

## 一种 Internet 优化速率控制机制

袁刚 王玉峰 王文东 程时端

(北京邮电大学国家重点实验室 北京 100876)

**摘要:** 该文提出了一种 Internet 优化速率控制机制 Morcs, 通过设计合理的链路拥塞费用函数, 实现了社会福利最大化的系统优化目标。理论和仿真均证明了 Morcs 机制的收敛性和稳定性, 仿真更进一步证明了 Morcs 比 REM 算法更稳定。本文接着进一步提出了 Morcs 机制的自适应实现方式 aMorcs。aMorcs 不需要网络链路的参与, 完全在用户端实现, 具有更好的可实现性, 仿真表明 aMorcs 具有和 Morcs 相似的性能。

**关键词:** 因特网, 优化速率控制, 社会福利, 比例公平性

**中图分类号:** TP393 **文献标识码:** A **文章编号:** 1009-5896(2005)02-0251-05

## An Optimal Rate Control Scheme in the Internet

Yuan Gang Wang Yu-feng Wang Wen-dong Cheng Shi-duan

(National Key Lab of Switching Tech. and Telecom. Networks,  
Beijing University of Posts & Telecom., Beijing 100876, China)

**Abstract** An optimal rate control scheme, which is referred to as Morcs, is proposed in this paper. Morcs can achieve system optimality of social welfare maximization by means of well-designed link pricing function, which is testified experimentally and theoretically. Furthermore, aMorcs, the adaptive version of Morcs, is proposed. Since aMorcs does not need the participation of links, it can be implemented more easily. Experimentations testify that aMorcs is similar in property to Morcs.

**Key words** Internet, Optimal rate control, Social welfare, Proportionally fair

### 1 引言

在 Internet 中有效控制业务流的发送速率, 特别是弹性业务<sup>[1]</sup> (如 Internet 中使用 TCP 传输的业务) 的速率, 对于网络拥塞控制来说是非常重要的。当前 Internet 常用的速率控制机制是著名的 TCP 拥塞控制机制。然而研究发现, TCP 机制采用的是黑盒式的处理方法, 它以分组丢失 (3 次重复 ACK 或超时) 作为网络拥塞指示, 终端节点逐步增加发送速率直到网络发生拥塞, 并发现包丢失, 再减小发送速率。这种黑盒式处理方法很容易在网络瓶颈造成数据堆积, 导致网络的严重拥塞, 产生大量的数据包丢失, 从而严重降低网络的利用率。另外, TCP 并不能以公平的方式在用户间分配资源。

最近提出的优化速率控制机制<sup>[2-6]</sup> 借鉴微观经济学原理<sup>[7]</sup>, 通过效用函数的形式来表示用户的带宽需求, 网络以用户效用函数的总和 (即社会福利函数) 作为优化的目标函数, 网络根据当前的拥塞状况不断向用户反馈拥塞费用, 用户在利益的驱动下 (如最大化利润), 根据反馈的费用不断调整用

户速率, 最后达到系统优化的均衡状态。研究表明, 优化速率控制机制可以实现用户之间资源共享的比例公平性<sup>[2-5]</sup>。

Kelly 最早在文献[2,5]中提出优化的速率控制问题。Low 等人提出了著名的(Random Early Marking, REM)算法<sup>[3,8]</sup>, 该算法以社会福利函数作为系统优化的原问题, 利用拉格朗日对偶方法得出原问题的对偶问题, 并通过每个用户分布式求解各自的优化目标函数来得出对偶问题的最优解, 最后利用 KuhnTucker 对偶理论<sup>[9]</sup>得出原问题的最优解, 从而实现优化的速率控制。REM 算法<sup>[8]</sup>属于对偶算法。

本文将提出一个新的优化速率控制机制——最小成本优化速率控制机制 (Minimal cost optimization rate control scheme, Morcs), 直接从用户的目标函数出发, 用户以最小化各自的成本函数为唯一目标, 根据网络反馈的拥塞计费来调整用户速率; 网络通过设计合理的拥塞计费函数, 引导用户行为到系统优化的操作点上 (即社会福利最大化), 从而达到系统优化的目标。接着本文进一步提出 Morcs 机制的用户

