

QoS 路由中一种有效的多受限优化路径选择算法^{*})

An Efficient Algorithm for Multi-Constrained Optimal Path Selection in QoS Routing

王建新 王新辉 彭革刚

(中南大学计算机科学与技术系 长沙 410083)

Abstract An important issue for providing better guarantees of Quality of Service (QoS) to applications is QoS routing. The task of QoS routing is to determine a feasible path that satisfies a set of constraints while maintaining high utilization of network resources. For the purpose of achieving the latter objective additional optimality requirements need to be imposed. In general, multi-constrained path selection problem is NP-hard so it cannot be exactly solved in polynomial time. Accordingly heuristics and approximation algorithms with polynomial or pseudo-polynomial time complexity are often used to deal with this problem. However, many of these algorithms suffer from either excessive computational complexity that cannot be used for online network operation or low performance. Moreover, they generally deal with special cases of the problem (e.g., two constraints without optimization, one constraint with optimization, etc.). In this paper, the authors propose a new efficient algorithm (EAMCOP) for the problem. Making use of efficient pruning policy, the algorithm reduces greatly the size of search space and improves the computational performance. Although the proposed algorithm has exponential time complexity in the worst case, it can get very good performance in real networks. The reason is that when the scale of network increases, EAMCOP controls efficiently the size of search space by constraint conditions and prior queue that improves computational efficiency. The results of simulation show that the algorithm has good performance and can solve effectively multi-constrained optimal path (MCOP) problem.

Keywords QoS routing, Multiple constraints, Optimize, Path selection

1 引言

目前,人们正期待着宽带集成服务网来支持各种各样的满足不同 QoS 要求的多媒体应用。在宽带体系结构的设计中的一个关键问题就是如何提供资源来满足每次连接的需求。毫无疑问,有效的 QoS 路由方案的建立是这种体系结构中的一个重要组成部分。的确,QoS 路由已经成为许多研究的主题。人们已经认识到一个有效的 QoS 路由方案的建立给我们带来了许多挑战。在算法上,QoS 路由带来的一个挑战就是需要迅速地找到一条可行路径使它满足一组限制条件同时获得较高的网络资源利用率^[1~3]。一般而言,QoS 路由是一个复杂的问题。首先,网络电话和分布式游戏等分布式应用在延迟、延迟抖动、丢失率和带宽等方面有许多不同的 QoS 限制。多个限制经常使得路由问题更加复杂。例如,寻找一条具有两个独立路径限制的可行路径是 NP 难的。其次,将来的集成服务网很可能既要传输 QoS 数据流又要传输尽力而为的数据,这就使得性能优化问题更加复杂。第三,由于瞬间的负载波动使得网络状态是动态变化的。不断增长的网络大小使得在动态环境中收集最新的状态信息日益困难。如果使用过时的状态信息,QoS 路由算法的性能将会严重降低。为了进行 QoS 路由,路由器需要得到可用网络资源的信息。这种获得可靠的网络资源信息的机制能够成为路径建立协议的一部分(例如:PNNI),或者象 MPLS 一样通过现有的路由协议来提供这种机制。MPLS 依赖 OSPF(开放最短路径优先)^[4]来提供链路

状态信息(例如:延迟、带宽等)。这就要求按文[8]对 OSPF 进行一些扩展。一般而言,网络路由由两个基本的任务组成^[5]:(1)收集状态信息同时保持对信息的不断更新;(2)根据收集的信息,寻找一条可行路径。在本文中作者关注于第二个任务同时认为对于每一个节点而言,网络信息是可靠的,节点利用这些信息确定端到端的路径。

在网络中的每一条链路与多个 QoS 参数相关,这些 QoS 参数可大致分为可加性和非可加性的^[9,10]。对于可加性参数(例如:延迟、跳数、抖动),路径的代价为这条路径上所有链路参数值的和。相反对于非可加性条件(例如:带宽),路径的代价由瓶颈链路上的参数值所决定。众所周知,通过首先剪去不满足这些限制的链路的一个预计步骤^[11],我们能够比较容易地处理非可加性参数。因此,本文作者主要研究基于可加性 QoS 参数的多受限优化路径选择算法,同时认为路径选择的优化性也由一个可加性参数所决定。

