

第四章 网络层：IP 路由原理

4.1 路由、路由表与静态路由

路由(route)就是分组在因特网上从一台计算机传输到另一台计算机的实际路径。这一概念看似简单，但读者应该明确两个要点：

第一，由于因特网是分组交换网络，因此每个 IP 分组都有其独立的路由；

第二，因特网是依靠路由器将物理网络连接而成的逻辑网络，在这一图像的背景上，路由实际上是 IP 分组所经过的一连串路由器的序列。

路由的概念毕竟是简单而直观的，真正微妙的问题在于：

因特网上的路由器是怎样为每个 IP 分组找到正确的路由的(routing)？

答案是：因特网上的每个路由器都存储有一张表，称为**路由表**(routing table)，
路由器正是依据路由表的内容将各个 IP 分组转发到正确的去处。

一个概念路由表的例子

请读者对照图 4-1 的网络拓扑仔细阅读表 4-1，不难领会出路由表表达路由信息的方式。

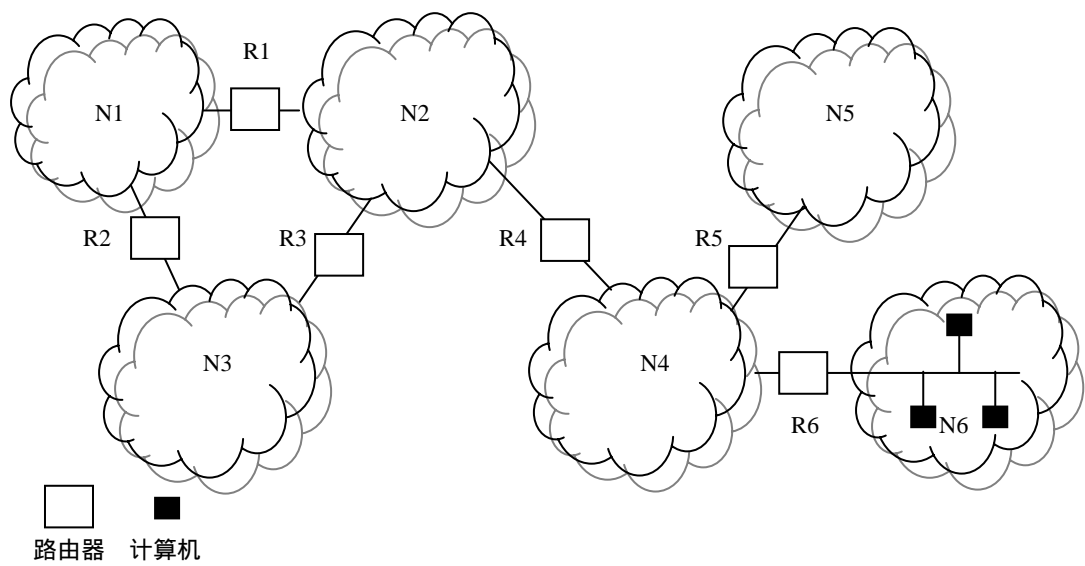


图 4-1 网络及路由器

表 4-1 路由器 R3 上的逻辑路由表

表 项	涵 义
(N1 , R2)	从 R3 到达 N1 的最优路径是经过 R2
(N2 , -)	N2 与 R3 邻接
(N3 , -)	N3 与 R3 邻接
(N4 , R4)	从 R3 到达 N4 的最优路径是经过 R4
(N5 , R4)	从 R3 到达 N5 的最优路径是经过 R4
(N6 , R4)	从 R3 到达 N6 的最优路径是经过 R4

路由表小结

一个路由器上的路由表反映的是该路由器与相邻路由器的连接关系，以及任何一个可达网络与某个邻接路由器的“目标-经过”关系；

一个路由器上的路由表从来不会去反映该路由器与任何非邻接路由器的连接关系。

每个表项 (D, N) 中的 D 多数情况下是目标网络的 IP 地址，而非目标计算机的 IP 地址，这样做的目的是为了最大限度地提高路由表示的效率。

表 4-1 表达的是概念化的路由表，路由器上实际存储的路由表要复杂得多，而且各个路由器开发商都有自己独特的实现方式，但是都较普遍地包含以下五个属性，这些也是任何路由表表项最重要的属性：

目标网络的IP地址：一个 32 位数¹。

目标网络的子网掩码：一个 32 位数。

下一步站 IP 地址：一个 32 位数。

离出接口名字：一个设备名字。在路由器上，每个网络接口都是一个唯一命名的设备，离出接口指示 IP 分组要去到下一站 IP 地址，应该从本机的哪一个接口被发送出去。注意接口名字仅仅对本路由器有意义，没有任何全局涵义。

本路由器到达目标网络的路由的度量：这个属性反映该路由的优劣，随路由协议的不同，该属性的具体数据是不同的，如有些协议使用路径上的路由器数量，有些协议使用路径上的当前往返时间、最窄链路带宽等。

一个实际路由表的例子

以图 4-2 中的网络为例说明实际路由表的内容。注意图 4-2 在每个路由器的网络接口旁边不仅标出了 IP 地址，还标出了接口的名字，如 R3 有两个接口 I31 和 I32。路由器 R2 上的路由表如表 4-2，其中距离属性表示路径上要经过的路由器的个数，下一站 IP 地址的属性值 C 表示目标网络是邻接网络。

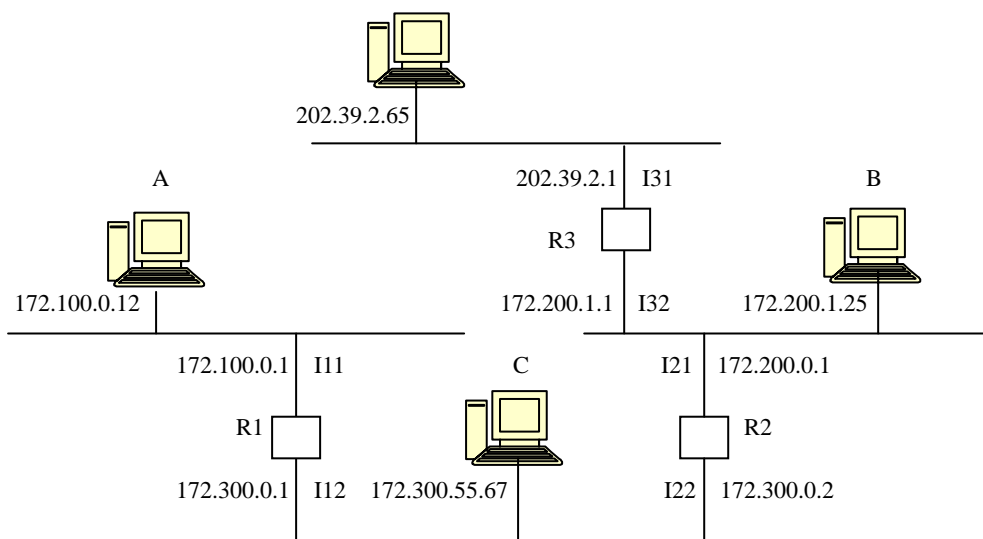


图 4-2 一个示例网络

表 4-2 路由器 R2 上的路由表

目标网络地址	掩 码	下一站 IP 地址	离出接口	距 离
172.100.0.0	255.255.0.0	172.300.0.1	I22	1
172.300.0.0	255.255.0.0	C	I22	0
172.200.0.0	255.255.0.0	C	I21	0
202.39.2.0	255.255.255.0	172.200.1.1	I21	1
0.0.0.0	0.0.0.0	172.200.1.1	I21	-

