计算机科学1998 Vol. 25 №. 4

单向非对称链路路由的研究与进展*)

Researchs and Advances in Routing with Unidirectional and Asymmetrical links

潘建平 顾冠群
(东南大学计算机系 南京 210096)

摘 要 新型单向或非对称链路为传统分组网络的互连和接入提供了更大的灵活性,但也直接影响了传统路由协议原有某些算法假设和协议机制,几乎现有的路由协议都无法直接运行在单向链路之上。本文描述单向路由短期研究的一些进展与成果以及我们在解决远期方案中邻接发现和路由生成所做的工作。

关键词 单向链路,非对称链路,路由协议

传统分组网络的路由协议总是假设传输媒体具有双向和对称的特征,无论是基于距离向量还是链路状态的路由协议,都认为从A到B的路径蕴含着从B到A的反向路径。但是随着Internet的不断发展,新型网络应用和传输服务的出现,原有假设的合理性和可行性就明显出现问题。首先,许多网络应用本身就具有非对称性。如文件下载和Web浏览等主流应用,从客户端到服务器仅存在少量的命令或反馈,而大量的数据则是从服务器下载到客户端。随着网络应用的进一步发展,这种非对称性将可能更加显著。需要指出的是,上述是指数据流的非对称性,因为单向的数据流也会伴随反向的协议控制,也就是从协议研究的角度而言,数据流和控制流仍然必须构成一个闭合回路。其次,许多新型的传输服务本身就是单向或非对称的。卫星通信的广播投递形式能够容易地构造新型应用所需的多点投递服务,并且卫星的覆盖面广、带宽大,能灵活地适应移动用户的高吞吐通信需求。但卫星通信的发射设备要远比接收设备庞大、复杂和昂贵,对小型和个体移动用户而言,仅携带接收设备更为合适。因此对于许多非对称应用可利用卫星信道提供大量的数据传输,使用其它通信设施(如地面带宽较小的公众电话网)传输少量的控制信息。这能够极大地缓解现有地面通信设施过度拥挤的局面。此外根据网络应用的发展趋势和调制编码的技术,许多新出现的用户环路传输技术,如Cabel Modem和xDSL,大多提供非对称的传输能力,一般下行带宽要远大于上行带宽。还有一些特殊的传输介质或是媒体访问控制的原因或受现有布线系统的影响,只能提供单向的传输能力。

尽管非对称的极限就是单向,但两者从协议研究的角度来讲是有区别的。单向是从图论的观点描述结点之间的有向边,传统的双向对称链路可分解成两个对称反向的单向链路;是否对称是依据链路本身的带宽分配而言的,两个容量等同反向伴随的链路可合并成一个双向对称链路。对于路由协议可以使用反向距离向量无穷大表示链路的单向,链路状态则关心是否容量对称。基于单向链路的路由协议更加困难,因为缺乏显式的反向通道。

1 对传统协议的影响

单向和非对称链路的出现,给传统网络层和运

*)国家自然科学基金,国家863高科技计划和国家教委博士点基金资助项目。潘建平 博士,研究方向为高速、高性能网络及协议。顾冠群 教授,博士导师,研究领域涉及计算机网络,分布式处理和CIMS等。

[5]Rishiyur S. Nikhil, Can dataflow subsume von Nueman Computing?, Proc. 16th Ann. Int'l Symp. Computer Architecture, 1989

[6]Gregory M. Papadopoulos and Kenneth R. Traub, Multithreading: A Revisionist View of Dataflow Architecture, ISCA, 1991. pp. 342-351

[7]Rishiyur S. Nikhil et al., * T: A Multithreaded Massively Parallel Architecture, Same to[4]

[8]Dean Michael Tullsen, Simultaneous Multithreading, Ph. D Dissertation, Univ. of Washington, 1996

[9]Jack L. Lo, etc., Coverting Thread-Level Parallelism to Instruction-Level Parallelism via Simultaneous Multithreading

[10]Bernarl Karl Gunther, Superscalar Peformance in a Multithreaded Microprocessor, Ph. D Dissertation, Univ. of California, 1993