我们用一个无向图 $G = \langle V, E \rangle$ 来表示一个具有 k 个限制条件的网络,其中 $V = \{v_1, v_2, \dots, v_n\}$ 表示节点的集合, $E = \{e_1, e_2, \dots, e_m\}$ 表示链路的集合。在 E 中的每条链路 $[v, w]$ 上都有一个由 k 个参数 $c_i(v, w), i = 1, 2, \dots, k$ 所组成的有序列表。对应于 k 个链路参数,存在 k 个路径评价函数 $F_i, i = 1, 2, \dots, k$,使得对于网络 G 中的每条路径 $P = \{w_1, w_2, \dots, w_k\}, F_i(P) = F_i(c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k))$ 是路径 P 上第 i 个链路参数的函数。一个 QoS 请求 $R = (s, t, B_1, B_2, \dots, B_{k-1})$ 由一个源节点 s , 一个目的节点 t 以及 $k-1$ 个参数限制

^{*} 国家海外杰出青年自然科学基金(69928201)、高等学校骨干教师资助计划资助。王建新 副教授,博士,专业方向:网络中的 QoS 问题、路由算法、网络性能评价。王新辉 硕士研究生,专业方向:网络路由算法的优化与性能评价。彭革刚 硕士研究生,专业方向:网络路由算法的优化与性能评价。

$B_i, i=1, 2, \dots, k-1$, 所组成。多受限的优化问题(MCOP)就是对于给定的一个 QoS 请求 $R=(s, t, B_1, B_2, \dots, B_{k-1})$ 来寻找一条从源节点到目的节点的路径使得 $F_i(P) \leq B_i, i=1, 2, \dots, k-1$ 同时 $F_k(P)$ 是最小(或最大)。

如果对于网络中的任何路径 $P=\{w_1, w_2, \dots, w_k\}$ 都满足(1)式, 我们称函数 F_i 是具有可加性的。可加性参数的一个例子就是传输延迟。

$$F_i(P) = F_i(c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k)) = c_i(w_1, w_2) + c_i(w_2, w_3) + \dots + c_i(w_{k-1}, w_k) \quad (1)$$

如果对于网络中的任何路径 $P=\{w_1, w_2, \dots, w_k\}$ 都满足(2)式或(3)式, 我们称函数 F_i 是具有可凹性的。带宽是可凹性参数的一个例子。

$$F_i(P) = F_i(c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k)) = \max\{c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k)\} \quad (2)$$

$$F_i(P) = F_i(c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k)) = \min\{c_i(w_1, w_2), c_i(w_2, w_3), \dots, c_i(w_{k-1}, w_k)\} \quad (3)$$

我们能够用一般为 NP 难的整数线性规划问题来系统地阐述多受限优化问题。如果限制条件个数为 1 同时这个参数是可加性的, 那么这个问题变成了众所周知的最短路径问题。该问题能够用时间复杂度为 $O(m \log n)$ 的 Dijkstra 算法加以解决^[9]。如果限制条件个数为 1 同时这个参数是可凹性的, 那么这个问题变成一个最大带宽路径问题。该问题能够用一个时间复杂度为 $O(m \log n)$ 的改进 Dijkstra 算法加以解决^[6]。在另一方面, 如果有大于 1 个的可加性参数, 那么该问题变成了 NP 难的^[7]。

2 一个用于 MCOP 问题的有效算法

EAMCOP 算法从源节点 S 开始进行节点扩展。然后检测扩展节点(最开始为源节点)的每一个邻接点, 并将满足扩展条件的邻接点激活, 使之成备选的扩展节点, 加入到当前的备选扩展节点表中, 然后在备选扩展节点表中确定下一个扩展节点继续搜索。当下一个扩展节点是目的节点时, 算法输出满足各约束条件的最优路径。备选扩展节点表为空时算法结束, 通知没有发现满足各约束条件的最优路径。

下一个扩展节点的确定原则体现在备选扩展节点表的数据结构中。在 EAMCOP 算法中我们采用优先队列的数据结构, 按优先级大小来扩展备选节点, 优先级最大的备选节点最先成为扩展节点。

在 EAMCOP 算法的优先队列中我们以传输费用的估值函数 $c(x) = f(x) + g(x)$ 作为主优先级, 其中 $f(x)$ 为从源节点到节点 x 已用的传输费用, $g(x)$ 为节点 x 到目的节点的最小传输费用, 以长度函数 $l(p') = \sum_{(i,j) \in p'} \sum_{k=1}^K \lambda_k w_k(i, j)$ 作为次优先级, 其中 p' 为当前扩展路径, $\lambda_k \geq 0$, 并定义了下面几个优先队列的基本运算:

- (1) MAKENULL(P): 将优先队列 P 置空;
- (2) EMPTY(P): 判断优先队列是否为空;
- (3) INSERT(x, P): 将具有主优先级 $p(x)$ 的节点 x 插入到优先队列 P 中, 在插入过程中当节点的主优先级相同时, 次优先级大的节点排在优先队列的前面。
- (4) DELETEMIN(P): 取出优先队列 P 中主优先级最大的节点, 并将其从优先队列 P 中删除, 当节点的主优先级相同时, 次优先级大的节点优先输出。

为了减少搜索空间大小和提高算法的效率, 在算法中我

们使用了有效的限界剪枝策略使只有当节点的 P'_k 值满足(4)式时, 该节点才能进入备选扩展节点表。

$$P'_k \leq C_k \quad k=1, 2, \dots, K \quad (4)$$

$$P'_k = P'_{1k} + P'_{2k} \quad k=1, 2, \dots, K \quad (5)$$

其中 P'_{1k} 表示当前扩展路径 P' 的各个链路的可加性参数值 w_k 之和, 即

$$P'_{1k} = \sum_{(i,j) \in P'} w_k(i, j) \quad k=1, 2, \dots, K \quad (6)$$

P'_{2k} 表示当前扩展节点 v 到目的节点的关于可加性参数值 w_k 的最小值, 即:

$$P'_{2k} = \min\{\sum_{(i,j) \in P} w_k(i, j) \mid P \text{ is a path in } G \text{ from } v \text{ to } D\} \quad (7)$$

EAMCOP 算法的具体描述如下:

输入: 网络拓扑图 $G=(V, E)$ 和一个 QoS 请求 $R=(S, D, C_1, C_2, \dots, C_k)$ 。

输出: 满足请求 R 的最优路径。

Step1: 对网络中的每个节点 v 计算其到目的节点 D 的关于传输费用的最小值:

$$Cost-D[v] = \min\{\sum_{(i,j) \in P} c(i, j) \mid P \text{ is a path in } G \text{ from } v \text{ to } D\}$$

Step2: 对网络中的每个节点 v 计算其到目的节点 D 的关于可加性参数值 w_k 的最小值:

$$P'_{2k}[v] = \min\{\sum_{(i,j) \in P} w_k(i, j) \mid P \text{ is a path in } G \text{ from } v \text{ to } D\}$$

Step3: 在网络 G 中寻找满足 QoS 请求 R 的路径

```

MAKENULL(Q);
INSERT(S, Q);
While (not EMPTY(Q))
    visting_node := DELETEMIN(Q);
    If (visting_node = D) then
        Output the routing path P and terminate Algorithm;
    Else
        For (all adjacent nodes of node visting_node which satisfy
            (4) ) Compute  $P'_{1k}$  of node X; /* node X denotes one of adjacent nodes of node visting_node which satisfies (4) */
            INSERT(X, Q)
    
```

Step4 report ("QoS request cannot be satisfied").

3 模拟结果和讨论

为了说明算法 EAMCOP 的性能, 作者用模拟作为评价算法的主要工具, 给出了算法 EAMCOP 与算法 H-MCOP 的一个性能比较^[8]。

3.1 模拟环境

在模拟实验中, 考虑链路两个可加性参数的情况。网络中各链路上的可加性参数值和传输费用都随机产生且服从表 1 所示的均匀分布^[8]。QoS 请求的源节点和目的节点也随机产生同时使两节点之间的跳数大于等于 3。QoS 请求中的其它参数也随机产生同时服从表 1 所示的分布。

表 1 模拟中各参数的分布

可加性参数的分布	传输费用分布	QoS 请求中的其它参数分布
$w_1(i, j) \sim$ uniform[1, 20]	$C(i, j) \sim$ uniform[1, 100]	$C_1 \sim [1, 60]$
$w_2(i, j) \sim$ uniform[1, 40]		$C_2 \sim [10, 80]$