自适应实现方式(adaptive Morcs, aMorcs), aMorcs 不需要网络链路的参与, 完全在用户端系统实现, 从而具有更好的可实现性。

## 2 最小成本优化速率控制机制

### 2.1 参数设定

考虑一个通用拓扑的 Internet 网络, 网络的单向链路集合  $L = \{1, \dots, L\}$ 。网络用户集合  $S = \{1, \dots, S\}$ , 这里的用户指的是某个应用实例。考虑离散时间模式,  $t=1, 2, \dots$  表示每个离散时间周期的开始。

(1) 对于每个用户  $s \in S$ , 定义以下相关参数:  $L_s$  为用户  $s$  使用的链路集合,  $L_s \subseteq L$ ;  $x_s(t)$  为用户  $s$  的发送速率,  $x_s \in I_s$ ,  $I_s = [m_s, M_s]$ ,  $0 \leq m_s \leq M_s < +\infty$ , 其中  $m_s$ ,  $M_s$  分别表示用户  $s$  的最小和最大发送速率;  $U_s(x_s)$  为用户  $s$  的效用函数, 表示在发送速率为  $x_s$  时用户得到的效用为  $U_s(x_s)$ ;  $p^s(t)$ : 用户  $s$  的单位带宽总费用, 等于用户  $s$  使用的各链路的单位带宽费用之和。  $J_s(x_s)$  为用户  $s$  的成本函数, 即用户的费用函数与效用函数之差:

$$J_s(x_s) = p^s(t) \cdot x_s(t) - U_s(x_s(t))$$

(2) 对于每条链路  $l \in L$ , 定义以下一些相关参数:  $c_l$  为每条链路的容量;  $S_l$  为使用链路  $l$  的用户集合, 即  $S_l = \{s \in S | l \in L_s\}$ , 显然  $s \in S_l \Leftrightarrow l \in L_s$ ;  $x^l(t)$  为链路  $l$  的汇聚速率, 即  $x^l(t) = \sum_{s \in S_l} x_s(t)$ ;  $b_l(t)$  为链路  $l$  的缓冲区排队长度;  $d_l(t)$  为链路  $l$  的排队时延;  $p_l^0$  为链路  $l$  收取的单位带宽基本费用, 对链路  $l$  来说,  $p_l^0$  是常数;  $p_l^c(t)$  为链路  $l$  在拥塞时收取的单位带宽拥塞费用,  $p_l^c(t) = \gamma d_l(t)$ ;  $p_l(t)$  为链路  $l$  的单位带宽总费用,  $p_l(t) = p_l^0 + p_l^c(t)$ , 用流体力学模型可以得出  $d_l(t) = b_l(t)/c_l$ , 因此  $p_l(t) = p_l^0 + \gamma b_l(t)/c_l$ 。

(3) 对于整个网络系统, 定义以下相关参数:

社会福利函数为所有用户效用函数的总和, 即  $\sum_{s \in S} U_s(x_s)$ ;  $A = [a_{sl}]$  为一个描述网络用户  $s$  与链路  $l$  的关系的 0-1 路由矩阵, 元素  $a_{sl} = 1$  表示用户  $s$  使用链路  $l$ , 否则  $a_{sl} = 0$ ;  $X = [x_s]$  为描述网络所有用户速率的  $(S \times 1)$  向量;  $C = [c_l]$  为描述网络所有链路容量的  $(L \times 1)$  向量。

### 2.2 最小成本优化速率控制机制

本文提出的 Morcs 机制的系统优化目标是在网络不拥塞的情况下最大化社会福利函数, 这也是 Internet 优化速率控制机制最常用的系统优化目标函数, 表示如下:

$$\begin{aligned} & \max_{x_s \in I_s} \sum_{s \in S} U_s(x_s) \\ & \text{subject to } A^T X \leq C \end{aligned}$$