表中的最后一项，目标地址是 0.0.0.0，这是所谓**默认路由项**，表示一旦路由器不知道应该向哪里转发 IP 分组，就按照该表项指示的下一站转发。例如，表 4-2 表明如果路由器 R2 接收到任何目标地址不同于 172.100.0.0、172.200.0.0、172.300.0.0 或 202.39.2.0 的 IP 分组，则将其转发到路由器 R3。

4.2 路由器与 IP 分组转发算法

路由器的功能是实现任何类型的网络之间的互联，结果是构造出一个统一的逻辑网络，并提供通用的网络传输服务。在具体的机理上，路由器实现网络互联的方式就是进行 IP 分组的正确转发。

路由器是一种专用的网络设备，从构成上看，它是一个完整的计算机系统，包括硬件、操作系统和应用软件，如图 4-3。

与通用计算机系统的不同之处在于，路由器硬件以通讯处理为主要设计目标，特别是 I/O 部分；路由器操作系统也主要针对协议处理及通讯功能进行了优化，这一点与支持通用计算的分时系统如 Windows NT、UNIX 不同，而是更偏向于实时处理能力。

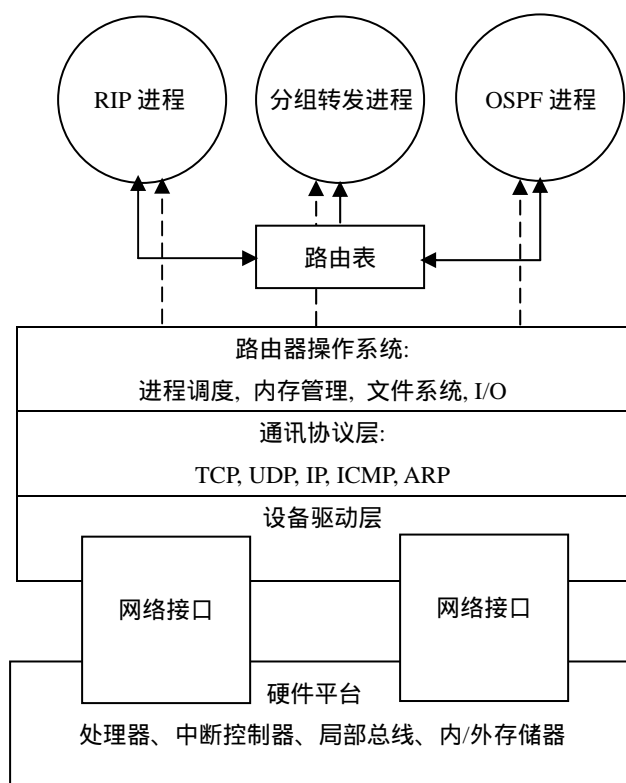


图 4-3 路由器硬件及软件的逻辑结构
虚线表示控制流，实线表示数据流

IP 分组转发算法的逻辑结构：

```

IP_Packet_Routing(IP 分组 P) {
    int found = FALSE;
    A=P 的目标 IP 地址;
    for( 每个非默认路由项(D, M, N, I, d) ) {
        /*D 是目标网络的 IP 地址, M 是子网掩码, N 是下一站路由器的 IP 地址,
        I 是离出口, d 是路由的性能度量 */
        if(A&M==D ^) { /* & 是逐位逻辑与 */
            if( d 表示 P 的目标网络并非邻接) { /* 例如, d 表示网络距离且 d>0. */
                将队列项[N, P]放入接口 I 的离出口;
                /*[N, P]的涵义是: 将分组 P 载入数据链路帧, 帧的目标 MAC 地址
                与 IP 地址 N 对应. */
            } else { /* d 表示目标网络就是接口 I 连接的邻接网段, 例如,
            d 表示网络距离且 d=0. */
                将队列项[A, P]放入接口 I 的离出口; /* 涵义同上 */
            }
            found = TRUE;
            break;
        } /* if... */
    } /* for... */
    if( ! found ) { /* 假设默认路由项是(0.0.0.0, 0.0.0.0, N*, I*, -) */
        将队列项[N*, P]放入接口 I*的离出口; /*涵义同上*/
    }
}

```

IP_Packet_Routing 算法被分组转发进程调用：

```

while(1) { /*无限循环*/
    P=get_IP_Packet(); /*取一个到达的 IP 分组*/
    IP_Packet_Routing(P); /*路由检索*/
    output_IP_Packet(P); /*发送分组*/
}

```

最后要强调一点，IP 分组在接受路由器处理、经过一个一个路由器传输的过程中，其源 IP 地址和目标 IP 地址始终不变，发生变化的是其承载帧的源、宿 MAC 地址，请读者务必要在概念上非常清楚。