输层的协议都带来了新的问题，涉及到对路由协议、多点投递、资源管理、及对运输层和高层应用的影响等诸多方面。

1.1 对路由协议的影响

传统的路由协议有自治域(AS)间和域内两类，AS是域内路由协议的管理范围。主流的域间路由有基于路径向量的BGPv4协议；域内路由可分为基于距离向量的RIP或基于链路状态的OSPF。RIP是Internet应用最为广泛的动态路由协议，大多运用在接入网络的边缘。RIP路由非常简单，每个结点独立地计算到达目的网络的接口和距离，并把上述信息扩散给其相邻结点。接收到相邻结点A路由信息的结点1这么认为：如果A到D的距离为X，则1到D的距离为X+1。如果1原来不能到达D、或1到达D的距离大于X+1，1将更新其路由表，分别将A和X+1作为到达D的下一跳(next hop)和距离。RIP相邻结点间周期性地在邻接链路相互交换上述路由信息，以适应网络的动态行为，如某些链路失败或新的链路。RIP在计算距离和交换路由的过程使用了双向链路的假设，这在单向链路环境是不适用的。1不再能从A到D的距离推算1到D的距离，也无法在邻接链路交换路由信息，即使能够通过其它路由回到原有结点，由于不是来自发出路由的接口，路由算法也会将其视为非法信息抛弃。这样A就无法获悉并利用A到1之间存在的单向链路。

基于链路状态的OSPF协议将其接口信息扩散到整个网络(AS域)，而不是RIP仅通知相邻结点，这样每个运行OSPF路由的结点最终将能获悉网络的全部拓扑，独立地进行路由计算和选择。除了RIP的距离向量(跳数)外，OSPF还会引入更多的链路度量，如链路带宽和延时等，供路由选择使用。从理论上讲，OSPF由于具有整个网络的完整拓扑，也能获悉单向链路的存在，只需在路由计算时标识为单向即可。但是OSPF的许多辅助功能，如邻接发现、全局一致和可靠扩散等，都需要交换信息的相邻结点进行应答确认，通过重发保证信息扩散的可靠性。大多数OSPF的实现都默认邻接链路具有双向的数据发送和应答能力，而对于单向链路来说这样的反向应答通道事实上根本上不存在。

对于覆盖广泛的卫星信道而言，单向链路甚至可能跨越不同的自治域，这对域间路由协议也会造成影响。BGPv4这样的域间路由协议本身就运行在可靠的TCP协议之上，需要下层能够提供双向的数据发送和应答能力，这显然是纯粹的单向链路无法胜任的。如果将单向链路涉及的范围作为一个单独的自治域，可将域间路由在单向链路遇到的问题转化为上述讨论的域内路由问题。因此解决单向链路路由问题的关键仍然是针对域内路由的。

1.2 对多点投递的影响

传统的路由协议仅支持单点投递路由，随着网络应用的不断发展，许多新型应用需要下层提供多点投递和群组管理的能力。IP多点投递扩展就是在传统单点投递IP协议的基础上扩充形成的，提供单点到多点的数据投递服务。当然需要具有多点投递能力路由协议的支持，DVMRP和MOSPF就是对原有基于距离向量和链路状态路由协议的多点投递扩充，前者广泛地使用在Internet的多点投递虚拟骨干网MBone。支持多点投递最简单的方法是分组扩散，即在输入接口以外的其它所有接口输出多点投递分组。但是这样无条件的扩散会造成网络和协议的严重负担和不必要的浪费。因此DVMRP采用"反向路径最短转发"的约束条件，当且仅当多点投递从离多点投递源结点最近距离的接口输入时才被转发，以避免过度的分组转发风暴。但是这就意味着任何结点需要获悉到达源点的反向距离，单向链路甚至不可能获得这样的反向距离，除非源点显式地通知该结点两者的正向距离。

接收方通过IGMP成员资格协议加入多点投递群组。为了进一步避免多点投递的过分扩散，若边缘路由器没有发现所在子网有该多点投递群组的成员，就可以不再进行转发，并且通知前一个多点投递路由器，以"反向裁剪"原来的广播扩散树。这也需要下层提供双向的传输能力。对于单向链路网络，就要通过间接的方法通知前一个路由器进行裁剪。

1.3 对资源管理的影响

传统无状态的分组网络仅提供尽力而为的分组转发，而许多新型应用要求网络能提供一定程度的服务质量保证，如传输延时和网络带宽等。对于需要提供完全服务保证的通信需求，资源预留(包括网络带宽、分组缓存和计算能力等)是不可避免的。RSVP是一个基于接收方的资源预留协议。发送方首先发出PATH报文，经过下层路由协议多点投递到每个可能的接收方；接收方根据各自对于服务质量的需求，给出资源预留的RESV报文，通过PATH途径的路径反向到达发送方，每个经过的路由器都将进行接纳控制和资源预留，如果失败则通知接收方，否则将资源预留请求递交给前一个路由器。为使接收方能够在路由器可接受的范围给出预留请求，PATH将携带发送方支持的通信需求，每个转发PATH分组的路由器将可能修改PATH分组有关可用资源的描述以通告接收方。无论上述何一过程都认为RESV报文反向途径PATH经过的路径