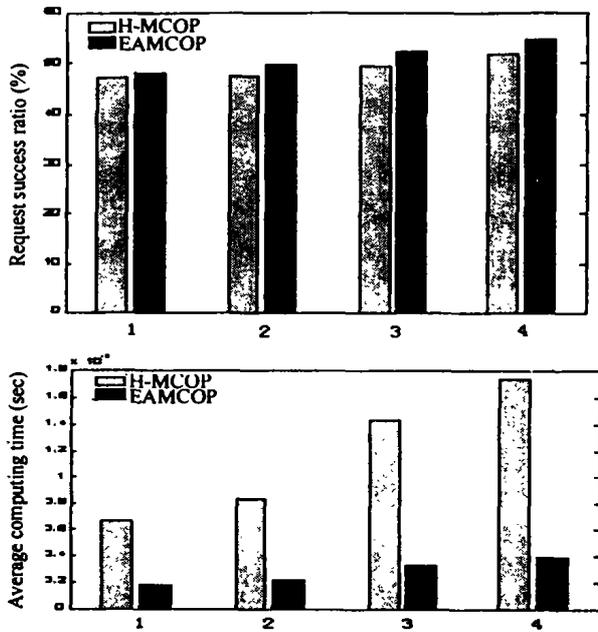
模拟主要分为两大组进行, 第一组模拟主要在一些经典的网络拓扑上进行^[9]。该网络拓扑结构是美国 Internet 网络主干形式, 包含 28 个节点和 45 条链路, 网络直径为 7, 节点平均出度为 3.2。

第二组模拟则是在一组随机网络拓扑上进行的, 作者使用 GT-ITM 拓扑生成器得到这组基于 Waxman 模型^[10, 11]的随机网络拓扑结构。

算法 EAMCOP 和 H-MCOP 的性能分别取决于模拟所使用的堆大小和参数 λ 。堆大小和参数 λ 值越大, 各自算法的性能就越大, 同时也带来了更大的计算开销。考虑到这些因素

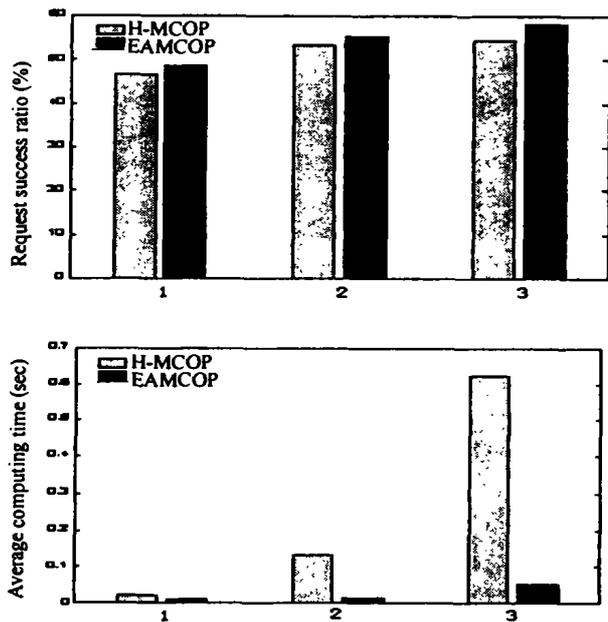
同时为了对两种算法做出更好的性能比较,在模拟中我们取 $\lambda=20$,同时对于第一组模拟取 $heapsize=10$,而对于第二组模拟取 $heapsize=30$ 。

3.2 模拟结果



1--ISP; 2--ARPA; 3--Mesh; 4--U-I
图1 不同经典拓扑下的模拟结果

图1是在经典网络拓扑上的模拟结果,从中我们可以看到算法 EAMCOP 的请求成功率优于 H-MCOP 算法而它的平均计算时间却远远低于 H-MCOP 算法。这种趋势在 Mesh 和 U-I 两个经典网络拓扑中表现得更为明显。这是因为 EAMCOP 算法通过有效的限界剪枝策略大大减少了搜索空间的大小,从而极大地提高了算法的性能。



1--50个节点; 2--100个节点; 3--200个节点
图2 不同随机拓扑下的模拟结果

图2是在一些随机网络拓扑上的模拟结果,从中我们可以看到随着网络拓扑结构的复杂化,在请求成功率方面,

EAMCOP 算法越来越优于 H-MCOP 算法,在平均计算时间方面, H-MCOP 算法的平均计算时间在明显增加,而算法 EAMCOP 的平均计算时间变化不是很大,这是因为当网络的规模增大时, EAMCOP 算法通过约束条件和优先队列大大减少了无效搜索,有效地控制了搜索空间的大小,提高了求解效率。这种变化趋势充分说明了 EAMCOP 是一种解决 MCOP 问题的有效算法。