4.3 动态路由的基本概念

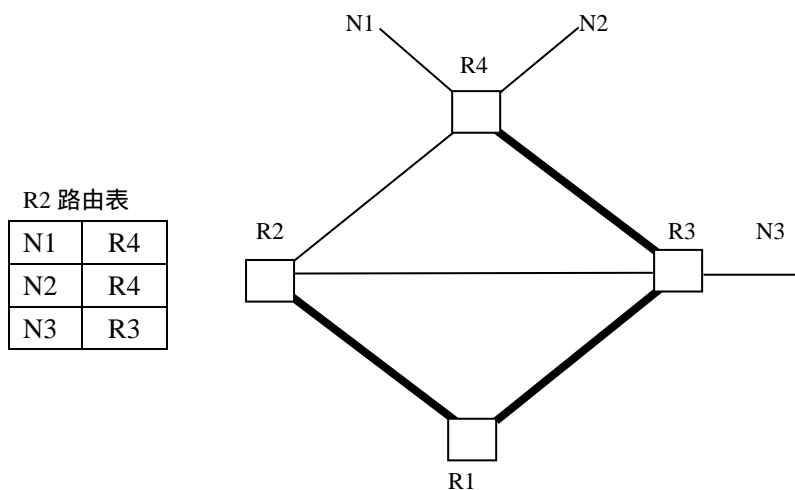


图 4-4 一个需要动态路由的例子

请读者自行写出 R1、R3、R4 上的路由表。问题：

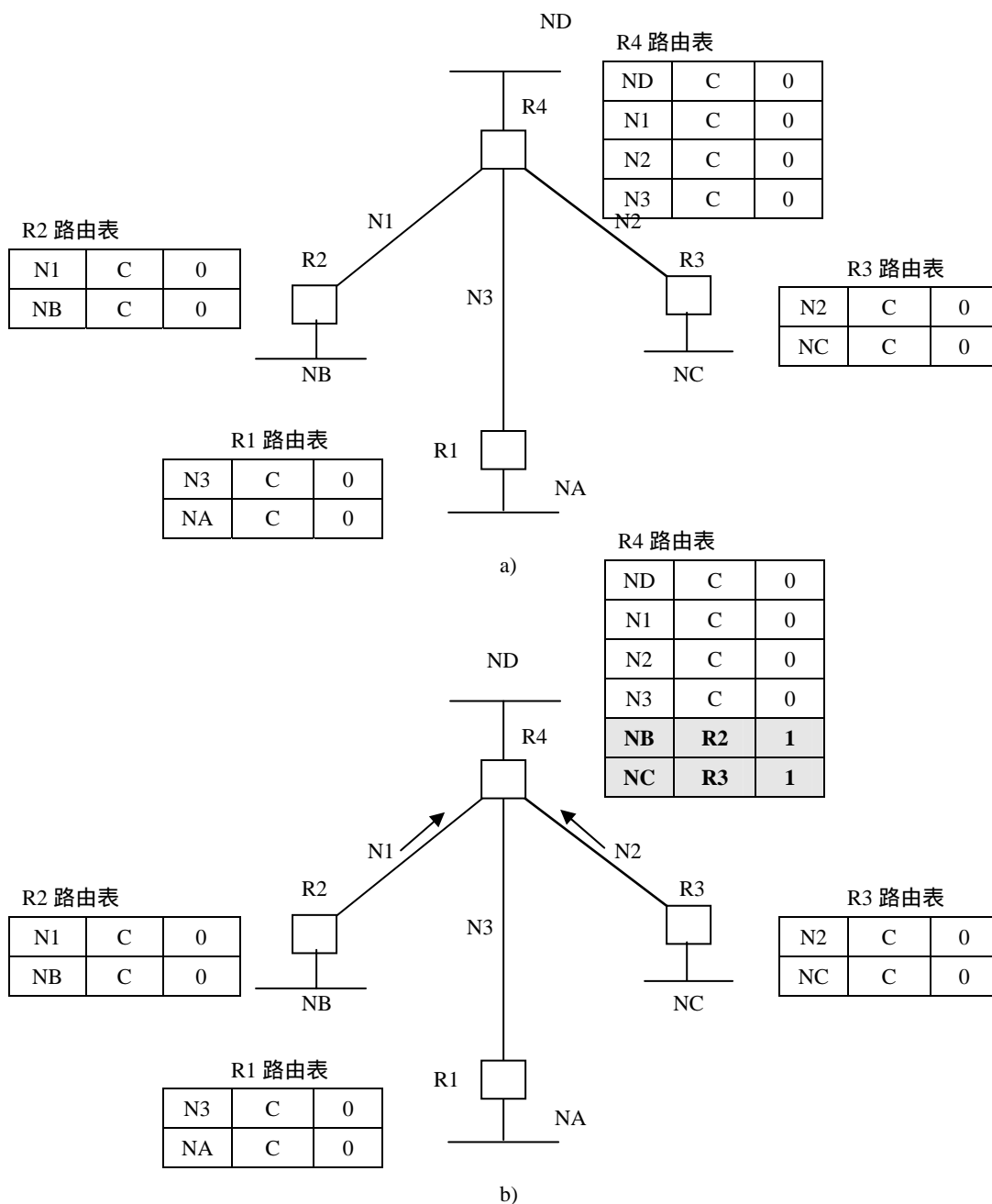
第一，可否设计一种机制使各个路由器能自己根据某些不完整的信息“学习”出较复杂的路由？

第二，以图 4-4 中 R2 的路由表为例，它表明从 R2 到 N1、N2 的路径是经过 R4，如果 R2-R4 这段链路失效，显然网络中还存在可以达到 N1、N2 的其它路径，例如粗线所示的路径，可否设计一种机制使路由器自动发现这条路径而将原来失效的路由“摆动”到当前的新路由？

如果能做到这两点，对提高网络系统适应变化的能力和容错能力是非常有意义的，特别是对那些具有复杂拓扑结构的、大规模的 IP 网络，例如当今的许多企业网，是非常有价值的。

因特网实现这种功能的具体方法，就是各类动态路由协议。动态路由协议本质上都是依靠各个路由器上运行的特殊进程在路由器-路由器之间不断交换路由信息来推断、学习出网络的全局路由。