(RESV 如何返回发送方及路由器的资源通告)。对于单向链路 PATH 和 RESV 就可能不再通过相同的路径,资源预留就无法如此实现。

资源管理是集成服务分组网络的重要内容,除传统网络提供的尽力而为服务外,集成服务还定义了可保证和可控制的服务。可保证的服务通过接纳控制、资源预留和分组调度确保传输吞吐和延时。可控制的服务在网络负载较轻的场合等同于传统的尽力而为服务,但在网络负载加重进入拥挤状态时,通过接纳控制和分组调度使得原有的传输吞吐和延时基本不变,而不象尽力而为服务那样造成有效吞吐的下降和传输延时的上升。可保证服务适用于具有严格吞吐和延时需求的应用,不会由于队列溢出产生分组丢失,但会降低网络可能的利用效率。可控制服务虽然没有保证吞吐下界和延时上界,但是通过协议控制能够维持吞吐和延时在轻负载网络的水平,由于队列溢出造成的分组丢失率也很低,同时最大程度地利用了网络的复用能力,提高了网络的可用率。但是集成服务同样也默认下层链路具有双向传输的能力。如在反馈控制时认为应答反馈和数据传输穿越相同的网络区域。

1.4 对运输协议的影响

从理论上讲,链路的变更不会严重地影响运输层的协议,这是层次型协议体系的优点。但是单向和非对称链路的存在仍然影响运输层协议原有的协议假设和协议机制的合理性。如卫星链路虽然带宽可以很大,但是受光速传播和同步通信卫星到地球距离的限制,传输延时较大。两者的乘积,通常称为数据管道,将会变得更大。这对于依赖反馈控制的协议来说,反馈控制的效率就需要重新考虑,甚至在控制信息返回动作方的同时,由于更多的数据注入管道,原有的控制信息就可能不再有效。TCP 协议的 Slow Start 和拥挤避免算法也会受到大数据管道的影响。TCP 在连接初始时候设置拥挤窗口为1个数据段,随着应答的返回逐步指数性地打开拥挤窗口到正常水平(在超过门限值后变为线性增长)。对于大带宽延时乘积的网络,因为传输延时非常大,通过应答打开拥挤窗口的过程会变得非常缓慢。对于 Web 浏览这样的应用,HTTP/1.0要求对每个页面对象使用单独的 TCP 连接,往往使连接的代价(如 TCP 的3次握手)甚至超过数据传输。TCP 依赖定时器的超时判断分组传输或接收的差错,出于同样的原因,差错恢复过程也将变得相对缓慢。

1.5 对高层应用的影响

某些应用对大带宽延时的网络并不敏感,如非交互的文件下载和电子邮件等批量数据传输,还会得益于大带宽带来的吞吐提升。但是对于交互型或事务处理类的应用,庞大的数据管道将会显著地影响应用的可见性能。如前所述 Web 浏览,除了 TCP 的3次握手建立连接以外,嵌入对象的数据应答甚至不足以能够完全打开数据管道。对于网络互连的场合,单向链路的引入使得 TCP 连接跨越更多带宽和延时相异更大的链路,拥挤控制更为重要。

2 近期的解决方案

针对单向和非对称链路产生的新问题,特别是直接对网络层路由协议的影响,Internet 技术领导者 IETF 专门设置单向链路路由工作组(UDLR)来解决上述问题,UDLR 主要集中近期方案的研究,就是在尽量不改变现有网络层协议和路由,以容纳单向和非对称链路,现有的解决方案包括通过封装的方法屏蔽单向链路,或适当地修改特殊位置的路由协议。

2.1 待解决的拓扑

近期需要解决的网络拓扑是在现有的完全双向链路之上附加一些单向链路。这些链路通常提供较大的传输带宽,以解决现有 Internet 长途链路的缺乏和某些区域的过度拥挤等。当然静态路由配置就能解决上述问题,但是静态配置没有可扩展性和可适应性。隧道封装和协议修改是解决动态路由的典型近期方案。

