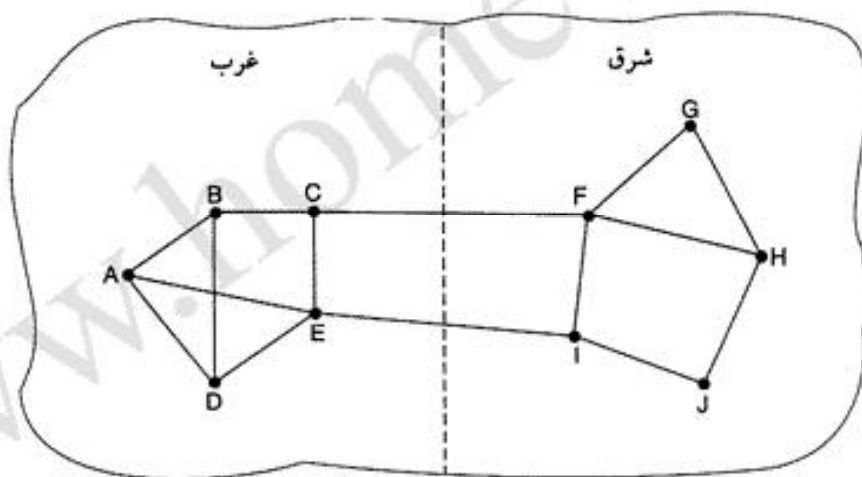
یک مسئله جالب آن است که آیا باید در اندازه گیری تأخیر، «میزان بار» (load) را نیز به حساب آورد؟ برای وارد کردن «میزان بار» در محاسبات، «تایمر سنجش زمان رفت و برگشت» (Round-Trip Timer) باید زمانی آغاز به کار کند که بسته Echo به انتهای صف ارسال وارد می شود. برای نادیده گرفتن میزان بار، تایمر زمانی شروع به اندازه گیری می کند که بسته Echo به سر صف رسیده باشد.

می توان در خصوص هر دوی این روشها بحث کرد. در نظر گرفتن تأخیرات ناشی از ترافیک در محاسبات، بدین معنی است وقتی یک مسیر یاب می خواهد از بین دو خط با پهنای باند مساوی یکی را انتخاب کند (در شرایطی که یکی از آنها برخلاف دیگری با بار سنگین مواجه است)، مسیر یاب خطی را که فاقد بار است به عنوان

۱. متقارن بودن تأخیر بدین معناست که تأخیر رسیدن از A به B یا تأخیر رسیدن از B به A یکسان است، در حالیکه در بسیاری از محیطها اینگونه نیست. -م

مسیر کوتاهتر در نظر می‌گیرد. چنین انتخابی، کارایی بهتری را در بر-واحد داشت. متأسفانه استدلالی بر علیه نظریهٔ «دخالت دادن میزان بار در مسیر تأخیر» وجود دارد: به زیرشبکه شکل ۱۲-۵ توجه کنید که در آن زیرشبکه به دو بخش شرقی و غربی تقسیم و توسط دو خط CF و EI بهم متصل شده‌اند.

فرض کنید که بیشتر حجم ترافیک بین شرق و غرب از طریق خط CF مبادله شود و در نتیجه این خط با بار سنگین و تأخیر بالایی مواجه است. در نظر گرفتن تأخیر انتظار (یعنی تأخیر صف) در محاسبات کوتاهترین مسیر باعث می‌شود که خط EI جلب توجه نماید. پس از آنکه جدول جدید مسیریابی تنظیم و اعمال شود خط EI به عنوان خط بهینه انتخاب شده و از آن به بعد بخش اعظم ترافیک شرق به غرب از طریق EI عبور خواهد کرد و طبعاً این خط با بار سنگین مواجه خواهد شد. طبعاً در بهنگام‌سازی بعدی، مجدداً خط CF به عنوان مسیر بهینه شناخته می‌شود. در نتیجه ممکن است جدول مسیریابی بطور نوسانی تغییر کرده و منجر به مسیریابی نامنظم شده و اشکالات بالقوهٔ فراوانی تولید کند. اگر از میزان بار چشمپوشی شده و فقط پهنای باند در نظر گرفته شود این مشکل رخ نخواهد داد. البته می‌توان بار را بر روی هر دو خط توزیع کرد ولیکن در این راه حل از مسیر بهینه، استفادهٔ کامل نخواهد شد. علیرغم این، برای اجتناب از بروز تغییرات نوسانی در انتخاب بهترین مسیر، شاید توزیع بار بر روی چند خط راهکاری معقول باشد (البته باید کسری از بار که بر روی هر خط توزیع می‌شود از قبل تعیین شده باشد).

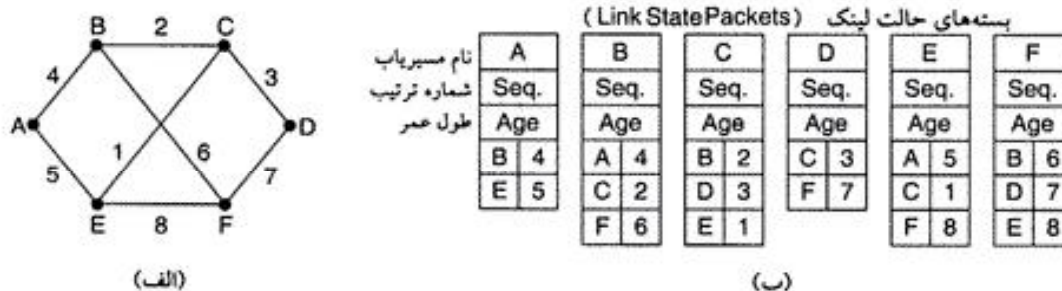


شکل ۱۲-۵. زیرشبکه‌ای که در آن دو بخش شرقی و غربی با دو خط به یکدیگر متصل شده‌اند.

ساخت بسته‌های حالت لینک (Building Link State Packet)

به محض آنکه اطلاعات لازم جهت مبادله گردآوری شد، گام بعدی هر مسیریاب، ساختن بسته‌ای است که این داده‌ها را در بر بگیرد.^۱ در ابتدای هر بسته، هویت فرستندهٔ آن درج می‌شود، سپس فیلدهای «شماره ترتیب» و «طول عمر» (Age) بسته می‌آید (در مورد طول عمر بسته در ادامه توضیح خواهیم داد)؛ در آخر نیز فهرست همسایه‌ها و تأخیر رسیدن بدانها، مشخص می‌شود. در شکل ۱۳-۵ الف یک زیرشبکه نمونه نشان داده شده که در آن، تأخیرها به صورت پرچسب عددی، بر روی هر خط مشخص شده‌اند. بسته‌های LS ایجاد شده در هر یک

۱. «بسته‌های حالت لینک» را یک ساختمان دادهٔ استاندارد همانند یک استراکچر در زبان C در نظر بگیرید. از این به بعد این بسته‌ها را بسته‌های LS می‌نامیم. -م



شکل ۵-۱۳. (الف) یک زیر شبکه (ب) بسته های حالت لینک (LS) برای این زیر شبکه.

از شش مسیر یاب، در شکل ۵-۱۳ ب مشاهده می شود.

ساخت بسته های LS ساده است. مشکل ترین بخش قضیه تعیین زمانی است که مسیر یابها باید به ساختن این بسته ها اقدام کنند. یک راه آن است که بطور متناوب و در فواصل زمانی مشخص ایجاد شوند. راه دیگر آن است که مسیر یاب فقط زمانی مبادرت به ساخت این بسته ها کند که اتفاق خاصی رخ بدهد، مثلاً یکی از خطوط یا مسیر یابهای همسایه از کار بیفتد یا مجدداً فعال شود یا ویژگیهای آن بطور محسوسی تغییر کند. [منظور از ویژگی، میزان تأخیر، بار، پهنای باند یا نظایر آنهاست. -م]

توزیع بسته های «حالت لینک»

ظریف ترین بخش این الگوریتم، توزیع مطمئن بسته های LS است. به محض آنکه بسته ها، توزیع و در جداول اعمال شدند، آن مسیر یاب که اول از همه این بسته ها را دریافت کند مسیرهای خود را تغییر می دهد [در حالیکه ممکن است بقیه هنوز چنین کاری نکرده باشند]. در نتیجه مسیر یابهای متفاوت ممکن است برای لحظاتی از نسخه های متفاوتی از توپولوژی زیر شبکه استفاده کنند که این مسئله منجر به مشکلاتی در مسیر یابی صحیح، بروز حلقه بینهایت، از دسترس خارج شدن برخی از ماشینها و مشکلات دیگر شود.

در ابتدا به تشریح الگوریتم اساسی توزیع می پردازیم و در ادامه اصلاحاتی را بر روی آن انجام خواهیم داد. ایده اصلی آن است که برای توزیع بسته های LS از روش «سیل آسا» استفاده شود. برای آنکه کنترل این فرآیند را در دست داشته باشیم هر بسته دارای شماره ترتیب است؛ به ازای تولید هر بسته جدید، به این شماره ترتیب یک واحد اضافه می شود. هرگاه مسیر یابها بسته ای از این نوع را دریافت کنند زوج آیتیم «آدرس مسیر یاب مبدا، شماره ترتیب» آنرا در جایی ذخیره می کنند. وقتی یک بسته LS جدید وارد شود ابتدا بررسی می شود که آیا این بسته قبلاً نیز دریافت شده است. اگر بسته جدید بود بر روی تمام خطوط خروجی (به استثنای خطی که از روی آن دریافت شده) ارسال می شود ولیکن اگر تکراری بود نادیده گرفته می شود. اگر بسته ای دریافت شود و شماره ترتیب آن از بزرگترین شماره ای که تاکنون دریافت شده، کوچکتر باشد به عنوان بسته قدیمی تلقی و در نظر گرفته نخواهد شد چرا که مسیر یاب، نسخه جدیدتری از آن را در اختیار دارد.

این الگوریتم چند مشکل دارد ولی این مشکلات قابل مدیریت و کنترل هستند: اول آنکه اگر شماره ترتیب با افزایشهای متوالی، در جایی به صفر برگردد مشکلاتی بروز خواهد کرد. راه حل آن است که از یک شماره ترتیب ۳۲ بیتی استفاده شود؛ در این صورت اگر در هر ثانیه یک بسته LS ارسال شود، ۱۳۷ سال طول می کشد تا این عدد به صفر برگردد لذا احتمال بروز چنین مشکلی قابل چشمپوشی است!

دوم آنکه اگر یکی از مسیر یابها از کار بیفتد روند شماره ترتیب بسته های خود را از دست خواهد داد و اگر بعد از راه اندازی مجدد، شماره ترتیب را از صفر شروع کند بسته های ارسالی او تکراری تلقی شده و

در نظر گرفته نمی شود.

سوم آنکه اگر شماره ترتیب به نحوی دچار خطا شود و مثلاً در اثر یک بیت خطا شماره ۴ به ۶۵۵۴۰ تبدیل شود از آن به بعد بسته های ۵ تا ۶۵۵۴۰ به عنوان بسته قدیمی دور انداخته خواهد شد چرا که گمان می رود که شماره فعلی ۶۵۵۴۰ است.

راه حل تمام این مشکلات آن است که پس از شماره ترتیب، طول عمر بسته نیز درج گردد و با گذشت هر ثانیه یک واحد از آن کم شود. وقتی عمر بسته به صفر برسد اطلاعاتی که از آن مسیریاب دریافت شده باید حذف گردد. در یک روال طبیعی (مثلاً هر ده ثانیه یکبار)، بسته های جدید LS وارد مسیریاب شده و طبقاً اطلاعات قبلی بطور منظم و قبل از انقضای مهلت اعتبارشان تازه سازی می شوند؛ بدین ترتیب اعتبار اطلاعات مربوط به هر مسیریاب زمانی منقضی خواهد شد که آن مسیریاب از کار بیفتد (یا آنکه مثلاً در اثر هر گونه رخداد، متوالیاً شش بسته LS از آن مسیریاب دریافت نشود). فیلد «طول عمر» (Age) در خلال فرآیند ارسال سیل آسا و توسط هر مسیریاب نیز یک واحد کاهش می یابد تا مطمئن شویم که هیچ بسته ای نمی تواند بطور نامحدودی زنده و در زیر شبکه سرگردان بماند. (هر گاه عمر بسته صفر شود، حذف خواهد شد.)

چند اصلاح در این الگوریتم، آنرا قدرتمندتر خواهد کرد. وقتی یک بسته LS برای ارسال به روش سیل آسا وارد یک مسیریاب می شود بلافاصله در صف ارسال وارد نخواهد شد بلکه ابتدا به یک «فضای انتظار» وارد می شود تا برای مدتی کوتاه، منتظر بماند. اگر قبل از ارسال بسته فعلی، بسته مشابه دیگری از همان مبدا وارد شود، شماره ترتیب این دو بسته مقایسه شده و در صورت مساوی بودن، بسته تکراری حذف خواهد شد و در صورت تکراری نبودن از بسته قدیمی تر صرف نظر می شود. برای پیشگیری از خطاهای احتمالی بر روی خطوط مستقیم بین دو مسیریاب، دریافت تمام بسته های LS ورودی، تصدیق (Ack) خواهد شد.^۱ وقتی یکی از خطوط خروجی مسیریاب آزاد گردد «فضای انتظار» به روش Round Robin (نوبت چرخشی) پویش می شود تا یک بسته LS یا یک بسته اعلام وصول (یعنی Ack) برای ارسال انتخاب گردد.

در جدول شکل ۵-۱۴ ساختمان داده مورد استفاده در مسیریاب B که برای زیر شبکه شکل ۵-۱۳ الف ایجاد شده، دیده می شود. در این جدول، هر سطر متناظر با یک بسته LS تازه وارد است که هنوز بطور کامل پردازش نشده است. در این جدول مبدا هر بسته، شماره ترتیب، طول عمر آن و مقداری داده ثبت می شود. بعلاوه به ازای هر یک از سه خط متصل به B (یعنی خطوط A و C و F) دو پرچم «ارسال» و «اعلام وصول»^۲ در نظر گرفته شده است. «پرچم ارسال» بدین معناست که بسته باید بر روی خط مشخص شده، ارسال شود. «پرچم اعلام وصول» بدین معناست که دریافت این بسته باید بر روی خط مربوطه، به آگاهی طرف مقابل برسد.

از جدول شکل ۵-۱۴، مشخص است که یک بسته LS مستقیماً از A دریافت شده و طبق آنچه که بیت های پرچم نشان می دهند این بسته باید برای C و F ارسال شود و ضمناً وصول آن به A اعلام گردد. به طور مشابه بسته ای که از F آمده باید به A و C نیز ارسال شده و دریافت آن به F اعلام گردد.

با این حال شرایط بسته سوم که از E می رسد متفاوت است. این بسته دو بار، یکبار از طریق EAB و یکبار از طریق EFB دریافت شده است. در نتیجه بنحوی که بیت های پرچم نشان می دهند، این بسته باید برای C ارسال شود ولیکن فقط کافی است وصول آن به A و F اعلام گردد.

اگر بسته ای تکراری دریافت شود در حالی که نسخه اصلی آن هنوز در بافر موجود است، بیت های پرچم باید تغییر کنند. به عنوان مثال اگر بسته متعلق به C قبل از آنکه مطابق با درایه چهارم (4th Entry) برای A و F ارسال

۱. دریافت هر بسته LS به مسیریاب همسایه (که آنرا فرستاده) اعلام خواهد شد نه مبدا آن. سم

۲. Send Flag and Acknowledgement Flag.

Source (مبدأ)	Seq.	Age	Send flags			ACK flags			Data
			A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

شکل ۵-۱۴. بافر بسته های LS برای مسیر یاب B در شکل ۵-۱۳.

شود یکبار دیگر از طریق F دریافت گردد، شش بیت پرچم به 100011 تغییر خواهد کرد بدین معنا که باید وصول بسته به F اعلام شود ولی لازم نیست خود بسته برای F ارسال گردد.

محاسبه مسیرهای جدید

به محض آنکه مسیر یاب مجموعه کامل بسته های LS را گردآوری کرد می تواند گراف کل زیر شبکه را تشکیل بدهد چرا که همه لینکها و هزینه آنها مشخص شده است. هر لینک در حقیقت دو بار معرفی شده است، یکبار در هر جهت. [چرا که مثلاً لینک بین A و B، یکبار توسط A و یکبار توسط B شناسایی و معرفی می شوند. -م] حال می توان الگوریتم دایکسترا را به صورت محلی اجرا کرد و کوتاهترین مسیر ممکن به تمام نقاط مقصد در زیر شبکه را بدست آورد.^۱ نتیجه این الگوریتم می تواند در جدول مسیریابی درج شده و مسیر یابها عملکرد طبیعی خود را از سر بگیرند.

برای زیر شبکه ای با n مسیر یاب که هر کدام k همسایه دارند، حافظه لازم برای ذخیره داده های جدول مسیریابی، متناسب با nxk است که در زیر شبکه های بزرگ می تواند مشکل ایجاد کند. در ضمن زمان محاسبه، نیز می تواند مسئله ساز باشد ولیکن علیرغم همه اینها «الگوریتم مسیریابی حالت لینک» (Link State Routing) در محیطهای واقعی بخوبی کار می کند.

با این حال، هرگونه اشکال سخت افزاری یا نرم افزاری می تواند مشکلات بسیار جدی برای این الگوریتم به بار بیاورد. (همچنین در عملکرد الگوریتم دیگر مسیر یابها نیز مشکل ایجاد می کند.) به عنوان نمونه، اگر یک مسیر یاب ادعا کند که خطی در اختیار دارد در حالی که نداشته باشد یا فراموش کند خطی را که در اختیار دارد اعلام کند، گراف زیر شبکه صحیح نخواهد بود. همچنین اگر یک مسیر یاب در ارسال بسته های LS مشکل بهم بزند یا آنها را در حین ارسال خراب کند مشکلاتی جدی بروز خواهد کرد. سرانجام آنکه اگر حافظه مسیر یاب سرریز شود یا نتیجه محاسبات مسیر اشتباه باشد رخدادهای ناگواری در مسیریابی بسته ها بوقوع می پیوندد. هرگاه رشد زیر شبکه به دهها، صدها یا حتی هزاران گره برسد احتمال از کار افتادن ناگهانی یک یا چند مسیر یاب را نمی توان نادیده گرفت. تنها راه ممکن آن است که تلاش شود تا در هنگام بروز چنین رخدادهایی میزان آسیبها و مشکلات در حد محدودی کنترل شود. پرلمن (Perlmán, 1988) این مشکلات و راه حل های آنها را تشریح کرده است. روش «مسیریابی حالت لینک» بطور گسترده ای در شبکه های واقعی بکار گرفته شده است لذا مختصری در

۱. اجرای الگوریتم دایکسترا به صورت محلی بدین معناست که هر یک از مسیر یابها خودشان مستقل از دیگری آن را اجرا می کنند. -م

مورد چند پروتکل نمونه که از آن بهره گرفته اند خالی از لطف نیست. پروتکل OSPF که در اینترنت بسیار رایج است از همین الگوریتم استفاده می کند. در بخش ۴-۶-۵، OSPF را تشریح خواهیم کرد. یکی دیگر از «پروتکل های مبتنی بر حالت لینک»، پروتکل IS-IS است که توسط DECnet طراحی شد و بعداً توسط ISO جهت بکارگیری در کنار پروتکل بی اتصال لایه شبکه خود یعنی CLNP، مورد پذیرش و تأیید قرار گرفت. البته بعداً در آن تغییراتی ایجاد شد تا بتواند با پروتکل های دیگری مثل IP نیز کار کند. پروتکل IS-IS در بخشهایی از ستون فقرات شبکه اینترنت (مثلاً در ستون فقرات قدیمی متعلق به NSFNET) و در برخی از سیستم های دیجیتال سلولی مثل CDDP بکار رفته است. شرکت Novell Netware نیز گونه ساده تری از IS-IS را برای هدایت بسته های IPX خود بکار گرفته است.

IS-IS اساساً تصویری از توپولوژی مسیریاب را در زیر شبکه توزیع می کند و از طریق آن کوتاهترین مسیرها محاسبه می گردد. هر مسیریاب در هنگام اعلام اطلاعات مربوط به «حالت لینک» (LS) مشخص می کند که به چه آدرس هایی مستقیماً دسترسی دارد. (آدرس ها مربوط به لایه شبکه و جهانی هستند). این آدرس ها می توانند IP، IPX، AppleTalk یا هر آدرس دیگر باشند. IS-IS همچنین می تواند بطور همزمان از چندین پروتکل لایه شبکه پشتیبانی نماید.

بسیاری از نوآوری هایی که در IS-IS طراحی شده بود بعداً مورد پذیرش و استفاده OSPF قرار گرفت. (OSPF چندین سال پس از IS-IS ابداع شد). برخی از اینها عبارت بودند از روشی برای خاتمه خودکار فرآیند ارسال بسته ها به روش سیل آسا، مفهوم «مسیریاب برگزیده» (Designated Router) در شبکه LAN و روش محاسبه و پشتیبانی از تقسیم مسیر و همچنین تعریف معیارهای چندگانه هزینه. در نتیجه اختلاف ناچیزی بین IS-IS و OSPF وجود دارد. مهمترین اختلاف این دو آن است که IS-IS به گونه ای بسته های LS را تعریف و کد کرده است که بسادگی و بطور همزمان می تواند اطلاعاتی در خصوص چندین پروتکل لایه شبکه با خود حمل کند، خصوصیتی که OSPF فاقد آن است. این حسن در محیط های بزرگ که با چندین پروتکل مختلف کار می کنند بسیار ارزشمند است.

۶-۲-۵ مسیریابی سلسله مراتبی (Hierarchical Routing)

با رشد اندازه شبکه، جداول مسیریابی به همان نسبت رشد می کنند. جداول مسیریابی رو به رشد، نه تنها حافظه بیشتری مصرف می کنند بلکه به زمان CPU بیشتری برای پویش و جستجوی این جدول و همچنین پهنای باند زیادتری برای ارسال بسته های LS (بسته های گزارش وضعیت لینک) نیاز خواهد بود. ممکن است رشد شبکه بدان حد برسد که دیگر امکان نگهداری یک «درایه» (Entry) به ازای هر مسیریاب، در جدول مسیریابی وجود نداشته باشد و مسیریاب مجبور به مسیریابی سلسله مراتبی (همانند شبکه تلفن) شود.

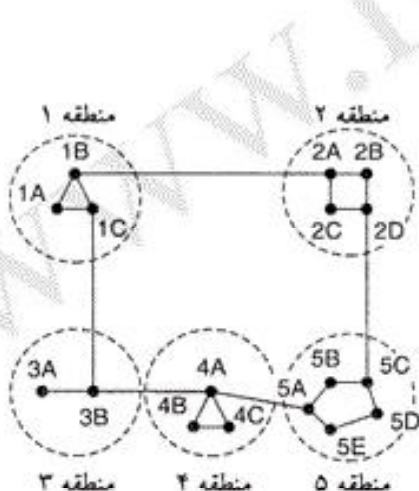
وقتی از مسیریابی سلسله مراتبی استفاده می شود، مسیریابها به تعدادی «منطقه» (Region) تقسیم می شوند؛ هر مسیریاب تمام جزئیات منطقه خود و مسیرهای دقیق رسیدن به هر مقصد در منطقه خود را می داند ولی چیزی در خصوص ساختار داخلی مناطق دیگر نمی داند. وقتی شبکه های مختلف بهم متصل می شوند طبیعی است که هر یک از آنها را به عنوان مناطق مجزا در نظر بگیریم تا مسیریابهای درون یک منطقه از دانستن ساختار توپولوژی ناحیه دیگر بی نیاز و فارغ باشند.

در شبکه های عظیم ممکن است سلسله مراتب دو سطحی کفایت نکند. ممکن است نیاز باشد که هر «منطقه» (Region) به تعدادی «دسته» (Cluster)، هر «دسته» به تعدادی «ناحیه» (Zone) و هر ناحیه به تعدادی «گروه»

(Group) تقسیم شود و به همین ترتیب ادامه یابد تا جایی که برای تقسیم بندی بیشتر اسمی باقی نماند!! به عنوان مثالی از سلسله مراتب چند سطحی، به چگونگی هدایت یک بسته از برکلی کالیفرنیا به مالدیدی در کنیا دقت کنید. مسیر یاب واقع در برکلی جزئیات توپولوژی زیر شبکه خود در کالیفرنیا را می داند ولیکن تمام ترافیک متعلق به خارج از ایالت کالیفرنیا را به مسیر یاب لوس آنجلس می فرستد. مسیر یاب واقع در لوس آنجلس تنها قادر به مسیریابی ترافیک داده ها بین مسیر یابهای داخل کشور است و هر گونه ترافیک خارجی را به نیویورک می فرستد. مسیر یاب واقع در نیویورک به گونه ای برنامه ریزی شده که این بسته را به سمت مسیریابی که در کشور مقصد مسئول دریافت ترافیک خارجی است (مثلاً مسیر یاب نایروبی) بفرستد. نهایتاً این بسته در کنیا به همین سیاق مسیر خود را ادامه می دهد تا به مالدیدی برسد.

در شکل ۵-۱۵ یک نمونه کمی از مسیریابی سلسله مراتبی دو سطحی با پنج منطقه را ارائه کرده است. مطابق با شکل ۵-۱۵ جدول کامل مسیریابی در مسیر یاب 1A دارای ۱۷ «درایه» است. وقتی مسیریابی به صورت سلسله مراتبی انجام می شود به ازای هر مسیر یاب محلی یک درایه در جدول مسیریابی درج می شود ولیکن مناطق دیگر فقط در یک مسیر یاب خلاصه می شوند لذا کل ترافیک مربوط به منطقه ۲ از خط 1B-2A عبور خواهد کرد و مابقی ترافیک خارجی از طریق 1C-3B هدایت خواهد شد. مسیریابی سلسله مراتبی، تعداد درایه های جدول مسیریابی 1A را از ۱۷ به ۷ تا کاهش می دهد. [۳] تا برای مسیر یابهای داخل منطقه و ۴ تا به ازای مناطق دیگر] هر چه نسبت تعداد مناطق به تعداد مسیر یابهای درون هر منطقه افزایش یابد میزان صرفه جویی در فضای جدول بیشتر خواهد بود.

متأسفانه صرفه جویی در فضای حافظه رایگان بدست نمی آید. بهایی که باید برای آن پرداخت، افزایش طول مسیرها [یا به عبارتی عدم بهینگی کامل مسیرها] است. به عنوان مثال بهترین مسیر از 1A به 5C از منطقه ۲ می گذرد در حالی که در مسیریابی سلسله مراتبی مسیر تمام ترافیک متعلق به ناحیه ۵ از منطقه ۳ تعیین شده چرا که برای



(الف)

جدول کامل مسیریابی 1A

(گام) Hops Line Dest. (مقصد)

1A	-	-
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(ب)

جدول مسیریابی سلسله مراتبی 1A

(گام) Hops Line Dest.

1A	-	-
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(ج)

بیشتر مسیر یابیهای مقصد واقع در ناحیه ۵، مسیر عبوری از منطقه ۲ مناسبتر است. سؤال قابل توجه آن است که وقتی یک شبکه واحد، بشدت رشد می کند، سلسله مراتب باید چند سطحی باشد؟ به عنوان مثال زیر شبکه ای با ۷۲۰ مسیر یاب را در نظر بگیرید. اگر سلسله مراتب وجود نداشته باشد هر مسیر یاب در جدول مسیر یابی خود به ۷۲۰ درایه نیاز خواهد داشت. اگر زیر شبکه به ۲۴ منطقه و ۳۰ مسیر یاب در هر منطقه تقسیم شود، هر مسیر یاب به ۳۰ درایه برای مسیر یابیهای محلی خود به علاوه ۲۳ درایه برای مناطق خارجی، نیاز دارد و بدین ترتیب جدول جمعاً ۵۳ درایه خواهد داشت. اگر سلسله مراتب، سه سطحی انتخاب شده و جمعاً هشت «دسته» (Cluster)، هر دسته شامل ۹ ناحیه و در هر ناحیه ۱۰ مسیر یاب تعریف شده باشد، هر مسیر یاب در جدول خود به ۱۰ درایه برای مسیر یابیهای محلی خود، ۸ درایه برای مسیر یابی در نواحی داخل دسته خودش و ۷ درایه برای دسته های خارجی (Distant Cluster) یعنی جمعاً ۲۵ درایه نیاز دارد. دو پژوهشگر بنامهای Kamoun و Kleinrock (۱۹۷۹) پی بردند که بهترین تعداد سطوح سلسله مراتب در زیر شبکه ای با N مسیر یاب، معادل $\ln(N)$ است و هر مسیر یاب به جمعاً $e \cdot \ln(N)$ عدد درایه نیاز خواهد داشت. همچنین نشان دادند که افزایش طول مؤثر مسیرها یا به عبارت دیگر کاهش بهینگی مسیرها، که از مسیر یابی سلسله مراتبی منشاء می گیرد بقدر کافی کم و نتیجه معمولاً قابل قبول است.

۷-۲-۵ مسیر یابی فراگیر (Broadcast Routing)

در برخی از کاربردها، ماشینهای میزبان نیازمند ارسال پیام به همه ماشینهای شبکه یا تعدادی از آنها هستند. به عنوان مثال برای خدمات توزیع گزارشات هواشناسی، بهنگام سازی اطلاعات بازار سهام یا برنامه های زنده رادیویی، راهکار مناسبتر آنست که داده ها را بصورت پخش فراگیر برای تمام ماشینها ارسال کرده و آنها را آزاد بگذاریم تا در صورت تمایل بسته های داده را دریافت کرده و بخوانند. «ارسال همزمان یک بسته به تمام ماشینهای مقصد، اصطلاحاً پخش فراگیر (Broadcasting) نامیده می شود و راهکارهای متنوعی برای اجرای آن پیشنهاد شده است.» یکی از روشهای پخش فراگیر که به هیچ ویژگی خاصی از زیر شبکه نیاز ندارد آن است که ماشین مبدا هر بسته خود را بطور جداگانه برای یکایک ماشینهای مقصد بفرستد. این روش نه تنها پهنای باند را تلف خواهد کرد بلکه ماشین مبدا نیازمند آن است که فهرست کاملی از تمام ماشینهای مقصد در اختیار داشته باشد. اگرچه گاهی در عمل این روش تنها راه ممکن است ولیکن روش چندان مطلوبی نیست.

روش ارسال سیل آسا، نامزد دیگری برای پخش فراگیر محسوب می شود. هر چند روش سیل آسا، راهکار نامناسبی برای شبکه های نقطه به نقطه است ولیکن برای ارسال فراگیر می توان بر روی آن حساب باز کرد، خصوصاً وقتی که هیچیک از راهکارهایی که در ادامه معرفی می شوند عملی نباشد. استفاده از روش ارسال سیل آسا همان مشکلی را دارد که در الگوریتم مسیر یابی نقطه به نقطه به آن اشاره کردیم: یعنی بسته های بسیار زیاد تولید می شود و پهنای باند بسیار وسیعی مصرف و تباه خواهد شد.

الگوریتم سوم «مسیر یابی چند مقصدی» (Multidestination Routing) نام دارد. اگر از این روش استفاده شود هر بسته فهرستی از کلیه مقصدهای مورد نظر یا «نقشه ای بیتی» (Bitmap) از این مقصدها با خود حمل می کند. هر گاه بسته ای به یک مسیر یاب برسد، آن مسیر یاب فهرست مقصدهای بسته را بررسی می کند تا مجموعه خطوط خروجی منتهی به هر مقصد تعیین شوند. (هر خط خروجی که حداقل به یکی از مقاصد مورد نظر منتهی شود، انتخاب می گردد.) مسیر یاب به ازای هر یک از خطوط خروجی تعیین شده، یک نسخه جدید از آن بسته را تولید کرده و در آن بسته فقط آدرس مقصدهایی را قرار می دهد که مسیر آنها بر روی خط خروجی مربوطه است. [بقیه آدرسهای مقصد حذف می شوند چرا که برای دیگر مقصدها خط خروجی جدا انتخاب و نسخه ای جداگانه تولید می شود. -م] در نتیجه مجموعه آدرسهای مقصد هر بسته بر روی خطوط جداگانه تقسیم و توزیع می شود.

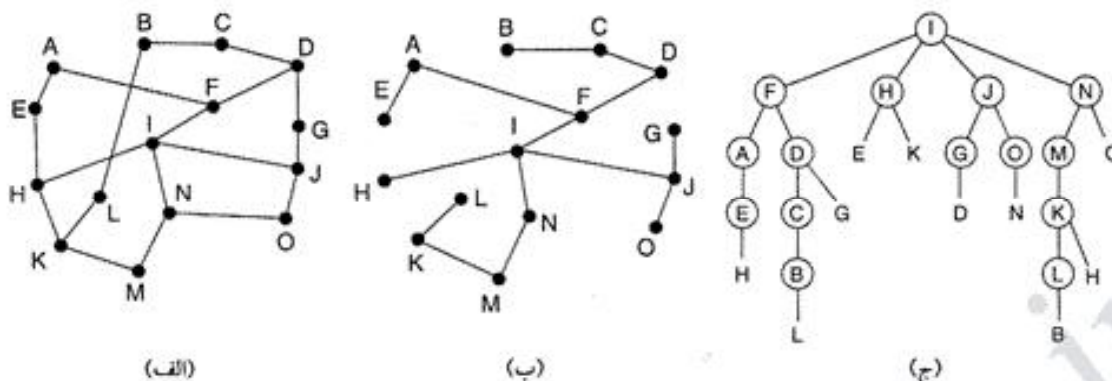
پس از چندین گام، هر بسته فقط یک آدرس مقصد با خود حمل می کند و با آن به عنوان یک بسته معمولی رفتار می شود. «مسیریابی چند مقصدی» همانند بسته های مستقل و با آدرس جداگانه عمل می کنند، با این تفاوت که بجای ارسال چندین بسته بر روی یک مسیر مشابه، فقط یک بسته ارسال و بدین نحو فقط هزینه ارسال یک بسته پرداخت می شود. [در این روش، هر بسته با رسیدن به یک مسیریاب به تعداد کافی تکثیر می شود. -م]

الگوریتم چهارم برای پخش فراگیر آن است که مسیریاب آغازگر این فرآیند، از درخت Sink Tree یا هر درخت پوشای مناسب (Spanning Tree) استفاده نماید. یک «درخت پوشا» زیر مجموعه ای از کل زیر شبکه است که تمام مسیریابها را در بر می گیرد ولیکن هیچ حلقه ای در آن نیست. اگر هر مسیریاب بداند که کدامیک از خطوط او در «درخت پوشا» قرار گرفته اند، می تواند یک بسته فراگیر ورودی را فقط بر روی خطوطی در خروجی، بفرستد که بر روی این درخت پوشا هستند. (البته به استثنای خطی که بسته از آن دریافت شده است.) بهره وری پهنای باند در این روش بسیار عالی است و برای انجام کار، کمترین بسته لازم تولید خواهد شد. تنها مشکل این روش آن است که هر مسیریاب برای یکارگیری این روش باید دانش و اطلاعات کافی در خصوص درختهای پوشا در زیر شبکه داشته باشد. برخی اوقات این اطلاعات در دسترس هستند (مثلاً در مسیریابی حالت لینک - LS) ولی گاهی موجود نیستند (مثل مسیریابی بردار فاصله - DV).

آخرین الگوریتم پخش فراگیر، آنست که حتی وقتی مسیریاب چیزی در خصوص درختهای پوشا نمی داند تلاش کند بصورت تقریبی، رفتار الگوریتم درخت پوشا را از خود نشان بدهد! این ایده، «هدایت بر روی مسیر معکوس» (Reverse Path Forwarding) نامیده شده و به نظر بسیار ساده می رسد. وقتی یک بسته از نوع پخش فراگیر به یک مسیریاب وارد می شود آن مسیریاب ابتدا بررسی می کند که «آیا بسته از روی همان خطی وارد شده که بسته های معمولی برای ارسال به مبدا آن بسته، بر روی همان خط ارسال می شوند؟» اگر اینگونه باشد این احتمال بالا وجود دارد که بسته خودش در مسیر بهینه قرار دارد و طبعاً اولین نسخه بسته ای است که او دریافت داشته است. در این حالت مسیریاب این بسته را بر روی تمام خطوط خود (به استثنای خطی که بسته از آن وارد شده) کپی و ارسال می کند. اگر بسته از روی خطی وارد شود که آن خط بر روی مسیر بهینه برای رسیدن بدان مبدا نیست بسته نادیده گرفته می شود چرا که احتمالاً تکراری است.

مثالی از روش «هدایت بر روی مسیر معکوس» در شکل ۵-۱۶ نشان داده شده است. بخش (الف) زیر شبکه را نشان می دهد و بخش (ب) درخت Sink Tree به مبدا مسیریاب I و بخش (ج) عملکرد الگوریتم «هدایت بر روی مسیر معکوس» را به تصویر کشیده است. در اولین گام، بسته ای فراگیر برای F و H و J و N (یعنی گروه های سطح دوی درخت) می فرستد. هر یک از این بسته ها از طریق خطی دریافت می شوند که بر روی مسیری بهینه به I قرار دارند^۱ (با فرض آنکه مسیرهای بهینه از F و H و J و N به مسیریاب I، بر روی درخت Sink Tree قرار گرفته اند)؛ در شکل به دور نام هر مسیریاب که بر روی مسیری بهینه به I قرار گرفته یک دایره ترسیم شده است. در گام دوم، بسته تولید می شود (در هر مسیریاب که در گام اول بسته ای را دریافت کرده، بسته تولید می گردد) از این هشت بسته ۵ عدد به مسیریابهای می رسند که بر روی مسیر بهینه به سمت I قرار دارند [یعنی A و D و G و O و M؛ سه بسته دیگر چون بر روی مسیر بهینه به I نیستند پس از دریافت می شوند. -م]. از شش بسته ای که در گام سوم تولید شده فقط سه تای آنها از طریق خطی که بر روی مسیر بهینه به I قرار دارد دریافت شده اند (از طریق C و E و K) و بقیه تکراری تلقی می شوند. پس از پنج گام و تولید ۲۴ بسته، فرآیند ارسال فراگیر به پایان می رسد، درحالیکه اگر ارسال بسته ها دقیقاً از طریق درخت Sink Tree انجام شود، تعداد گامها ۴ و کل بسته های تولیدی

۱. عبارت دیگر مسیرهای N، J، H، F همگی برای ارسال یک بسته معمولی برای مبدا بسته یعنی I، از همان خطی استفاده می کنند که بسته پخش از روی همان خط وارد شده است. -م



شکل ۵-۱۶. «هدایت بر روی مسیر معکوس» (الف) یک زیر شبکه (ب) یک درخت (ج) Sink Tree که در روش هدایت بر روی مسیر معکوس ساخته می شود.

۱۴ عدد خواهد بود.

مزیت اساسی روش «هدایت بر روی مسیر معکوس» آن است که هم در حد معقولی کارآمد و هم پیاده سازی آن ساده است. در این روش هر مسیریاب نیازی به دانستن درختهای پوشا (Spanning Tree) ندارد و سربار ناشی از قرار دادن فهرست مقاصد چندگانه در بسته های فراگیر تحمیل نخواهد شد. همچنین این روش نیاز به مکانیزم خاصی جهت خاتمه دادن به پروسه تولید نامحدود بسته ها ندارد در حالی که در روش ارسال سیل آسا به چنین مکانیزمی نیاز است (یعنی در این روش به تمهیداتی مثل درج شمارنده گام در هر بسته یا اطلاع قبلی از قطر زیر شبکه یا ذخیره فهرست بسته هایی که تاکنون از یک مبدا منتشر شده اند، نیاز نیست).

۸-۲-۵ مسیریابی چندپخششی (Multicast Routing)

در برخی از کاربردها نیاز است که پروسه های مجزا و پراکنده، در یک گروه با یکدیگر کار کنند؛ به عنوان نمونه می توان به گروهی از پروسه ها که یک «سیستم پایگاه توزیع شده اطلاعات»^۱ را پیاده سازی کرده اند، اشاره نمود. در چنین شرایطی، بطور متناوب لازم می شود که یک پروسه برای دیگر پروسه های عضو گروه خود، پیام بفرستد. اگر این گروه کوچک باشد می توان پیام را بطور جداگانه برای یکایک اعضا ارسال کرد ولیکن اگر گروه بزرگ باشد این راهکار بسیار پرهزینه است. برخی اوقات می توان از روش پخش فراگیر (Broadcasting) بهره گرفت اما روش پخش فراگیر برای رساندن پیامی به ۱۰۰۰ ماشین بر روی شبکه ای که یک میلیون گره دارد، بهیچوجه کارآمد نیست زیرا اکثر گیرندگان پیام هیچ تمایلی به دریافت این گونه پیامها ندارند (بدتر از همه آنکه دیگران علاقمند به دریافت چنین پیامهایی باشند ولی مجاز به دریافت آنها نباشند) بنابراین به راهکاری نیازمندیم که بتوان پیامهایی را برای گروه کاملاً مشخصی ارسال کرد؛ گروهی که اگرچه از لحاظ تعداد، بزرگ به نظر می رسد ولی در مقایسه با کل شبکه بسیار کوچک است.

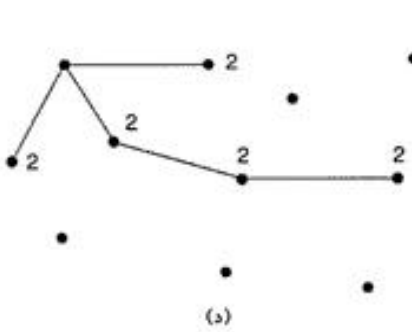
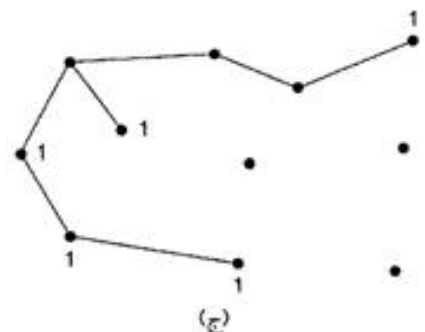
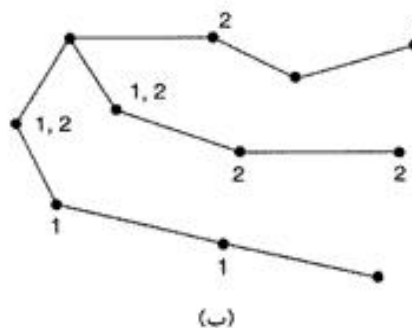
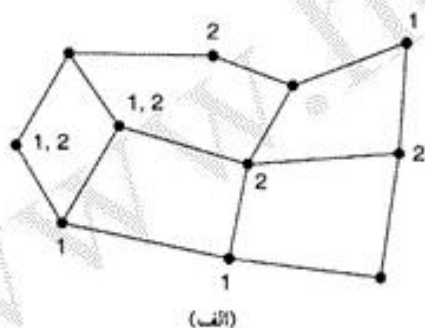
ارسال یک پیام به یک چنین گروهی، «چندپخششی» (Multicasting) و به الگوریتم مسیریابی آن «مسیریابی چندپخششی» (Multicast Routing) گفته می شود. در این بخش یکی از روشهای «مسیریابی چندپخششی» را مطالعه خواهیم نمود. برای کسب آگاهی بیشتر به مراجع زیر نگاهی بیندازید:

(Chu et al., 2000; Costa et al., 2001; Kaseria et al., 2000; Madruga and Garcia-Luna-Aceves, 2001; Zhang and Ryu, 2001).

مسیریابی چندپخشی به مدیریت گروه نیاز دارد؛ یعنی به روشهایی احتیاج است که بتوان گروه ایجاد و حذف کرد و پروسه ها اجازه داشته باشند به گروههایی بپیوندند یا از آنها جدا شوند. چگونگی انجام چنین کارهایی به الگوریتم مسیریابی ربطی ندارد. وقتی پروسه ای به گروهی می پیوندد باید ماشین میزبان خود را از این موضوع آگاه کند. آنچه اهمیت دارد آنست که مسیریاب می داند هر یک از ماشینهای میزبان او عضو چه گروههایی هستند. ماشینهای میزبان باید هر گونه تغییر عضویت خود در یک گروه را به اطلاع مسیریابهای خود برسانند یا آنکه مسیریابها باید خودشان بطور متناوب در این خصوص از ماشینهای میزبان سؤال کنند. به هر حال، مسیریابها از اینکه چه ماشینی عضو چه گروهی است باخبر می شوند. از آن به بعد، مسیریابها به همسایه های خود خبر می دهند و بدین ترتیب این اطلاعات در کل زیرشبکه منتشر خواهد شد.

برای انجام عملیات مسیریابی چندپخشی، هر مسیریاب یک «درخت پوشا» (Spanning Tree) که کل مسیریابها را در بر می گیرد، ایجاد می کند. به عنوان مثال در شکل ۱۷-۵ الف، دو گروه ۱ و ۲ را تعریف کرده ایم. به نحوی که در این شکل ملاحظه می کنید برخی از مسیریابها به ماشینهای میزبانی متصلند که بعضاً عضو یک یا هر دوی این گروهها هستند. [در این شکل، برچسب ۱ یا ۲ بر روی هر مسیریاب، مبین آنست که ماشینهای میزبان متصل به آن مسیریاب عضو چه گروهی هستند. م] در شکل ۱۷-۵ ب، یک درخت پوشا برای مسیریاب سمت چپ نشان داده شده است.

وقتی پروسه ای، یک بسته چندپخشی (Multicast Packet) را برای گروهی ارسال می کند اولین مسیریاب، درخت پوشای خود را بررسی کرده و آن را پیرایش (Prune) می کند، یعنی تمام خطوطی را که نهایتاً به ماشینهای عضو این گروه ختم نمی شوند، حذف می نماید. در این مثال شکل ۱۷-۵ ج درخت پوشا و پیرایش شده گروه ۱ را نشان می دهد. به روش مشابه شکل ۱۷-۵ د درخت پوشا و پیرایش شده گروه ۲ را به تصویر کشیده است. بسته های چندپخشی فقط از طریق درخت پوشای متناسب با گروه مقصد هدایت می شوند.



شکل ۱۷-۵. (الف) ساختار یک شبکه (ب) درخت پوشا برای مسیریاب سمت چپ (ج) درخت چندپخشی برای گروه ۱ (د) درخت چندپخشی برای گروه ۲.

روشهای گوناگونی برای پیرایش درخت پوشا وجود دارد. ساده‌ترین روش، زمانی قابل استفاده است که از روش «مسیریابی حالت لینک» (LS) بهره گرفته شده باشد و هر مسیریاب از توپولوژی کامل شبکه و از جمله عضویت ماشینهای میزبان در گروه‌ها آگاه باشد. در این حالت می‌توان با شروع از انتهای هر مسیر و حرکت به سمت ریشه درخت و حذف تمام مسیریابهایی که در گروه مورد نظر عضو نیستند، درخت پوشا را پیرایش کرد. در روش «مسیریابی بردار فاصله» (DV) می‌توان از راهکار متفاوتی برای پیرایش درخت پوشا بهره گرفت. الگوریتم اصلی همان روش هدایت بر روی مسیر معکوس^۱ است (Reverse Path Forwarding)، ولیکن هرگاه یک مسیریاب دارای هیچ ماشینی عضو یک گروه خاص، نبوده و همچنین به هیچ مسیریاب دیگری که علاقمند به دریافت بسته‌های آن گروه است، متصل نشده باشد با ارسال یک بسته خاص به نام PRUNE به فرستنده اعلام می‌کند که بسته‌های متعلق بدان گروه خاص را برایش نفرستند. اگر یک مسیریاب از تمام خطوط ورودیش پیغام PRUNE دریافت کند و خودش نیز ماشینی که در آن گروه عضو است نداشته باشد، او نیز پیغام PRUNE را صادر می‌نماید. در چنین حالتی، زیرشبکه به صورت بازگشتی (Recursively) پیرایش می‌شود.

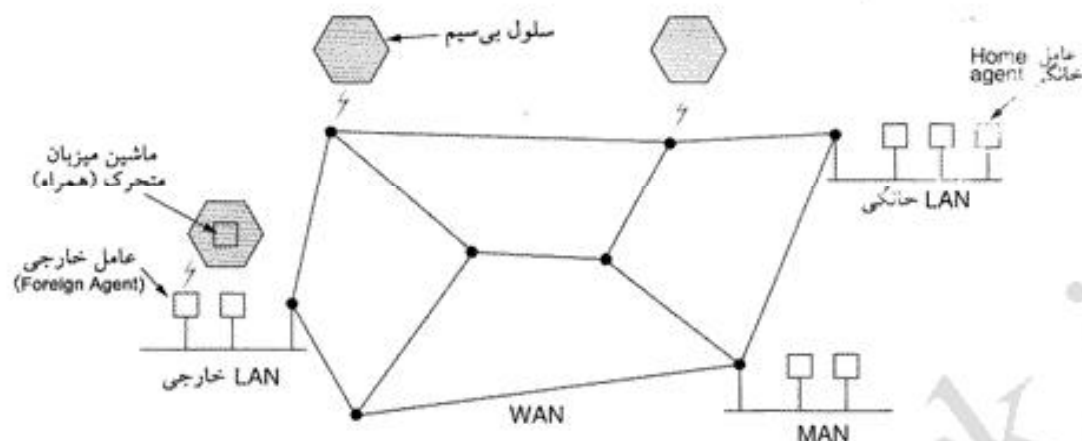
اشکال بالقوه این الگوریتم آن است که در مقیاس شبکه‌های بزرگ، ضعیف عمل می‌کند. فرض کنید که شبکه‌ای دارای n گروه و در هر گروه بطور متوسط m عضو وجود داشته باشد. به ازای هر گروه باید m «درخت پیرایش شده پوشا» ایجاد و ذخیره گردد که جمعاً معادل $m \times n$ درخت است. وقتی گروه‌های زیادی ایجاد شده باشد برای ذخیره این درختها، به حجم حافظه قابل توجهی نیاز خواهد بود.

در طرحی دیگر، از «درختهای مبتنی بر هسته» (Core-Based Tree) بهره گرفته شده است. (Ballardie et al., 1993) در این طرح به ازای هر گروه فقط یک درخت پوشا محاسبه می‌شود، بگونه‌ای که «ریشه درخت» (یا عبارتی هسته) در نزدیکی مرکز آن گروه قرار می‌گیرد. برای ارسال یک پیام چندبخشی، ماشین میزبان آن را به سوی هسته می‌فرستد و هسته نیز عملیات پخش آن بسته را در گروه، از طریق درخت پوشا انجام می‌دهد. اگرچه این درخت برای تمام مسیریابهای میزبان کاملاً بهینه نیست ولی در عوض کاهش هزینه ذخیره‌سازی m درخت به یک درخت در هر گروه، صرفه جویی بسیار قابل توجهی محسوب می‌شود.

۹-۲-۵ مسیریابی برای ماشینهای متحرک

امروزه میلیون‌ها نفر از مردم کامپیوترهای قابل حمل دارند و عموماً تمایل دارند در هر کجای دنیا که باشند نامه‌های الکترونیکی خود را بخوانند و به سیستم فایل همیشگی خود دسترسی داشته باشند. این ماشینهای متحرک مشکل جدیدی را بوجود می‌آورند: قبل از هدایت یک بسته به سوی ماشین متحرک، در ابتدا شبکه باید آن ماشین را پیدا کند. موضوع پیوستن ماشینهای میزبان متحرک (Mobile Hosts) به شبکه، موضوعی بسیار جدید و نو است ولیکن در این بخش به ارائه برخی از راهکارها در این زمینه خواهیم پرداخت. مدلی که طراحان شبکه عموماً برای ساختار ارتباطی جهان در نظر می‌گیرند در شکل ۵-۱۸ نشان داده شده است. در این شکل یک WAN با تعدادی مسیریاب و ماشین میزبان دیده می‌شود. به WAN تعدادی LAN،

۱. عبارت ساده‌تر در ابتدای کار هر بسته چندبخشی بروش «هدایت بر روی مسیر معکوس» که در بخش قبل بررسی شد بر روی خطوط خروجی مسیریابها هدایت می‌شوند و تا اینجای کار هیچ تفاوتی با پخش فراگیر ندارد. به محض پخش اولین بسته بصورت فراگیر، تمام مسیریابهایی که انتظار دریافت چنین بسته‌ای را نداشته‌اند با ارسال بسته کنترلی Prune از همسایه خود که این بسته را برایشان فرستاده خوااهش می‌کنند این کار را تکرار نکنند. وقتی یک مسیریاب خودش هیچ عضوی در این گروه ندارد و تمام همسایه‌هایش نیز با ارسال Prune از او خواسته‌اند که بسته‌های آن گروه را برایشان نفرستد، لزومی به دریافت بسته‌های آن گروه نمی‌بیند و او هم پیغام Prune به همسایه قبلی خود می‌فرستد. بدین ترتیب از آخر مسیر به اول درخت پوشای فرضی اصلاح می‌شود. -م



شکل ۵-۱۸. یک شبکه WAN که شبکه های LAN، MAN و سلولهای بی سیم بدان متصل شده اند.

MAN و شبکه بی سیم (از نوعی که در فصل ۲ مطالعه کردیم)، متصل شده است. به ماشینهای میزبان که حرکت نمی کنند ماشینهای ثابت (Stationary) گفته می شود. این ماشینها از طریق سیمهای مسی یا فیبرهای نوری به شبکه متصل شده اند. برعکس، دو نوع دیگر ماشین میزبان، قابل رؤیت است: «ماشینهای میزبان مهاجر» (Migratory Hosts) ماشینهای ثابتی هستند که گاه به گاه از یک سایت ثابت به سایتی دیگر نقل مکان می کنند ولی فقط زمانی در شبکه قرار می گیرند که اتصال فیزیکی آنها برقرار شود. «ماشینهای میزبان سیار» (Roaming Hosts) در حال حرکت هم کار می کنند و باید ارتباط خود را در حین حرکت حفظ کنند. ما این دو رده از ماشینهای میزبان را «ماشینهای میزبان متحرک» (Mobile Hosts) می نامیم: یعنی تمام ماشینها دور از محل استقرار همیشگی خود هستند و می خواهند به شبکه متصل شوند.

فرض بر آن است که تمام ماشینها دارای یک «محل استقرار دائمی» (Home Location) هستند که هیچگاه تغییر نمی کند. همچنین هر ماشین میزبان دارای آدرس ثابتی در محل استقرار دائم خود است که موقعیت این محل را مشخص می کند (شبیه به یک شماره تلفن در آمریکا مثل ۱۲۱۲۰۵۵۵۱۲۱۲، با کد کشوری ۱ و کد منطقه ۲۱۲). هدف نهایی فرآیند مسیریابی در سیستمی با ماشینهای متحرک، آنست که بتوان بسته ها را به کمک آدرس دائم آنها به ماشینها رساند و ماشینهای میزبان (در هر کجا که باشند) بتوانند بسته های خود را تحویل بگیرند. ظریفترین بخش قضیه آنست که ابتدا باید این ماشینها را پیدا کرد.

در مدل شکل ۵-۱۸ کل دنیا از نظر جغرافیایی به چندین بخش کوچک تقسیم شده است. اجازه بدهید آنها را «ناحیه» بنامیم؛ «ناحیه»، خود یک شبکه LAN یا یک «سلول بی سیم» (Wireless Cell) است. در هر ناحیه یک یا چند «عامل خارجی» (Foreign Agent) وجود دارد که در عمل پروسه هایی هستند که فهرست ماشینهای متحرک و میهمان آن ناحیه را پیگیری و مدیریت می کنند. علاوه بر ناحیه یک «عامل خانگی» (Home Agent) دارد که فهرست ماشینهایی را نگهداری می کند که محل استقرار دائمی آنها همین ناحیه است ولی فعلاً در نواحی دیگر به سر می برند.

وقتی ماشین جدیدی وارد یک ناحیه می شود (از طریق اتصال فیزیکی با شبکه مثلاً با اتصال کابل آن به LAN و یا با پرسه زدن در درون سلول بی سیم)، باید خودش را در «عامل خارجی» ثبت نام نماید. روال ثبت نام عموماً به شکل زیر انجام می شود:

۱. هر «عامل خارجی» بطور متناوب یک بسته فراگیر پخش کرده و حضور خود و آدرسش را به اطلاع همه

می‌رساند. یک ماشین متحرک تازه وارد باید منتظر چنین پیامهایی بماند ولیکن اگر در اسرع وقت، چنین پیامی نرسد ماشین متحرک می‌تواند بسته‌ای فراگیر برای همه بفرستد با این مضمون که: «آیا هیچ عامل خارجی در اینجا هست؟»

۲. ماشین متحرک در عامل خارجی ثبت‌نام می‌کند، آدرس محل استقرار دائم خود، آدرس فیزیکی (یعنی آدرس لایه پیوند داده‌ها) فعلی خود و برخی اطلاعات امنیتی (جهت احراز هویت) را ارائه می‌دهد.

۳. عامل خارجی با عامل خانگی آن ماشین متحرک تماس گرفته و می‌گوید: یکی از ماشینهای شما در ناحیه ما به سر می‌برد! در این پیام که از سوی «عامل خارجی» به سوی «عامل خانگی» ارسال می‌شود آدرس شبکه عامل خارجی [آدرس خودش] مشخص می‌شود. این پیام همچنین شامل برخی اطلاعات امنیتی است تا عامل خانگی را متقاعد کند که ماشین متحرک واقعاً در ناحیه اوست.

۴. عامل خانگی اطلاعات امنیتی ارائه شده را که محتوی «مهر زمان» (Timestamp) نیز هست بررسی می‌کند تا برایش ثابت شود که این پیام در خلال همین چند ثانیه اخیر تولید شده است. اگر متقاعد شود به «عامل خارجی» اجازه ادامه کار را می‌دهد.

۵. وقتی «عامل خارجی» پیام تأیید «عامل داخلی» را دریافت کند یک درایه (Entry) در جدول خود ایجاد کرده و به ماشین متحرک اعلام می‌کند که ثبت‌نام شده است.

آرامی آن است که وقتی یک ماشین میزبان ناحیه‌ای را ترک می‌کند به همان ترتیب اعلام کند تا عضویت آن لغو شود، ولی بسیاری از کاربران به محض آنکه کارشان تمام می‌شود بلافاصله کامپیوتر خود را خاموش می‌کنند و مهلتی برای اعلام ترک ناحیه، باقی نمی‌ماند!

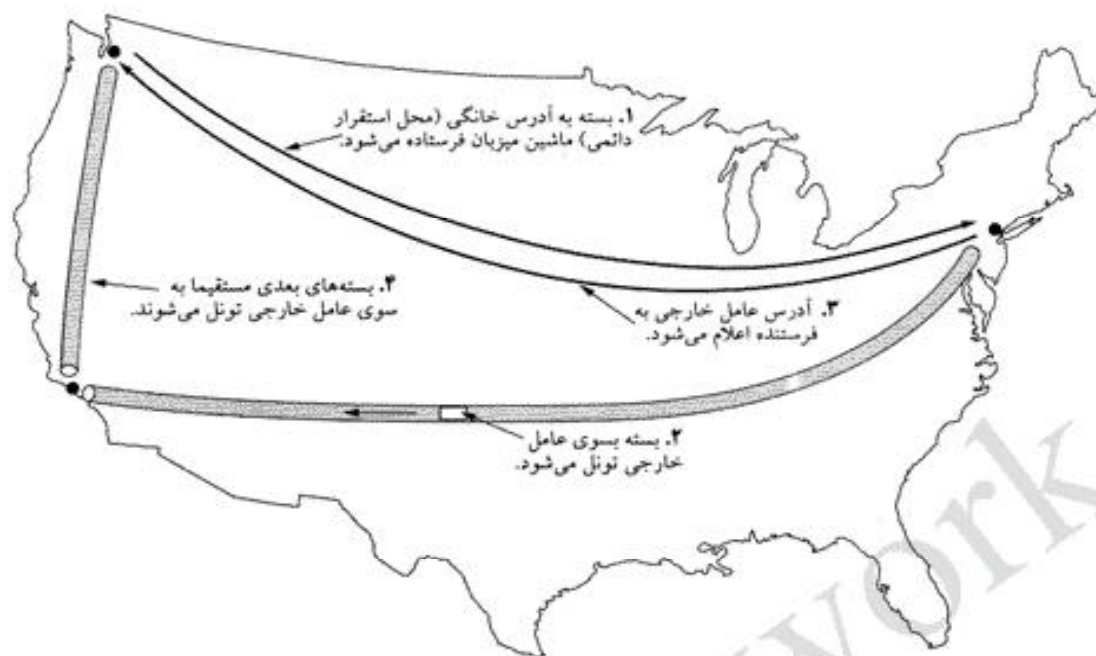
وقتی بسته‌ای برای یک ماشین متحرک ارسال می‌شود ابتدا آن بسته به سوی شبکه LAN خانگی او [یعنی محل استقرار دائم او] مسیریابی و هدایت می‌شود زیرا در حقیقت بسته به آدرس دائم او ارسال شده است. (مرحله ۱ از شکل ۵-۱۹) در این شکل فرستنده‌ای در شهری از سیاتل آمریکا می‌خواهد بسته‌ای را برای یک ماشین میزبان متحرک در نیویورک بفرستد. بسته ابتدا به آدرس شبکه LAN خانگی او در نیویورک ارسال می‌شود و در آنجا توسط «عامل خانگی» تحویل گرفته می‌شود. عامل خانگی در حافظه خود محل استقرار جدید (و موقت) ماشین متحرک را جستجو کرده و آدرس عامل خارجی که آن ماشین را در پوشش خود گرفته (مثلاً در لوس آنجلس) پیدا می‌کند.

در این لحظه، عامل خانگی دو کار انجام می‌دهد: اول آن که این بسته را در درون فیلد داده از یک بسته بیرونی دیگر جاسازی کرده و آن را برای عامل خارجی می‌فرستد. (مرحله ۲ از شکل ۵-۱۹) به این مکانیزم «ایجاد تونل» (Tunneling) گفته می‌شود و بعداً به تفصیل در مورد آن صحبت خواهیم کرد. پس از دریافت بسته جاسازی شده توسط عامل خارجی، بسته اصلی از درون فیلد داده آن جدا شده و به ماشین متحرک تحویل می‌شود.

دوم آنکه عامل خانگی به فرستنده بسته اعلام می‌کند که از این به بعد بسته‌های خود را با جاسازی درون یک بسته دیگر (که به صراحت آدرس «عامل خارجی» در آن درج شده) مستقیماً به عامل خارجی بفرستد. (مرحله ۳) از آن به بعد بدون در نظر گرفتن موقعیت دائمی ماشین متحرک، بسته‌ها مستقیماً به سوی عامل خارجی هدایت شده و تحویل ماشین متحرک می‌شود.^۱

روشهایی که تاکنون معرفی شده‌اند از چندین جهت با هم تفاوت دارند: اولین مورد آن که چه مقدار از وظایف

۱. در فرآیند تونل یک بسته کامل در درون یک بسته دیگر جاسازی می‌شود: آدرس بسته بیرونی مربوط به «عامل خارجی» است و آدرس بسته درونی هویت صاحب اصلی بسته یعنی ماشین متحرک را مشخص می‌کند. -م



شکل ۵-۱۹. مسیریابی بسته ها بسوی ماشینهای متحرک.

این پروتکل توسط مسیربایها و چه مقدار توسط ماشینهای میزبان انجام می شود و آن بخش از وظایف که در ماشین میزبان باید انجام شود در چه لایه ای تعریف شده است. دوم آن که در برخی از روشها، مسیربایهای واقع بر روی مسیر قادرند «نگاشتهای آدرس» [یعنی تغییر آدرس ماشین متحرک به آدرس جدید] را ثبت کرده و بدون هیچ دخالت بیرونی در میانه راه جلوی ترافیک ارسالی به آدرس قبلی را گرفته و آن را به سمت آدرس جدید تغییر مسیر بدهند.

سوم آن که در برخی دیگر از این روشها به هر ماشین متحرک که میهمان شبکه ای جدید شده یک آدرس موقت ولیکن منحصر به فرد داده می شود در حالی که در برخی دیگر این آدرس موقت مربوط به آن عامل خارجی می باشد که هدایت ترافیک تمام میهمانان خود را بر عهده گرفته است.

تفاوت چهارم این روشها آنست که وقتی بسته ها برای رسیدن به یک ماشین دیگر آدرس دهی شده اند چگونه تحویل ماشین دیگری می شوند. [بدین معنا که بسته هایی که به سوی آدرس قبلی یک ماشین روانه شده اند به چه نحو برای هدایت به جایی دیگر تغییر آدرس می دهند. -م] گزینه اول آنست که در هر بسته، آدرس مقصد عوض شود و بسته تغییر یافته از نو ارسال شود. گزینه دوم آنست که کل بسته به همراه آدرس دائم (خانگی) و با کلیه مشخصات [به صورت دست نخورده]، در درون فیلد داده از یک بسته دیگر جاسازی (کپسوله) شود و بسته جدید به آدرس موقت ماشین ارسال گردد.

تفاوت آخر آنکه روشهای مختلف از دیدگاه تمهیدات امنیتی با یکدیگر فرق می کنند. عموماً وقتی یک ماشین میزبان یا مسیربای، پیامی با مضمون این مثال دریافت می کند: «لطفاً از هم اکنون به بعد تمام نامه های استفانی را برای من بفرست!» دو سؤال مطرح می شود: این پیام واقعاً متعلق به کیست (با چه کسی صحبت می شود) و آیا چنین کاری اصولاً به صلاح است؟ در مراجع زیر چندین پروتکل در خصوص ماشینهای متحرک تشریح و مقایسه شده اند:

(Hac and Guo, 2000; Perkins, 1998a; Snoeren and Balakrishnan, 2000; Solomon, 1998; Wang and Chen, 2001)

۱۰-۲-۵ مسیریابی در شبکه‌های ویژه (Routing in Ad Hoc Networks)

تاکنون روشهای مسیریابی در محیطهایی را که ماشینهای میزبان متحرکند ولی مسیریابها ثابت هستند، بررسی کرده‌ایم. یک حالت استثنایی آن است که مسیریابها نیز خودشان متحرک باشند! از چنین محیطهایی می‌توان به موارد زیر اشاره کرد:

۱. وسایل نقلیه نظامی در صحنه نبرد بدون دسترسی به هیچگونه زیرساخت ارتباطی

۲. ناوگان دریایی بر روی دریا

۳. نیروی امداد در شرایط اضطراری مثل زلزله که کلیه زیرساختها را نابود کرده است

۴. گروههایی افراد با کامپیوتر کیفی، در ناحیه‌ای بدون شبکه بی سیم 802.11

در تمام این موارد (و نظایر آن) هر گره شامل یک مسیریاب و یک ماشین میزبان است که اغلب هر دوی آنها بر روی یک ماشین قرار می‌گیرند. شبکه‌ای از این گره‌ها که در کنار هم قرار می‌گیرند اصطلاحاً «شبکه‌های ویژه» یا MANET نامیده می‌شود. اجازه بدهید مختصراً این شبکه‌ها را بررسی نماییم. برای آگاهی بیشتر می‌توان به (Perkins, 2001) مراجعه کرد.

آنچه که «شبکه‌های ویژه» را از «شبکه‌های سیمی» متمایز می‌کند آن است که تمام قواعد طبیعی در خصوص توپولوژی ثابت، همسایه‌های شناخته شده ثابت، تناظر دائم بین موقعیت فیزیکی و آدرس IP ماشینها و مواردی از این قبیل در خصوص «شبکه‌های ویژه» صادق نیست. مسیریابها در رفت و آمد هستند و ممکن است در هر لحظه از محل جدیدی سر در آورند! در یک شبکه سیمی هر گاه یک مسیریاب، مسیری معتبر به برخی از نقاط مقصد در شبکه داشته باشد آن مسیر تا ابد معتبر خواهد بود (مگر آن که در جایی از سیستم خرابی بوجود بیاید). در یک شبکه ویژه، توپولوژی شبکه به طور دائم در تغییر است لذا اعتبار مسیرها و بهینگی آنها به ناگاه و بی هیچ هشدار قبلی تغییر می‌کند. آشکار است که با چنین وضعیتی، مسیریابی در شبکه‌های ویژه کاملاً متفاوت از شبکه‌های ثابت می‌باشد.

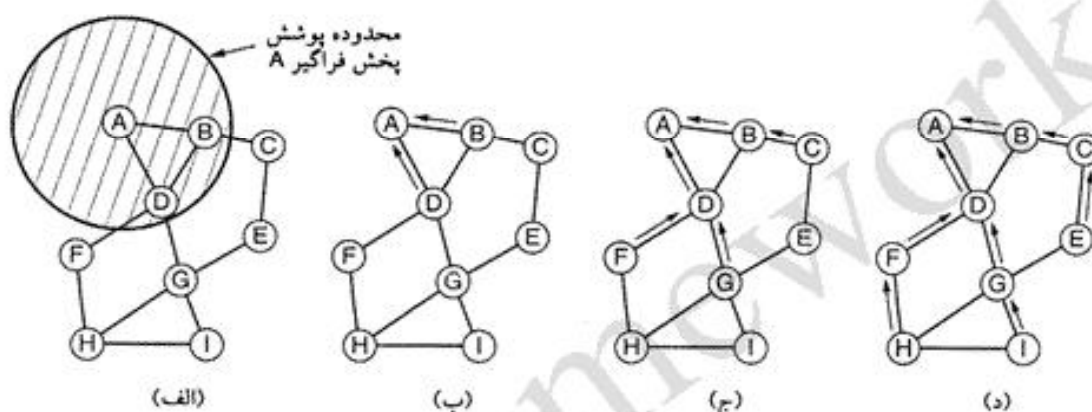
گونه‌های متعددی از الگوریتمهای مسیریابی برای شبکه‌های ویژه پیشنهاد شده است. یکی از جالبترین آنها الگوریتم مسیریابی AODV^۱ است. (Perkins and Royer, 1999) این الگوریتم گونه‌ای از «الگوریتم بردار فاصله بلمن-فورده» محسوب می‌شود که برای کار در محیطهای متحرک تطبیق داده شده و در آن پهنای باند محدود و عمر کم باتری ماشینها در این محیط [در محاسبات مربوط به مسیرهای بهینه] در نظر گرفته شده است. یکی دیگر از ویژگیهای نامتعارف این روش آن است که الگوریتم «برحسب تقاضا» (On-Demand) عمل می‌کند بدین معنا که مسیر رسیدن به برخی از نقاط مقصد، فقط وقتی تعیین می‌شود که کسی بخواهد بسته‌ای را بدان مقصد بفرستد. اجازه بدهید ببینیم قضیه از چه قرار است.

کشف مسیر

یک شبکه ویژه را در هر لحظه از زمان می‌توان با گرافی از گره‌ها توصیف کرد. (گره‌ها عبارتند از مسیریابها + ماشینهای میزبان) اگر دو ماشین بتوانند از طریق سیستم رادیویی خود مستقیماً با یکدیگر ارتباط برقرار کنند می‌گوییم این دو گره (Node) بهم متصل هستند. (یعنی در گراف شبکه، بین این دو گره یک کمان وجود دارد). از آنجایی که امکان دارد یکی از این دو ماشین، فرستنده پرقدرت‌تری نسبت به دیگری داشته باشد لذا این امکان وجود دارد که ارتباط A به B برقرار باشد ولی ارتباط B با A میسر نباشد؛ ولیکن برای سادگی فرض را بر آن می‌گذاریم که تمام ارتباطات، دو طرفه و متقارن هستند. همچنین باید بدین نکته اشاره کرد که هر گاه دو گره در برد

رادیویی یکدیگر باشند تضمینی وجود ندارد که ارتباط آنها برقرار باشد؛ ممکن است بین آنها ساختمان، تپه یا موانع دیگری وجود داشته باشد که جلوی ارتباط آنها را بگیرد. [هر چند در برد یکدیگر واقعند.]

برای توصیف الگوریتم، شبکه ویژه شکل ۵-۲۰ را در نظر بگیرید که در آن یک پروسه در گره A می خواهد بسته ای را برای گره I بفرستد. در الگوریتم AODV، هر گره دارای جدولی است که کلید این جدول، آدرس مقصد است^۱ و هر یک از رکوردهای این جدول، اطلاعاتی در خصوص مقصد و آنکه برای رساندن بسته ای به آن مقصد باید بسته را به کدامیک از همسایه های آن فرستاد، در خود نگاهداری می کنند. فرض کنید که A در جدول خود جستجو کرده و هیچ درایه ای متناظر با I در آن نمی یابد. حال بایستی مسیری به I کشف کند. همین ویژگی که مسیرها فقط در هنگام لزوم کشف می شوند به الگوریتم، ویژگی On-Demand یعنی «برحسب تقاضا» داده است.