路由学习的例子：



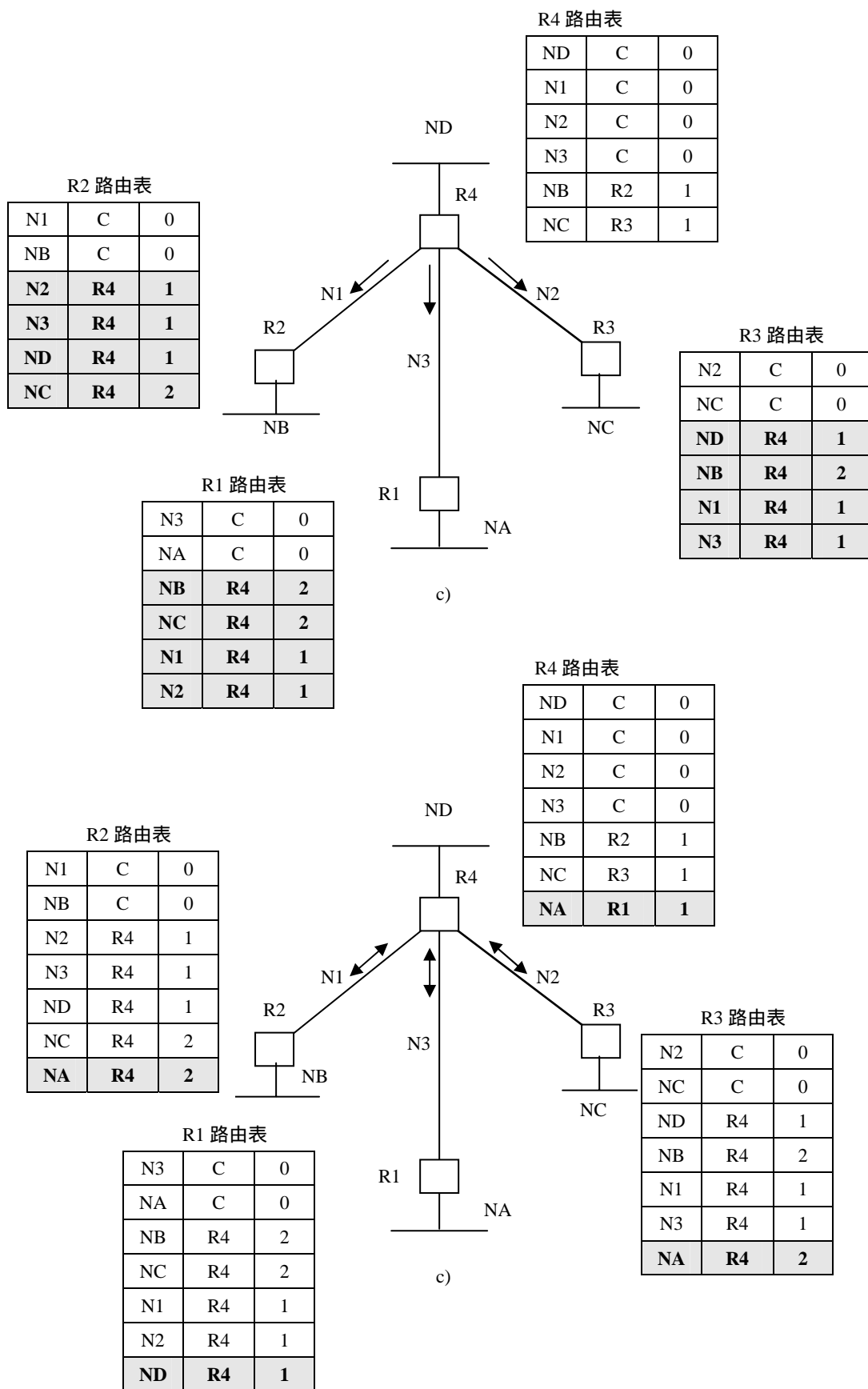


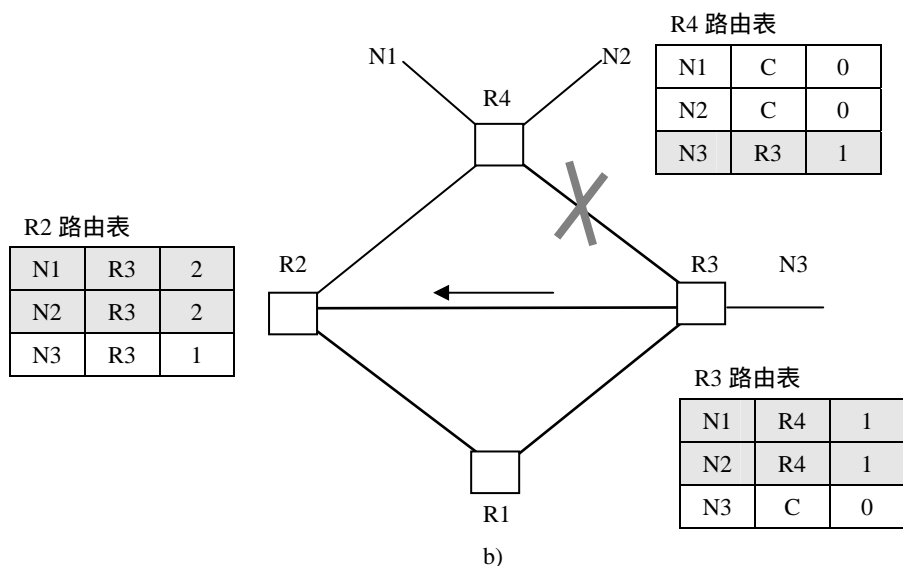
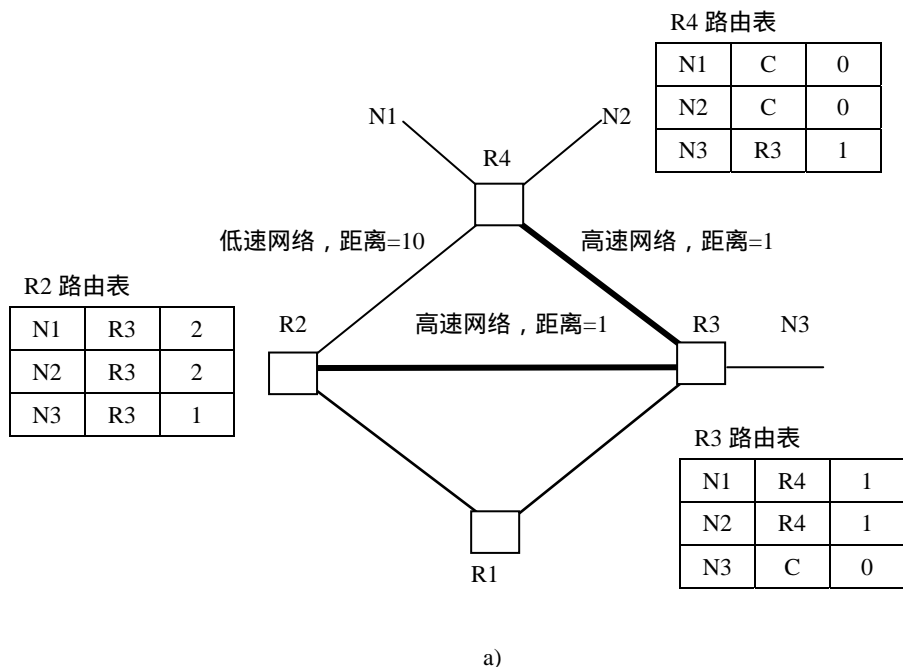
图 4-5 动态路由的工作过程

链路失效后重新建立新路由的例子

图 4-6(a)是一个路由稳态，根据图中的路由表，R2 到达 N1、N2 的路由经过 R3。

图 4-6(b)是 R3、R4 之间的邻接网络失效的情形，显然这时 R2 经 R3 到达 N1、N2 的路由不存在了。这引起两个问题：R2 如何识别出路由失效？R2 又如何学习出一条新路由？

这两个问题的答案都在于前面已经阐述过的路由报告机制。在前面的例子中已经提到，即使达到了路由稳态，路由报告仍然周期性地地进行，只不过不再引起路由更新，但它却有一个重要功能：每个路由器向其邻接的所有路由器表明当前自己的存在(alive)，一旦连续几个周期收不到一个邻接路由器的报告，其它路由器将判定连接该路由器的链路已经失效，这意味



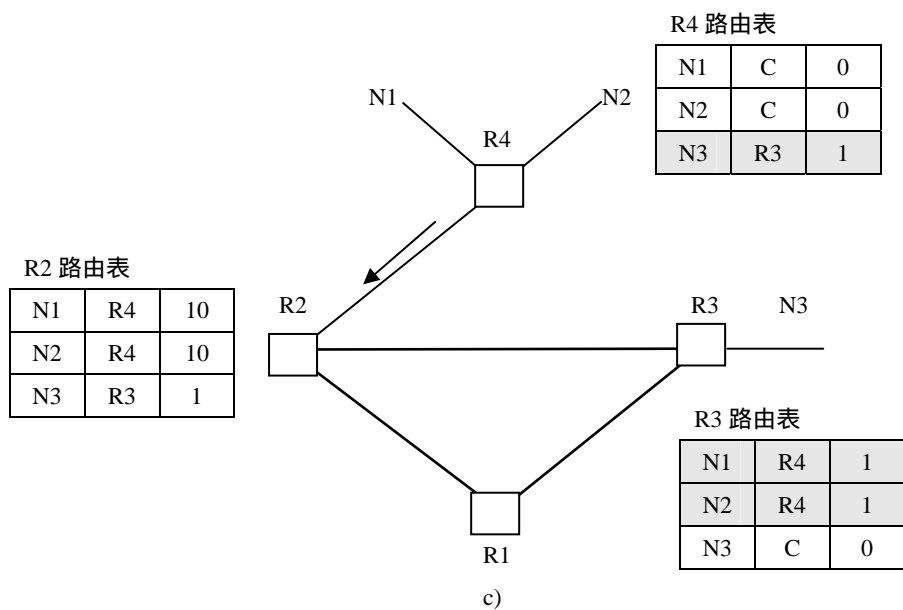


图 4-6 路由失效后收敛到新路由的例子

4.4 距离-矢量动态路由协议: RIP

*4.4.1 距离-矢量路由更新算法

距离-矢量路由协议的工作机理实际上就是图 4-5 中的过程所描述的那样，各个路由器从初始状态开始，通过与近邻路由器不断交换路由报告来学习新路由。在距离-矢量路由协议中，一个路由器 R 发布的路由报告是一组表项(N, D)，N 是一个目标网络，D 是 R 到达 N 的距离最短的路径上所要经过的路由器的个数。对每个目标网络，距离-矢量路由协议需要从接收到的路由报告中计算出最短路由上的下一站路由器。如果有多条距离最短的路由，协议将只取其中一条。用于距离-矢量动态路由协议的一般算法的 C-伪代码如下：

```
/* T 是来自邻接路由器 R 的路由报告;
   本路由器是R0，其当前路由表是T0 */
Vector_Distance_Shortest_Route_Discover(R, T) {
    for( T的每一项(Nk, Dk)) {
        if(T0中不存在形如(Nk, . , . )的表项) {
            insert(T0, (Nk, R, Dk+1)); /* R0发现了一个新目标网络及其路由*/
            continue;
        }
        if(T0中存在形如(Nk, R, D)的表项&& D > Dk+1) {
            /* R0到目标网络N的路由的距离发生了变化, R0无条件地接受这一变化3 */
            将T0的表项(Nk, R, D)更新为(Nk, R, Dk+1);
            continue;
        }
        if(T0中存在形如(Nk, Rk, D)的表项&& Dk+1 < D) { /* 这时一定R = Rk */
            /* R0发现了到达目标网络Nk的距离更短的路由*/
            将T0的表项(Nk, Rk, D)更新为(Nk, R, Dk+1);4
            continue;
        }
    }
}
```

³ 请读者注意, T₀中存在形如(N_k, R, D)的表项意味着当前R₀到N_k的最短路由的下一站是R, 而这时正是R在路由报告明确表示该路由的距离已发生了变化, 因此R₀无条件地更新到达N_k的路由距离是当前R到达N_k的最短距离+1, 无论新的距离是比以前更大还是更小。

⁴ 参考概念题 8 中的Hamilton-Jaccobi-Bellman方程(1)。

31	23	15	0
RIP 命令(1-5)	版本号(1)	全	0
地址类型		全	0
IP 地 址			
		全	0
		全	0
		距	离
IP 地 址			
		全	0
		全	0
		距	离
... ..			