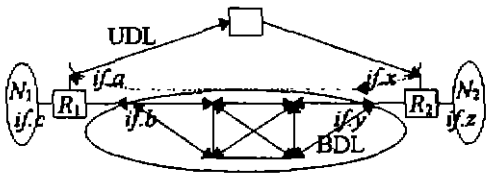

图1 近期解决的网络结构

2.2 隧道的方法

最大程度避免修改现有路由协议的方案就是使用封装隧道的方法给单向链路"构造"伴随的反向回路,提供相应的虚拟链路特性。路由协议可以不加修改地运行在普通的链路和经过隧道封装的单向链路。封装的含义就是在虚拟链路数据由网络层协议而不是链路层传输,通过在单向链路的两端网络接口和路由协议之间嵌入封装功能,单向链路就可模拟双向链路。通常从单向链路接收方到发送方的路由信息经过封装后通过虚拟链路返回单向链路发送方;经过封装解除后单向链路发送方将路由信息返回给路由软件,路由协议认为上述路由信息似乎来自单向链路的网络接口,因此可以进行路由计算和选择。通常大带宽的单向链路因覆盖面广而距离向量较小,或有较优的链路状态,一旦路由被发现,数据将通过单向链路传输,通过伴随的虚拟链路返回

应答控制等。

图1的 R_1 和 R_2 之间就存在单向链路 UDL，虚线是伴随的反向回路。R_1 通过 UDL 将其 if.a 和 if.b 的 IP 地址告诉 R_2（及新加入的结点）；R_2 无法直接通过 UDL 将其网络接口告知 R_1，但根据 if.b 的 IP 地址和借助双向链路 BDL 存在的路由可将其 if.y 的 IP 地址通知 R_1。这样 R_1 和 R_2 就可构造 UDL 伴随的反向回路。WIDE 的隧道封装方案采用一个真实和一个虚拟双向链路，前者完全由 BDL 构成，用于交换 UDL 两端的 IP 地址，后者由 BDL 和 UDL 构成，用于构造封装隧道供交换路由信息使用，也就是仅有路由信息需要封装。Hughes 的封装方案仅采用一个虚拟 BDL，IP 地址和路由信息的交换都需封装。

2.3 路由协议的修改

尽管隧道封装的方法比较简单，但是隧道本身的配置是需要额外支持的，通常只适用在规模较小和较为稳定的网络。传统路由协议的修改即最大程度地保持原有协议，又尽量适应单向链路，是较为合适的选择。对协议的修改主要是去除对双向链路的假设。对距离向量的路由协议，仅在单向链路两端需要修改；而对链路状态的协议，由于每个路由结点独立地计算和选择路由，因为所有的结点都需要修改。作为近期解决方案，协议修改主要针对基于距离向量的 RIP 和 DVMRP 路由协议。

图2的 R_1 通过其接口 if.c 获悉网络 N_1 是直接可达的，R_1 使用 RIP 协议通过 UDL 告知 R_2 有关 R_1 的可达信息。修改过的 R_2 并不直接通过 UDL 告知其 if.z 接口到 N_2 的可达信息，也不将 R_1 的可达信息立即扩散出去。等到 R_2 通过 BDL 告知 R_1 其可达性信息，N_1 通过 R_1 和 UDL 及 R_2 到达 N_2 的路由就产生了，即 R_1 获悉到从 N_1 到 N_2 的最短距离为2。这样数据将从 N_1 通过 R_1-UDL-R_2 到达 N_2。对于从 N_2 到 N_1 的数据仍然需要通过传统的 BDL。

对于多点投递 DVMRP，也需要进行类似的修改。当单向链路的接收方获得从 UDL 来的多点投递路由，并不能直接获得自身到多点投递源点的距离（传统的反向度量不再适用）。对于 UDL 链路，接收方不再计算反向距离，而是由 UDL 的发送方给出的正向度量。UDL 接收方根据正向度量决定是否需要重发多点投递分组。对于 IGMP 反向裁剪，UDL 接收方需要和 RIP 路由扩散一样通过 BDL 告知 UDL 的发送方。

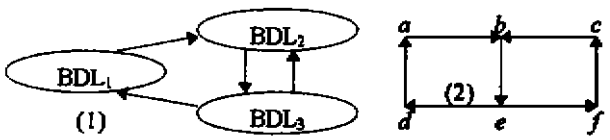

图2 远期解决的网络结构

3 远期的解决目标

尽管隧道封装和协议修改等方法能在近期解决单向非对称链路给网络层路由协议带来的问题，但是上述两种方法都需要其它协议机制的支持，路由协议不是独立的，只能够适合非常简单的网络拓扑，即在传统连通的双向链路之上额外增加几条单向链路。