شکل ۵-۲۰. (الف) محدوده پوشش پخش فراگیر A پس از آنکه B و D پخش فراگیر A را دریافت کردند. (ج) پس از آنکه C و F و G پخش فراگیر A را دریافت کردند. (د) پس از آنکه E و H و I پخش فراگیر A را دریافت کردند. گره های سایه دار دریافت کنندگان جدید هر مرحله محسوب می شوند. فلشها مسیر معکوس (مسیر برگشت) را مشخص می کنند.

برای پیدا کردن موقعیت I، گره A یک بسته خاص به نام ROUTE REQUEST (تقاضای مسیر) ساخته و آن را به صورت فراگیر منتشر می کند. به گونه ای که در شکل ۵-۲۰ الف مشهود است، این بسته به B و D می رسد. در حقیقت دلیل آنکه در این گراف، B و D به A متصل شده اند آنست که قادرند سیگنال مخابراتی A را دریافت کنند. به عنوان مثال بین گره A و F در گراف هیچ کمائی ترسیم نشده است چرا که F نمی تواند سیگنال رادیویی A را دریافت کند. بنابراین ارتباط مستقیمی بین A و F وجود ندارد.

قالب بسته ROUTE REQUEST در شکل ۵-۲۱ نشان داده شده است. این بسته شامل «آدرس مبدا» و «آدرس مقصد» است و مشخص می کند که چه کسی در جستجوی چه کسی است. (عموماً از آدرسهای IP آنها استفاده می شود) این بسته همچنین حاوی یک «شناسه تقاضا» (Request ID) است. این شناسه در حقیقت یک شمارنده محلی است که در هر گره وجود دارد و هرگاه که یک بسته ROUTE REQUEST منتشر گردید یک واحد به آن اضافه می شود. ترکیب «آدرس مبدا» و فیلد «شناسه تقاضا»، هویت بسته ROUTE REQUEST را به صورت یکتا و منحصر بفرد تعیین کرده و بدین ترتیب گره ها قادرند بسته هایی که احتمالاً به صورت تکراری دریافت می شوند را تشخیص داده و حذف کنند.

هر گره به غیر از «شمارنده شناسه تقاضا» یک شمارنده دیگر هم دارد که هرگاه بسته ROUTE REQUEST

۱. یعنی جستجوی رکوردها در این جدول براساس آدرس مقصد انجام می شود. م

Source address	Request ID	Destination address	Source sequence #	Dest. sequence #	Hop count
----------------	------------	---------------------	-------------------	------------------	-----------

شکل ۵-۲۱. قالب بسته ROUTE REQUEST

ارسال شود (یا پاسخی برای بسته ROUTE REQUEST دیگران دریافت گردد) یک واحد بدان اضافه می‌گردد. عملکرد این شمارنده تا حدودی شبیه به یک ساعت است و کاربرد آن تشخیص مسیر جدید از مسیر قدیمی است. [یعنی وقتی دو بسته یکسان در مورد مشخصات یک مسیر دریافت می‌شود این شمارنده مشخص می‌کند که کدام جدیدتر از دیگری است] فیلد چهارم از شکل ۵-۲۱ مقدار همین شمارنده را برای گره مبدا (که در این مثال A است) مشخص می‌کند. [در این مثال A در جستجوی I است لذا در بسته ROUTE REQUEST شماره ترتیب تقاضای خود را درج کرده است]. فیلد پنجم مشخص کننده آخرین و جدیدترین شماره بسته‌ای است که A در مورد I دریافت کرده است. (و صفر است اگر تاکنون در مورد I چیزی دریافت نکرده باشد). آخرین فیلد یعنی «شمارنده گام» (Hop Counter) مشخص می‌کند که بسته تاکنون چند گام را طی کرده است. مقدار اولیه این فیلد صفر است و به ازای عبور از هر گره یک واحد به آن اضافه می‌شود.

وقتی بسته ROUTE REQUEST به یک گره می‌رسد (در این مثال به B و D) طبق مراحل زیر پردازش می‌شود:

۱. ابتدا جفت مشخصه (آدرس مبدا، شناسه تقاضا) در یک جدول محلی (که سوابق دریافت چنین بسته‌هایی را نگهداری می‌کند) جستجو می‌شود تا مشخص گردد که آیا این بسته قبلاً نیز دریافت و پردازش شده است؟ اگر تکراری بود بسته حذف شده و پردازش آن در همین جا خاتمه می‌یابد. اگر تکراری نبود این زوج مشخصه بسته را در جدول سوابق وارد می‌کند تا در آینده نیز بسته‌ای مشابه با این بسته پردازش نشود. سپس فرآیند پردازش ادامه می‌یابد.

۲. سپس گیرنده در جدول مسیریابی خود، آدرس مقصد را جستجو می‌کند. اگر یک مسیر جدید و «تازه» به این مقصد پیدا شود یک بسته ROUTE RERLY برای مبدا برگردانده می‌شود تا مبدا نیز از چگونگی رسیدن به مقصد مورد نظر آگاه شود. (اغلب، پاسخ بدین نحو است: از طریق من عمل کن!) «تازه» بودن مسیر بدین معنی است که فیلد «شماره ترتیب مقصد» ذخیره شده در جدول مسیریابی باید بزرگتر یا مساوی همین فیلد در بسته ROUTE REQUEST باشد. اگر کمتر بود بدین معنی تلقی می‌شود که مشخصات ذخیره شده در حافظه، قدیمی‌تر از مسیری است که خود مبدا برای رسیدن به مقصد در اختیار دارد^۱، در این صورت مرحله سوم به اجرا در می‌آید.

۳. از آنجایی که گیرنده هیچ مسیر جدیدی بدان مقصد نمی‌شناسد، به مقدار فیلد «شمارنده گام» (Hop Count) یک واحد اضافه کرده و بسته ROUTE REQUEST را مجدداً پیرامون خود منتشر می‌نماید. البته داده‌های درون بسته را استخراج کرده و آن را به عنوان یک درایه جدید در جدول «مسیرهای معکوس» ذخیره می‌کند.^۲ این اطلاعات بدان جهت مفید است که می‌توان مسیرهای معکوس ایجاد کرد و از

۱. کاربرد فیلد شماره ترتیب مقصد (Destination Sequence #) را در ذهن خود اینگونه فرض کنید که مثلاً A به همسایه‌های خود اعلام می‌کند که من خودم اطلاعاتی در خصوص مسیر رسیدن به A دارم که شماره ترتیب آن (مثلاً) ۱۲۳۴ = Destination Sequence است. اگر شما اطلاعات جدیدتری دارید لطفاً برای من بفرستید! -م

۲. یعنی اگرچه مبدا به دنبال یافتن یک مقصد خاص است ولیکن در بسته تقاضا حداقل خود را معرفی کرده است فلذا مسیریابهای گیرنده این تقاضا، لااقل می‌توانند از حضور این مبدا و مسیر رسیدن به آن کسب آگاهی کنند؛ به چنین مسیری «مسیر معکوس» گفته می‌شود. -م

طریق آن، در آینده پاسخ این تقاضا را به مبدا برگرداند. فلشهایی که بر روی شکل ۵-۲۰ دیده می شوند چگونگی ساختن مسیرهای معکوس را نشان می دهند. به محض ایجاد یک مسیر معکوس و جدید، یک زمان سنج (تایمر) برای آن تنظیم می شود. اگر مهلت این زمان سنج منقضی شود و پاسخی به بسته ROUTE REQUEST برنگردد، مسیر ایجاد شده حذف خواهد شد.

هیچیک از گره های B و D نمی دانند که I کجاست لذا این دو نیز ضمن ساختن مسیر معکوس جهت بازگشت پاسخ به A، با تغییر فیلد Hop Count به ۱، آنرا مجدداً منتشر می کنند. (به شکل ۵-۲۰ دقت کنید.) انتشار مجدد بسته توسط B، به C و D می رسد. C نیز درایه ای در جدول مسیرهای معکوس خود ایجاد و آن را از نو منتشر می کند. در مقابل، D آنرا به عنوان بسته ای تکراری حذف می کند. [چون قبلاً از طریق A دریافت کرده است.] به همین طریق بسته منتشره توسط D در B تکراری تشخیص داده شده و حذف می گردد. ولیکن به گونه ای که در شکل ۵-۲۰-ج دیده می شود بسته منتشره توسط D در F و G پذیرفته و ذخیره می شود.

پس از آن که E و H و I نیز بسته منتشر شده را دریافت کردند، عاقبت پیغام ROUTE REQUEST به مقصد خود می رسد (یعنی به خود I یا به گره ای که از موقعیت I خبر دقیق دارد می رسد). (شکل ۵-۲۰-د را ببینید.) اگرچه ما کل فرآیند انتشار را در سه مرحله جدا نشان داده ایم ولیکن انتشار بسته ها از گره های مختلف به صورت هماهنگ و همزمان انجام نمی شود.

گره I در پاسخ به تقاضای ورودی، یک بسته ROUTE REPLY مطابق با شکل ۵-۲۲ ایجاد می کند. فیلدهای «آدرس مبدا»، «آدرس مقصد» مستقیماً از بسته تقاضا استخراج و در بسته پاسخ کپی می شوند ولیکن «شماره ترتیب مقصد» (Destination Sequence Number) برگرفته از مقدار شمارنده ای است که درون حافظه I قرار دارد. فیلد «شمارنده گام» نیز به صفر تنظیم می شود. فیلد «طول عمر» (Lifetime) مشخص می کند که مشخصات مسیر اعلام شده تا چه زمانی معتبر است. این بسته صرفاً برای گره ای ارسال می شود که بسته تقاضا (یعنی ROUTE REQUEST) از طریق او دریافت شده است (در این مثال گره G). این بسته، مسیر معکوس خود به D و نهایتاً به A را طی می کند. در هر گره به مقدار فیلد شمارنده پیام یک واحد اضافه می شود تا هر گره که آن را می بیند بفهمد که فاصله اش تا گره مقصد (در این مثال I) چقدر است.

Source address	Destination address	Destination sequence #	Hop count	Lifetime
----------------	---------------------	------------------------	-----------	----------

شکل ۵-۲۲. قالب بسته ROUTE REPLY.

هر گره میانی این بسته را در راه بازگشت، بررسی می کند. اگر یکی از سه شرط زیر برقرار باشد، اطلاعاتی در خصوص مسیر رسیدن به I، در جدول مسیریابی هر گره میانی ذخیره خواهد شد:

۱. اگر هیچ مسیر شناخته شده ای به انداخته باشد.
۲. اگر شماره ترتیب I (یعنی شماره ای که I در فیلد Destination Seq.No. گذاشته است) بزرگتر از شماره ای باشد که در جدول مسیریابی درج شده است.
۳. اگر شماره ترتیب یکسان باشد ولی مسیر جدید کوتاه تر باشد. [کوتاه بودن از فیلد شمارنده گام مشخص می شود.]

در این روش تمام گره هایی که بر روی مسیر معکوس قرار دارند، مجاناً از مسیر رسیدن به I آگاه می شوند (یکی از محاسن جانبی کشف مسیر توسط A، آگاهی گره های میانی است). گره هایی که بسته ROUTE REQUEST را

در مسیر رفت دریافت کرده اند ولیکن در مسیر معکوس نیستند (در این مثال B و C و E و F و H) پس از منقضی شدن زمان سنج (تایمر)، مسیر معکوس به A را حذف می کنند.

در شبکه های عظیم، این الگوریتم بسته های فراگیر بسیار زیادی را تولید می کند، حتی اگر گره مقصد در نزدیکی مبدا باشد. برای کاهش تولید بسته ها در فرآیند انتشار، می توان بدین نحو عمل کرد: فیلد TTL (Time To Live) در بسته IP، توسط فرستنده آن به مقداری نزدیک به «قطر»^۱ شبکه تنظیم شده و به ازای عبور از هر گره، یک واحد از آن کسر گردد؛ هر گاه به صفر رسید، بسته به جای انتشار مجدد حذف شود.

فرآیند کشف مسیر را می توان بدین نحو اصلاح کرد: برای یافتن موقعیت مقصد، فرستنده، بسته ROUTE REQUEST را با تنظیم فیلد TTL به ۱ ارسال می کند. اگر در یک مدت زمان معقول پاسخی برگشت بسته بعدی را با TTL معادل ۲ ارسال می نماید. همین روال با مقادیر ۳ و ۴ و ۵... ادامه می یابد تا بالاخره پاسخی برگردد. در این روش، جستجو ابتدا به صورت محلی و در پیرامون گره شروع شده و در هر مرحله محدوده جستجو گسترده تر می شود.

نگهداری مسیر (Route Maintenance)

از آنجایی که گره ها می توانند جابجا شده یا کلاً خاموش شوند لذا توپولوژی شبکه گاه به گاه تغییر می کند. به عنوان مثال در شکل ۵-۲۰ اگر G به ناگاه خاموش شود، A نخواهد فهمید مسیری که برای رسیدن به I در اختیار داشته (یعنی مسیر ADGI) دیگر برقرار نیست. این الگوریتم باید بتواند به نحوی این مسئله را حل و فصل کند.

هر گره در شبکه بطور متناوب «پیغام سلام» (Hello Message) منتشر می کند. انتظار می رود که همسایه ها به این پیام پاسخ بدهند. اگر پاسخی باز نگشت، متشکرکننده پیام آگاه می شود که همسایه او از بردش خارج شده و دیگر ارتباط آنها برقرار نیست. به روش مشابه اگر بسته ای معمولی برای همسایه خود بفرستد و پاسخی نگیرد متوجه می شود که آن همسایه در دسترس نیست.

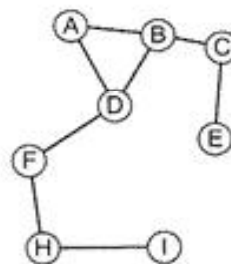
از این اطلاعات می توان برای پاک کردن مسیرهایی که دیگر کار نمی کنند بهره گرفت. هر گره مثل N برای یکایک گره های مقصد که همسایه هایش در خلال ΔT ثانیه گذشته بسته ای را از طریق او بدان مقصد ارسال کرده اند، فهرستی را نگه می دارد. این فهرست اصطلاحاً «همسایه های فعال N» نامیده می شوند. بدین منظور گره N دارای جدولی است که کلید آن، آدرس مقصد گره های شبکه است؛ همچنین در این فهرست، گره بعدی برای رسیدن بدان مقصد، تعداد گام برای رسیدن به آن مقصد، «آخرین شماره ترتیب» و «فهرست همسایه های فعال» درج شده است. به عنوان نمونه جدول مسیریابی گره D برای توپولوژی مثال قبلی، چیزی شبیه به شکل ۵-۲۳-الف است.

وقتی یکی از همسایه های N از دسترس خارج می شود، گره N جدول مسیریابی خود را بررسی می کند تا ببیند کدامیک از گره های شبکه در مسیرهای خود از گره حذف شده استفاده می کرده اند. این موضوع به همسایه های فعال اطلاع داده می شود که: تمام مسیرهای آنها از طریق N نامعتبر است (چرا که N بسته های آنها از طریق گره حذف شده ارسال می شده است)؛ همسایه های فعال نیز به همسایه های فعال خود خبر می دهند و مکرراً این کار انجام می شود تا تمام مسیرهایی که وابسته به گره تازه از دست رفته بوده اند از کل جداول مسیریابی حذف شوند. به عنوان مثالی از روش «نگهداری مسیر»، مثال قبلیمان را مدنظر قرار بدهید ولیکن فرض کنید که G به ناگاه خاموش شده است. توپولوژی تغییر یافته شبکه در شکل ۵-۲۳-ب نشان داده شده است؛ وقتی D متوجه می شود که G از میان رفته است به جدول مسیریابی خود مراجعه می کند و می بیند که G بر روی مسیری قرار داشته که به E،

۱. قطر شبکه به معنای طول بزرگترین مسیر در شبکه است. -م

Dest.	Next hop	Distance	Active neighbors	Other fields
A	A	1	F, G	
B	B	1	F, G	
C	B	2	F	
E	G	2		
F	F	1	A, B	
G	G	1	A, B	
H	F	2	A, B	
I	G	2	A, B	

(الف)



(ب)

شکل ۵-۲۳. (الف) جدول مسیریابی D قبل از آنکه G از کار بیفتد. (ب) گراف پس از حذف G از شبکه.

G و I ختم می شده است. اجتماع مجموعه همسایه های فعال این سه گره مقصد عبارت است از {A و B}. به عبارت دیگر A و B در برخی از مسیرهای خود به G متکی بوده اند لذا باید به این مسیریابها اطلاع داد که G دیگر کار نمی کند. D با ارسال بسته های خاصی این موضوع را بدانشان اطلاع داد تا آنها نیز جدول مسیریابی خود را اصلاح نمایند. خود D نیز درایه های متناظر با E، G و I را از جدول مسیریابی خود حذف می کند.^۱

شاید از توضیحاتی که در خصوص الگوریتم AODV ارائه کردیم مشخص نشده باشد که بنیانی ترین تفاوت بین این الگوریتم و الگوریتم پلمن-فورده آن است که گره ها بطور متناوب کل جدول مسیریابی خود را منتشر نمی کنند، بلکه فقط بخش کوچکی از آنرا و آنهم بر حسب تقاضا، ارسال می نمایند. این ویژگی، در پهنای باند مصرفی و طول عمر باتری گره های متحرک صرفه جویی می کند.

AODV همچنین قادر به پخش داده های فراگیر (Broadcast) و مسیریابی چندپخشی (Multicast) است. برای آگاهی بیشتر به مرجع (Perkins and Royer, 2001) مراجعه نمایید. مسیریابی در شبکه های ویژه، یک موضوع پژوهشی داغ و جدید است و اخیراً در این خصوص مقالاتی تألیف و منتشر شده است. برخی از آنها را می توان در مراجع ذیل یافت:

Chen et al., 2002; Hu and Johnson, 2001; Li et al., 2001; Raju and Garcia-Luna-Aceves, 2001; Ramanathan and Redi, 2002; Royer and Toh, 1999; Spohn and Garcia-Luna-Aceves, 2001; Tseng et al., 2001; and Zadeh et al., 2002).

۱۱-۲-۵ جستجوی گره در شبکه های همتا به همتا (Peer-to-Peer)

یکی از پدیده های نسبتاً جدید، شبکه های همتا به همتا است که در آن تعداد کثیری از افراد برای به اشتراک گذاشتن منابع مشترک خود، مستقیماً با یکدیگر در تماسند (این افراد عموماً از طریق یک اتصال ثابت و باسیم به اینترنت متصل شده اند). اولین کاربرد بسیار گسترده از تکنولوژی «همتا به همتا» به یکی از بحث برانگیزترین ماجراهای «گناه جمعی» بدل شد: «۵ میلیون از کاربران Napster می توانستند فایل های موزیک و آواز را که حق امتیاز آنها در

۱. توضیحی برای رفع ابهام از مفهوم همسایه های فعال خالی از لطف نیست. به سطر آخر از جدول ۵-۲۳-ب دقت کنید. این جدول فرضاً متعلق به D است و آخرین رکورد آن بیان می کند که برای رسیدن به گره I باید بسته ها به G فرستاده شوند و تا رسیدن به I فقط دو گام باقیمانده است؛ در فیلد «همسایه های فعال» نام A و B درج شده است یعنی: A و B برای ارسال بسته هایشان به سوی I آنها را به من -D- می دهند. لذا در صورت خرابی G باید به این دو همسایه اطلاع داد تا آنها نیز از این موضوع آگاه شده و جداول خود را اصلاح کنند. -م

تملك دیگران بود بدون اجازه از صاحبین آنها با یکدیگر رد و بدل کنند؛ نهایتاً Napster پس از مناقشات فراوان به حکم دادگاه تعطیل شد.^۱ بهر تقدیر، تکنولوژی همنا به همنا کاربردهای قانونی بسیار جالبی دارد. این تکنولوژی با مسائل و مشکلات مسیریابی مواجه است؛ ولیکن این مسائل با آنچه که تاکنون مطالعه کرده ایم کاملاً یکسان نیست، لذا مروری اجمالی بر آن خالی از لطف نخواهد بود.

آنچه که سیستمهای «همنا به همنا» را جذاب کرده آنست که سیستمی کاملاً توزیع شده ایجاد می کند یعنی تمام گره ها متقارن هستند و هیچگونه کنترل مرکزی یا سلسله مراتب خاصی بر آن حاکم نیست. در یک سیستم رایج همنا به همنا، هر یک از کاربران دارای مقداری اطلاعات هستند که ممکن است برای کاربران دیگر جالب باشد. این اطلاعات می تواند نرم افزارهای رایگان و عمومی، موزیک، عکس و نظایر آنها باشد. هر گاه تعداد کاربران زیاد باشد، یکدیگر را نمی شناسند و نمی دانند آنچه را که در جستجوی آن هستند در کجا پیدا کنند. یک راه حل آن است که یک پایگاه داده مرکزی و بسیار بزرگ از فهرست آنها داشته باشیم ولی ممکن است به دلایلی امکان پذیر نباشد (مثلاً هیچکس حاضر به میزبانی و نگهداری آن نشود) بنابراین مسئله اساسی آن است که در غیاب یک پایگاه مرکزی یا یک فهرست مرکزی، یک کاربر چگونه می تواند گره ای را که اطلاعات مورد نیاز او را در اختیار دارد، پیدا کند.

اجازه بدهید فرض را بر آن بگذاریم که هر کاربر یک یا چند آیتم داده مثل موزیک، عکس، برنامه، فایل و نظائر آنها در اختیار دارد و کاربران دیگر علاقمند به خواندن آنها هستند. هر آیتم توسط یک رشته ASCII نامگذاری شده است. کاربران احتمالی فقط همین رشته ASCII را می دانند و می خواهند بدانند آیا فردی یا افرادی آن را در اختیار دارند و اگر دارند آدرس IP آنان چیست؟

به عنوان مثال به پایگاه توزیع شده مردم شناسی (شجره نامه مردم) دقت کنید. هر کسی که به مردم شناسی علاقمند است فرضاً در خصوص اجداد و بستگان خود اطلاعاتی مثل عکس، صدا یا حتی قطعات ویدیویی در اختیار دارد. ممکن است جذ اعلای بسیاری از افراد یکی باشد و امکان دارد اطلاعات مربوط به آن در چندین «گره» وجود داشته باشد. نام هر رکورد عموماً نام فرد مورد نظر (در شکل و قالب مشخص) است. حال در جایی یک نفر که علاقمند به تحقیق در مورد شجره نامه خود بوده متوجه می شود که نام جذ اعلای او در آرشیو یک گره خاص وجود دارد و ساعت جیبی و طلائی خود را برای برادرزاده اش به ارث گذاشته است!! حال او می خواهد بداند که نام این برادرزاده چیست و آیا در جایی دیگر رکوردی در این رابطه وجود دارد. بدون وجود یک پایگاه اطلاعات مرکزی، چگونه می توان کسی که چنین رکوردی را در اختیار دارد، پیدا کرد؟

برای حل این مسئله الگوریتمهای گوناگونی پیشنهاد شده است. الگوریتمی که بررسی می کنیم «الگوریتم Chord» نام دارد. (Dabek et al., 2001; Stoica et al., 2001) شرح ساده عملکرد آن بدینگونه است: سیستم Chord از مجموعه n کاربر شرکت کننده تشکیل شده است. هر یک از این کاربران ممکن است رکوردهایی را در خود ذخیره کرده و همچنین آمادگی داشته باشند بخشی از فهرست (ایندکس) سیستم را برای استفاده دیگر کاربران نگهداری و عرضه کنند. هر یک از کاربران دارای یک آدرس IP هستند که این آدرس را می توان توسط یک «تابع درهم سازی» (Hash Function) به یک آدرس m بیتی تبدیل کرد. در سیستم Chord از تابع درهم سازی SHA-1 استفاده می شود؛ SHA-1 در مبحث رمزنگاری کاربرد دارد و در فصل هشتم نگاهی بدان خواهیم انداخت. فعلاً به همین اندازه بسنده می کنیم که این تابع یک رشته از بایتها با طول دلخواه را گرفته و براساس آن یک عدد ۱۶۰ بیتی

۱. در حقیقت کاری که Napster انجام می داد این بود که مشترکین خود را که فایل های موزیک بر روی کامپیوتر خود داشتند بهم معرفی می کرد تا بتوانند به صورت بی واسطه و رو در رو (یعنی همان همنا به همنا) فایل های خود را رد و بدل کنند. اینکار خشم دست اندرکاران صنعت موسیقی را برانگیخت! -م-

بشدت تصادفی تولید می کند. بنابراین می توانیم یک آدرس IP را به عددی ۱۶۰ بیتی و یکتا تبدیل کنیم که به این عدد اصطلاحاً «شناسه گره» (Node Identifier) گفته می شود.

از دیدگاه مفهومی تجسم کنید که تمام ۲^{۱۶۰} شماره شناسه گره ها به صورت صعودی پیرامون یک دایره بزرگ چیده شده اند. برخی از این شماره ها مربوط به گره های شرکت کننده (کاربران فعال) هستند ولی بیشتر آنها پوچند. در شکل ۵-۲۴ الف ما دایره شماره ها را برای $m=5$ نشان داده ایم. (فعلاً به کمانهای میانی کاری نداشته باشید). $m=5$ بدین معناست که شماره ها پنج بیتی هستند و شماره های از ۰ تا ۳۱ را در بر می گیرند. در این مثال گره های ۱، ۴، ۷، ۱۲، ۱۵، ۲۰ و ۲۷ مربوط به گره های واقعی بوده و در شکل به صورت سایه دار نشان داده شده اند. مابقی شماره ها وجود خارجی ندارند. (شماره های پوچ)

تابع $successor(k)$ را بدین مضمون تعریف می کنیم که اولین «شناسه گره» مربوط به اولین کاربر واقعی که پس از عدد k قرار گرفته را بر می گرداند. به عنوان مثال $successor(6)=7$ ، $successor(8)=12$ و $successor(22)=27$.

اسامی رکوردها (مثل نام فایل های آواز و نظائر آن) نیز توسط تابع درهم سازی (یعنی SHA-1) به یک رشته ۱۶۰ بیتی تبدیل می شوند که این رشته «کلید» (Key) نامیده می شود. لذا برای تبدیل یک نام (یعنی نام ASCII هر رکورد) به «کلید» متناظر با آن، رابطه $Key = hash(name)$ را تعریف کرده ایم. برای این کار فقط کافی است پروسیجر محلی $hash$ را فراخوانی نماییم. اگر کسی که یک رکورد مفید با نام $name$ در اختیار دارد، بخواهد آن را در دسترس عموم قرار بدهد ابتدا یک شاخص دوتایی شامل (IP Address, name) ساخته و برای ذخیره کردن این شاخص بر روی یک گره مناسب، تابع $successor(hash(name))$ را فراخوانی می نماید. اگرچه ممکن است چندین رکورد با همین نام در گره های متفاوت موجود باشد ولیکن شاخص دوتایی آنها در یک گره ذخیره می شود.^۱ بدین طریق، ایندکس رکوردها (شاخص دوتایی هر رکورد) به صورت تصادفی بر روی گره ها توزیع می شوند. برای آن که تحمل خرابی (Fault Tolerance) این سیستم بالا باشد می توان از تعداد p تابع مختلف $hash$ استفاده کرد تا هر شاخص دوتایی در p گره مختلف ذخیره شود ولیکن فعلاً به این موضوع نخواهیم پرداخت.

اگر بعداً کاربری بخواهد آیتمی با نام $name$ را جستجو نماید ابتدا آن را توسط تابع $hash$ درهم سازی می کند تا کلید آن بدست آید. سپس تابع $successor(key)$ را فراخوانی می کند تا آدرس IP گره ای که فهرست شاخصها را ذخیره کرده، پیدا شود. اگرچه اولین مرحله ساده است ولی مرحله دوم چندان ساده نیست. برای آن که بتوان آدرس IP گره ای را که متناظر با یک کلید خاص است پیدا کرد، هر گره باید چندین «ساختمان داده نظارتی»^۲ در خود ذخیره نماید. یکی از این ساختمانهای داده، آدرس IP گره ای است که در دایره «شناسه گره ها»، پس از خود گره قرار گرفته است. به عنوان مثال در شکل ۵-۲۴، گره بعدی ۴، گره ۷ است و گره بعدی ۷، گره ۱۲ است.

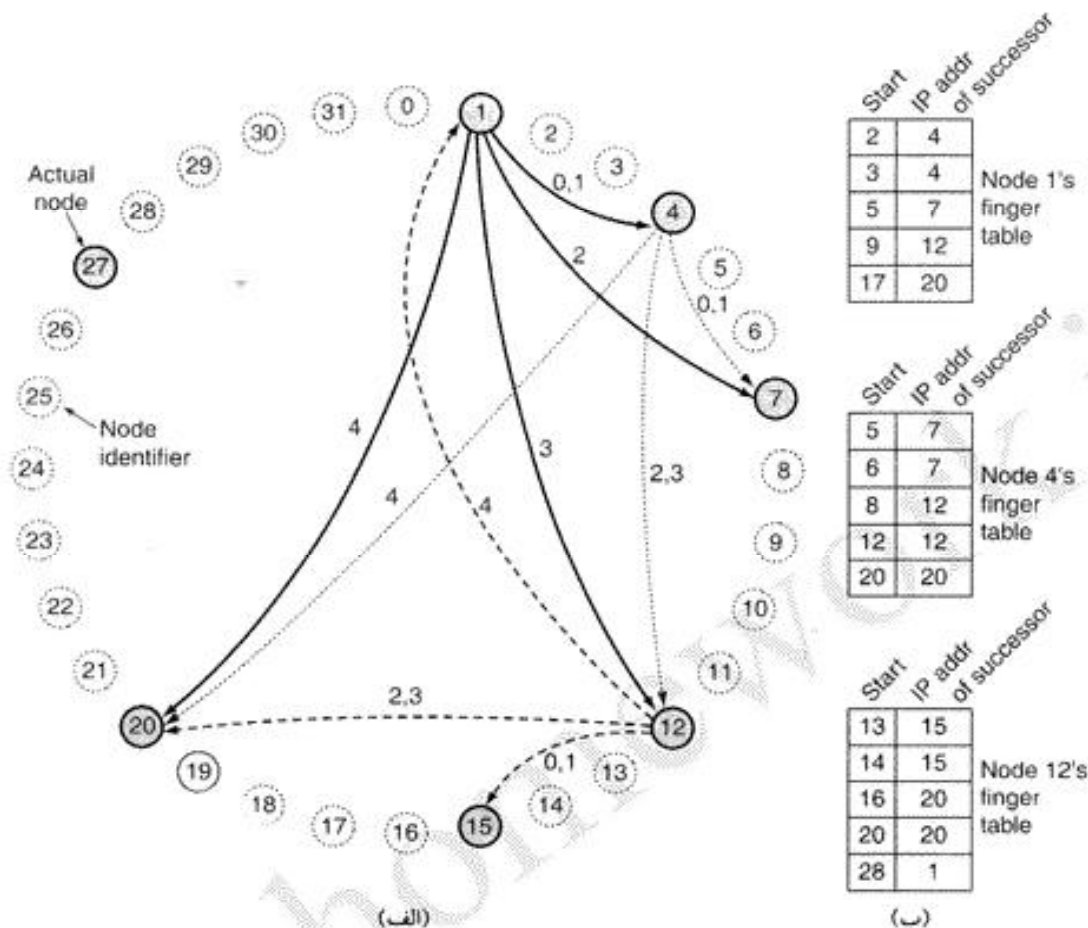
حال مراحل جستجو به ترتیب زیر ادامه می یابد: گره متقاضی بسته ای را برای گره بعدی خود ارسال کرده و ضمن اعلام آدرس IP خود، کلید مورد جستجو را مشخص می نماید. این بسته حول این دایره منتشر می شود تا آن که گره مورد جستجو پیدا شود. هر گره بررسی می کند که آیا اطلاعات مورد جستجو و منطبق با کلید را در اختیار دارد یا خیر. اگر داشته باشد آن اطلاعات را مستقیماً به آدرس IP متقاضی ارسال می نماید و در غیر این صورت بسته تقاضا را برای گره بعدی خود در دایره می فرستد.

اولین بهینه سازی این سیستم آن است که هر گره آدرس IP گره قبلی و بعدی خود را داشته باشد تا تقاضاها بتوانند در دو جهت ساعتگرد و پادساعتگرد (بسته به آن که کدام مسیر کوتاهتر است)، ارسال شوند. به عنوان مثال

۱. در حقیقت شناسنامه هر رکورد مثل فایل های صدا، تصویر یا برنامه، که شامل نام آن و آدرس IP ماشین نگهدارنده آنست، بر

۲. Administrative Data Structure.

روی ماشینهای دیگری قرار می گیرد. -م



شکل ۵-۲۴. (الف) مجموعه‌ای از ۳۲ شناسه گره که پیرامون یک دایره مرتب شده‌اند. گره‌های خاکستری رنگ متناظر با یک ماشین واقعی هستند. فلش‌ها اشاره گره‌های Finger جدول را نشان می‌دهند. برجسبهای هر فلش اندیسهای جدول Finger هستند. (ب) مثالی از جداول Finger

در شکل ۵-۲۴، گره ۷ می‌تواند برای یافتن گره ۱۰ در جهت ساعتگرد اقدام کند در حالی که برای یافتن گره ۳ می‌تواند به صورت پادساعتگرد عمل نماید.

حتی با وجود دو جهت برای جستجو، در سیستمهای عظیم «همتا به همتا» جستجوی خطی (Linear Search) بسیار ناکارآمد و کند عمل می‌کند چرا که در هر جستجو بطور متوسط باید $n/2$ گره بررسی شود. برای بالا بردن سرعت جستجو، هر گره دارای جدولی است که در سیستم Chord اصطلاحاً Finger Table نامیده شده است. جدول Finger حاوی m درایه (Entry) است که از صفر تا $m-1$ شماره‌گذاری می‌شود و هر یک از آنها به آدرس گره‌های واقعی اشاره می‌کنند. هر یک از این درایه‌ها دارای دو فیلد هستند: فیلد start و آدرس IP گره‌ای که شناسه آن معادل $successor(start)$ است. در شکل ۵-۲۴ ب، جدول Finger برای سه گره نمونه نشان داده شده است. مقادیر فیلدهای مربوط به درایه i در گره k عبارتست از:

$$start = k + 2^i \quad (\text{modulo } 2^m)$$

IP address of $successor(start[i])$

دقت کنید که هر گره، آدرس IP تعداد نسبتاً کمی از گره‌ها را در خود ذخیره می‌کند و اغلب این گره‌ها دارای شماره

شناسه نزدیک بهم هستند. [یعنی در هر گره آدرس IP گره هایی نگهداری می شوند که بر روی دایره شناسه ها، تقریباً در کنار آن گره باشند. -م]

با استفاده از جدول Finger، جستجوی کلید Key در گره k به ترتیب زیر انجام می شود: اگر مقدار key بین k و $successor(k)$ باشد، گره ای که اطلاعاتی در خصوص key در اختیار دارد همان $successor(k)$ است و جستجو خاتمه می یابد. در غیر اینصورت در جدول Finger جستجو می شود تا درایه ای که در آن، مقدار فیلد $start$ ، به $predecessor(k)$ [یعنی شناسه گره قبلی k در دایره] نزدیکتر است پیدا شود. این تقاضا مستقیماً به آدرس IP گره فوق الذکر ارسال می شود تا ادامه جستجو در آنجا پیگیری شود. (آدرس IP گره مذکور از جدول Finger بدست می آید.) از آنجایی شناسه این گره نزدیک به key و در عین حال کمتر از آن است لذا این شانس خوب وجود دارد که با تعداد کمی پرس و جوی دیگر نتیجه مورد نظر بدست آید. در حقیقت چون که در هر جستجو فاصله باقیمانده تا هدف مورد نظر نصف می شود می توان نشان داد که متوسط تعداد جستجوها $\log_2 n$ است.

به عنوان اولین مثال فرض کنید در گره ۱، کلید $key=3$ را جستجو می کنیم. از آنجایی که گره ۱ می داند که ۳ بین خودش و گره بعدی او یعنی ۴ واقع شده، لذا گره مورد نظر همان ۴ است و جستجو خاتمه یافته و آدرس IP گره ۴ بدست می آید.

در مثال دوم، فرض کنید در جستجوی کلید $key=14$ بر روی گره ۱ هستیم. چون که ۱۴ بین ۱ و ۴ نیست، از جدول Finger کمک گرفته می شود. نزدیکترین عدد قبل از ۱۴، ۹ است بنابراین، تقاضا به سوی آدرس IP گره حقیقی پس از ۹ که در جدول ۱۲ تعیین شده است هدایت می شود. گره ۱۲ می بیند که گره ۱۴ بین خودش و گره بعدیش یعنی ۱۵ قرار گرفته، لذا آدرس IP گره ۱۵ را برمی گرداند.

به عنوان مثال سوم، فرض کنید بر روی گره ۱ به دنبال کلید $key=16$ هستیم. مجدداً این تقاضا برای گره ۱۲ ارسال می شود ولی در اینجا گره ۱۲ پاسخ این تقاضا را نمی داند و به همین دلیل نزدیکترین شماره قبل از ۱۶ یعنی ۱۴ را یافته و از طریق جدول خود آدرس IP گره ۱۵ را بدست می آورد؛ سپس تقاضا برای آن گره ارسال می شود. گره ۱۵ می بیند که عدد ۱۶ بین خودش و گره بعدی یعنی ۲۰ قرار دارد لذا آدرس IP گره ۲۰ را به سوال کننده بر می گرداند و به همین ترتیب کار ادامه می یابد تا به گره ۱ برسد.

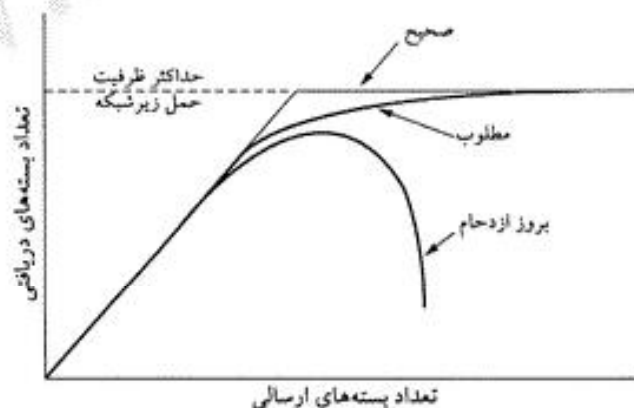
از آنجایی که گره ها بطور متوالی به شبکه می پیوندند یا از آن جدا می شوند فلذا Chord نیازمند روشی است که بتواند این مسئله را نیز حل و فصل کند. فرض را بر آن می گذاریم که وقتی این سیستم آغاز به کار کرده تعداد گره ها آنقدر کم بوده که توانسته اند مستقیماً با یکدیگر مبادله اطلاعات کرده و اولین دایره و جدول Finger را بسازند. پس از آن، به یک روال خودکار نیاز است: وقتی گره جدید ۲ می خواهد به این سیستم بپیوندد باید با یکی از گره های موجود تماس برقرار کرده و از او بخواهد که آدرس IP گره $successor(r)$ را برایش پیدا کند. پس از پیدا شدن، گره جدید از $successor(r)$ می خواهد که گره قبلی خود خود را معرفی کند. در آخر، گره جدید از این دو گره می خواهد که او را به عنوان گره ۲ در دایره وارد کند. به عنوان مثال اگر در شکل ۵-۲۴ گره ۲۴ بخواهد به سیستم بپیوندد از یکی از گره های فعال تقاضا می کند که $successor(24)$ را برایش پیدا نماید که در این جا ۲۷ است. سپس از ۲۷ در مورد گره ماقبل او یعنی ۲۰ سؤال می کند. پس از آن که حضور خود را به این دو گره (یعنی ۲۰ و ۲۷) اعلام کرد گره ۲۰ را به عنوان گره بعدی خود و گره ۲۷ آن را به عنوان گره قبلی خود به رسمیت می شناسند. مضاف بر این، گره ۲۷ آن دسته از درایه هایی را که کلیدشان در محدوده ۲۱ تا ۲۴ است به گره ۲۴ تقدیم می دارد. در این لحظه گره ۲۴ بطور کامل به جمع پیوسته است. ولیکن در این لحظه بسیاری از جداول Finger غلط هستند. برای اصلاح این جداول، هر گره یک پروسه را که در پس زمینه اجرا می شود، اجرا می کند تا جداول مربوطه (با فراخوانی متناوب تابع $successor$) از نو محاسبه و اصلاح شوند. هر گاه در خلال عملیات تازه سازی، یکی از این

تقاضاها به گره جدیدی بر بخورد درایه مربوط به آن نیز در جدول Finger بهنگام می شود. وقتی یک گره به صورت مسالمت آمیز سیستم را ترک می کند کلیدهای خود را به گره بعدی خود در حلقه تسلیم می کند تا به او اطلاع بدهد که در آستانه خروج از سیستم است و آن گره بتواند پیوند خود را با گره ماقبل از گره در آستانه خروج، برقرار نماید. وقتی گره ای به ناگاه از کار بیفتد مشکل پیش می آید چراکه گره ماقبل از گره خراب شده، هیچ گره معتبر و فعال بعدی ندارد. برای کاهش اثر این مشکل، هر گره نه تنها آدرس مستقیم گره بعدی خود را نگه می دارد بلکه آدرس مستقیم s گره بعد از خود را نیز دارد تا حتی اگر s-1 گره متوالی از کار بیفتد باز هم بتوان دایره را اصلاح کرد.

از سیستم Chord برای ایجاد سیستم توزیع شده فایبل (Dabek et al., 2001) و چند کاربرد دیگر استفاده شده و تحقیقات بر روی آن کماکان ادامه دارد. یک سیستم همتابه همتای دیگر به نام Pastry و کاربردهای آن در مرجع (Rowstron and Druschel, 2001) تشریح شده است. سیستم سومی به نام Freenet نیز معرفی شده که مستندات آن در (Clark et al., 2002) در دسترس عموم قرار دارد. چهارمین سیستم از این نوع نیز در مرجع (Ratnasamy, 2001) تشریح شده است.

۳-۵ الگوریتمهای کنترل ازدحام

وقتی به بخشی از زیر شبکه، تعداد بسیار زیادی بسته تحویل شود کارایی آن کاهش می یابد. بدین وضعیت «ازدحام» (Congestion) گفته می شود. شکل ۵-۲۵، علائم بروز این وضعیت را نشان می دهد. هر گاه تعداد بسته هایی که توسط ماشینهای میزبان به زیر شبکه سرازیر می شوند متناسب با ظرفیت حمل زیر شبکه باشد تمام این بسته ها تحویل مقصدشان خواهند شد (به استثنای آنهایی که در اثر خطای انتقال آسیب می بینند) و تعداد بسته های تحویلی متناسب با تعداد بسته های ارسالی است. ولیکن به محض افزایش بی رویه ترافیک، مسیرها به قدری شلوغ می شوند که بسته ها شروع به از دست رفتن می کنند. این مسئله حادث تر نیز می شود و در ترافیک بسیار بالا کارایی بطور کامل سقوط کرده و هیچ بسته ای تحویل مقصد نخواهد شد!



شکل ۵-۲۵. وقتی ترافیک تحویلی به شبکه بیش از اندازه باشد ازدحام بوجود می آید و کارایی بشدت افت می کند.

چندین عامل می تواند به بروز ازدحام بینجامد. اگر به ناگاه دنباله ای از بسته ها بر روی سه یا چهار خط ورودی دریافت شده و خط خروجی همه آنها یکی باشد صف تشکیل خواهد شد و اگر فضای حافظه کافی برای نگهداری تمام آنها وجود نداشته باشد، بسته ها از بین می روند. شاید اضافه کردن حافظه مفید به نظر برسد ولیکن «ناگل» (Nagle, 1987) بدین نتیجه رسید که حتی اگر حافظه مسیر یاب نامحدود باشد، وضعیت ازدحام نه تنها بهتر

نمی شود بلکه بدتر هم خواهد شد زیرا بسته ها، زمانی به سر صف می رسند که مهلت رسیدنشان به مقصد تمام شده و نسخه های تکراری آن ارسال شده اند. تمام این بسته های تکراری نیز برحسب وظیفه شناسی به مسیریاب بعدی هدایت شده و در تمام مسیر رسیدن به مقصد، «بار» افزایش می یابد. [عبارت دیگر، «ازدحام» قابلیت انتشار در کل یک مسیر دارد.]

پردازنده های کُند نیز می توانند عامل بروز ازدحام باشند. اگر پردازنده اصلی مسیریاب در انجام وظائف محوله به خود (مثل عملیات صف بندی، بهنگام سازی جداول مسیریابی و نظائر آن) کُند عمل نماید، صف ایجاد می شود، حتی وقتی که ظرفیت خطوط خروجی بیش از حد مورد نیاز است.

به دلیل مشابه، خطوط با پهنای باند کم نیز می توانند منجر به بروز ازدحام شوند. ارتقاء پهنای باند خطوط بدون تغییر در پردازنده ها یا بالعکس اغلب فایده چندانی ندارد و فقط جای گلوگاه و منشاء مشکل را تغییر می دهد. همچنین ارتقاء کامل یک بخش کوچک [مثل تغییر یک مسیریاب و ارتقاء ظرفیت خطوط آن] بدون تغییر در کل سیستم، فقط محل بروز مشکل و گلوگاه را جابجا می کند و در بهبود کارایی کل سیستم تأثیر چشمگیری نخواهد داشت. مشکل اساسی یک سیستم، عدم تطابق و تناسب بخشهای مختلف آنست. این مشکل تا زمانی که کلیه مؤلفه های سیستم متعادل نشوند باقی خواهد ماند.

باید به صراحت تفاوت بین «کنترل ازدحام»^۱ و «کنترل جریان»^۲ و همچنین ارتباط ظریف این دو مکانیزم را تبیین کنیم. «کنترل ازدحام» مکانیزمی است جهت ایجاد اطمینان از این که زیر شبکه قادر به حمل ترافیک عرضه شده به آن هست. این مشکل یک مورد همگانی و سراسری است و از رفتار و عملکرد تمام ماشینهای میزبان، کلیه مسیریابها، عملیات پردازشی «ذخیره و هدایت» (Store & Forward) درون مسیریابها یا هر عامل دیگری که ظرفیت حمل زیر شبکه را کاهش بدهد، ناشی می شود.

در مقابل، «کنترل جریان» به ترافیک نقطه به نقطه بین یک فرستنده و گیرنده مفروض مربوط می شود و وظیفه اصلی آن ایجاد اطمینان از این موضوع است که یک فرستنده سریع نمی تواند متوالیاً داده ها را با سرعتی بیش از توانایی دریافت گیرنده، ارسال نماید. در کنترل جریان، گزارشات و فیدبکهای مستقیمی از گیرنده به فرستنده ارسال می شود تا به فرستنده طرف مقابل تفهیم کند که کارها را چگونه انجام بدهد.

برای درک اختلاف بین این دو مفهوم، یک شبکه فیبرنوری با ظرفیت 1000 Ggigabits/sec را در نظر بگیرید که در آن یک ابرکامپیوتر سعی می کند با سرعت یک گیگابایت بر ثانیه فایلی را برای یک کامپیوتر شخصی، ارسال کند! اگرچه هیچگونه ازدحامی رخ نخواهد داد (خود شبکه مشکلی ندارد) ولیکن برای آن که بتوان به نحوی ابرکامپیوتر را وادار کرد که هرازگاهی متوقف شود و اجازه نفس کشیدن به کامپیوتر شخصی بدهد به مکانیزمهای کنترل جریان نیاز است.

در سمت مقابل، یک شبکه «ذخیره و هدایت» با خطوط 1Mbps و هزار کامپیوتر بزرگ را در نظر بگیرید که در آن نیمی از کامپیوترها سعی می کنند با سرعت 100Kbps برای نیم دیگر، فایل ارسال کنند! در اینجا مشکل تحت فشار قرار گرفتن گیرنده کُند توسط فرستنده سریع مطرح نیست بلکه مسئله آنست که ترافیک تحمیل شده به شبکه از میزان ظرفیت و توانایی آن بیشتر است.

دلیل آن که مفهوم کنترل ازدحام، اغلب با مفهوم کنترل جریان اشتباه می شود آن است که در برخی از الگوریتمهای کنترل ازدحام، هنگام بروز مشکل برای شبکه، پیغامهایی را برای ماشینهای مبداء ارسال کرده و به آنها تفهیم می کند که باید سرعت خود را پایین بیاورند. بنابراین یک ماشین میزبان ممکن است به دو دلیل پیغام

Slowdown (آهسته تر ارسال کن) را دریافت کند: اول آن که گیرنده نتواند با همان سرعتی که فرستنده ارسال می کند داده ها را دریافت نماید. دوم آن که شبکه با ازدحام مواجه شود و از عهده ارسال داده ها بر نیاید. بعداً باز هم به این موضوع بر می گردیم.

مطالعات خود پیرامون کنترل ازدحام را با نگاهی به مدل و روشهای عمومی برخورد با آن آغاز می کنیم. سپس به راهکارهای گسترده ای خواهیم پرداخت که سعی می کنند در همان ابتدا از بروز این مشکل پیشگیری کنند. همچنین الگوریتمهای پویایی را مرور می کنیم که پس از بروز مشکل ازدحام، سعی در حل و فصل آن می کنند.

۵-۳-۱ اصول کلی در کنترل جریان

بسیاری از مشکلات و مسائل سیستمهای پیچیده مثل شبکه های کامپیوتری را می توان از دیدگاه نظریه کنترل بررسی کرد. در این روش تمام راه حلها به دو گروه تقسیم می شوند: «حلقه باز» و «حلقه بسته»^۱. راه حلهای «حلقه باز» سعی می کنند یک مسئله را با طراحی خوب حل کرده و از همان ابتدا اطمینان بدهند که مشکلی رخ نخواهد داد. وقتی این سیستم شروع به کار و انجام فعالیت نمود هیچگونه نظارت یا تصحیح عملکرد ممکن نیست.

تعمیداتی که برای کنترل ازدحام به سبک «حلقه باز» پیش بینی شده عبارتند از: تصمیم گیری در خصوص زمان پذیرش ترافیک جدید، تصمیم گیری در خصوص زمان حذف بسته ها، انتخاب بسته هایی که باید حذف شوند و تصمیم گیری در خصوص زمان بندی صحیح در شبکه؛ حقیقت مشترک در تمام این موارد آنست که تصمیم گیری آنها مبتنی بر شرایط جاری شبکه نیست.

در مقابل، راه حلهای «حلقه بسته» مبتنی بر مفهوم حلقه های فیدبک هستند. اینگونه راهکارهای کنترل ازدحام، سه بخش را در بر می گیرند:

۱. نظارت بر سیستم به منظور تشخیص آنکه در کجا و چه وقت ازدحام رخ داده است.

۲. تحویل این اطلاعات به محلی که بایستی واکنش نشان بدهد.

۳. تنظیم عملکرد سیستم برای رفع مشکل

برای نظارت و تشخیص بروز ازدحام در زیر شبکه می توان معیارهای گوناگونی را بکار گرفت. مهمترین آنها عبارتند: درصد بسته هایی که به دلیل فقدان فضای کافی بافر حذف می شوند، متوسط طول صف، تعداد بسته هایی که مهلت ارسال آنها منقضی شده و از نو ارسال گردیده اند، متوسط تأخیر بسته و انحراف معیار^۲ تأخیر بسته. در تمام این معیارها رشد اعداد نمایانگر افزایش ازدحام است.

دومین مرحله از «حلقه فیدبک» آن است که اطلاعاتی در خصوص ازدحام، از محل تشخیص و بروز آن به محلی که می تواند کاری برای حل آن انجام بدهد، انتقال یابد. بدیهی ترین روش آن است که مسیریاب کشف کننده ازدحام بسته ای خاص برای ماشین یا ماشینهای مبدا، بفرستد و مشکل را به آنها اعلام نماید. البته این بسته های اضافی خودشان به بار شبکه می افزایند، آن هم دقیقاً زمانی که شبکه ظرفیت هیچ بار اضافی ندارد یعنی دقیقاً حین ازدحام در زیر شبکه!

با این وجود روشهای دیگری نیز امکان پذیر است. به عنوان مثال می توان یک بیت یا یک فیلد در هر بسته کنار گذاشت تا هر گاه ازدحام از یک حد آستانه فراتر رفت، مسیریاب آنرا پر کند. هر گاه مسیریاب تشخیص بدهد که ازدحام رخ داده در تمام بسته های خروجی، این فیلد را پر می کند تا به همسایه های خود در این خصوص هشدار بدهد.

راهکار دیگر آن است که ماشینهای میزبان یا مسیریابها بطور متناوب بسته های «آزمون» تولید و ارسال کرده و

۱. Open loop , Close loop

۲. Standard deviation

مستقیماً در خصوص ازدحام کسب آگاهی کنند. بکمک این اطلاعات می توان بسته ها را بنحوی مسیریابی کرد که از ناحیه بروز مشکل، ردّ نشوند. به مثابه ایستگاههای کنترل ترافیک که هلی کوپترهای آنها بر روی شهرها به پرواز درمی آیند تا به شنودگان در حال حرکت هشدار بدهند تا بسوی نقاط بحران، حرکت نکنند.

در تمام روشهای مبتنی بر فیدبک، انتظار می رود که آگاهی از ازدحام موجب شود ماشینهای میزبان برای کاهش ازدحام از خود واکنش مناسبی نشان بدهند. برای آنکه این روشها به درستی کار کنند بایستی مقیاس زمان به دقت تنظیم شده باشد. اگر وقتی دو بسته پشت سر هم می رسند مسیریاب فریاد بزند «ایست» و به محض آنکه فقط برای ۲۰ میکروثانیه آزاد می شود فریاد بزند: «حرکت»، سیستم بشدت نوسانی عمل می کند و هیچگاه همگرا و پایدار نخواهد شد. برعکس اگر قبل از اعلام هر چیزی برای اطمینان به مدت ۳۰ دقیقه صبر کند، مکانیزم کنترل ازدحام آنقدر کند واکنش نشان می دهد که عملاً هیچ سودی در دنیای واقعی نخواهد داشت. برای آنکه عملکرد خوبی انتظار داشته باشیم به حالت میانه ای نیازمندیم ولیکن بدست آوردن «ثابت زمانی سیستم» (Time Constant) مسئله چندان ساده ای نیست.

الگوریتمهای متعددی برای کنترل ازدحام معرفی شده اند. برای آنکه بتوان این الگوریتمها را به روش قابل فهمی سازماندهی کرد، Yang و Reddy (۱۹۹۵) الگوریتمهای کنترل ازدحام را طبقه بندی کرده اند. آنها به همان ترتیبی که در بالا تشریح شد تمام الگوریتمها را به دو رده «حلقه باز» و «حلقه بسته» تقسیم بندی نمودند. مضاف بر این، الگوریتمهای «حلقه باز» را برحسب اینکه در ماشینهای مبدا عمل می کنند یا در ماشینهای مقصد به دو رده تقسیم کردند. الگوریتمهای «حلقه بسته» نیز به دو رده فرعی تقسیم شده اند: «الگوریتمهای مبتنی بر فیدبک مستقیم» و «الگوریتمهای مبتنی بر فیدبک ضمنی». در الگوریتمهای مبتنی بر فیدبک مستقیم، برای هشدار دادن به مبدا، بسته های فیدبک دقیقاً از محل بروز ازدحام برگردانده می شوند. در الگوریتمهای مبتنی بر فیدبک ضمنی، مبدا با استناد به برخی از مشاهدات و علائم محلی (همانند زمان لازم برای برگشت بسته های ACK) بروز ازدحام را حدس می زند.

بروز ازدحام بدین معناست که «بار شبکه» موقتاً از «منابع» موجود در برخی از بخشهای سیستم بیشتر شده و شبکه از عهده این بار بر نمی آید. [منظور از منابع پهنای باند، توان پردازشی CPU، حافظه مسیریاب و نظایر آنهاست. سم] دو راه حلّ به ذهن متبادر می شود: افزایش منابع یا کاهش بار. به عنوان مثال زیر شبکه می تواند برای افزایش پهنای باند بین دو نقطه، موقتاً از یک خطّ تلفن (Dialup) کمکی بهره بگیرد. در سیستمهای ماهواره ای، افزایش توان فرستنده اغلب پهنای باند را افزایش می دهد. در ضمن اگر بجای استفاده دائمی از بهترین مسیر، بخشی از بار بر روی مسیرهای دیگر تقسیم شود پهنای باند شبکه بطور مؤثری افزایش می یابد. نهایتاً آنکه می توان مسیربایهای یدکی را که فقط به عنوان پشتیبان در شبکه نصب شده اند (برای آنکه سیستم تحمل خرابی داشته باشد) به خدمت گرفت تا در هنگام بروز ازدحام، ظرفیت زیر شبکه افزایش یابد.

علیرغم این راهکارها، بسیاری از اوقات افزایش ظرفیت زیر شبکه ممکن نیست یا آنکه این ظرفیت تا آخرین حدّ افزایش یافته است. تنها راه باقیمانده برای رفع ازدحام کاهش بار است. چندین روش برای کاهش بار وجود دارد که از آن جمله می توان به این موارد اشاره کرد: (۱) اجتناب از سرویس دادن به برخی از کاربران (۲) کاهش سطح سرویس دهی به برخی یا تمام کاربران (۳) مجبور کردن کاربران به زمان بندی تقاضاهای خود به روشی قابل پیش بینی.^۱

۱. از آنجایی که برخی از ماشینها ترافیک انفجاری و غیر قابل پیش بینی تولید می کنند لذا می توان آنها را وادار کرد تا بکمک مکانیزمهای خاص (همانند مکانیزم بافرینگ یا مکانیزمهای شکل دهی به ترافیک) حجم ترافیک خود را متعادل و قابل پیش بینی کنند. سم

برخی از این روشها که به اختصار آنها بررسی خواهیم کرد در زیر شبکه های نوع «مدار مجازی» به بهترین نحو قابل اعمال هستند. در زیر شبکه های مدار مجازی، این روشها در لایه شبکه پیاده می شوند. در زیر شبکه های دیتاگرام علیرغم اصراری که بر تفکیک وظایف لایه ها وجود دارد بخشی از این روشها باید در اتصالات لایه انتقال بکار گرفته شوند. در این فصل به کاربرد این روشها در لایه شبکه می پردازیم و در فصل بعدی به عملیات مدیریت ازدحام در لایه انتقال، نگاهی خواهیم انداخت.

۲-۳-۵ سیاستهای پیشگیری از ازدحام

اجازه بدهید مطالعه روشهای کنترل ازدحام را با بررسی سیستمهای «حلقه باز» شروع کنیم. اینگونه سیستمها به گونه ای طراحی می شوند تا بجای آنکه بگذارند ازدحام اتفاق بیفتد و بعداً واکنش نشان بدهند، در همان ابتدا ازدحام را به حداقل برسانند. این روشها برای رسیدن بدین هدف، از سیاستهای خاص و مناسب در سطوح مختلف، بهره می گیرند. در شکل ۵-۲۶ سیاستهای مختلفی که در لایه پیوند داده، لایه شبکه و لایه انتقال، می تواند بر پدیده ازدحام تأثیر بگذارد، فهرست شده اند. (Jain, 1990)

سیاستها	لایه
- سیاستهای ارسال مجدد - سیاستهای ذخیره بسته هایی که خارج از ترتیب می رسند. - سیاستهای تصدیق وصول بسته ها (Ack) - سیاستهای کنترل جریان - تعیین زمان انقضای مهلت تایمرها	انتقال
- مدار مجازی در مقابل روش دیتاگرام در زیر شبکه - مکانیزمهای صف بندی بسته ها و روشهای متنوع سرویس دهی - سیاستهای حذف بسته - الگوریتم مسیریابی - مدیریت طول عمر بسته ها	شبکه
- سیاستهای ارسال مجدد - سیاستهای ذخیره بسته هایی که خارج از ترتیب می رسند. - سیاستهای تصدیق وصول بسته ها (Ack) - سیاستهای کنترل جریان	پیوند داده

شکل ۵-۲۶. سیاستهایی که بر پدیده ازدحام تأثیر می گذارند.

اجازه بدهید از لایه پیوند داده شروع کرده و سپس به سمت بالا حرکت کنیم. سیاستهای ارسال مجدد (Retransmission Policy) با مسائلی از این قبیل سروکار دارد: (۱) با چه سرعتی مهلت فرستنده (برای دریافت Ack یک بسته) خاتمه می یابد؟ (۲) در اثر انقضای مهلت چه چیزی ارسال می شود؟

یک فرستنده عجول که مهلت آن به سرعت خاتمه می یابد و مبتنی بر پروتکل Go Back n تمام بسته های ارسالی را از نو می فرستد، بار بسیار سنگینتری را نسبت به فرستنده ای که از روش Selective Repeat استفاده کرده به زیر شبکه تحمیل می کند. یکی دیگر از مسائل که ارتباط تنگاتنگی با این موضوع دارد سیاستهای بافرینگ است. اگر گیرنده، بسته هایی را که خارج از ترتیب می رسند دور بیندازد تمام آنها باید از نو ارسال شوند و بار اضافی به زیر شبکه تحمیل خواهد شد. از دیدگاه کنترل ازدحام، روش «تکرار انتخابی» (Selective Repeat) به وضوح بهتر از روش Go Back n عمل می کند.

سیاست تصدیق دریافت بسته ها (Acknowledgement Policy) نیز بر روی ازدحام تأثیر می گذارد. اگر هر

بسته فوراً اعلام وصول شود، بسته های اعلام وصول (Ack) ترافیک زائد و اضافی تحمیل خواهند کرد. در عوض اگر برای صرفه جویی، از روش Piggybacking استفاده شود و اعلام وصول داده ها از طریق ترافیک معکوس انجام شود ممکن است مهلت فرستنده داده ها متوالیاً منقضی شود و تکرار ارسال رخ بدهد که آن هم منجر به تحمیل بار اضافه خواهد شد. استفاده از یک روش محکم و سختگیرانه در الگوی کنترل جریان (مثلاً داشتن پنجره کوچک) می تواند نرخ ارسال داده را کاهش داده و به کاهش ازدحام کمک کند.

در لایه شبکه، انتخاب بین روش «مدار مجازی» یا «دیتاگرام» تأثیر مستقیمی بر پدیده ازدحام دارد چرا که بسیاری از الگوریتم های کنترل ازدحام فقط در زیر شبکه های مدار مجازی قابل اعمال هستند. صف بندی بسته ها و سیاست های سرویس دهی نیز در کنترل ازدحام موثرند. این سیاست ها بدین موضوع مرتبطند که آیا مسیر یاب به ازای هر خط ورودی یک صف دارد، به ازای هر خط خروجی یک صف دارد یا هر دو صف را تشکیل می دهد. همچنین ترتیب پردازش بسته ها (مثلاً نوبت بندی چرخشی Round Robin- یا روش مبتنی بر اولویت بندی) جزو سیاست ها صف بندی محسوب می شود. «سیاست حذف بسته ها» قاعده ای است که براساس آن تعیین می شود که وقتی فضایی برای نگهداری بسته ها نیست کدامیک از بسته ها را باید حذف کرد. اتخاذ یک سیاست مناسب می تواند به کاهش ازدحام کمک کند و سیاست های غلط شرایط ازدحام را بدتر خواهد کرد.

یک الگوریتم مسیر یابی خوب می تواند با توزیع مناسب ترافیک بین خطوط مختلف به پیشگیری از ازدحام کمک کند در حالی که یک الگوریتم بد می تواند با ارسال ترافیک بسیار زیاد بر روی یک خط که با ازدحام مواجه شده شرایط را بدتر کند. مدیریت طول عمر بسته (Packet Lifetime) با این مسئله سر و کار دارد که چه مدت طول می کشد تا بسته [در صورت نرسیدن به مقصد] حذف شود. اگر این زمان بیش از حد طولانی باشد بسته های گم شده و سرگردان ممکن است تا قبل از موعد حذف شدن، شبکه را با انباشتگی و ازدحام مواجه کنند، در حالی که اگر این زمان کوتاه در نظر گرفته شود ممکن است بسته ها قبل از آنکه فرصت رسیدن به مقصد را داشته باشند حذف شوند و به تکرار ارسال بینجامد.

در لایه انتقال همان موارد و سیاست هایی که در لایه پیوند داده وجود داشت دنبال می شود ولی مضاف بر آنها روش تعیین بازه های زمانی انقضای مهلت (Timeout Interval) پیچیده تر است چرا که زمان عبور بسته ها از میان یک شبکه [با تعدادی مسیر یاب و کانال های متعدد] در مقایسه با عبور بسته ها از یک سیم واحد، چندان قابل پیش بینی نیست. اگر بازه های انقضای مهلت بیش از اندازه کوتاه باشد بسته های بی مصرف زیادی ارسال خواهد شد. اگر این زمان بیش از اندازه طولانی باشد، ازدحام کاهش می یابد ولیکن وقتی بسته ای به هر دلیل از دست برود زمان پاسخ [یعنی تأخیر ارسال مجدد] مشکل آفرین خواهد شد.

۳-۳-۵ کنترل ازدحام در زیر شبکه های مدار مجازی

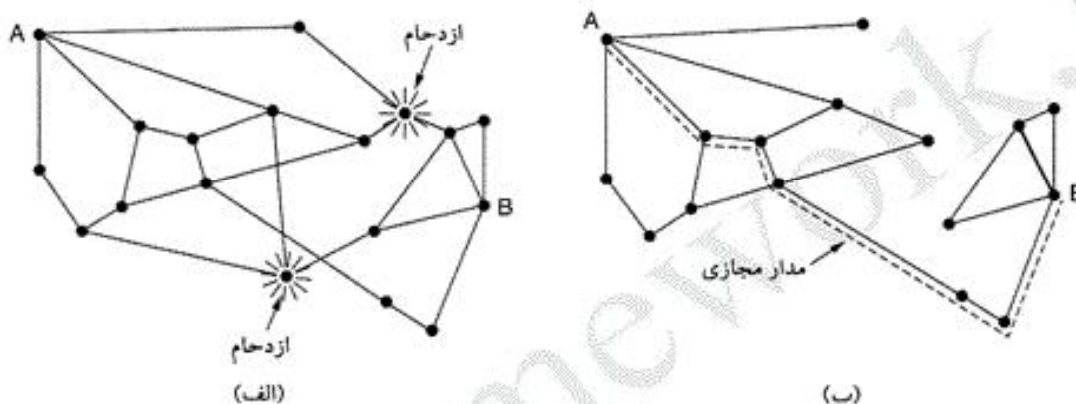
روش های کنترل ازدحام که در بالا معرفی شدند اساساً «حلقه باز» هستند یعنی تلاش می کنند در همان ابتدا و قبل از بروز، از ازدحام پیشگیری کنند. در این بخش راهکارهایی را بررسی خواهیم کرد که در زیر شبکه های مدار مجازی به صورت پویا ازدحام را کنترل می کنند. در دو بخش آتی نیز نگاهی به تکنیک های قابل استفاده در هر نوع زیر شبکه خواهیم انداخت.

یکی از تکنیک هایی که بطور گسترده ای از آن در جهت پیشگیری از وخیم شدن مشکل ازدحام (که از قبل شروع شده) استفاده می شود روش «کنترل پذیرش»^۱ (Admission Control) نام دارد. ایده این روش ساده است: هرگاه ازدحام گزارش شود تا رفع کامل مشکل، هیچگونه تقاضای تنظیم مدار مجازی پذیرفته نخواهد شد. بدین

۱. بمعنای محدود کردن پذیرش ارتباط و امتناع از ایجاد مدار مجازی جدید

ترتیب هر گونه تلاش برای ایجاد اتصال در لایه انتقال با شکست مواجه خواهد شد چراکه پذیرش افراد جدید وضع ازدحام را وخیم تر خواهد کرد. هر چند این روش خام و ناشیانه است ولی در عوض ساده و سهل الاجراست. در سیستمهای تلفن نیز هر گاه یک سونچ با بار بسیار زیاد مواجه شود، با قطع بوق آزاد (Dial Tone)، پذیرشهای جدید را محدود می نماید.

راهکار دیگر آن است که اجازه تنظیم مدار مجازی جدید داده شود ولیکن مسیرهای انتخابی برای مدارات مجازی جدید در خارج از نواحی بحران زده انتخاب گردد. به عنوان مثال زیر شبکه شکل ۵-۲۷-الف را در نظر بگیرید که در آن دو مسیر یاب با ازدحام مواجه شده اند.



شکل ۵-۲۷. (الف) یک زیر شبکه مواجه با ازدحام (ب) ترسیم مجدد گراف زیر شبکه پس از حذف مناطق درگیر ازدحام. مدار مجازی مناسب بین A و B نشان داده شده است.

فرض نمایید که ماشین میزبان متصل به مسیر یاب A بخواهد یک «اتصال» با ماشین متصل به B برقرار نماید. طبیعتاً این اتصال از یکی از مسیر یابهای دارای مشکل عبور خواهد کرد. برای اجتناب از این وضعیت می توانیم زیر شبکه را مطابق با شکل ۵-۲۷-ب از نو ترسیم کرده و مسیر یابهای دچار ازدحام و تمام خطوط آنها را حذف نماییم. خط نقطه چین، مسیر ممکن برای ایجاد مدار مجازی بین A و B را نشان می دهد. در این مسیر، مسیر یابها و خطوط دچار ازدحام وجود ندارد.

استراتژی دیگر در زیر شبکه های مدار مجازی، آنست که در حین تنظیم یک مدار مجازی، بین ماشین میزبان و زیر شبکه توافقاتی صورت بگیرد. در این توافقات، عموماً حجم و شکل ترافیک، کیفیت مورد نیاز خدمات و پارامترهای دیگر مشخص می شوند. به منظور عمل به مفاد این قرارداد، زیر شبکه در هنگام تنظیم مسیر، منابع مورد نیاز را در طول مسیر، از قبل کنار می گذارد. این منابع شامل فضای بافر، فضای جدول در هر مسیر یاب و پهنای باند خطوط است. در این روش احتمال بروز ازدحام در مدارات مجازی نادر است چرا که تمام منابع مورد نیاز از قبل رزرو و موجود بودن آن تضمین می شود.

این گونه رزرو سازی می تواند به صورت یک روال عملیاتی استاندارد همیشه انجام شود و یا فقط در شرایطی اتخاذ شود که شبکه با ازدحام مواجه شده است. اشکال استفاده همیشگی از این روش آن است که می تواند منابع را هدر بدهد. به عنوان مثال اگر شش مدار مجازی که هر یک به پهنای باند 1Mbps نیاز مندند همگی از یک خط فیزیکی 6Mbps عبور کرده باشند مسیر یاب مجبور به علامت گذاری آن به عنوان خط پر خواهد بود در حالی که به ندرت اتفاق می افتد که هر شش مدار مجازی بطور همزمان مورد استفاده قرار بگیرند. نتیجتاً در شرایط معمولی هزینه ای که برای کنترل ازدحام پرداخت می شود پهنای باند بلا استفاده (تلفاتی) است.

۵-۳-۴ کنترل ازدحام در زیر شبکه های دیتاگرام

حال اجازه بدهید به راهکارهایی بپردازیم که می توان از آنها در زیر شبکه های دیتاگرام (و همچنین زیر شبکه های مدار مجازی) بهره گرفت. هر مسیریاب می تواند پراحتی بر میزان بهره وری (Utilization) خطوط خروجی و دیگر منابع خودش نظارت داشته باشد. به عنوان مثال مسیریاب می تواند به هر یک از خطوط خود یک متغیر اعشاری u که مقداری بین 0.0 تا 1.0 دارد نسبت بدهد. مقدار این متغیر میزان بهره وری خط را منعکس می کند. برای آنکه بتوان تخمین دقیقی از u داشت می توان بطور متناوب و در لحظات خاص از میزان بهره وری خط^۱ نمونه برداری کرد و با فرض آنکه این مقدار f محاسبه شده باشد (f نیز بین صفر و یک است)، مقدار u را طبق رابطه زیر بهنگام سازی نمود:

$$u_{\text{new}} = \alpha \cdot u_{\text{old}} + (1 - \alpha) \cdot f$$

α یک ثابت است که تعیین می کند مسیریاب با چه سرعتی وضعیت گذشته را فراموش خواهد کرد.