地址类型值=2 表示 IP 地址, 距离值=16 表示无穷大。

图 4-7 RIP 路由报告消息

```

/* T 是来自邻接路由器 R 的路由报告;
本路由器是R0, 其当前路由表是T0 */
Vector_Distance_Shortest_Route_Discover_RIP(R, T) {
    for( T的每一项(Nk, Dk) ) {
        if(Dk==16&& T0中存在形如(Nk, R, D)的表项&& D > 16){ /*失效的路由*/
            将T0的表项(Nk, R, D)更新为(Nk, R, 16);
            启动表项(Nk, R, 16)的失效-删除定时器;
            /* 超时后若该失效项仍存在则将其永久删除 */
            continue;
        }
        if(T0中不存在形如(Nk, . . . )的表项) {
            insert(T0, (Nk, R, Dk+1)); /* R0发现了一个新目标网络及其路由*/
            启动表项(Nk, R, Dk+1)的寿命定时器;
            continue;
        }
        if(T0中存在形如(Nk, R, D)的表项) {
            if( D > Dk+1 ) {
                /* R0到目标网络N的路由的距离发生了变化, R0无条件地接受这一变化5*/
                将T0的表项(Nk, R, D)更新为(Nk, R, Dk+1);
            }
            启动表项(Nk, R, Dk+1)的寿命定时器; /*复位-重启*/
            continue;
        }
    }
}

```

⁵ 同前面的注 7。

```

    }
    if( $T_0$ 中存在形如( $N_k, R_k, D$ )的表项&&  $D_k+1 < D$ ) { /* 这时一定有  $R < R_k$  */
        /*  $R_0$ 发现了到达目标网络 $N_k$ 的距离更短的路由*/
        将 $T_0$ 的表项( $N_k, R_k, D$ )更新为( $N_k, R, D_k+1$ );6
        启动表项( $N_k, R, D_k+1$ )的寿命定时器;
        continue;
    }
}
}

```

RIP 的报告周期为 30 秒，路由表项的寿命时限为 180 秒，删除定时器时限为 120 秒。路由表的删除动作在定时器的时间超限例程中执行，这里没有写出。

*4.4.2 路由振荡现象

距离-矢量路由算法如果收敛，则一定收敛到最短距离的路由。但从实用的角度看，不仅要求路由发现算法能够收敛，而且收敛时间越短越好。一个收敛时间过长的算法是没有实用价值的。距离-矢量路由算法的收敛时间一般会有多长？甚至有没有可能不收敛？我们看图 4-8 的例子，通过分析我们将看到即使对这样一个非常简单的网络，如果不进行特殊的处理，距离-矢量算法确实有可能不收敛。这种不收敛或长时间进行路由更新而达不到稳态路由的现象，称为路由振荡。

假定网络开始时处于稳态，路由器 R1、R2 上指示目标网络 N 的可达性的路由表项分别是 (N, G, d)、($N, R1, d+1$)。在某一时刻，R1 到达网络 N 的链路失效，R1 在连续一段时间(对 RIP 是 180 秒)内没有接收到到达 N 的下一站路由器 G(图 4-8 中未画出)发布的路由报告，判定 N 不可达，将失效的路由表项标记为 (N, G, ∞)。根据距离-矢量协议的规定，R1 将在下一个周期点(对 RIP 是 30 秒)发布路由报告时发布该目标网络不可达。然而，如果在下一发布点之前，R2 开始发布其路由报告⁷，R1 将接收到一个表示 N 可达的路由报告项 ($N, R2, d+1$)。

根据前面的算法，R1 将当前标记为失效的路由表项 (N, G, ∞) 更新为 ($N, R2, d+2$)，并且在下一次发布路由报告时包括表示 N 可达的表项 ($N, R2, d+2$)。

R2 接收到 R1 的路由报告后，由于 R2 在其当前的路由表中有表项 ($N, R1, d+1$)，这表明 R2 到达 N 的最短路由上的下一站是 R1，而 R1 路由报告中的项 ($N, R2, d+2$) 表明 R1 到达 N 的距

⁶ 同前面的注 8。

⁷ 这里再次强调，网络中不存在任何机制来同步各路由器的行为，例如，不存在统一的全局时钟提供所谓统一时间。有兴趣的读者或许愿意深入思考如何提供这一类机制，但注意这类机制带来的问题一般要比它们所解决的问题更多。

离已经变化，因此 R1 上的算法无条件地将路由项(N, R1, d+1)更新为(N, R1, d+2)。

至此，读者已看到 R1、R2 上表示 N 可达性的路由项(N, R2, d+2)和(N, R1, d+2)，这实际上表示了一个环路！读者不难想象，这时一个目标地址为 N 的 IP 分组在被 R2(或 R1)转发时将在 R1、R2 之间不停地循环，直到其分组寿命超限而最终被废弃。

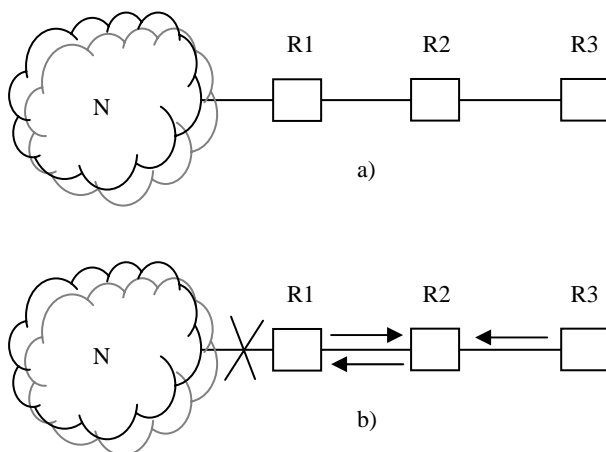


图 4-8 RIP 路由振荡的例子

不仅如此，R1、R2 的路由报告-更新过程还将继续下去，例如下一次 R2 发布的报告将包括项(N, d+2)，这将使 R1 无条件地将其路由项(N, R2, d+2)更新为(N, R2, d+3)，并在下一次发布的报告中包含项(N, d+3)，从而导致 R2 无条件地将其路由项(N, R1, d+2)更新为(N, R1, d+4)，从此反复。

这里要指出一点，路由振荡这类现象是某些分布式算法共同具有的特征，并非距离-矢量动态路由协议所独有，它本质上起源于并发进程在时间上的不正确的异步关系，这在分布式软件中是一个普遍出现而又颇为复杂的问题。

RIP 处理路由振荡的方法很简单，即用一个人工规定的距离上限来强制报告-更新循环终止。RIP 将距离的上限定为 15，距离 16 表示无穷大，因此上面的循环最终会导致 R1、R2 的报告中包含项(N, 16)，从而最终都“意识”到 N 是不可达的。