3.1 待解决的拓扑

图2给出几种较为可能需要远期解决的网络拓扑，UDL 连接的 BDL 孤岛和完全 UDL 连通网络。前者是一个层次的网络结构，每个 BDL 网络又可能是由 BDL 或 UDL 或两者的混合构造而成；后者则是完全极端的情况，即将 BDL 看作一对 UDL。无论上述何种情况，隧道封装和协议修改都无法适用，因为某些结点之间没有 UDL 伴随的 BDL 作为伴随回路。如图2(2)，结点 a 到 f 之间仅又一条 UDL 构成的回路，并且需要两次通过链路〈b,e〉，这种情况在传统路由的场合是不可能出现的。

3.2 网络和路由的表示

图2(2)所示的 UDL 连通网络可表示成图 $G=(V,E)$，其中结点集合 $V=\{a,b,c,d,e,f\}$，边的集合 $E=\{\langle a,b\rangle,\langle c,b\rangle,\langle d,a\rangle,\langle b,e\rangle,\langle f,c\rangle,\langle e,d\rangle,\langle e,f\rangle\}$，$\langle u,v\rangle$ 表示从结点 u 到 v 的有向边。此外 $C:E\to R^m$ 是从 E 到 m 维实数域 R 的映射，表示每个链路所对应的度量（有 m 个因素）。u_1 和 u_n 之间路径 $\{\langle u_1,u_2\rangle,\langle u_2,u_3\rangle,\cdots,\langle u_{n-1},u_n\rangle\}$ 的集合记作 $R_{u_1\to u_n}$，$\|R_{u_1\to u_n}\|$ 是所有可能路径的个数。

路由问题可表示为给定图 $G=(V,E)$ 及其映射 C，任取 $v_1,v_2\in V$，获得 $R_{v_1\to v_2}$ 和 $R_{v_2\to v_1}$，依据 C 的度量约束，挑选某个（或某些）特定的 $r_{v_1\to v_2}$ 和 $r_{v_2\to v_1}$ 构成回路。上述描述蕴含每个结点都有图的完整拓扑，若无此约束，仅能得到 $R_{v_1\to v_2}$ 和 $R_{v_2\to v_1}$ 的子集。此外回路 $v_1\to v_2\to v_1$ 和 $v_2\to v_1\to v_2$ 并没有对称的必要，两者在 $R_{v_1\to v_2}$ 和 $R_{v_2\to v_1}$ 集合进行选择的因素并不尽相同。

对多点投递源 f 和宿集合 $S=(s_1,s_2,\cdots)$，$R_{f\to S}$ 定义为超集 $\{R_{f\to s1},R_{f\to s2},\cdots\}$，$R_{S\to f}$ 则为 $\{R_{s1\to f},R_{s2\to f},\cdots\}$。依据 C 的度量约束，多点路由挑选一组特定的 $\{\{r_{f\to s1},r_{s1\to f}\},\{r_{f\to s2},r_{s2\to f}\},\cdots\}$ 构成 f 与任一 s_i 的回路。多

点投递仅定义 $f \mapsto s_i \mapsto f$ 的回路，因为仅有从源到宿的数据流。在大多数情况 $R_{f \leftrightarrow S}$ 构成一棵以 f 为根，s_i 为叶或中间结点的有向树。

在完全 UDL 网络会出现一条回路多次穿越一条链路的情况，这给路由算法和协议带来很多困难，如环的避免和缩小扩散等。庆幸的是，我们能够证明任何两个结点之间的回路穿越特定链路的次数没有必要超过2。如果将回路 $v_1 \mapsto v_2 \mapsto v_1$ 经过所有的结点接连成字符串 $A = v_1 \ldots v_2 \ldots v_1$，则穿越任一链路〈x，y〉必将导致在 A 出现相继的 x 和 y。首先可确信 v_1 只需在 A 出现1次（如果出现2次或2次以上，则仅有最后1次是有效的）。其次假设某条链路〈a，b〉被穿越3次。若 $v_2 \in \{a, b\}$，则与前提产生矛盾；否则3个相继的 ab 必将 A 分割成4个部分，v_2 仅可能出现在其中之一，根据抽屉原则，无论 v_2 位于哪一部分，均在 v_2 一侧出现2个 ab 子串，则2个子串仅后一个是有效的，也产生矛盾。所以任一链路没有必要出现3次。