هر گاه u از یک مقدار آستانه (Threshold) [یا به عبارتی از حد مجاز] تجاوز کند آن خط خروجی در وضعیت «هشدار» وارد می شود و در این صورت باید عملیاتی برای خروج از این وضعیت انجام گیرد. این عملیات می تواند یکی از گزینه های زیر باشد که در ادامه آنها را تشریح خواهیم کرد.

بیت هشدار (The Warning Bit)

در معماری قدیم شبکه های DECNET، «وضعیت هشدار» با تنظیم یک بیت خاص در سرآیند بسته ها، اعلام می شد. در شبکه Frame Relay نیز همینگونه است. وقتی بسته ای به مقصد خود می رسد، لایه انتقال همان بیت را در درون بسته Ack کپی کرده و آن را برای مبدا پس می فرستاد و بدین نحو ماشین مبدا ترافیک خود را تقلیل می داد.

مادامیکه مسیریاب در وضعیت هشدار قرار داشت، تنظیم این بیت نیز ادامه می یافت بدین معنا که ماشین مبدا در بسته های بعدی Ack باز هم این بیت را دریافت می کرد. مبدا نیز با شمارش بسته های Ack حساب می کرد که در چه کسری از این بسته ها بیت هشدار وجود دارد و براساس آن نرخ انتقال خود را تنظیم می نمود. تا وقتی که دریافت این بسته ها ادامه می یافت مبدا نیز به کاهش نرخ ارسال خود ادامه می داد. پس از کاهش نرخ ارسال تا حد نهایی، مجدداً نرخ ارسال شروع به افزایش می کرد [در صورت خروج از وضعیت هشدار]. دقت کنید که چون هر مسیریاب واقع بر روی مسیر می توانست «بیت هشدار» را به ۱ تنظیم کند لذا فقط زمانی ترافیک افزایش می یافت که هیچ یک از مسیریابها مشکلی نداشتند.

بسته های دعوت به آرامش (Choke Packet)

الگوریتم کنترل ازدحام قبلی نسبتاً زیرکانه است چرا که با یک بیان غیر مستقیم و تلویحی به مبدا تفهیم می کند که باید از سرعت خود بکاهد. چرا این کار مستقیماً انجام نشود؟ برای این کار مسیریاب یک بسته به نام «دعوت به آرامش» (Choke Packet) به ماشین مبدا بر گردانده و در آن آدرس مقصد بسته را نیز درج می کند [تا ماشین مبدا آگاه شود بسته های ارسالی او در راه رسیدن به کدام مقصد در مسیر پر ازدحام قرار گرفته اند و بداند که نرخ ارسال برای چه مقصدی را باید کاهش بدهد چرا که یک ماشین ممکن است بطور همزمان برای چند ماشین بسته ارسال کند -م]. البته بسته اصلی، علامتگذاری شده و به راه خود ادامه می دهد (فقط یک بیت خاص در سرآیند

۱. نسبت داده های ارسالی بر روی خط به ظرفیت کل خط را بهره وری آن خط در نظر بگیرید. اندازه گیری این نسبت برای هر خط چندان دشوار نیست و برخی از سخت افزارهای شبکه امکان این اندازه گیری را در اختیار می گذارند. -م

بسته تنظیم می شود). این علامتگذاری از آن جهت انجام می شود که بسته های «دعوت به آرامش» بیشتری در طول مسیر تولید نشود؛ سپس بسته اصلی بروش معمول به سوی مقصد خود هدایت می شود.

هر گاه ماشین مبدا، بسته دعوت به آرامش دریافت کند، ملزم به کاهش ترافیک ارسالی بدان مقصد خاص (تا X درصد) می باشد. از آنجایی که ممکن است بسته های دیگری که قبلاً به همان مقصد روانه شده اند در همان مسیر حرکت کرده باشند و آنها نیز منجر به تولید بسته های «دعوت به آرامش» دیگری شده باشند لذا ماشین مبدا پس از دریافت اولین بسته «دعوت به آرامش» به مدت زمان ثابت و مشخصی بسته هایی از این نوع را نادیده می گیرد. پس از طی این مدت، ماشین میزبان مجدداً به اندازه زمان مشخصی گوش می دهد که ببیند آیا «بسته دعوت به آرامش» دیگری دریافت می شود؟ اگر چنین بسته ای دریافت شد، خط کماکان در وضعیت ازدحام است فلذا باز هم جریان داده های خود را کاهش می دهد و مجدداً برای مدتی بسته های دعوت به آرامش را نادیده می گیرد. برعکس اگر در خلال مدت گوش دادن هیچ بسته دعوت به آرامش دریافت نشد، ماشین میزبان می تواند جریان داده های خود را افزایش بدهد. در این پروتکل «فیدبک ضمنی» کمک می کند تا بتوان قبل از آنکه جریان بسته ها بطور کل مسدود شود از ازدحام پیشگیری کرد مگر آنکه مشکلی جدی رخ بدهد.

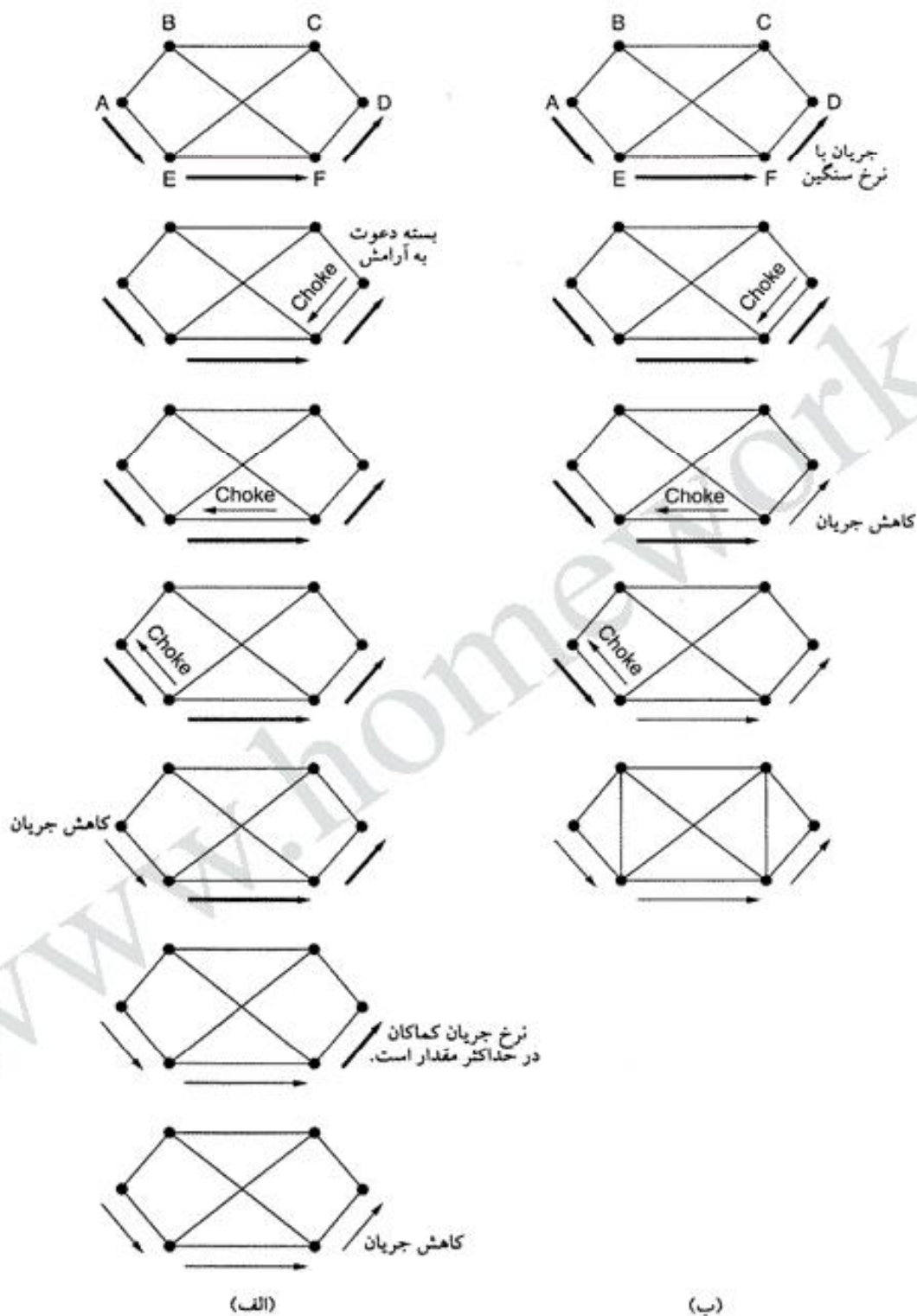
ماشینهای میزبان می توانند با تنظیم پارامترها و تغییر در سیاستهای کنترل ازدحام مثل اندازه پنجره، ترافیک ارسالی خود را کاهش بدهند. عموماً دریافت یک بسته «دعوت به آرامش» باعث پنجاه درصد کاهش در نرخ قبلی ارسال می شود. بسته بعدی، ترافیک را به ۲۵ درصد تقلیل می دهد و به همین ترتیب. [به عبارت دیگر دریافت بسته دعوت به آرامش مقدار فعلی نرخ ترافیک را عموماً نصف می کند. -م] افزایش نرخ ارسال معمولاً ضریب رشد کمتری دارد تا از بروز مجدد و پیاپی ازدحام پیشگیری شود. [یعنی مثلاً اگر دریافت بسته دعوت به آرامش نرخ ارسال را نصف می کند عدم دریافت مجدد آن در یک دوره زمانی مشخص، باعث دو برابر شدن نرخ ارسال نخواهد شد. -م]

چندین گونه از این الگوریتم برای کنترل ازدحام ارائه شده است. مثلاً در یکی از آنها، مسیر یاب چندین سطح آستانه (Threshold) در نظر می گیرد. بسته به آنکه وضعیت از کدامیک از سطوح تجاوز کرده باشد بسته های دعوت به آرامش می توانند در برگیرنده یکی از سه هشدار «وضعیت احتیاط»، «وضعیت حاد» و «بحران قطعی» باشند.

در گونه دیگری از این الگوریتم برای صدور علامت هشدار دهنده، بجای استفاده از معیار بهره وری خط از معیار طول صف یا میزان استفاده از فضای بافر بهره گرفته می شود. البته می توان از یک رابطه وزن دهی نمایی (Exponential Weighting) استفاده کرد و براساس تمامی این معیارها یک مقدار تلفیقی برای میزان بهره وری (u) تعریف نمود.

بسته های دعوت به آرامش گام به گام (Hop-by-Hop Choke Packet)

در سرعت های بالا یا در مسیرهای طولانی، ارسال بسته های «دعوت به آرامش» به ماشین مبدا، کارآیی خوبی نخواهد داشت چراکه واکنش این روش بسیار کند است. به عنوان مثال فرض کنید که ماشینی در سان فرانسیسکو (مثلاً مسیر یاب A در شکل ۵-۲۸) ترافیکی از بسته ها را با سرعت ۱۵۵ مگابیت در ثانیه برای ماشینی در نیویورک (مثلاً مسیر یاب D در شکل ۵-۲۸) ارسال کند. اگر بافر ماشین در نیویورک، رو به پر شدن بگذارد حدوداً سی میلی ثانیه طول می کشد تا بسته های دعوت به آرامش به سان فرانسیسکو برگردد و از مبدا بخواهد کندتر ارسال کند. در شکل ۵-۲۸ الف انتشار و بازگشت «بسته دعوت به آرامش» در مراحل دوم، سوم و چهارم نشان داده شده است. در خلال این سی میلی ثانیه تاخیر، ۴/۶ مگابیت اطلاعات دیگر فرستاده خواهد شد. حتی اگر پس از دریافت



شکل ۵-۲۸. (الف) یک بسته دعوت به آرامش که فقط مبداء را تحت تاثیر قرار می دهد. (ب) یک بسته دعوت به آرامش که در هر گام، بر روی مسیرهایی واقع بر مسیر تاثیر می گذارد.

این هشدار ماشین واقع در سانفرانسیسکو فوراً ارسال خود را بطور کل قطع نماید، ۴/۶ مگابیت اطلاعات بر روی خط به سوی ماشین روانه هستند و باید به نحوی تکلیف آنها مشخص شود. فقط در مرحله هفتم از شکل ۵-۲۸-الف مسیر یاب واقع در نیویورک متوجه کاهش ترافیک خواهد شد.

راهکار دیگر آن است که بسته دعوت به آرامش در هر گام از مسیری که می‌پیماید تأثیری بر عملکرد مسیر یابهای میانی بگذارد. (به شکل ۵-۲۸-ب دقت کنید). در اینجا به محض آنکه بسته دعوت به آرامش به مسیر یاب F می‌رسد، F ملزم به کاهش ترافیک ارسالی به D خواهد بود. انجام این کار منوط به آن است که F فضای بافر بیشتری را برای جریان بسته‌ها اختصاص بدهد چرا که ماشین مبدا با سرعت کامل در حال ارسال است و F در این میان می‌تواند کمکی موقت و سریع به D بنماید [با ایجاد توقف مصنوعی]؛ این کار را می‌توانید همانند آگهی‌های تجاری فرض کنید که شما را موقتاً از سردرد ناشی از دیدن یک برنامه طولانی نجات می‌دهند!! در گام بعدی بسته دعوت به آرامش به E می‌رسد و به او نیز تفهیم می‌شود که ترافیک خود به F را کاهش بدهد. این عمل نیز مستلزم وجود فضای بافر بیشتری در E است ولی F را موقتاً یاری می‌دهد. نهایتاً بسته دعوت به آرامش به A می‌رسد و جریان بطور واقعی و از مبدا آن کاهش می‌یابد.

تأثیر نهایی این روش گام به گام (Hop-by-Hop) آن است که در لحظه بروز ازدحام به سرعت چاره‌ای اندیشیده می‌شود ولی این راهکار به بهای افزایش فضای بافر ارسال، تمام خواهد شد. در این روش می‌توان بدون از دست رفتن هیچ بسته‌ای، مانع از افزایش ازدحام شد. این نظریه و نتایج شبیه‌سازی آن در مرجع (Mishra and Kanakia, 1992) تشریح شده است.

۵-۳-۵ دور ریختن بار (Load Shedding)

هر گاه هیچیک از روشهای فوق مشکل ازدحام را رفع نکنند، مسیر یابها می‌توانند آخرین تیر ترکش خود را بیازمایند: «دور ریختن بار»! عمل دور ریختن بار، آخرین و بدترین روشی است که به مسیر یاب اجازه می‌دهد هرگاه با سیل بسته‌هایی که نمی‌تواند از عهده هدایت آنها برآید، مواجه شود آنها را دور بریزد. (این اصطلاح از ادبیات شرکتهای تولید انرژی برق گرفته شده که در آنجا نیز مثلاً در یک روز گرم تابستان که مصرف برق از میزان تولید بیشتر می‌شود، برق بخشی از مناطق عمداً قطع می‌شود تا کل مناطق با مشکل قطع برق مواجه نشوند!)

یک مسیر یاب که غرق در بسته‌های اطلاعاتی شده می‌تواند تصادفاً برخی از آنها را انتخاب و حذف کند ولی از این بهتر هم می‌تواند عمل کند. اینکه کدام بسته باید حذف شود به نوع برنامه کاربردی که آنرا تولید کرده بستگی دارد. در انتقال فایل بسته‌های قدیمی ارزشمندتر از بسته‌های جدید هستند چرا که حذف بسته ۶ و نگه داشتن بسته‌های ۷ تا ۱۰ باعث ایجاد یک شکاف در میان داده‌های گیرنده شده و ممکن است فرستنده مجبور شود بسته‌های ۶ تا ۱۰ را نیز از نو ارسال کند (اگر گیرنده بطور طبیعی بسته‌های خارج از ترتیب را حذف کند). در یک فایل ۱۲ بسته‌ای، حذف بسته ۶ ممکن است منجر به تکرار ارسال بسته‌های ۶ تا ۱۰ شود در حالی که حذف بسته ۱۰ ممکن است فقط به تکرار بسته‌های ۱۰ تا ۱۲ نیاز داشته باشد. در طرف مقابل برای کاربردهای چند رسانه‌ای (مثل ارسال صدا یا تصویر)، بسته‌های جدید مهمتر از بسته‌های قدیمیند.

برای آن که سطح هوشمندی الگوریتم حذف بسته‌ها از این هم فراتر برود به همکاری فرستنده بسته‌ها نیاز است. در بسیاری از برنامه‌های کاربردی، برخی از بسته‌ها بسیار مهمتر از بقیه هستند. به عنوان مثال در برخی الگوریتمهای ارسال ویدیوی فشرده شده، در لحظات خاصی یک فریم تصویر کامل ارسال می‌شود و پس از آن فریمهای تصویر بعدی کامل نیستند بلکه فقط تفاوت این فریمها با آخرین فریم کامل محاسبه و پس از فشرده‌سازی ارسال می‌شوند. در چنین حالتی حذف بسته‌ای که بخشی از فریمهای فرعی محسوب می‌شود ارجح‌تر از حذف بسته‌ای است که به فریم اصلی و کامل تعلق دارد. به عنوان مثالی دیگر، انتقال یک سند

(Document) حاوی تصویر و متن ASCII را در نظر بگیرید. از دست دادن یک خط از نقاط تصویر در برخی از عکسها، کم ضررتر از دست دادن یک خط از متن است.

برای پیاده سازی یک سیاست هوشمند برای حذف بسته ها، برنامه های کاربردی باید رده اولویت مورد نظر را در بسته های خود مشخص کنند تا میزان اهمیت آنها مشخص گردد. اگر چنین کاری انجام شده باشد زمانی که مسیریاب مجبور به حذف بسته ها می شود می تواند از بسته های با اولویت پایین آغاز کند و بعداً به سراغ رده های بالاتر برود. البته فقط در موارد خاص می توان بسته هایی را در بالاترین رده اولویت علامتگذاری کرد.

البته چون هر ماشین در ارسال بسته ها با هر اولیوی، آزادی عمل دارد لذا باید به نحوی در کاربر انگیزه ایجاد کرد تا بسته هایش را با اولویت پایین بفرستد؛ پول می تواند این انگیزه را ایجاد کند؛ اگر برای حمل بسته های با اولویت بالا هزینه بیشتری گرفته شود کاربران سعی می کنند پی مورد از اولویتهای بالا استفاده نکنند. البته فرستنده ممکن است اجازه ارسال بسته های با اولویت بالا را در شرایط پار سبک به کاربر بدهد ولیکن با افزایش بار حذف خواهند شد. این موضوع کاربران را تشویق می کند از ارسال پی مورد بسته های با اولویت بالا صرف نظر کنند.

گزینه دیگر آنست که به ماشینهای میزبان اجازه بدهیم گاهی بیش از حد توافق شده در هنگام ایجاد مدار مجازی، ارسال داشته باشند ولیکن مشروط بدانکه ترافیک اضافی اولویت پایینی داشته باشد و به محض آشکار شدن علائم ازدحام حذف شود. چنین روشی ایده بدی نیست چراکه از منابع آزاد سیستم استفاده مفید می شود؛ به ماشینهای میزبان اجازه داده ایم مادامی که کس دیگر به منابع آزاد نیاز نداشته باشد از آن استفاده کنند ولیکن در شرایط دشوار [بار بالا] حق برای کسی ایجاد نکرده ایم.

تشخیص زود هنگام (Random Early Detection)

بر هر کسی روشن است که رفع ازدحام به محض کشف علائم آن بسیار کارآمدتر از آن است که بگذاریم اوضاع را بهم بریزد و سپس به رفع آن اقدام کنیم. چنین تجربه ای بدین ایده منتهی می شود که قبل از اشباع شدن کل فضای بافر، مسیریاب به حذف برخی از بسته ها اقدام کند. الگوریتم رایجی که در این خصوص کاربرد دارد اصطلاحاً الگوریتم RED (Random Early Detection) نامیده می شود. در برخی از پروتکل های لایه انتقال (مثل TCP) اگر بسته ها در حین انتقال از بین بروند مبداء، نرخ ارسال بسته هایش را کاهش می دهد. استدلالی که در پشت این منطق نهفته است آن است که TCP در اصل برای شبکه های سیمی طراحی شده و از آنجایی که شبکه های سیمی بسیار قابل اعتماد و کم خطا هستند لذا دلیل از دست رفتن بسته ها اغلب ناشی از پر شدن بافر است تا خطاهای انتقال. از این حقیقت می توان برای کمک به کاهش ازدحام بهره گرفت.

دلیل آنکه اجازه می دهیم مسیریاب قبل از آنکه وضعیت وخیم شود بسته ها را حذف کند آن است که بتوان قبل از دیر شدن کاری انجام داد. برای تعیین زمان شروع حذف بسته ها، هر مسیریاب میانگین طول صفهای خود را نگه می دارد. هر گاه طول متوسط صف بر روی برخی از خطوط، از حد مجاز تجاوز کرد آن خط با ازدحام مواجه شده و عملیات حذف شروع می شود.

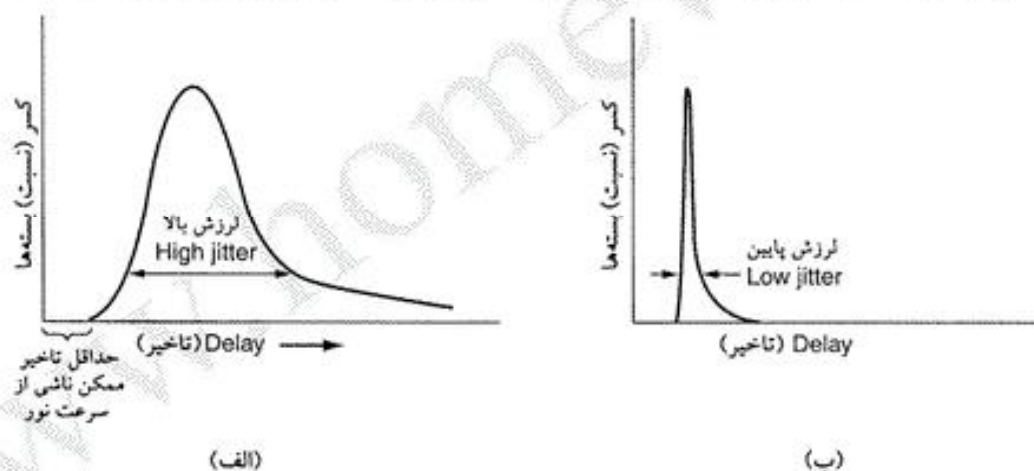
از آنجا که مسیریاب احتمالاً نمی تواند بفهمد کدام مبداء باعث مشکل بیشتری شده لذا این ایده که یکی از بسته ها را تصادفاً از صف دچار ازدحام بیرون بکشد، می تواند ایده خوبی باشد.

مسیریاب چگونه می تواند بروز مشکل را به مبداء بسته اعلام کند؟ یک روش آن است که یک «بسته دعوت به آرامش» برای مبداء بسته ارسال شود. اشکال این روش در آن است که بار بیشتری را بر روی شبکه ای که از قبل دچار ازدحام شده، تحمیل می کند. راهکار دیگر آن است که بسته انتخاب شده را حذف کرده و هیچ گزارشی هم اعلام نکند. در این صورت مبداء بسته از عدم برگشت بسته اعلام وصول (Ack) متوجه حذف بسته شده و اقدام

لازم را صورت می دهد. از آنجایی که مبداء بسته می داند که از بین رفتن بسته ها ناشی از ازدحام و حذف بسته هاست لذا بجای آنکه سخت تر تلاش کند، سرعت خود را کاهش می دهد. این «فیدبک ضمنی» صرفاً زمانی کار می کند که مبداء بسته وقتی متوجه از دست رفتن آن شود سرعت خود را کاهش بدهد. در شبکه های بی سیم که از بین رفتن بسته ها بیشتر ناشی از نویز لینک رادیویی است نمی توان از این روش استفاده کرد.

۶-۳-۵ کنترل لرزش (Jitter Control)

برای کاربردهایی مثل ارسال جریان داده های صدا و تصویر (ویدئو)، این که آیا ۲۰ میلی ثانیه طول می کشد تا بسته ای تحویل مقصد شود یا ۳۰ میلی ثانیه، موضوع مهمی نیست بلکه مهم آنست که این زمان انتقال ثابت باشد. میزان تغییر در زمان رسیدن بسته ها (یا به عبارتی انحراف معیار زمان تحویل بسته ها) اصطلاحاً «لرزش» (Jitter) نامیده می شود. وقتی «میزان لرزش» بالاست یعنی مثلاً وقتی که برخی از بسته ها در ۲۰ میلی ثانیه و برخی دیگر در ۳۰ میلی ثانیه به مقصد می رسند، کیفیت ارسال صدا و تصویر ویدیویی را کاهش می دهد. نمودار «لرزش» در شکل ۲۹-۵ به تصویر کشیده شده است. در این مثال، هر گاه ۹۹ درصد از بسته ها تأخیری بین ۲۴/۵ تا ۲۵/۵ میلی ثانیه داشته باشند، برای ارسال صدا و تصویر احتمالاً مناسب و قابل قبول خواهد بود. البته باید بازه تغییرات تأخیر، قابل تحقق و امکان پذیر باشد. در ضمن باید تأخیر ناشی از انتقال سیگنال با سرعت نور و همچنین حداقل تأخیر عبور از مسیر یابها را به حساب آورد ولیکن می توان برخی از تأخیرهای کوچک و غیر قابل پیش بینی را نادیده گرفت.



شکل ۲۹-۵. (الف) لرزش زیاد (ب) لرزش کم.

«لرزش» را می توان با محاسبه میانگین زمان انتقال در هر گام از کل مسیر، محدود کرد: وقتی بسته ای به یک مسیر یاب می رسد آن مسیر یاب بررسی می کند تا ببیند آن بسته به چه مقدار از برنامه زمانی خود جلوتر یا عقب تر است. این اطلاعات در درون هر بسته ذخیره شده و در هر گام بهنگام می شود. اگر بسته از برنامه زمانی خود جلو باشد آنقدر معطل می شود تا به برنامه زمانی خود برگردد ولیکن اگر از برنامه عقب افتاده باشد مسیر یاب سعی می کند آنرا به سرعت بر روی خروجی بفرستد.

در حقیقت الگوریتمی که تعیین می کند کدامیک از بسته های منتظر ارسال، برای خروج از یک خط اولویت دارد می تواند بسته ای را انتخاب کند که بیشتر از بقیه، از برنامه زمانی خود عقب افتاده است. بدین ترتیب بسته هایی که از برنامه زمانی خود جلوتر هستند آهسته تر و بسته هایی که از برنامه زمانی خود عقب افتاده اند سریعتر هدایت می شوند و در هر دو حالت میزان لرزش کاهش خواهد یافت.

در برخی از کاربردها مثل ارسال «ویدئو بر حسب تقاضا» (Video on-Demand)، خود گیرنده می تواند با بافر

کردن بسته ها و استخراج آنها از بافر جهت نمایش [در زمان مناسب] میزان لرزش را کاهش بدهد (بجای آنکه این کار به صورت بی درنگ به شبکه محول شود). ولیکن در برخی دیگر از کاربردها خصوصاً زمانی که نیاز به محاوره بی درنگ بین مردم باشد (مثل تلفن اینترنتی یا کنفرانسهای ویدیویی از راه دور)، تأخیر ناشی از بافر کردن داده ها، قابل قبول نخواهد بود.

روشهای کنترل ازدحام، حوزه ای فعال برای پژوهش و تحقیق است. پژوهشهای تازه و دستاوردهایی که در این زمینه حاصل شده در مرجع (Gevros et al., 2001) جمع بندی و ارائه شده است.

۴-۵ کیفیت خدمات (Quality of Service)

تکنیکهایی که در بخشهای قبلی مرور کردیم به منظور کاهش ازدحام و بهبود کارایی شبکه طراحی شده بودند. با این حال با رشد شبکه های چند رسانه ای، این ارزیابی های خاص و تمهیدات موردی کفایت نمی کند و به تلاشی جدی در تضمین کیفیت خدمات شبکه و طراحی پروتکل ویژه نیاز است. در بخشهای آتی نیز مطالعات خود را در خصوص کارایی شبکه، ادامه خواهیم داد یا این تفاوت که تمرکز ویژه ای بر روی روشهای «تضمین کیفیت خدمات» متناسب با نیاز برنامه های کاربرد، خواهیم داشت. ولی در همین ابتدا باید متذکر شویم که بسیاری از این ایده ها در حال تغییر هستند و عوض خواهند شد.

۴-۵-۱ نیازها

به دنباله ای از بسته ها که از مبدا به سوی مقصد روانه می شوند اصطلاحاً «جریان» (Flow) گفته می شود. در شبکه های اتصال گرا تمام بسته هایی که به یک «جریان» تعلق دارند، مسیر مشابهی را طی می کنند در حالی که در شبکه های بدون اتصال (Connectionless) بسته ها ممکن است از مسیرهای متفاوتی حرکت کنند. نیازهای مطلوب برای هر «جریان» را می توان با چهار پارامتر ابتدایی مشخص کرد: (۱) قابلیت اطمینان (۲) تأخیر (۳) لرزش (Jitter) (۴) پهنای باند. مجموعه این چهار پارامتر، «کیفیت خدمات» یا به اختصار QoS مورد نیاز یک «جریان» را تعیین می کند. در شکل ۵-۳ برخی از کاربردهای رایج و سطح نیازمندیهای آنها فهرست شده اند.

نوع کاربرد	قابلیت اطمینان	تأخیر	لرزش	پهنای باند
پست الکترونیکی	بالا	پایین	پایین	پایین
انتقال فایل	بالا	پایین	پایین	متوسط
دسترسی به وب	بالا	متوسط	پایین	متوسط
ورود به سیستم از راه دور	بالا	متوسط	متوسط	پایین
دریافت صوت بر حسب تقاضا	پایین	پایین	بالا	متوسط
دریافت ویدیو بر حسب تقاضا	پایین	پایین	بالا	بالا
تلفن اینترنتی	پایین	بالا	بالا	پایین
ویدیو کنفرانس	پایین	بالا	بالا	بالا

شکل ۵-۳. سطح نیازمندی برنامه های کاربردی به کیفیت خدمات.

چهار کاربرد اول نیاز شدیدی به «قابلیت اطمینان» (Reliability) دارند یعنی هیچ یک از بسته ها نباید اشتباه تحویل شوند. [به عبارت دیگر حتی یک بیت خطا در کل جریان داده ها قابل قبول نیست.] با اضافه کردن کدهای کشف خطا به هر بسته و بررسی آنها در مقصد می توان به این هدف نائل شد. اگر بسته ای در حین انتقال آسیب دید، وصول آن اعلام و تصدیق نمی شود و طبقاً باید از نو ارسال شود. این استراتژی قابلیت اطمینان بالایی برای بسته ها

فراهم می‌کند. چهار کاربرد آخر (صدا / ویدیو) می‌توانند اندکی خطا را تحمل کنند لذا کدهای کشف خطا برای هر بسته محاسبه و بررسی نخواهد شد.

کاربردهای انتقال فایل، شامل پست الکترونیکی یا دریافت تصاویر، حساسیت چندانی به تأخیر ندارند. حتی اگر تمام بسته‌ها به طور یکنواخت تا چند ثانیه هم تأخیر داشته باشند هیچ مشکل حادی پدید نمی‌آید. کاربردهای محاوره‌ای (Interactive) همانند جستجو در وب یا ورود به سیستم از راه دور (Remote Login) حساسیت بیشتری به تأخیر دارند.

کاربردهای بی‌درنگ مثل تلفن یا کنفرانس از راه دور حساسیت شدیدی به تأخیر دارند. اگر کلمات ادا شده در یک تماس تلفنی همگی با 2.000 ثانیه تأخیر برسند. کاربران چنین تماسی را نامناسب و غیرقابل قبول خواهند دانست. برعکس، اجرای فایل‌های صدا یا ویدیو که بر روی یک سرویس دهنده ذخیره شده به تأخیر پایین نیاز ندارند.^۱

سه کاربرد اول حساسیت چندانی به رسیدن بسته‌ها با فاصله زمانی نامشخص (نسبت به یکدیگر) ندارند؛ به عبارت بهتر لرزش بالا (High Jitter) مشکل چندانی برای این کاربردها ایجاد نمی‌کند. کاربرد چهارم یعنی ورود به سیستم از راه دور (Remote Login) تا حدودی به لرزش حساستر هستند چراکه در اثر لرزش زیاد ممکن است کاراکترها به صورت ناگهانی و نامتعارف بر روی صفحه نمایش ظاهر شوند. تصاویر ویدیویی و خصوصاً صدا شدیداً به «لرزش» حساس هستند. همانگونه که اشاره شد کاربری که در حال تماشای یک نمایش ویدیویی غیرزنده از روی شبکه است و فریمهای تصویر، همگی با 2.000 ثانیه تأخیر می‌رسند مشکلی پدید نخواهد آمد در حالی که اگر زمان انتقال بین ۱ تا ۲ ثانیه متغیر باشد، مشکلات جدی ایجاد خواهد شد. برای «صدا» لرزش چند میلی‌ثانیه‌ای نیز بوضوح شنیده خواهد شد.

در آخر، کاربردها نیاز متفاوتی به پهنای باند دارند: پست الکترونیکی و Remote Login به پهنای باند زیادی نیاز ندارند در حالی که ارسال ویدیو در تمام اشکال [فشرده شده یا غیرفشرده] پهنای باند بسیار زیادی را می‌طلبد. شبکه‌های ATM، «جریان» (Flow) را براساس QoS مورد نیاز در چهار رده وسیع دسته‌بندی کرده است:

۱. ارسال با نرخ ثابت (مثل تلفن از طریق شبکه)
۲. ارسال بی‌درنگ با نرخ متغیر (مثل کنفرانس ویدیویی فشرده شده)
۳. ارسال غیربی‌درنگ با نرخ متغیر (مثل تماشای نمایش غیرزنده از طریق اینترنت)
۴. ارسال با هر نرخ ممکن (برای انتقال فایل) (Available Bit Rate)

این رده‌ها در شبکه‌های دیگر و اهداف دیگر نیز مفید و راهگشا هستند. رده «ارسال با نرخ ثابت» تلاشی است در جهت شبیه‌سازی یک اتصال سیمی با پهنای باند ثابت و تأخیر یکنواخت.

«ارسال با نرخ متغیر» زمانی مفید خواهد بود که تصاویر ویدیویی، فشرده شده باشند و ضریب فشرده سازی فریمهای تصویر با هم فرق کند. بنابراین ارسال یک فریم تصویر با جزئیات و ظرافت زیاد نیاز به ارسال تعداد بیت زیادی دارد در حالیکه ارسال یک فریم تصویر که از یک دیوار سفید گرفته شده ممکن است تا حد بسیار بالایی فشرده شود. رده «ارسال با هر نرخ ممکن» برای کاربردهایی مثل پست الکترونیکی کاربرد دارد که به تأخیر یا لرزش حساس نیستند.

۲-۴-۵ راهکارهای دستیابی به کیفیت خوب خدمات

تا اینجا چیزهایی را در خصوص نیازهای QoS آموخته‌ایم؛ سؤال این است که چگونه به آنها برسیم؟ در همین جا

۱. چراکه مشاهده یک فیلم یا شنیدن یک موسیقی غیرزنده با چند ثانیه تأخیر دائم مشکل حادی ایجاد نمی‌کند. -م

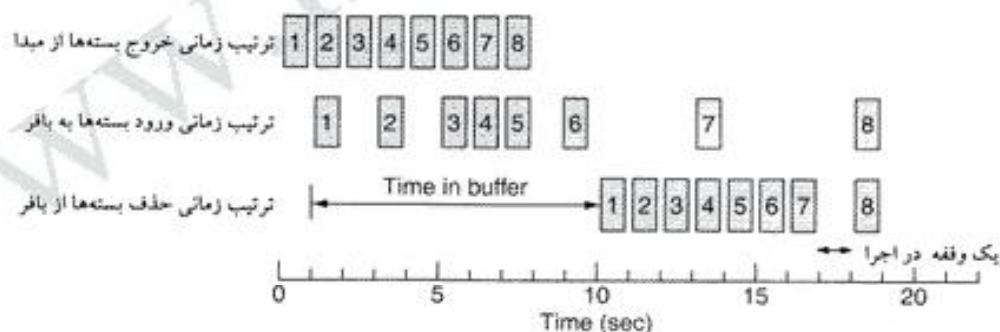
باید اذعان داشت که هیچ راهکار واحد و جادویی برای نیل به این اهداف وجود ندارد. به عبارت دیگر روشی واحد که تمام نیازهای QoS را به صورت بهینه برآورده نماید موجود نیست بلکه راهکارهای گوناگونی تعریف شده که در آنها برای ارائه راه حلهای عملی، ترکیبی از چندین روش مختلف بکار گرفته شده است. حال به بررسی برخی از این راهکارها که طراحان سیستم برای تأمین کیفیت خدمات از آنها بهره گرفته اند می پردازیم.

تمهیدات کافی

یک راه حل ساده آن است که برای مسیریاب ظرفیت پردازش، فضای بافر و پهنای باند زیادتری تدارک دیده شود تا بسته ها بتوانند بسادگی و بسرعت از آنها بگذرند. اشکال این راه حل آن است که گران تمام می شود! به مرور زمان، طراحان ایده های بهتری در خصوص پیش بینی میزان منابع مورد نیاز مطرح می کنند؛ در آن زمان شاید بتوان از این روش در عمل استفاده کرد. از این دیدگاه سیستم تلفن، سیستمی با تمهیدات کافی است و به ندرت گواشی تلفن را برمی دارید و فوراً بوق آزاد آنرا نمی شنوید زیرا ظرفیت این سیستم، بیش از حد معمولی تقاضا در نظر گرفته شده است.

بافر کردن

«جریان» داده ها می تواند در طرف گیرنده و قبل از تحویل به پروسه اصلی، بافر شود. بافر کردن جریان تأخیری در قابلیت اطمینان ندارد، تأخیر را افزایش می دهد ولیکن در عوض میزان لرزش را متعادل تر می کند. برای «صدا و تصویر برحسب تقاضا»^۱، لرزش یک مشکل اساسی است و این روش کمک زیادی به حل آن می کند. تفاوت بین لرزش زیاد و لرزش کم را در شکل ۵-۲۹ مشاهده کردیم. در شکل ۵-۳۱ دنباله ای از بسته ها را می بینید که با «لرزش» قابل توجهی تحویل می شوند. [فرض شده این بسته ها حاوی صدا یا تصویر هستند.] بسته اول در لحظه $t=0$ ارسال و در لحظه $t=1\text{sec}$ تحویل ماشین گیرنده شده است. بسته دوم با تأخیر بیشتری مواجه شده و ۲ ثانیه در راه بوده تا برسد. به محض آنکه این بسته ها از راه می رسند در ماشین گیرنده بافر می شوند.



شکل ۵-۳۱. یکنواخت کردن استریم خروجی یکمک بافرسازی بسته ها.

در $t=10\text{sec}$ اجرای آنها (Playback) شروع می شود. در این لحظه بسته های ۱ تا ۶ بافر شده اند فلذا می توان در فواصل زمانی مشخص و یکنواخت آنها را از بافر استخراج کرد و اجرای متعادل و مناسبی را انتظار داشت. متأسفانه لحظه ای که نوبت به اجرای بسته ۸ فرا می رسد به دلیل تأخیر نامتعارف، در بافر موجود نیست و اجرای آن باید تا زمان رسیدن آن متوقف شود که این مسئله یک توقف نامطلوب در اجرای موزیک یا فیلم ایجاد می کند. این مشکل را می توان با ایجاد تأخیر بیشتر در زمان شروع اجرا کاهش داد ولی در عوض به حجم بافر بزرگتری نیاز

است. سایت های وب تجاری که دارای صدا یا تصویر هستند از برنامه هایی برای اجرای صدا و تصویر استفاده می کنند که قبل از شروع اجرا، داده ها را تا ۱۰ ثانیه بافر می کنند.

شکل دهی ترافیک (Traffic Shaping)

در مثال بالا ماشین مبداء، بسته ها را در فواصل زمانی یکنواخت بیرون می فرستد، در حالی که مواردی وجود دارد که بسته ها بطور نامنظم تولید و ارسال می شوند و این مسئله ممکن است منجر به بروز ازدحام در شبکه شود. خروجی غیر یکنواخت عموماً وقتی است که سرویس دهنده مربوطه، بطور همزمان درگیر ارسال چندین «جریان ویدیویی یا صدا» است و در ضمن امکان عملیات دیگری مثل «جلو و عقب رفتن سریع» (Fast Forward/Rewind) در اجرای صدا و تصویر) و همچنین احراز هویت کاربران و نظایر آنرا نیز فراهم آورده است. از طرفی در برخی از کاربردها مثل کنفرانس از راه دور، بافر کردن داده ها بطور کل منطقی نیست.^۱ با این حال اگر بتوان کاری انجام داد تا سرویس دهنده (و کلاً ماشینهای میزبان) مجبور به ارسال با نرخ یکنواخت باشند، کیفیت خدمات (QoS) بهتر خواهد بود. در اینجا به بررسی روشی با عنوان «شکل دهی به ترافیک» می پردازیم که بجای آنکه بر ماشین گیرنده تمرکز داشته باشد، ترافیک تولیدی در سمت سرویس دهنده را متعادل و یکنواخت می کند.

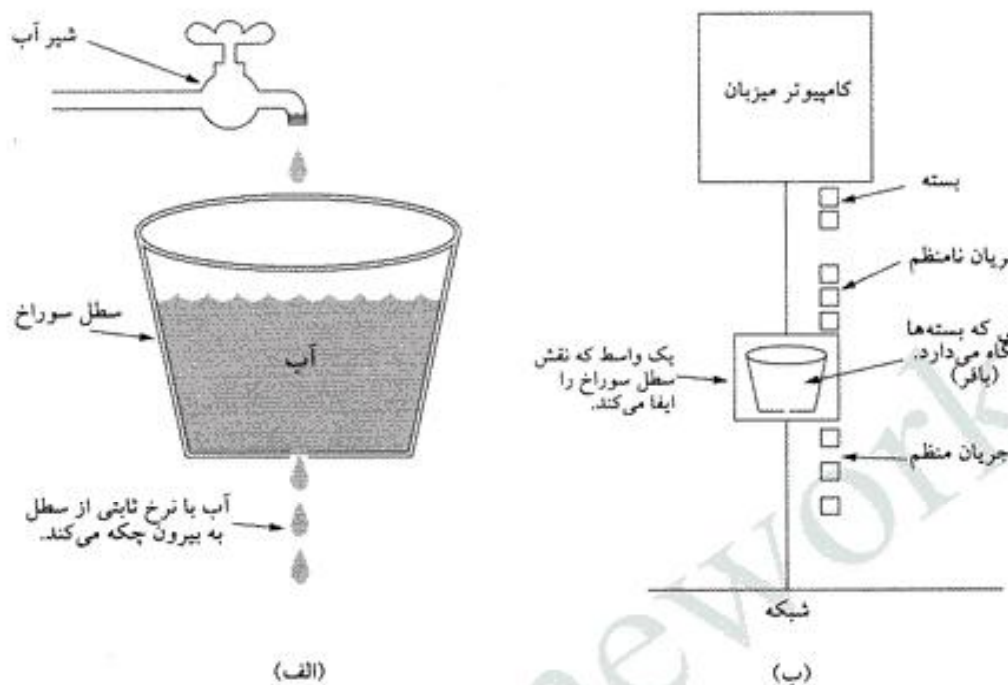
«شکل دهی به ترافیک» در خصوص منظم کردن نرخ متوسط (و کاهش حالت انفجارگونه) تولید داده های ارسالی است. «پروتکل پنجره لغزان» (Sliding Window Protocol) که قبلاً مطالعه کردیم فقط میزان داده هایی را که بطور همزمان ارسال می شوند، محدود می کند ولی متوسط نرخ ارسال آنها را محدود نمی کند. در روش شکل دهی به ترافیک، وقتی یک «اتصال» تنظیم می شود، کاربر و زیر شبکه (یا به عبارتی مشتری و حامل) بر روی الگوی خاصی از ترافیک (شکل ارسال ترافیک) برای آن مدار توافق می کنند. گاهی به این توافق، اصطلاحاً «توافق بر روی سطح خدمات» (Service Level Agreement) گفته می شود. مادامیکه مشتری بر طبق این توافق عمل کند و بسته ها را براساس قرارداد مورد توافق ارسال نماید، حامل (یعنی زیر شبکه) نیز متعهد است که بسته ها را بطور منظم تحویل بدهد. این عمل ازدحام را کاهش داده و اجازه می دهد زیر شبکه حامل بتواند برای عمل به تعهدات خود به فعالیت طبیعی ادامه بدهد. چنین توافقی برای عملیاتی مثل انتقال فایل چندان مهم نیست بلکه برای داده های بی درنگ مثل ارسال صدا و تصویر که شدیداً به کیفیت خدمات نیازمند است، اهمیت حیاتی دارد.

در نتیجه برای عملیات شکل دهی به جریان، مشتری به حامل می گوید: «الگوی ارسال من بدین نحو است؛ آیا از عهده آن بر می آیی؟» اگر زیر شبکه حامل این الگو را پذیرفت، مورد دیگری به میان خواهد آمد: حامل چگونه متوجه می شود که آیا مشتری از مفاد توافق پیروی می کند؟ و اگر پیروی نکرد چه عکس العملی باید نشان بدهد؟ نظارت بر جریان ترافیک اصطلاحاً «اعمال سیاست بر ترافیک» (Traffic Policing) نامیده می شود. توافق در خصوص شکل ترافیک و نظارتهای بعدی، در زیر شبکه های مبتنی بر مدار مجازی ساده تر از زیر شبکه های مبتنی بر دیتاگرام است. با این حال حتی در زیر شبکه های دیتاگرام، از همین ایده ها می توان برای اتصالات ایجاد شده در لایه انتقال استفاده کرد.

الگوریتم سطل سوراخ (Leaky Bucket Algorithm)

سطحی را در نظر بگیرید که همانند شکل ۵-۳۲ الف رخنه کوچکی در کف آن وجود دارد. بدون توجه به نرخ ورود آب به این سطل، جریان نشستی از آن ثابت است؛ مادامیکه آب در سطل وجود دارد نرخ خروجی معادل ρ

۱. بافر کردن مثلاً ۱۰ ثانیه از یک کنفرانس ویدیویی آنرا از حالت زنده خارج خواهد کرد و بهیچوجه مطلوب کاربران نیست. -م



شکل ۵-۳۲. (الف) یک سطل سوراخ دار آب (ب) یک سطل سوراخ دار «بسته».

وقتی سطل خالی است معادل صفر است. همچنین اگر آب اضافی به سطل پر وارد شود، از اطراف آن بیرون ریخته و هدر می رود. (یعنی در ظرفی دیگری که زیر این سوراخ قرار گرفته وارد نخواهد شد.)

به نحوی که در شکل ۵-۳۲-ب نشان داده شده از همین ایده می توان برای بسته ها نیز استفاده کرد. بطور مفهومی هر ماشین میزبان از طریق یک کارت واسط به شبکه متصل شده که این کارت واسط دارای سطلی سوراخ (یعنی یک صف داخلی بی نهایت) است. اگر بسته ای بخواهد در حالی که صف پر است وارد آن شود، حذف خواهد شد. به عبارت دیگر هرگاه یک یا چند پروسه در ماشین میزبان سعی در ارسال بسته ای نمایند در حالی که تعداد بسته های صف به حداکثر ممکن رسیده باشد، بسته جدید بدون هیچ تشریقاتی حذف خواهد شد. این عملیات می تواند توسط سخت افزار واسط انجام شود یا آنکه توسط سیستم عامل ماشین میزبان شبیه سازی شود. این الگوریتم برای اولین بار توسط Turner (۱۹۸۶) پیشنهاد و الگوریتم سطل سوراخ نام گرفت. در حقیقت این روش چیزی نیست مگر یک سیستم صف بندی با یک سرویس دهنده واحد که سرویس دهی به عناصر منتظر در صف با نرخ ثابتی انجام می شود.

ماشین میزبان اجازه دارد در هر تیک ساعت یک بسته بر روی شبکه قرار بدهد. این کار نیز می تواند توسط کارت واسط شبکه یا سیستم عامل ماشین انجام و مدیریت شود. این مکانیزم جریانی نامنظم و بی قاعده از بسته های تولید شده توسط پروسه های کاربری ماشین را به جریانی منظم بر روی شبکه تبدیل کرده و بدین ترتیب حالت انفجارگونه ترافیک، معتدلتر شده و احتمال ازدحام کاهش چشمگیری می یابد.

اگر بسته ها دارای اندازه ای ثابت و یکسان باشند (مثل سلولهای ATM) می توان طبق توضیح بالا از این الگوریتم بهره گرفت: «یک سلول در هر تیک ساعت». ولیکن وقتی اندازه بسته ها متغیر باشد، بهتر آن است که در هر تیک ساعت به جای یک بسته، تعداد بایت ثابتی ارسال شود. بدین ترتیب اگر قرار باشد در هر تیک ساعت ۱۰۲۴ بایت ارسال شود، می توان یک بسته ۱۰۲۴ بایتی یا دو بسته ۵۱۲ بایتی یا چهار بسته ۲۵۶ بایتی و به همین

ترتیب، ارسال کرد. اگر تعداد بایتهای باقیمانده ناچیز باشد، بسته بعدی باید تا تیک ساعت بعدی منتظر بماند. پیاده‌سازی الگوریتم سطل سوراخ (نسخه اصلی آن) ساده است. سطل سوراخ متشکل از یک صف متناهی (محدود) است. هر گاه بسته‌ای دریافت شود و فضای کافی در صف وجود داشته باشد به آخر صف ملحق می‌شود و در غیر این صورت حذف می‌شود. در هر تیک ساعت یک بسته ارسال می‌شود (مگر آنکه صف خالی باشد). «الگوریتم سطل سوراخ مبتنی بر شمارش بایت» نیز تقریباً به همین روش پیاده‌سازی می‌شود. در هر تیک ساعت شمارنده بایت، به مقدار n تنظیم می‌شود. اگر بسته سر صف تعداد بایت کمتری از مقدار فعلی شمارنده داشته باشد، ارسال شده و به تعداد بایتهای بسته از شمارنده کم می‌شود. مادامیکه مقدار شمارنده از بسته‌های موجود در سر صف بیشتر است می‌توان این بسته‌ها را ارسال کرد. وقتی مقدار شمارنده از اندازه بسته بعدی در سر صف کمتر شود، عمل ارسال تا تیک بعدی متوقف می‌شود تا مقدار شمارنده بایت مجدداً به مقدار n تنظیم شده و جریان بسته‌ها ادامه یابد.

به عنوان مثالی از الگوریتم سطل سوراخ، کامپیوتری را در نظر بگیرید که می‌تواند داده‌هایی با سرعت ۲۵ میلیون بایت در ثانیه (معادل 200 Mbps) تولید کند و شبکه نیز با همین سرعت کار می‌کند، ولیکن مسیر یاب فقط می‌تواند چنین نرخ داده‌ای را در یک فاصله زمانی محدود بپذیرد (تا زمانی که بافرهایش پر شود). در زمانهای طولانی بهتر آن است که نرخ ارسال از ۲ میلیون بایت در ثانیه تجاوز نکند.^۱ حال فرض کنید هر ۴۰ میلی‌ثانیه، داده‌ها در یک حالت انفجارگونه و بصورت توده‌های یک میلیون بایتی ارسال شوند. [یعنی هر ۴۰ میلی‌ثانیه یک توده یک میلیون بایتی با حداکثر سرعت یعنی ۲۵ میلیون بایت در ثانیه تولید و ارسال می‌شوند.] برای آنکه نرخ متوسط ارسال به 2MByte/Sec کاهش یابد، باید از «سطلی سوراخ» با $\rho = 2\text{MByte/sec}$ و فضای بافر 1MByte بهره گرفته شود. این بدین معناست که توده‌های داده‌ای که انفجارگونه ارسال می‌شوند با طول حداکثر 1MByte قابل دریافت و مدیریت هستند و چیزی از دست نخواهد رفت. این سطل نیز در طول ۵۰۰ میلی‌ثانیه تخلیه می‌شود و اینکه داده‌ها با چه سرعتی وارد می‌شوند اهمیتی ندارد.

در شکل ۵-۳۳ الف ورود داده‌ها به سطل سوراخ در خلال ۴۰ میلی‌ثانیه و با نرخ 25 MByte/Sec صورت گرفته است [معادل ۱ مگابایت داده و متناسب با ظرفیت سطل]. در شکل ۵-۳۳ ب می‌بینیم که این مقدار از داده در خلال ۵۰۰ میلی‌ثانیه و با نرخ 2MByte/sec تخلیه و ارسال شده است.

الگوریتم سطل نشانه‌دار (The Token Bucket Algorithm)

الگوریتم سطل سوراخ الگوی خروجی سختگیرانه و با نرخ میانگین و ثابتی را تحمیل می‌کند و آنکه ترافیک تا چه اندازه انفجاری است برایش مهم نیست.^۲ در بسیاری از کاربردها وقتی توده‌ای انفجاری از داده‌ها می‌رسد بهتر آن است که اجازه بدهیم سرعت تا حدی افزایش یابد؛ بنابراین به الگوریتمی احتیاج داریم که قابلیت انعطاف بیشتری داشته باشد و ترجیحاً هیچ داده‌ای از دست نرود. یکی از این الگوریتمها «الگوریتم سطل نشانه‌دار» (Token Bucket Algorithm) است. در این الگوریتم، سطل سوراخ (یا به عبارتی بافر موجود در کارت شبکه یا

۱. یعنی اگرچه ظرفیت ارسال شبکه 25 Mbyte/Sec است ولی مسیر یاب بدلیل محدودیت سرعت، فقط از عهده پردازش 2Mbyte/Sec برمی‌آید و در غیر اینصورت با ازدحام مواجه می‌شود. اگر فرستنده در لحظاتی با نرخ بیشتری ارسال کرد در عوض باید برای لحظاتی متوقف شود. -م

۲. یادآوری می‌کنیم که انفجاری بودن ترافیک بدین معناست که فقط در لحظات کوتاهی حجم انبوهی اطلاعات تولید و ارسال می‌شود و در بخش زیادی از زمان ارسال داده‌ها ناچیز است. فاکتور انفجاری بودن ترافیک را «حاصل تقسیم ماکزیمم ترافیک بر میانگین ترافیک» در نظر بگیرید. -م

سیستم عامل)، نشانه ای (اصطلاحاً یک توکن^۱) را نگه می دارد که این توکن در هر ΔT ثانیه یکبار تولید می شود. در شکل ۵-۳۴ الف، سطلی را مشاهده می کنید که در آن سه نشانه (توکن) وجود دارد و همچنین پنج بسته منتظر ارسال هستند. هر گاه بسته ای بخواهد ارسال شود باید ابتدا یک نشانه بدست آورده و آن را از بین ببرد. در شکل ۵-۳۴ ب می بینیم که سه بسته از مجموع پنج بسته جواز خروج گرفته و ارسال شده اند در حالی که دو بسته باقیمانده در انتظار تولید دو نشانه دیگر به سر می برند.

نوع و مکانیزم شکل دهی به ترافیک در «الگوریتم سطل نشانه دار» نسبت به «الگوریتم سطل سوراخ» متفاوت است. الگوریتم سطل سوراخ به ماشینهای بیکار اجازه نمی دهد که وقتی بیکار هستند سهمیه ارسال خود را نگه داشته و از این سهمیه به یکباره جهت ارسال انفجارگونه بسته ها استفاده کنند. در طرف مقابل، الگوریتم سطل نشانه دار اجازه می دهد که ماشین سهمیه خود را تا حداکثر حجم سطل (n بسته) پس انداز کنند. این ویژگی بدین معناست که توده ای انفجاری از بسته ها را (تا حداکثر n بسته) می توان یکجا فرستاد. [البته به شرط پس انداز تعداد n نشانه]. این ویژگی اجازه می دهد که خروجی بتواند تا حدودی حالت انفجارگونه داشته باشد و هنگامی که ورود بسته ها انفجاری است، پاسخ سریعتری داده شود.

تفاوت دیگر این دو الگوریتم آن است که الگوریتم سطل نشانه دار در هنگامی که سطل پر شود نشانه ها را دور می اندازد (یا به عبارتی ظرفیت ارسال را) در حالی که هرگز بسته ها را حذف نمی کند. در مقابل، الگوریتم سطل سوراخ به محض پر شدن سطل، بسته ها را حذف می نماید.

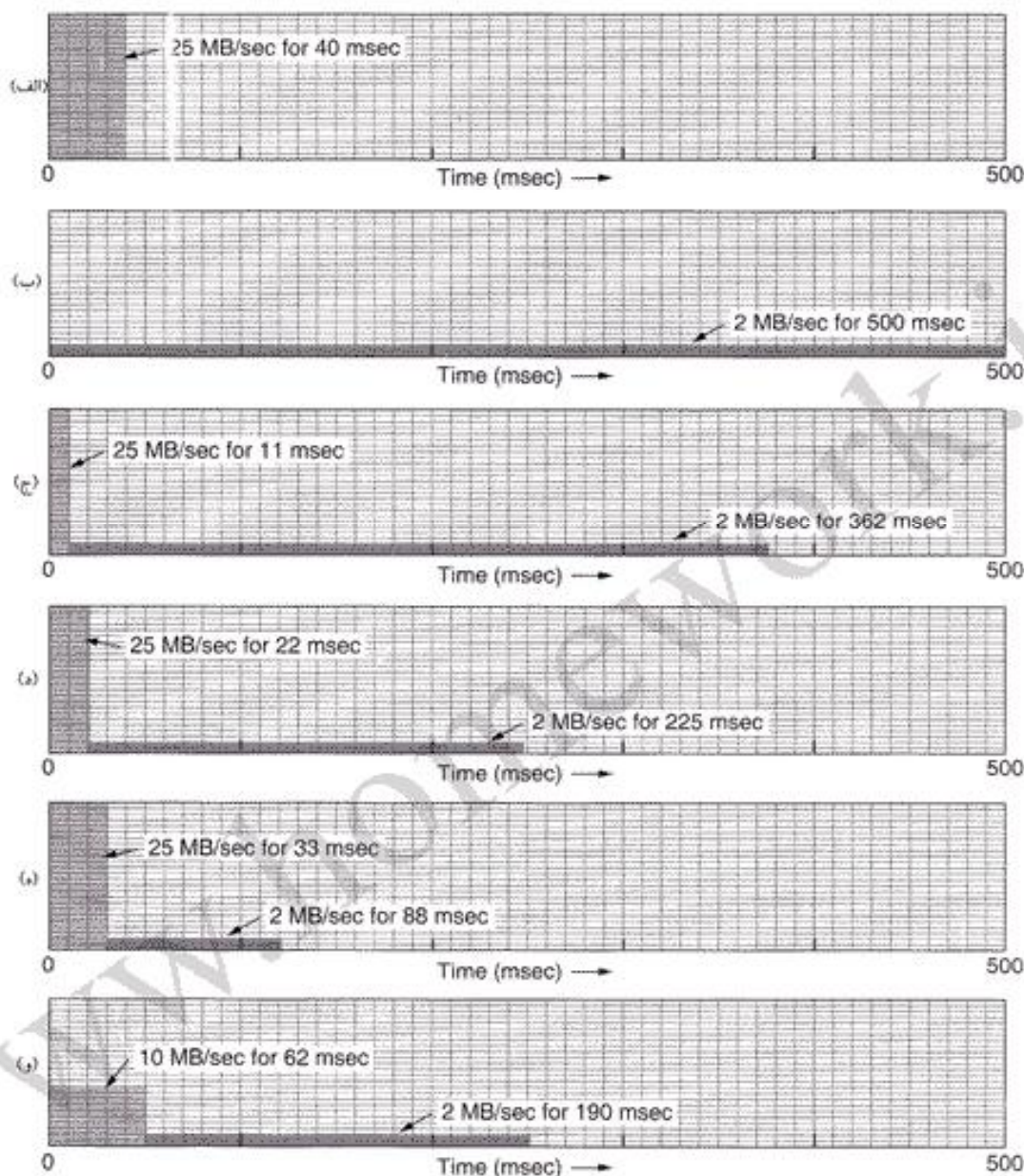
در اینجا یک تغییر کوچک ممکن است و آن اینکه هر نشانه (توکن) مجوز ارسال k بایت تلفی شود نه ارسال یک بسته. در این حالت یک بسته را فقط زمانی می توان ارسال کرد که به تعداد کافی نشانه (معادل با اندازه بسته) پس انداز شده باشد. نشانه هایی که سهمیه ارسال آنها (برحسب بایت) کوچکتر از اندازه بسته فعلی است برای استفاده بعدی نگه داشته می شود.

الگوریتم سطل سوراخ و الگوریتم سطل نشانه دار را می توان به همان نحوی که برای منظم کردن ترافیک خروجی ماشینهای میزبان بکار می رود برای متعادل کردن ترافیک بین مسیرهای نیز استفاده کرد. با این حال یک تفاوت بدیهی بین این دو کاربرد وجود دارد و آن هم اینکه الگوریتم سطل نشانه دار می تواند یک ماشین میزبان را وادار به توقف ارسال نماید، در حالی که وادار کردن یک مسیر یا به توقف (در حالی که بافرهای ورودی آن پر است)، می تواند به از دست رفتن داده ها بینجامد.

پیاده سازی الگوریتم سطل نشانه دار، به سادگی تعریف یک متغیر است تا این متغیر تعداد نشانه ها (توکنها) را بشمارد. این شمارنده در هر ΔT ثانیه یک واحد افزایش داده شده و با ارسال یک بسته، کاهش می یابد. وقتی این شمارنده به صفر برسد هیچ بسته ای را نمی توان ارسال کرد. در گونه دیگر این الگوریتم یعنی روش شمارش بایت، شمارنده هر ΔT ثانیه k واحد (k بایت) افزایش یافته و با ارسال بسته به اندازه طول آن از شمارنده کسر می شود.

اصولاً کاری که الگوریتم سطل نشانه دار انجام می دهد آن است که اجازه ارسال انفجارگونه بسته ها را صادر می کند ولیکن با سقف حداکثر و تنظیم شده ای که قابل کنترل و اداره باشد. برای مثال به شکل ۵-۳۳ ج دقت کنید. در اینجا یک سطل نشانه دار با ظرفیت ۲۵۰ کیلو بایت در اختیار داریم. نشانه ها (توکنها) متناسب با نرخ ۲ مگابایت بر ثانیه تولید می شوند. [یعنی نشانه ها با نرخی تولید می شوند که فقط اجازه ارسال ۲ مگابایت اطلاعات در ثانیه را صادر می کنند. -م] فرض کنید هنگامی که یک توده انفجارگونه ۱ مگابایتی تولید می شود، سطل پر است. سطل، قادر است بمدت حدود یازده میلی ثانیه و با سرعت 25MByte/sec، داده های خود را خالی کند. از آن به بعد

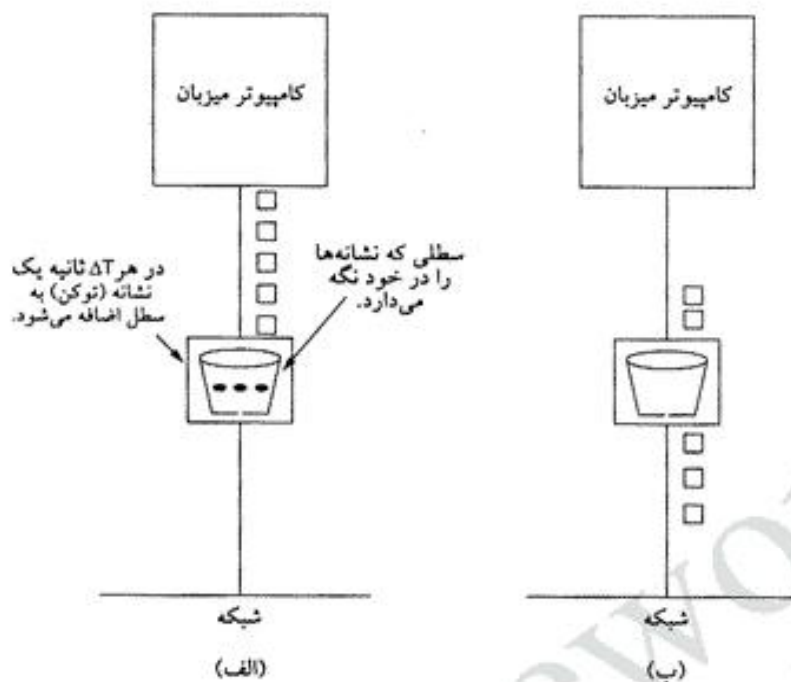
۱. نشانه یا «توکن» را در اینجا جواز یا سهمیه ارسال تعداد مشخصی بایت در نظر بگیرید. -م



شکل ۵-۳۳. (الف) ورودی به یک سطل سوراخ دار. (ب) خروجی از سطل سوراخ دار. خروجی از یک سطل نشانه دار با ظرفیت: (ج) 250 KByte (د) 500 KByte (ه) 750 KByte (و) خروجی یک سطل نشانه دار با ظرفیت 500KByte به یک سطل سوراخ دار با نرخ خروجی 10-Mbps اعمال شده است.

مجبور است تا وقتی که توده انفجاری داده ارسال نشده سرعت خروجی را بر روی 2MB/sec ثابت نگه دارد.^۱ محاسبه طول زمانی که می توان بصورت انفجاری و با نرخ حداکثر، توده تولید شده را ارسال کرد، اندکی پیچیده است. یعنی سقف حداکثر زمان ارسال انفجارگونه، معادل تقسیم «سهامیه موجود در سطل» بر «نرخ ارسال»

۱. یادآوری می کنیم که در الگوریتم سطل نشانه دار اگر بسته ها بیش از ظرفیت (سهامیه) موجود در سطل تولید شوند، حذف نخواهند شد.



شکل ۳۴-۵. الگوریتم سطحی نشانه دار. (الف) قبل از ورود بسته (ب) بعد از خروج بسته.

نیست چراکه در خلال ارسال این توده از بسته ها، نشانه های جدیدی تولید می شوند. اگر طول زمان ارسال انفجاری بسته ها را S بنامیم، ظرفیت موجود در سطح نشانه دار C بایت، نرخ تولید نشانه ρ بایت بر ثانیه و حداکثر نرخ خروجی M bytes/sec باشد، حداکثر نرخ ارسال انفجارگونه خروجی $C + \rho.S$ بایت است.^۱ همچنین می دانیم که اگر طول زمان ارسال، S ثانیه و حداکثر نرخ ارسال M بایت بر ثانیه باشد، تعداد بایتی که در این زمان ارسال می شود $M.S$ بایت خواهد بود. بنابراین داریم:

$$C + \rho.S = M.S$$

با حل این معادله بدست می آوریم:

$$S = \frac{C}{M - \rho}$$

در مثال فوق با پارامترهای $C=250$ KByte، $\rho=2$ MByte/sec و $M=25$ MByte/sec حداکثر زمان ارسال انفجارگونه بسته ها یازده میلی ثانیه بدست خواهد آمد. شکل های ۵-۳۳-د و ۵-۳۳-ه سطح نشانه دار را برای ظرفیت ۵۰۰ کیلوبایت و ۷۵۰ کیلوبایت نشان می دهد.

مشکل بالقوه الگوریتم سطح نشانه دار آن است که کماکان اجازه می دهد توده ای از بسته ها به صورت انفجارگونه تولید و ارسال شود (حتی اگر طول زمان تولید به دقت و براساس انتخاب صحیح ρ و M تنظیم شده باشد). اغلب مطلوب آن است که نرخ حداکثر تولید بسته ها کاهش یابد ولی بدون آنکه بخواهیم به مقدار کم و ثابت

۱. یادآوری می کنیم که اگر سهمیه موجود در سطح C بایت باشد، می توان C بایت از داده ها را با نرخ حداکثر و بصورت انفجاری ارسال کرد و پس از آن باید نرخ ارسال را متناسب با نرخ تولید نشانه کاهش داد. در خلال ارسال انفجاری این C بایت که بعد از S ثانیه طول می کشد تعدادی نشانه جدید (سهمیه جدید) تولید می شود و اجازه می دهد حجم ارسال انفجاری اندکی افزایش یابد. این حجم معادل $C + \rho.S$ بایت است: C بایت سهمیه قبلی و $\rho.S$ بایت سهمیه جدید تولید شده در زمان ارسال. -م.

الگوریتم سطل سوراخ بسنده نمایم.

یک روش برای متعادل کردن ترافیک آن است که بعد از «سطل نشانه دار» یک «سطل سوراخ» قرار داده شود. نرخ خروجی سطل سوراخ باید بیشتر از نرخ خروجی سطل نشانه دار (یعنی مقدار ρ) انتخاب شود ولی این مقدار باید از حداکثر نرخ مجاز شبکه کمتر باشد. شکل ۵-۳۳ و خروجی یک سطل نشانه دار با ظرفیت ۵۰۰ کیلوبایت که پس از آن یک سطل سوراخ 10MByte/sec قرار داده شده را نشان می دهد.

نظارت بر این روشها ممکن است اندکی پیچیده باشد. اصولاً شبکه باید این الگوریتم را شبیه سازی کند و مطمئن گردد که بسته یا بایتهای بیشتر از حد مجاز و مشخص ارسال نمی شود. در هر حال این ابزارها روشهایی برای شکل دهی به ترافیک شبکه به گونه ای قابلیت مدیریت است تا بتوان نیازهای QoS (کیفیت خدمات) را برآورده نمود.

رزرو منابع (Resource Reservation)

توانایی شکل دهی و تنظیم ترافیک ارسالی، تمهید خوبی برای تضمین «کیفیت خدمات» (QoS) محسوب می شود ولیکن استفاده از این روشها زمانی کارآمد خواهد بود که تمام بسته ها از مسیر یکسانی عبور کنند. پراکندگی تصادفی بسته ها بر روی مسیرهای متفاوت، تضمین هر چیزی را بسیار دشوار می کند. بنابراین برای تامین کیفیت خدمات باید بین مبداء و مقصد چیزی شبیه به یک مدار مجازی ایجاد و تنظیم شود و تمام بسته های یک «جریان» از این مسیر حرکت کنند.

هر گاه برای جریان داده ها، مسیر ویژه داشته باشیم می توان منابع لازم را در طول این مسیر، رزرو کرده و موجود بودن ظرفیت مورد نیاز را تضمین کرد. سه نوع متفاوت از منابع را می توان از قبل رزرو کرد:

۱. پهنای باند

۲. فضای بافر

۳. سیکل های CPU [ظرفیت پردازش مورد نیاز]

رزرو پهنای باند آشکارترین مورد است: اگر یک جریان به 1Mbps پهنای باند نیاز داشته باشد و ظرفیت خط خروجی 2Mbps باشد تلاش برای هدایت سه جریان همزمان بر روی این خط عملی نیست. لذا رزرو پهنای باند به معنای آن نیست که می توان یک خط خروجی را بیش از ظرفیت آن رزرو کرد.

دومین منبع که اغلب با محدودیت مواجه است، فضای بافر می باشد. وقتی یک بسته دریافت می شود معمولاً توسط سخت افزار در حافظه کارت شبکه ذخیره می گردد. بعداً نرم افزار مسیریاب باید آن را به یک بافر در فضای RAM اصلی منتقل کرده و پس از پردازش و تعیین مسیر، آنرا در صف مربوط به خط خروجی منتخب، وارد نماید. اگر هیچ بافری موجود نباشد، بسته دریافتی حذف خواهد شد چرا که فضایی برای ذخیره آن موجود نیست. برای تضمین کیفیت خوب خدمات، باید برای هر «جریان» مشخص مقداری بافر کنار گذاشت تا لازم نباشد بسته های آن جریان برای بدست آوردن فضای بافر با دیگر بسته ها رقابت کنند. در این صورت به محض نیاز به بافر، فضای لازم در اختیار خواهد بود (البته تا حد معقول و مشخص).

در آخر، «سیکل های CPU» نیز منبع کمیابی است: پردازش یک بسته به اندازه معینی وقت مسیریاب را می گیرد لذا هر مسیریاب قادر است در یک ثانیه، تعداد مشخص و محدودی از بسته ها را پردازش کند. برای آنکه بتوان مطمئن شد که بسته ها در روالی منظم و به موقع پردازش می شوند باید مراقب بود که بار CPU بیش از حد زیاد نشود.