然而，这样的处理并未真正解决问题，因为路由振荡在使距离最终达到人为的上限之前毕竟还是持续了一段时间，对 RIP，这一时间最长是 16×30 秒 = 8 分钟。从上面的分析读者已经看到，在这期间网络中的全局路由是不正确的，含有一个因振荡造成的环路，这意味着网络的不正确状态最多会持续 8 分钟，这对许多承载高性能、高可靠性业务的网络是不可接受的。另外，人为规定最大路由距离为 16 也限制了 RIP 能被应用的网络的实际大小，使 RIP 不能

直接应用于有距离大于 15 的路由的较大规模的网络⁸。

因此，人为规定距离的上限为 16 只是缓解而非真正抑制路由振荡现象，RIP 还提供另外几种补救措施。一种是所谓水平分裂(split horizon update)，读者从上面的分析中能看到，之所以出现路由振荡，原因在于 R1 不能认识到 R2 所报告的关于网络 N 的路由实际上恰恰来自于自己从前的报告，对 R2 也是一样。如果抑制这种反向报告，即规定如果一个路由器从某个近邻的报告中学习到一组新路由，则永远不再向该近邻发布包含这些路由项的报告，或更准确地说，是不再向该近邻所在的网段上组播包含这些路由项的报告。对图 4-8 中的网络，水平分裂使 R2 不在连接 R1、R2 的链路上发布路由项(N,d+1)，因此 270(=180+90)秒之后 R1 将废弃关于 N 的路由项，而始终不会出现路由振荡和循环。

然而，水平分裂并非适合于所有网络，例如对概念题 9 中的网络，即使 R2 遵循水平分裂规则也仍然存在路由振荡现象。

第二种方法是逆向毒化(poisoned reverse)，要求路由器一旦识别出到达目标网络 N 的路由失效，就立刻发布路由报告，并在报告中明确包含不可达项(N,16)，而不是等到下一次路由报告的周期点才发布，从而使该路由器接收到一个错误的路由可达性报告的机会最小。对上面的例子，逆向毒化要求 R1 立刻发布含(N,16)的路由报告，使 R2 能尽快意识到它所拥有的路由项(N,R1,d+1)已失效，下次将以失效路由的形式通告该项，从而使 R1 接收到 R2 的含有效项(N,d+1)的路由报告、以至于错误地撤消 N 的不可达性的机会大为减小。

然而，逆向毒化也并非完美的解决方案，主要问题是根据 RIP 路由更新算法，及时报告机制会触发连锁的及时报告，结果是虽然逆向毒化及时通告了路由失效，但却在短时间内引起网络负荷剧烈增加，特别是在一个网段上连接了多个路由器的情形。

实际上，路由振荡是距离-矢量算法的内在矛盾，要彻底解决这一问题需要开发新的、基于不同机理的动态路由协议。下一章将要学习的 OSPF 就是一种收敛时间更快、适用网络规模更大的动态路由协议。当然，OSPF 比 RIP 要复杂得多，管理和配置起来也要复杂得多，因此 RIP 对小规模、中等性能要求的网络仍然被广泛应用，包括其增强版 RIPv2。

4.4.3 RIP 的特点及 RIPv2

通过前面两节的讨论，对 RIP 的特点总结、归纳如下：

所能应用的网络直径⁹不超过 15；

⁸ 对这类大网络，如需要应用 RIP，可以将其拆分成较小的网络，每个小网络的最长路由不超过 15，小网络与小网络之间用另一种路由协议联结。这当然增加了组网的复杂性。

依据静态度量计算路由；

每次报告的是完整的路由表，因此路由报告引起的网络负载较大；

收敛速度慢；

缺乏对网络负载的均衡能力。

所有这些特点都决定了 RIP 主要适用于小型网络。RIP 协议目前有两个版本，上面讨论的主要是 RIPv1。RIPv2 的基本原理本质上与 RIPv1 相同，特别是有着与 RIPv1 完全相同的距离-矢量路由更新算法，因此 RIPv2 仍然具有以上那些 RIPv1 的特点。RIPv2 支持子网掩码，因此可以交换无类型 IP 地址的路由表，这是 RIPv1 所不具备的能力，也是两者最主要的差别。

⁹ 一个网络中最长路径的长度，称为该网络的直径。

4.5 因特网自治域与分层路由

在学习 IP 地址时，我们向读者定性地指出过因特网上的路由机制实际上是有层次的，在最粗略的意义上也至少可以划分为两个层次：主干层次路由和各局域网的内部路由，每个层次的路由器都运行相应的动态路由协议。分层路由的最大优点是能适应因特网拓扑的变化，而这种变化正是今天因特网飞速发展所面临的现实。

在因特网上，将一个可以完全自主管理的网络称为一个自治域(AS: Autonomous System)，在实践中，这常常是一个企业所属的网络、或一个大区域内的广域网。一个自治域的本质特点在于，该域内部如何规划网络拓扑、如何规划路由以及运行何种动态路由协议，与其它自治域完全无关，因特网可以看做是各个自治域互联而成的逻辑网络，主干路由所表达的正是自治域之间的连接拓扑关系，因此主干路由协议也称为域间路由协议，运行这种类型协议的路由器称为域的边界路由器或域的边界网关；而上面讨论的 RIP 协议所要发现和表达的网络拓扑仅限于一个自治域内，是典型的域内路由协议。这两者的关系表示在图 4-9 的例子中，其中 G1、G2 之间交换域间路由信息，反映自治域是如何连接的，而 R11、G1、R12 之间交换 AS1 内部的路由信息，G2、R21、R22 之间交换 AS2 内部的路由信息，分别反映 AS1、AS2 内部的网络结构，但(比如说)R11 和 R22 之间从来不会交换任何路由信息。域间动态路由协议和域内动态路由协议在许多方面都是很不相同的，读者在继续学习下面两章时将会理解更多的细节。

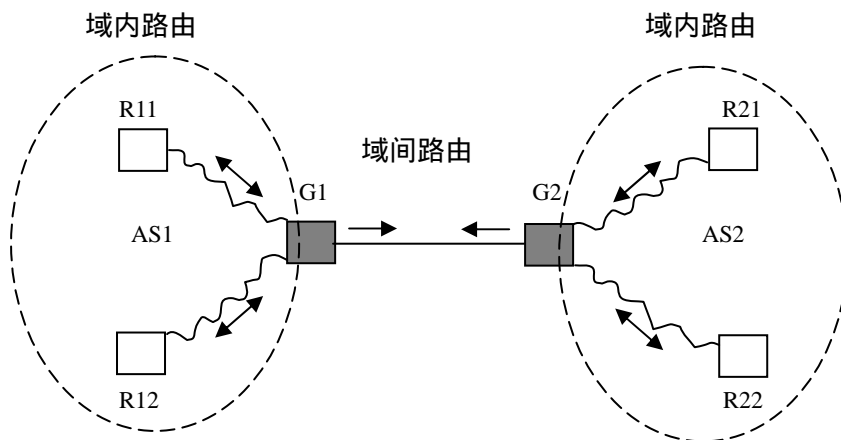
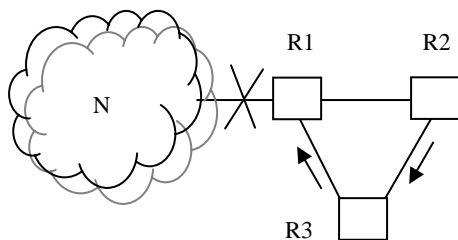