为了对算法进行更好的比较,在模拟过程中又引入了 FS(Full Search)算法,FS 算法的基本思想是从请求的源节点开始搜索到请求目的节点的所有可能路径,当找到满足 QoS 请求最优的路径后,输出路径,结束本次搜索。显然,这种算法具有指数级的时间复杂度。同时为了更好地说明堆的大小与 EAMCOP 算法的性能关系,在图中作者用两条不同的曲线来描述算法 EAMCOP 的性能。其中一条称为可行路径曲线,它表示的是可行路径的请求接受率。而另一条曲线称为最优路径曲线,它表示的是最优路径的请求接受率。图3和图4分别是节点数为100和200的随机拓扑上的模拟结果,从中能够看到,随着堆的大小的不断增加,可行路径曲线和最优路径曲线都不断逼近最优算法 FS 的曲线。在图3中当 $heapsize=20$ 时,可行路径曲线的请求成功率就基本达到了最优算法 FS 的水平并逐渐趋于稳定, $heapsize=60$ 时,最优路径曲线的请求成功率也与算法 FS 基本相等并趋于稳定状态。在图4中也能够观察到类似的整体趋势。这充分说明了: EAMCOP 算法的性能取决于堆大小 $heapsize$,随着 $heapsize$ 的增大, EAMCOP 算法的性能很快达到最佳,并趋于稳定,但是需要更多的计算时间和空间。

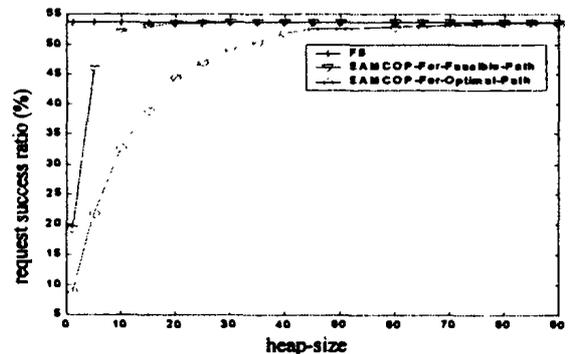


图3 在100个节点随机拓扑中的请求接受率

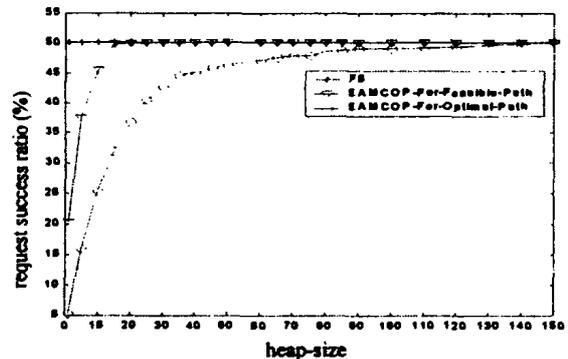


图4 在200个节点随机拓扑中的请求接受率

结论 在本文中,作者提出了一种新的多受限优化路径选择算法。该算法通过有效的限界剪枝策略大大减少了搜索空间的大小,极大地提高了算法的性能,通过模拟测试进一步

一种适于 Internet 拥塞控制的自校正队列管理算法

A Self-Tuning Queue Management Algorithm for Internet Congestion Control

张敬轺 谢剑英 刘立祥

(上海交通大学电子信息学院 上海 200030)

Abstract As an effective method for congestion control, Active Queue Management plays an important role in improving the Internet QoS. In this paper, we first analyze the properties of RED, then design an effective queue management algorithm based on gradient descent approach. With the application of this algorithm, routers in IP network adjust its packet drop probability according to the queue length in the buffer. The main advantage of this algorithm is that the queue length can keep stable at a low level in a variety of network environments. Simulations show that this self-tuning queue management algorithm is efficient, stable and outperforms RED queue management algorithms significantly.