در نگاه اول شاید به نظر برسد که اگر پردازش هر بسته مثلاً یک میکروثانیه طول بکشد، پس مسیریاب

می تواند یک میلیون بسته را در هر ثانیه پردازش و هدایت نماید. چنین نتیجه گیری غلط است زیرا به دلیل تغییرات آماری بار [یعنی ثابت نبودن میزان بار]، پردازنده در دوره هایی از زمان بیکار می ماند. اگر CPU موظف باشد که در هر سیکل، کار پردازش را شروع و تکمیل کند، حتی از دست رفتن چند سیکل (ناشی از بیکاریهای مقطعی) انباشته شده و قابل جبران نخواهد بود.

با این حال حتی وقتی میزان بار کمتر از ظرفیت تئوریک مسیریاب باشد باز هم ممکن است صف تشکیل و تأخیر ایجاد شود. وضعیتی را در نظر بگیرید که در آن بسته ها بطور کاملاً تصادفی و با نرخ متوسط λ بسته بر ثانیه وارد می شوند. زمان مورد نیاز برای پردازش هر بسته نیز تصادفی است و بطور متوسط μ بسته در هر ثانیه پردازش می شود. با این فرض که تابع توزیع دریافت بسته ها و همچنین پردازش آنها «پواسون» باشد می توان به کمک «نظریه صف» (Queuing Theory) ثابت کرد که متوسط تأخیری که هر بسته با آن مواجه خواهد شد (یعنی T) معادل است با:

$$T = \frac{1}{\mu} \times \frac{1}{1-\lambda/\mu} = \frac{1}{\mu} \times \frac{1}{1-\rho}$$

که در آن ρ معادل است با λ/μ و از آن به عنوان «میزان بهره وری CPU» (CPU Utilization) یاد می شود. فاکتور $1/\mu$ ، زمان لازم برای پردازش یک بسته است. (هنگامی که رقابت و صف وجود ندارد). فاکتور $(1-\rho)$ ، $1/\mu$ میزان کاهش سرعت پردازش، در اثر تشکیل صف (رقابت) تلقی می شود. به عنوان مثال اگر $950000 \text{ Packet/sec} = \lambda$ و $\mu = 1,000,000 \text{ Packet/sec}$ باشد میزان بهره وری CPU یعنی ρ معادل با 95% است و متوسط تأخیر بسته ها بجای 1 میکروثانیه، 20 میکروثانیه خواهد بود. این زمان در برگزیده زمان انتظار در صف و زمان سرویس دهی [زمان پردازش] بسته است. اگر مثلاً سی مسیریاب در مسیر جریان بسته ها قرار گرفته باشد زمان تأخیر انتظار صف در کل مسیر بنهایی معادل با 600 میکروثانیه است.

کنترل پذیرش (Admission Control)

حال در مرحله ای هستیم که ترافیک ورودی از یک «جریان» (Flow) خاص به خوبی شکل و نظم داده شده و بسته ها از یک مسیر واحد حرکت می کنند و پیشاپیش ظرفیت مورد نیاز در طول مسیر، پیش بینی و رزرو شده است. با چنین فرضی، هر گاه جریانی از بسته ها به یک مسیریاب تسلیم شود براساس ظرفیت موجود خود و سطح تعهداتی که در خصوص دیگر جریانها پذیرفته، باید در خصوص قبول یا رد آن تصمیم بگیرد.

تصمیم گیری در خصوص پذیرش یا رد یک «جریان»، یک مقایسه ساده بین میزان پهنای باند، بافر و سیکل های CPU مورد نیاز و ظرفیت باقیمانده در مسیریاب نیست. این تصمیم گیری اندکی پیچیده تر از این مقایسه ساده است. اگرچه برخی از برنامه های کاربردی ممکن است میزان نیاز خود به پهنای باند را بدانند ولیکن آگاهی کمی از بافر یا حجم پردازش مورد نیاز بسته ها در مسیریابهای واقع بر روی مسیر دارند لذا حداقل می توان گفت که به روشی متفاوت برای توصیف و ارزیابی جریان نیاز است. در ثانی، برخی از کاربردها در مقابل برآورده نشدن انتظاراتشان، تحمل بیشتری از خود نشان می دهند. در آخر آنکه ممکن است برخی از کاربردها بخواهند که در خصوص «پارامترهای جریان» چانه زنی و مذاکره کنند در حالی که امکان دارد برخی دیگر از کاربردها چنین خصوصیتی نداشته باشند. به عنوان مثال یک برنامه نمایش دهنده فیلم که معمولاً 30 فریم تصویر در ثانیه را نمایش می دهد ممکن است بخواهد در صورت عدم وجود پهنای باند کافی برای این حجم از داده، تعداد فریمهای تصویر را تا 25 Frame/sec کاهش بدهد. به همین ترتیب ممکن است تعداد نقاط تصویر در هر فریم (Pixel Per Frame)، پهنای باند صدا و دیگر ویژگیها نیز قابل تغییر باشد.

چونکه برای رسیدن به توافق نهایی در خصوص تامین نیازهای یک «جریان»، باید مولفه های متعددی در

مذاکرات شرکت داشته باشند (اعم از فرستنده، گیرنده و تمام مسیرهای واقع بر روی مسیر)، لذا هر «جریان» باید برحسب پارامترهای مشخصی بدقت توصیف شود تا بتوان بر روی این پارامترها مذاکره و توافق کرد. مجموعه چنین پارامترهایی اصطلاحاً «مشخصات توصیفی جریان» (Flow Specification) نامیده می شود. بدین ترتیب یک فرستنده (مثل سرویس دهنده ویدیو) مشخصات توصیفی جریان را به صورت پارامترهای پیشنهادی و مورد نظر خود تعریف می نماید. این پارامترهای پیشنهادی در طول مسیر منتشر می شود و هر مسیریاب واقع بر مسیر آنها را بررسی کرده و در صورت نیاز در آنها تغییراتی ایجاد می کند. این تغییرات فقط کاهشی است نه افزایشی (یعنی مثلاً نرخ مورد نظر ارسال داده ها را کاهش می دهد نه افزایش). وقتی این پارامترها به طرف مقابل برسد، به اجرا گذاشته می شوند.

در مثال شکل ۵-۳۵، نمونه ای از پارامترهای توصیف جریان، مبتنی بر RFC 2210 و RFC 2211 نشان داده شده است. در اینجا پنج پارامتر وجود دارد که اولین آنها پارامتر Token Bucket Rate (نرخ تولید نشانه در الگوریتم سطل نشانه دار) است و تعداد بایتهایی را مشخص می کند که در هر ثانیه به سطل وارد می شوند. در حقیقت این پارامتر حداکثر نرخ ارسال مجاز و قابل قبولی است که فرستنده می تواند ارسال نماید و میانگین ارسال در یک زمان طولانی تلقی می شود.

واحد	پارامترهای توصیف جریان
Bytes/sec	Token bucket rate
Bytes	Token bucket size
Bytes/sec	Peak data rate
Bytes	Minimum packet size
Bytes	Maximum packet size

شکل ۵-۳۵. مثالی از توصیف جریان.

پارامتر دوم، حجم سطل را برحسب بایت مشخص می کند. به عنوان مثال اگر پارامتر Token Bucket Rate یک مگابایت در ثانیه و حجم سطل 500KByte باشد حتی در صورت توقف ارسال، این سطل می تواند تا چهار ثانیه پر شود و پس از آن سرریز خواهد شد. [سطلی با گنجایش ۵۰۰ کیلوبایت معادل ۴ مگابایت، می تواند جریانی از داده ها با نرخ 1Mbps را بمدت ۴ ثانیه بافر کند. -م]

پارامتر سوم یا Peak Data Rate (حداکثر نرخ ارسال) بیانگر حداکثر نرخ قابل تحمل حتی در بازه های زمانی کوتاه است. فرستنده هیچگاه نباید از این نرخ تجاوز نماید.

دو پارامتر آخر مقدار حداقل و حداکثر طول بسته ها را مشخص می نماید. (این طول شامل سرآیند بسته لایه شبکه و لایه انتقال است). حداقل طول بسته ها نیز اهمیت دارد چراکه پردازش هر بسته (حتی اگر کوچک باشد) مدت زمان ثابتی طول می کشد. یک مسیریاب ممکن است آمادگی پذیرش و هدایت ۱۰,۰۰۰ بسته یک کیلوبایتی در هر ثانیه را داشته باشد ولیکن آمادگی پذیرش و هدایت ۱۰۰,۰۰۰ بسته ۵۰ بایتی در ثانیه را نداشته باشد، هر چند که مورد دوم از لحاظ حجم داده کمتر از مورد اول است. طول حداکثر هر بسته نیز به دلیل محدودیتهای درونی شبکه اهمیت دارد و نباید از این مقدار حداکثر تجاوز شود. به عنوان مثال اگر بخشی از مسیر از درون شبکه اترنت بگذرد، حداکثر طول بسته نباید بیش از ۱۵۰۰ بایت باشد [به دلیل معماری سخت افزار اترنت].

یک سؤال جالب آن است که یک مسیریاب با استفاده از مشخصات توصیفی جریان، چگونه می تواند منابع مورد نیاز را رزرو نماید. نگاشت «مشخصات توصیفی جریان» به میزان منابع مورد نیاز، در مقوله پیاده سازی

می گنجد و استاندارد سازی نشده است. فرض کنید که یک مسیریاب قادر به پردازش صد هزار بسته در ثانیه باشد. اگر «جریان» پیشنهادی با نرخ 1MB/sec و مقدار حداقل و حداکثر طول بسته ها ۵۱۲ بایت باشد، مسیریاب بدین نتیجه می رسد که از این جریان در هر ثانیه ۲۰۴۸ بسته دریافت خواهد کرد و بدین منظور ۲ درصد از CPU خود را برای پردازش این بسته ها در نظر می گیرد. ترجیحاً این مقدار باید بیشتر هم باشد تا از تأخیرات ناشی از ایجاد صفهای طولانی احتراز شود. اگر سیاستهای یک مسیریاب بر این اساس باشد که هیچگاه بیش از ۵ درصد از وقت CPU خود را اختصاص ندهد (یعنی تلویحاً تأخیرات را دو برابر فرض کرده است) و قبلاً ۴۹ درصد از این وقت رزرو شده باشد نمی تواند یک «جریان» با مشخصات فوق را بپذیرد [چرا که به دو درصد از CPU احتیاج دارد در حالی که فقط یک درصد باقیمانده است]. برای منابع دیگر [مثل حافظه و پهنای باند] نیز محاسبات و پیش بینی های مشابه فوق انجام می شود.

توصیف دقیقتر «جریان»، برای مسیریابها مفیدتر خواهد بود. اگر در توصیف یک «جریان» بیان شود که پارامتر Token Bucket Rate (نرخ تولید نشانه در الگوریتم سطل) معادل 5MB/sec است ولیکن طول بسته ها بین ۵ تا ۱۵۰۰ بایت متغیر اعلام شود در این توصیف نادقیق، نرخ ارسال بسته ها بین ۳۵۰۰ تا ۱۰۵۰۰۰ بسته در ثانیه متغیر خواهد بود و ممکن است مسیریاب از عدد ۱۰۵۰۰۰ نگران شده و چنین جریانی را نپذیرد در حالی که اگر مقدار حداقل طول بسته ۱۰۰۰ بایت پیشنهاد شود احتمالاً این جریان پذیرفته خواهد شد.

مسیریابی نسبی

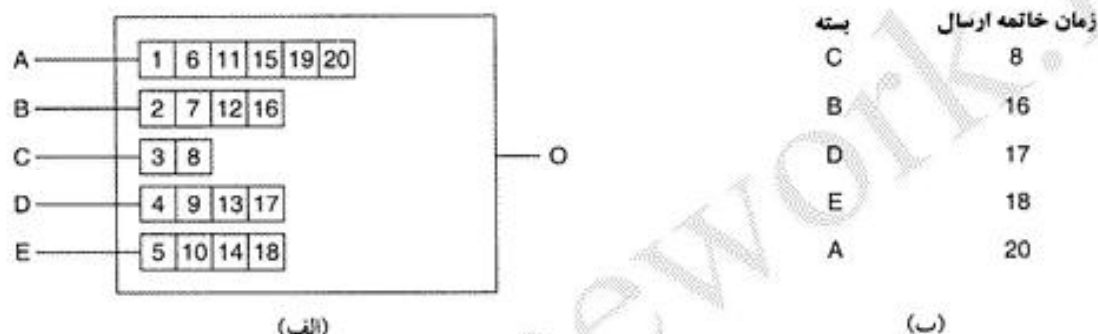
در اغلب الگوریتمهای مسیریابی سعی بر آن است که بهترین مسیر به هر مقصد پیدا شود و پس از آن تمام ترافیک به یک مقصد از مسیر بهینه ارسال خواهد شد. راهکار دیگر برای ارائه کیفیت بهتر خدمات آنست که ترافیک بسته های ارسالی برای یک مقصد، از چندین مسیر هدایت و ارسال شود. از آنجایی که مسیریابها دید جامعی از ترافیک سرتاسر شبکه ندارند لذا تنها راه ممکن برای توزیع ترافیک بر روی چندین مسیر، استفاده از اطلاعات موجود و محلی است. یک روش ساده آن است که ترافیک بطور مساوی یا به تناسب ظرفیت هر یک از خطوط خروجی مسیریاب، بر روی آنها توزیع شود. با این حال الگوریتمهای پیچیده تری در این خصوص وجود دارد. (Nelakuditi and Zhang, 2002)

زمان بندی بسته ها

هر گاه یک مسیریاب هدایت چندین «جریان» را بر عهده داشته باشد این خطر وجود دارد که یک «جریان» از حدود و ظرفیت مجاز خود تجاوز نماید و در نتیجه جریانهای دیگر را با کمبود منابع (starvation) مواجه سازد. اگر پردازش بسته ها به ترتیب ورودشان انجام گیرد باعث می شود که یک فرستنده متجاوز بتواند بیشتر ظرفیت مسیریابی را که بر روی خط سیر بسته های او هستند اشغال کرده و کیفیت خدمات دیگران کاهش یابد. برای خنثی کردن چنین تلاشی، الگوریتمهایی جهت زمان بندی بسته ها پیشنهاد شده است. (Bhatti and Crowcroft, 2000)

یکی از اولین روشها، الگوریتم «صف بندی بی طرفانه» (Fair Queuing) است. (Nagle, 1987) جوهره این الگوریتم آنست که مسیریابها باید برای هر خط خروجی و به ازای هر «جریان» که از آن خط خروجی می گذرد، صفهای جداگانه ای تشکیل بدهند. هر گاه خطی بیکار شود، مسیریاب صفها را به ترتیب پویش کرده و از سر هر صف یکی را بر می دارد. بدین ترتیب، در شرایطی که n ماشین میزبان برای یک خط خروجی رقابت می کنند، از هر n بسته ارسالی بر روی خط یک بسته به هر ماشین میزبان تعلق می گیرد. افزایش نرخ ارسال بسته ها، در نسبت سهم هر ماشین تغییری ایجاد نخواهد کرد.

البته در همین ابتدا، الگوریتم فوق دارای یک مشکل است: آن ماشینهای میزبان که طول بسته‌هایشان بزرگ است نسبت به ماشینهایی که بسته‌های کوچک تولید می‌کنند، سهم بیشتری از پهنای باند را بخود اختصاص خواهند داد. پژوهشگری به نام Demer و همکاران او (۱۹۹۰) پیشنهادی جهت بهبود این الگوریتم ارائه دادند. پیشنهاد آن بود که بجای ارسال یک بسته از هر صف به صورت نوبت چرخشی (Round Robin)، سهم ارسال هر صف بر حسب بایت باشد. در نتیجه صف‌ها بطور متوالی و بایت به بایت پویش شده و بسته‌های هر صف برحسب زمان خاتمه ارسالشان مرتب شده و بهمان ترتیب ارسال می‌شوند. یعنی به جای آنکه از هر صف فقط یک بسته انتخاب شود تعدادی بایت و متناسب با سهم زمانی هر صف ارسال می‌شود. این الگوریتم در شکل ۵-۳۶ به تصویر کشیده شده است.



شکل ۵-۳۶. (الف) مسیریابی که در آن پنج بسته برای خروج از خط O به صف شده‌اند. (ب) زمان خاتمه ارسال این پنج بسته.

در شکل ۵-۳۶ الف بسته‌هایی به طول ۲ تا ۲۰ بایت می‌بینیم. در هر تیک ساعت (مجازی) اولین بایت از بسته دریافتی از خط A ارسال می‌شود. در تیک بعدی اولین بایت از بسته دریافتی از خط B ارسال می‌گردد و کار به همین ترتیب ادامه می‌یابد. اولین بسته‌ای که ارسال آن پس از هشت تیک ساعت خاتمه خواهد یافت بسته C است.^۱ در شکل ۵-۳۶ ب فهرست مرتب شده بسته‌ها به ترتیب ارسال مشخص شده است. هرگاه بسته جدیدی دریافت نشود بسته‌ها به ترتیب فوق‌الذکر (از C تا A) ارسال خواهد شد.

یک اشکال این الگوریتم آن است که به تمام ماشینهای میزبان، اولویت یکسانی می‌دهد. در بسیاری از محیط‌ها مطلقاً آن است که به سرویس دهنده‌های ویدیو (Video Server) اولویت بیشتری نسبت به یک سرویس دهنده معمولی فایل داده شود و در هر تیک ساعت، سهم آن دو یا چند بایت باشد. این الگوریتم اصلاح شده به نام «الگوریتم صف‌بندی بی‌طرفانه وزن‌دار» (Weighted Fair Queuing) مشهور است و کاربرد گسترده‌ای دارد. گاهی اوقات وزن هر صف معادل یا تعداد «جریان» (Flow) منشعب از یک ماشین در نظر گرفته می‌شود و بدین ترتیب هر پروسه مولد جریان، پهنای باند یکسانی دریافت می‌دارد.^۲ روش پیاده‌سازی مؤثر این الگوریتم در مرجع (Shreedhar & Varghese, 1995) تشریح شده است. امروزه در عمل، هدایت بسته‌ها توسط سخت‌افزار مسیریاب یا سوئیچ انجام می‌شود و الگوریتم‌های فوق بر روی سخت‌افزاری پیاده‌سازی شده‌اند. (Elhanany et al., 2001)

۱. دقت کنید که شماره‌هایی که درون مربع‌های هر بسته از شکل ۵-۳۶ نوشته شده شماره ترتیب ارسال تلقی می‌شود و هر مربع شماره‌دار صرفاً معادل یک بایت است. -م

۲. به عبارت دیگر به جای آنکه به ماشینهای میزبان پهنای باند مساوی داده شود به پروسه‌های هر ماشین پهنای باند یکسان داده می‌شود. -م

۳-۴-۵ خدمات مجتمع (Integrated Services)

در خلال سالهای ۱۹۹۵ تا ۱۹۹۷، تلاش IETF بر آن بود که برای انتقال داده های مالتی مدیا (Multimedia Streaming) معماری مناسبی ابداع کند. نتیجه کار چندین RFC به شماره های ۲۲۰۵ تا ۲۲۱۰ بود. این پروژه با نام کلی «الگوریتمهای مبتنی بر جریان» (Flow-based algorithms) یا «خدمات مجتمع» (Integrated Services) شناخته می شود و کاربردهای چندپخش (Multicast) و تک پخش (Unicast) را در بر می گیرد. به عنوان یک نمونه از کاربردهای تک پخش، کاربری را در نظر بگیرید که قطعه ای ویدیو را از یک سایت تماشا می کند. به عنوان مثالی از کاربردهای چندپخش، ایستگاههای پخش تلویزیون دیجیتال را در نظر بگیرید که برنامه های خود را در قالب جریانی از بسته های IP به گیرندگان بی شمار و پراکنده خود ارسال می دارند. در ادامه بر روی کاربردهای چندپخش متمرکز خواهیم شد چرا که کاربردهای تک پخش حالت خاص چندپخش هستند. در اغلب کاربردهای چندپخش، گروههای مختلف می توانند به صورت پویا عضویت خود را تغییر بدهند: مثلاً افرادی را مجسم کنید که در یک کنفرانس ویدیویی وارد می شوند و پس از مدتی حوصله آنها سر می رود و به یک کانال پخش آواز می پیوندند! در چنین شرایطی روش رزرو پهنای باند به خوبی کار نخواهد کرد چرا که هر فرستنده باید دائماً ورود و خروج اعضای خود را پیگیری نماید. برای سیستمی که مثلاً برای پخش تلویزیونی طراحی شده و میلیونها مشترک دارد این روشها هرگز کار نخواهد کرد!

۱. RSVP: پروتکل رزرو منابع

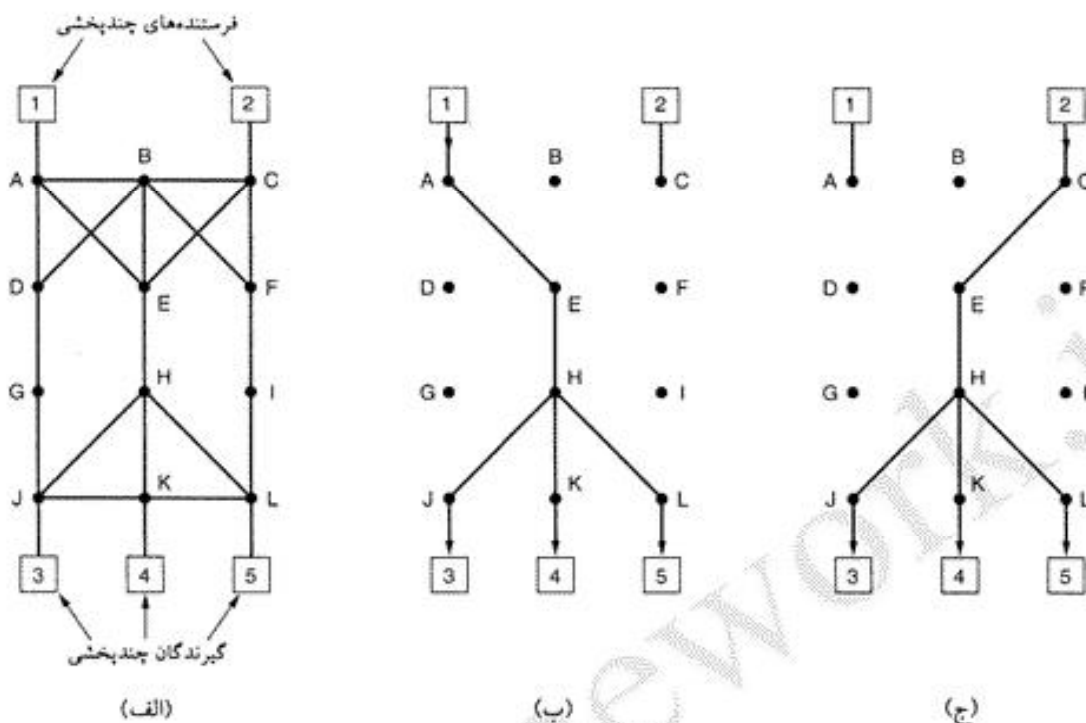
اصلیترین پروتکل پیشنهاد شده توسط IETF برای ارائه خدمات مجتمع، RSVP نامیده می شود. این پروتکل در RFC 2205 تشریح شده است و برای رزرو کردن پهنای باند بکار می آید. پروتکلهای دیگری که در RFC 2206 تا RFC 2210 تشریح شده اند چگونگی ارسال داده ها را توصیف می کنند. RSVP اجازه می دهد که چندین فرستنده بتواند برای چندین گروه از گیرندگان خود داده بفرستد و همچنین امکان آن را فراهم کرده که گیرندگان بتوانند کانال مورد نظر خود را آزادانه عوض کنند. در عین حال پروتکل RSVP، استفاده از پهنای باند را بهینه سازی کرده و از بروز ازدحام جلوگیری می کند.

در ساده ترین حالت، این پروتکل از روش «مسیریابی چندپخش مبتنی بر درخت پوشا»^۲ که قبلاً تشریح شد، بهره می گیرد. به هر گروه یک آدرس یکتا انتساب داده می شود و برای ارسال یک بسته به گروه خاص، آدرس آن گروه در بسته قرار می گیرد. سپس توسط الگوریتم استاندارد مسیریابی چندپخش، یک درخت پوشا که تمام اعضای آن گروه را در بر می گیرد، ایجاد می گردد. الگوریتم مسیریابی چندپخش جزو استاندارد RSVP محسوب نمی شود و تنها تفاوت آن با الگوریتم معمولی مسیریابی چندپخش آنست که بطور متناوب، مقداری اطلاعات اضافی برای هر گروه ارسال می شود تا مسیریابهای واقع بر روی درخت، آنها را در ساختمان داده خاصی در حافظه خود ذخیره نمایند.

به عنوان مثال شبکه شکل ۵-۳۷-الف را در نظر بگیرید. ماشینهای میزبان ۱ و ۲ فرستنده های چندپخش و ماشینهای ۳ و ۴ و ۵ گیرندگان چندپخش هستند. در این مثال فرستنده ها و گیرنده ها کاملاً از هم جدا (Disjoint) هستند ولی در حالت کلی ممکن است مجموعه ماشینهای فرستنده و گیرنده عضو مشترک هم داشته باشند. درختهای چندپخش برای فرستنده شماره ۱ و شماره ۲ در شکلهای ۵-۳۷-ب و ۵-۳۷-ج نشان داده شده اند. برای دریافت بهتر و جلوگیری از ازدحام، هر یک از گیرندگان یک گروه می توانند پیامی برای رزرو پهنای باند

۱. Resource Reservation Protocol

۲. Spanning Tree Multicast Routing (به بخش ۵-۲-۸ مراجعه کنید).

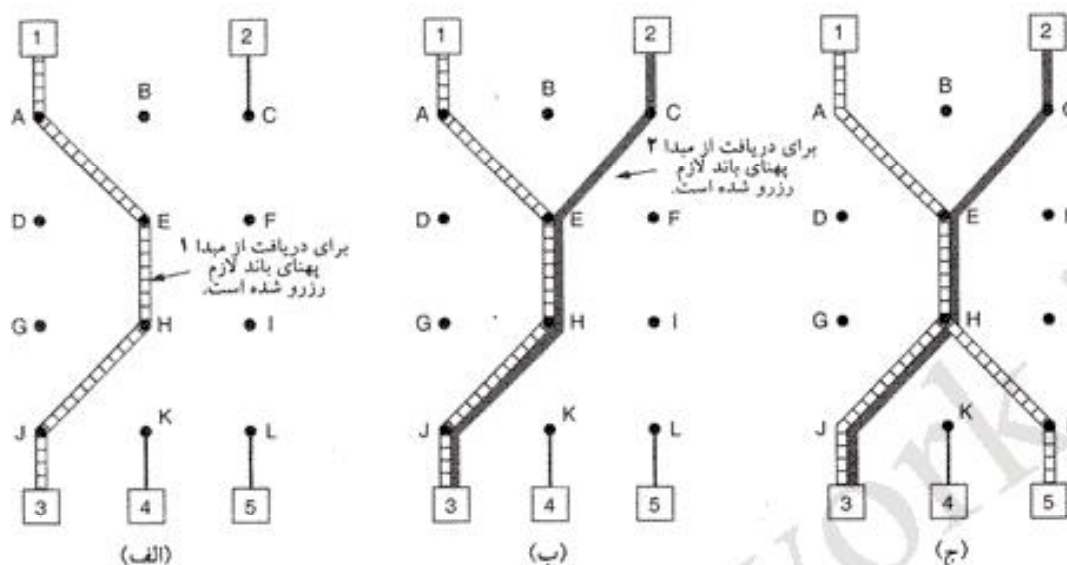


شکل ۵-۳۷. (الف) ساختار یک شبکه (ب) «درخت پوشای چندبخشی» برای ماشین میزبان ۱ (ج) «درخت پوشای چندبخشی» برای ماشین میزبان ۲.

ارسال کنند تا از طریق درخت به فرستنده برسد. این پیام به کمک «الگوریتم هدایت در مسیر معکوس»^۱ که قبلاً تشریح شد، در درخت انتشار می‌یابد. در هر گام، مسیریاب به این پیام توجه کرده و پهنای باند لازم را رزرو می‌نماید. اگر پهنای باند کافی موجود نباشد، «پیغام شکست» برخواهد گشت. اگر این پیغام موفقیت‌آمیز به مبدا آن برگردد، پهنای باند لازم در کل مسیر رزرو شده است.

مثالی از چگونگی رزروسازی در شکل ۵-۳۸-الف نشان داده شده است. در اینجا ماشین میزبان ۳ تقاضای برقراری یک کانال با ماشین ۱ را داده است. به محض برقراری چنین کانالی، بسته‌ها می‌توانند بدون مسئله ازدحام از ۱ به ۳ جاری شوند. حال ببینیم اگر ماشین ۳ بعداً بخواهد کانالی دیگر با ماشین میزبان ۲ برقرار کند چه اتفاقی می‌افتد (زیرا مثلاً می‌خواسته دو کانال تلویزیونی را بطور همزمان تماشا کند). مسیر دوم مطابق شکل ۵-۳۸-ب رزرو می‌شود. دقت کنید که از ماشین گیرنده ۳ تا مسیریاب E به دو کانال مستقل نیاز است چراکه قرار است دو «استریم ویدیویی مستقل»^۲ ارسال شود.

در شکل ۵-۳۸-ج، ماشین میزبان ۵ تصمیم می‌گیرد برنامه‌ای را که توسط ماشین ۱ پخش می‌شود تماشا کند و او نیز رزرواسیون لازم را انجام می‌دهد. ابتدا پهنای باند لازم تا H رزرو می‌شود. از H به بعد، مسیریابها متوجه می‌شوند که از قبل کانالی با ماشین ۱ دارند و از داده‌های آن تغذیه می‌شوند، لذا نیازی به رزرو نیست. دقت کنید که ممکن است ماشینهای ۳ و ۵ به پهنای باند متفاوتی نیاز داشته باشند (چرا که مثلاً ماشین ۳ تصاویر تلویزیونی را سیاه و سفید نگاه می‌کند و نیازی به اطلاعات رنگی ندارد)، فلذا ظرفیت رزرو شده باید آنقدر زیاد باشد تا بتواند از گیرنده‌ای را که به بیشترین مقدار پهنای باند احتیاج دارد، برآورده سازد.



شکل ۵-۳۸. (الف) ماشین میزبان ۳ تقاضای کانالی به ماشین ۱ می کند. (ب) ماشین میزبان ۳ تقاضای کانالی دیگر به ماشین ۲ می کند. (ج) ماشین میزبان ۵ تقاضای کانالی به ماشین ۱ می کند.

در RSVP به هر گیرنده آزادی عمل داده شده تا بتواند با یکبار عملیات رزرو، کانالهایی با بیش از یک فرستنده، ایجاد نماید. همچنین می تواند مشخص کند که آیا انتخاب او برای دوره ای از زمان ثابت است یا آنکه حق تغییر مبدا ارسال (فرستنده) را برای خود محفوظ نگاه داشته است. مستیرباب می تواند به کمک این اطلاعات برآورد بهینه و مناسبی از پهنای باند داشته باشد.

دلیل اتخاذ این استراتژی در محیطهای کاملاً پویا و در حال تغییر آنست که پهنای باند رزرو شده از انتخاب مبدا (فرستنده) مستقل و مجزاست: به محض آنکه یک گیرنده که قبلاً پهنای باند لازم را برای دریافت از یک مبدا خاص رزرو کرده، بخواهد مبدا دریافت خود را تغییر بدهد بخشی از مسیر قبلی برای مبدا جدید نیز معتبر است و نیازی به تخصیص پهنای باند ندارد. در ضمن اگر ماشین ۲ در حال ارسال همزمان چندین جریان ویدیویی (Video Streams) باشد، ماشینی مثل ۳ می تواند بدون نیاز به هیچگونه رزرواسیون مجدد و تغییر، از بین این کانالهای ویدیویی یکی را انتخاب نماید زیرا مسیربابها به آنچه که گیرنده تماشا می کند دقت و اعتنائی نمی کنند.^۱

۵-۴۵ خدمات متمایز (Differentiated Services)

«الگوریتمهای مبتنی بر جریان»^۲ قابلیت عرضه کیفیت خوب خدمات به یک یا چند جریان را دارند زیرا در طول مسیر هر منبعی را که نیاز است از قبل رزرو می کنند. ولی این روشها یک اشکال دارند: در این الگوریتمها نیاز است که برای هر جریان (Flow)، پیشاپیش تنظیمات لازم انجام شود در حالی که در مقیاس کلان یعنی وقتی که هزاران یا میلیونها «جریان» وجود دارد قابلیت اجرایی خود را از دست می دهند. از طرفی در هر مسیرباب «وضعیت» هر

۱. به عبارت روشنتر وقتی یک گیرنده مثل A کانالی با پهنای باند معین با فرستنده B رزرو می کند و B بطور همزمان مثلاً ده استریم ویدیویی بخش می کند، گیرنده A در انتخاب یکی از این ده استریم آزادی عمل دارد و نیازی نیست که به مسیربابها در این خصوص اطلاع داده شود چرا که آنها پهنای باند لازم را رزرو کرده اند و اصراری ندارند که بدانند فرستنده چه چیزی برای گیرنده

۲. Flow-Based Algorithms.

می فرستند. - م

جریان بطور جداگانه نگهداری می شود و عملکرد این الگوریتم ها در مقابل خرابی یک مسیریاب آسیب پذیر خواهد بود. نهایتاً آنکه برای تنظیم و ایجاد «جریان» باید تبادل اطلاعات پیچیده ای بین مسیریابها انجام گیرد. در نتیجه RSVP یا الگوریتمهای مشابه آن، بسیار کم پیاده سازی عملی شده اند.

به همین دلایل، IETF راهکاری ساده تر برای تأمین کیفیت خدمات (QoS) ابداع کرد؛ روشی که بدون نیاز به هیچ تنظیمات قبلی یا تعیین کل مسیر، می تواند به صورت محلی و مجزا در هر مسیریاب پیاده سازی شود. این راهکار اصطلاحاً «روش مبتنی بر کلاس» (Class-Based) برای تضمین کیفیت خدمات نامیده می شود (در مقابل روشهای مبتنی بر جریان). IETF یک معماری مناسب به نام «خدمات متمایز» برای آن طراحی و استانداردسازی کرده است که در مستندات RFC به شماره های ۲۴۷۴ و ۲۴۷۵ و مستندات دیگر تشریح شده است. در ادامه به تشریح این روش می پردازیم.

«خدمات متمایز» (که به اختصار DS گفته می شود) می تواند توسط مجموعه ای از مسیریابها که در یک «حوزه مدیریت واحد» (Administrative Domain) قرار می گیرند (مثلاً یک ISP یا شرکت مخابرات)، عرضه شود. مدیریت مسئول شبکه، مجموعه ای از کلاسهای متفاوت خدمات و متناظر با آن، قواعد هدایت بسته ها (Forwarding Rules) را تعریف می کند.

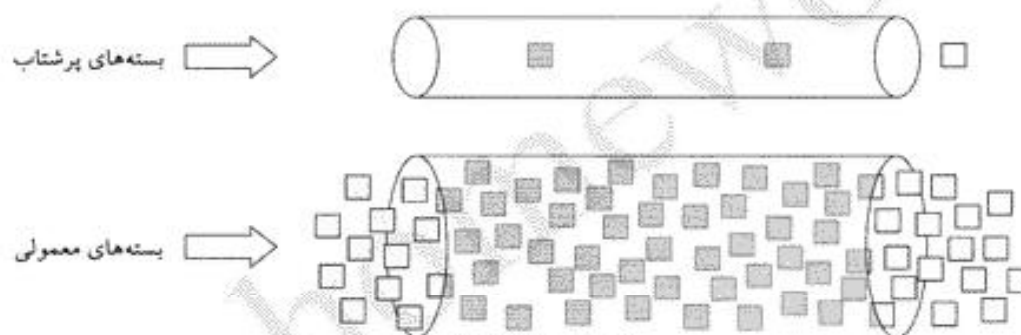
اگر یک مشتری برای دریافت خدمات نوع DS تقاضای ورود به شبکه را بدهد، بسته های ارسالی او در ورود به حوزه، فیلد «نوع خدمات» (Type of Service) را با خود حمل می کنند تا به برخی از آنها خدمات بهتری (مثل خدمات ویژه^۱) ارائه شود. ممکن است لازم باشد ترافیک تعریف شده در هر کلاس از شکل خاصی پیروی نماید (مثلاً باید از الگوریتم سطل سوراخ با نرخ خروجی مشخص تبعیت کند). متصدی شبکه با گرایشهای اقتصادی و تجاری ممکن است برای انتقال «بسته های ویژه» (Premium Packets) هزینه اضافی بگیرد یا مثلاً به ازای بهای اشتراک ثابت و ماهانه، تعداد N بسته ویژه از کاربر پذیرفته و هدایت شود. دقت کنید که این الگو نیاز به تنظیمات قبلی، رزروسازی منبع و نیازی به اتلاف وقت برای مذاکره بین طرفین نهایی در هر «جریان» ندارد. به همین دلیل پیاده سازی خدمات DS بسیار آسان است.

«خدمات مبتنی بر کلاس» در صنایع دیگر نیز وجود دارد. به عنوان مثال، شرکتهای تحویل محموله های پستی (Package Delivery) نیز سه نوع خدمات عرضه می کنند: شبانه روزی، دو روزه یا سه روزه. یا مثلاً خطوط هواپیمایی «خدمات کلاس برتر» (First Class)، «کلاس تجاری» (Business Class) و «کلاس معمولی» عرضه می کنند. در قطارهای دورپیما نیز خدمات در کلاسهای متفاوتی عرضه می شود و حتی قطارهای زیرزمینی (مترو) نیز در دو کلاس مختلف خدمات ارائه می کنند. برای بسته های حاوی اطلاعات، کلاسهای متفاوت خدمات برحسب میزان «تأخیر»، «لرزش» (Jitter)، احتمال حذف بسته در صورت بروز ازدحام و امکاناتی نظیر همینها تعیین می شود.

برای آنکه تفاوت بین «کیفیت خدمات مبتنی بر جریان» و «کیفیت خدمات مبتنی بر کلاس» روشنتر شود نمونه ای مثل «تلفن اینترنتی» را مدنظر قرار بدهید. در روش مبتنی بر جریان، هر تماس تلفنی منابع خاص خود و تضمینهای لازم را شبکه اخذ می کند. در روش مبتنی بر کلاس تمام تماسهای تلفنی همگی از منابع رزرو شده ای که برای «کلاس تلفنی» تهیه دیده شده، استفاده می کنند. این منابع در اختیار بسته هایی که در کلاس انتقال فایل یا کلاسهای دیگر هستند، قرار نمی گیرد و صرفاً برای «کلاس تلفنی» پیش بینی شده است ولی اینگونه هم نیست که برای هر تماس تلفنی منابع اختصاصی و مجزا در نظر گرفته شود.

هدایت پُرشتاب (Expedited Forwarding)

انتخاب کلاس خدمات بر عهده کارفرمای شبکه است ولیکن از آنجایی که بسته ها از چندین زیرشبکه مجزا (یا کارفرمای مستقل) عبور می کنند (و ممکن است کلاس خدمات هر زیرشبکه متفاوت و سلیقه ای باشد)، IETF در حال کار بر روی تعریفی واحد برای کلاسهای خدمات است به گونه ای که مستقل از نوع شبکه باشد. ساده ترین کلاس، کلاس «هدایت پرشتاب» است که با آن شروع می کنیم. این کلاس در RFC 3246 تشریح شده است. ایده ای که در پشت روش «هدایت پرشتاب» نهفته است ساده به نظر می رسد. خدمات در دو کلاس قابل ارائه است: «معمولی» و «پرشتاب» (Regular & Expedited). بخش اعظم ترافیک شبکه از نوع معمولی هستند در حالی که فقط کسر کوچکی از آن نیاز به «خدمات پُرشتاب» دارند. بسته های پُرشتاب باید بگونه ای در زیرشبکه حرکت کنند که گویی هیچ بسته دیگری وجود ندارد. توصیفی نمادین از مفهوم سیستم «دو تونلی» (Two-Tube) در شکل ۵-۳۹ ارائه شده است. به خاطر داشته باشید که کماکان یک خط فیزیکی در اختیار است؛ دو لوله منطقی که در شکل نشان داده شده فقط نماد رزرو بخشی از پهنای باند هستند نه آنکه یک خط فیزیکی دیگر هم وجود داشته باشد.



شکل ۵-۳۹. بسته های پرشتاب با شبکه ای بدون ترافیک مواجه می شوند.

یکی از روشهای پیاده سازی این استراتژی آن است که مسیریابها به نحوی برنامه ریزی شوند که برای هر یک از خطوط خروجی خود دو صف مجزا تشکیل بدهند: یکی برای بسته های پرشتاب و دیگری برای بسته های معمولی. بسته های دریافتی برحسب نوع آنها به یکی از این صفها وارد می شوند. برای زمان بندی بسته ها می توان از روشی مثل «روش صف بندی وزن دار» (Weighted Queuing) بهره گرفت. مثلاً اگر ده درصد از بسته ها از نوع پرشتاب و مابقی از نوع معمولی باشند، تخصیص ۲۰ درصد از پهنای باند برای ترافیک پرشتاب و هشتاد درصد باقیمانده برای ترافیک معمولی مناسب خواهد بود. با این کار پهنای باند اختصاص داده شده به ترافیک پرشتاب دو برابر مقدار مورد نیاز آن است و بدین ترتیب تأخیر پائینی خواهد داشت. برای اجرای چنین راهکاری می توان به ازای ارسال ۴ بسته معمولی یک بسته پرشتاب ارسال کرد (البته با فرض آن که اندازه بسته ها توزیعی مشابه و یکنواخت داشته باشد). انتظار می رود در این روش حتی در صورت سنگین بودن بار زیرشبکه، بسته های پرشتاب زیرشبکه را بی بار و خلوت ببینند.

هدایت تضمین شده (Assured Forwarding)

برای مدیریت انواع کلاسهای خدمات، روشی دقیقتر به نام «هدایت تضمین شده» ارائه و در RFC 2597 تشریح شده است. در این روش چهار کلاس اولویت تعریف شده و هر کلاس منابع خاص خود را در اختیار دارد. علاوه بر این، سه احتمال برای حذف بسته در اثر بروز ازدحام تعریف شده است: احتمال پایین، متوسط و زیاد. مجموع

ترکیبات مختلف این دو عامل، دوازده کلاس خدمات متفاوت ایجاد می کند. در شکل ۴۰-۵ یک روش برای پردازش بسته ها به منظور هدایت تضمینی آنها، نشان داده شده است. در مرحله اول بسته ها برحسب کلاس اولویتشان در یکی از چهار کلاس، رده بندی می شوند. این مرحله می تواند در ماشین میزبان فرستنده بسته ها انجام شود (به نحوی که در شکل نشان داده شده است) یا آنکه در اولین مسیر یاب (مدخل ورودی به زیر شبکه) انجام شود.



شکل ۴۰-۵. پیاده سازی مکانیزم هدایت تضمین شده برای یک «جریان داده».

در مرحله ۲ بسته ها بر حسب کلاسشان علامتگذاری می شوند. بدین منظور در سرآیند هر بسته به فیلد خاصی نیاز است. خوشبختانه یک فیلد هشت بیتی به نام Type of Service (نوع خدمات) در بسته IP وجود دارد که در آینده آن را به اختصار بررسی خواهیم کرد. در RFC 2597 شش بیت از این هشت بیت برای تعیین کلاس بسته ها تعریف شده و دو بیت باقیمانده برای استفاده هایی که از قبل داشته یا استفاده در آینده، رها شده اند.

در مرحله سوم بسته ها از یک «فیلتر شکل دهنده / حذف کننده» (Shaper / Dropper Filter) عبور کرده و برای آنکه ترافیک بسته های هر یک از چهار کلاس شکل قابل قبولی داشته باشند به برخی از آنها تأخیر مصنوعی تحمیل می شود (مثلاً به کمک الگوریتم سطل سوراخ یا سطل نشانه دار). در صورتی که تعداد بسته ها در هر کلاس، از حد مجاز بیشتر شده باشد در این مرحله برخی از آنها حذف می گردند. (روشهای دقیقتری نیز برای حذف بسته ها وجود دارد که از فیدبک بهره می گیرند).

در این مثال هر سه مرحله فوق الذکر توسط ماشین فرستنده انجام شده و جریان خروجی بسته ها به اولین مسیر یاب ارسال می شود. بدیهی است که این مراحل می تواند توسط یک نرم افزار خاص شبکه یا سیستم عامل هر ماشین انجام شود تا نیازی به تغییر در برنامه های کاربردی موجود نباشد.

۵-۴-۵ سوئیچ برچسب و MPLS

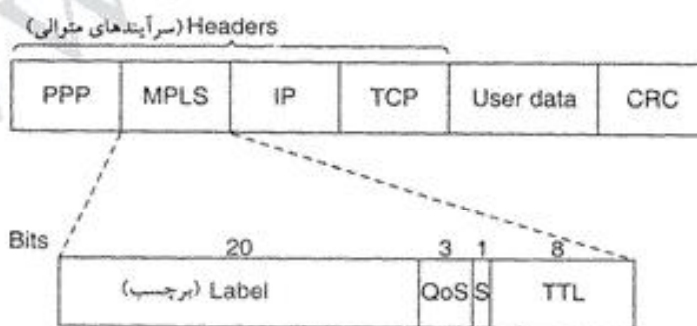
در خلال زمانی که IETF بر روی موضوع «خدمات مجتمع» کار می کرد، چندین تولیدکننده محصولات مسیریابی نیز بر روی روشهای بهتر هدایت بسته ها متمرکز شده بودند. کار آنها بر این محور بود که در ابتدای هر بسته یک «برچسب» (Label) اضافه شود و بجای آنکه مسیریابی و هدایت بسته ها مبتنی بر آدرس مقصد باشد براساس این «برچسب» انجام شود. با استفاده از این «برچسب» به عنوان یک اندیس در جدول داخلی هر مسیر یاب، خط خروجی صحیح و مناسب برای هر بسته پیدا می شود. به کمک این روش، مسیریابی بسته ها به سرعت انجام شده و

منابع مورد نیاز در طول مسیر رزرو خواهد شد.

البته برچسب گذاری بر روی هر «جریان» شباهت عجیبی به مدارهای مجازی پیدا می کند. در شبکه های ATM، X.25 و Frame Relay یا هر زیر شبکه مدار مجازی دیگر نیز یک «برچسب» (یا به عبارتی یک شناسه مدار مجازی)^۱ در هر بسته قرار داده می شود و با استفاده از آن به عنوان یک اندیس برای درایه های جدول^۲، مسیر مناسب بدست می آید. علیرغم آنکه بسیار از افراد در جامعه اینترنت از شبکه های اتصال گرا بشدت گریزان هستند، به نظر می رسد که این ایده با هدف مسیریابی سریع و تأمین کیفیت خدمات (QoS) بار دیگر به صحنه برگشته است. ولیکن بین روشی که در اینترنت برای تعیین مسیر بکار می رود و روشی که در شبکه های مدار مجازی اعمال می شود تفاوت های بنیادی وجود دارد و تکنیک برچسب گذاری بسته ها با روش سنتی سوئیچینگ متفاوت است. ایده جدید سوئیچینگ با نامهای متنوعی مثل «سوئیچینگ برچسب»^۳ یا «سوئیچینگ علامت»^۴ شناخته می شود. در نهایت IETF آن را تحت نام MPLS^۵ استاندارد کرد. ما نیز در ادامه از نام MPLS استفاده می کنیم. این استاندارد در RFC 3031 و چندین RFC دیگر تشریح شده است.

مضاف بر این، برخی افراد بین «مسیریابی» و «سوئیچینگ» فرق می گذارند. مسیریابی فرایند جستجو در جدول مسیریابی به دنبال آدرس مقصد هر بسته و پیدا کردن خط مناسب برای آن است. برعکس در فرایند سوئیچینگ از برچسب هر بسته به عنوان یک اندیس در جدول مسیریابی استفاده می شود و با استفاده از این اندیس بلافاصله خط خروجی پیدا می شود، بدون آن که نیازی به جستجو باشد. البته این تعاریف و تعبیر جهان شمول و همگانی نیستند.

اولین مسئله آنست که این برچسب در کجا قرار داده شود. از آنجایی که بسته های IP برای شبکه های مدار مجازی طراحی نشده بودند، طبعاً هیچ فیلدی در سرآیند بسته IP برای درج شماره های مدار مجازی وجود ندارد. به همین دلیل سرآیند جدید MPLS، باید در جلوی سرآیند هر بسته IP قرار بگیرد. در خطوط مستقیم بین هر دو مسیریاب که مبتنی بر «فریمینگ PPP» کار می کنند ترتیب سرآیندها طبق شکل ۵-۴۱ عبارتند از: سرآیند PPP، سرآیند MPLS، سرآیند IP و نهایتاً سرآیند TCP. در واقع باید MPLS را در لایه ۲/۵ فرض کرد!!!



شکل ۵-۴۱. ارسال یک قطعه TCP (TCP Segment) با استفاده از IP، MPLS، و PPP.

سرآیند عمومی MPLS (MPLS Header) چهار فیلد دارد که مهمترین آنها فیلد Label (فیلد برچسب) است که در آن یک اندیس درج می شود. فیلد QoS، کلاس خدمات را مشخص می کند. فیلد S بدان منظور تعریف شده که در شبکه های سلسله مراتبی چندین سرآیند MPLS متوالیاً به بسته اضافه گردد. (این موضوع در زیر تشریح

شده است). فیلد TTL زمان حیات بسته را مشخص می‌کند و به ازای هر گام یک واحد از آن کم می‌گردد؛ هر گاه مقدار این فیلد به صفر برسد، بسته حذف می‌شود. این ویژگی بدان منظور مفید است که از حلقه بی‌نهایت که در اثر ناپایداری (عدم همگرایی) جدول مسیریابی بروز می‌کند، اجتناب شود.

از آنجایی که سرآیند MPLS بخشی از بسته لایه شبکه یا فریم لایه پیوند داده‌ها محسوب نمی‌شود لذا MPLS تا حد زیادی مستقل از هر دو لایه است. از بین تمام محاسن دیگر، دستاورد ویژگی «استقلال از دیگر لایه‌ها» آنست که می‌توان سوییچهای MPLS را به گونه‌ای ساخت که بتواند هم بسته‌های IP و هم سلولهای ATM را برحسب مورد، هدایت کند. این ویژگی همانی است که براساس آن کلمه Multiprotocol در ابتدای نام MPLS ظاهر شده است.

وقتی یک بسته یا سلول غنی شده با سرآیند MPLS در یک مسیریاب MPLS دریافت می‌شود از برحسب آن به عنوان اندیسی در جدول داخلی مسیریاب استفاده شده و خط خروجی متناسب با آن تعیین می‌شود و قبل از خروج بسته از آن خط، برحسب جدیدی در فیلد مربوطه درج می‌گردد. تغییر در برحسبها در تمام زیرشبکه‌های مدار مجازی معمول و متعارف است چرا که برحسبها در هر مسیریاب معنای محلی دارند و دو مسیریاب متفاوت ممکن است بسته‌های نامربوط را با برحسبی یکسان برای مسیریاب دیگر بفرستند چرا که این بسته‌ها همگی در بخشی از مسیر مشترکند.^۱ به همین دلیل در هر گام برحسبهای بسته قبل از انتقال بر روی خط خروجی به برحسب جدید و معتبر در مسیریاب بعدی نگاشته می‌شود. این مکانیزم را در شکل ۵-۳ مشاهده کردیم. MPLS نیز از روش مشابهی بهره گرفته است.

یکی از تفاوت‌های MPLS با شبکه‌های مدار مجازی، «میزان تجمیع» (Aggregation Level) و صرفه‌جویی در تعداد درایه‌های جدول^۲ مسیریابی است. در MPLS این امکان وجود دارد که هر «جریان» در زیرشبکه، دارای مجموعه برحسبهای خاص خود باشد ولی این قابلیت مهم نیز وجود دارد که گروهی از جریانها که همگی به یک مسیریاب خاص یا یک LAN ختم می‌شوند با برحسب یکسان و واحدی مشخص شوند. [بدین ترتیب تعداد درایه‌های جدول مسیریابی کاهش یافته و اصطلاحاً عمل تجمیع یا Aggregation انجام می‌شود.] گروهی از جریانها که با یک برحسب واحد مشخص می‌شوند اصطلاحاً به یک FEC^۳ مشابه متعلق هستند. [یعنی همه آنها به یک طریق هدایت می‌شوند.] کلاس FEC نه تنها مقصد همه بسته‌ها را مشخص می‌کند، بلکه کلاس خدمات مورد نیاز آنها را نیز تعیین می‌نماید (از دیدگاه انواع خدمات متمایز که در بخش ۵-۴-۴ بدان اشاره شد). طبعاً فرآیند هدایت تمام بسته‌های یک FEC یکسان خواهد بود.

در مسیریابی متعارف به روش مدار مجازی، این قابلیت که بتوان چندین مسیر مجزا با نقاط پایانی متفاوت را با «شناسه مدار مجازی» واحد مشخص کرد وجود ندارد چرا که در این صورت راهی برای مشخص کردن مقصد نهایی بسته‌ها وجود نخواهد داشت در حالی که در MPLS بسته‌ها [به غیر از سرآیند ۴ بایتی MPLS] آدرس واقعی ماشین مقصد را نیز با خود حمل می‌کنند و بدین ترتیب در انتهای مسیری که با برحسب مشخص شده می‌توان سرآیند حاوی برحسب را حذف کرد و هدایت بسته‌ها به روش معمول و مبتنی بر آدرس لایه شبکه [مثل آدرس IP] ادامه یابد.

یکی از تفاوت‌های بنیادی بین MPLS و شبکه‌های مدار مجازی در چگونگی تشکیل جداول مسیریابی است.

۱. عبارت دیگر برحسب هر بسته هویت آنرا مشخص نمی‌کند بلکه مسیر خروج آن از مسیریاب فعلی را مشخص می‌کند لذا ضمیمی است که بسته‌های خروجی از یک مسیریاب که هیچ ربطی بهم ندارند ولی لااقل گام بعدی مسیر آنها یکی است (یعنی از خط مشابهی وارد مسیریاب بعدی و از خط مشابهی، از آن خارج می‌شوند) دارای برحسب یکسانی باشند. -م

در شبکه های مدار مجازی وقتی یک کاربر بخواهد یک «اتصال» ایجاد کند، توسط لایه شبکه یک بسته خاص جهت تنظیم مسیر به زیرشبکه روانه می شود تا ضمن ایجاد یک مسیر درایه های لازم در جداول مسیریابی درج شود. MPLS بدین نحو عمل نمی کند چرا که در آن عمداً هیچ مرحله ای برای تنظیم اتصال پیش بینی نشده است. (زیرا در غیر این صورت نرم افزارهای موجود اینترنت دچار شکاف و ناسازگاری می شد.) در عوض برای تنظیم و ایجاد درایه های جدول مسیریابی از دو راهکار جدید استفاده شده است.

در راهکار اول که «روش متکی به داده» (Data driven) نامیده می شود هرگاه بسته ای در اولین مسیریاب دریافت شود، آن مسیریاب با مسیریاب واقع بر روی مسیر جریان، تماس گرفته و از او می خواهد که یک برچسب برای این جریان ایجاد نماید. این فرآیند به صورت بازگشتی (Recursive) ادامه می یابد تا مجموعه برچسبها ایجاد شوند. در واقع این روش را می توان ایجاد «مدار مجازی برحسب تقاضا»^۱ فرض کرد.

پروتکل هایی که عمل برچسب دهی را انجام می دهند مراقب هستند تا از بروز حلقه اجتناب شود. برای این کار از تکنیکی به نام «رسمانهای رنگی» (Colored Thread) بهره گرفته می شود. انتشار معکوس یک FEC را می توان با یک «رسمان رنگی و یکتا» [تمثیلی از یک مسیر در شبکه] در زیرشبکه مقایسه کرد. اگر مسیریاب رنگی را مشاهده کند که خودش نیز به همان رنگ است متوجه می شود که در انتخاب مسیر، حلقه ایجاد شده و برای رفع آن اقدام می کند.^۲ روش «برچسب دهی متکی به داده» (Data driven) در شبکه هایی کاربرد دارد که زیر ساخت انتقال آنها ATM است. (همانند بیشتر سیستم های تلفن)

راهکار دیگر برای برچسب دهی به جریان داده ها، در شبکه هایی کاربرد دارد که زیربنای آنها ATM نیست. این روش اصطلاحاً «روش متکی به کنترل» (Control Driven) نامیده می شود و گونه های متنوعی از آن وجود دارد. یکی از این گونه ها به ترتیب ذیل عمل می کند: وقتی یک مسیریاب راه اندازی (بوت) می شود ابتدا بررسی می کند که در انتهای چه مسیرهایی قرار دارد (یعنی مثلاً چه ماشین هایی بر روی LAN متصل به او قرار دارند). سپس برای تمام آنها یک یا چند FEC [شناسه یک گروه با کلاس معادل] تولید کرده و ضمن تخصیص یک برچسب به هر یک از این گروه ها، آنها را به همسایه های خود اطلاع می دهد. آنها نیز به ترتیب برچسبها را در جدول مسیریابی خود وارد کرده و با تعیین برچسبی جدید [متناظر با هر برچسب قبلی] آنها را به همسایه های خود اطلاع می دهند تا آنکه تمام مسیریابها از مسیرهای جدید آگاه شوند. در حین ایجاد مسیر می توان منابع لازم را نیز برای تضمین کیفیت خدمات رزرو کرد.

MPLS می تواند بطور همزمان در چندین سطح عمل کند. در بالاترین سطح، هر زیرشبکه حامل را می توان یک نوع Metarouter فرض کرد که بین هر مبدا و مقصد مسیری وجود دارد که از این متاروترها می گذرد؛ در این مسیر از MPLS استفاده می شود.^۳ با این حال در درون یک زیرشبکه حامل نیز می توان از MPLS بهره گرفت و بدین ترتیب مسیریابهای داخلی نیز برچسب دومی به هر بسته می افزایند و برچسب گذاری سطح دوم پدید می آید. در حقیقت یک بسته می تواند دنباله ای از برچسبهای MPLS را به همراه داشته باشد. بیت S در شکل ۴۱-۵ مسیریاب را آگاه می کند که آیا برچسبهای دیگری هم وجود دارد. در آخرین برچسب، بیت S یک است؛ در حالیکه در بقیه برچسبها بیت S صفر می باشد. در عمل می توان از این قابلیت برای پیاده سازی VPN یا تونلهای بازگشتی

۱. On-Demand Virtual Circuit.

۲. به عبارت بهتر اگر یک مسیر بین دو نقطه را در قالب یک رسمان رنگی مجسم کنیم هر بسته ای که با رنگ همان مسیر دو بار دریافت شود نشان می دهد که بسته در یک حلقه قرار گرفته است وگرنه باید تا رسیدن به مقصد راه خود را ادامه بدهد. -م

۳. یعنی مسیر بین مبدا و مقصد از زیرشبکه های متفاوتی می گذرند که هر یک از این زیرشبکه های حامل یک مسیریاب واحد به نام «متاروتر» فرض شده است. -م

(Recursive Tunnel) بهره گرفت.

اگرچه ایده بنیادی MPLS ساده است ولیکن جزئیات آن بی نهایت پیچیده است و تنوع و بهینه سازیهای گسترده ای دارد، لذا ما بیش از این به موضوع فوق نخواهیم پرداخت. برای آگاهی بیشتر از مراجع ذیل استفاده کنید:

Davie and Rekhter, 2000; Lin et al., 2002; Pepelnjak and Guichard, 2001; Wang, 2001.

۵- بهم بندی شبکه ها (Internetworking)

تا اینجا تلویحاً فرض کرده ایم که تنها یک شبکه واحد و همگن وجود دارد که تمام ماشینهای چنین شبکه ای، در تمام لایه ها از پروتکل مشابهی بهره گرفته اند. متأسفانه چنین فرضی خیلی خوش باورانه است. شبکه ها اعم از LAN، MAN و WAN، انواع بسیار گوناگونی دارند. در هر لایه نیز از پروتکل های متعددی استفاده می شود. در بخشهای آتی مواردی را موشکافی خواهیم کرد که در وصل دو یا چند شبکه و تشکیل «اینترنت»^۱ (internet) با آن مواجه خواهیم بود.

مناقشات گسترده ای پیرامون این سؤال وجود دارد که آیا کثرت بسیار زیاد انواع شبکه ها در حال حاضر، وضعیتی گذرا و موقتی است و به محض آنکه عموم مردم به شگفتی های یک شبکه خاص (مثلاً شبکه مورد نظر شما!) پی برند این کثرت به وحدت می رسد یا آنکه تکثر انواع امری اجتناب ناپذیر و همیشگی در جهان است و باقی خواهد ماند. وجود شبکه های متفاوت مستلزم داشتن پروتکل های متفاوت است.

اعتقاد ما بر آن است که به دلایل ذیل گونه های متفاوتی از شبکه ها (و به تبع آن پروتکل های متفاوت) تا ابد وجود خواهد داشت: اول آنکه شبکه های بسیار متنوعی در محیط های متفاوتی نصب شده اند: تقریباً تمام کامپیوترهای شخصی بر مبنای TCP/IP کار می کنند. بسیاری از مؤسسات تجاری دارای کامپیوترهای بزرگ (Mainframe) با معماری شبکه SNA (متعلق به شرکت IBM) هستند. تعداد قابل توجهی از شرکت های مخابرات تلفنی، شبکه های ATM را به خدمت گرفته اند. در برخی از شبکه های محلی هنوز از پروتکل NCP/IPX (متعلق به شرکت Novell) یا AppleTalk (متعلق به شرکت Apple) بهره می گیرند. و از همه گذشته شبکه های بی سیم با پروتکل های متنوعی در حال ظهور هستند. این رویه برای سالها ادامه خواهد داشت چرا که تکنولوژی های جدید ظهور می کند، اشکالات و ناکارآمدی های گذشته آشکار می شود؛ در عین حال با عرصه تکنولوژی جدید نمی توان یک شبه مشتریان را وادار کرد سیستمی جدید را بپذیرند و سیستم های قدیمی خود را دور بریزند.

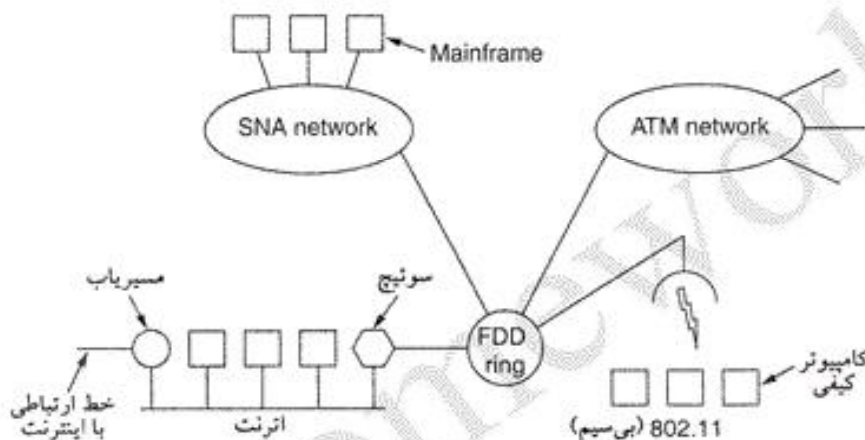
دلیل دوم آنکه کامپیوترها و شبکه ها روز به روز ارزانتر می شوند و تصمیم گیری در خصوص انتخاب و پیاده سازی شبکه ها به رده های پائینی یک سازمان محول می شود. بسیاری از شرکتها این سیاست را پیش گرفته اند که خریدهای بالای یک میلیون دلار باید توسط مدیر ارشد آن شرکت تأیید شود و خریدهای بالای صد هزار دلار توسط مدیران میانی مجاز است در حالی که خریدهای زیر صد هزار دلار توسط مدیران هر واحد بدون اجازه مدیران مافوق انجام می گیرد. این رویه به آنجا منتهی می شود که مثلاً واحد فنی مهندسی در یک سازمان، ایستگاههایی مبتنی بر یونیکس و پروتکل TCP/IP پیاده کند و واحد فروش، کامپیوترهای Mac با پروتکل AppleTalk را به خدمت بگیرد.

دلیل سوم آنکه شبکه های مختلف (مثل ATM و بی سیم) تکنولوژی شدیداً متفاوتی دارند لذا دور از انتظار نیست که وقتی سخت افزار جدیدی خلق می شود نرم افزار جدیدی نیز برای به خدمت گیری آن ایجاد شود. به

^۱ و فنی کلمه internet تماماً با حروف کوچک نوشته می شود به شبکه عظیم و جهانی اینترنت اشاره نمی کند بلکه مراد از آن وصل چند شبکه کوچک و بزرگ و یکپارچه سازی آنهاست؛ مخفف internetwork

عنوان مثال امروزه منازل که در آنها کامپیوتر وجود دارد همانند ادارات در ده سال قبل است: مملو از کامپیوترهایی که ارتباطی با هم ندارند! در آینده ممکن است تلفن، تلویزیون و دستگاههای الکترونیکی خانگی نیز با یکدیگر شبکه شوند و بتوان از راه دور آنها را کنترل کرد. این تکنولوژی جدید بی شک شبکه های جدید و پروتکل های جدیدی را به صحنه خواهد آورد.

به عنوان مثالی از چگونگی اتصال شبکه های متفاوت به یکدیگر به شکل ۵-۴۲ دقت کنید. در این شکل شبکه یکپارچه ای را می بینیم که اجزای آن در چندین موقعیت فیزیکی پراکنده هستند و از طریق یک شبکه گسترده ATM بهم متصل شده اند. در یکی از مکانها یک شبکه فیبرنوری FDDI به عنوان ستون فقرات نصب شده است تا ارتباط یک شبکه اترنت، یک شبکه بی سیم 802.11 و یک شبکه متمرکز SNA را با یکدیگر برقرار کند.



شکل ۵-۴۲. مجموعه ای از شبکه های بهم متصل.

هدف از اتصال تمام این شبکه ها آن است که کاربران هر یک از آنها بتوانند با یکدیگر مبادله اطلاعات داشته باشند یا آنکه هر کاربر بتواند به اطلاعات مورد نظر خود در هر نقطه از شبکه دسترسی پیدا کند. رسیدن بدین هدف متضمن آن است که بسته ها، بین این شبکه ها رد و بدل شوند. از آنجایی که شبکه ها اختلافات بنیانی با یکدیگر دارند، رساندن بسته ها از یک شبکه به شبکه دیگر به نحوی که در ادامه خواهیم دید، چندان هم ساده نیست.

۱-۵-۵ شبکه ها از چه دیدگاهی متفاوتند؟

شبکه ها در موارد مختلفی با یکدیگر تفاوت ذاتی دارند. برخی از این تفاوتها مثل تکنیکهای مدولاسیون یا قالب فریم، مربوط به لایه فیزیکی یا لایه پیوند داده ها است. اینگونه تفاوتها مدنظر ما نیستند. در مقابل، در شکل ۵-۴۳ برخی از تفاوتهایی را که در لایه شبکه بروز می کنند، فهرست نموده ایم. تشریح این تفاوتهاست که نشان می دهد بهم بستن شبکه ها دشوارتر از کار کردن در یک شبکه واحد و همگون است.

وقتی بسته ارسالی از یک مبدا در یکی از شبکه ها، مجبور باشد برای رسیدن به شبکه مقصد از یک یا چند شبکه خارجی عبور کند (که این شبکه ها نیز ممکن است با شبکه مبدا اختلاف بنیادی داشته باشند)، در مرز ارتباطی بین دو شبکه مشکلات عدیده ای رخ می دهد. اولین مورد آن است که وقتی بسته هایی از یک شبکه «اتصال گرا» مجبور به عبور از یک شبکه «بدون اتصال» باشند (که احتمالاً ترتیب بسته ها را بهم می ریزد) این مشکل بروز می کند که فرستنده بسته ها انتظار چنین رخدادی را ندارد و گیرنده نیز کاری نمی تواند انجام بدهد. [چرا که فرض فرستنده و گیرنده آن است که بسته ها به ترتیب ارسال و به ترتیب نیز دریافت می شود. -م]

غالباً بین دو شبکه به تبدیل پروتکل نیاز است و اگر عملکرد مورد نیاز برآورده نشود این تبدیل دشواریهایی را در

مورد اختلاف	برخی از رویکردهای ممکن
نوع سرویس ارائه شده	سرویسهای اتصال گرا در مقابل سرویسهای بدون اتصال
انواع پروتکل	IP, IPX, SNA, ATM, MPLS, AppleTalk, ...
الگوی آدرس دهی	روش مسطح (Flat) مثلا در استانداردهای ۸۰۲ در مقابل روش سلسله مراتبی در IP
چندپخش	در برخی از شبکه ها از آن پشتیبانی می شود و در برخی نمی شود.
اندازه بسته	هر شبکه برای خودش یک سقف حداکثر برای طول بسته تعریف کرده است.
کیفیت خدمات (QoS)	در برخی از شبکه ها از آن پشتیبانی می شود (آنها در رده های متفاوت) و در برخی نمی شود.
مدیریت خطا	تحويل مطمئن و به ترتیب در مقابل تحويل غیر قابل اطمینان و خارج از ترتیب
کنترل جریان	پنجره لغزان، کنترل نرخ ارسال یا حتی بدون مکانیزم کنترل جریان
کنترل ازدحام	اعمال الگوریتم سطل سوراخ، الگوریتم سطل نشانه دار، بسته های دعوت به آرامش و نظائر آن
امنیت	قوانین امنیتی، اعمال روشهای رمزنگاری و نظائر آن
پارامترها	مقادیر مختلف زمان انقضای مهلت تایمرها، پارامترهای متفاوت توصیف جریان و نظائر آن
حسابرسی و دریافت هزینه	بر حسب زمان اتصال، بر حسب تعداد بسته، بابت یا حتی هیچکدام

شکل ۵-۴۳. موارد بی شمار اختلاف شبکه ها.

بی خواهد داشت. همچنین اغلب به تبدیل و نگاشت آدرسها نیاز است که متضمن وجود گونه ای از یک «سیستم فهرست» (Directory System) خواهد بود. از طرفی عبور یک بسته چندپخش (Multicast) از شبکه ای که نمی تواند از مسیریابی چندپخش پشتیبانی کند مستلزم تولید بسته های جداگانه برای یکایک ماشینهای مقصد است.

تفاوت در مقدار حداکثر طول هر بسته داده در شبکه های مختلف، می تواند یک معضل اساسی باشد: چگونه می توان یک بسته ۸۰۰۰ بایتی را از شبکه ای عبور داد که حداکثر طول بسته های آن ۱۵۰۰ بایت است؟ تفاوت در کیفیت خدمات دو شبکه (QoS) نیز موردی است که برای تحويل بسته های بی درنگ از طریق شبکه ای که بی درنگ بودن ارسال را تضمین نمی کند، به یک معضل جدی تبدیل می شود.

کنترل خطا، کنترل جریان و کنترل ازدحام نیز در شبکه های مختلف، تفاوت دارد. اگر مبداء و مقصد، هر دو انتظار داشته باشند که تمام بسته ها به ترتیب و بدون خطا تحويل شوند ولی یک شبکه میانی به محض احساس بروز ازدحام برخی از بسته ها را حذف نماید، بسیاری از برنامه های کاربردی در هم خواهند شکست. همچنین اگر بسته ها مدتی را بی هدف سرگردان شوند و به ناگاه راه خود را پیدا کرده و تحويل داده شوند و گیرنده انتظار چنین رفتاری را نداشته و از عهده دریافت این بسته ها بر نیاید معضلی جدی پدیدار می شود. همچنین مکانیزمهای تضمین امنیت، تنظیم پارامترها، قواعد دریافت هزینه (حسابرسی) و حتی قوانین ملی متفاوت، می تواند منجر به بروز مشکلاتی شود.

۲-۵-۵ چگونگی اتصال شبکه ها به یکدیگر

به گونه ای که در فصل چهارم بررسی کردیم شبکه ها را می توان به کمک ابزارهای متفاوتی به یکدیگر متصل کرد. اجازه بدهید اجمالاً به آن مفاد بپردازیم: در لایه فیزیکی شبکه ها را می توان توسط «تکرارکننده» (Repeater) یا هاب به یکدیگر متصل کرد. این دو ابزار فقط بین آنها را از یک شبکه به شبکه ای از همان نوع منتقل می نمایند. اینها اغلب ابزارهای آنالوگ هستند و هیچ درکی در خصوص پروتکل های دیجیتال [پروتکل های لایه بالاتر] ندارند و فقط سیگنالهای دریافتی را «باز تولید» (Regenerate) می نمایند.