图 4-9 因特网的两层路由结构

习 题

一、概念题

- 1 设想图 4-1 中每个网络上有 1000 台计算机，如果每个路由表项 (D, N) 中的 D 表示目标计算机的 IP 地址而非目标网络的 IP 地址，R3 上的路由表将有多少个表项？有多少表项具有相同的“下一站”IP 地址？冗余信息的比例有多大？
- 2 (1) 写出图 4-1 中的所有路由器 R1- R6 上的路由表，并解释每个表项的涵义。
(2) 根据这六张路由表，而不看图 4-1，你能够恢复出图 4-1 吗？
* (3) 如果能做到，你能写一个算法来一般性地完成 (2) 吗？
- 3 阅读任何一份路由器产品规格说明书，注意该路由器的性能指标、能支持的路由协议有哪些、能支持哪些网络接口(即能连接到那些类型的链路上)。建议你多去几家路由器厂商的网站上查询一番，看看其产品规格的异同之处。
- 4 考虑图 4-2，假设一个目标地址为 202.39.2.65 的 IP 分组从接口 I22 到达 R2，请读者对照路由表 4-2 仔细检验 4.2 节中 IP_Packet_Routing 算法的每一步。如果到达的是一个目标地址为 104.9.12.25(非本局域网地址)的 IP 分组呢？
- *5 用一种较实际的数据结构实现路由表，以实现 4.2 节中 IP_Packet_Routing 算法。
- 6 写出图 4-4 中其它路由器上的路由表。
- 7 在图 4-6 中，如果 R3、R4 在一段时间内直接接收不到对方的路由报告，是否意味着一定是 R3-R4 所在的公共网段或链路发生失效？
- 8 (1)在任何一个图上，如果将节点 R 到节点 N 的最短距离记做 $d(R, N)$ ，证明以下方程成立：
$$d(R_0, N) = 1 + \min_{R \text{ 是 } R_0 \text{ 的一个邻节点}} d(R, N)$$
(2) γ_0 是节点 R_0 到达节点 N 的最短路径，若 γ_0 经过的 R_0 的邻节点是 R^* ，从 R^* 到 N 的 γ_0 的子路径记做 γ^* ，则 γ^* 是 R^* 到 N 的最短路径。
(3) 根据以上结论，证明 4.4.1 小节的算法确实给出本路由器 R_0 到达目标网络 N 的最短路由和最短距离。
- *9 看下图的网络，根据水平分裂的约定，R2 不在与 R1 邻接的接口上发布路由报告，但仍然存在路由振荡，为什么？



*10 RIP 的参数数值是经过一定折衷平衡考虑的结果。试考虑，如果最大距离过小会怎样？如果过大又会怎样？还可以注意到，要想减小振荡持续时间、加快收敛速度，可以缩短报告周期，但报告周期过短会有什么后果？如果过长又怎样呢？

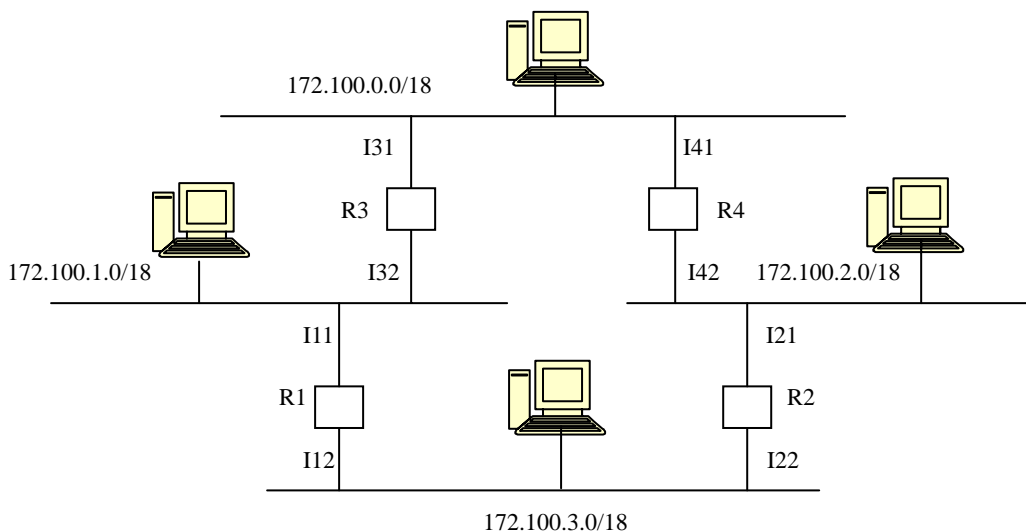
二、应用题

1 路由表并非路由器所专有，主机同样也会有路由表。你的计算机上哪些命令可以查看路由表？用它们来实际看看你机器上的路由表。你能详细解释显示出来的路由表信息吗？请查阅相应的手册。

根据 IP 的规定，主机一般不进行分组转发，主机上的路由表更多是用来发送 IP 分组，例如一个有多个邻接网络的服务器。如果读者对为什么不允许主机进行分组转发的原因感兴趣，可以阅读前一章介绍的 Comer 的著作第一卷 8.10 节。

2 写出图 4-2 中的路由器 R1 和 R3 路由表，假设图中的网络 202.39.2.0/24 通过一个边界路由器 G（图中未画出）连接到主干因特网，而且 G 是所有网络能到达主干因特网的唯一的一个边界路由器。

3 下图中有 4 个网段，在地址空间 172.100.0.0 上以 18 位子网掩码为每个网段分配地址空间，如图所示。（1）为图中每个路由器的网络接口指派合理的 IP 地址；（2）写出路由器 R1 的路由表。



- 4 一个路由器 R 上的路由表如下，试画出该路由器与其邻居路由器的连接关系，并在图上标出通过每个邻居路由器可到达的目标网络。

目标网络地址	掩 码	下一步站 IP 地址	离出接口	距 离
202.204.65.0	255.255.255.0	C	Vlan160	0
202.204.64.0	255.255.255.0	C	Vlan159	0
202.14.71.0	255.255.255.0	202.124.254.9	Vlan2	3
202.38.70.0	255.255.255.0	202.124.254.9	Vlan2	4
202.124.254.0	255.255.255.0	C	Vlan2	0
176.20.0.0	255.255.0.0	202.204.65.1	Vlan160	1

- 5 下面是一个真实网络上的 Cisco 路由器上的路由表显示：

```

C    202.204.65.0/24  Directly Connected, Vlan160
C    202.204.64.0/24  Directly Connected, Vlan159
C    202.204.254.0/24  Directly Connected, Vlan2
O    202.204.71.0/24 [110/2] via 202.204.254.9, Vlan2
O    202.204.70.0/24 [110/2] via 202.204.254.9, Vlan2
OE1  202.112.62.240/30 [110/22] via 202.204.254.250, Vlan2
OE1  202.38.125.0/29 [110/21] via 202.204.254.250, Vlan2
S*   0.0.0.0[1/0] via 202.204.254.250
    
```

“C 202.204.65.0/24 Directly Connected, Vlan160”表示网络 202.204.65.0/24 通过接口 Vlan160 与本路由器邻接，C 表示邻接，“O 202.204.71.0/24 [110/2] via 202.204.254.9, Vlan2”表示到

达目标网络 202.204.71.0/24 的路由的下一站是路由器 202.204.254.9，该路由器在接口 Vlan2 上与本路由器邻接，O 表示该路由项来自 OSPF 路由发现协议，[110/2]是该路由的度量，在本题中暂时忽略。其他项的意义类似，OE1 表示该项来自 OSPF 和 EIGRP 两个协议，最后一项 S*是默认路由。