3.3 路由协议过程

邻接发现对 BDL 是显而易见的（〈a，b〉⇒〈b，a〉），但对 UDL 却是最大的困难（蕴含条件不再成立）。如图2(2)的 a 发现存在2个网络接口，其中一个输出另一个输入。初始时 a 仅具有自己的接口信息，并将上述信息通过输出接口告知下一结点（注意 a 并不知 b 是下一个结点），b 获得 a 的接口信息后（即导出〈a，b〉），但〈a，b〉本身对 b 的路由没有作用，b 将〈a，b〉及其接口信息一起再传递给下一结点。类似地 e 在其所有输出接口发出〈a，b〉〈b，e〉及其接口信息。d/f/c 有着相似的过程。a 从 d 获得〈a，b〉〈b，e〉〈e，d〉及〈d，a〉后可导出 $a \mapsto b$，$a \mapsto e$，$a \mapsto d$ 路径。因为上述信息和 a 初始发出的信息（蕴含〈c，a〉）不同，a 将继续将链路和路径信息发给下一结点。b 从 c 获得〈a，b〉〈b，e〉〈e，f〉〈f，c〉及〈c，b〉后可导出 $b \mapsto e$，$b \mapsto f$，$b \mapsto c$，与 a 同样原因，b 将继续发出链路和路径信息。b 再次从 a 获得路由信息将获得路径 $b \mapsto a$，结合 a 给出的 $a \mapsto b$，b 将获悉 a 和 b 之间已形成回路（此后也被 a 获悉）。连接发现过程稳定以后，每个结点将具有整个路由范围的拓扑信息，每个结点将独立地进行路由计算与选择。

产生路由信息以后，结点之间需要维持上述路由，并适应链路的动态特性（如新的链路加入和原有链路失败等）。每个结点周期性给其下游结点 Direct-Hello 分组以通知下游结点该结点及邻接链路的活性。如果下游结点在一段时间没有获得来自上游结点的 Direct-Hello，就认为该结点或邻接链路失败。下游结点将扩散链路失败的消息给其下游结点，直到所有结点均获得此信息。下游结点也通过周期性的 Routed-Hello 分组经过路由通知对应上游结点两者链路的活性，如果上游结点在很长一段时间没有获得来自下游结点的 Routed-Hello，将认为两者之间的链路失败，需要重新计算和选择路由，并在其它输出接口通知其它结点。如图2(2)的〈a，b〉链路失败，b 没有收到 Direct-Hello 认为 a 或〈a，b〉失败，将通知后继结点 e，类似地 f/d/c/a 也将获得获悉上述信息。经过 Routed-Hello，a 最终将获悉〈a，b〉链路的失败。除了 b/c/e/f 结点可能形成回路，其它结点至多存在某个路径而无法形成回路。新加入结点将触发局部的邻接发现，若某个结点得到新的路由，将传递给后继直到拓扑信息稳定。

结束语 单向和非对称路由无论从理论或实际都是非常困难的问题，特别是单向链路将使许多原来简单问题变得非常复杂，如拓扑邻接的发现等。此外新的传输服务和应用需求的出现，也给传统的路由协议提出新的要求，如多路径路由、多点投递路由和基于策略的源路由等。近期解决方案如隧道封装和协议修改只能适应在传统连通双向网络之上额外添加单向链路和简单路由需求的场合，无法支持更加复杂的网络拓扑和路由需求。

参考文献

[1]C. Malkin, Routing Information Protocol-RIP, Request for Comments 1058, Jun 1988

[2]J. Moy, Open Shortest Path First-OSPF Version 2, Request for Comments 1583, March 1994

[3]Y. Pekhter, A Border Gateway Protocol 4-(BGP-4), Request for Comments 1771, March 1995

[4]S. Deering, et al., Distance vector multicast routing protocol-DVMRP, RFC 1075, Nov. 1988

[5]Thierry Ernst, Dynamic routing in networks with unidirectional links, Thesis, University of Nice-Sophia-Antipolis, France, Feb. 1997

[6]Emmanuel Duros et al., Supporting unidirectional links in the Internet, In Proc. of 1st Int'l Workshop on Satellite-based Services, Rye, New-York, Nov. 1996

[7]Walid Dabbous et al., Dynamic routing in networks with unidirectional links, In Prof. of 2nd Int'l Workshop on Satellite-based Services, Budapest, Hungary, Oct. 1997

[8]顾冠群、潘建平等，单向链路网络路由技术研究[技术报告]，SEU-NRG/CSD-HPN 1997.7

[9]潘建平、顾冠群，单向链路邻接发现和路径生成的研究[技术报告]，SEU-NRG/CSD-HPN 1997.10