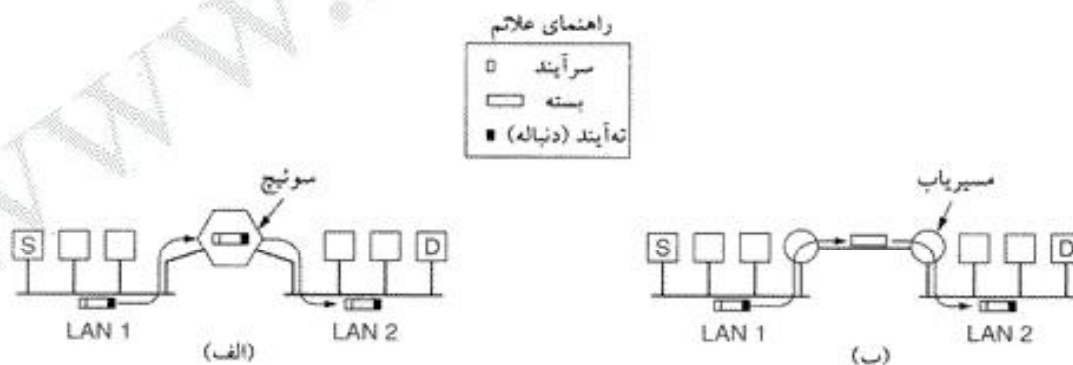
در یک لایه بالاتر به پلها و سوئیچها بر می خوریم که در لایه پیوند داده ها کار می کنند. این دستگاهها فریمها را می پذیرند، آدرسهای MAC را بررسی می کنند و آنها را به شبکه ای دیگر هدایت می نمایند. در ضمن اگر لازم باشد تبدیل پروتکل نیز انجام می دهند مثلاً آترنت را به FDDI یا 802.11 تبدیل می کنند.

در لایه شبکه، مسیریابها را داریم که می توانند دو شبکه را به یکدیگر متصل کنند. اگر دو شبکه، لایه های شبکه متفاوتی داشته باشند، مسیریاب ممکن است بتواند قالب بسته ها را به یکدیگر ترجمه نماید اگرچه امروزه ترجمه و تبدیل بسته ها به یکدیگر، به ندرت انجام می شود. یک مسیریاب را که بتواند با چندین پروتکل مختلف کار کند، اصطلاحاً «مسیریاب چند پروتکلی» (Multiprotocol Router) می نامند.

در لایه انتقال به «دروازه های انتقال» (Transport Gateway) می رسمیم که می توانند واسطه بین دو اتصال در لایه انتقال شوند. به عنوان مثال یک «دروازه انتقال» می تواند این امکان را فراهم کند که جریان بسته ها بین یک شبکه TCP و یک شبکه SNA (که پروتکل لایه انتقال آنها متفاوت است)، مبادله شود. این دروازه، یک «اتصال TCP» (TCP connection) را به یک «اتصال SNA» می چسباند.

در آخر به لایه کاربرد و «دروازه کاربرد» (Application Gateway) می رسمیم که این دروازه محتوای پیامها را بهم ترجمه می نماید. به عنوان مثال یک دروازه بین سیستم پست الکترونیکی در اینترنت (مثل سیستم RFC822) و سیستم پست الکترونیکی X.400، باید بتواند محتوای پیام نامه های الکترونیکی را تجزیه و تحلیل کرده و فیلدهای مختلف سرآیند آنها را به یکدیگر تبدیل نماید.

در این فصل به موضوع بهم بندی شبکه ها در لایه شبکه خواهیم پرداخت. برای آنکه ببینید هدایت اطلاعات در لایه پیوند داده ها چه تفاوتی با هدایت در لایه شبکه دارد، شکل ۵-۴۴ را در نظر بگیرید. در شکل ۵-۴۴ الف ماشین مبدا یعنی S می خواهد بسته ای را برای ماشین مقصد D بفرستد. این دو ماشین بر روی دو شبکه اتترنت جدا که از طریق سوئیچ بهم متصل شده اند، واقع هستند. ماشین S بسته ای را در درون یک فریم جاسازی کرده و آن را به خروجی می فرستد. این فریم به سوئیچ رسیده و با بررسی آدرس MAC مشخص می شود که باید به LAN2 برود. سوئیچ، این فریم را از LAN1 برداشته و به LAN2 روانه می کند.



شکل ۵-۴۴. (الف) دو شبکه اتترنت که از طریق سوئیچ بهم متصل شده اند. (ب) دو شبکه اتترنت که از طریق مسیریاب بهم متصل شده اند.

حال همین وضعیت را در شرایطی در نظر بگیرید که این دو شبکه اتترنت به جای سوئیچ از طریق یک جفت مسیریاب به یکدیگر متصل شده اند. ارتباط بین مسیریابها از طریق یک خط نقطه به نقطه برقرار شده است؛ این خط می تواند یک خط استیجاری با هزاران کیلومتر طول باشد. در اینجا فریم توسط مسیریاب اول دریافت شده و بسته جاسازی شده در درون آن، از فیلد داده فریم استخراج می شود. مسیریاب، آدرس درون بسته را (مثلاً آدرس IP

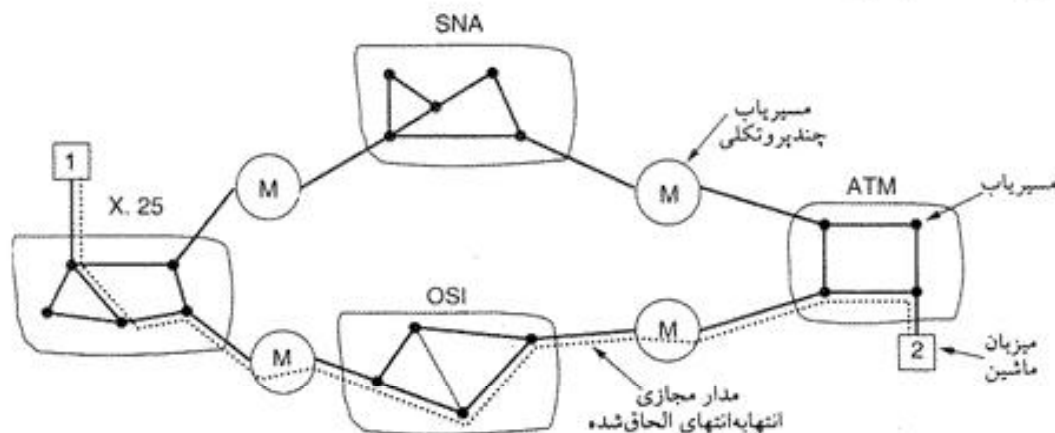
را) بررسی کرده و آنرا در درون جدول مسیریابی خود جستجو می نماید؛ سپس براساس این آدرس به نتیجه می رسد که باید بسته را به مسیریاب راه دور بفرستد و احتمالاً برای این کار مجبور خواهد شد که مبتنی بر پروتکل این خط، آنرا در فریم جدید و متفاوتی جاسازی نماید. در مسیریاب مقابل این بسته مجدداً درون فیلد داده از یک فریم اترنت جاسازی شده و بر روی LAN2 ارسال می شود.

تفاوت بنیادی بین حالتی که سوئیچ (یا پل) در میان است با وقتی که مسیریاب اتصال شبکه را برقرار کرده، آنست که در سوئیچ یا پل کل یک فریم براساس آدرس MAC آن هدایت و منتقل می شود در حالیکه در یک مسیریاب، یک بسته از درون فریم استخراج شده و سپس از آدرس درون بسته برای تصمیم گیری در خصوص محل ارسال آن استفاده می شود. سوئیچها مجبور نیستند که برای هدایت بسته ها درکی از پروتکل لایه شبکه داشته باشند در حالیکه مسیریابها اینگونه اند.

۳-۵-۵ مدارات مجازی الحاق شده (Concatenated Virtual Circuit)

دو روش برای بهم بندی شبکه ها ممکن است: روش الحاق زیر شبکه های مدار مجازی به صورت اتصال گرا و روش الحاق دیتاگرام. به نوبت این دو روش را بررسی خواهیم کرد ولی در ابتدا به ذکر نکته ای می پردازیم. در گذشته اکثر شبکه های عمومی اتصال گرا بودند (و شبکه هایی مثل SNA, Frame Relay, و ATM هنوز هم اینگونه اند). با رشد و مقبولیت سریع اینترنت، شبکه دیتاگرام مد روز شد ولیکن این تصور که شبکه های دیتاگرام ابدی هستند، اشتباه است. در دنیای شبکه ها تنها چیزی که جاوید و ابدی می ماند «تغییر» است. با رشد شبکه های چند رسانه ای، احتمالاً اتصال گرایی با یکی از اشکال خود مجدداً به صحنه برمی گردد چراکه در شبکه های اتصال گرا راحتتر می توان کیفیت خدمات را تضمین نمود. لذا در ابتدا شبکه های اتصال گرا را مورد بحث و بررسی قرار می دهیم.

در مدل الحاق شبکه های مدار مجازی که در شکل ۴۵-۵ نشان داده شده، ایجاد «اتصال» با یک ماشین در شبکه ای دور دست، بروشی مشابه با روش معمولی انجام می گیرد: زیر شبکه می بیند که مقصد در شبکه ای دیگر واقع شده لذا یک مدار مجازی با آن مسیریاب که به شبکه مقصد نزدیکتر است، ایجاد می کند. از آن مسیریاب نیز یک مدار مجازی با یک «دروازه خارجی» ایجاد می شود. (دروازه خارجی یا External Router، یک مسیریاب چند پروتکلی است). آن «دروازه» نیز مشخصات این مدار مجازی را در جدول خود درج کرده و کار را با ایجاد یک مدار مجازی جدید با مسیریاب زیر شبکه دیگر ادامه می دهد. این فرآیند ادامه می یابد تا آنکه مدار مجازی به ماشین مقصد ختم شود.



شکل ۴۵-۵. بهم بندی شبکه ها: کمک مدارات مجازی الحاق شده.

هرگاه یک بسته داده، در مسیری جریان یابد، هر یک از دروازه‌های میانی این بسته را رله می‌کنند و در صورت نیاز قالب بسته را تبدیل نموده و شماره‌های مدار مجازی را تغییر می‌دهند. بدیهی است که تمام بسته‌های داده باید به یک ترتیب از دروازه‌ها بگذرند؛ نتیجتاً ترتیب بسته‌های متعلق به یک جریان هرگز به هم نخواهد ریخت. ویژگی بنیادی این راهکار آنست که دنباله‌ای از مدارات مجازی بین ماشین مبدا و مقصد (از طریق دروازه‌های میانی) ایجاد و تنظیم می‌شود. هر دروازه جدولی را در خود نگاهداری می‌کند که این جدول فهرست مدارات مجازی را که از آن دروازه می‌گذرند و همچنین طریقه مسیریابی و شماره‌های جدید مدار مجازی را تعیین کرده است.

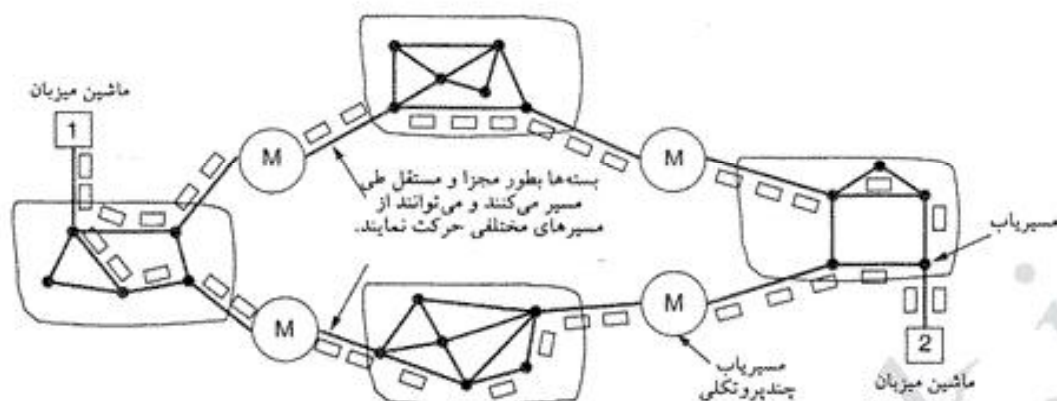
این ساختار زمانی به بهترین نحو کار می‌کند که تمام شبکه‌های میانی از ویژگیهای مشابهی برخوردار باشند. به عنوان مثال اگر تمام آنها تحویل مطمئن بسته‌های لایه شبکه را تضمین کرده باشند، آنگاه جریان بین مبدا و مقصد، مطمئن و قابل اعتماد خواهد بود (به استثنای وقتی که یکی از مسیرهای میانی از کار بیفتند). به دلیل مشابه اگر هیچیک از شبکه‌های میانی تحویل مطمئن بسته‌ها را تضمین نکرده باشند، الحاق مدارات مجازی نیز نامطمئن خواهد بود. ولیکن اگر ماشین مبدا بر روی شبکه‌ای باشد که تحویل مطمئن بسته‌ها را تضمین کرده در حالی که یکی از شبکه‌های میانی استعداد از بین بردن بسته‌ای را داشته باشد، الحاق مدارات مجازی منجر به نامطمئن شدن کل مدار مجازی شده و طبیعت خدمات پیش‌بینی شده مثل تحویل مطمئن و حفظ ترتیب بسته‌ها تغییر می‌کند. الحاق مدارات مجازی در لایه انتقال نیز رایج است. بویژه این امکان وجود دارد که یک «خط انتقال بیت» (Bit Pipe) بین شبکه‌ای مثل SNA که به یک دروازه منتهی می‌شود و اتصالی TCP با دروازه دیگر دارد، ایجاد نمود. بدین ترتیب می‌توان یک مدار مجازی «انتها به انتها» (End To End) ایجاد کرد در حالی که چندین شبکه با پروتکل‌های مختلف در میانه راه قرار گرفته‌اند.

۵-۵-۵ بهم‌بندی شبکه‌های بدون اتصال (Connectionless Internetworking)

مدلی دیگر از بهم‌بندی شبکه‌ها، مدل دیتاگرام است؛ (به شکل ۵-۴۶ دقت کنید). در این مدل تنها خدمتی که لایه شبکه به لایه انتقال ارائه می‌دهد آن است که بسته‌های دیتاگرام را بر روی زیرشبکه توزیع کند؛ زیرشبکه نیز حداکثر تلاش خود را در جهت تحویل آن به عمل می‌آورد. در این مدل هیچگونه نشانی از مدار مجازی در سطح لایه شبکه وجود ندارد و فقط شبکه‌ها به هم متصل و ملحق می‌شوند. در این مدل نیازی نیست که بسته‌های متعلق به یک اتصال [تولید شده توسط یک ماشین] به ترتیب از دروازه‌های یکسانی بگذرند. در شکل ۵-۴۶ دیتاگرام‌های ارسالی توسط ماشین ۱ که به سوی ماشین ۲ روانه شده‌اند از مسیرهای متفاوتی در شبکه عبور کرده‌اند. تصمیم‌گیری در خصوص مسیر هر بسته، بطور جداگانه انجام می‌شود و این تصمیم‌گیری بستگی به ترافیک لحظه‌ای ارسال بسته دارد. در این استراتژی از چندین مسیر بهره گرفته می‌شود و طبعاً پهنای باند بیشتری در مقایسه با مدل الحاق شبکه‌های مدار مجازی حاصل خواهد شد. ولی در مقابل تضمینی در به ترتیب رسیدن بسته‌ها به مقصد وجود ندارد، بلکه فقط فرض بر تحویل بسته‌هاست نه حفظ ترتیب آنها.

مدل شکل ۵-۴۶ به همین سادگی که بنظر می‌رسد نیست. اولین مورد اشکال آنکه، اگر هر یک از شبکه‌های میانی، پروتکل لایه شبکه خاص خودشان را داشته باشد، انتقال بسته از یک شبکه به شبکه‌ای دیگر ممکن نخواهد بود. شاید کسی تصور کند که یک مسیر یاب چند پروتکلی قادر به ترجمه قالب بسته‌ها به یکدیگر است در حالی که این تصور زمانی درست است که قالب بسته‌ها نزدیک به یکدیگر بوده و فیلدهای اطلاعاتی هر بسته مشابه باشند و در غیر این صورت چنین تبدیلی ناقص بوده و محکوم به شکست است. به همین دلیل به ندرت تلاش می‌شود چنین تبدیلی انجام گیرد.

دومین مشکل جدی، مسئله آدرس دهی است. یک حالت ساده را مدنظر قرار بدهید: یک ماشین بر روی شبکه



شکل ۵-۴۶. بهم پندگی شبکه های بدون اتصال.

اینترنت تلاش می کند یک بسته IP برای ماشینی بر روی یک شبکه SNA (متصل به اینترنت) بفرستد. آدرسهای IP و SNA متفاوت از هم هستند. آدرسهای IP و SNA باید در هر دو جهت به یکدیگر نگاشته و ترجمه شوند. مضاف بر این، در شبکه های متفاوت مفهوم «چیزهایی که قابل آدرس دهی هستند» فرق دارد. در IP ماشینهای میزبان (یا در حقیقت کارتهای واسط شبکه) دارای آدرس هستند. در SNA هر «موجودیت» (Entity) مثل هر ابزار سخت افزاری می تواند آدرس داشته باشد. در بهترین حالت هر دروازه باید دارای یک پایگاه اطلاعاتی بوده و بتواند آدرسها را بهم بنگارد (تبدیل کند) ولی همین کار منشاء بروز مشکلات جدی است.

یک نظریه دیگر آن است که یک بسته جهانی و استاندارد طراحی شود و تمام مسیریابها آنرا به رسمیت بشناسند. این راهکار در حقیقت همین IP است که بسته های آن به گونه ای طراحی شده که هر شبکه ای قادر به حمل و هدایت آنهاست. البته ممکن است به نظر برسد IPv4 (پروتکل فعلی اینترنت) نهایتاً تمام ساختارها و پروتکل های دیگر را از صحنه خارج می کند و IPv6 (پروتکل آینده اینترنت) نیز راه به جایی نمی برد و هیچ پروتکل جدیدی ابداع نمی شود! ولی تاریخ نشان داده که اینگونه نیست. جلب موافقت عموم افراد برای پذیرش یک قالب واحد بسیار دشوار است چرا که شرکت های مختلف مصالح و سود خود را در آن می بینند که قالب و ساختار اختصاصی و تحت کنترل خود را داشته باشند.

حال بایبید به اختصار مروری بر دو روش شبکه بندی داشته باشیم. مدل الحاق شبکه به روش مدار مجازی همان مزایایی را دارد که مدار مجازی در یک زیر شبکه واحد خواهد داشت: یعنی پیشاپیش می توان بافرها را از قبل رزرو کرد، ترتیب بسته ها حفظ می شود، سرآیند کوتاهتری برای بسته ها نیاز است و از مشکلاتی که ناشی از تکراری شدن بسته هایی که با تأخیر می رسند، احتراز می شود.

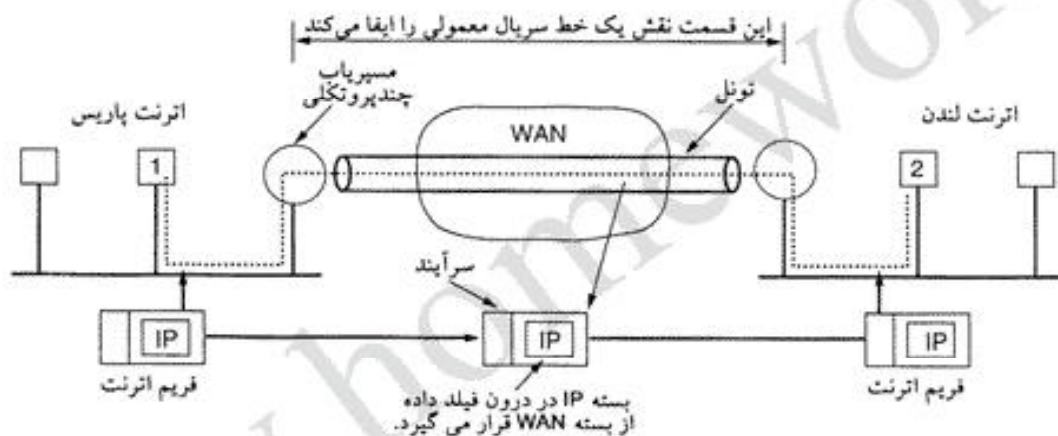
این روش معایبی نیز دارد: در مسیریابها به فضای قابل توجهی برای نگهداری جدول اتصالات باز نیاز است، برای احتراز از مناطق مواجه با ازدحام مکانیزم مسیریابی و مسیرها تغییر نمی کند و مسیرها نسبت به خرابی مسیریابها بسیار آسیب پذیر هستند. همچنین این اشکال بزرگ وجود دارد که اتصال چنین شبکه ای به یک شبکه نامطمئن دیتاگرام اگر غیر ممکن نباشد بسیار دشوار است.

ویژگی بهم پندگی شبکه ها به روش دیتاگرام، دقیقاً مشابه با ویژگی زیر شبکه های دیتاگرام است: استعداد بروز ازدحام دارد ولی در عوض قابلیت بیشتری نیز برای رفع آن دارد، در مواجهه با خرابی یک مسیریاب قابلیت تحمل بیشتری از خود نشان می دهد و به سرآیند طولانی تری برای هر بسته نیاز است؛ استفاده از انواع الگوریتم های وفقی و پویای مسیریابی میسر است و در مجموع تمام مزایا و معایب یک زیر شبکه واحد و منفرد را بطور مشابه دارد.

بزرگترین مزیت روش دیتاگرام برای بهم پندی شبکه آن است که می توان این روش را برای وصل هر شبکه ای که در درون از مدار مجازی بهره گرفته، استفاده کرد. بسیاری از شبکه های LAN، شبکه های متحرک (Mobile) مثل شبکه ناوگانهای هوایی و دریایی و حتی برخی از شبکه های WAN، در رده شبکه های دیتاگرام قرار می گیرند و هر گاه استراتژی بهم پندی این شبکه ها مبتنی بر مدارات مجازی باشد، مشکلات جدی بروز خواهد کرد.

۵-۵-۵ ایجاد تونل (Tunneling)

اتصال دو شبکه مختلف در حالت کلی بسیار سخت است ولیکن یک حالت خاص و رایج وجود دارد که راهگشا و قابل مدیریت است. این حالت زمانی اتفاق می افتد که ماشینهای مبدا و مقصد بر روی شبکه هایی از یک نوع هستند ولی یک شبکه متفاوت در میان آنها قرار گرفته است. به عنوان مثال یک بانک بین المللی را مدنظر قرار دهید که یک شبکه اترنت مبتنی بر پروتکل TCP/IP در پاریس و همچنین یک اترنت با پروتکل TCP/IP در لندن دارد ولی به نحوی که در شکل ۵-۴۷ نشان داده شده، یک شبکه بغیر از IP (مثل ATM) در میان آنها قرار گرفته است.



شکل ۵-۴۷. ارسال یک بسته از طریق ایجاد تونل بین پاریس و لندن.

راهکار حل این مشکل تکنیکی به نام «ایجاد تونل» است. برای ارسال یک بسته IP به ماشین ۲، ماشین ۱، بسته ای حاوی آدرس IP ماشین ۲ ساخته و آنرا در درون فریم شبکه اترنت قرار داده و آدرس مسیریاب چند پروتکلی واقع در پاریس را در فیلد آدرس این فریم درج کرده و آن را بر روی اترنت قرار می دهد. هرگاه این «مسیریاب چند پروتکلی» این فریم را دریافت کند بسته IP را از درون فریم استخراج کرده و آن را در فیلد حمل داده (Payload) از بسته لایه شبکه WAN قرار داده و آدرس بسته جدید را آدرس مسیریاب چند پروتکلی واقع در لندن قرار می دهد. [به عبارتی دو بسته تودرتو تشکیل می شود که آدرس بسته درونی، آدرس ماشین مقصد و آدرس بسته بیرونی آدرس مسیریاب لندن است.] هرگاه این بسته به مسیریاب واقع در لندن برسد آن مسیریاب بسته IP اصلی را از درون آن استخراج کرده و آنرا با قرار دادن در یک فریم اترنت برای ماشین ۲ می فرستد.

شبکه WAN را می توان یک تونل بزرگ در نظر گرفت که از یک مسیریاب چند پروتکلی شروع و به مسیریابی دیگر از همین نوع، در طرف دیگر WAN ختم می شود. یک بسته IP در حالی که درون یک بسته دیگر جا گرفته از یک طرف تونل شروع به طی مسیر به طرف دیگر تونل می کند و جای هیچگونه نگرانی در خصوص ساختار WAN وجود ندارد [چرا که تنها کاری که WAN در این میان انجام می دهد آنست که مبتنی بر پروتکل فعلی خود، بسته IP را به عنوان یک قطعه داده خام، در یک نقطه دریافت و در نقطه دیگر تحویل می دهد. -م] در این مثال در خصوص ماشینهای هر یک از شبکه های اترنت نیز نگرانی وجود ندارد [چون هر دو از یک نوع و مبتنی بر پروتکل

مشابه هستند. فقط «مسیریاب چندپروتکلی» است که باید هم بسته های IP و هم بسته های WAN را بفهمد و مدیریت کند. در حقیقت کل شبکه ای که در میان این دو مسیریاب چندپروتکلی قرار گرفته نقشی شبیه به یک خط سریال ایفاء می کند.

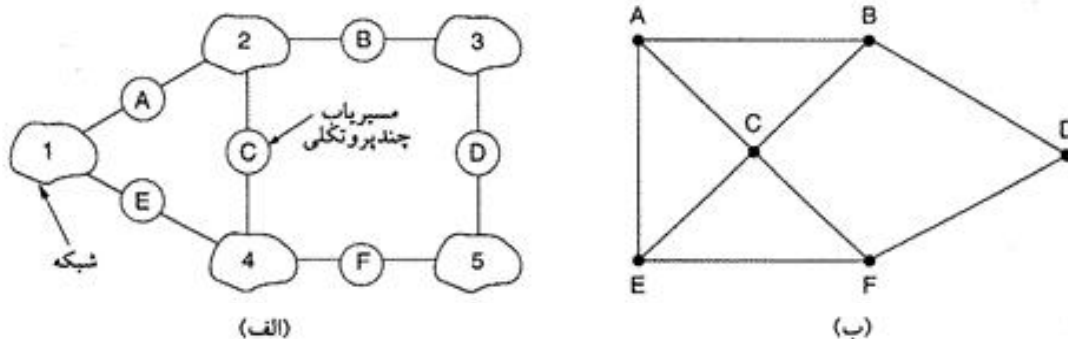
شاید یک تمثیل بتواند مفهوم تونل را روشنتر کند. شخصی را در نظر بگیرید که با خودروی خودش از پاریس به لندن مسافرت می کند. در فرانسه او با خودروی خود تا کنار کانال انگلیس رانندگی می کند ولی با رسیدن به این کانال (که رانندگی در آن ممنوع است)، خودروی او در یک قطار سریع السیر بار شده و از طریق قطار به انگلیس منتقل می شود. در حقیقت، به نحوی که در شکل ۵-۴۸ نشان داده شده، این خودرو همانند یک محموله عادی جابجا می شود. در طرف دیگر، این خودرو مجدداً بر زمین گذاشته شده و اجازه می یابد با نیروی محرکه خودش در جاده های انگلیس حرکت نماید. ایجاد تونل برای بسته ها از طریق یک شبکه خارجی مشابه با همین عملکرد است.



شکل ۵-۴۸. تونل کردن (Tunneling) یک خودرو از فرانسه به انگلستان.

۶-۵-۵ مسیریابی بین شبکه های بهم متصل

مسیریابی بین چند شبکه متصل بهم مشابه با مسیریابی در یک زیر شبکه واحد است (البته با پیچیدگی های بیشتر). به عنوان مثال ساختار ارتباطی شبکه الف-۴۹ را در نظر بگیرید که در آن پنج شبکه توسط شش مسیریاب (احتمالاً از نوع چندپروتکلی) بهم متصل شده اند. تشکیل مدل گراف از این وضعیت پیچیده است چراکه در گراف، هر مسیریاب باید مستقیماً با مسیریاب دیگر در ارتباط باشد (بسته بفرستد) در حالیکه اینجا مسیریابها مستقیماً به یک شبکه میانی متصلند. مثلاً مسیریاب B در شکل ۵-۴۹-الف می تواند از طریق شبکه ۲ به مسیریابهای A و C دسترسی داشته باشد، همچنین مسیریاب D از طریق شبکه ۳ به B دسترسی دارد. اگر هر یک از شبکه ها را به مثابه یک خط سریال فرض نماییم، گراف شکل ۵-۴۹-ب بدست می آید.



شکل ۵-۴۹. (الف) شبکه های از شبکه ها (ب) گراف متناظر.

به محض آنکه گراف شبکه تشکیل شد می توان یکی از الگوریتمهای شناخته شده مسیریابی مثل «الگوریتم بردار فاصله» یا «الگوریتم حالت لینک» (Link State Algorithm) را بر روی مجموعه این مسیریابهای چندپروتکلی اعمال کرد. در نتیجه یک الگوریتم مسیریابی دوسطحی ایجاد می شود: در درون هر شبکه از یک «پروتکل مسیریابی درونی» (Interior Gateway Protocol) استفاده می شود در حالی که بین شبکه ها از «پروتکل مسیریابی بیرونی» (Exterior Gateway Protocol) بهره گرفته می شود. (اصطلاح Gateway نام قدیمی مسیریاب است.) در حقیقت، چون شبکه ها مستقل هستند ممکن است از الگوریتمهای مسیریابی مختلفی در درون شبکه بهره گرفته باشند. از آنجایی که باهم پندی شبکه ها هر شبکه استقلال داخلی خود را حفظ می کند، به هر یک از این شبکه های مستقل «سیستم خودمختار» یا به اختصار AS گفته می شود.

در چنین شبکه ای، یک بسته نوعی از یک ماشین بر روی LAN مسیر خود را آغاز کرده و به آدرس «مسیریاب چندپروتکلی» متصل به آن LAN ارسال می شود (با قرار دادن آدرس آن مسیریاب در فیلد آدرس از فریم MAC). پس از آن که بسته به مسیریاب رسید، نرم افزار آن مسیریاب به کمک جدول مسیریابی خود مشخص می کند که این بسته باید به سوی کدامیک از مسیریابهای دیگر هدایت شود. اگر بسته را بتوان با همان پروتکل لایه شبکه (که بسته براساس آن تولید شده) به مقصد رساند، این بسته مستقیماً هدایت و ارسال می شود. در غیر این صورت از مکانیزم ایجاد تونل (Tunneling) استفاده شده و کل بسته درون یک بسته متناسب با پروتکل مسیریابهای میانی جاسازی و ارسال می گردد. این فرآیند آنقدر تکرار می شود تا بسته به شبکه مقصد برسد.

یکی از تفاوت های بین مسیریابی درون یک شبکه خودمختار (Intranetwork Routing) و مسیریابی بین چند شبکه خودمختار آن است که در مورد دوم ممکن است بسته ها نیاز به ترد از مرزهای بین المللی کشورها داشته باشند. در چنین حالتی، ملاحظات قانونی متعدد مطرح خواهد شد چرا که مثلاً طبق قوانین کشور سوئد، خارج کردن اطلاعات فردی شهروندان سوئدی از کشور مجاز نیست. یا مثلاً طبق قوانین کشور کانادا، ترافیک داده هایی که مبداء آنها در کشور کانادا و مقصد آنها نیز در کانادا است نباید از کشور خارج شود. طبق این قانون ترافیک داده هایی که مبداء آن شهر «وینزور» یا «انتاریو» و مقصد آن «ونکوور» است نمی تواند از طریق مسیر نزدیکتری که از «دیترویت» می گذرد، مسیریابی و هدایت شود حتی اگر استفاده از این مسیر سریعتر و ارزانتر باشد.

تفاوت دیگر بین مسیریابی درونی و مسیریابی بیرونی (Interior/Exterior Routing) موضوع هزینه (قیمت) است. در یک شبکه خودمختار و واحد، بطور معمول از یک الگوریتم مشخص برای تعیین هزینه استفاده می شود ولیکن در شبکه های مختلف که مدیریت متفاوتی دارند ممکن است یک مسیر از لحاظ قیمت از مسیر دیگر مقرون به صرفه تر باشد. همینطور سطح کیفیت خدمات عرضه شده در شبکه های متفاوت فرق می کند و همین مورد ممکن است دلیل انتخاب یک مسیر از بین دیگر مسیرها باشد.

۷-۵-۵ قطعه سازی بسته ها (Fragmentation)

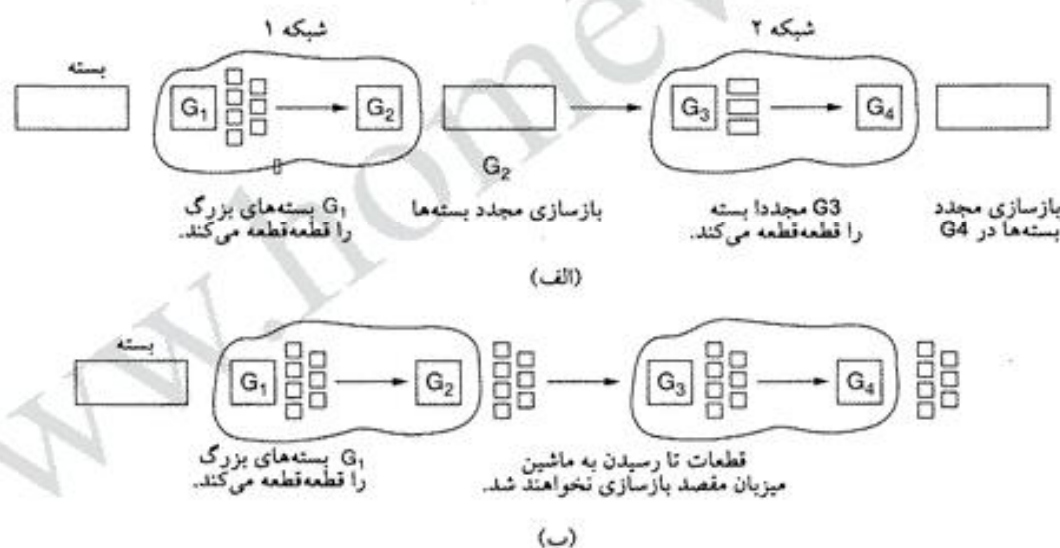
در هر شبکه اندازه هر بسته نمی تواند از یک حد مجاز بیشتر باشد. این محدودیت، علل مختلفی دارد که از این میان می توان به موارد ذیل اشاره کرد:

۱. سخت افزار (مثلاً طول فریم اترنت نمی تواند از حدود ۱۵۰۰ بایت بیشتر باشد).
۲. سیستم عامل (مثلاً بافرها ۵۱۲ بایتی هستند).
۳. پروتکل (مثلاً تعداد بیت های فیلدی که میزان داده های موجود در هر بسته را مشخص می کند کم است).
۴. سازگاری با برخی از استانداردهای بین المللی
۵. تمایل به کاهش حجم ارسال مجدد داده هایی که در اثر خطای انتقال از بین می روند
۶. اجتناب از اشغال بیش از حد کانال توسط یک بسته بزرگ

نتیجه این عوامل آن می شود که دست طراحان شبکه در انتخاب حداکثر طول بسته ها باز نیست. حداکثر طول داده هایی که می تواند درون یک بسته قرار بگیرد از ۴۸ بایت (در سلولهای ATM) تا ۶۵۵۱۵ بایت (در بسته های IP) متغیر است اگرچه طول داده ها در لایه های بالاتر، می تواند از این هم بیشتر باشد.

یکی از مشکلات بدیهی، زمانی رخ می دهد که یک بسته بزرگ بخواهد به شبکه ای وارد شود که طول حداکثر بسته آن کوچک است. یک راه حل آن است که از همان ابتدا این اطمینان حاصل شود که چنین مشکلی پیش نمی آید. به عبارت دیگر در اتصال شبکه ها، از یک الگوریتم مسیریابی استفاده شود که از ارسال بسته ها به شبکه ای که از عهده پذیرش آن بر نمی آید اجتناب کند، ولیکن این راه حل همیشه عملی نیست: اگر همه بسته های تولید شده توسط مبدا، دارای اندازه ای باشند که شبکه مقصد از عهده دریافت آن بر نمی آید چه اتفاقی می افتد؟ الگوریتم مسیریابی به ندرت می تواند چنین مقصدی را نادیده گرفته و آنرا کنار بگذارد.

اصولاً در چنین مواقعی می توان برای حل مشکل، به «دروازه» (Gateway) اجازه داد که بسته ها را به چند «قطعه» (Fragment) شکسته و هر یک از این قطعات را در پوشش یک بسته مستقل ارسال کند؛ ولیکن والدین کودکان خردسال به خوبی می دانند که تکه تکه کردن اشیاء بزرگ به قطعات کوچک ساده تر از سرهم آنهاست!!! (فیزیکدانان برای این پدیده نامی انتخاب کرده اند: قانون دوم ترمودینامیک) شبکه های سوئیچ بسته نیز برای بازسازی قطعات به بسته اصلی با مشکل روبرو هستند.



شکل ۵-۵. (الف) قطعه قطعه کردن نامرئی (شفاف) (ب) قطعه قطعه کردن غیر شفاف (مرئی).

برای بازسازی قطعات و تشکیل بسته اصلی، دو استراتژی متضاد قابل اعمال است: استراتژی اول آن است که قطعه قطعه سازی بسته ها در شبکه ای که بسته های کوچک دارد به گونه انجام شود که این عمل از دید مقصد نهایی بسته، پنهان بماند. در این راهکار که در شکل ۵-۵-الف نشان داده شده است، در هر شبکه با بسته کوچک یک «دروازه» (یا به عبارتی یک مسیر یاب خاص) وجود دارد که واسطه دیگر شبکه هاست. وقتی بسته ای با طول بیش از حد به این دروازه می رسد، آن را قطعه قطعه می کند؛ سپس هر یک از قطعات به آدرس دروازه خروجی در مقصد ارسال شده و در آنجا قطعات به شکل اصلی بازسازی می شوند. بدین ترتیب عبور بسته های بزرگ از شبکه ای با بسته کوچک، از دید شبکه های بیرونی پنهان می ماند؛ به عبارتی شبکه های بیرونی از قطعه قطعه شدن بسته ها آگاه نمی شوند. به عنوان مثال، شبکه ATM دارای سخت افزار خاصی است که به صورت نامرئی، بسته ها را به سلولهای

۵۳ پاشی شکسته و در طرف مقابل آنها را به بسته اصلی بازسازی می نماید. در شبکه ATM به عمل شکستن بسته ها اصطلاحاً «قطعه بندی» (Segmentation) گفته می شود و مفهومی معادل با آنچه که در بالا بدان اشاره شد دارد، بلکه فقط جزئیات پیاده سازی آن متفاوت است.

استراتژی قطعه قطعه کردن بسته به صورت نامرئی، اگرچه ساده و سرراست است ولیکن مشکلاتی را در بر دارد. اولین مورد آن است که دروازه خروجی باید تشخیص بدهد که آیا تمام قطعات یک بسته را دریافت کرده است یا آنکه هنوز قطعاتی در راه هستند و به همین دلیل باید یک بیت خاص پایان دنباله قطعات هر بسته را مشخص نماید. مورد دیگر آن است که تمام قطعات یک بسته بایستی از یک دروازه مشابه و یکسان خارج شوند. وقتی اجازه نمی دهیم که قطعات یک بسته برای رسیدن به مقصد نهایی خود از مسیرهای مختلفی حرکت کنند، بخشی از کارایی و بهینگی را از دست خواهیم داد. آخرین اشکال آنست که وقتی یک بسته بزرگ مجبور است مکرراً از شبکه هایی بگذرد که آن را قطعه قطعه و بازسازی می کنند، سربار نسبتاً زیادی تحمیل خواهد شد. به هر تقدیر شبکه ای مثل ATM به قطعه قطعه سازی نامرئی [استراتژی فوق الذکر] نیاز دارد.

استراتژی دیگر برای قطعه قطعه کردن بسته ها آنست که از بازسازی بسته ها در دروازه های میانی اجتناب شود: هرگاه بسته ای قطعه قطعه شد، با هر یک از قطعات به مثابه یک بسته واقعی و اصلی رفتار شود. در این استراتژی به نحوی که در شکل ۵-۵۰-ب نشان داده شده تمام قطعات به همان نحو از دروازه خروجی شبکه گذر می کنند. بازسازی بسته ها فقط در ماشین مقصد انجام می شود. IP به همین نحو عمل می کند.^۱

قطعه قطعه سازی غیرشفاف نیز با اشکالاتی مواجه است. به عنوان مثال، هر ماشین باید قادر به بازسازی بسته باشد. مشکل دیگر آنست که وقتی یک بسته بزرگ به قطعات کوچک تقسیم می شود، سربار داده ها افزایش می یابد زیرا هر قطعه نیاز به سرآیند مستقل دارد، در حالی که در استراتژی اول به محض خروج قطعات از شبکه ای با بسته کوچک، سربار تحمیل شده حذف می شود ولیکن در این روش سربار ناشی از سرآیند اضافی، تا خاتمه مسیر باقی خواهد ماند. مزیت روش قطعه قطعه سازی غیرشفاف آن است که قطعات هر بسته می توانند از مسیری مجزا و دروازه های خروجی متفاوت به سوی مقصد نهایی خود هدایت شوند و طبعاً کارایی بالاتر و بهینه تری حاصل خواهد شد. البته اگر شبکه های مختلف طبق مدل مدار مجازی به یکدیگر متصل و ملحق شده باشند [بخش ۵-۵-۳] این مزیت بکار نخواهد آمد.

هرگاه بسته ای قطعه قطعه شود، قطعات آن باید به نحوی شماره گذاری شوند که بتوان ترتیب اصلی داده ها را بازیابی کرد. یکی از روشهای شماره گذاری، روش درختی است: اگر بسته شماره ۰ نیاز به قطعه شدن داشته باشد قطعات به صورت ۰.۰، ۰.۱، ۰.۲ و به همین ترتیب، شماره گذاری می شوند. اگر هر یک از این قطعات، باز هم نیاز به شکسته شدن داشته باشند ترتیب شماره ها (از چپ به راست) به صورت ۰.۰، ۰.۱، ۰.۲، ۰.۳، ...، ۰.۱۰، ۰.۱۱، ۰.۱۲، ۰.۱۳، ...، ۰.۱۰۰، ۰.۱۰۱، ۰.۱۰۲، ... خواهد بود. اگر برای این منظور به تعداد کافی فیلد در سرآیند بسته پیش بینی شده باشد، الگوی فوق این اطمینان را می دهد که بتوان بسته ها را در مقصد بازسازی کرد. (حتی اگر قطعات به ترتیب دریافت نشوند.)

ولی اگر در یکی از شبکه های بین راه قطعه ای گم شود یا حذف گردد نیاز است که مجدداً کل بسته از مبدا آن ارسال و از نو قطعه قطعه و ارسال شود. فرض کنید که یک بسته ۱۰۲۴ بیتی در ابتدا به چهار قطعه مساوی با شماره های ۰.۰، ۰.۱، ۰.۲ و ۰.۳ تقسیم شده باشد. اگر قطعه ۰.۱ از دست برود ولی بقیه قطعات سالم به مقصد

۱. بدین استراتژی، قطعه قطعه سازی غیرشفاف و مرئی (Nontransparent Fragmentation) گفته می شود. هرگاه بسته ای

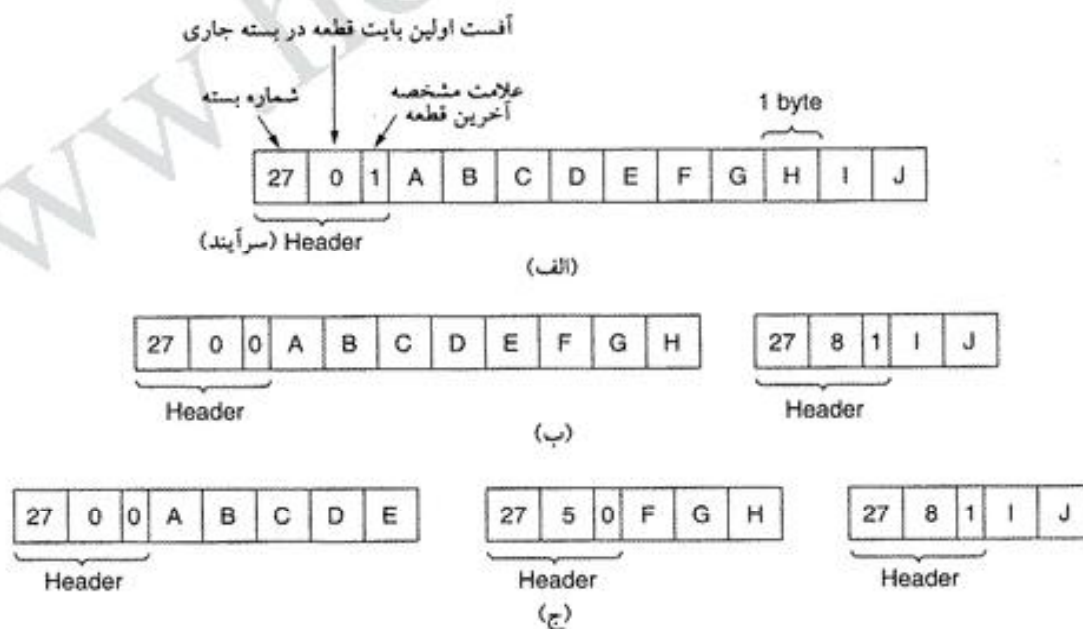
در هر نقطه از مسیر قطعه قطعه شد هیچکسی آنها را بازسازی نخواهد کرد مگر ماشین مقصد نهایی. -م

برسند، پس از مدتی مهلت فرستنده بسته به سر آمده و کل بسته را از نو ارسال می‌کند. حال فرض کنید که به ناگاه محدودیت طول بسته از ۲۵۶ بیت به ۵۱۲ بیت افزایش پیدا کرده و در اینجا بسته به جای آنکه ۴ قسمت شود به دو قطعه تقسیم گردد. حال وقتی بسته شماره ۰.۱ به مقصد می‌رسد گیرنده تصور می‌کند که تمام چهار قطعه‌ای که منتظر آن بوده، تکمیل شده و بسته را به صورت اشتباه بازسازی می‌کند!

یک روش شماره‌گذاری متفاوت و بهتر آن است که در پروتکل شبکه، یک اندازه پایه و حداقل برای بسته‌ها تعریف شود و این اندازه به قدری کوچک باشد که بسته بتواند از هر شبکه‌ای عبور نماید. هرگاه بسته‌ای قطعه‌قطعه می‌شود، اندازه تمام قطعات به مقدار پایه تنظیم خواهد شد (به استثنای قطعه آخر که ممکن است کوچکتر باشد). در سرآیند هر یک از قطعات باید شماره بسته اصلی و همچنین شماره قطعه مشخص شده باشد. همچنین باید در سرآیند قطعه یک بیت در نظر گرفته شود تا بتوان آخرین قطعه هر بسته را مشخص نمود. (یعنی هر بسته را فقط یکبار می‌توان قطعه قطعه کرد.) در این روش، در سرآیند هر بسته به دو فیلد شماره ترتیب نیاز است: یکی برای شماره بسته اصلی و دیگری برای شماره قطعه.

می‌توان یک حالت بینابین برای شماره‌گذاری بسته‌ها انتخاب کرد: به جای آن که برای هر قطعه شماره آن در سرآیند هر قطعه درج شود، آفست آن قطعه در بسته اصلی، مشخص شود. [یعنی آن که مشخص شود قطعه جاری در کجای بسته اصلی قرار می‌گیرد.] اگر مطابق شکل ۵-۵۱ بجای شماره هر قطعه، آفست محل قرار گرفتن قطعه در بسته اصلی در نظر گرفته شود می‌توان بسته را به هر اندازه ممکن (حتی یک بایت) کوچک کرد.

در برخی از پروتکل‌های شبکه بطور عام از این روش استفاده شده است حتی تا جایی که کل داده‌های ارسالی بر روی یک مدار مجازی به عنوان یک بسته بسیار بزرگ تلقی می‌شود و شماره هر قطعه، شماره ترتیب «اولین بایت قطعه» نسبت به «اولین بایت بسته» فرض می‌شود. [یعنی محل قرار گرفتن قطعه نسبت به شماره اولین بایت بسته اصلی مشخص می‌شود.]



شکل ۵-۵۱. قطعه‌سازی بسته‌ها با فرض آنکه مبنای پایه طول داده‌ها ۱ بایت باشد. (الف) بسته اصلی حاوی ده بایت داده. (ب) مجموعه قطعات پس از عبور از شبکه‌ای که در آن طول حداکثر بسته‌ها با احتساب سرآیند، ۸ بایت است. (ج) مجموعه قطعات پس از عبور از «دروازه‌ای» (Gateway) که در آن طول حداکثر بسته‌ها، ۵ بایت است.

۶-۵ لایه شبکه در اینترنت

قبل از آنکه به توصیف لایه شبکه در اینترنت بپردازیم، مروری بر اصول طراحی آن در گذشته و دلایل موفقیت آن در حال حاضر، خالی از لطف نیست. اصولی که به نظر می رسد اکنون به دست فراموشی سپرده شده اند. این اصول و قواعد در سند RFC 1958 تشریح شده اند و مطالعه آن بسیار ارزشمند است. (مطالعه آن برای طراحان پروتکل باید الزامی شود و در پایان نیز از آنان براساس همین سند امتحان به عمل آید!) این RFC پشتت تحت تأثیر افکاری است که دو محقق Clark (۱۹۸۸) و Saltzer (همکاران و ۱۹۸۴) ارائه نموده اند. در اینجا به اختصار ده مورد از اصول اساسی طراحی پروتکل لایه شبکه را (به ترتیب اهمیت) بررسی می نماییم:

۱. اطمینان از عملکرد صحیح: طراحی یا استاندارد را هیچگاه نهایی و مختومه فرض نکنید مگر آنکه چندین نسخه اولیه و آزمایشی (پروتوتایپ) آن بتوانند با یکدیگر مبادله داده نمایند. بسیار اتفاق افتاده که طراحان شبکه یک استاندارد ۱۰۰۰ صفحه ای تدوین و آنرا به تأیید رسانده اند ولی بعداً متوجه اشکالات اساسی و عدم عملکرد آن شده و مجبور به نوشتن نسخه ۱.۱ استاندارد خود شده اند. این روش به نتیجه نخواهد رسید.

۲. پروتکل را ساده طراحی کنید: در هر کجا که تردید دارید، از ساده ترین راه حل بهره بگیرید. «ویلیام اوگم» این اصل را در قرن چهاردهم بیان کرده است. در یک عبارت امروزی و مدرن: «حداقل ویژگیهای زائد» اگر داشتن یک ویژگی، حیاتی و اساسی نیست آنرا رها کنید بالاخص زمانی که این ویژگی را می توان براساس ترکیبی از ویژگیهای دیگر بدست آورد.

۳. تصمیمات روشن و شفاف بگیرید: اگر برای انجام یک کار چندین راه حل پیش رو دارید فقط یکی را انتخاب نمایید. اگر دو یا چند روش را مدنظر قرار بدهید برای خود مشکل تراشیده اید. امروزه بسیاری از استانداردها دارای گزینه های متعدد، حالات و پارامترهای مختلف هستند چرا که هر یک از طراحان، بر آنکه روش آنها بهترین است، پافشاری کرده اند. طراحان باید با قدرت در برابر این گرایشات فردی مقاومت کرده و فقط بگویند نه!!

۴. طراحی شما مایه جوار باشد: این اصل بدین نظریه منتهی خواهد شد که باید پشته ای از پروتکلها را داشت و هر لایه مستقل از دیگر لایه ها باشد. بدین ترتیب هر گاه نیاز شد یک مایه جوار یا یک لایه تغییر کند، بقیه لایه ها تحت تأثیر قرار نخواهند گرفت.

۵. ناهمگونی عناصر را مدنظر قرار بدهید: در یک شبکه بزرگ، انواع سخت افزار، امکانات مخابراتی و برنامه های کاربردی پیدا می شود. برای آن که بتوان همه آنها را به خدمت گرفت و اداره کرد، طراحی شبکه باید ساده، عمومی و قابل انعطاف باشد.

۶. از گزینه ها و پارامترهای ثابت پرهیز نمایید: اگر در جایی الزاماً به تعریف پارامتر نیاز دارید (مثلاً تعریف طول حداکثر بسته ها) بهترین کار آن است که به جای تعریف پارامترهای ثابت اجازه بدهید فرستنده و گیرنده با یکدیگر مذاکره و بر سر مقدار آن توافق کنند.

۷. به دنبال طراحی خوب باشید: لازم نیست که یک طراحی خوب ایده آل و بی نقص باشد؛ بسیاری از طراحان، طرح خوبی را در اختیار دارند ولی طرح آنان نمی تواند از عهده برخی از موارد نادر و عجیب برآید. به جای آن که اینگونه طرحها را بهم بریزید بایستی در پی یک طراحی خوب بوده و از خود در مقابل خواسته های عجیب و غریب سلب مسئولیت نمایید.

۸. هنگام «ارسال» سخت گیر و دقیق و هنگام «دریافت» با تحمل باشید: به عبارتی دیگر فقط بسته هایی را

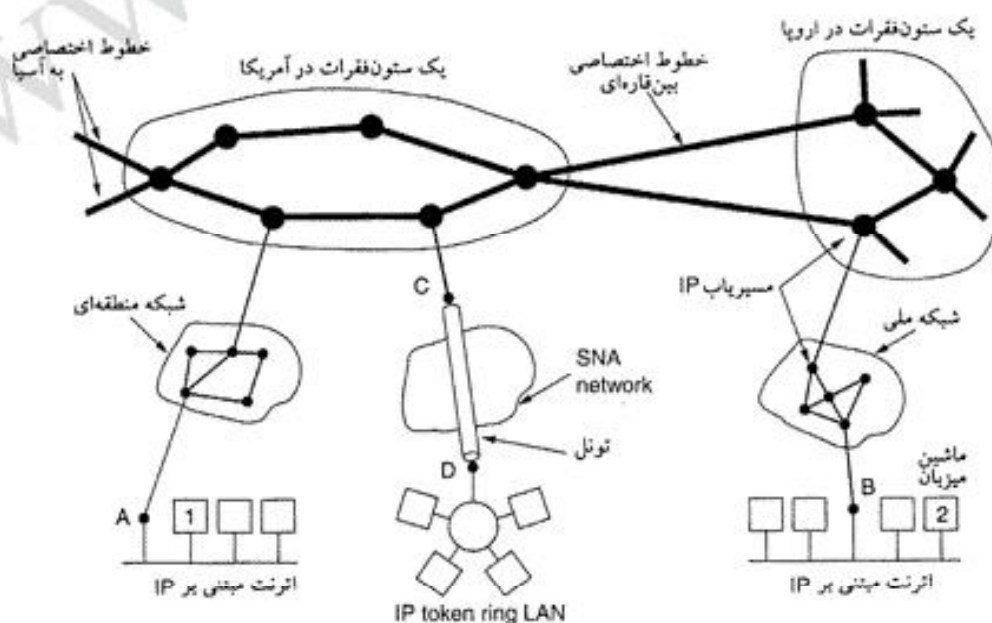
بر روی شبکه بفرستید که دقیقاً منطبق با استاندارد است ولی انتظار آن را داشته باشید که بسته‌های دریافتی انطباق کامل با استاندارد نداشته باشد.

۹. در اندیشه قابلیت گسترش و توسعه پذیری باشید: اگر یک سیستم مجبور به مدیریت میلیون‌ها ماشین و میلیارد‌ها کاربر است، استفاده از هیچ نوع پایگاه اطلاعات مرکزی قابل تحمل و به صلاح نیست و باید بار آن تا حد ممکن بر روی منابع موجود پخش شود.

۱۰. به کارایی و هزینه دقت داشته باشید: اگر شبکه‌ای کارایی پایین یا هزینه سرسام‌آور داشته باشد هیچکس از آن استقبال نخواهد کرد.

حال اصول کلی طراحی را کنار گذاشته و به جزئیات لایه شبکه در اینترنت خواهیم پرداخت. از دیدگاه لایه شبکه، اینترنت را می‌توان مجموعه‌ای از زیر شبکه‌ها یا اصطلاحاً «شبکه‌های خود مختار» در نظر گرفت که بهم متصل شده‌اند. هیچ ساختار مشخص و قطعی بر اینترنت حاکم نیست ولی چندین ستون فقرات (Backbone) برای آن وجود دارد. ستون فقرات از خطوط با پهنای باند بسیار بالا و مسیریابهای سریع تشکیل شده است. شبکه‌های منطقه‌ای (Regional Network) به ستون فقرات اینترنت متصل می‌شوند و شبکه‌های محلی دانشگاهها، شرکتها و مؤسسات ارائه دهنده خدمات اینترنت (ISP) یا شبکه‌های منطقه‌ای در ارتباط هستند. نمادی از این ساختار نسبتاً سلسله‌مراتبی در شکل ۵-۵۲ نشان داده شده است.

آنچه که تمام اجزاء شبکه اینترنت را به هم چسبانده است همان پروتکل IP است. (IP/Internet Protocol) برخلاف بسیاری از پروتکل‌های قدیمی لایه شبکه، این پروتکل از صفر طراحی شده و از همان ابتدا در تفکر وصل شبکه‌ها به یکدیگر بوده است. برای آن که بتوانید وظیفه لایه شبکه را در ذهن خود مجسم کنید بدان بیندیشید که این لایه حداکثر تلاش خود را می‌کند تا یک دیتاگرام را از مبدا به مقصد برساند. [دیتاگرام را یک توده اطلاعات مستقل و دارای هویت در نظر بگیرید.] IP رسیدن بسته‌ها را تضمین نمی‌کند (بلکه حداکثر تلاش خود را مصروف آن می‌کند) و اینکه آیا ماشینهای مبدا و مقصد بر روی یک شبکه واقعهند یا آن که شبکه‌های دیگری در این بین قرار گرفته‌اند، مهم نیست.

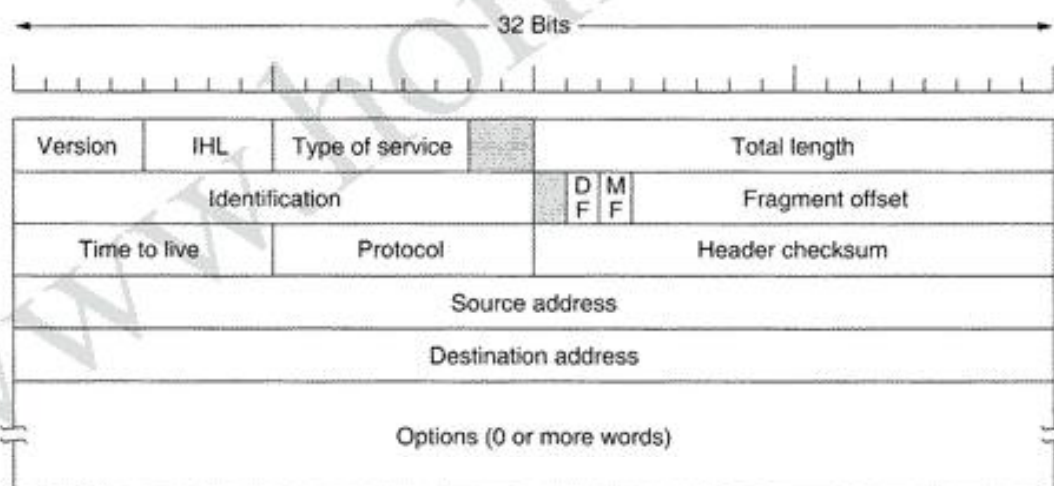


شکل ۵-۵۲. اینترنت مجموعه تعداد بی شماری شبکه است.

ارتباط در اینترنت بدین ترتیب است که: لایه انتقال یک دنباله از داده ها را تحویل گرفته و آنها را به چند دیتاگرام تقسیم می کند. در تئوری، طول حداکثر هر دیتاگرام می تواند تا ۶۴ کیلوبایت باشد ولی در عمل، بطور معمول حول و حوش ۱۵۰۰ بایت است (تا بتواند در یک فریم اترنت جا بگیرد). هر دیتاگرام از طریق اینترنت ارسال می شود و در صورت نیاز، در طول مسیر به قطعات کوچکتر شکسته می شود. وقتی تمام قطعات به ماشین مقصد رسیدند توسط لایه شبکه (IP) به دیتاگرام اصلی بازسازی می شود. این دیتاگرام تحویل لایه انتقال شده و توسط این لایه به پروسه گیرنده تسلیم می شود. همانگونه که در شکل ۵-۵۲ می بینید، بسته ای که از ماشین ۱ منشاء گرفته برای رسیدن به ماشین ۲ باید از شش شبکه بگذرد. در عمل این تعداد از شش تا هم بیشتر است.

۵-۱۶ پروتکل IP

برای شروع به تحلیل پروتکل IP، بهترین کار آن است که قالب بسته های IP را بررسی نماییم.^۱ یک بسته IP در برگزیده دو بخش سرآیند و بخش محتوی است. سرآیند هر بسته دارای یک بخش ثابت ۲۰ بیتی و یک بخش اختیاری با طول متغیر است. ساختار سرآیند بسته IP در شکل ۵-۵۳ نشان داده شده است. ترتیب ارسال بسته IP از بایت پرارزش به کم ارزش است یعنی بایتها به ترتیب از چپ به راست ارسال می شوند؛ بنابراین اولین فیلدی که ارسال می شود فیلد Version (شماره نسخه پروتکل) است. ماشینهای SPARC داده ها را از پرارزش به کم ارزش ذخیره و ارسال می کنند و به آنها ماشینهای Big Endian گفته می شود در حالیکه ماشینهای مثل پنتیوم برعکس هستند یعنی داده ها را از کم ارزش به پرارزش ذخیره و ارسال می کنند (ماشینهای Little Endian) و بالطبع نرم افزار اینگونه ماشینها باید قبل از ارسال یا پس از دریافت، تبدیل لازم را انجام بدهند.



شکل ۵-۵۳. سرآیند IPv4 در پروتکل اینترنت.

فیلد Version مشخص می کند که بسته براساس چه نسخه ای از پروتکل IP سازماندهی و ارسال شده است. با قرار دادن شماره نسخه پروتکل در هر دیتاگرام می توان فرآیند گذر از یک نسخه قدیمی به نسخه جدید را به صورت تدریجی و در خلال چندین سال به انجام رساند و در این مدت برخی از ماشینها از نسخه قدیمی و برخی از نسخه جدید استفاده کنند. اکنون در حال گذر از IPv4 به IPv6 هستیم و اگرچه سالهاست که انتظار این جایگزینی می رود ولی به نظر می رسد هنوز هم به سالها وقت نیاز دارد.

۱. به بسته های IP اصطلاحاً «دیتاگرام IP» گفته می شود. «دیتاگرام IP» یک قطعه داده دارای هویت و شناسنامه است. -م

(Durand, 2001; Wiljakka, 2002; Waddington and Chang, 2002) حتی برخی افراد معتقدند این تغییر هیچگاه محقق نخواهد شد. (Weiser, 2001) مضاف بر این، IPv5 نیز یک بعنوان پروتکل آزمایشی برای انتقال بی درنگ استریم داده^۱ طراحی گردید ولیکن استفاده چندانی از آن نشد.

از آنجایی که طول سرآیند ثابت نیست لذا فیلد IHL بدان منظور در نظر گرفته شده که مشخص کند طول سرآیند (بر مبنای کلمات ۳۲ بیتی) چقدر است. مقدار حداقل این فیلد (یعنی زمانی که هیچ داده‌ای در بخش اختیاری سرآیند وجود ندارد) معادل ۵ است. حداکثر مقدار این فیلد ۴ بیتی، ۱۵ است و طبعاً طول سرآیند به ۶۰ بایت (۱۵×۴) محدود می‌شود؛ یعنی حداکثر طول بخش اختیاری ۴۰ بایت (۶۰-۲۰) است. این فضای ۴۰ بایتی برای برخی از گزینه‌هایی که می‌توان درون این فیلد اختیاری درج کرد (مثل گزینه ثبت مسیر طی شده توسط بسته) بسیار ناکافی است و عملاً فیلد اختیاری را بی‌استفاده کرده است.

فیلد Type of Service (نوع خدمات)، یکی از معدود فیلدهایی است که مضمون و عملکرد آن در طول این سالها تغییر کرده است. از ابتدا (و همچنین اکنون) این فیلد به منظور مشخص کردن کلاسهای مختلف خدمات مدنظر بوده است. در این فیلد می‌توان ترکیبی از خدمات مثل قابلیت اعتماد (Reliability) و سرعت (Speed) را برای بسته تقاضا کرد. برای انتقال دیجیتال صدا، تحویل سریع داده‌ها ارجح‌تر از تحویل مطمئن و بدون خطاست. برعکس، برای انتقال فایل ارسال بدون خطا بسیار مهم‌تر از سرعت انتقال است.

فیلد شش بیتی Type of Service (به ترتیب از چپ به راست) در برگرنده یک فیلد سه بیتی به نام Precedence (فیلد تقدم) و سه بیت علامت به نامهای D و T و R است. فیلد Precedence اولویت بسته را مشخص می‌کند: از صفر (اولویت معمولی) تا ۷ (بالاترین اولویت برای بسته‌های کنترلی شبکه). سه بیت علامت به ماشین میزبان اجازه می‌دهد که از بین سه ویژگی (D: تأخیر، T: ظرفیت خروجی، R: قابلیت اعتماد) نیازهای خود را توصیف نماید. از دیدگاه تئوری این فیلدها امکان آنرا فراهم آورده‌اند که مسیریاب مثلاً از بین یک خط ماهواره‌ای با ظرفیت خروجی بالا و تأخیر زیاد و همچنین یک خط اجاره‌ای با ظرفیت خروجی کم و تأخیر پایین، انتخاب مناسبی داشته باشد، ولیکن در عمل مسیریابها فیلد Type of service را نادیده می‌گیرند.

عاقبت IETF متقاعد شد که مضمون این فیلد را تغییر بدهد تا براساس آن بتوان رده‌های مختلف خدمات مورد نیاز را توصیف کرد. ۶ بیت این فیلد برای مشخص کردن رده کلاس خدماتی است که بسته بدان کلاس تعلق دارد. این کلاسها که قبلاً در بخش ۵-۴-۴ معرفی شدند، شامل: ۴ اولویت صف‌بندی، ۳ احتمال حذف بسته و تعدادی کلاس برای سازگاری با قبل هستند.

فیلد Total Length طول کل بسته را (شامل سرآیند و داده) مشخص می‌نماید. حداکثر طول یک بسته ۶۵۵۳۵ بایت است. در حال حاضر این مقدار حداکثر کفایت می‌کند ولی در آینده با رواج اترنت گیگابیت ممکن است به دیناگرام با طول بیشتری نیاز باشد.

فیلد Identification بدان جهت نیاز است که مشخص کنیم قطعه دریافتی، به کدام دیناگرام متعلق است. تمام قطعات یک دیناگرام واحد دارای مقداری مشابه در فیلد Identification هستند. [بخش ۵-۷ را مطالعه نمایید.] در ادامه یک بیت بلااستفاده و سپس یک فیلد تک بیتی به نام DF^۲ قرار گرفته است. این بیت به مسیریابها فرمان می‌دهد که دیناگرام (بسته) جاری را قطعه‌قطعه نکنند چرا که مقصد از بازسازی آن عاجز است. مثلاً وقتی

۱. Real-Time Stream Protocol.

۲. DF مخفف کلمات Don't Fragment به معنای عدم قطعه‌قطعه‌سازی بسته است.

کامپیوتری بوت می شود، ROM بوت کننده آن ممکن است تقاضا کند که «تصویری از حافظه» (Memory Image) برای او ارسال شود.^۱ با علامتگذاری در بیت DF، فرستنده مطمئن خواهد شد که کل بسته به صورت یکجا تحویل خواهد شد، حتی اگر این بسته به دلیل محدودیتهایی که مسیر بیهینه (در هدایت بسته های بزرگ) دارد مجبور به عبور از مسیرهای غیربیهینه شود. تمام ماشینها ملزم به پذیرش قطعات با طول ۵۷۶ بایت (یا کمتر) هستند. بیت MF (مخفف More Fragment) به معنای آنست که دنباله قطعات هنوز ادامه دارد. در تمام قطعات به استثنای قطعه آخر این بیت ۱ است. به این بیت از آن جهت نیاز است که متوجه شویم آیا تمام قطعات یک دیتاگرام دریافت شده یا هنوز دنباله قطعات ادامه دارد.

فیلد Fragment Offset مشخص کننده آنست که قطعه جاری در کجای دیتاگرام اصلی واقع شده است. طول تمام قطعات (به استثنای قطعه آخر) باید ضربی از ۸ بایت باشد. از آنجایی که این فیلد ۱۳ بیتی است حداکثر می توان ۸۱۹۲ قطعه در هر دیتاگرام داشت و بدین ترتیب طول حداکثر هر دیتاگرام ۶۵۵۳۶ بایت خواهد شد. فیلد Time to live (زمان حیات بسته) شمارنده ای است که طول عمر بسته را تعیین می نماید. فرض بر آنست که این فیلد زمان را برحسب ثانیه مشخص کند و طول عمر بسته حداکثر ۲۵۵ ثانیه باشد. به ازای هر گام (یعنی عبور بسته از یک مسیر یاب)، یک واحد از این فیلد کاسته می شود. همچنین فرض شده که هر گاه بسته درون یک مسیر یاب، زمان زیادی را در صف منتظر ماند تعداد بیشتری از این فیلد کم شود. با تمام این تفاسیل، در عمل فقط تعداد گام محاسبه می شود.^۲ به محض آن که مقدار این فیلد به صفر برسد بسته حذف شده و یک پیغام هشدار به مبدا آن برگردانده می شود. این ویژگی از سرگردان ماندن بسته ها در زیر شبکه (که در اثر اشتباه در جدول مسیر یابی بروز می کند) جلوگیری می نماید.

هر گاه لایه شبکه یک دیتاگرام کامل را بازسازی کرد باید بداند که چه کاری با آن بکند. فیلد Protocol (شماره پروتکل)، مشخص می کند که بسته را باید به کدام پروتسه در لایه انتقال تحویل داد. پروتسه TCP یکی از گزینه هاست؛ ولی پروتسه UDP و چندین پروتسه دیگر نیز می توانند دیتاگرام را تحویل بگیرند. شماره پروتکلها جهانی و در سرتاسر اینترنت استاندارد هستند. پروتکلها و دیگر شماره های استاندارد در سند RFC 1700 ثبت شده است؛ این شماره ها را می توانید در پایگاه اطلاعاتی www.iana.org بدست بیاورید.

فیلد Header Checksum یک کد کشف خطا برای سرآیند است. این کد کشف خطا که صرفاً سرآیند بسته را در برمی گیرد قادر است خطاهایی را که از خرابی در حافظه مسیر یاب ناشی می شود آشکار نماید. الگوریتم کشف خطا بدین ترتیب است که کل سرآیند، در همان بدو ورود در قالب نیم کلمات ۱۶ بیتی با یکدیگر جمع شده و نهایتاً «مکمل یک»^۳ آن محاسبه می شود. اگر سرآیند بسته بدون خطا باشد جمع تمام نیم کلمات ۱۶ بیتی سرآیند، باید صفر باشد. این الگوریتم قدرتمندتر از جمع معمولی است. دقت کنید که فیلد Header Checksum در هر گام باید از نو محاسبه شود چرا که حداقل یکی از فیلدهای سرآیند تغییر می کند (یعنی فیلد Time To Live) ولی برای سرعت بخشیدن به این محاسبه می توان از راهی میانبر استفاده کرد.

فیلدهای Source Address (آدرس مبدا) و Destination Address (آدرس مقصد) شماره شبکه و شماره ماشین مبدا و مقصد را مشخص می کنند. این دو آدرس را در بخشی مجزا بررسی خواهیم کرد.

فیلد Options بدان منظور طراحی شده تا در نسخه های بعدی پروتکل بتوان اطلاعاتی را در سرآیند قرار داد که در نسخه اصلی آن پیش بینی نشده است. همچنین می توان از آن برای آزمودن ایده های جدید بدون درهم

۱. تصویری از هسته سیستم عامل که بصورت کدهای اجرایی ارسال می شود.