(1) 请根据这张路由表，画出该路由器与邻接路由器的连接情况，并标记出在每个相邻路由器上可达的目标网络；

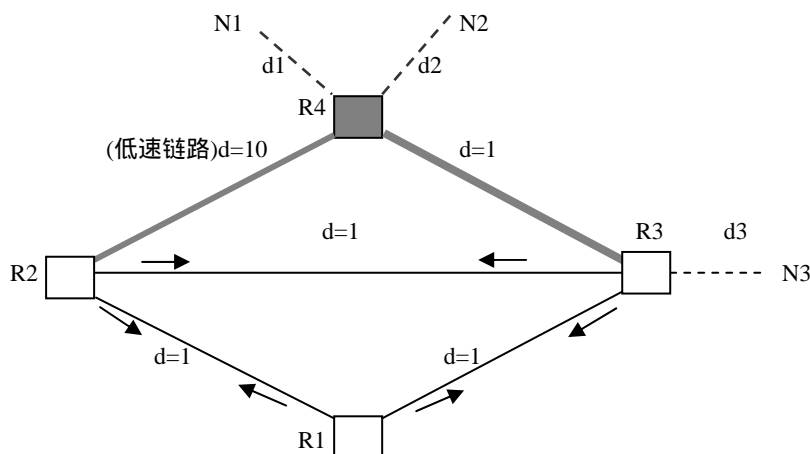
(2) 根据你画出的连接图，写出路由器 202.204.254.250 和路由器 202.204.254.9 上的路由表（按第 4 题中的形式）。

6 下面是另一个真实网络上的 Cisco 路由器上的路由表显示，重做上题：

```

S      202.205.149.0/24[1/0] via 202.112.41.118   Vlanx
S      202.205.148.0/24[1/0] via 202.112.41.118   Vlanx
C      202.112.62.240/30   Directly Connected, GigabitEthernet42
OE1    202.38.125.0/29[110/22] via 202.204.253.18, GigabitEthernet60
C      202.112.41.144/28   Directly Connected, FastEthernet6
OE1    202.112.41.128/28[110/11] via 202.204.253.100, GigabitEthernet60
S      202.112.41.16/28[1/0] via 202.112.7.33, Vlany
C      202.112.7.0/24   Directly Connected, Vlany
C      202.204.253.0/24   Directly Connected, GigabitEthernet60
    
```

7 在下图的网络中，各边的 RIP-度量如所标记的那样取值，假定 R1-R4 上运行 RIP 路由协议。(1)写出各个路由器上的 RIP 路由表；(2)设从某一时刻起 R4 失效，试考虑一种可能发生的 RIP 路由振荡过程。注意 R4 失效意味着两条链路 R2-R4 和 R3-R4 均不再可用。



参考答案

一、概念题

2 (2) 能。请务必做一遍这个练习，你将深切体会出尽管每个路由表仅仅表达“下一站”这一局部信息，但联合起来却可以完整地表达网络的全局拓扑。这是下一章将要学习的动态路由协议的基础，任何一种路由协议实现路由自动发现的基础都在于能够从路由器之间相互交换的局部信息中提取出完整的网络拓扑。

7 未必，可能是路由器自己的网络接口失效、对方路由器失效，但无论哪种情形，都意味着经过对方的路由不再可用。

8 对边赋权(有向或无向)图,即每个边 (u,v) 指派有一个正数 $w(u,v)$ ¹⁰作为其长度的图,若定义一条路径的长度等于其中各边长度之和,则(1)更普遍的形式是

$$d(R_0, N) = \min_{R \text{ 是 } R_0 \text{ 的一个邻节点}} [w(R_0, R) + d(R, N)].$$

(1)和(2)实际上就是求解最短路径问题的著名的 Dijkstra 算法的依据,4.4.1 小节的算法实际上是公式(1)的分布式实现。

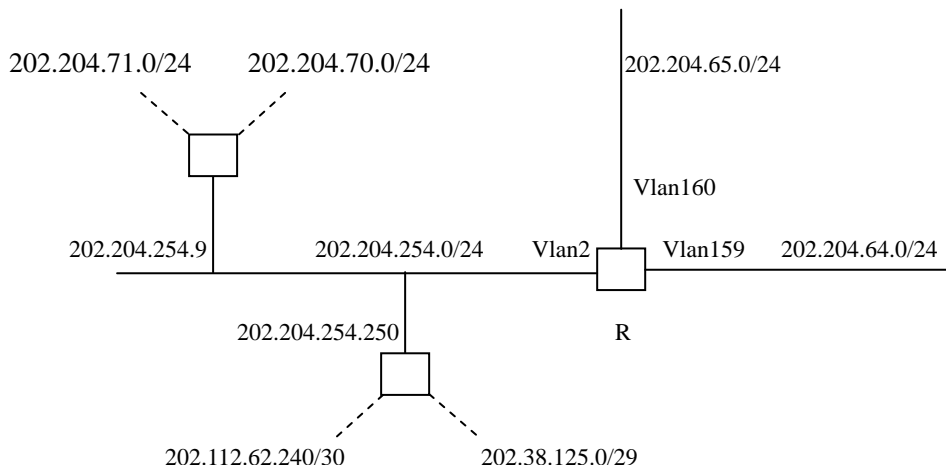
10 RIP 最大距离不宜很大,报告周期也不宜过长,这样会使收敛时间很长。但两者也不宜太小,前者过小会限制网络的大小,后者过小则导致 RIP 报告频繁、增大网络拥塞、降低传输效率。

二、应用题

1 netstat -r 是可用的命令之一。

5 该路由器 R 的邻接图如下：

¹⁰ 这里正数的要求是实质性的,否则公式不成立。



7 路由表的主要表项如下：

R1 路由表=(N1, R3, d1+2) (N2, R3, d2+2) (N3, R3, d3+1)

R2 路由表=(N1, R3, d1+2) (N2, R3, d2+2) (N3, R3, d3+1)

R3 路由表=(N1, R4, d1+2) (N2, R4, d2+2) (N3, G, d3+1) G 在图中未画出

一种可能的振荡时序：

R3 检测出 R3-R4 链路失效；

R2 收到 R3 的 RIP 报告，报告 N1、N2 不可达，由此 R2 将 N1、N2 暂时标记为不可达；

R1 恰在其收到 R3 的报告之前发布 RIP 报告，因此报告中含 N1、N2 经 R3 可达，距离有限，次分别为 d1+2、d2+2；

R2 收到 R1 的 RIP 报告，因此 R2 置 N1、N2 经 R1 可达，距离分别为 d1+3、d2+3；

R2 发布其 RIP 报告；

R3 收到 R2 的 RIP 报告，因其中报告 N1、N2 可达，R3 置 N1、N2 经 R2 可达，距离分别为 d1+4、d2+4；

R3 向 R2、R1 发布 RIP 报告，因在此之前 R1 已收到来自 R3 的关于 N1、N2 的不可达报告，现在 R1 将置 N1、N2 经 R3 可达，距离分别为 d1+5、d2+5；

至此，R1、R2、R3 关于 N1、N2 的路由信息形成 R1-R2-R3 之间的一个环路。

R2 再次收到 R1 发布的 RIP 报告时，要对 N1、N2 (R2 的下一跳正是 R1) 的距离依据 R1 报告的当前距离如实更新，故 R2 置 N1、N2 的目标距离为 d1+5、d2+5；

如此，每个路由器如实更新到达 N1、N2 的距离，最终超过最大值 15，表示 N1、N2 不可达。至此，振荡过程停止。振荡持续时间=RIP 报告周期性×RIP 最大距离=30 秒×15=450 秒

= 8 分钟！在此期间 N1、N2 事实上不可达，IP 分组的路由是环路。