Keywords Active queue management, Congestion control, Self-tuning, Gradient descent

1. 引言

近年来,随着计算机和网络技术的迅猛发展以及多媒体应用的急剧增加,人们对 Internet 的服务质量提出了更高的要求。虽然目前 TCP 所采用的基于窗口的端到端拥塞控制机制对 Internet 的鲁棒性起到了关键性的作用^[1]。但传统的去尾(drop-tail)先进先出(FIFO)的队列管理方式,不可避免地会导致过高的传输延迟和延迟抖动。为适应越来越多的实体媒体传输的要求,人们开始研究更为有效的队列管理算法,从而使网络在采用 TCP 拥塞控制算法的基础上,实现效率最高并尽可能减小路由器中的平均队列长度,即主动队列管理技术(Active Queue Management, AQM)。典型的主动队列管理算法如随机早期检测算法(Random Early Detection, RED),根据路由器中的队列长度调整数据包的丢弃或标记(mark)概率,从而提高 TCP 连接的吞吐量。但该算法需要复杂的参数调节,而且对网络状态的变化不具备适应能力。为解决上述问题,本文采用梯度下降方法设计了一种有效的自校正主动队列管理算法,该算法不需经过复杂的参数调节便可根据网络状况实时调节算法的参数,从而使路由器中的队列长度在不同的网络状况下都能稳定在参考值附近。

2. 随机早期检测(RED)算法

RED 算法在每个路由器监控数据包的排队长度,并在缓

存占满之前就按一定的概率丢弃进入路由器的数据包,这样就可以及早通知源端减小拥塞窗口,以减少进入网络的数据流量。对于面向连接的 TCP 数据流来说,RED 有可能避免丢弃属于同一连接的连续数据包。从而避免多个数据流的同步,提高连接的吞吐量。RED 算法主要包含两部分^[2]:

1)平均队列长度的计算 Internet 上数据流量的突发性,使用平均队列长度比使用瞬时值更能精确获取链路的拥塞情况。RED 算法根据权值 q_w , 通过(1)式计算平均队列长度 $ave-q$ 。

$$ave-q = (1-q_w) * ave-q + q_w * n_{queued} \quad (1)$$

其中, n_{queued} 为采样时刻的队列长度。式(1)的加权平均队列长度的计算类似于一个低通滤波器,权值决定滤波器的时间常数。

2)计算数据包的丢弃概率 根据(1)式所计算的平均队列长度,RED 算法的丢弃概率如(2)式所示。

$$p = \begin{cases} 0, & 0 \leq ave-q \leq Q_{min} \\ \frac{ave-q - Q_{min}}{Q_{max} - Q_{min}} P_{max}, & Q_{min} \leq ave-q \leq Q_{max} \\ 1, & Q_{max} \leq ave-q \leq B \end{cases} \quad (2)$$

其中, Q_{min} , Q_{max} 为阈值, P_{max} 为最大丢弃概率, B 为缓存大小(如图 1)。

对于上述 RED 算法,参数 q_w , Q_{min} , Q_{max} , P_{max} 都可根据需要进行调节。但自从 RED 出现以来,其参数调节一直是一

张敬轺 博士生,主要研究方向为网络传输控制及多媒体技术应用。谢剑英 教授,博导,主要研究方向为复杂工业过程控制及计算机网络与信息处理。刘立祥 博士生,主要研究方向为信号处理。

说明了该算法的有效性。虽然提出的算法在最差情况下具有指数级的时间复杂度,但是在实际的网络中该算法能够取得良好的性能,可以有效地解决 MCOP 问题。

参考文献

- Chen S, et al. An overview of quality-of-service routing for the next generation high-speed networks: Problems and solutions. *IEEE Network*, 1998, 12(6): 64~79
- Apostolopoulos G, et al. Quality of service based routing: A performance perspective. In: *ACM SIGCOMM '98 Conference*, Vancouver, British Columbia, Canada, Aug. 1998, 17~28
- Crawley E, et al. A framework for QoS-based routing in the Internet. *Internet draft*, IETF, July 1998
- Moy J, et al. OSPF version 2. RFC 2328, IETF, April 1998
- Vogel R, et al. QoS-based routing of multimedia streams in computer networks. *IEEE Journal on Selected Areas in Communications*, 1996, 14(7): 1235~1244
- Cormen T H, et al. *Introduction to Algorithms*, 2nd Edition. McGraw-Hill, Boston, 2001
- Wang Z, et al. OoS routing for supporting resource reservation. *IEEE Journal on Selected Areas in Communications*, 1996, 14(7): 1228~1234
- Korkmaz T, et al. Multi-constrained optimal path selection. In: *IEEE INFOCOM 2001 Conf.* Anchorage, Alaska, 2001
- Rampal S, et al. An evaluation of routing and admission control algorithms for multimedia traffic. *Computer Communication*, 1995, 18(10): 755~768
- Waxman B. M. Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications*, 1988, 6(12): 1617~1622
- Calvert K I. Modeling internet topology. *IEEE Communications Magazine*, 1997, 35(6): 160~163