۲. یعنی امروزه فقط به ازای عبور بسته از یک مسیر یاب، یک واحد از مقدار این فیلد کم می شود و زمان (برحسب ثانیه) هیچ

تأثیری در این فیلد ندارد. ۳. One's Complement

ریختن آن بهره گرفت. فیلد Option دارای طولی متغیر است؛ هر یک از گزینه‌های مندرج در این فیلد با یک کد بایستی شروع می‌شود تا ماهیت گزینه را مشخص کند. در برخی از گزینه‌ها پس از این کد یک بایستی، فیلد یک بایستی دیگر، طول گزینه را برحسب بایت مشخص می‌نماید. نهایتاً به فیلد Options تعدادی بایت اضافی و زائد افزوده می‌شود تا طول آن ضریبی از ۴ شود. در ابتدا فقط پنج گزینه فهرست شده در شکل ۵-۵۴ تعریف شده بود ولی پس از آن تعدادی گزینه دیگر بدان افزوده شد. فهرست کامل این گزینه‌ها در آدرس www.iana.org/assignments/ip-parameters در دسترس عموم قرار گرفته است.

گزینه	توصیف عملکرد
Security	میزان محرمانه بودن دیتاگرام جاری را تعیین می‌کند.
Strict source routing	فهرست کامل مسیری را که باید دنبال شود تعریف می‌کند.
Loose source routing	فهرستی از مسیریابها که بسته حتماً باید از آنها رد شود.
Record route	مسیریابها را وادار می‌کند تا آدرس IP خود را در بسته درج کنند.
Timestamp	مسیریابها را وادار می‌کند تا آدرس IP خود را به همراه مهر زمان در بسته درج کنند.

شکل ۵-۵۴. برخی از گزینه‌های IP.

گزینه Security مشخص کننده آن است که اطلاعات موجود در بسته تا چه حد محرمانه است. از دیدگاه تئوری، کاربرد این گزینه آن است که مسیریابهای نظامی این بسته را از مسیریابی که از کشورها یا مناطقی می‌گذرند که به زعم آنها نامطمئن و خطرناک هستند، هدایت نکنند. در عمل تمام مسیریابها این فیلد را نادیده می‌گیرند و تنها کاربرد عملی آن می‌تواند کمک به جاسوسان برای انتخاب بسته‌های مورد نظرشان باشد!

با گزینه Strict Source Routing می‌توان مسیر کامل بین مبدا و مقصد را در قالب دنباله‌ای از آدرس‌های IP مشخص کرد. بسته ارسالی ملزم به عبور از این مسیر است. این گزینه به مدیران شبکه کمک می‌کند تا در هنگام خرابی جداول مسیریابی یا عملکرد غیر صحیح مسیریابها بتوانند بسته‌های اضطراری خود را به مقصد برسانند یا می‌تواند برای اندازه‌گیری زمانی و تخمین ترافیک یک مسیر مفید واقع شود.

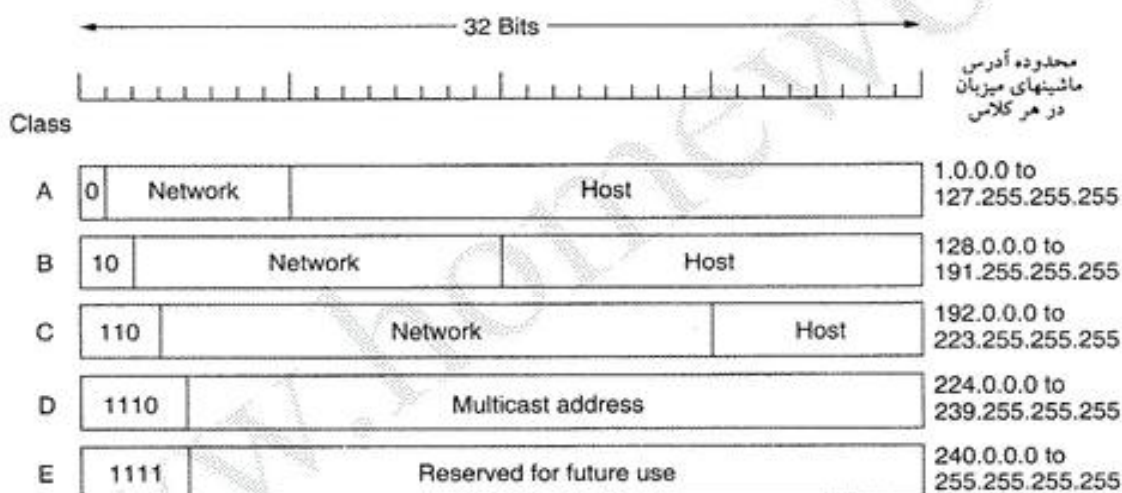
با گزینه Loose Source Routing، بسته ملزم به عبور از مسیریابهای مشخص (طبق ترتیب تعیین شده) می‌باشد ولی اجازه دارد در مسیر خود از مسیریابهای دیگری نیز که در فهرست نیامده، عبور کند. [یعنی در حقیقت فقط برخی از نقاط مسیر مشخص می‌شود و بقیه نقاط توسط مسیریاب انتخاب می‌شود.] بطور معمول در این گزینه فقط تعدادی از مسیریابها مشخص می‌شوند و همین تعداد می‌تواند مسیر مورد نظر را تبیین نماید. به عنوان مثال برای آنکه بسته‌ای را مجبور کنیم برای رفتن از لندن به سیدنی به جای شرق از سمت غرب حرکت کند کافی است در این گزینه سه مسیریاب در نیویورک، لوس آنجلس و هانولولو مشخص شود. این گزینه زمانی سودمند خواهد بود که به دلایل سیاسی یا اقتصادی بخواهیم بسته‌ها از مسیریابی که در برخی کشورهای خاص واقعند عبور نکنند.

گزینه Record Route (ثبت مسیر) به مسیریابهای واقع بر روی مسیر تفهیم می‌کند که باید آدرس IP خود را در همین فیلد (فیلد Option) درج نمایند. این گزینه به مدیران شبکه کمک می‌کند تا بتوانند اشکالات احتمالی در الگوریتم مسیریابی را پیگیری و کشف نمایند (مثلاً: «چرا بسته‌ای که از هیوستون به دالاس روانه شده در توکیو دیده شده است؟») زمانی که برای اولین بار ARPANET راه‌اندازی شد کل مسیریابهای این شبکه ۹ تا بیشتر نبود و فضای چهل بایستی فیلد Options کفایت می‌کرد ولی امروزه به دلیل رشد سرسام‌آور شبکه اینترنت این فضای چهل بایستی بسیار ناکافی و عملاً بی‌کاربرد است.

در آخر، گزینه Timestamp (مهر زمان) عملکردی شبیه به گزینه قبلی (یعنی Record Route) دارد با این تفاوت که به غیر از ثبت آدرس IP هر مسیر یاب، یک «مهر زمان» ۳۲ بیتی نیز در کنار آن درج می شود. این گزینه نیز برای اشکال زدایی از الگوریتمهای مسیر یابی کاربرد دارد.

۲-۶-۵ آدرسهای IP

هر ماشین میزبان و هر مسیر یاب در شبکه اینترنت دارای یک آدرس IP است که در آن شماره شبکه و شماره ماشین، مشخص می شود. ترکیب این دو شماره، منحصر به فرد و یکتا است: بر اساس این اصل اساسی که هیچ دو ماشینی در اینترنت نباید آدرس IP یکسانی داشته باشند. تمام آدرسهای IP، ۳۲ بیتی هستند و برای مشخص کردن آدرس ماشین مبدا و مقصد، در درون فیلدهای مربوطه در بسته های IP قرار می گیرند. اشاره به این نکته بسیار مهم است که آدرس IP در حقیقت یک ماشین میزبان را مشخص نمی کند بلکه به یک «کارت واسط شبکه» اشاره دارد، فلذا اگر ماشینی بر روی دو شبکه واقع شده باشد باید دو آدرس IP داشته باشد ولی در عمل بیشتر ماشینها فقط بر روی یک شبکه قرار گرفته اند و طبعاً یک آدرس IP بیشتر ندارند.^۱



شکل ۵-۵۵. قالب آدرسهای IP.

در چند دهه ابتدایی، آدرسهای IP در پنج رده مختلف دسته بندی شده بودند. این پنج رده در شکل ۵-۵۵ نشان داده شده است. تخصیص فضای آدرس مطابق با این روش، اصطلاحاً روش «آدرس دهی دارای کلاس» (Classful Addressing) نامیده می شود. این روش امروزه کاربرد خود را از دست داده است [از حدود سال ۱۹۹۶] ولی هنوز هم در کتب مختلف به آن اشاره می شود. این روش آدرس دهی را به اختصار توضیح می دهیم. کلاسهای A و B و C و D این امکان را فراهم آورده اند تا بتوان به ترتیب: ۱۲۸ شبکه با ۱۶ میلیون ماشین میزبان (در کلاس A)، ۱۶۳۸۴ شبکه با ۶۵۵۳۶ ماشین (در کلاس B) و حدود ۲ میلیون شبکه با ۲۵۶ ماشین (در کلاس C) را آدرس دهی کرد (اگرچه برخی از این آدرسها برای کارهای ویژه در نظر گرفته شده اند و به هیچ ماشینی منتسب نمی شوند). همچنین در کلاس D از ارسال چند پخش (Multicast) پشتیبانی می شود و می توان بسته های را مستقیماً برای چندین ماشین در شبکه فرستاد.

۱. مسیر یابها که چندین کارت واسط دارند بیش از یک آدرس IP خواهند داشت.

آدرسهای که با چهار بیت ۱۱۱۱ شروع می‌شوند (یعنی کلاس E) هیچ استفاده‌ای ندارند و برای کاربردهای بعدی رزرو شده‌اند. امروزه بیش از ۵۰۰,۰۰۰ شبکه به اینترنت متصل شده‌اند و این تعداد سال به سال افزایش می‌یابد. شماره‌های شبکه [یعنی فیلد Network در آدرس IP] توسط یک موسسه غیرانتفاعی به نام ICANN^۱ مدیریت می‌شود تا تناقضی در انتساب شماره‌های تکراری پدید نیاید. ICANN تخصیص بخشهایی از فضای آدرس‌دهی را به تعدادی «مراکز مجاز منطقه‌ای» واگذار کرده تا آنها آدرسهای IP را بین سرویس دهنده‌های اینترنتی (ISP) و دیگر شرکتها توزیع نمایند.

آدرسهای IP که ۳۲ بیتی هستند معمولاً با نماد دهمی نقطه‌دار نمایش داده می‌شوند. در این روش هر یک از چهار بایت آدرس به صورت مجزا و در مبنای ده (از صفر تا ۲۵۵) نوشته می‌شود. مثلاً آدرس ۳۲ بیتی C0290614 (در مبنای شانزده) به صورت 192.41.6.20 نمایش داده می‌شود. کمترین آدرس IP معادل 0.0.0.0 و بزرگترین آن معادل 255.255.255.255 است.

مقدار صفر و ۱- (یعنی فیلدی که تمام بیتهایش ۱ است) معانی خاص دارد. شکل ۵-۵۶ آدرسهای خاص را مشخص کرده است. مقدار ۰ خود ماشین یا شبکه را مشخص می‌نماید.^۲ مقدار ۱- برای پخش فراگیر (Broadcast) کاربرد دارد، یعنی اگر بیتهای آدرس مقصد بسته تماماً ۱ باشد، گیرنده بسته، تمام ماشینهای آن شبکه می‌باشد.

0 0	This host (همین ماشین)
0 0 ... 0 0	Host A host on this network (ماشینی بر روی همین شبکه)
1 1	Broadcast on the local network (پخش فراگیر بر روی شبکه محلی)
Network 1 1 1 1 ... 1 1 1 1	Broadcast on a distant network (پخش فراگیر بر روی شبکه راه دور)
127	(Anything) Loopback (آدرس حلقه بازگشت)

شکل ۵-۵۶. آدرسهای IP خاص.

آدرس IP معادل 0.0.0.0 توسط ماشینهای میزبان و در حین راه‌اندازی (بوت) مورد استفاده قرار می‌گیرد. هر گاه شماره شبکه در آدرس IP صفر باشد چنین آدرسی به شبکه فعلی اشاره می‌کند. این آدرسها اجازه می‌دهد که ماشینها بدون دانستن شماره شبکه خود، به آن اشاره کنند [یعنی اگر ماشینی بخواهد برای ماشین دیگری در شبکه خودش بسته‌ای بفرستد، دانستن شماره شبکه، مهم نیست]. البته باید هر ماشین، حداقل کلاس آدرس شبکه خود را بداند تا تشخیص بدهد فیلد شماره شبکه چند بیتی است و آن را با صفر پر کند. آدرسی که تمام بیتهای آن ۱ است اجازه می‌دهد که بسته به صورت فراگیر بر روی شبکه محلی پخش شود. [یعنی بسته‌ای با چنین آدرسی را همه ماشینهای موجود بر روی شبکه محلی دریافت می‌دارند]. اگر شماره شبکه با یک مقدار درست تنظیم شده ولی تمام بیتهای شماره ماشین مساوی ۱ باشد می‌توان بسته‌ای را برای تمام ماشینهای یک شبکه LAN، در هر کجای اینترنت ارسال نمود. (البته بسیاری از مسئولین شبکه این ویژگی را غیر فعال و ناممکن کرده‌اند.)

۱. Internet Corporation for Assigned Names and Numbers

۲. یعنی اگر فیلد Network در آدرس IP معادل صفر باشد به معنای خود شبکه و اگر فیلد Host صفر باشد به معنای خود آن ماشین است. بسته‌ای با این آدرسها از شبکه یا ماشین بیرون نخواهد رفت. -م

تمام آدرسهای که به شکل 127.xx.yy.zz هستند به نام «حلقه بازگشت» (Loopback) مشهورند و برای آزمایش نرم افزار، کنار گذاشته شده اند: هر بسته ای که به چنین آدرسهایی ارسال شود به صورت واقعی بر روی سیم منتقل نخواهد شد بلکه به صورت داخلی پردازش شده و با آن همانند بسته ورودی رفتار می شود. این آدرس اجازه می دهد بدون آنکه ماشین فرستنده شماره خود را بداند برای خودش بسته بفرستد.

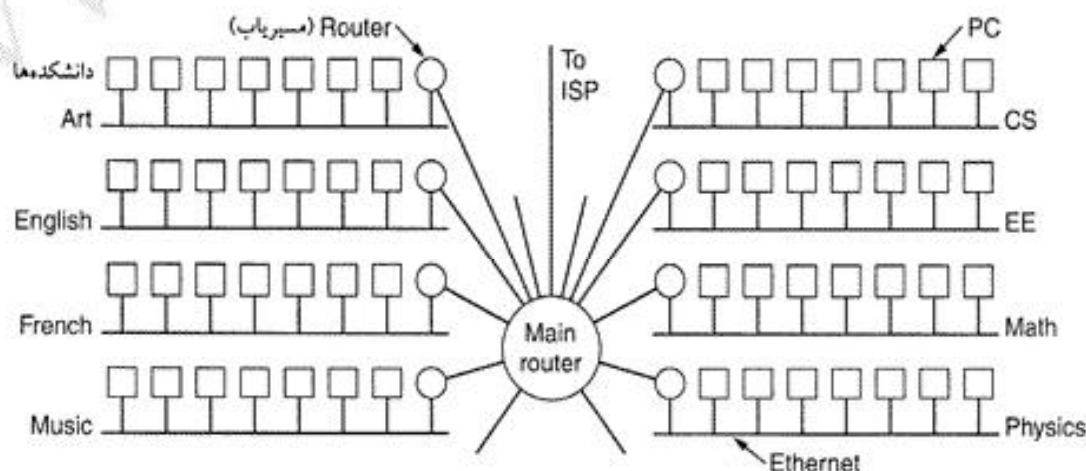
زیرشبکه ها

همانگونه که قبلاً بررسی کردیم تمام ماشینهای یک شبکه باید دارای شماره شبکه یکسانی باشند. با رشد شبکه، این ویژگی در آدرس دهی IP، باعث بروز مشکلاتی می شود. به عنوان مثال دانشگاهی را مدنظر قرار بدهید که در ابتدا با یک آدرس کلاس B شروع کرده و آن را در شبکه اترنت متعلق به دانشکده علوم کامپیوتر بکار گرفته است. یک سال بعد دانشکده مهندسی برق می خواهد که به اینترنت متصل شود و با خرید یک تکرار کننده (Repeater) سعی می کند به شبکه اترنت دانشکده علوم کامپیوتر متصل شود. با گذشت زمان، دانشکده های دیگر نیز تقاضای وصل به شبکه می کنند و بدین ترتیب به دلیل محدودیت در تعداد تکرار کننده های اترنت این کار غیر ممکن می شود. بنابراین به سازماندهی متفاوتی نیاز است.

از طرف دیگر، ثبت و دریافت یک کلاس آدرس مجزا برای شبکه های جدید نیز دشوار است چرا که آدرسهای شبکه محدود و کم هستند و مضاف بر این، آدرس کلاس B اختصاص داده شده به این دانشگاه برای آدرس دهی به بیش از ۶۰۰۰۰ ماشین کافی است. مشکل اینجا است که یک کلاس A، B یا C به یک شبکه واحد اشاره می کند نه به یک مجموعه از شبکه های LAN متصل بهم.

پس از آنکه سازمانهای متعددی به مشکل کمبود آدرس شبکه، دچار شدند یک تغییر بسیار کوچک در این آدرسها مشکل را حل کرد: راه حل آن است که اجازه بدهیم شبکه به چندین بخش مستقل داخلی تقسیم شده ولی کماکان در دنیای خارج به عنوان یک شبکه واحد تلقی شود.

یک «شبکه دانشگاهی» (Campus)، نمادی شبیه به شکل ۵-۵۷ دارد که در آن یک مسیریاب مرکزی از طرفی به یک ISP (یا یک شبکه منطقه ای) و از طرف دیگر به چندین شبکه اترنت در دانشکده های مختلف وصل شده است. هر یک از شبکه های اترنت دارای یک مسیریاب محلی هستند که از طریق آن به مسیریاب مرکزی متصل شده اند. (البته ممکن است به جای مسیریاب مرکزی یک ستون فقرات قرار گرفته باشد ولیکن چگونگی اتصال مسیریابها در این مثال مهم نیست.)



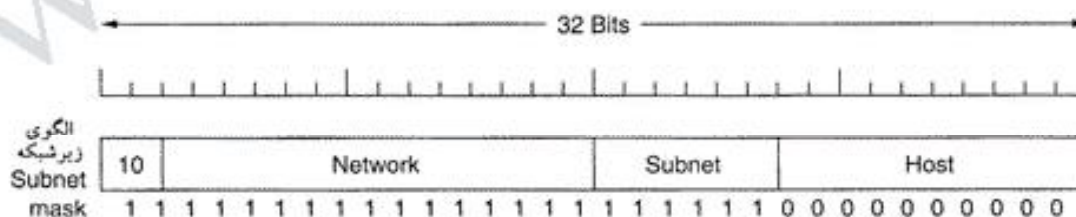
شکل ۵-۵۷ یک شبکه دانشگاهی که شامل چندین LAN متعلق به دانشکده های مختلف است.

در ادبیات شبکه اینترنت، به هر بخش از یک شبکه بزرگ و خودمختار، اصطلاحاً «زیر شبکه» (Subnet) گفته می شود. (مثلاً در مثال بالا هر یک از شبکه های اینترنت یک «زیر شبکه» تلقی می شوند.) همانطور که در فصل اول اشاره کردیم، این استعاره می تواند با واژه «زیر شبکه» به معنای مجموعه ای از مسیریابها و خطوط ارتباطی در یک شبکه اشتباه شود. البته زمینه و مفاد یک بحث، معنای مورد نظر آن را مشخص خواهد کرد. در این بخش و بخش آتی، هر گاه واژه «زیر شبکه» را بکار بردیم، تعریف جدید آن مد نظر است.

وقتی بسترهای به مسیریاب مرکزی وارد می شود، مسیریاب چگونه تشخیص می دهد که آنرا به کدام زیر شبکه (اینترنت) تحویل بدهد؟ یک راه آن است که جدولی با ۶۵۵۳۶ درایه (Entry) داشته باشیم و هر درایه محل هر ماشین میزبان را در کل شبکه مشخص کند. این ایده عملی است ولی به یک جدول بسیار بزرگ در مسیریاب مرکزی نیاز است و در صورت اضافه شدن، حذف یا تغییر محل ماشینها، باید تنظیمات لازم بصورت دستی انجام شود.

به جای این روش، الگوی جدیدی ابداع شده است. در این روش به جای آنکه یک کلاس B (با ۱۴ بیت برای شماره شبکه و ۱۶ بیت برای شماره ماشین) داشته باشیم، بخشی از فضای شانزده بیتی شماره ماشین برای تعیین «شماره زیر شبکه» در نظر گرفته می شود. به عنوان مثال اگر دانشگاه ۳۵ دانشکده داشته باشد، می تواند ۶ بیت را برای شماره زیر شبکه و ده بیت باقیمانده را برای شماره ماشین در نظر بگیرد. بدین ترتیب می تواند حداکثر ۶۴ زیر شبکه اینترنت با حداکثر ۱۰۲۲ ماشین میزبان داشته باشد. (قبلاً اشاره شد که شماره های ۰ یا ۱- کاربرد خاص داشته و قابل استفاده نیست). این تقسیم بندی را بعداً می توان تغییر داد.

برای تعریف و پیاده سازی زیر شبکه ها، مسیریاب مرکزی به یک «الگوی زیر شبکه» (Subnet Mask) نیاز دارد تا مطابق با شکل ۵-۵۸ بتوان مرز بین شماره ماشین و شماره شبکه و شماره زیر شبکه را مشخص و تفکیک کرد. الگوی زیر شبکه (Subnet Mask) نیز به صورت نماد نقطه دار دهدهی نوشته می شود. همچنین برای راحتی نمایش، می توان پس از آدرس IP یک علامت ۳/ گذاشت و پس از آن تعداد بیت های ۱ موجود در الگوی زیر شبکه را تعیین کرد. به عنوان مثال در شکل ۵-۵۸ الگوی زیر شبکه را می توان به صورت 255.255.252.0 نوشت. راه دیگر آن است که از نماد 22/ استفاده نماییم که مشخص کننده آن است که الگوی زیر شبکه ۲۲ بیت طول دارد. (۲۲ بیت ۱ از سمت چپ به راست)



شکل ۵-۵۸. یک کلاس B به ۶۴ زیر شبکه بخش بندی شده است.

خارج از محدوده این شبکه، این چنین تقسیم بندی و تعداد زیر شبکه ها مشهود و مشخص نیست لذا تعریف زیر شبکه جدید مستلزم آن نیست که با ICANN تماس گرفته شده یا تغییری در پایگاه اطلاعات خارج از آن شبکه داده شود. در این مثال، اولین زیر شبکه از آدرسهای استفاده می کند که مثلاً از 130.50.4.1 شروع می شوند. آدرسهای دومین زیر شبکه از 130.50.8.1 و آدرس سومین زیر شبکه از 130.50.12.1 شروع می شوند. برای آنکه ببینیم چرا شماره زیر شبکه ها چهار تا چهار تا اضافه می شود، شکل دودویی این آدرسها را در نظر می گیریم:

زیر شبکه ۱	10000010 00110010 000001 00 00000001
زیر شبکه ۲	10000010 00110010 000010 00 00000001
زیر شبکه ۳	10000010 00110010 000011 00 00000001

در الگوی بالا خط عمودی که به صورت | نشان داده شده مرز بین شماره زیر شبکه و شماره ماشین را مشخص می نماید. در سمت راست، ۶ بیت برای شماره زیر شبکه و درست چپ ۱۰ بیت برای شماره ماشین در نظر گرفته شده است.

بررسی عملکرد زیر شبکه ها مستلزم آنست که تشریح کنیم بسته های IP در مسیریابها چگونه پردازش می شوند. هر مسیریاب جدولی را در اختیار دارد که در آن مجموعه ای از آدرسهای IP به صورت زوجهای (، شماره شبکه) و (شماره ماشین، همین شبکه)^۱ درج شده است. نوع اول مشخص می کند که چگونه می توان به یک شبکه راه دور رسید. نوع دوم، آدرس ماشینهای محلی را [که مستقیماً به آن مسیریاب متصلند] مشخص می کند. همچنین در این جدول، آدرس کارت واسط شبکه برای رسیدن به هر یک از این زیر شبکه ها (و پاره ای اطلاعات اضافی) درج شده است.

وقتی یک بسته IP توسط یک مسیریاب دریافت می شود، آدرس مقصد آن در جدول مسیریابی جستجو می شود. اگر مقصد بسته یک شبکه راه دور باشد از طریق یکی از کارتهای واسط به سوی مسیریاب بعدی هدایت می شود. اگر مقصد بسته یکی از ماشینهای محلی باشد (مثلاً یکی از ماشینهای LAN متصل به مسیریاب)، مستقیماً برای آن ماشین ارسال می شود. اگر مقصد بسته وجود نداشته یا شناسایی نشود، بسته به سوی یک مسیریاب پیش فرض (Default Router) هدایت می شود. این مسیریاب دارای جدول مسیریابی کاملتر بوده و احتمالاً یک «مسیریاب مرزی» است. پس طبق این الگوریتم، هر مسیریاب فقط باید مشخصات دیگر شبکه ها (یا به عبارت بهتر زیر شبکه ها) و همچنین ماشینهای محلی خود را بداند و بدین ترتیب اندازه جدول مسیریابی شدت کاهش می یابد.

وقتی زیر شبکه جدیدی تعریف می شود، جدول مسیریابی تغییر می کند و در آن درایه هایی به شکل (0، شماره زیر شبکه، شماره همین شبکه) و (0، شماره همین زیر شبکه، شماره همین شبکه)^۲ اضافه می شود. بدین ترتیب یک مسیریاب که به زیر شبکه k متصل است فقط می داند که چگونه باید به زیر شبکه های دیگر برسد و همچنین آدرس تمام ماشینهای متصل به زیر شبکه خودش (یعنی زیر شبکه k) را می داند. بنابراین مسیریابها مجبور نیستند جزئیات آدرس ماشینهای واقع در زیر شبکه های دیگر را بدانند. در حقیقت تنها کاری که مسیریاب در برخورد با آدرسهای IP متعلق به ماشینهای خارج از زیر شبکه خود انجام می دهد آنست که: آدرسهای IP را با «الگوی زیر شبکه» (Subnet Mask) به صورت بولین AND می کند تا بخش شماره ماشین آن حذف شود. سپس آدرس حاصل را در جدول خود جستجو می کند. به عنوان مثال فرض کنید بسته ای به آدرس 130.50.15.6 ارسال و توسط مسیریاب مرکزی (شکل ۵.۵۷) دریافت شده باشد. مسیریاب مرکزی این آدرس را با الگوی زیر شبکه 255.255.252.0/22 به صورت بولین AND می کند و نتیجه 130.50.12.0 بدست می آید. این آدرس در جدول مسیریابی جستجو می شود تا خط خروجی مناسب برای رسیدن به زیر شبکه ۳ مشخص گردد. تعریف زیر شبکه در حقیقت با ایجاد یک سلسله مراتب سه سطحی حجم جدول مسیریابی را کاهش می دهد. این سلسله مراتب سه سطحی تشکیل شده از: شماره شبکه + شماره زیر شبکه + شماره ماشین میزبان.^۳

۱. (Network, 0) (this-network, host)

۲. (this-network, subnet, 0) (this-network, this subnet, host)

۳. مسیریابی سلسله مراتبی را در بخش ۵-۲-۶ از همین فصل مطالعه کنید.

CIDR: مسیریابی براساس آدرسهای بدون کلاس

چندین دهه است که از پروتکل IP استفاده گسترده ای می شود. عملکرد این پروتکل بسیار عالی بوده و همانگونه که اشاره کردیم به رشد نمایی شبکه اینترنت کمک کرده است. متأسفانه IP در حال قربانی شدن در پای محبوبیت خود است چراکه با کمبود فضای آدرس مواجه شده است. از زمانی که افق این مشکل پدیدار شد، بحثها و مناقشات بسیار گسترده ای را در جامعه اینترنت دامن زد که چه کاری می توان برای آن انجام داد. در بخش جاری، این مشکل و راه حل های پیشنهادی را تشریح می نماییم.

تا سال ۱۹۸۷ افراد دوراندیشی که حدس می زدند در روزگاری تعداد شبکه های متصل به اینترنت به مرز ۱۰۰,۰۰۰ برسد، انگشت شمار بودند و حتی اغلب متخصصین این فن نیز چنین ایده ای را به مسخره می گرفتند ولی صدهزارمین شبکه دنیا در سال ۱۹۹۶ به اینترنت پیوست و به نحوی که در بالا اشاره شد اینترنت در حال مواجه شدن با کمبود آدرسهای IP است. در اصل بیش از دو میلیارد آدرس IP وجود دارد ولی در عمل به دلیل تقسیم بندی نامناسب فضای آدرس در قالب چند کلاس، میلیون ها آدرس IP به هدر رفته است. (شکل ۵-۵۵ را ببینید.) بالاخص، بیشترین محدودیت در کلاس B است. برای بسیاری از مؤسسات یک شبکه کلاس A با شانزده میلیون آدرس، بسیار بزرگ و بی مصرف و کلاس C با ۲۵۶ آدرس بسیار کوچک و ناکافی است؛ فقط شبکه کلاس B با ۶۵۵۳۶ آدرس مناسب است. این مشکل در فرهنگ رایج اینترنت به نام مشکل «سه خرس» مشهور شده است. (برگرفته از داستان موطلایی و سه خرس)

حقیقت آنست که حتی کلاس B هم برای بسیاری از سازمانها و مؤسسات، بسیار بزرگ است. بررسیها نشان داده که بیش از نیمی از شبکه های کلاس B، کمتر از ۵۰ ماشین دارند! برای چنین شبکه هایی استفاده از کلاس C کفایت می کند ولی بی تردید تمام مؤسساتی که متقاضی کلاس B بوده اند بدان می اندیشیده اند که روزی فیلد هشت بیتی کلاس C برای شبکه آنها کافی نخواهد بود. نگاهی به گذشته نشان می دهد که بهتر آن بوده در کلاس C فیلد شماره شبکه به جای هشت بیت، ده بیت می بود تا در هر شبکه ۱۰۲۲ ماشین میزبان قابل آدرس دهی باشد. اگر چنین حالتی را می داشتیم شاید بسیاری از مؤسسات و سازمانها به کلاس C اکتفا می کردند و در عین حال نیم میلیون از چنین شبکه هایی قابل تعریف بود. (برخلاف کلاس B که فقط ۱۶۳۸۴ شبکه قابل تعریف است).

نمی توان تقصیر را به گردن طراحان اینترنت انداخت که چرا تعداد کلاسهای B را بیشتر (و با فضای آدرس کوچکتر) در نظر نگرفتند. در روزگاری که تصمیم گرفته شد فقط سه کلاس وجود داشته باشد، اینترنت شبکه ای تحقیقاتی بود که فقط مراکز پژوهشی و دانشگاهی مهم ایالات متحده را بهم متصل می کرد. (البته به اضافه معدود شرکتها و سایت های نظامی که آنها نیز در کار پژوهش بودند.) هیچکس نمی توانست پیش بینی کند که اینترنت روزی بتواند در حد وسیع و به عنوان سیستمی ارتباطی حتی با شبکه های تلفن رقابت نماید! در آن زمان هیچکس در این ادعا شکی نداشت که اگر تمام ۲۰۰۰ کالج و دانشگاه ایالات متحده و حتی اکثر دانشگاههای جهان به اینترنت پیوندند باز هم ۱۶۰۰۰ تا نمی شود چراکه این تعداد دانشگاه در کل دنیا وجود نداشت. به علاوه در آن زمان برای آنکه پردازش بسته ها سریعتر انجام بشود بهتر بود فیلد شماره ماشین در آدرس IP، تعداد صحیحی از بایت باشد. [به دلیل سرعت پایین پردازنده ها در آن زمان، پردازش بیتی فیلدهای آدرس، از سرعت مسیریابی می کاست.]

ولیکن در طرف مقابل اگر مثلاً برای فیلد شماره شبکه در کلاس B، بیست بیت کنار گذاشته می شد مشکل دیگری بروز می کرد: «مشکل رشد انفجاری جداول مسیریابی». از دید مسیریابها فضای آدرسهای IP، سلسله مراتب دوسطحی شامل شماره شبکه و شماره ماشینها است. مسیریابها مجبور به دانستن شماره ماشینهای میزبان

نیستند ولی باید شماره شبکه ها را بدانند. اگر حتی نیمی از شبکه های کلاس C بکار گرفته شده باشند هر مسیریاب در کل اینترنت نیاز به جدولی با نیم میلیون درایه (Entry) دارد تا بتواند خط خروجی مناسب برای رسیدن به هر شبکه را مشخص نماید. (گذشته از اطلاعات دیگری که به ازای هر شبکه باید در جدول مسیریابی درج شود). شاید تهیه فضای فیزیکی لازم برای ذخیره نیم میلیون درایه (Entry) در جدول مسیریابی، امکان پذیر باشد هر چند برای مسیریابهایی که جداول مسیریابی را در حافظه نوع ایستا (Static RAM) ذخیره می کنند، چنین فضایی بسیار گران تمام می شود. مشکل اساسی در پیچیدگی الگوریتمهای مدیریت و پردازش چنین جدولی است. پیچیدگی زمانی این الگوریتمها غیرخطی است.^۱ از این بدتر آنکه نرم افزار یا سخت افزار طراحی شده برای مسیریابهای موجود، زمانی طراحی شده که به اینترنت بیش از هزار شبکه متصل نبود و به نظر می رسید که یک دهه طول می کشد تا این تعداد به ۱۰۰۰۰ برسد. امروزه اینگونه طراحیها بهینه نیستند.

مضاف بر این، در الگوریتمهای مختلف مسیریابی نیاز است که جداول مسیریابی بطور متناوب ارسال و مبادله شود. (مثل الگوریتم بردار فاصله) هر چه جداول مسیریابی بزرگتر باشد احتمال آنکه بخشی از آن در حین مبادله از دست برود بیشتر خواهد شد و به نقص داده های جدول مسیریابی و احتمالاً ناپایداری فرآیند هدایت بسته ها خواهد انجامید.

مشکل جداول مسیریابی را می توان با افزایش «سطوح سلسله مراتب» حل کرد. مثلاً می توان آدرسهای IP را بدین نحو تعریف کرد که شامل فیلدهای کشور، ایالت / استان، شهر، شبکه و شماره ماشین میزبان باشد. بدین ترتیب مسیریابهای ستون فقرات در جهان فقط باید در خصوص مسیریابهای رسیدن به هر کشور آگاهی داشته باشند، مسیریابهای درون کشور در خصوص مسیریابهای رسیدن به هر ایالت یا استان، مسیریابهای ایالت یا استان در مورد مسیریابهای رسیدن به هر شهر و مسیریابهای هر شهر فقط باید راه رسیدن به هر شبکه را بدانند. متأسفانه چنین راه حلی نیازمند فضایی بزرگتر از ۳۲ بیت برای آدرس IP است و طبعاً از فضای آدرس استفاده بهینه نخواهد شد. (چرا که مثلاً در این الگو کشور لیختن اشتاین به همان تعداد بیت در آدرس IP دارد که کشور ایالات متحده!)

کوتاه سخن آنکه هر یک از راه حلهای ارائه شده مشکلی را حل و مشکل جدیدی را ایجاد می کردند. راه حل نهایی و پیاده شده در اینترنت که توانست به اینترنت اجازه نفس کشیدن بدهد، روش CIDR (مسیریابی براساس آدرسهای بدون کلاس) بود. ایده اصلی در CIDR که در سند RFC 1519 تشریح شده آنست که آدرسهای IP بدون در نظر گرفتن کلاس و به صورت بلوکهایی با طول متغیر تخصیص یابد. مثلاً اگر یک سایت نیاز به ۲۰۰۰ آدرس داشته باشد یک بلوک آدرس ۲۰۴۸ تایی به او داده می شود.

حذف کلاسهای آدرس، فرآیند هدایت بسته ها را پیچیده تر می کند. در سیستم مبتنی بر کلاس، فرآیند هدایت بدین نحو بود که وقتی بسته ای به یک مسیریاب می رسید، یک کپی از آدرس IP به اندازه ۲۸ بیت به راست شیفت داده می شد تا فقط چهار بیت سمت چپ آدرس (که کلاس آدرس را مشخص می کند) باقی بماند. براساس این چهار بیت (۱۶ حالت مختلف) بسته ها در یکی از کلاسهای A و B و C (و D در صورت پشتیبانی از آن) مرتب می شدند. (از این ۱۶ حالت مختلف، هشت حالت برای کلاس A است - 0xxx - چهار حالت برای کلاس - 10xx - B، دو حالت برای کلاس C - 110x - و دو حالت برای کلاسهای D و E است.) پس از تشخیص کلاس آدرس، برای بدست آوردن شماره شبکه، آدرس IP با یکی از الگوهای 8-، 16-، 24- به صورت بولی AND و بخش شماره ماشین حذف می شد. سپس شماره شبکه در هر یک از جداول مربوط به آدرسهای کلاس A، B و C

۱. پیچیدگی غیرخطی این الگوریتمها عموماً $O(n^2)$ و $O(n \log n)$ است. - م

جستجو می‌شد. جداول مسیریابی برای کلاسهای A و B برحسب شماره شبکه ایندکس شده بودند.^۱ در عوض جدول مسیریابی برای کلاس C مبتنی بر روش جداول Hash (Hash Table) پیاده شده بود. پس از آنکه درایه متناظر با آدرس شبکه در یکی از این جداول پیدا می‌شد خط خروجی متناسب با آن شبکه مشخص شده و بسته بر روی آن خط هدایت می‌گردید.

در CIDR این الگوریتم ساده، کار نخواهد کرد. در عوض به هر یک از درایه‌های جدول مسیریابی یک فیلد ۳۲ بیتی جدید افزوده شده که الگوی آدرس را [از طریق یک MASK سی و دو بیتی] مشخص می‌کند. بدین ترتیب برای تمام شبکه‌ها فقط یک جدول مسیریابی یکتا وجود دارد که در حقیقت یک آرایه سه ستونی متشکل از آدرس IP، الگوی زیرشبکه (Subnet Mask) و خط خروجی است. وقتی بسته‌ای وارد می‌شود ابتدا آدرس IP آن استخراج می‌شود. سپس جدول مسیریابی درایه به درایه (Entry by Entry) جستجو و آدرس مقصد بسته پس از AND شدن با الگوی زیرشبکه از هر درایه با آدرس IP از آن درایه مقایسه می‌شود. این فرآیند آنقدر تکرار می‌گردد تا به موارد مطابقت برسد. این امکان وجود دارد که چندین درایه با یک آدرس IP مطابقت داشته باشد (به دلیل طول متفاوت الگوهای زیرشبکه). در این حالت درایه‌ای که طول الگوی زیرشبکه آن از همه بزرگتر است از بین آنها انتخاب می‌شود. به عبارتی اگر دو مورد تطابق با طول الگوی 20/ (255.255.240.0) و الگوی 24/ (255.255.255.0) پیدا شود، درایه دوم انتخاب می‌شود.

برای سرعت بخشیدن به فرآیند جستجو و مطابقت، الگوریتمهای پیچیده‌ای ابداع شده است. (Ruiz-Sanches et al. 2001) مسیریابهای تجاری در بازار امروز از تراشه‌های VLSI خاصی بهره گرفته‌اند که الگوریتم مذکور را به صورت یک «سخت‌افزار درون‌کار» (Embedded Hardware) پیاده‌سازی کرده‌اند.

برای آنکه فهم فرآیند هدایت بسته‌ها در CIDR را ساده‌تر کنیم مثالی را مدنظر قرار بدهید که در آن میلیون‌ها آدرس تعریف شده است و آدرس شروع 194.24.0.0 است. فرض کنید که دانشگاه کمبریج به ۲۰۴۸ آدرس نیاز دارد و آدرسهای 194.24.0.0 تا 194.24.7.255 به آن اختصاص داده شده است. (الگوی زیرشبکه نیز 255.255.248.0 است.) بعداً دانشگاه آکسفورد تقاضای ۴۰۹۶ آدرس IP می‌دهد. از آنجایی که بلوکهای آدرس ۴۰۹۶ تایی باید در مرز ۴۰۹۶ بایستی قرار بگیرد نمی‌توان آدرسهای که از 194.24.8.0 شروع می‌شود را به آن اختصاص داد. در عوض آدرس اختصاص داده شده به او در محدوده 194.24.16.0 تا 194.24.31.255 و با الگوی 255.255.240.0 خواهد بود. در اینجا دانشگاه ادینبورو تقاضای ۱۰۲۴ آدرس داده و فضای 194.24.8.0 تا 194.24.11.255 با الگوی 255.255.252.0 به او تعلق می‌گیرد. این انتسابها در جدول ۵-۵ خلاصه شده‌اند.

الگوی نمایش	تعداد آدرس	آخرین آدرس	اولین آدرس	دانشگاه
194.24.0.0/21	2048	194.24.7.	194.24.0.0	Cambridge
194.24.8.0/22	1024	194.24.11.255	194.24.8.0	Edinburgh
194.24.12/22	1024	194.24.15.255	194.24.12.0	در دسترس و آزاد
194.24.16.0/20	4096	194.24.31.255	194.24.16.0	Oxford

شکل ۵-۵. انتساب آدرسهای IP.

۱. به عبارت ساده به دلیل کم بودن تعداد شبکه‌ها جداول مسیریابی برای کلاسهای A و B در ساختمان داده‌ای شبیه به آرایه ذخیره می‌شد. -م

حال جداول مسیریابی در تمام مسیریابهای واقع بر ستون فقرات اینترنت در جهان باید با این سه درایه جدید به‌هنگام شود. هر درایه دارای یک آدرس مبنا و یک الگوی زیرشبکه است. این درایه‌ها در مبنای دو عبارتند از:

الگوی زیرشبکه (Subnet Mask)	آدرس
C: 11000010 00011000 00000000 00000000 11111111 11111111 11111000 00000000	
E: 11000010 00011000 00001000 00000000 11111111 11111111 11111100 00000000	
O: 11000010 00011000 00010000 00000000 11111111 11111111 11110000 00000000	

حال ببینیم وقتی که بسته‌ای با آدرس 194.24.17.4 وارد یک مسیریاب می‌شود چه اتفاقی می‌افتد. این آدرس به صورت دودویی عبارت است از:

11000010 00011000 00010001 00000100

ابتدا این آدرس با الگوی زیرشبکه کمبریج، AND می‌شود و نتیجه زیر بدست می‌آید:

11000010 00011000 00010000 00000000

این مقدار با آدرس مبنای دانشگاه کمبریج مطابقت ندارد. حال مجدداً آدرس اصلی با الگوی زیرشبکه دانشگاه ادینبورو AND شده و نتیجه زیر بدست می‌آید:

11000010 00011000 00010000 00000000

این مقدار نیز با آدرس مبنای دانشگاه ادینبورو تطابق ندارد و همین کار برای دانشگاه آکسفورد تکرار شده مقدار زیر بدست می‌آید:

11000010 00011000 00010000 00000000

این مقدار با آدرس مبنای دانشگاه آکسفورد مطابقت دارد. اگر هیچ مورد تطبیق دیگری در جدول یافت نشد بسته بر روی خطی ارسال می‌شود که در درایه متناظر با شبکه دانشگاه آکسفورد درج شده است.

حال اجازه بدهید، آدرس این سه دانشگاه را از دید یک مسیریاب در نبراسکای اوهاما بررسی کنیم. این مسیریاب چهار خط به مینیاپولیس، نیویورک، دالاس و دنور دارد. وقتی نرم‌افزار مسیریاب اوهاما، این سه درایه جدید را جهت درج در جدول مسیریابی خود دریافت می‌دارد، متوجه می‌شود که قادر است هر سه تای آنها را در یک «درایه واحد و تجمیع شده» (Aggregate Entry) به صورت 194.24.0.0/19 ادغام نماید.^۱ آدرس و الگوی زیرشبکه در مبنای دو به صورت زیر است:

11000010 00000000 00000000 00000000 11111111 11111111 11100000 00000000

طبق این درایه تمام بسته‌هایی که به مقصد یکی از این سه دانشگاه روانه شده‌اند به سوی نیویورک هدایت می‌شوند. با تجمیع این سه درایه، مسیریاب اوهاما توانسته به میزان دو درایه حجم جدول خود را کاهش بدهد.

به همین ترتیب اگر مسیریاب نیویورک برای تمام ترافیک متبقی به انگلستان فقط یک خط به لندن داشته باشد او نیز سه درایه فوق را در یک درایه ادغام می‌کند ولیکن اگر برای لندن و ادینبورو دو خط مجزا داشته باشد باید هر سه تای آنها را بطور مجزا در جدول خود ذخیره کند. عمل تجمیع (Aggregation) در اینترنت به طور گسترده‌ای مورد استفاده قرار گرفته تا حجم جداول مسیریابی کاهش یابد.

آخرین نکته در این مثال آن است که بر طبق درایه ادغام شده در جدول مسیریابی مسیریاب واقع در اوهاما

۱. از آن جهت امکان تجمیع این سه آدرس وجود داشته که بسته‌هایی که مقصدشان هریک از این سه دانشگاه است باید بر روی خط خروجی یکسانی بروند. سم

حتی بسته‌هایی که به آدرس اختصاص داده نشده روانه هستند [یعنی آدرسهای بین 194.24.12.0 تا 194.24.15.255] نیز به سوی نیویورک هدایت می‌شوند. مادامی که این آدرسها به کسی اختصاص داده نشده هیچ مشکلی به وجود نمی‌آید چرا که بنا نیست بسته‌هایی با این آدرسها تولید شوند. ولی اگر این بلوک آدرس، به شرکتی در کالیفرنیا داده شود باید درایه‌ای جدید به شکل 194.24.12.0/22 در جداول مسیریابی تمام مسیریابها درج شود تا بسته‌هایی به مقصد این شبکه نیز بدرستی مسیریابی شوند.

NAT: ترجمه آدرسهای شبکه

آدرسهای IP کمپاب و ارزشمند هستند: یک ISP ممکن است یک بلوک آدرس با الگوی 16/ (همان کلاس B سابق) و توانایی آدرس دهی ۶۵۵۳۴ ماشین میزبان، داشته باشد. اگر تعداد مشتریان این ISP از این تعداد بیشتر شود مشکل بهم می‌زند. برای مشتریان خانگی که از طریق خطوط تلفن متصل می‌شوند، راه حل این مشکل آن است که وقتی مشتری شماره‌گیری کرد و وارد شد به او موقتاً یک آدرس IP پویا اختصاص داده شود و پس از پایان نشست و قطع ارتباط، این آدرس پس گرفته شود. در این روش شبکه‌ای با آدرس 16/ (کلاس B) می‌تواند حداکثر ۶۵۵۳۴ کاربر فعال داشته باشد که این تعداد حتی برای یک ISP با چند صد هزار مشتری نیز کفایت می‌کند. به محض آنکه یک نشست خاتمه یافت آدرس IP منتسب شده قبلی، به تماس گیرنده بعدی داده می‌شود. این استراتژی اگرچه برای یک ISP با تعداد متوسطی از کاربران خانگی به خوبی کار می‌کند ولی برای ISPهایی که به مشتریان اداری خدمات می‌دهند مفید نیست.

مشکل از اینجا ناشی می‌شود که مشتریان اداری انتظار دارند که حداقل در ساعات کاری روز خطی دائم و فعال (On-Line) داشته باشند. امروزه، چه دفاتر کوچک اداری مثل یک آژانس مسافرتی با سه کارمند و چه شرکت‌های بزرگ که دارای تعداد زیادی کامپیوتر و شبکه محلی هستند، نیاز به خط دائم و فعال دارند. برخی از این کامپیوترها، PC کارمندان و برخی دیگر مثلاً سرورهای وب هستند. عموماً در هر LAN یک مسیریاب وجود دارد که از طریق یک خط اجاره‌ای (Leased) به ISP متصل شده است. چنین ساختاری متضمن آن است هر کامپیوتر آدرس IP خود را داشته باشد و به طور روزانه تغییر نکند. در نتیجه، تعداد کل کامپیوترهایی که در اختیار مشتریان اداری است نباید از تعداد آدرسهای IP متعلق به ISP بیشتر شود. برای آدرس 16/، حداکثر تعداد کامپیوترها ۶۵۵۳۴ است. برای یک ISP با دهها هزار مشتری اداری، این فضا سریعاً اشباع می‌شود.

آنچه که مشکل را حادتر می‌کند آن است که روز به روز بر تعداد مشترکین اینترنت از طریق مودمهای کابلی ADSL افزوده می‌شود. ویژگی چنین سرویسی عبارت است از: (۱) کاربر یک آدرس IP دائم و ثابت می‌گیرد. (۲) هزینه شماره‌گیری و اتصال ندارد (مگر یک هزینه ثابت ماهانه) بدین ترتیب اینگونه کاربران همیشه در شبکه حضور دارند. این موضوع، مشکل کمبود آدرسهای IP را افزایش می‌دهد. در اینجا تخصیص موقت آدرسهای IP (شیبه به مکانیزمی که برای کاربران تلفنی داشتیم) عملی نیست.

حتی از این هم پیچیده‌تر آنکه ممکن است کاربران ADSL و اینترنت کابلی دارای دو یا چند کامپیوتر در خانه باشند و تمام اعضای خانواده از طریق همین خط فعال و مشترک به ISP متصل شوند. یک راه حل آن است که تمام PCها از طریق یک LAN به هم متصل شده و با یک مسیریاب به ISP وصل شوند. از دیدگاه ISP شبکه این خانواده فرقی با یک دفتر اداری کوچک ندارد.

مشکل کمبود آدرسهای IP یک مسئله تئوریک نیست که در آینده‌ای دور رخ بدهد. همین الان با این مشکل مواجه هستیم. راه حل طولانی مدت و همیشگی این مشکل آنست که به سوی IPv6 (که آدرسهای آن ۱۲۸ بیتی

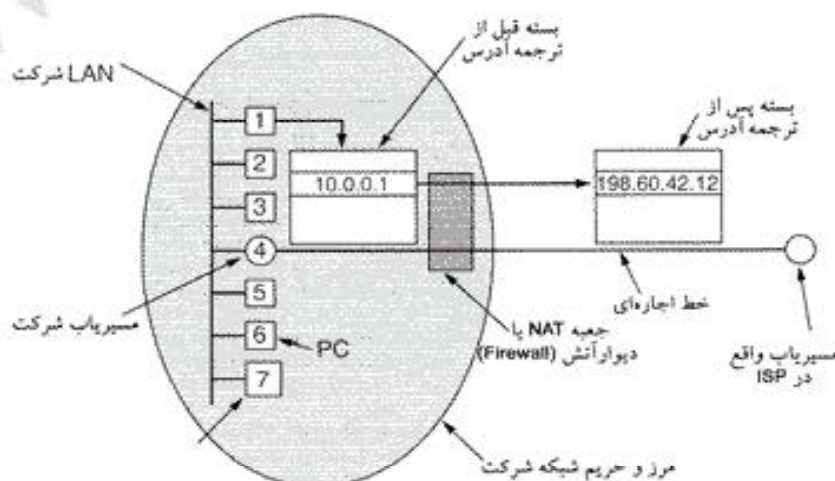
است) حرکت نماییم ولیکن گذار از نسخه ۴ به نسخه ۶ به آهستگی صورت می گیرد و سالها طول می کشد تا این تغییر به طور کامل انجام شود. در نتیجه بسیاری افراد احساس کردند که به یک راه حل سریع و کوتاه مدت نیاز است. این راه حل سریع، NAT (ترجمه آدرس شبکه) است که در RFC 3022 تشریح شده و در ادامه آنرا مختصراً بررسی می نماییم. برای کسب آگاهی بیشتر به مرجع (Dutcher, 2001) مراجعه نمایید.

ایده اصلی در NAT آن است که به هر شرکت یک یا تعداد کمی آدرس IP معتبر و جهانی اختصاص بدهیم. درون این شرکت، هر کامپیوتر دارای یک آدرس IP یکتا است که برای مسیریابی ترافیک داخلی بکار می آید. با این حال وقتی بسته ای بخواهد شرکت را ترک کرده و به ISP برود باید قبل از خروج ترجمه آدرس صورت بگیرد. برای آنکه این روش ممکن باشد سه محدوده از فضای آدرس IP جهت بکارگیری در شبکه های داخلی، به صورت «خصوصی» (Private) تعریف شده است و شرکتها می توانند به صورت دلخواه از آنها استفاده کنند. [نیازی به ثبت جهانی آنها نیست.] تنها قانون آن است که هیچ بسته ای نباید با چنین آدرسی بر روی اینترنت ظاهر شود. این سه محدوده رزرو شده عبارتند از:

10.0.0.0	- 10.255.255.255/8	(16,777,216 Hosts)
172.16.0.0	- 172.31.255.255/12	(1,048,576 Hosts)
192.168.0.0	- 192.168.255.255/16	(65535 Hosts)

در محدوده اول ۱۶۷۷۷۲۱۶ آدرس (به استثنای ۰ و ۱-) در دسترس است و عموماً اکثر شرکتها از آن استفاده می کنند هر چند نیازی به چنین تعداد آدرسی نداشته باشند.

عملکرد NAT در شکل ۵-۶ نشان داده شده است. ماشینها در درون شبکه دارای یک آدرس یکتا به فرم 10.x.y.z هستند. ولیکن وقتی بسته ای بخواهد مرز شرکت را ترک کند ابتدا باید از درون یک «جعبه NAT» (NAT BOX) عبور کرده و آدرس مبدا آن با آدرس IP حقیقی شرکت جانشین شود. مثلاً در شکل ۵-۶ آدرس 10.0.0.1 با آدرس 198.60.42.1 عوض شده است. اغلب «جعبه NAT» در یک «دیوار آتش» (Firewall) ادغام می شود تا این ابزار ضمن ترجمه آدرس، امنیت شبکه را نیز با نظارت دقیق بر ورود و خروج اطلاعات تضمین نماید. در فصل ۸ مفهوم «دیوار آتش» را بررسی خواهیم کرد. همچنین می توان «جعبه NAT» را در مسیریاب شرکت قرار داد. اکثر مسیریابهای امروزی از فرآیند NAT پشتیبانی می کنند.



شکل ۵-۶. مکان و عملکرد «جعبه NAT»

تا اینجا جزئیات کمی از فرآیند NAT مطرح کرده ایم: وقتی پاسخ یک بسته بر می گردد (مثلاً از سرویس دهنده وب) طبقاً آدرس ماشین گیرنده پاسخ، 192.60.42.1 است. سؤال این است که جعبه NAT از کجا بداند که آدرس کدام ماشین داخلی را به جای آن قرار بدهد؟ مسئله اصلی در NAT همین نکته است. اگر فیلدی اضافی در سرآیند بسته IP وجود داشت می شد از آن برای درج آدرس واقعی گیرنده بسته بهره گرفت ولیکن در سرآیند بسته تنها یک بیت بلااستفاده مانده است. همچنین می توان یک گزینه جدید [در فضای فیلد اختیاری Option] تعریف کرد تا آدرس حقیقی ماشین مبدا بسته را نگاه دارد ولی انجام این کار مستلزم آن است که کُد نرم افزار IP در تمام ماشینها و در کل اینترنت تغییر کند تا گزینه جدید به رسمیت شناخته شده و به درستی تعبیر شود. این راه حل نیز فرآیند زمان بری است و مشکل را در کوتاه مدت حل نخواهد کرد.

آنچه که بطور واقعی اتفاق می افتد به نحو ذیل است: طراحان NAT بدین نتیجه رسیده بودند که اغلب بسته های IP در درون فیلد داده خود یک بسته TCP یا UDP حمل می کنند. هرگاه در فصل ششم TCP و UDP را بررسی کردیم، خواهید دید که هر دوی این پروتکلها دارای سرآیندی برای بسته های خود هستند که دو فیلد «شماره پورت مبدا» و «شماره پورت مقصد» جزو آنهاست. در زیر اگرچه تمرکز ما بر پورتهای TCP است ولی همین قضیه برای پورتهای UDP نیز صادق است. این پورتهای که اعداد صحیح ۱۶ بیتی هستند، مشخص می کنند که اتصال TCP از چه پروسه ای شروع و به چه پروسه ای ختم می شود. این شماره پورتهای فیلدهای لازم برای عملکرد NAT را فراهم آورده اند.

هرگاه یک پروسه بخواهد یک اتصال TCP با یک پروسه راه دور برقرار کند یک شماره پورت بلااستفاده برای خود بر میگزیند. این پورت، اصطلاحاً «پورت مبدا» نام دارد و به کد برنامه TCP تفهیم می کند که باید بسته های ورودی با این شماره پورت را برای او بفرستد. همچنین هر پروسه یک شماره پورت مقصد تعیین می کند تا مشخص شود که بسته ها باید به کدام پروسه در ماشین مقصد تحویل شود. شماره پورتهای صفر تا ۱۰۲۳ برای سرویس دهنده های مشهور رزرو شده است. به عنوان مثال پورت شماره ۸۰ توسط سرویس دهنده های وب بکار گرفته شده است، لذا برنامه های مشتری (Client) براحتی با آنها ایجاد ارتباط می کنند. کوتاه سخن آنکه، هر پیام خروجی از TCP دارای شماره پورت مبدا و شماره پورت مقصد است و این دو شماره پورت هویت پروسه های طرفین ارتباط را مشخص می نمایند.

تمثیلی از یک نمونه می تواند به فهم شماره های پورت کمک کند: یک شرکت را در نظر بگیرید که دارای یک شماره تلفن اصلی و واحد است. وقتی افراد با این شماره تماس می گیرند بلافاصله اپراتور مربوطه از آنها سؤال می کند که کدام شماره داخلی مدنظر آنهاست؛ سپس خط داخلی را وصل می کند. شماره اصلی به مثابه آدرس IP است و شماره های داخلی، مشابه با شماره پورت هستند. پورتهای ۱۶ بیت آدرس اضافی دیگر هستند که هویت پروسه گیرنده بسته ها را مشخص می نمایند.

با استفاده از فیلد «شماره پورت مبدا» می توان مشکل نگاشت آدرسها در NAT را حل کرد هرگاه بسته ای برای خروج از شبکه به NAT وارد شود، آدرس مبدا آن که به شکل 10.x.y.z است با آدرس IP حقیقی و معتبر شرکت عوض می شود. مضاف بر این فیلد شماره پورت مبدا (TCP Source Port) با عددی عوض می شود که در حقیقت این عدد اندیس جدول ترجمه آدرس در جعبه NAT است. هر یک از درایه های این جدول، آدرس IP اصلی و همچنین شماره پورت واقعی آن بسته را نگه می دارند. در آخر کد کشف خطای بسته TCP و بسته IP از نو محاسبه و در بسته قرار داده می شود. [چرا که هم فیلد شماره پورت و هم فیلد آدرس IP مبدا در جعبه NAT عوض می شود. -م] عوض کردن مقدار فیلد پورت مبدا (Source Port) الزامی است چرا که ممکن است بطور همزمان از دو ماشین به آدرسهای 10.0.0.1 و 10.0.0.2 یک اتصال TCP اتفاقاً با شماره پورت مبدا یکسان (مثلاً

۵۰۰۰) ایجاد شود، لذا شماره پورت مبدا نمی تواند هویت واقعی پروسه ارسال کننده بسته ها را مشخص کند.^۱ وقتی بسته ای از طریق ISP به جعبه NAT وارد می شود ابتدا مقدار فیلد پورت مبدا استخراج شده و از آن به عنوان اندیس جدول نگاشت در جعبه NAT استفاده می شود. پس از پیدا شدن درایه متناظر، آدرس IP داخلی و شماره اصلی پورت مبدا بسته استخراج شده و در درون بسته قرار می گیرد. سپس این بسته از طریق مسیر یاب داخلی شرکت، برای تحویل به آدرس 10.x.y.z مسیر طبیعی خود را طی می کند.

همچنین از NAT می توان برای تخفیف مشکل کمبود آدرس IP برای کاربران ADSL و کاربران کابلی بهره گرفت. هر گاه ISP بخواهد به هر یک از این کاربران آدرسی انتساب بدهد، از آدرسی در فضای 10.x.y.z بهره می گیرد. قبل از آنکه بسته های ماشین کاربران، ISP را ترک کنند و به اینترنت وارد شوند باید ابتدا وارد جعبه NAT شده و آدرس غیرحقیقی و محلی آنها به آدرس واقعی متعلق به ISP نگاشته شود. در مسیر برگشت، عکس فرایند نگاشت انجام می شود. بدین نحو از دیدگاه اینترنت، این ISP (و کاربران ADSL یا کابلی آن) دقیقاً مثل یک شرکت بزرگ به نظر می رسند، هر چند تعداد آدرسهای واقعی و معتبر ISP ناچیز است.

اگرچه این روش مشکل کمبود آدرسهای IP را حل می کند ولیکن بسیاری از افراد در جامعه اینترنت از آن به عنوان کاری بی ارزش و مردود یاد می کنند. برخی از مخالفت های آنان را به اختصار ارائه می نماییم. اول آنکه NAT مدل معماری IP را نقض می کند چراکه در این مدل بیان شده که آدرس IP به صورت یکتا ماشینی واحد را در کل جهان مشخص می نماید. ساختار تمام نرم افزارهای اینترنت با تکیه بر این واقعیت بنیان گذاشته شده است. با NAT ممکن است هزاران ماشین از آدرس 10.0.0.1 استفاده کنند (و می کنند).

دوم آنکه NAT اینترنت را از حالت «بدون اتصال» به شبکه ای «اتصال گرا» تبدیل می نماید. مسئله اینجاست که جعبه NAT باید اطلاعاتی را در خصوص نگاشت اتصالهایی که از آن می گذرند در خود نگاه دارد. نگهداری وضعیت هر اتصال ویژگی شبکه های اتصال گراست و سختی با شبکه های بدون اتصال ندارد. اگر جعبه NAT به ناگاه از کار بیفتد و جدول نگاشت آن از دست برود تمام اتصالات TCP برقرار شده از دست می رود. بدین ترتیب با وجود NAT، اینترنت به یک شبکه آسیب پذیر مدار مجازی تبدیل می شود.

سوم آنکه، NAT اصول بنیانی لایه بندی پروتکلها را نقض می کند: لایه k نباید هیچ تصویری از آنچه که لایه k+1 در فیلد حمل داده از بسته او قرار می دهد، داشته باشد یا در آن دخالتی کند.^۲ اصل اساسی در معماری لایه به لایه آنست که تمام لایه ها مستقل از دیگری باشند. اگر مثلاً زمانی TCP به نسخه 2-TCP ارتقاء یابد و سرآیند بسته ها تغییر کنند (مثلاً شماره پورتها ۳۲ بیتی شوند)، NAT از کار خواهد افتاد. ایده اصلی در پروتکل های لایه ای آن بوده که تغییر در یک لایه نیازی به تغییر در لایه های دیگر نداشته باشد در حالیکه NAT این عدم وابستگی را از بین می برد.

چهارم آنکه پروسه های اینترنت مجبور به استفاده از TCP یا UDP نیستند. اگر فرضاً کاربری بر روی ماشین A تصمیم بگیرد برای محاوره و مبادله داده با کاربری بر روی ماشین B از یک پروتکل جدید در لایه انتقال استفاده کند (مثلاً برای کاربردهای چند رسانه ای)، وجود NAT منجر به عدم کارکرد آن برنامه کاربردی خواهد شد چراکه NAT نخواهد توانست فیلد پورت مبدا را به درستی پیدا کرده و از آن استفاده نماید.

پنجم آنکه برخی از برنامه های کاربردی آدرس IP ماشین خود را در متن اطلاعات ارسالی قرار می دهند. گیرنده نیز این آدرس را استخراج کرده و از آن در جایی استفاده می کند. از آنجایی که NAT چیزی در مورد این

۱. بعبارت روشنتر چون تمام بسته ها در برگشت آدرس IP یکسانی دارند فلذا این آدرس پورت هر بسته است که هویت گیرنده واقعی بسته را مشخص می کند و طبیعتاً NAT باید در هنگام خروج بسته ها ضمن عوض کردن شماره پورت، یکتا بودن آنها تضمین کند. - م

۲. بعبارتی از دیدگاه لایه k آنچه که از لایه k+1 می رسد تعدادی بابت خام است. - م

آدرسهای مخفی نمی داند فلذا قادر به تغییر آنها نبوده و هر گونه تلاش برای استفاده از این آدرسهای ناصحیح (در ماشین راه دور) با شکست مواجه می شود. FTP، یعنی استاندارد انتقال فایل در اینترنت به همین ترتیب عمل می کند و با وجود NAT از کار می افتد مگر آنکه اقدامات احتیاطی خاصی به عمل آید. به دلیل مشابه، پروتکل H.323 که برای تلفن اینترنتی کاربرد دارد (و در فصل هفتم به معرفی آن خواهیم پرداخت) با وجود NAT کار نخواهد کرد. البته می توان با تغییرات اصلاحی در NAT آن را بکار گرفت ولی این که با معرفی هر برنامه کاربردی جدید مجبور به اصلاح NAT شویم، اصلاً ایده جالبی نیست.

ششم آنکه چون فیلد آدرس پورت مبدا، ۱۶ بیتی است، حداکثر ۶۵۵۳۶ ماشین را می توان به یک آدرس IP واحد نگاشت. این تعداد حقیقتاً مقدار کمی است (گذشته از آن، ۴۰۹۶ شماره پورت نیز برای کاربردهای خاص کنار گذاشته شده اند)، ولیکن اگر تعداد آدرسهای IP معتبر و جهانی که در اختیار ISP قرار دارد، بیش از یکی باشد به ازای هر یک می توان ۶۱۴۴۰ ماشین را با آدرسهای غیر حقیقی مدیریت کرد.

این مشکلات به همراه مسائل دیگر NAT، در RFC 2993 تشریح شده است. عموماً مخالفین NAT می گویند که با حل نازیا و موقتی مسئله کمبود آدرسهای IP، اصرار بر روی راه حل واقعی و نهایی که همانا رفتن به طرف IPv6 است، کم می شود و تعویق انداختن در این تغییر و تحول اصلاً خوب نیست!

۳-۶-۵ پروتکلای کنترل اینترنت

اینترنت مضاف بر IP که برای انتقال داده ها کاربرد دارد، چندین پروتکل کنترلی دیگر دارد که همگی در لایه شبکه به کار گرفته می شوند. این پروتکلها عبارتند از: ICMP، ARP، RARP، BOOTP و DHCP. در این بخش، این پروتکلها را به ترتیب مرور می کنیم.

ICMP: پروتکل ارسال پیامهای کنترلی در اینترنت

عملکرد اینترنت توسط مسیریابها به صورت دقیق نظارت و کنترل می شود. هر گاه اتفاق غیرمنتظره ای بیفتد، رخداد مزبور توسط پروتکل ICMP گزارش می شود. این پروتکل همچنین برای آزمایش و رفع عیب شبکه کاربرد دارد. تقریباً یک دوجین از پیامهای ICMP به منظور اطلاع رسانی تعریف شده است. مهمترین این پیامها در شکل ۶۱-۵ فهرست شده اند. هر پیام ICMP مستقیماً درون یک بسته IP جاسازی و ارسال می شود.

توصیف عملکرد	نوع پیام
بهر دلیلی بسته را نمی توان به مقصد تحویل داد.	Destination unreachable
زمان حیات بسته به صفر رسیده است.	Time exceeded
فیلدی از سرآیند بسته مقدار نامعتبر داشته است.	Parameter problem
بسته دعوت به آرامش	Source quench
حارای اطلاعاتی در خصوص جغرافیای مسیر و اعلام اشتباه در مسیریابی	Redirect
درخواست از یک ماشین تا اگر فعال است پاسخ بدهد.	Echo
پاسخ به پیام Echo بمنظور تایید فعالیت	Echo reply
همانند پیام Echo به همراه مهر زمان	Timestamp request
همانند پیام Echo Reply به همراه مهر زمان	Timestamp reply

شکل ۶۱-۵. انواع پیامهای اساسی ICMP.

از پیام DESTINATION UNREACHABLE زمانی استفاده می‌شود که زیرشبکه یا مسیر یاب نتواند موقعیت مقصد را تشخیص بدهد یا وقتی که بیت DF^۱ در بسته‌ای به ۱ تنظیم شده باشد و آن بسته مجبور به عبور از شبکه‌ای باشد که طول بسته‌های آن کوچک است. [طبعاً بسته حذف می‌شود].

پیام TIME EXCEEDED زمانی ارسال می‌شود که بسته به دلیل صفر شدن شمارنده TTL (شمارنده زمان حیات) حذف گردد. این رخداد نشانه آن است که بسته‌ها در حلقه بی‌نهایت افتاده‌اند یا آنکه ازدحام سنگینی رخ داده یا آنکه مقدار اولیه این شمارنده بسیار کم بوده است.

پیام PARAMETER PROBLEM مشخص‌کننده آنست که در سرآیند بسته IP مقدار نامعتبر و اشتباهی درج شده است. [لذا مسیر یاب آن را حذف کرده و این پیام را ارسال نموده است.] این مشکل از وجود یک اشکال در نرم‌افزار IP ماشین فرستنده یا احتمالاً نرم‌افزار مسیر یابهای میانی پرده بر می‌دارد.

پیام SOURCE QUENCH سابقاً برای آن بکار می‌رفته تا به ماشینهایی که بیش از اندازه بسته ارسال می‌کنند اخطار داده شود. هر ماشین که این پیام را دریافت کند باید نرخ ارسال بسته‌های خود را کاهش بدهد. ولیکن از این پیام به ندرت استفاده می‌شود چرا که وقتی ازدحام رخ بدهد، ارسال چنین پیامهایی بر هیزم این آتش می‌افزاید. کنترل ازدحام در اینترنت عمدتاً در لایه انتقال انجام می‌شود. این روش را در فصل ششم خواهیم آموخت.

از پیام REDIRECT زمانی استفاده می‌شود که مسیر یاب احساس کند بسته‌ای در مسیر غلط قرار گرفته است. این پیام برای گزارش خطای احتمالی به ماشین مبدا بسته، کاربرد دارد.

پیامهای ECHO و ECHO REPLY بدین منظور کاربرد دارند که ببینیم آیا یک ماشین مقصد در دسترس و فعال است یا خیر. انتظار می‌رود هر ماشین که پیام ECHO دریافت می‌کند، در پاسخ به آن، پیام ECHO REPLY را برگرداند.

پیامهای TIMESTAMP REQUEST و TIMESTAMP REPLY مشابه با دو پیام قبلی هستند با این تفاوت که در بدنه پاسخ، زمان دریافت پیام و همچنین زمان ارسال پیام درج می‌شود. از این قابلیت می‌توان برای اندازه‌گیری کارایی شبکه بهره گرفت.

مضاف بر پیامهای فوق‌الذکر، پیامهای دیگری نیز تعریف شده‌اند که می‌توانید فهرست آنها را در آدرس زیر بیابید: www.iana.org/assignments/icmp-parameters

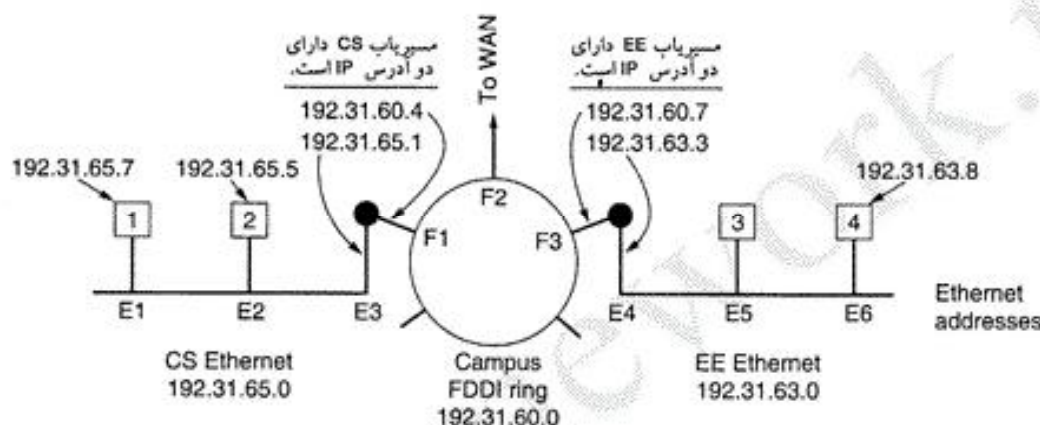
۲. ARP: پروتکل تحلیل آدرس

اگرچه هر ماشین در اینترنت دارای یک (یا چند) آدرس IP است، ولیکن این آدرسها حقیقتاً نمی‌توانند برای ارسال بسته‌ها بکار گرفته شوند چرا که سخت‌افزار لایه پیوند داده‌ها هیچ درکی از آدرسهای IP ندارد. امروزه اغلب ماشینهای میزبان در شرکتها یا دانشگاهها به کمک یک کارت واسط شبکه به LAN متصل شده‌اند و این کارتها فقط آدرسهای MAC (آدرس سخت‌افزاری کارت شبکه) را می‌شناسند. به عنوان مثال هر کارت شبکه اترنت پس از تولید در کارخانه، دارای یک آدرس ۴۸ بیتی یکتا است. سازندگان کارتهای اترنت برای تعبیه آدرس در کارتهای تولیدی خود ابتدا از یک مرکز مجاز تقاضا می‌کنند که یک بلوک آدرس به آنها اختصاص بدهد و بدین ترتیب اطمینان حاصل می‌شود که هیچ دو کارت شبکه در کل جهان دارای آدرس یکسان نیست. (تا از هرگونه برخورد دو کارت شبکه با شماره‌های یکسان بر روی LAN اجتناب شود). کارتهای شبکه، فریمها را براساس آدرسهای ۴۸ بیتی اترنت، ارسال یا دریافت می‌نمایند و هیچ چیزی در خصوص آدرسهای IP نمی‌دانند. اکنون این سؤال پیش می‌آید که چگونه آدرسهای IP به آدرسهای لایه پیوند داده (مثل آدرسهای اترنت)

۱. Don't Fragment

۲. Address Resolution Protocol

نگاشته و تبدیل می شوند؟ برای تشریح عملیات نگاشت آدرس از مثال شکل ۵-۶۲ استفاده کرده ایم. در این شکل یک دانشگاه کوچک با شبکه ای با چند کلاس C (به عبارتی شبکه ۲۴/۰) به تصویر کشیده شده است. در این شکل دو شبکه اترنت وجود دارد که یکی از آنها با آدرس ۱۹۲.۳۱.۶۵.۰ به دانشکده کامپیوتر و دیگری با آدرس ۱۹۲.۳۱.۶۳.۰ به دانشکده برق تعلق دارد. این دو شبکه از طریق یک شبکه حلقه (مثل FDDI) با آدرس ۱۹۲.۳۱.۶۰.۰ که نقش ستون فقرات شبکه کل دانشگاه را ایفاء می کند، به یکدیگر متصل شده اند. هر ماشین متصل به اترنت دارای یک آدرس MAC یکتا است که در شکل با برجسبهای E1 تا E6 مشخص شده اند. هر ماشین متصل به شبکه حلقه نیز آدرس FDDI خود را دارد که آنها نیز با F1 تا F3 نشان داده شده اند.



شکل ۵-۶۲. سه شبکه بهم متصل با الگوی ۲۴/۰: دو شبکه اترنت و یک شبکه حلقه از نوع FDDI.

حال ابتدا بررسی کنیم که کاربری بر روی ماشین میزبان ۱ چگونه بسته ای را برای کاربری بر روی ماشین ۲ می فرستد. فرض را بر آن گذاشته ایم که فرستنده، نام گیرنده مورد نظر را می داند و مثلاً این نام mary@eagle.cs.uni.edu است. در اولین گام باید آدرس IP ماشین eagle.cs.uni.edu را بدست بیاورد. این جستجو از طریق «سیستم نام دامنه» (DNS) که در فصل هفتم آن را مطالعه خواهیم کرد، انجام می گیرد. در اینجا فرض می کنیم که DNS، آدرس IP ماشین ۲ را ۱۹۲.۳۱.۶۵.۵ برگردانده است.

در اینجا نرم افزار لایه بالاتر در ماشین ۱ بسته ای می سازد و درون فیلد «آدرس مقصد» آن مقدار ۱۹۲.۳۱.۶۵.۵ را درج می کند و آن را جهت ارسال تحویل نرم افزار IP می دهد. نرم افزار IP با بررسی آدرس بدین نتیجه می رسد که ماشین مقصد، در شبکه خودش قرار گرفته است^۱ ولی محتاج روشی است تا بتواند آدرس اترنت ماشین مقصد را پیدا کند. یک راه حل آنست که یک فایل پیکربندی در جایی از سیستم ذخیره شده باشد و از طریق آن آدرس IP به آدرسهای اترنت نگاشته شود. اگرچه این روش عملی است ولی برای موسساتی که هزاران ماشین دارند به روز نگه داشتن چنین فایلهایی، کاری بسیار وقتگیر و منشاء خطاهای سهوی است.

راه حل بهتر آنست که ماشین میزبان ۱ بسته ای را به صورت فراگیر (Broadcast) بر روی اترنت پخش کند و از همه سؤال کند که: «آدرس ۱۹۲.۳۱.۶۵.۵ متعلق به کیست؟» این اعلام همگانی به یکایک ماشینهای شبکه اترنت با آدرس ۱۹۲.۳۱.۶۵.۰ خواهد رسید. تنها ماشین ۲ به این سؤال پاسخ خواهد داد و آدرس اترنت خود یعنی E2 را اعلام خواهد کرد. بدین طریق ماشین ۱ متوجه می شود که آدرس ۱۹۲.۳۱.۶۵.۵ متعلق به ماشینی است که آدرس

۱. نرم افزار IP از آنجا متوجه می شود که ماشین مقصد در شبکه محلی خودش واقع است که طبق الگوی ۲۴/۰ بخش شماره شبکه خودش و شماره شبکه مقصد یکسان است. (شماره شبکه هر دو ۱۹۲.۳۱.۶۵.۰ است.) - سم

اترنت آن E2 است. پروتکلی که برای این پرسش و پاسخ بکار می رود **ARP** نام دارد و تقریباً تمام ماشینهای اینترنت آنرا اجرا کرده اند. **ARP** در سند RFC 826 تبیین شده است.

مزیت استفاده از **ARP** به جای ذخیره فایل های پیکربندی، سادگی آنست. مدیر سیستم فقط باید برای هر ماشین، آدرس IP و الگوی زیرشبکه (Subnet Mask) آنرا مشخص کند. مابقی کارها را **ARP** انجام می دهد.

پس از بدست آمدن آدرس E2، نرم افزار IP در ماشین میزبان ۱، یک فریم اترنت به آدرس E2 ساخته و بسته IP (با آدرس 192.31.65.5) را در درون فیلد داده آن قرار داده و آن را بر روی اترنت روانه می کند. کارت شبکه ماشین ۲ این فریم را گرفته و تشخیص می دهد که متعلق به خود اوست؛ آنرا پذیرفته و یک «وقفه»^۱ تولید می کند. نرم افزار راه انداز اترنت، بسته IP را از درون فریم استخراج کرده و آنرا به نرم افزار IP تحویل می دهد. نرم افزار IP متوجه می شود که این بسته به آدرس صحیحی آمده و طبقاً آن را پردازش می نماید.

برای آنکه **ARP** کارآمدتر عمل کند می توان بهینه سازیهایی انجام داد. اولین بهینه سازی آنست که هرگاه **ARP** اجرا شد و آدرسی را بدست آورد، آن را در «حافظه نهان»^۲ خود ذخیره نماید تا در دفعات بعدی بتواند با استفاده از این آدرس، سریعاً با ماشین مربوطه ارتباط برقرار کند. دفعه بعد، **ARP** نگاشت آدرس IP به آدرس اترنت را در حافظه نهان خود خواهد یافت و نیاز مجدد به سؤال همگانی نیست. در بسیاری از حالات، ماشین ۲ (پس از دریافت بسته) باید پاسخی پس بفرستد و در نتیجه او نیز مجبور است برای یافتن آدرس اترنت فرستنده تقاضا، **ARP** را اجرا نماید. برای آنکه در اینجا از ارسال فراگیر بسته های **ARP** جلوگیری شود می توان بدین نحو عمل کرد که وقتی مثلاً ماشین ۱ برای یافتن آدرس ماشین ۲ به صورت پخش فراگیر از همه سؤال می کند، «نگاشت آدرس IP به آدرس اترنت» خود را نیز اعلام نماید. وقتی بسته های پخش **ARP** به ماشین ۲ می رسد جفت آدرس (E1 و 192.31.65.7) وارد حافظه نهان **ARP** می شود تا برای استفاده های آتی نیاز به سؤال نباشد. در حقیقت تمام ماشینهای متصل به اترنت می توانند این جفت آدرس نگاشته شده را در حافظه نهان **ARP** خود ذخیره نمایند.^۳

روشی دیگر برای بهینه سازی **ARP** آنست که هر ماشین به محض راه اندازی (بوت) «نگاشت آدرس سخت افزاری» به آدرس IP خود را به صورت پخش فراگیر به همه اعلام نماید.^۴ عموماً این کار بدین نحو انجام می گیرد که ماشین در حال بوت، آدرس سخت افزاری معادل با IP خودش را سؤال می کند! طبقاً چنین سؤالی هیچ پاسخی ندارد ولی نتیجه جانبی آن این است که در اثر پخش این سؤال (که پاسخ خود را به همراه دارد) بر روی کل شبکه محلی، نگاشت آدرس او به صورت یک درایه (Entry) درون حافظه نهان **ARP** ماشینهای فعال درج می شود. اگر پاسخ چنین سؤالی به صورت غیرمنتظره دریافت گردد، مشخص می شود که دو ماشین دارای آدرس IP مشابه هستند. ماشین دوم باید این موضوع را به مدیر سیستم اطلاع داده و بوت نشود.

برای آنکه جدول نگاشت آدرس در **ARP** بتواند تغییر کند، درایه های موجود در حافظه نهان **ARP** باید پس از چند دقیقه اعتبار خود را از دست بدهند و با سؤال مجدد، نگاشت آدرسها از نو انجام شود چراکه مثلاً اگر یک کارت شبکه اترنت، خراب و با کارتی جدید عوض شود آدرس اترنت معادل با آدرس IP آن ماشین عوض می شود و اگر بقیه ماشینها بخواهند از جدول نگاشت قدیمی خود استفاده کنند این ماشین از دسترس دیگران

۱. Interrupt

۲. ARP Cache

۳. عبارت دیگر وقتی یک ماشین، آدرس سخت افزاری یک ماشین دیگر را سؤال می کند، جفت آدرس سخت افزاری و IP خود را نیز به همه اعلام می کند و از آن به بعد ماشینها نیازی به سؤال کردن ندارند. بدین نحو حافظه نهان **ARP** سریعاً پر شده و فرآیند پرسش و پاسخ بهبود یافته و اضافی انجام نخواهد شد. —

۴. واژه های آدرس MAC، آدرس فیزیکی، آدرس سخت افزاری، آدرس اترنت همگی معادلند و به آدرس تعبیه شده بر روی کارت شبکه (آدرس لایه پیوند داده ها) اشاره دارند. —

خارج می شود.

حال مجدداً به شکل ۵-۶۲ نگاهی بیندازید: هنگامی که ماشین ۱ بخواهد بسته ای برای ماشین ۴ (با آدرس 192.31.63.8) بفرستد با شکست مواجه می شود زیرا ماشین ۴ نمی تواند پخش فراگیر بسته های پرسش را دریافت کند (مسیریابها بسته های پخش فراگیر را به شبکه های دیگر هدایت نمی کنند). برای حل این مشکل دو راه وجود دارد: اول آنکه مسیریاب CS به گونه ای پیکربندی شود تا به تمام بسته های ARP که در مورد شبکه 192.31.63.0 (یا سایر شبکه های محلی) آدرسی را سؤال می کنند، جواب بدهد و آدرس سخت افزاری خودش را اعلام نماید. در این حالت، ماشین ۱ در جدول خود یک درایه به صورت (E3 و 192.31.63.8) درج می نماید و تمام ترافیک خود برای ماشین ۴ را برای مسیریاب محلی می فرستد. این راه حل اصطلاحاً Proxy ARP نامیده می شود. (ARP وکالتی) راه حل دوم آنست ماشین ۱ فوراً تشخیص بدهد که مقصد مورد نظر او بر روی شبکه ای دیگر قرار گرفته و تمام ترافیک راه دور و غیر محلی خود را به آدرس اترنت مسیریاب پیش فرض که در این مثال E3 است بفرستد. [تشخیص محلی یا غیر محلی بودن آدرسهای IP به کمک الگوی زیر شبکه Subnet Mask - میسر است.] در این راه حل نیازی نیست که مسیریاب CS بداند که به چه شبکه هایی سرویس می دهد.

در هر حال آنچه که در ادامه اتفاق می افتد آنست که: ماشین ۱ بسته IP را درون فیلد حمل داده از فریم اترنت جاسازی کرده و مقصد فریم را آدرس اترنت E3 قرار می دهد. وقتی مسیریاب CS این فریم اترنت را دریافت می کند و بسته IP را از درون آن جدا کرده و آدرس IP آن را درون جدول مسیریابی خود جستجو می نماید و بر این اساس متوجه می شود که تمام بسته های متعلق به شبکه 192.31.63.0 باید به مسیریاب با آدرس 192.31.60.7 تحویل شود و اگر از قبل آدرس سخت افزاری کارت FDDI متناظر با 192.31.60.7 را نداند، با پخش فراگیر یک بسته ARP بر روی حلقه، آن را سؤال کرده و متوجه می شود که این آدرس F3 است. لذا بسته IP را درون فیلد حمل داده از فریم FDDI قرار داده و با درج آدرس F3 در فریم جدید آن را بر روی حلقه قرار می دهد.

در مسیریاب EE، نرم افزار راه انداز کارت FDDI، بسته را از درون فریم استخراج کرده و آن را به نرم افزار IP تحویل می دهد و IP نیز به روش مشابه متوجه می شود که باید این بسته را به آدرس 192.31.63.8 بفرستد. اگر این آدرس IP درون حافظه نهان ARP پیدا نشد، مجدداً با پخش فراگیر یک بسته ARP بر روی شبکه اترنت EE، آن را سؤال می کند و متوجه می شود که آدرس سخت افزاری مقصد E6 است، لذا یک فریم اترنت به آدرس E6 ساخته و بسته را مجدداً درون فیلد حمل داده آن قرار داده و فریم را بر روی شبکه اترنت EE قرار می دهد. وقتی فریم اترنت به ماشین ۴ می رسد، بسته از درون آن استخراج شده و جهت پردازش تحویل نرم افزار IP می شود. حرکت بسته ها از ماشین ۱ به یک شبکه دور در WAN نیز به روشی مشابه با روش قبلی انجام می شود با این تفاوت که در مثال بالا مسیریاب CS از طریق جدول مسیریابی خود متوجه می شود که باید بسته را برای مسیریاب متصل به WAN با آدرس F2 بفرستد.

پروتکل های RARP، BOOTP و DHCP

پروتکل ARP مسئله پیدا کردن آدرس اترنت متناظر با یک IP مشخص را حل می کند. [به عبارت ساده تر آدرس IP یک ماشین را گرفته و آدرس سخت افزاری آن ماشین را پیدا کرده، بر می گرداند.] برخی اوقات وارون این مسئله باید حل شود، یعنی با داشتن آدرس اترنت یک ماشین، به آدرس IP متناظر با آن نیاز است. بالاخص زمانی با این مسئله مواجه هستیم که یک ایستگاه بدون دیسک سخت بخواهد از طریق شبکه بوت شود. بطور معمول چنین ماشینی «تصویر باینری سیستم عامل» خود را از یک سرویس دهنده فایل تحویل گرفته و بارگذاری می کند. ولی چگونه می تواند آدرس IP خود را بفهمد؟

اولین راه حل ابداعی برای این مسئله آن بود که از پروتکل RARP^۱ (تشریح شده در سند RFC 903) استفاده شود. این پروتکل امکان آن را فراهم آورده که ماشین تازه بوت شده آدرس اترنت خود را به صورت فراگیر بر روی شبکه پخش کند و بگوید: «آدرس اترنت ۴۸ بیتی من مثلاً 14.04.05.18.01.25 است. آیا کسی آدرس IP مرا می داند؟» سرویس دهنده RARP این درخواست را می بیند و در فایل های پیکربندی خودش، آدرس اترنت اعلام شده را جستجو می نماید و آدرس IP متناظر با آن را بر می گرداند.

استفاده از RARP بهتر از آنست که آدرس IP یک ماشین در «تصویر حافظه»^۲ ارسالی جاسازی شود چراکه این امکان فراهم می آید که از «تصویر حافظه» مشابهی برای تمام ماشینها بهره گرفته شود. اگر قرار باشد آدرس IP درون تصویر حافظه ارسالی جاسازی شود، هر ایستگاه در شبکه به «تصویر» خاص خود احتیاج خواهد داشت. اشکال RARP آنست که بیهای فیلد آدرس مقصد را در تمام فریمهای ارسالی خود^۱ می گذارد تا بدین ترتیب این فریمها به صورت پخش فراگیر به سرویس دهنده RARP برسند ولیکن فریمهای پخش شده بر روی شبکه محلی، توسط مسیریابها به خارج از شبکه هدایت نمی شوند، فلذا بر روی هر شبکه محلی باید یک سرویس دهنده RARP وجود داشته باشد. برای حل این مشکل یک پروتکل خاص دیگر به نام BOOTP برای راه اندازی ایستگاههای بدون دیسک (Diskless) ابداع شده است. این پروتکل می تواند به غیر از آدرس IP ایستگاه بدون دیسک، اطلاعات اضافه تری را مثل آدرس IP سرویس دهنده فایل (که تصویر اجرایی سیستم عامل را در اختیار دارد)، آدرس IP مسیریاب پیش فرض، الگوی زیر شبکه (Subnet Mask)، به ایستگاهها ارائه بدهد. برخلاف RARP، در پروتکل BOOTP از بسته های UDP استفاده شده و مسیریابها این بسته ها را هدایت می نمایند. [لذا به ازای چندین شبکه محلی که از طریق مسیریاب بهم متصل شده اند فقط به یک سرویس دهنده BOOTP نیاز است.] پروتکل BOOTP در RFC های 951، 1048 و 1084 تشریح شده است.

مشکل جدی پروتکل BOOTP آنست که جدول نگاشت آدرس IP به آدرس اترنت باید به صورت دستی تنظیم و پیکربندی شود. وقتی ماشین جدیدی به LAN اضافه می شود قادر به بوت شدن نیست مگر آنکه مسئول شبکه یک آدرس IP به آن انتساب داده و آن را در قالب: (آدرس IP + آدرس اترنت) به صورت دستی در فایل پیکربندی BOOTP وارد نماید. برای آنکه این مرحله اشکالزا حذف شود پروتکل BOOTP پیشرفته تر شد و با نام جدید DHCP^۳ معرفی گردید. پروتکل DHCP این امکان را فراهم آورده که بتوان آدرس IP ایستگاهها را هم به صورت دستی و هم به صورت خودکار به آنها انتساب داد. این پروتکل در RFC های ۲۱۳۱ و ۲۱۳۲ تشریح شده است و در اغلب سیستمها جایگزین RARP و BOOTP شده است.

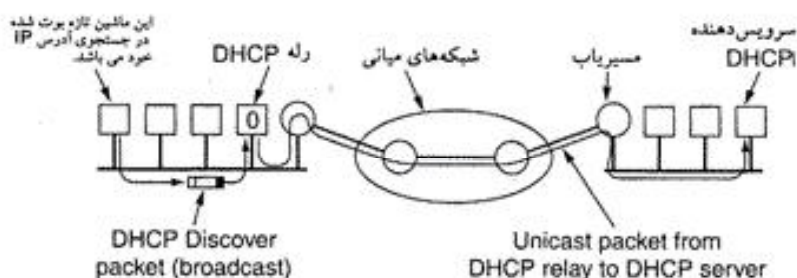
شبهه به RARP و BOOTP، پروتکل DHCP نیز متکی به یک سرویس دهنده ویژه در شبکه است تا به ماشینهایی که تقاضای آدرس IP می کنند، آدرس و اطلاعات لازم را تقدیم نماید. لازم نیست که این سرویس دهنده بر روی همان LAN باشد که ماشین متقاضی آدرس بر روی آن واقع شده است و از آنجایی که ممکن است دسترسی به این سرویس دهنده از طریق پخش فراگیر بسته های تقاضا میسر نباشد فلذا بر روی هر LAN به یک «عامل رله DHCP» (DHCP Relay Agent) نیاز است. (به شکل ۵-۶۳ نگاه کنید.)

ماشینی که تازه بوت شده برای بدست آوردن آدرس IP خود، بسته ای به نام DHCP DISCOVER را به صورت فراگیر بر روی LAN خود منتشر می کند. «عامل رله DHCP» در هر LAN تمام بسته های پخش شده

۱. Reverse Address Resolution Protocol

۲. منظور از تصویر حافظه یا Memory Image قطعه کد اجرایی و باینری از هسته سیستم عامل است که بلافاصله پس از بارگذاری، اجرا می شود. -م

۳. Dynamic Host Configuration Protocol



شکل ۵-۶۳. عملکرد DHCP.

DHCP را می گیرد و وقتی متوجه شود که یک بسته از نوع DHCP DISCOVER است آن را به صورت تک پخش (Unicast) برای سرویس دهنده اصلی DHCP (که می تواند بر روی شبکه ای دور واقع باشد) می فرستد. «عامل رله DHCP» فقط به آدرس IP از سرویس دهنده DHCP احتیاج دارد. یکی از مسائلی که در خصوص انتساب خودکار آدرسها پیش می آید آنست که یک آدرس IP برای چه مدت زمانی در اختیار ماشین قرار بگیرد، اگر ماشینی بدون اطلاع قبلی شبکه را ترک کند و آدرس IP خود را به سرویس دهنده DHCP برگرداند، آن آدرس برای همیشه گم می شود. به مرور زمان ممکن است آدرسهای زیادتری به این نحو گم شوند. برای آنکه چنین مشکلی اتفاق نیفتد، انتساب آدرسهای IP فقط برای مدت زمان محدود و ثابتی انجام می شود. این تکنیک «اجاره IP» (IP Leasing) نام دارد. هر ماشین قبل از آنکه مهلت اجاره آدرس IP او به سر برسد، باید با ارسال بسته ای خاص تقاضای تجدید اجاره کند. اگر ماشین موفق به ارسال این تقاضا نشد یا تقاضای تجدید پذیرفته نشود، ماشین میزبان نمی تواند بیش از این از آدرس IP اجاره ای خود استفاده نماید.^۱

۵-۶ OSPF: پروتکل مسیریابی برای دروازه های درونی

تا اینجا بررسی پروتکل های کنترل ایتترنت را به اتمام رسانده ایم. زمان آن فرا رسیده که به مورد بعدی بپردازیم: «مسیریابی در ایتترنت». قبلاً اشاره کردیم که شبکه ایتترنت از تعداد بسیار زیادی «سیستم خودمختار» (Autonomous System) تشکیل شده است. هر سیستم خودمختار که اصطلاحاً AS نامیده می شود توسط سازمان یا نهاد خاصی پیاده و اداره می شود و طبیعی است که آن مؤسسه برای مسیریابی بسته ها در درون شبکه، از الگوریتم مورد نظر خود بهره بگیرد. به عنوان مثال شبکه های داخلی سه شرکت X و Y و Z، از منظر ایتترنت در قالب سه AS دیده می شود. (البته به شرطی که به ایتترنت متصل شده باشند.) از دیدگاه ایتترنت، جزئیات درونی یک شبکه AS قابل رویت نیست. علیرغم آنکه جزئیات درونی هر AS مستقل از دیگری است ولیکن داشتن یک استاندارد برای مسیریابی در درون یک AS پیاده سازی «مرز»^۲ AS ها را آسان می کند. در این بخش مسیریابی در درون یک AS و در بخش بعدی مسیریابی بین AS ها را مطالعه خواهیم کرد. مسیریابی در درون یک AS اصطلاحاً «پروتکل دروازه درونی»^۳ و الگوریتم مسیریابی بین AS ها «پروتکل دروازه خارجی»^۴ نامیده می شود.

۱. در DHCP مسئول شبکه تعدادی از آدرس IP یا محدوده ای از فضای آدرس IP مورد نظر خود را مشخص می کند تا این سرویس دهنده آنها را برحسب نیاز به ماشینهای شبکه اجاره دهد. به این آدرسها اصطلاحاً IP Pool گفته می شود. م.

۲. منظور از «مرز» نقطه ای است که AS ها به یکدیگر متصل می شوند و مسیریابهایی که این اتصال را برقرار می کنند مسیریابهای مرزی (Border Gateway) نامیده می شوند. این مسیریابها هم با درون AS و هم با دیگر AS ها در ارتباط فعال هستند؛ یعنی هم مسیرهای بهینه داخلی و هم مسیرهای بهینه بین AS ها را می دانند. م.

۳. Interior Gateway Protocol.

۴. Exterior Gateway Protocol.

اولین پروتکل «دروازه درونی اینترنت»، یک پروتکل بردار فاصله با نام RIP بود که از الگوریتم «پلمن-فورده» بهره می گرفت. [بخش ۵-۲-۴] اگرچه این پروتکل در سیستمهای کوچک به خوبی کار می کند ولی با بزرگ شدن ASها، کارایی خود را از دست می دهد. این پروتکل همچنین از مشکل «شمارش تابانی» رنج می برد و «همگرایی» کندی دارد. به همین دلیل در ماه می ۱۹۷۹ یک پروتکل مبتنی بر «حالت لینک» (Link State) جایگزین آن شد. در ۱۹۸۸، IETF کار را بر روی یک پروتکل جدید آغاز کرد. این پروتکل OSPF (Open Shortest Path First) نام گرفت و در سال ۱۹۹۰ استاندارد شد. امروزه اغلب تولیدکنندگان مسیر یاب از آن حمایت می کنند و تقریباً به مهمترین پروتکل مسیر یابی دروازه های درونی تبدیل شده است. در زیر شمائی از عملکرد OSPF ارائه می دهیم. برای آگاهی دقیقتر از کل قضیه به سند RFC 2328 مراجعه نمایید.

گروه طراح این پروتکل با داشتن تجربه طولانی از دیگر پروتکل های مسیر یابی، تصمیم گرفتند با تدوین فهرست بالابندی از نیازها و انتظارات، آنرا از نو طراحی کنند. نخستین نیاز آن بود که به صورت «باز» (Open) طراحی شود بدین معنا که امتیاز راه حلها و روشهای خاص آن برای موسسه خاصی ثبت نشود و همچنین به بستر سخت افزاری یا نرم افزاری ویژه وابسته نباشد. حرف 'O' در نام OSPF به همین مضمون است.

نیاز دوم آن بود که پروتکل جدید بتواند برای تعیین مسیر بهینه، از معیارهای گوناگون هزینه مثل معیار «فاصله فیزیکی»، «تاخیر» و نظایر آن پشتیبانی کند. نیاز سوم آن بود که الگوریتم پویا باشد و هر گونه تغییر در توپولوژی زیر شبکه را به سرعت و به صورت خودکار تشخیص داده و خود را با آن تطبیق بدهد.

نیاز چهارم آنکه OSPF می بایست بسته ها را بر حسب نوع خدمات درخواستی، مسیر یابی و هدایت نماید. این پروتکل باید قادر می بود که ترافیک داده های بی درنگ را نسبت به ترافیک داده های معمولی به روش متفاوتی مسیر یابی نماید. پروتکل IP در هر بسته یک فیلد به نام Type of Service تعریف کرده بود که هیچ پروتکل مسیر یابی از آن حمایت نمی کرد. اگرچه این فیلد در OSPF گنجانده شد ولیکن باز هم کسی از آن استقبال نکرد و عاقبت حذف گردید.

نیاز پنجم (که با موارد فوق مرتبط است) آن بود که پروتکل جدید مکانیزم «موازنه بار» (Load Balancing) را اعمال کند و بار را بر روی چندین خط تقسیم نماید. اغلب پروتکلها تمام بسته ها را بر روی بهترین مسیر ارسال می کنند و از مسیری که از لحاظ بهینگی در رتبه دو قرار می گیرد هیچ استفاده ای نمی شود؛ در بسیاری از حالات تقسیم بار بر روی چندین خط، کارایی بهتری دارد.

ششم آنکه نیاز بود از مسیر یابی سلسله مراتبی پشتیبانی شود. تا سال ۱۹۸۸ اینترنت آنقدر بزرگ شده بود که نمی شد انتظار داشت یک مسیر یاب بتواند توپولوژی کل آن را بداند. پروتکل جدید می بایست به گونه ای طراحی می شد که هیچ مسیر یابی نیاز به دانستن توپولوژی کل شبکه نداشته باشد.

هفتم آنکه به سطحی از امنیت نیاز بود تا یک دانشجوی بازیگوش نتواند با ارسال اطلاعات جعلی و نادرست در خصوص مسیرها، مسیر یاب را به اشتباه بیندازد. در آخر آنکه به تمهیداتی نیاز بود تا مسیر یابهایی که با مکانیزم «ایجاد تونل» (Tunneling) و از طریق اینترنت بهم متصل می شوند را به درستی مدیریت نماید.

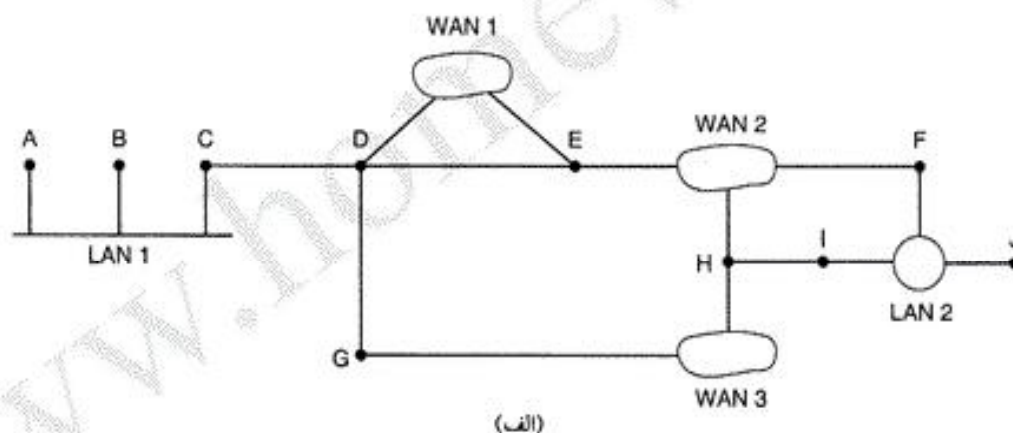
OSPF از سه نوع شبکه و خطوط اتصال پشتیبانی می کند:

۱. خطوط نقطه به نقطه بین دو مسیر یاب
۲. شبکه های با دسترسی چندگانه که کانال آنها از نوع پخش فراگیر (Broadcast) است. (مثل اغلب شبکه های LAN)
۳. شبکه های با دسترسی چندگانه بدون آنکه کانال آنها از نوع پخش فراگیر باشد. (مثل اغلب شبکه های سوئیچ بسته در WAN)

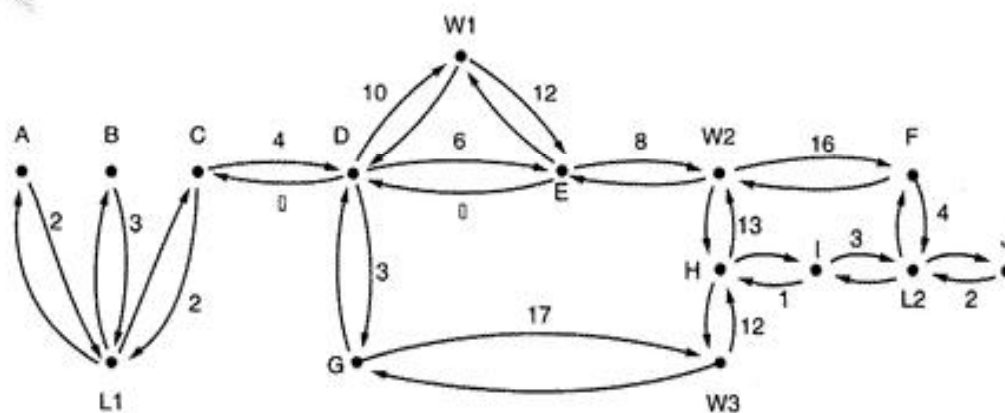
شبکه با دسترسی چندگانه (Multiaccess Network)، شبکه‌ای است که می‌تواند چندین مسیر یاب داشته باشد و هر یک از مسیر یابها می‌توانند مستقیماً با یکدیگر در ارتباط باشند. تمام شبکه‌های LAN و WAN دارای این ویژگی هستند. شکل ۵-۶۴-الف یک AS را نشان می‌دهد که در آن هر سه نوع شبکه وجود دارد. دقت کنید که از دیدگاه OSPF، ماشینهای میزبان شبکه عموماً نقش خاصی را ایفاء نمی‌کنند.

OSPF بدین نحو عمل می‌کند که مجموعه شبکه‌ها، مسیر یابها و خطوط ارتباطی را در قالب یک «گراف جهت‌دار» (Directed Graph) مدل می‌کند و به هر کمان در گراف (Arc) یک مقدار «هزینه» (مثل تأخیر، فاصله یا امثال آن) انتساب می‌دهد. سپس براساس وزن هر یک از کمانها، مسیر بهینه را پیدا می‌کند. یک خط ارتباطی سریال بین دو مسیر یاب، در گراف با یک جفت کمان (Arc) نشان داده می‌شود (یک کمان به ازای هر جهت) و وزنهای هر کمان می‌تواند با دیگری متفاوت باشد. یک شبکه با دسترسی چندگانه (LAN)، با یک گره به ازای خود شبکه و یک گره به ازای هر مسیر یاب مدل می‌شود. وزن کمانی که از گره شبکه به یک مسیر یاب وارد می‌شود، صفر در نظر گرفته شده و از گراف حذف می‌گردد.

شکل ۵-۶۴-ب، نمایش گراف متناظر با شبکه ۵-۶۴-الف را نشان می‌دهد. وزنهای متناظر هستند مگر آنهایی که به صراحت مشخص شده‌اند. آنچه که OSPF انجام می‌دهد مدل کردن شبکه در قالب گرافی شبیه به این مثال و سپس محاسبه مسیرهای بهینه از هر مسیر یاب به هر مسیر یاب دیگر است.



(الف)



(ب)

شکل ۵-۶۴. (الف) یک سیستم خودمختار (ب) نمایش گراف از شکل الف.

بسیاری از AS ها در اینترنت خودشان بسیار عظیم هستند و مدیریت آنها ساده نیست. OSPF این امکان را فراهم آورده که بتوان چنین شبکه هایی را به تعدادی «ناحیه شماره گذاری شده» (Numbered AREA) تقسیم کرد. هر ناحیه خود یک شبکه یا مجموعه ای از شبکه های بهم پیوسته مجاور است. نواحی با یکدیگر همپوشانی ندارند^۱ ولی لازم نیست که نواحی تعریف شده کل شبکه AS را پوشش بدهد و ممکن است برخی از مسیرهای در هیچ ناحیه ای قرار نگیرند. یک ناحیه، شکل کلی و عمومی یک زیرشبکه مستقل است و در خارج از ناحیه، توپولوژی و جزئیات درونی آن مشهود نیست.

در هر شبکه خودمختار (AS) ناحیه ای به نام «ستون فقرات» وجود دارد که ناحیه صفر نامیده می شود. تمام نواحی به ستون فقرات متصل می شوند (به صورت مستقیم یا توسط ایجاد تونل) لذا براحتی می توان به کمک ستون فقرات از هر ناحیه در شبکه AS به ناحیه دیگر رفت. یک «تونل» نیز در گراف توسط یک کمان مشخص می شود که دارای هزینه است. هر مسیریاب که به دو یا چند ناحیه متصل باشد (یعنی مسیرهای مشترک بین دو یا چند ناحیه) جزیی از ستون فقرات شبکه محسوب می شود. همانند بقیه نواحی، توپولوژی ناحیه ستون فقرات نیز در خارج از آن مشهود نیست. (بعبارتی مسیرهای درون دیگر نواحی از توپولوژی ستون فقرات چیزی نمی دانند.)

درون یک ناحیه، هر یک از مسیرهای نسخه مشابهی از پایگاه اطلاعاتی در خصوص مسیرها و هزینه ها در اختیار دارند و الگوریتم محاسبه کوتاهترین مسیر آنها یکسان است. هر مسیریاب وظیفه دارد که کوتاهترین مسیرها از خودش به تمام مسیرهای دیگر ناحیه را محاسبه نماید. از جمله هر مسیریاب باید کوتاهترین مسیر از خود تا یک مسیریاب واقع بر روی ستون فقرات را پیدا کند. یک مسیریاب که در مرز دو ناحیه واقع است باید پایگاه اطلاعاتی هر دو ناحیه را در اختیار داشته باشد و الگوریتم کوتاهترین مسیر را بطور جداگانه بر روی آنها اجرا کند. [تا تمام مسیرهای بهینه از خودش تا بقیه مسیرها در هر دو ناحیه بدست بیاید.]

در حین عملیات طبیعی، احتمالاً به سه نوع مسیر نیاز است: (۱) مسیرهای درون ناحیه^۲ (Intra-Area) (۲) مسیرهای بیرون ناحیه (Inter-Area) (۳) مسیرهای بین AS (Inter-AS).

مسیرهای درون ناحیه ساده ترین نوع مسیر هستند چراکه مسیریاب مبدا، از قبل و به دقت کوتاهترین مسیر رسیدن به مسیریاب مقصد را می داند. مسیریابی بین نواحی (Inter-Area)، همیشه در سه مرحله انجام می گیرد: حرکت از مبدا تا ستون فقرات، سپس حرکت از ستون فقرات به سوی ناحیه مقصد و نهایتاً حرکت در درون ناحیه به سمت مقصد. این الگوریتم ایجاب می کند که در پروتکل OSPF یک توپولوژی «ستاره» برای مسیرها قائل شویم: ستون فقرات در مرکز قرار می گیرد و بقیه نواحی از آن منشعب می شوند. در OSPF بسته ها به همان نحوی که از مبدا تولید شده اند به سوی مقصد مسیریابی و هدایت می شوند یعنی بسته ها در درون بسته دیگری «جاسازی» (Encapsulate) یا «تونل» (Tunnel) نمی شوند، مگر آنکه بسته مجبور باشد برای رسیدن از یک ناحیه به ستون فقرات، از مسیری حرکت کند که از طریق تونل ایجاد شده است.^۳ شکل ۵-۵ بخشی از اینترنت را با چهار AS و تعدادی ناحیه نشان می دهد.

OSPF چهار کلاس مسیریاب را به رسمیت می شناسد:

۱. مسیریابهای درونی (که کاملاً در داخل یک ناحیه قرار می گیرند).
۲. مسیریابهای واقع بر مرز دو ناحیه (که دو یا چند ناحیه را به هم متصل می کنند).

۱. یعنی هر مسیریاب صرفاً به یک ناحیه متعلق است. -م

۲. یعنی مسیریابی که از یک مسیریاب در درون ناحیه شروع و به یک مسیریاب در همان ناحیه ختم می شود. -م

۳. مفهوم تونل را در بخش ۵-۵-۵ مطالعه نمایید.

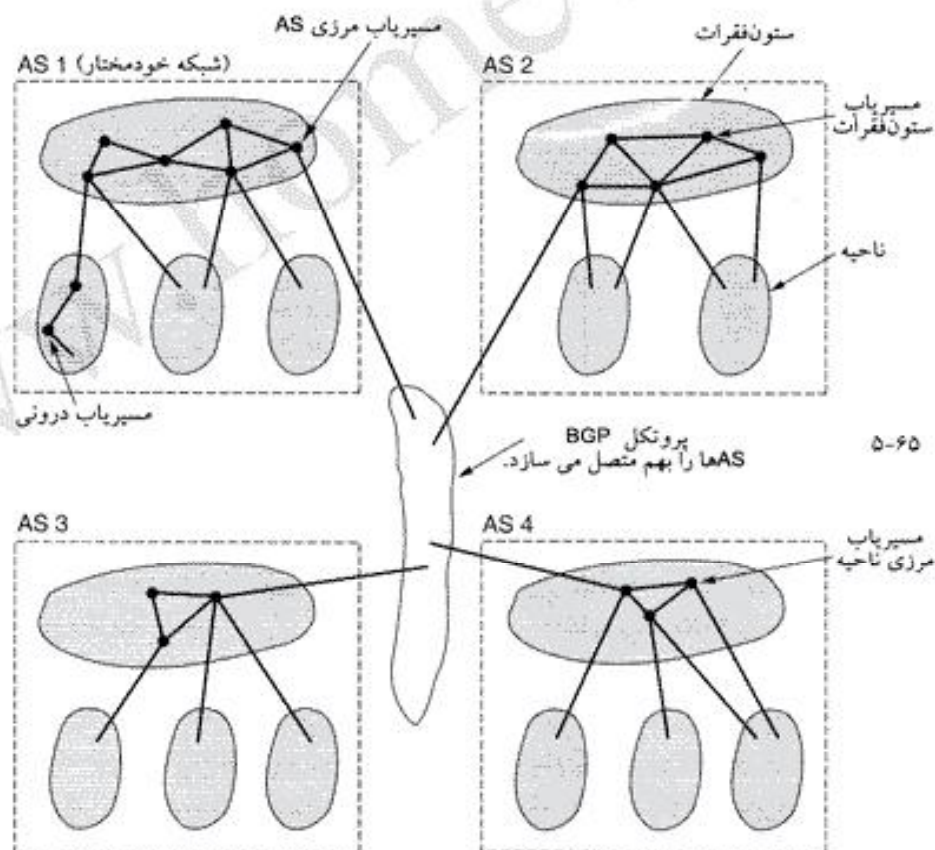
۳. مسیریابهای ستون فقرات (که بر روی ستون فقرات قرار می گیرند).

۴. مسیریابهای مرزی AS که می توانند با مسیریابهای AS دیگر محاوره کنند.

البته امکان آن وجود دارد که مسیریابها در دو یا چند کلاس قرار بگیرند. به عنوان مثال تمام مسیریابهای مرزی، به صورت خودکار جزئی از ستون فقرات نیز هستند. مضاف بر این، یک مسیریاب که بر روی ستون فقرات واقع است ولی در هیچ ناحیه ای قرار نگرفته، خودش یک مسیریاب داخلی محسوب می شود. در شکل ۵-۶۵ نمونه ای از تمام کلاسهای مسیریاب، دیده می شود.

وقتی یک مسیریاب بوت می شود، ابتدا بر روی تمام خطوط نقطه به نقطه و کانالهای اشتراکی (مثل کانالهای LAN که چند مسیریاب از طریق آن بهم متصل شده اند)، پیامی به نام «پیام سلام» (Hello Message) می فرستد. در خطوط WAN [مثل خطوط ISDN] احتمالاً قبل از هر گونه مبادله پیام به اطلاعات پیکربندی خاص نیاز است تا هویت تماس گیرنده مشخص شود.^۱ هر مسیریاب با دریافت پاسخ سلام، مسیریابهای همسایه خود را شناسایی می کند. مسیریابهایی که به یک LAN واحد متصلند، همگی همسایه یکدیگر محسوب می شوند.

عملکرد OSPF مبتنی بر مبادله اطلاعات با «مسیریابهای مجاور» (Adjacent) است. مفهوم مسیریابهای مجاور با مفهوم «مسیریابهای همسایه» (Neighbor) فرق می کند؛ اینکه هر مسیریاب متصل به LAN بتواند با مسیریاب دیگر محاوره و مبادله پیام کند کافی و کارآمد نیست. برای اجتناب از این وضعیت، از بین کل مسیریابهای



شکل ۵-۶۵. ارتباط بین AS ها، ستون فقرات و نواحی در OSPF.

واقع بر LAN، یک مسیریاب به نام «مسیریاب برگزیده» (Designated Router) انتخاب می گردد. این مسیریاب، «مسیریاب مجاور» بقیه مسیریابهای واقع بر LAN محسوب می شود و با آنها به مبادله اطلاعات می پردازد. برای «مسیریاب برگزیده» یک مسیریاب پشتیبان نیز در نظر گرفته شده که آن نیز همیشه اطلاعات به روز از وضعیت مسیره را در اختیار دارد تا در صورت از کار افتادن «مسیریاب برگزیده»، به سرعت جایگزین آن شود.

در حین عملکرد طبیعی، هر مسیریاب بطور متناوب پیامهای LINK STATE UPDATE خود را به صورت سیل آسا (Flooding) برای تمام مسیریابهای مجاور خود می فرستد. [این پیامها حاوی اطلاعاتی در خصوص هویت همسایه ها، لینکها و هزینه آنهاست.] بدین ترتیب، با این پیامها حالت هر لینک (Link State) و هزینه آن، برای درج در «پایگاه اطلاعات توپولوژیکی» به دست می آید. وصول پیامهایی که به صورت سیل آسا ارسال می شوند، تصدیق خواهد شد تا از دریافت آنها اطمینان حاصل شود.^۱ هر پیام یک شماره ترتیب دارد تا مسیریاب بفهمد آیا پیام LINK STATE UPDATE قدیمی یا جدید است. ارسال این پیامها به صورت سیل آسا زمانی آغاز می شود که خطی از کار بیفتد یا مجدداً فعال شود یا هزینه آنها تغییر کند. البته حتی اگر چنین اتفاقی نیفتد، این پیامها بطور متناوب و هر از چند ده ثانیه ارسال خواهند شد.

پیام DATABASE DESCRIPTION تمام شماره های ترتیب از درایه های جدول حالت لینک را که اخیراً در پایگاه اطلاعاتی فرستنده آن ذخیره شده، اعلام می کند. هر یک از گیرندگان این پیام، شماره های اعلام شده را با شماره های ترتیب درایه های خودش مقایسه می کند تا بفهمد چه کسی جدیدترین مقادیر را در اختیار دارد. از این پیامها زمانی استفاده می شود که خطی فعال شود.

هر یک از طرفین یک لینک می توانند با استفاده از پیام LINK STATE REQUEST، «اطلاعات حالت پیوند» یکدیگر را مطالبه نمایند. نتیجه این الگوریتم آنست که هر جفت مسیریاب مجاور، آخرین اطلاعات به روز شده یکدیگر را بررسی کرده و بدین نحو اطلاعات جدید در کل ناحیه پخش می شود. تمام این پیامها به کمک بسته های معمولی IP ارسال می شود. فهرست پنج پیام فوق در جدول ۵-۶۶ خلاصه شده است.

توصیف عملکرد	نوع پیام
از این پیام برای شناسایی همسایه ها استفاده می شود.	Hello
هزینه فرستنده پیام تا همسایه هایش را معین می کند.	Link state update
دریافت بسته Link State Update را تایید می کند.	Link state ack
مسیریاب با این پیام فهرست درایه های بهنگام سازی خود را اعلام می کند.	Database description
از شریک خود اطلاعاتی را درخواست می کند.	Link state request

شکل ۵-۶۶. پنج نوع از پیامهای OSPF.

حال می توان مجموعه عوامل فوق را بدین نحو جمع بندی کرد: به کمک روش ارسال سیل آسا، هر مسیریاب به تمام اعضاء ناحیه خود، از همسایه هایش و هزینه رسیدن به آنها خبر می دهد. این اطلاعات امکان آن را فراهم می آورد تا یکایک مسیریابها بتوانند گراف ناحیه خود را تشکیل داده و کوتاهترین مسیره را محاسبه نمایند. ستون فقرات نیز همین کار را می کند. [چرا که ستون فقرات خود یک ناحیه مستقل است.] مضاف بر این، مسیریابهای واقع بر روی ستون فقرات، اطلاعات ارسالی از مسیریابهای مرزی هر ناحیه را هم می پذیرند تا بتوانند کوتاهترین مسیره را از ستون فقرات تا دیگر مسیریابها را محاسبه نمایند. این اطلاعات مجدداً به مسیریابهای مرزی هر ناحیه

۱. یعنی به ازای دریافت هر پیام از این نوع یک پیام ACK بر می گردد.

بازگردانده می شود تا آنها نیز در درون ناحیه خودشان اعلام نمایند. به کمک این اطلاعات، یک مسیریاب که می خواهد بسته ای را به خارج از ناحیه خود بفرستد، بهترین مسیریاب مرزی متصل به ستون فقرات را به عنوان دروازه خروج بسته برمیگزیند.

۵-۶-۵ BGP: پروتکل مسیریابی برای دروازه خارجی

در درون یک AS واحد، پروتکل مسیریابی OSPF، بهترین گزینه است (اگرچه OSPF، تنها پروتکل مورد استفاده و رایج به حساب نمی آید). بین AS ها از پروتکل متفاوتی به نام BGP استفاده می شود. به دلیل آنکه اهداف پروتکل مسیریابی درونی با پروتکل مسیریابی خارجی فرق می کند به پروتکل متفاوتی برای مسیریابی بین AS ها نیاز است. پروتکل مسیریابی درونی باید به سریعترین وجه ممکن بسته ها را از مبدا به مقصد برساند و اهمیتی به سیاستهای جانبی نمی دهد.

پروتکلهای مسیریابی خارجی بایستی در حد وسیعی ملاحظات سیاسی را مدنظر قرار بدهند. (Metz, 2001) برای مثال ممکن است یک شبکه AS بخواهد که بسته ها را به هر سایت اینترنت بفرستد یا بسته ها را از هر سایت اینترنت دریافت کند ولی تمایلی به حمل بسته هایی که از یک AS خارجی تولید شده و به یک AS خارجی دیگر می رود نداشته باشد، حتی اگر AS او بر روی کوتاهترین مسیر بین دو AS خارجی قرار گرفته باشد. (یعنی با این استدلال که: «مشکل آنها به ما ربطی ندارد!» بسته های دیگران را ترانزیت نکنند.) از طرف دیگر ممکن است بخواهد در ازای دریافت هزینه خدمات، ترافیک همسایه های خود و AS های خاصی را مسیریابی و هدایت کند. مثلاً شرکتهای مخابرات تلفن ممکن است مایل باشند به عنوان حامل مشتریان خود عمل کنند ولی نه برای دیگران (و بدون اخذ وجه)! پروتکلهای مسیریابی خارجی بطور عام و پروتکل BGP بطور خاص این امکان را فراهم کرده اند که بتوان سیاستهای گوناگونی را بر روی ترافیک بین AS ها اعمال کرد.

سیاستهای کلی حول مسائل سیاسی، امنیتی یا ملاحظات اقتصادی دور می زند. چند مثال از ملاحظات مسیریابی عبارتند:

۱. عدم اجازه عبور ترافیک داده ها از میان AS های خاص
۲. مسیری که از پنتاگون شروع می شود هرگز از عراق عبور نکند!
۳. مسیر رسیدن از بریتیش کلمبیا به اونتاریو از ایالات متحده نگذرد!
۴. فقط زمانی از مسیر آلبانی عبور شود که هیچ راه دیگری به مقصد وجود نداشته باشد!
۵. ترافیک داده هایی که از IBM شروع شده یا بدان ختم می شود از مایکروسافت عبور نکند!!

عموماً سیاستهای مسیریابی، به صورت دستی بر روی مسیریابهای BGP تنظیم و پیکربندی می شود (یا در قالب نوعی «اسکرپت» بر روی مسیریاب اجرا می گردد). البته این تنظیمات جزئی از پروتکل بحساب نمی آید. از دیدگاه یک مسیریاب BGP، کل جهان از چندین AS و خطوط ارتباطی بین آنها تشکیل شده است. دو AS وقتی متصل تلفی می شوند که بین مسیریابهای مرزی هر یک از آنها حداقل یک خط وجود داشته باشد. با توجه به تأکید BGP بر حمل (یا عبارتی ترانزیت) ترافیک، هر شبکه AS در یکی از سه رده ذیل قرار می گیرد: (۱) رده اول «شبکه های پایانی» (Stub Network) نامیده می شوند؛ این گونه از شبکه ها فقط و فقط یک اتصال با گراف BGP دارند و نمی توانند ترافیک داده ها را از خود عبور بدهند چرا که در طرف دیگر آنها شبکه ای وجود ندارد. (۲) رده بعدی، «شبکه های چندانصالی» (Multiconnected Networks) هستند. این شبکه ها می توانند حمل ترافیک

۱. Border Gateway Protocol

۲. شبکه های پایانی را به مثابه یک کوچه بن بست فرض کنید. -م

یک جفت مسیر یاب BGP از طریق برقراری یک اتصال TCP، با یکدیگر مبادله اطلاعات می نمایند. این مکانیزم، امکان مبادله مطمئن و بدون خطای داده ها را فراهم کرده و جزئیات درونی شبکه را پنهان نگاه می دارد. پروتکل BGP ذاتاً مبتنی بر الگوریتم بردار فاصله است ولی از بسیاری جهات با RIP فرق دارد؛ تفاوت بنیانی در اینجاست که به جای آنکه فقط هزینه و خط رسیدن به یک مقصد در شبکه محاسبه و نگهداری شود، مسیر یاب BGP کل مسیر رسیدن به هر مقصد را نگه می دارد. همچنین به جای آنکه به همسایه های خود در خصوص هزینه رسیدن به هر مقصد در شبکه، خبر بدهد کل مسیرهای واقعی را اعلام می کند.

اطلاعاتی که F از همسایه‌های خود در خصوص D دریافت می‌کند.

B از "I use BCD"
 G از "I use GCD"
 I از "I use IFGCD"
 E از "I use EFGCD"

(الف)

(ب)

شکل ۵-۶۷. (الف) مجموعه‌ای از مسیر یابهای BGP. (ب) اطلاعات ارسالی برای F

پروتکل BGP به سادگی مشکل «شمارش تاب‌ناهی» (که الگوریتمهای بردار فاصله را آزار می‌دهد) حل کرده است.^۲ به عنوان مثال فرض کنید G از کار بیفتد یا خط FG قطع شود. لذا F فقط از سه همسایه باقیمانده خود (یعنی B و I و E) اطلاعات مسیر می‌گیرد. مسیرهائی که همسایه‌های او برای رسیدن به D اعلام می‌کنند عبارتند از:

Scoring Function ⁹

BCD, IFGCD و EFGCD. مسیریاب F بلافاصله متوجه می شود که مسیرهای دوم و سوم بی فایده هستند چراکه از خود F می گذرند، لذا FBCD به عنوان مسیر جدید انتخاب می شود. الگوریتمهای دیگر «بردار فاصله» بدان دلیل در انتخاب مسیر به اشتباه می افتند که همسایه ها نمی توانند اعلام کنند کدامیک از مسیرهای رسیدن به مقصد، مسیر مستقلی است. (یعنی از خود همسایه نمی گذرد). پروتکل BGP در اسناد RFC 1771 تا RFC 1774 تشریح شده است.

۶-۶-۵ ارسال چندپخششی در اینترنت (Internet Multicasting)

عملکرد طبیعی پروتکل IP، مبادله بسته های داده بین یک گیرنده و یک فرستنده است؛ با این وجود در برخی از کاربردها این قابلیت که یک پروسه بتواند بطور همزمان برای تعداد بسیار زیادی گیرنده، ارسال داشته باشد، کارساز و مفید است. مثالهایی از این قبیل عبارتست از: به هنگام سازی همزمان نسخه های توزیع شده یا تکراری پایگاههای اطلاعاتی، اعلام همزمان قیمت سهام به متولیان و واسطه ها، مدیریت و اجرای کنفرانسهای دیجیتال و گردهم آیی تلفنی.

IP با استفاده از آدرسهای کلاس D از فرآیند چندپخششی (Multicasting) پشتیبانی می نماید. هر آدرس کلاس D، هویت یک «گروه» از ماشینها را مشخص می کند. برای مشخص کردن هر گروه در کلاس D، ۲۸ بیت در اختیار است، لذا بطور همزمان می توان بیش از ۲۵۰ میلیون گروه تعریف کرد. هرگاه یک پروسه، بسته ای را به یک آدرس کلاس D ارسال کند، برای رساندن آن بسته به یکایک اعضای گروه، مسیریابها حداکثر تلاش ممکن را بعمل می آورند ولیکن هیچ تضمینی داده نمی شود و احتمال دارد برخی از اعضا بسته را دریافت نکنند!

در IP از دو نوع آدرس گروهی حمایت می شود: آدرسهای دائم (Permanent) و آدرسهای موقت (Temporary). «گروه دائم» همیشه وجود دارد و نیازی به تنظیم و هماهنگی قبلی نیست. هر گروه «دائم دارای آدرس ثابت و بلا تغییر است. چند نمونه از آدرس گروههای ثابت عبارتند از:

گروه 224.0.0.1 : تمام سیستمهای متصل به یک LAN

گروه 224.0.0.2 : تمام مسیریابهای متصل به یک LAN

گروه 224.0.0.5 : تمام مسیریابهای OSPF متصل به یک LAN

گروه 224.0.0.6 : تمام مسیریابهای «برگزیده» OSPF متصل به یک LAN

گروههای موقتی باید قبل از استفاده، ایجاد شوند. یک پروسه می تواند از ماشین خود درخواست کند که به یک گروه خاص بپیوندد؛ همچنین می تواند از آن بخواهد که گروه را ترک کند. وقتی که آخرین پروسه در یک ماشین، گروهی را ترک کند، ماشین مربوطه از آن گروه خارج خواهد شد. هر ماشین می داند که پروسه های او عضو چه گروههایی هستند.

ارسال چندپخششی توسط مسیریابهای خاصی پیاده سازی و پشتیبانی می شود و طبقاً ممکن است در کنار مسیریابهای استاندارد، اینگونه مسیریابها وجود نداشته باشد، لذا توانایی ارسال چندپخششی به نوع مسیریابهای شبکه و پیکربندی آنها برمی گردد. مسیریابهای چندپخششی تقریباً در هر دقیقه یکبار، یک پیام سخت افزاری (یعنی یک فریم لایه پیوند داده ها) را به صورت چندپخششی برای تمام ماشینهای متصل به LAN خودش (یعنی به آدرس 224.0.0.1) می فرستد و از آنها می خواهد که اعلام کنند پروسه هایشان در چه گروههایی عضو هستند. در پاسخ به این سؤال، یکایک ماشینها، آدرسهای کلاس D گروههای مورد نظر خود را بر می گردانند.

این بسته های پرسش و پاسخ مبتنی بر پروتکلی به نام IGMP^۱ است که تا حدودی به ICMP شبیه است. در

^۱ Internet Group Management Protocol

این پروتکل فقط دو نوع بسته تعریف شده است: بسته پرسش و بسته پاسخ. هر یک از این بسته‌ها دارای قالبی ساده و ثابت هستند: در اولین کلمه از فیلد داده (Payload) برخی اطلاعات کنترلی قرار می‌گیرد و در کلمه بعدی یک آدرس کلاس D درج می‌شود. [کلمات چهار بایتی هستند]. این پروتکل در سند RFC 1112 تشریح شده است.

مسیریابی چندپخش در اینترنت به کمک «درختهای پوشا» (Spanning Tree) انجام می‌شود. هر مسیریاب که از فرآیند چندپخش حمایت می‌کند، به کمک «پروتکل اصلاح شده بردار فاصله» یا همسایه‌های خود اطلاعاتی را مبادله می‌کند تا هر کدام بتوانند به ازای هر گروه یک «درخت پوشا» تشکیل بدهند. (به نحوی که تمام اعضای هر گروه را در بر بگیرد). به منظور آنکه درخت به نحوی سازماندهی و پیرایش شود تا مسیریابها یا شبکه‌های غیر عضو در گروه از ساختار درخت حذف گردند، بهینه‌سازی‌هایی صورت گرفته است. این پروتکل از مکانیزم «ایجاد تونل» استفاده زیادی می‌کند تا بدین نحو برای گره‌هایی که در درخت پوشا قرار ندارند سردرگمی و اشکال ایجاد نشود.

۷-۶-۵ IP متحرک (Mobile IP)

بسیاری از کاربران اینترنت از کامپیوترهای کیفی قابل حمل استفاده می‌کنند و طبیعتاً علاقمندند وقتی به یک محل جدید نقل مکان می‌نمایند یا حتی در طول راه، اتصالشان با اینترنت برقرار بماند. متأسفانه، سیستم آدرس دهی IP به گونه‌ای طراحی شده که کار کردن با آن دور از خانه، فقط در گفتار ساده است تا در عمل! در این بخش این مشکل و راه حل آن را بررسی می‌کنیم. (Perkins, 1998a)

نقطه ضعف واقعی IP، در روش آدرس دهی آن نهفته است. یک آدرس IP متشکل از یک شماره شبکه و یک شماره ماشین است. به عنوان مثال ماشینی با آدرس 160.80.40.20/16 را در نظر بگیرید. شماره شبکه 160.80 (یا 8272 در مبنای ده) و شماره ماشین 40.20 (یا 10260 در مبنای ده) است. مسیریابهای کل جهان یک جدول مسیریابی دارند که در آن خط مناسب برای رسیدن به شبکه 160.80 مشخص شده است. وقتی بسته‌ای با آدرس IP بشکل 160.80.xxx.yyy، به مسیریاب وارد گردد بر روی آن خط ارسال خواهد شد.

بدین نحو اگر ماشینی با آدرس فوق به محل جدیدی نقل مکان کند، کماکان بسته‌های ارسالی برای او به شبکه یا مسیریاب خانگی هدایت شده و از آن به بعد صاحب ماشین قادر به دریافت نامه‌های الکترونیکی یا نظائر آن نخواهد بود. انتساب آدرس IP جدید (متناظر با محل جدید) به ماشین سیار، راهکار جالبی نیست چراکه این تغییر باید به تعداد زیادی از افراد، بانکهای اطلاعاتی و برنامه‌های کاربردی اطلاع داده شود.

راهکار دیگر آنست که مسیریابها در جدول مسیریابی به جای آنکه فقط شماره شبکه را نگه دارند آدرس کامل ماشینها را درج نمایند ولیکن این روش نیز مستلزم ذخیره میلیونها درایه (Entry) در جدول مسیریابی است و هزینه‌های نجومی به اینترنت تحمیل می‌کند.

از زمانی که تقاضا برای این قابلیت (که افراد بتوانند در هر کجا که هستند به اینترنت متصل شوند)، بالا گرفت، IETF یک گروه ویژه برای پیدا کردن راه حل مناسب تشکیل داد. این گروه کاری به سرعت اهدافی را که هر پیشنهاد یا راه حل باید آنها را برآورده می‌ساخت، تدوین و تعریف کردند. مهمترین این اهداف عبارت بود از:

۱. هر ماشین متحرک باید بتواند هر جا که هست از آدرس IP همیشگی و خانگی خود استفاده کند.
۲. هر گونه تغییر نرم‌افزاری در ماشینها مجاز شمرده نمی‌شود.
۳. هر گونه تغییر در نرم‌افزار مسیریابها یا جداول آنها مجاز نیست.
۴. اکثر بسته‌هایی که به سوی ماشینهای متحرک هدایت می‌شوند نباید از مسیر واقعی منحرف شوند.

۵. وقتی ماشین میزبان در محل همیشگی خود است نباید هیچ سربرار اضافه ای تحمل کند.^۱

راه حل انتخابی همانی بود که در بخش ۵-۲-۸ آن را تشریح کردیم. در یک مرور اجمالی: هر سائیتی که تمایل داشته باشد امکان سیار بودن را برای کاربران خود فراهم کند موظف به ایجاد یک «عامل خانگی» (Home Agent) است. همچنین هر سائیتی که تمایل به پذیرش کاربران خارجی (کاربران سیار) داشته باشد موظف به ایجاد یک «عامل خارجی» (Foreign Agent) است. هر گاه یک ماشین متحرک در محدوده یک سایت خارجی ظاهر می شود، ابتدا با عامل خارجی تماس برقرار کرده و ثبت نام می کند. عامل خارجی با عامل خانگی آن کاربر تماس گرفته و آدرسی جدید از محل کاربر متحرک به آن اعلام می کند. این آدرس عموماً آدرس IP خود عامل خارجی است.

وقتی بسته ای به LAN محل استقرار دائمی کاربر می رسد قبل از انتشار بر روی LAN به مسیریاب متصل به آن وارد می شود. آن مسیریاب به روش معمول و همیشگی سعی می کند ماشین آن کاربر را پیدا کند لذا یک بسته ARP منتشر کرده و مثلاً می پرسد: «آدرس اترنت ماشین 160.80.40.20 چند است؟». «عامل خانگی» به نیابت از کاربر، به این سؤال پاسخ داده و آدرس اترنت خود را اعلام می کند. از آن به بعد، مسیریاب تمام بسته های متعلق به 160.80.40.20 را برای «عامل خانگی» می فرستد. «عامل خانگی»، با مکانیزم ایجاد تونل بسته اصلی را درون یک بسته IP جدید جاسازی کرده و آدرس مقصد بسته جدید را آدرس عامل خارجی قرار داده و آنرا ارسال می نماید. عامل خارجی پس از دریافت، بسته اصلی را از درون آن استخراج کرده و آنرا به آدرس MAC ماشین متحرک (یعنی آدرس سخت افزاری تعریف شده در لایه پیوند داده ها) ارسال می دارد. مضاف بر این، عامل خانگی کاربر، آدرس IP عامل خارجی را (که کاربر فعلاً مهمان اوست) به فرستنده بسته ها اعلام می کند تا بسته های بعدی به کمک مکانیزم تونل مستقیماً به آدرس عامل خارجی ارسال شوند. این راه حل تمام نیازها و اهداف فوق الذکر را برآورده می کند.

اشاره به برخی از جزئیات ارزشمند است: در لحظه ای که ماشین متحرک، محل خود را ترک می کند احتمالاً مسیریاب در جدول نهان ARP آدرس اترنت این ماشین را (که دیگر معتبر نیست) درج کرده و از آن استفاده می کند.^۲ برای جایگزین کردن آدرس اترنت «ماشین عامل» به جای آدرس ماشین متحرک از راهکاری زیرکانه به نام «Gratuitous ARP» استفاده می شود. این روش مبتنی بر ارسال یک پیام خاص و ناخواسته برای مسیریاب است که باعث می شود یک درایه دلخواه در جدول ARP به مقداری جدید تغییر کند. در اینجا درایه ای که باید تغییر کند متعلق به ماشین متحرک و در حال خروج از شبکه است. وقتی بعداً ماشین متحرک به شبکه خود باز می گردد از همین راهکار برای بهنگام سازی مجدد جدول ARP در مسیریاب، استفاده می شود.^۳

هیچ مانعی وجود ندارد که یک ماشین متحرک نتواند «عامل خارجی» خودش باشد ولی این راهکار زمانی عملی است که ماشین متحرک در موقعیت جدید خود، به اینترنت دسترسی داشته باشد. در ضمن ماشین متحرک باید بتواند در محل جدید یک آدرس IP موقت جهت اعلام به «عامل خانگی» خودش بدست بیاورد. این آدرس

۱. یعنی اگر سربرار کوچکی تحمیل می شود باید فقط زمانی باشد که ماشین میزبان در محل همیشگی خود نیست. -م

۲. ARP Cache

۳. یعنی چون پس از خروج ماشین متحرک، هنوز آدرس MAC آن در جدول ARP مسیریاب، وجود دارد تا موقعی که این آدرس اعتبار خود را حفظ می کند بسته ها به آدرسی که دیگر وجود ندارد ارسال خواهد شد. -م

۴. این راهکار آنست که «ماشین عامل» بدون آنکه هیچ سؤالی از او شده باشد یک بسته پاسخ ARP (یعنی ARP Response) به صورت مصنوعی تولید و ارسال می کند. بدین ترتیب مسیریاب به اشتباه افتاده و محتوای آنرا در جدول خود درج می کند، هر چند هرگز در این خصوص سؤالی نپرسیده بود! -م

IP متعلق به LAN جدیدی است که ماشین متحرک موقتاً بدان متصل شده و مهمان آنست. راه حل پیشنهادی IETF برای ماشینهای متحرک، مسائل متعدد دیگری را هم حل کرد که تا آن زمان بدان توجه نشده بود. به عنوان مثال یکی از مسائل این بود که ماشینهای عامل چگونه پیدا شوند؟ راه حل این مسئله آنست که هر عامل بطور متناوب آدرس خود و نوع سرویسهایی را که علاقمند به ارائه آنهاست، به صورت فراگیر منتشر نماید. وقتی ماشین متحرک به جایی وارد می شود ابتدا سعی می کند به این پیامهای انتشاری که اصطلاحاً «اعلان» (Advertisement) نامیده می شود، گوش فرایدهد. گزینه دیگر آنست که ماشین متحرک ورود خود را با انتشار بسته ای فراگیر (Broadcast) اعلام کرده و در انتظار پاسخ ماشین «عامل خارجی» باقی بماند. مشکل دیگری که باید حل شود آنست که با ماشینهای متحرکی که با عجله و بدون خداحافظی شبکه فعلی خود را ترک می کنند، چه باید کرد؟ راه حل آنست که حضور یک ماشین متحرک فقط برای مدت زمان ثابت و محدودی اعتبار داشته باشد. اگر ماشین متحرک بطور مرتب، اعتبار حضور خود را تمدید نکند و مهلت اعتبار او منقضی شود، عامل خارجی جداول خود را از مشخصات او پاک می نماید.

مشکل دیگر، موضوع امنیت است: وقتی یک عامل خانگی پیامی دریافت می کند که از او خواسته شده تمام بسته های متعلق به کاربری به نام روبرتا را به آدرس IP خاصی بفرستد، بهتر است این تقاضا پذیرفته نشود مگر آنکه اثبات شود که مبداء این درخواست، واقعاً روبرتا است نه کسی که خودش را به جای او جا زده است. بدین منظور از پروتکل های احراز هویت مبتنی بر رمزنگاری استفاده می شود. این پروتکلها را در فصل هشتم تشریح خواهیم کرد.

آخرین نکته ای که گروه کاری IETF بدان پرداخت در خصوص «سطوح تحرک» (Levels of Mobility) ماشینهاست. هواپیمایی را در نظر بگیرید که برای وصل کامپیوترهای مخصوص ناوبری و کنترل پرواز شبکه ای از نوع اترنت را بکار گرفته است. بر روی این شبکه یک مسیریاب استاندارد وجود دارد که از طریق یک لینک رادیویی با شبکه اینترنت مستقر بر روی زمین در ارتباط است. از طرفی این ایده در ذهن برخی از افراد زیرک جرقه زده که در کنار دسته صندلی هر مسافر یک کانکتور اترنت تعبیه شود تا مسافرتین بتوانند در خلال پرواز، کامپیوترهای کیفی خود را به آن متصل کرده و به اینترنت وصل شوند. در این حالت باید برای کامپیوترهای متحرک دو سطح قائل شد: کامپیوترهای خود هواپیما، که نسبت به اترنت موجود در آن ثابت (Stationary) محسوب می شوند و کامپیوتر مسافران که نسبت به آن متحرک هستند. البته مسیریاب هواپیما نسبت به مسیریاب مستقر بر روی زمین، متحرک است. می توان متحرک بودن کامپیوترها نسبت به مسیریابهایی که خودشان متحرک هستند را از طریق ایجاد تونلهای متوالی پیاده سازی و اداره کرد.

۸-۶-۵ IPv6

اگرچه ممکن است مکانیزمهای CIDR و NAT چند سالی دیگر به دوام نسخه چهارم IP کمک کنند ولی تقریباً بر همه آشکار شده که نفسهای پروتکل IP در شکل کنونی آن، به شماره افتاده است. مضاف بر مشکلات فنی IP، برخی از موارد پشت صحنه و زمینه ای دیگر نیز مطرح است. در سالهای اولیه، از اینترنت عموماً در دانشگاهها، صنایع پیشرفته و دولت ایالات متحده (خصوصاً وزارت دفاع) استفاده می شد. با گرایش بسیار زیاد مردم به اینترنت که از اواسط دهه نود شروع شد، گروههای مختلفی از افراد به آن رو آوردند؛ افرادی که نیازها و انتظارات متفاوتی داشتند. یکی از موارد آنست که افراد با کامپیوترهای بی سیم قابل حمل برای در ارتباط بودن با محل استقرار دائمی خود (ایستگاههای خانگی) می خواهند از اینترنت بهره بگیرند. مورد دیگر آن که با همگرایی قریب الوقوع صنایع کامپیوتر و مخابرات و صنایع تولید بازی و ابزار تفریح، دیری نخواهد پایید که حتی دستگاههای تلفن و تلویزیون در دنیا، به عنوان گرهی از اینترنت، به آن خواهد پیوست و در آن زمان میلیاردها

ماشین، از صدا و تصویر بهره خواهند گرفت. با در نظر داشتن چنین چشم اندازی، IP بوضوح نیازمند تغییرات اساسی است و باید انعطاف بیشتری داشته باشد.

IETF که چنین افقی را پیش روی خود می دید در اوان ۱۹۹۰ کار را بر روی نسخه جدیدی از پروتکل IP شروع کرد که در آن فضای آدرس هرگز با کمبود مواجه نشود و مشکلات عدیده ای را حل کند؛ قابلیت انعطاف بیشتری داشته باشد و در ضمن کارآمدتر باشد. اهداف عمده IPv6 عبارت بودند از:

۱. پشتیبانی از میلیاردها ماشین میزبان حتی در صورتی که تخصیص فضای آدرس ناکارآمد و با اسراف انجام شود.
۲. کاهش اندازه جداول مسیریابی
۳. ساده سازی پروتکل به منظور افزایش سرعت پردازش مسیریابیها
۴. ارائه امنیت بهتر در مقایسه با نسخه فعلی IP (شامل احراز هویت و سرّی ماندن داده ها)
۵. توجه بیشتر به نوع خدمات و QoS، به ویژه برای داده های بی درنگ
۶. کمک به فرآیند ارسال چندپخششی از طریق توصیف حوزه ها (Scopes)
۷. فراهم آوردن امکان جابجایی ماشینهای میزبان بدون تغییر در آدرس
۸. امکان ایجاد تغییر و پیشرفت در آینده
۹. امکان همزیستی پروتکل های جدید و قدیم در طی سالها

برای توسعه پروتکلی که تمام نیازهای فوق الذکر را برآورده نماید، IETF با انتشار RFC 1550 و در طی یک فراخوان، خواستار پیشنهادات دیگران در این خصوص شد. ۲۱ پیشنهاد دریافت گردید که اغلب آنها جامع نبودند. تا دسامبر ۱۹۹۲ فقط هفت طرح پیشنهادی قابل توجه در دستور کار قرار داشت. این طرحهای پیشنهادی از اصلاحات جزئی در نسخه فعلی IP تا پیشنهاد دور انداختن آن و جایگزینی با یک پروتکل کاملاً متفاوت را شامل می شد.

یک پیشنهاد آن بود که TCP بر روی CLNP اجرا شود؛ پروتکلی که با آدرسهای ۱۶۰ بیتی فضای آدرس دهی نامحدود و جاویدان را فراهم کرده بود و دو پروتکل عمده و مهم لایه شبکه را متحد و یکپارچه می کرد. ولیکن بسیاری افراد احساس کردند که پذیرش آن مهر تأییدی است بر این ادعا که هر کاری که OSI انجام داده صحیح تلقی می شود، داعیه ای که لاقابل در حوزه اینترنت به دلایل خاص نادرست است. الگوی CLNP بسیار شبیه به IP بود و این دو، تفاوت چندانی با هم ندارند. در آخر نیز طرحی انتخاب شد که تفاوت بسیار زیادی با IP و از آن بیشتر با CLNP دارد. ضربه دیگری که CLNP خورد از آنجا بود که پشتیبانی ضعیفی از «نوع خدمات» (Type of Service) می کرد، خصوصیتی که برای انتقال کارآمد داده های چند رسانه ای به شدت نیاز بود.

سه تا از بهترین طرحهای پیشنهادی در ژورنال IEEE Network منتشر شد.^۱ پس از مباحثات فراوان، بازبینی، ارزیابی موقعیت و سنجش استقبال عمومی، نسخه ای ترکیبی از طرحهای پیشنهادی Deering و Francis که SIPP نامیده می شد به عنوان طرح برگزیده معرفی و با عنوان IPv6 معرفی گردید.

IPv6 بخوبی اهداف مورد نظر را برآورده می کند: ویژگیهای خوب IP را نگه داشته، ویژگیهای بد را کنار گذاشته با کم رنگ کرده و ویژگیهای جدیدی به آن افزوده است. بطور کلی IPv6 با IPv4 سازگار نیست ولی با تمام پروتکل های جانبی اینترنت مثل TCP، UDP، ICMP، IGMP، OSPF، BGP و DNS سازگار است. (البته

۱. Deering, 1993; Francis, 1993; and Katz and Ford, 1993

۲. Simple Internet Protocol Plus

ممکن است به دلیل آنکه آدرسها طولانی‌تر شده‌اند نیاز به اندکی تغییر داشته باشند.) ویژگیهای اساسی IPv6 در زیر تشریح شده است. برای آگاهی بیشتر در خصوص آن به RFCهای 2460 تا 2466 مراجعه نمایید.

اولین و مهمترین ویژگی آنست که IPv6 آدرسهای بسیار طولانی‌تری نسبت به IPv4 دارد. این آدرسها ۱۶ بایت طول دارند و دقیقاً مشکلی را حل کرده که به همان دلیل طراحی شد: یعنی تقریباً فضای نامحدودی از آدرسهای IP را فراهم آورده است. در این خصوص بیشتر صحبت خواهیم کرد.

دومین پیشرفت عمده IPv6 ساده‌سازی سرآیند آنست. این سرآیند جمعاً هفت فیلد دارد (در مقابل سیزده فیلد در IPv4). این تغییر، امکان آنرا فراهم آورده که مسیر یاب بسته‌ها را سریعتر پردازش نماید و ظرفیت مفید مسیر یاب را افزایش و تأخیر را کاهش دهد. در ادامه مختصراً به سرآیند خواهیم پرداخت.

سومین بهبود عمده آن پشتیبانی از گزینه‌های اختیاری (Options) است. این تغییر برای سرآیند جدید حیاتی بود چراکه برخی از فیلدهایی که در نسخه قبلی وجودشان الزامی است در نسخه فعلی اختیاریند. مضاف بر این، روش درج گزینه‌ها در این فیلد متفاوت از قبل است و اجازه می‌دهد مسیر یابها بتوانند به سادگی از گزینه‌هایی که برایشان مهم نیست رد شوند. [یعنی در نسخه جدید، دسترسی تصادفی و مستقیم به گزینه‌ها ممکن شده است.] این ویژگی سرعت پردازش بسته‌ها را افزایش می‌دهد.

چهارمین موضوعی که IPv6 در آن پیشرفت عمده‌ای داشته است، «امنیت» است. IETF با انبوهی از گزارشهای مطبوعاتی در خصوص نابغه‌های دوازده ساله‌ای مواجه بود که با کامپیوترهای شخصی خود، به شبکه‌های بانکی یا پایگاههای نظامی در سراسر اینترنت، نفوذ کرده بودند و جو شدیدی براه افتاده بود که باید برای بهبود امنیت شبکه‌ها کاری کرد. در نسخه جدید IP، «احراز هویت» (Authentication) و «حفظ امنیت اطلاعات» (Privacy) جزو ویژگیهای کلیدی به شمار می‌رود. البته این ویژگیها بعداً به IPv4 نیز افزوده شد [با عنوان IPsec]. فلذا در حال حاضر این دو، در زمینه امنیت تفاوت چندانی با هم ندارند.

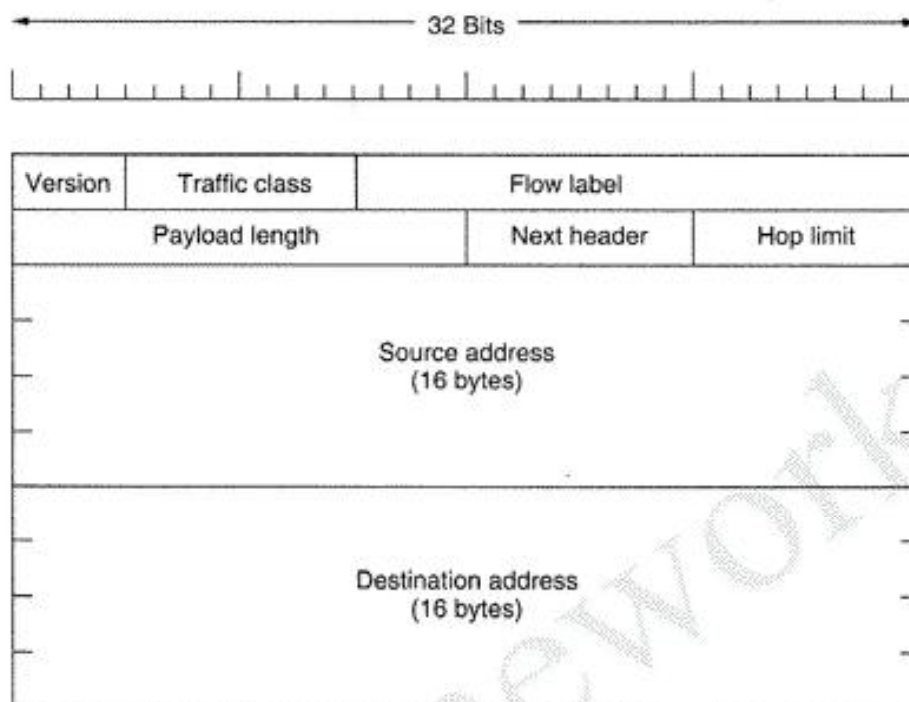
موضوع آخر آنکه در نسخه جدید به «کیفیت خدمات» (QoS) دقت بیشتری شده است. قبل از آن نیز تلاشهای جسته و گریخته‌ای در این رابطه انجام شده بود ولیکن با رشد کاربردهای چند رسانه‌ای در اینترنت، این موضوع جدی‌تر و حساس‌تر به نظر می‌رسید.

سرآیند اصلی IPv6

سرآیند اصلی IPv6 در شکل ۵-۶۸ نشان داده شده است. فیلد Version (شماره نسخه پروتکل) برای IPv6 همیشه ۶ است. (کما اینکه برای IPv4 نیز همیشه ۴ است!). در دوران گذار از IPv4 به نسخه جدید که ممکن است یک دهه طول بکشد، مسیر یابها قادرند با بررسی این فیلد تشخیص بدهند که با چه نوع بسته‌ای روبرو هستند. البته از آنجایی که بررسی این فیلد به چندین دستورالعمل اجرایی CPU نیاز دارد و این کار زمان مفید پردازش هر بسته را هدر می‌دهد لذا در بسیاری از پیاده‌سازیهای عملی، برای اجتناب از این زمان تلفاتی، تشخیص آنکه یک بسته از نوع IPv4 است یا IPv6، با استفاده از فیلد خاصی در سرآیند لایه پیوند داده‌ها بر عهده سخت‌افزار گذاشته شده است.^۱ بدین ترتیب بسته‌ها براساس نوعشان مستقیماً به نرم‌افزار مناسب در لایه شبکه هدایت می‌شوند. البته این الزام که لایه پیوند داده از جزئیات نوع بسته‌های لایه شبکه آگاه باشد با این اصل اساسی که «هر لایه نباید از معنای بیتهایی که از لایه بالاتر تحویل او می‌شود، آگاه باشد» در تناقض است. بدون شک بحث و مناقشه بین طرفداران ایده‌های «انجام اصولگرایانه و صحیح کار» و «تسریع کار» به شدت ادامه خواهد داشت.

فیلد Traffic Class (کلاس ترافیک) برای تشخیص تفاوت بسته‌ها از لحاظ نیازمندیهای تحویل بی‌درنگ و

۱. مثلاً در فریم اینترنت فیلد Type نوع بسته درون فریم را تعیین می‌کند. -م



شکل ۵-۶۸. سرآیند ثابت و الزامی در IPv6.

QoS درخواستی، بکار می آید. فیلدی با همین منظور از ابتدا در IP وجود داشت ولیکن استفاده از آن به صورت پراکنده و سلیقه‌ای بر روی مسیریابها پیاده‌سازی شد و اغلب مسیریابها آن را نادیده می‌گرفتند. اکنون تجربیات گذشته چراغ راهی شده تا بتوان بهترین راه و روش تحویل بسته‌های اطلاعات چندرسانه‌ای را تعیین کرد. فیلد Flow Label (برچسب جریان) همچنان آزمایشی است ولی کاربرد مورد نظر آن، این بوده که بتوان یک «شبه اتصال» (Pseudoconnection) بین مبدا و مقصد، با ویژگیها و نیازمندیهای خاص ایجاد کرد. به عنوان مثال، جریانی از بسته‌ها که از یک پروسه در مبدا خاص تولید و به سوی یک پروسه بر روی مقصد خاص روانه می‌شوند احتمالاً نیاز به تضمین تأخیر محدود و مشخص دارد و در نتیجه باید پهنای پاند لازم را رزرو کرد. در چنین مواردی می‌توان پیشاپیش یک «جریان» (Flow) با مشخصات درخواستی تنظیم کرد و به آن یک شناسه اختصاص داد. هر گاه مسیریاب بسته‌ای دریافت کند و فیلد Flow Label آن غیرصفر باشد، با مراجعه به جداول درونی خود تشخیص می‌دهد که با این بسته چگونه رفتار کند. در حقیقت استفاده از مفهوم «جریان»^۱ در IPv6، تلاشی است برای رسیدن به قابلیت انعطاف در زیرشبکه‌های دیتاگرام و تضمین کیفیت خدمات در زیرشبکه‌های مدار مجازی.

هویت هر «جریان» برحسب آدرس مبدا، آدرس مقصد و شماره جریان (برچسب جریان) مشخص می‌شود فلذا بین دو مبدا و مقصد در شبکه می‌توان بطور همزمان چندین «جریان» فعال تنظیم کرد. همچنین در این روش حتی اگر دو جریان متفاوت با شماره جریان یکسان از دو ماشین میزبان مختلف تولید و از مسیریابهای مشابهی عبور کنند، مسیریابها به کمک آدرس مبدا و مقصد قادر به تشخیص آنها خواهند بود. انتظار آنست که «برچسبهای جریان» به جای آنکه به صورت ترتیبی و از ۱ شروع شوند به صورت کاملاً تصادفی انتخاب گردند تا مسیریاب

۱. برای آشنایی با مفهوم جریان رجوع کنید به بخش ۵-۴-۱.

بتواند آنها را در Hash Table خود درج کند.^۱

فیلد Payload Length (طول قسمت حمل داده) مشخص می کند که پس از سرآیند ۴۰ بایتی در شکل ۵-۶۸ چند بایت داده قرار گرفته است. همین فیلد در IPv4 با نام Total Length وجود داشت. تغییر نام به آن دلیل بوده که در نسخه جدید، سرآیند جزو طول بسته به حساب نمی آید بلکه فقط اندازه بخش حمل داده تعیین می شود. فیلد Next Header ساختار بسته را سبکبار کرده است! دلیل آنکه سرآیند بسته ساده شده آنست که می توان در صورت لزوم سرآیند اضافی و انتخابی داشت. این فیلد مشخص می کند که پس از سرآیند ۴۰ بایتی کدامیک از سرآیندهای ششگانه اضافی قرار گرفته است (در صورت وجود). اگر سرآیند اخیر، آخرین سرآیند بسته IP باشد، این فیلد مشخص می کند که کدام پروتکل در لایه انتقال محتوای بسته را تحویل خواهد گرفت (مثلاً TCP، UDP و نظائر آن).^۲

کاربرد فیلد Hop Limit آنست که بسته ها عمر محدودی داشته باشند. این فیلد در عمل مشابه با فیلد Time to Live (زمان حیات بسته) در IPv4 است یعنی به ازای عبور بسته از یک مسیر یاب، یک واحد از مقدار آن کاسته می شود. در تئوری، مبنایی که IPv4 برای این فیلد در نظر داشت، «زمان بر مبنای ثانیه» بود در حالی که هیچ مسیر یابی از چنین مبنایی استفاده نمی کند (بلکه به ازای هر گام یک واحد از آن می کاهد) لذا نام این فیلد را به گونه ای عوض کردند که عملکرد واقعی آن را نشان بدهد. هر گاه مقدار این فیلد در یک بسته به صفر برسد آن بسته حذف خواهد شد.

در ادامه فیلدهای Source Address و Destination Address (آدرس مبدا و مقصد) قرار گرفته اند. در طرح پیشنهادی آقای Deering ۸ بایتی انتخاب شده بودند در حالی که در مراحل بازمینی دیگران احساس کردند که شاید این فضای آدرس نیز در خلال چند دهه، IPv6 را نیز با کمبود فضای آدرس مواجه کند، در حالی که با آدرسهای ۱۶ بایتی هرگز چنین کمبودی رخ نخواهد داد. برخی از افراد معتقد بودند که آدرسهای ۱۶ بایتی بیش از حد بزرگ هستند در حالی برخی دیگر اعتقاد داشتند باید از آدرسهای ۲۰ بایتی استفاده شود تا با پروتکل دیتاگرام پیشنهادی OSI سازگار باشد. گروه دیگری نیز به آدرسهای با طول متغیر گرایش داشتند. پس از بحث و جدل فراوان، به این نتیجه رسیدند که آدرسهای با طول ثابت ۱۶ بایتی بهترین انتخاب است. با توجه به طول زیاد آدرسهای IP، نماد جدیدی برای نوشتن آنها پیشنهاد شد. این آدرسها به صورت هشت گروه که با علامت :: از هم جدا شده، نوشته می شوند. هر گروه نیز به صورت چهار رقم هگزادسیمال نمایش داده می شود:

8000:0000:0000:0000:0123:4567:89AB:CDEF

از آنجایی که در آدرسها، تعداد ارقام صفر زیاد است، سه نوع بهینه سازی مجاز شمرده شده: صفرهای سمت چپ در هر گروه نوشته نمی شوند یعنی 0123 به صورت 123 نشان داده می شود؛ دوم آنکه اگر یک یا چند گروه شانزده بیتی تماماً صفر باشند با یک زوج علامت :: نشان داده می شود. بنابراین آدرس مثال بالا به صورت زیر نوشته خواهد شد:

8000::123:4567:89AB:CDEF

نهایتاً آنکه آدرسهای IPv4 را می توان با یک جفت :: و سپس آدرس نقطه دار قدیمی، نشان داد:

::192.31.20.46

۱. به عبارت دیگر مسیر یاب انتظار دارد این شماره ها را Hash کند لذا این شماره ها نباید متوالی باشند.

۲. به عبارت دیگر محتوای این فیلد به صورت بازگشتی سرآیندهای بعدی را مشخص می کند تا نهایتاً به سرآیند آخر برسد که نوع بسته لایه انتقال را تعیین می نماید. -م

شاید لازم به گفتن نباشد که آدرسهای شانزده بیتی، فضایی معادل 2^{128} آدرس هستند که چنین فضایی تقریباً معادل 3×10^{38} آدرس است. اگر کل کره زمین شامل خشکیها و دریاها پر از کامپیوتر شوند باز هم IPv6 می تواند برای هر مترمربع 7×10^{23} آدرس IP فراهم کند. دانشجویان رشته شیمی می دانند که این عدد حتی از عدد آووگادرو نیز بزرگ است. [عدد آووگادرو 6.02×10^{23} است]. چون در نظر نبوده که حتی به مولکولهای سطح زمین آدرس بدهیم آدرسهای IP شانزده بیتی، بهیچوجه کم نخواهد آمد!!

در عمل از فضای آدرس IP، بخوبی استفاده نخواهد شد. (دقیقاً همانند فضای شماره های تلفن که مثلاً فضای شماره های تلفن منتهن با پیش شماره ۲۱۲ پر شده ولی فضای شماره های ویومینگ (Wyoming) با پیش شماره ۳۰۷ تقریباً خالی مانده است.) دو نفر به نامهای Durand و Huitema در سند RFC 3194 محاسبه کرده اند که با ایده گرفتن از تخصیص شماره های تلفن و حتی در بدبینانه ترین حالت ممکن، باز هم می توان برای هر مترمربع از کره زمین، ۱۰۰۰ آدرس IP کنار گذاشت. در حالت کلی نیز می توان تریلیونها آدرس IP برای هر مترمربع از زمین در نظر گرفت. کوتاه سخن آنکه، در آینده هیچگاه به مشکل فضای آدرس بر نخواهیم خورد.

مقایسه سرآیند IPv4 (شکل ۵-۵۳) با سرآیند IPv6 (شکل ۵-۶۸) از این دیدگاه که چه فیلدی و چرا حذف شده است، آموزنده خواهد بود: فیلد IHL حذف شده زیرا سرآیندهای IPv6 طول ثابتی دارد. فیلد «پروتکل» وجود ندارد چرا که فیلد Next Header مشخص می کند که پس از آخرین سرآیند چه بسته دیگری آمده است (بسته TCP، UDP یا نظائر آن).

تمام فیلدهایی که در ارتباط با «قطعه قطعه سازی» بسته ها در IPv4 تعریف شده بود در IPv6 حذف گردیده است زیرا پروتکل اخیر راهکار دیگری برای مکانیزم قطعه قطعه سازی برگزیده است. انتظار آنست که تمام ماشینهای سازگار با IPv6 بتوانند به صورت خودکار و پویا اندازه دیتاگرامها را تعیین کنند و بدین نحو نیاز به قطعه قطعه شدن بسته ها کمتر اتفاق می افتد. همچنین حداقل طول بسته ای که هر ماشین موظف به پذیرش آنست از ۵۷۶ بایت به ۱۲۸۰ بایت افزایش یافته تا بتوان یک قطعه داده ۱۰۲۴ بیتی را به همراه تعداد زیادی سرآیند (۲۵۶ بایت)، بدون نیاز به قطعه قطعه شدن ارسال و دریافت کرد. مضاف بر این، وقتی ماشین یک بسته بیش از حد بزرگ IPv6 را ارسال می دارد مسیر یاب ناتوان از هدایت آن، به جای قطعه قطعه کردن بسته آن را حذف کرده و پیام خطایی را باز می گرداند. این پیام به ماشین میزبان تفهیم می کند که باید بسته هایش را بشکند. البته اگر ماشین میزبان خودش بسته ها را با اندازه مناسب ارسال کند کارآمدتر از آنست که بسته ها در طول مسیر شکسته شود.

فیلد Checksum نیز حذف شد زیرا محاسبه آن کارآیی و سرعت پردازش بسته ها را به نحو چشمگیری کاهش خواهد داد. با توجه به قابلیت اعتماد شبکه های کنونی و با در نظر داشتن این حقیقت که در لایه پیوند داده و لایه انتقال نیز (بطور مجزا) صحت داده ها بررسی می شود، محاسبه یک کد کشف خطای دیگر مثل checksum در مقایسه با کاهش کارآیی ارزشی ندارد. حذف این ویژگیهای زائد، IPv6 را به پروتکل متعادل و جمع و جور تبدیل کرده است. بدین ترتیب IPv6 به اهداف مورد نظر خود که همانا انعطاف، سرعت و فضای بزرگ آدرس بوده، نائل شده است.

سرآیندهای اضافی (سرآیندهای توسیع یا Extension Header)

گاهی به برخی از فیلدهای حذف شده IPv4 نیاز می شود و به همین منظور در IPv6 مفهوم جدیدی به نام «سرآیندهای توسیع» معرفی شده است. این سرآیندهای اختیاری برای افزودن اطلاعات به هر بسته بکار می آیند ولیکن روش کدینگ (و جاسازی) آنها کارآمد و سریع است. شش نوع مختلف سرآیند توسیع که تاکنون معرفی شده، در شکل ۵-۶۹ فهرست گردیده است. هر کدام از این سرآیندها اختیاریند ولیکن اگر به پیش از یک سرآیند نیاز باشد باید بطور پیاپی، پس از سرآیند ثابت و ترجیحاً به ترتیب فهرست، قرار بگیرند.

توصیف عملکرد	نام سرآیند توسیع (سرآیند اضافی)
حاوی اطلاعات گوناگون برای مسیریابها	Hop-by-hop options
اطلاعات اضافی برای مقصد	Destination options
فهرست ناکاملی از مسیریابها که بسته باید از آنها بگذرد.	Routing
مدیریت قطعات دیتاگرام	Fragmentation
بررسی هویت فرستنده	Authentication
اطلاعاتی در خصوص محتوای رمزنگاری شده بسته	Encrypted security payload

شکل ۵-۶۹. سرآیندهای توسیع در IPv6 (Extension Header).

برخی از سرآیندها دارای قالب ثابتی هستند در حالی که برخی دیگر تعداد متغیری فیلد با طول متفاوت دارند. به همین دلیل هر آیتم در قالب سه تایی (نوع، طول، مقدار)^۱ سازماندهی و کُد می شود. فیلد Type مشخص می کند که نوع گزینه چیست. مقدار فیلد نوع (Type) به نحوی انتخاب شده است که دو بیت ابتدایی آن به مسیریابهایی که نمی دانند آن گزینه را چگونه پردازش کنند، راه و چگونگی کار را نشان می دهند. این راهکارها عبارتند از: (۱) گزینه مربوط را نادیده بگیر (۲) بسته را حذف کن (۳) بسته را حذف و یک بسته ICMP برگردان (۴) بسته را حذف کن و یک بسته ICMP برگردان ولیکن بسته ICMP را برای آدرسهای «چندپخش» (Multicast) نفرست. (تا یک بسته چندپخش اشتباه، منجر به تولید میلیونها گزارش ICMP نشود). فیلد یک بایتی طول (Length) مشخص می کند که فیلد مقدار (Value) چند بایتی است. (صفر تا ۲۵۵ بایت). فیلد مقدار (Value) در برگیرنده اطلاعات مورد نیاز است و حداکثر می تواند ۲۵۵ بایت باشد. «سرآیند توسیع Hop-by-Hop» (گام به گام) برای حمل اطلاعاتی کاربرد دارد که مسیریابهای واقع بر مسیر باید آنها را بررسی نمایند. قبلاً یکی از گزینه ها را معرفی کردیم: پشتیبانی از دیتاگرامهایی با طول بیش از ۶۴ کیلوبایت. قالب این سرآیند در شکل ۵-۷۰ نشان داده شده است. وقتی از این سرآیند استفاده می شود باید فیلد Payload Length (در سرآیند اصلی) به صفر مقداردهی شود.

Next header	0	194	4
Jumbo payload length			

شکل ۵-۷۰. «سرآیند توسیع Hop-by-Hop» (گام به گام) برای دیتاگرامهای بسیار بزرگ (جامبوگرام).

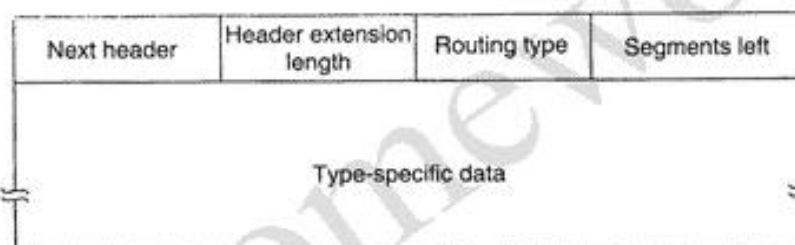
همانند تمام سرآیندهای اختیاری دیگر، این سرآیند نیز با فیلدی یک بایتی به نام Next Header شروع می شود و مشخص می کند که سرآیند بعدی از چه نوع است. پس از این بایت، بایت دیگری قرار گرفته که طول سرآیند Hop-by-Hop را بر مبنای بایت تعیین می کند ولیکن در مقدار آن، ۸ بایت ابتدایی (که وجود آن الزامی است) لحاظ نمی شود. تمام سرآیندهای توسیع دیگر نیز به همین نحو شروع می شوند. در ادامه فیلدی دو بایتی آمده که بایت اول مشخص می کند که این گزینه (Option) قرار است اندازه دیتاگرام را تعریف کند (کد ۱۹۴) و بایت بعدی مشخص می کند که اندازه دیتاگرام یک شماره چهار بایتی است. چهار بایت آخر این سرآیند، طول دیتاگرام را مشخص می کند. اندازه زیر ۶۵۵۳۶ مجاز نیست و منجر به حذف بسته در اولین مسیریاب و بازگشت پیام خطای ICMP خواهد شد. دیتاگرامهایی که از این سرآیند اختیاری (یعنی Hop-by-Hop Header) استفاده

^۱ (Type, Length, Value)

کرده‌اند اصطلاحاً Jumbogram (دیتاگرام عظیم) نامیده می‌شوند. [بدین ترتیب در IPv6 می‌توان قطعات داده بسیار بزرگ را به کمک سرآیند فوق به صورت یکجا ارسال کرد.] کاربرد جامبوگرامها در سوپر کامپیوترها که باید چندین گیگابایت اطلاعات را از طریق اینترنت منتقل کنند، بسیار حیاتی است.

«سرآیند توسعه Destination Options» برای درج فیلدهایی در نظر گرفته شده که صرفاً توسط ماشین مقصد پردازش و تفسیر می‌شوند. در نسخه اولیه IPv6، مقدار این گزینه پوچ در نظر گرفته شده و کاربردی نداشته است. وجود چنین فیلدی برای آن بوده که نرم افزار ماشینهای میزبان و مسیربایها چنین سرآیندی را به رسمیت بشناسند تا اگر روزگاری به آن نیاز شد، شرایط مهیا باشد وگرنه باید پروتکل عوض شود.

«سرآیند توسعه Routing»، فهرست مسیربایهایی را مشخص می‌نماید که بسته باید در راه رسیدن به مقصد از آنها عبور کند. این گزینه شباهت زیادی به گزینه Loose Source Routing در IPv4 دارد. فهرست آدرسهای که در این سرآیند مشخص شده باید در طول مسیر و به ترتیب ملاقات شوند ولی این امکان وجود دارد که مسیربایهایی هم که آدرس آنها در فهرست نیست مابین مسیر باشند. قالب سرآیند Routing در شکل ۷۱-۵ مشخص شده است.



شکل ۷۱-۵. سرآیند توسعه برای مسیربایی (Routing).

چهار بایت اول از این سرآیند، شامل چهار فیلد یک بایتی است: دو فیلد Next Header و Header Extension Length را قبلاً تعریف کردیم. فیلد Routing Type، ساختار مابقی سرآیند را مشخص می‌نماید: مقدار صفر مشخص کننده آنست که پس از کلمه چهار بایتی اول، یک کلمه چهار بایتی دیگر قرار گرفته و پس از آن آدرس IPv6 (یعنی آدرسهای ۱۲۸ بیتی) مسیربایها قرار می‌گیرد. به غیر از این ساختار، فعلاً ساختار دیگری تعریف نشده مگر آنکه در آینده چیز جدیدی ابداع شود. فیلد آخر یعنی Segment Left تعداد آدرسهای را مشخص می‌کند که هنوز ملاقات نشده‌اند. مقدار اولیه این فیلد معادل با تعداد مسیربایهایی است که آدرس آنها در فهرست مورد نظر درج شده است و به ازای ملاقات در هر مسیربای که آدرس آن در فهرست آمده یک واحد از این فیلد کاسته می‌شود. وقتی مقدار این فیلد در بسته به صفر برسد بسته روال طبیعی طی مسیر خود را از سر می‌گیرد بدون آنکه اجبار به عبور از مسیر خاصی داشته باشد. معمولاً در چنین لحظه‌ای بسته به مقصد خود نزدیک شده است.

سرآیند Fragmentation مشابه با IPv4، با مسئله قطعه‌قطعه سازی بسته‌ها سر و کار دارد. در این سرآیند نیز فیلدهای «شماره شناسایی دیتاگرام»، «شماره قطعه» و یک بیت MF تعریف شده‌اند که بیت MF مشخص می‌کند که آیا قطعه جاری آخرین قطعه دیتاگرام است یا آنکه قطعات دیگری در ادامه وجود دارند. البته در IPv6 برخلاف IPv4، فقط ماشین مبدا می‌تواند بسته‌ای را قطعه‌قطعه کند و مسیربایهای واقع بر روی مسیر قادر به چنین کاری نیستند. اگرچه این موضوع از لحاظ فلسفی یک واپسگرایی محسوب می‌شود ولی در عوض کار مسیربایها را ساده‌تر کرده و فرآیند مسیربایی سریعتر خواهد شد. همانگونه که قبلاً اشاره کردیم هر گاه یک مسیربای با بسته‌ای

بیش از حد بزرگ مواجه گردد آن را حذف کرده و بسته ICMP (حامل پیغام خطا و اطلاعات مفید دیگر) به مبدا آن برمی‌گرداند. اطلاعات ارسالی به مبدا بسته، امکان آنرا می‌دهد که به کمک این سرآیند، بسته را به قطعات کوچکتر تقسیم و آنها را از نو ارسال کند.

سرآیند Authentication (سرآیند احراز هویت) مکانیزمی را فراهم آورده تا گیرنده بتواند از هویت فرستنده بسته مطمئن شود. سرآیند Encrypted Security Payload اجازه می‌دهد تا محتوای بسته رمزنگاری شود و بدین ترتیب فقط گیرنده مورد نظر قادر به خواندن آنست. این گونه سرآیندها برای انجام مأموریت خود از تکنیکهای رمزنگاری بهره می‌گیرند.

اختلاف نظرها و مناقشات

نظر به آنکه فرآیند طراحی IPv6، «باز» بوده و افراد درگیر در طراحی، بر عقاید خود تأکید داشته‌اند فلذا شگفت‌آور نیست که بسیاری از گزینه‌های انتخابی در IPv6 متناقض باشند. در زیر اجمالاً برخی از آنها را بررسی خواهیم کرد. برای آگاهی از جزئیات ماجرأ به RFCهای مربوطه مراجعه نمایید.

قبلاً اشاره کردیم که بحث و جدل گسترده‌ای پیرامون طول آدرسها وجود داشت و توافق نهایی آن بود که آدرسها با طول ثابت و ۱۶ بیتی باشند.

جدل دیگری بر سر حداکثر تعداد گام (Hop Limit) در گرفت. یک گروه احساس می‌کرد که محدود کردن حداکثر تعداد گام (Hop) به ۲۵۵ یک اشتباه محض است چراکه اگرچه در آن زمان حداکثر طول مسیرها عموماً از ۳۲ تجاوز نمی‌کرد ولی مدعی بودند که ممکن است ده سال بعد مسیرها طولانی‌تر از ۲۵۵ باشند. استدلال آنها این بود که فضای آدرس ۱۶ بیتی آینده‌نگری بیش از اندازه و در عوض مقدار کم Hop Count، کوتاه‌نظری است. از دیدگاه آنها بزرگترین اشتباه یک دانشمند کامپیوتر، آنست که برای هر فیلدی، تعداد بیت کمی در نظر بگیرد.

پاسخ گروه مقابل آن بود که افزایش بی‌مورد فضای هر فیلد منجر به تشکیل یک سرآیند حجیم خواهد شد. همچنین استدلال دیگرشان آن بود که وظیفه فیلد Hop Count جلوگیری از سرگردانی بسته‌ها به مدت طولانی است و ۶۵۵۳۵ گام بیش از حد زیاد است. استدلال آخر آنکه با رشد اینترنت، لینکهای بسیار طولانی ساخته می‌شوند و این امکان فراهم می‌شود که برای رسیدن از یک کشور به کشور دیگر به کمتر از ده گام نیاز باشد. اگر یک بسته برای رسیدن از مبدا به مقصد مجبور شود از ۱۲۵ مسیر یاب بین‌المللی بگذرد، ستون فقرات این شبکه بین‌المللی در جایی اشکال دارد! بدین ترتیب طرفداران فیلد ۸ بیتی در عقیده خود پیروز شدند.

یکی دیگر از بحثهای داغ بر سر حداکثر طول بسته‌ها بود. سوپرکامپیوترها به بسته‌هایی با طول بیش از ۶۴ کیلوبایت احتیاج داشتند. وقتی یک سوپرکامپیوتر شروع به ارسال می‌کند و به کار خود مشغول می‌شود نباید به ازای هر ۶۴ کیلوبایت یکبار متوقف شود. استدلال گروه مخالف آن بود که اگر یک بسته یک مگابایتی در طول مسیر به یک خط T1 برسد آن خط به مدت حداقل ۵ ثانیه اشغال شده و کاربران دیگری که در این خط سهیم هستند با تأخیر قابل توجهی روبرو خواهند شد.^۱ توافق نهایی بدینجا ختم شد که بسته‌های معمولی حداکثر ۶۴ کیلوبایتی باشند ولی به کمک سرآیند اختیاری Hop-by-Hop بتوان جامبوگرامهایی با هر طول دلخواه ارسال کرد.

موضوع سوم مناقشه، حذف فیلد IPv4 checksum (کد تشخیص خطاهای احتمالی در سرآیند) بود. برخی از افراد حذف این فیلد را مشابه با برداشتن ترمزهای یک خودرو می‌دانستند که اگرچه ماشین را سبکبار و سریعتر می‌کند ولی اگر اتفاق غیر مترقبه‌ای رخ بدهد مشکل جدی بوجود می‌آید. لذا، گروه مقابل از آن بود که هر برنامه کاربردی که نگران صحت داده‌های خود است باید از پروتکلی در لایه انتقال بهره بگیرد که داده‌ها را از لحاظ

۱. نرخ ارسال خط T1 را 1.544 مگابیت در نظر بگیرید.

سلامت بررسی می‌کند. لذا اضافه کردن کد کنترلی دیگر به لایه IP برای کشف خطا (در حالی که هر بسته یکبار هم در لایه پیوند داده بررسی می‌شود) بیهوده و زائد است. مضاف بر آن، تجربه نشان داده بود که محاسبه جمع کنترلی (Checksum) در IPv4 هزینه بالایی دارد. در این مناقشه نیز طرفداران حذف کد کشف خطا پیروز شدند.

موضوع دیگر، بحث ماشینهای همراه بود: وقتی یک کامپیوتر قابل حمل، در نیمی از کل دنیا حرکت می‌کند (مثلاً درون هواپیما)، آیا می‌تواند با همان آدرس IPv6 قبلی، کار خود را ادامه بدهد یا آنکه مجبور به استفاده از ساختار «عامل خانگی» و «عامل خارجی» است؟ ماشینهای همراه مشکل «عدم تقارن» را به سیستم مسیریابی تحمیل می‌کنند: یک کامپیوتر همراه و کوچک بر راحتی قادر به شنیدن سیگنال قوی منتشره از مسیریاب ثابت خود هست ولی مسیریاب ثابت بر راحتی قادر به احساس سیگنال ضعیف ارسال شده توسط کامپیوتر همراه نیست. در نتیجه برخی از افراد گرایش داشتند که در IPv6 از ماشینهای همراه حمایت شود ولی تمام این تلاشها به دلیل آنکه بر روی هیچیک از طرحهای پیشنهادی توافقی بدست نیامد، با شکست مواجه گردید.

شاید بزرگترین مناقشه بر سر موضوع «امنیت» بود: همه بر این اصل که «امنیت لازم است» اشتراک نظر داشتند. دعوا بر سر چگونگی رسیدن به امنیت و محل پرداختن به آن بود. اولین محل پرداختن به امنیت لایه شبکه است. استدلال موافقین مبنی بر آن بود که پیاده‌سازی امنیت در لایه شبکه، سرویسی استاندارد فراهم می‌کند که تمام برنامه‌های کاربردی بدون هیچگونه برنامه‌ریزی قبلی می‌توانند از آنها بهره بگیرند. استدلال مخالفین نیز آن بود که برنامه‌های کاربردی امن، عموماً به هیچ مکانیزمی کمتر از رمزنگاری انتهابه‌انتها (End-to-End Encryption) احتیاج ندارند، به نحوی که پروسه مبدا، خودش داده‌های ارسالی خود را رمز کرده و پروسه مقصد آنها را از رمز خارج کند. هر چیزی کمتر از این، می‌تواند کاربر را با خطراتی مواجه کند که از اشکالات امنیتی لایه شبکه ناشی می‌شود و هیچ تقصیری از او نیست. پاسخ به این استدلال آن بود که کاربر می‌تواند امنیت لایه IP را نادیده بگیرد و کار خودش را انجام بدهد! پاسخ نهایی مخالفین نیز آن بود که افرادی که به عملکرد صحیح شبکه (در خصوص امنیت) اعتماد ندارند چرا باید هزینه پیاده‌سازی سنگین و کندی IP را بپردازند!!

یکی دیگر از جنبه‌های مربوط به امنیت این حقیقت بود که بسیاری از کشورها قوانین سخت‌گیرانه‌ای در مورد صادرات محصولات مرتبط با رمزنگاری وضع کرده‌اند. از مثالهای بارز می‌توان به فرانسه و عراق اشاره کرد که حتی استفاده از رمزنگاری در داخل را نیز محدود کرده‌اند و عموم افراد نمی‌توانند چیزی را از پلیس مخفی نگه دارند. در نتیجه هرگونه پیاده‌سازی از IP که از روشهای رمزنگاری قوی استفاده می‌کند مجوز صدور از ایالات متحده (و بسیاری از کشورهای دیگر) را نخواهد گرفت. پیاده‌سازی دو نرم‌افزار یکی برای کاربرد داخلی و یکی برای صادرات، موضوعی است که عرضه‌کنندگان صنعت کامپیوتر با آن مخالفند.

موضوعی که پیرامون آن هیچ اختلاف نظر پیش نیامد آن بود که نمی‌توان انتظار داشت صبح روز یکشنبه IPv4 را در اینترنت از کار انداخت و صبح دوشنبه IPv6 را روشن نمود. در عوض مسیریابها و ماشینهایی که به IPv6 مجهز می‌شوند به مثابه جزایر مستقل با استفاده از تونل با یکدیگر مبادله داده می‌کنند و با افزایش این جزایر، در یکدیگر ادغام شده و جزیره بزرگتری پدید می‌آید. در نهایت تمام این جزایر به هم می‌پیوندند و اینترنت کاملاً متحول می‌شود.

سرمایه‌گذاری حجمی که از قبل بر روی مسیریابهای مبتنی بر IPv4 صورت گرفته، فرآیند تغییر و تحول اینترنت را سالها به تأخیر می‌اندازد. به همین دلیل تلاش زیادی صورت می‌گیرد تا این اطمینان حاصل شود که گذار از IPv4 به IPv6 حتی الامکان بدون زحمت و گرفتاری انجام گیرد. برای کسب آگاهی بیشتر در خصوص IPv6 به مرجع (Loshin, 1999) مراجعه نمایید.

۷-۵ خلاصه

لایه شبکه به لایه انتقال خدماتی را عرضه می کند. این لایه می تواند مبتنی بر «مدارات مجازی» یا «دیتاگرام» باشد. در هر دو حالت، وظیفه اصلی این لایه آنست که بسته های داده را از مبدا به مقصد برساند. در «زیرشبکه های مدار مجازی»، تصمیم گیری در خصوص مسیر هدایت بسته ها، فقط یکبار و آن هم در حین تنظیم مدار مجازی انجام می شود. در «زیرشبکه های دیتاگرام»، این تصمیم گیری به ازای هر بسته تکرار می شود.

در شبکه های کامپیوتری از الگوریتمهای مسیریابی متنوعی استفاده می شود: الگوریتمهای ایستا نظیر «مسیریابی کوتاهترین مسیر» و «ارسال سیل آسا» (Flooding) و الگوریتمهای پویا شامل «مسیریابی بردار فاصله» و «مسیریابی حالت لینک» (Link State). در اغلب شبکه های واقعی بزرگ، از یکی از این دو روش مسیریابی پویا استفاده شده است. از موارد مهم دیگر در مسیریابی می توان به روش مسیریابی سلسله مراتبی، مسیریابی در محیط ماشینهای سنّار، مسیریابی پخش فراگیر، مسیریابی چندبخشی، و مسیریابی در شبکه های همتابه همتا اشاره کرد.

زیرشبکه ها ممکن است به سادگی با ازدحام مواجه شوند که منجر به افزایش تأخیر و کاهش ظرفیت مفید خروجی مسیریابها خواهد گردید. طراحان شبکه سعی می کنند با طراحی مناسب از بروز آن جلوگیری نمایند. این تکنیکها عبارتند از: تغییر در سیاستهای ارسال مجدد بسته ها، ذخیره سازی در حافظه نهان، کنترل جریان و نظائر آنها. به هر حال اگر ازدحام رخ داد باید به نحوی به رفع آن اقدام کرد. می توان «بسته های دعوت به آرامش» (Choke Packet) پس فرستاد، بخشی از بار را دور ریخت و یا مکانیزمی نظیر به اینها را بکار بست.

مرحله بعدی پس از حل مسئله ازدحام، تلاش در تضمین کیفیت خدمات است. بدین منظور می توان از روشهایی نظیر «بافر کردن داده ها در ماشین مشتری»، «شکل دهی به ترافیک»، «رزرو منابع» و «کنترل پذیرش» استفاده کرد. راهکارهایی که برای تضمین کیفیت خوب خدمات طراحی و پیاده سازی شده عبارتند از: خدمات مجتمع (شامل RSVP)، خدمات متمایز و MPLS.

شبکه ها از جهات متعدد با هم تفاوت دارند، لذا وقتی می خواهند به یکدیگر متصل شوند مشکلاتی رخ خواهد داد. برخی از این مشکلات را می توان با تکنیک ایجاد تونل (Tunneling) از میان شبکه واسط کاهش داد ولی اگر شبکه های مبدا و مقصد خودشان متفاوت باشند این راهکار نیز با شکست مواجه می شود. وقتی شبکه های متفاوت حداکثر طول بسته هایشان فرق کند به تکنیک قطعه قطعه سازی بسته ها نیاز خواهد شد.

در اینترنت مجموعه وسیعی از پروتکل های مرتبط با لایه شبکه وجود دارد. از بین تمام اینها، پروتکل IP انتقال واقعی بسته ها را بر عهده دارد ولیکن پروتکل های کتلی دیگر مثل ICMP، ARP و RARP و همچنین پروتکل های مسیریابی مثل OSPF و BGP در کنار آن به فرآیند هدایت و انتقال بسته ها کمک می کنند. اینترنت سریعاً با مشکل کمبود فضای آدرس مواجه شد و برای رفع این مشکل نسخه جدید IP یعنی IPv6 طراحی گردید.

مسائل

۱. دو مسأله از برنامه های کاربردی کامپیوتر ارائه بدهید که برای آنها سرویس اتصال گرا (Connection Oriented) مناسب است. دو نمونه دیگر از برنامه های کاربردی معرفی کنید که در آنها سرویس بدون اتصال مناسبتر است.
۲. آیا وضعیتی پیش می آید که در سرویس اتصال گرا بسته ها خارج از ترتیب تحویل مقصد شوند؟ پاسخ خود را تشریح کنید.
۳. زیرشبکه های دیتاگرام، هر بسته را به صورت واحدهای مجزا و مستقل از هم مسیریابی و هدایت می کنند. زیرشبکه های مدار مجازی بدین نحو عمل نمی کنند فلذا بسته های داده، مسیری از قبل تعیین شده را

- می پیمایند. آیا این تعبیر بدان معناست که زیر شبکه های مدار مجازی به این قابلیت که بتوانند بسته های مجزا و مستقل را از یک مبداء دلخواه به مقصد مورد نظر هدایت کنند نیاز ندارند؟ پاسخ خود را شرح دهید.
۴. سه نمونه از پارامترهایی را که می توان در حین تنظیم یک اتصال، بر روی آنها مذاکره و توافق کرد، نام ببرید.
۵. به مسئله زیر در خصوص پیاده سازی «سرویس مدار مجازی» دقت نمایید: اگر در درون زیر شبکه از الگوی مدار مجازی استفاده شده باشد هر بسته داده باید یک سرآیند سه بیتی داشته باشد و هر مسیریاب نیز برای شناسایی هر مدار مجازی به ۸ بایت احتیاج دارد. اگر در درون زیر شبکه از الگوی دیناگرام استفاده شده باشد، به سرآیندهای ۱۵ بیتی نیاز خواهد بود ولی در عوض مسیریاب به فضای حافظه اضافی نیاز نخواهد داشت. هزینه ارسال را برای هر 10^6 بایت یک سنت (به ازای عبور از هر مسیریاب) فرض کنید. حافظه بسیار سریع لازم برای مسیریابها را یک سنت به ازای هر بایت در نظر بگیرید که با فرض ۴۰ ساعت کار در هر هفته، در عرض دو سال مستهلک می شود. بطور متوسط هر نشست هزار ثانیه طول می کشد و در خلال نشست ۲۰۰ بسته منتقل می شود.^۱ هر بسته نیز بطور متوسط از چهار مسیریاب عبور می کند. پیاده سازی کدامیک از این الگوها ارزان تر تمام می شود و به چه میزان؟
۶. فرض کنید که تمام مسیریابها و ماشینها به درستی کار می کنند و تمام نرم افزارها از خطا مصون باشند. آیا باز هم احتمال آنکه یک بسته به اشتباه تحویل ماشینی دیگر شود وجود دارد؟ (حتی اگر این احتمال بسیار ناچیز باشد).
۷. به شبکه شکل ۵-۷ دقت کنید ولیکن وزنه های خطوط را نادیده بگیرید. فرض کنید مسیریابی طبق الگوریتم سبیل آسا (Flooding) انجام می شود. اگر بسته های ارسالی از A به D دارای «شماره گام» (Hop count) معادل ۳ باشند فهرست مسیریابی را که بسته از آنها عبور می کند، مشخص نمایید. همچنین تعیین کنید که کل گامهایی که بسته ها پیموده و پهنای باندی که به ازای آن تلف می شود چقدر است.
۸. روشی ساده و ذهنی برای پیدا کردن دو مسیر از یک مبداء مشخص به مقصدی خاص در شبکه ارائه دهید به نحوی که بتواند در مقابل از بین رفتن خطوط انتقال، دوام بیاورد. (با فرض آنکه حداقل دو مسیر وجود دارد). مسیریابها را قابل اعتماد فرض کنید لذا نگران از کار افتادن آنها نباشید.
۹. به زیر شبکه شکل ۵-۱۳ الف دقت کنید. در این زیر شبکه از «مسیریابی بردار فاصله» استفاده شده و بردارهای ذیل توسط مسیریاب C دریافت شده است:
- از B: (5,0,8,12,6,2) از D: (16,12,6,0,9,10) از E: (7,6,3,9,0,4)
- تاخیر اندازه گیری شده تا B و D و E به ترتیب ۶ و ۳ و ۵ است. جدول جدید مسیریابی در C چیست؟ خط خروجی و تاخیر تخمینی رسیدن به هر مقصد را مشخص نمایید.
۱۰. اگر میزان تاخیر به صورت اعداد هشت بیتی ذخیره شود و شبکه ۵۰ مسیریاب داشته باشد و بردارهای تاخیر در هر ثانیه دو بار مبادله شوند چه مقدار از پهنای باند یک خط دو طرفه همزمان (Full Duplex)، در اثر استفاده از الگوریتم مسیریابی توزیع شده [بردار فاصله] تلف می شود؟ فرض کنید هر مسیریاب سه خط با مسیریابهای دیگر دارد.
۱۱. در شکل ۵-۱۴، حاصل OR کردن دو مجموعه بیت های ACF با یکدیگر ۱۱۱ می شود. آیا این موضوع تصادفی است یا در هر زیر شبکه و با هر وضعیتی رخ می دهد؟
۱۲. برای مسیریابی سلسله مراتبی در زیر شبکه ای با ۴۸۰۰ مسیریاب، اندازه هر منطقه (Region) و دسته (Cluster) چقدر باشد تا جدول مسیریابی (در سلسله مراتب سه سطحی) حداقل شود؟ نقطه شروع مناسب

۱. نشست را در یک عبارت غیر فنی می توانید ارتباط و محاوره یک زوج ماشین در طول زمان فرض کنید.

آنست که فرض کنید داشتن k دسته و k منطقه و k مسیر یاب تقریباً بهینه است، فلذا k تقریباً ریشه سوم 4800 (تقریباً ۱۶) می شود. با سعی و خطا در همسایگی عدد ۱۶، بهترین تعداد دسته، منطقه و مسیر یاب را پیدا کنید.

۱۳. در متن گفته شد که وقتی ماشین متحرک در محل همیشگی خود نیست، بسته های ارسالی به سوی LAN خانگی، متوقف و به سمت LAN موقت او تغییر مسیر داده می شود. در یک شبکه IP که بر روی اترنت پیاده شده، این تغییر مسیر چگونه انجام می شود؟

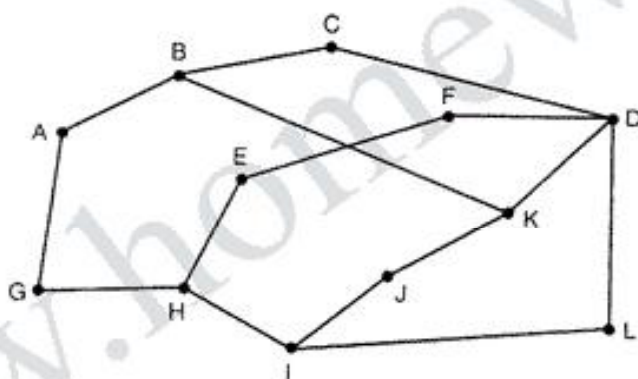
۱۴. با در نظر گرفتن زیر شبکه شکل ۵-۶، اگر B بسته ای را به صورت فراگیر پخش کند، چه تعداد بسته در زیر شبکه تولید می شود اگر:

الف) اگر از روش هدایت در مسیر معکوس (Reverse Path Forwarding) استفاده شود؟

ب) از روش Sink Tree استفاده شود؟

۱۵. شبکه شکل ۵-۱۶ الف را مدنظر قرار بدهید. فرض کنید که خط جدیدی بین F و G اضافه می شود ولی کماکان Sink Tree شکل ۵-۱۶ الف بی تغییر باقی می ماند. در شکل ۵-۱۶ ج چه تغییری حاصل می شود؟

۱۶. برای زیر شبکه بعدی درخت پوشای چند پخششی (Multicast Spanning Tree) را برای مسیر یاب C ترسیم کنید به نحوی که اعضای گروه مورد نظر مسیر یابهای A و B و C و D و E و F و I و K باشند.



۱۷. در شکل ۵-۲۰، آیا گره های H یا I در جستجویی که از مبدا A شروع شده، هیچگونه ارسال فراگیر داشته اند؟

۱۸. در شکل ۵-۲۰ فرض کنید که گره B از نو راه اندازی شده است و هیچگونه اطلاعات مسیر یابی در جدول خود ندارد. این گره به ناگاه احتیاج به یافتن مسیری به H پیدا می کند و بسته هایی پخششی (Broadcast) با مقدار TTL معادل ۱ و ۲ و ۳... را منتشر می کند. این کار چند دور طول می کشد تا نهایتاً مسیری پیدا شود؟

۱۹. در ساده ترین نسخه الگوریتم Chord برای «جستجوی همتایه همتا» (Peer-to-Peer)، برای جستجو از جدول Finger استفاده نمی شود و به جای آن، جستجو به صورت خطی (یعنی گره به گره) در دو جهت بر روی دایره انجام می گیرد. آیا یک گره می تواند به درستی حدس بزند که جستجو را باید در کدام یک از جهات انجام بدهد؟ پاسخ خود را شرح بدهید.

۲۰. دایره Chord نشان داده شده در شکل ۵-۲۴ را مدنظر قرار بدهید. فرض کنید که گره ۱۰ به ناگاه فعال و وارد خط می شود. آیا ورود این گره در جدول Finger از گره ۱ تأثیری می گذارد و اگر این چنین است چگونه؟

۲۱. به عنوان یک مکانیزم کنترل ازدحام در زیر شبکه ای که در درون از مدار مجازی بهره گرفته است، هر مسیر یاب می تواند از اعلام وصول یک بسته (Ack) طفره برود مگر آنکه: (۱) آگاه شود که آخرین ارسال او

بر روی مدار مجازی، با موفقیت دریافت شده است و (۲) بافر آزاد داشته باشد. برای سادگی فرض کنید مسیریاب برای اعلام وصول از پروتکل «توقف و انتظار» (Stop & Wait) استفاده کرده و برای هر مدار مجازی یک بافر اختصاصی (برای ترافیک هر یک از دو جهت) اختصاص داده است. اگر انتقال هر بسته T ثانیه طول بکشد (انتقال بسته یا پیام Ack به یک اندازه طول می کشد) و n مسیریاب بر روی مسیر قرار گرفته باشد، در چنین شرایطی بسته ها با چه نرخ تحویل ماشین مقصد می شوند؟ فرض کنید که خطای انتقال بسیار ناچیز و اتصال بین ماشین و مسیریاب بی نهایت سریع است.

۲۲. در یک زیرشبکه دیتاگرام به مسیریابها اجازه داده شده تا در صورت نیاز بسته ها را حذف نمایند. احتمال حذف بسته در هر مسیریاب را p فرض کنید. حالتی را در نظر بگیرید که ماشین مبدا به مسیریاب مبدا، مسیریاب مبدا به مستقیماً به مسیریاب مقصد و مسیریاب مقصد نیز مستقیماً به ماشین مقصد متصل شده است. اگر یکی از این مسیریابها بسته ای را حذف نمایند، نهایتاً مهلت ماشین مبدا به سر می رسد و آن را از نو ارسال می نماید. اگر خط بین ماشین و مسیریاب را مثل خط بین دو مسیریاب، یک «گام» (Hop) فرض کنیم، مقدار هر یک از موارد زیر را حساب کنید:

الف) میانگین تعداد گامی که یک بسته در هر بار ارسال طی می کند.

ب) میانگین دفعات ارسال یک بسته

ج) میانگین گامهایی که یک بسته دریافتی طی کرده است.

۲۳. دو تفاوت عمده روش «بیت هشدار» (Warning Bit) و روش RED را توضیح بدهید.

۲۴. دلیلی ارائه بدهید که چرا در الگوریتم «سطل سوراخ»، در هر تیک ساعت مجوز ارسال یک بسته صادر می شود، فارغ از آنکه طول بسته چقدر است.

۲۵. گونه ای از الگوریتم سطل سوراخ که مبتنی بر شمارش بابت است در سیستم خاصی بکار گرفته شده است. قاعده آنست که در هر تیک ساعت یک بسته ۱۰۲۴ بایتی یا دو بسته ۵۱۲ بایتی یا معادل آن، ارسال شود. یکی از محدودیتهای جدی چنین سیستمی را که در متن درس بدان نیرداخته ایم، بیان کنید.

۲۶. در یک شبکه ATM از روش سطل نشانه دار برای شکل دهی به ترافیک استفاده شده است. در هر پنج میکروثانیه یک توکن جدید در سطل قرار داده می شود. هر توکن برای ارسال یک سلول حاوی ۴۸ بایت داده، مناسب می باشد. حداکثر نرخ ارسال مجاز چقدر است؟

۲۷. ترافیک یک کامپیوتر متصل به شبکه 6 Mbps طبق الگوریتم سطل نشانه دار، شکل دهی و منظم می شود. سطل نشانه دار با نرخ یک مگابیت بر ثانیه پر می شود. ظرفیت سطل هشت مگابیت است که در همان ابتدا پر شده است. کامپیوتر در چه مدت می تواند با سرعت ۶ مگابیت بر ثانیه ارسال داشته باشد؟

۲۸. فرض کنید در توصیف مشخصات یک «جریان» حداکثر طول هر بسته ۱۰۰۰ بایت، نرخ سطل نشانه دار ده میلیون بایت در ثانیه، اندازه سطل نشانه دار یک میلیون بایت و حداکثر نرخ ارسال ۵۰ میلیون بایت در ثانیه تعیین شده است. یک ماشین در چه مدتی قادر به ارسال انفجاری داده ها با حداکثر سرعت خود می باشد؟

۲۹. شبکه شکل ۳۷-۵ از RSVP و درخت چندپخشی نشان داده شده برای ماشینهای ۱ و ۲، بهره گرفته است. فرض کنید که ماشین ۳ برای دریافت یک «جریان» از ماشین ۱، تقاضای کانالی با پهنای باند 2MB/sec می دهد. همچنین برای دریافت جریانی از ماشین ۲ تقاضای 1MB/sec می دهد. بطور همزمان ماشین ۴، 2MB/sec برای دریافت جریان از ماشین ۱ و همچنین ماشین ۵ برای دریافت جریان از ماشین ۲، 1MB/sec پهنای باند درخواست می کنند. مجموع کل پهنای باند رزرو شده در مسیریابهای A و B و C و E و H و J و K و L برای این درخواستها چقدر است؟

۳۰. بردازنده یک مسیریاب قادر به پردازش ۲ میلیون بسته در هر ثانیه است. بار عرضه شده به آن ۱/۵ میلیون بسته در ثانیه است. اگر در مسیر رسیدن از مبدا به مقصد، ۱۰ مسیریاب قرار گرفته باشند، تأخیر ناشی از صفبندی و پردازش بسته چقدر است؟
۳۱. فرض کنید کاربری از خدمات متمایز و «هدایت پرشتاب» بهره می‌گیرد. آیا تضمینی وجود دارد که بسته‌های پرشتاب تأخیری کمتر از بسته‌های معمولی داشته باشند؟ دلیل خود را تشریح کنید.
۳۲. آیا به فرآیند قطعه‌قطعه‌سازی بسته‌ها در شبکه‌های بهم متصل شده مدار مجازی نیز احتیاج است یا آنکه این نیاز فقط در شبکه‌های دیتاگرام وجود دارد؟
۳۳. فرآیند ایجاد تونل از میان یک شبکه الحاق شده مدار مجازی ساده و سراسر است: مسیریاب چند پروتکلی در یکی از دو طرف یک مدار مجازی با مسیریاب طرف دیگر ایجاد و تنظیم می‌کند و بسته‌ها را از طریق این مدار مجازی مبادله می‌نماید. آیا ایجاد تونل از میان یک شبکه دیتاگرام نیز ممکن است؟ اگر جواب مثبت است چگونه؟
۳۴. فرض کنید که ماشین میزبان A به مسیریاب R1، R1 نیز به مسیریاب دیگری به نام R2، و R2 نیز به ماشین میزبان B متصل است. فرض کنید که یک پیام TCP حاوی ۹۰۰ بایت داده و ۲۰ بایت سرآیند، جهت تحویل به B به نرم‌افزار IP در ماشین A تسلیم می‌شود. محتوای فیلدهای Identification، Total Length، DF، MF و Fragment offset را در هر بسته IP که از یکی از سه لینک A-R1، R1-R2، R2-B عبور می‌کند، مشخص نمایید؛ لینک A-R1 می‌تواند از فریمی با طول حداکثر ۱۰۲۴ بایت (که ۱۴ بایت آن هم سرآیند فریم لایه پیوند داده‌ها محسوب می‌شود) پشتیبانی می‌کند. لینک R1-R2 می‌تواند از فریمی با طول ۵۱۲ بایت حمایت کند (با احتساب ۸ بایت سرآیند فریم) و لینک R2-B از فریمهایی با طول حداکثر ۵۱۲ بایت (با احتساب ۱۲ بایت سرآیند فریم) پشتیبانی می‌کند.
۳۵. یک مسیریاب بسته‌های IP با طول ۱۰۲۴ بایت (با احتساب داده و سرآیند) در خروجی خود گسیل می‌دارد. با فرض آنکه بسته‌ها فقط برای ده ثانیه زنده می‌مانند، مسیریاب حداکثر با چه سرعتی می‌تواند بسته‌ها را بفرستد بدون آنکه خطر به صفر برگشتن فیلد شماره شناسایی دیتاگرام (Datagram ID) وجود داشته باشد؟
۳۶. یک دیتاگرام IP که از گزینه Strict Source Routing استفاده کرده، مجبور است قطعه قطعه شود. به نظر شما آیا این گزینه بایستی در تمام قطعات آن قرار داده شود یا آنکه قرار دادن این گزینه در قطعه اول کفایت می‌کند؟ پاسخ خود را شرح بدهید.
۳۷. فرض کنید در کلاس B به جای ۱۶ بیت برای شماره شبکه، ۲۰ بیت در نظر گرفته می‌شد. در چنین حالتی چند شبکه کلاس B می‌توان داشت؟
۳۸. آدرس IP با نمایش هگزادسیمال C22F1582 را به نماد نقطه‌دار تبدیل کنید.
۳۹. شبکه‌ای در اینترنت، از الگوی زیر شبکه 255.255.240.0 استفاده کرده است. این شبکه حداکثر چند ماشین میزبان می‌تواند داشته باشد؟
۴۰. تعداد بسیار زیادی آدرس IP متوالی از نقطه شروع 198.16.0.0 در اختیار می‌باشد. فرض کنید چهار سازمان A و B و C و D به ترتیب ۴۰۰۰، ۲۰۰۰، ۴۰۰۰ و ۸۰۰۰ آدرس IP درخواست می‌کنند. برای هر یک از اینها اولین آدرس IP، آخرین آدرس IP و الگوی زیر شبکه را به شکل w.x.y.z/s، معین کنید.
۴۱. یک مسیریاب آدرسهای IP جدیدی طبق فهرست زیر دریافت می‌کند (جهت درج در جدول مسیریابی):
57.6.120.0/21 و 57.6.112.0/21 و 57.6.104.0/21 و 57.6.96.0/21
- اگر برای رسیدن به تمام این شبکه‌ها از خط مشترکی استفاده شود، آیا می‌توان این آدرسها را «تجمیع»

(Aggregate) کرد؟ اگر می توان به چه نحو؟ اگر خیر، چرا؟

۴۲. مجموعه ای از آدرسهای IP از 29.18.0.0 تا 29.18.128.255 به صورت 29.18.0.0/17 «تجمیع» شده اند. ولیکن یک فاصله خالی ۱۰۲۴ نایی متناسب نشده در محدوده 29.18.60.0 تا 29.18.63.255 وجود داشته که به ناگاه به شبکه ای اختصاص داده می شود که خط خروجی رسیدن به آن، باقیه فرق می کند. آیا در اینجا باید فضای تجمیع شده آدرسها را مثل اول به بلوکهای اولیه تقسیم کرد و بلوک جدید به جدول اضافه و فرآیند تجمیع از نو انجام گیرد؟ اگر نه چه کار دیگری می توان انجام داد؟
۴۳. یک مسیریاب در جدول مسیریابی خود درایه های CIDR زیر را در اختیار دارد:

Address/mask (آدرس/الگوی زیر شبکه)	Next Hop (گام بعدی)
135.46.56.0/22	Interface 0
135.46.60.0/22	Interface 1
192.53.40.0/23	Router 1
default	Router 2

اگر بسته ای با یکی از آدرسهای IP زیر دریافت شود، مسیریاب با آن بسته چه می کند:

- الف) 135.46.63.10
ب) 135.46.57.14
ج) 135.46.52.2
د) 192.53.40.7
ه) 192.53.56.7

۴۴. سیاست بسیاری از شرکتها بر آنست که دو (یا چند) مسیریاب متصل به اینترنت در شبکه خود داشته باشند تا در صورت از کار افتادن یکی از آنها، از دیگری استفاده شود. آیا با چنین سیاستی باز هم می توان از NAT بهره گرفت؟ پاسخ خود را تشریح کنید.
۴۵. فرض نمایید پروتکل ARP را برای دوستان توضیح داده اید. پس از اتمام توضیحاتتان او می گوید: «متوجه شدم: ARP سرویسی را به لایه شبکه ارائه می دهد و طبعاً جزئی از لایه پیوند داده ها است!» چه حرفی برای گفتن به او دارید؟
۴۶. ARP و RARP هر دو آدرسهای را از یک فضای فضا به دیگری می نگارند (IP به MAC و بالعکس). از این دیدگاه هر دو مشابه یکدیگر هستند، ولی پیاده سازی آنها متفاوت از یکدیگر است. عمده ترین تفاوت آنها در چیست؟
۴۷. روشی را برای بازسازی قطعات یک بسته IP در مقصد، معرفی و تشریح نمایید.
۴۸. اغلب الگوریتمهای بازسازی دیتاگرامهای IP، دارای زمان سنج (تایمر) خاصی هستند تا در صورت از بین رفتن یک قطعه، بقیه قطعات تا ابد در بافر نمانند. فرض کنید یک دیتاگرام به چهار قطعه تقسیم شده است. سه قطعه اول سر موقع دریافت می شوند ولی قطعه آخر با تأخیر مواجه می شود. عاقبت مهلت دریافت آن منقضی شده و سه قطعه دیگر از حافظه گیرنده، پاک می گردد. اندکی بعد، آخرین قطعه از راه می رسد. گیرنده با آن چه باید بکند؟
۴۹. چه در IP و چه در ATM، کد کشف خطای checksum فقط سرآیند بسته را در بر می گیرد نه بخش داده را. فکر می کنید منطق این طراحی چه بوده است؟
۵۰. شخصی که در بوستون زندگی می کند در سفری به مینیاپولیس کامپیوتر کیفی خود را به همراه می برد. خوشبختانه، شبکه LAN او در مینیاپولیس یک شبکه محلی بی سیم و مبتنی بر IP است و او مجبور به وصل

- کابل به جایی نیست. آیا برای آنکه ترافیک پست الکترونیکی یا دیگر داده ها به درستی تحویل او شود، کماکان لازم است که از دو عامل خانگی و خارجی استفاده شود؟
۵۱. آدرسها در IPv6 شانزده بیتی هستند. اگر در هر پیکوتانیه یک بلوک یک میلیون تایی آدرس IP رزرو شود چه مدت طول می کشد تا آدرسها تمام شوند؟
۵۲. فیلد پروتکل که در IPv4 وجود داشت اکنون در سرآیند ثابت IPv6 وجود ندارد. به چه دلیل؟
۵۳. آیا وقتی که پروتکل IPv6 به صحنه عمل آمد نیازی به تغییر در ARP هست؟ اگر این چنین است آیا این تغییرات در عملکرد و مفهوم است یا در پیاده سازی فنی؟
۵۴. برنامه ای بنویسید که فرآیند مسیریابی به روش سیل آسا را شبیه سازی نماید. هر بسته دارای یک فیلد شمارنده است که به ازای هر گام (Hop) یک واحد از آن کم می شود. وقتی این شمارنده به صفر برسد، بسته حذف می گردد. زمان را «گسته» (Discrete) فرض کنید و در هر واحد زمان فقط یک بسته بر روی خط ارسال یا دریافت می شود. برنامه خود را در سه نسخه تهیه کنید: (۱) بسته ورودی بر روی تمام خطوط ارسال گردد. (۲) بسته ورودی بر روی تمام خطوط به استثنای خطی که از آن داخل شده، فرستاده شود. (۳) بسته ورودی بر روی k تا از بهترین خطوط (به صورت آماری و غیر دقیق) ارسال گردد. روش سیل آسا را با روش معمولی (یعنی وقتی بسته ورودی بر روی بهترین خط خروجی ارسال می شود) از لحاظ پهنای باند و تأخیر مقایسه نمایید.
۵۵. برنامه ای بنویسید که یک شبکه کامپیوتری را به صورت زمان گسته شبیه سازی کند. در هر واحد زمان، بسته ای که سر صف هر یک از خطوط خروجی مسیریابها قرار گرفته اند، ارسال و یک گام جلو می روند. هر مسیریاب فضای بافر محدودی دارد. اگر بسته ای دریافت شود ولی جایی برای آن وجود نداشته باشد، حذف شده و از نو ارسال نخواهد شد. در عوض یک پروتکل انتها به انتها (End to End) وجود دارد که در صورت ارسال داده ای که دریافت آن تصدیق نشده از مبدا، داده ها را مجدداً ارسال می کند. توان مفید (ظرفیت مفید یا Throughput) شبکه را برحسب تابعی از زمان انقضای مهلت (Timeout) و پارامتر نرخ خطا ترسیم نمایید.
۵۶. تابعی بنویسید که عملیات هدایت (Forwarding) را در یک مسیریاب مبتنی بر IP انجام بدهد. این تابع یک پارامتر ورودی دارد و آن هم آدرس IP است. فرض کنید این تابع به یک جدول مسیریابی سراسری دسترسی دارد. این جدول آرایه ای است که سه ستون دارد و محتویات هر ستون، اعداد صحیح هستند. این ستونها عبارتند از: آدرس IP، الگوی زیرشبکه (Subnet Mask) و شماره خط خروجی مورد استفاده [برای رسیدن به شبکه ای با آدرس IP متناظر]. این تابع، باید آدرس IP دریافتی از پارامتر ورودی را در جدول فوق الذکر به روش CIDR جستجو کرده و شماره خط خروجی متناسب با آن را به عنوان مقدار برگشتی باز گرداند.
۵۷. با استفاده از فرامین اجرایی traceroute (در یونیکس) یا tracert (در ویندوز)، مسیر رسیدن از کامپیوتر خودتان به چند دانشگاه در دیگر قاره ها را پیدا کنید. برخی از سایتهای متعلق به دانشگاههای معروف که می توانید در بررسی خود از آنها استفاده کنید، عبارتند از:

www.berkeley.edu	(کالیفرنیا)
www.mit.edu	(ماساچوست)
www.vu.nl	(آمستردام)
www.ucl.ac.uk	(لندن)
www.usyd.edu.au	(سیدنی)
www.u-tokyo.ac.jp	(توکیو)
www.uct.ac.za	(کیپ تاون)