用户  $s$  的成本函数

$$\begin{aligned} J_s(x_s, t) &= p^s(t) \cdot x_s(t) - U_s(x_s(t)) \\ &= x_s(t) \cdot \sum_{l \in L_s} p_l(t) - U_s(x_s(t)) \end{aligned} \quad (1)$$

根据每个用户最小化成本函数的目标, 用户在进行速率调整时采用最速下降法<sup>[9]</sup>:

$$\begin{aligned} x_s(t+1) &= x_s(t) + \alpha(-\partial J_s / \partial x_s) \\ &= x_s(t) - \alpha \cdot [p^s(t) - U'_s(x_s(t))] \end{aligned} \quad (2)$$

式(2)表明用户速率  $x_s(t)$  的变化方向与用户成本函数  $J_s(x_s)$  的变化方向是相反的, 即用户速率沿着成本函数负梯度的方向变化, 这样就可以达到最小化成本函数的目的。

Morcs 机制将借鉴 REM 算法<sup>[8]</sup>的费用反馈方法, 即只用一个比特位就可以实现网络费用的反馈。描述如下: 链路  $l$  以概率  $m_l(t) = 1 - \phi^{-n_l(t)}$  ( $\phi > 1$  为常数)来标记每个未标记数据包, 因此用户  $s$  的数据包被标记概率为  $m^s(t) = 1 - \prod_{l \in L_s} (1 - m_l(t)) = 1 - \phi^{-p^s(t)}$ 。用户端可以通过统计包标记概率来估计  $m^s(t)$ , 从而求出总费用  $p^s(t) = -\log_\phi(1 - m^s(t))$ 。

Morcs 机制的用户算法和链路算法分别为

用户算法 在  $t+1$  时间周期:

(1) 计算标记概率  $m_l(t)$  = 标记的包数/在周期  $t$  内接收的总包数;

(2) 计算用户单位带宽总费用:

$$p^s(t) = -\log_\phi(1 - m^s(t))$$

(3) 计算  $J_s(t) = \partial J_s / \partial x_s = p^s(t) - U'_s(x_s(t))$ , 根据式 (2) 计算  $t+1$  周期的发送速率:

$$x_s(t+1) = \begin{cases} [x_s(t) + \alpha \cdot J_s]_{m_s}^{M_s}, & J_s < 0 \\ [x_s(t) - \beta \cdot J_s]_{m_s}^{M_s}, & J_s \geq 0 \end{cases}$$

其中  $[z]_a^b = \min\{\max\{z, a\}, b\}$ ,  $\alpha, \beta$  为调整系数, 且满足  $0 < \alpha \leq \beta$ ;

(4) 调整发送速率为  $x_s(t+1)$ , 并进入下一个周期。

链路算法 在  $t$  时间周期内:

(1) 测量和计算  $b_l(t)$ ,  $b_l = (1 - \delta) \cdot b_l + \delta \cdot b_{l, \text{new}}$

(2) 并计算链路总费用  $p_l(t) = p_l^0 + \gamma b_l(t)/c_l$ ;

(3) 以概率  $m_l(t) = 1 - \phi^{-n_l(t)}$  来标记每个未标记数据包

( $\phi > 1$  为常数)。

与 REM 采用对偶算法相比，我们提出的 Morcs 机制属于原算法，即直接从用户的目标函数出发，用户以最小化各自的成本函数为唯一目标，根据网络反馈的拥塞计费来调整用户速率，最后达到系统优化的目标，因此我们的算法更为简洁。同时由于 Morcs 机制的用户算法采用了慢增快减 (Additive Increase Multiplicative Decrease, AIMD) 的迭代算法，使得 Morcs 机制更加稳定，这将在后面的仿真中得到验证，理论证明参见附录。

### 3 Morcs 机制的用户自适应实现方式

在第 2 节中我们提出了最小成本速率控制机制，并利用 REM 算法中的费用反馈方法，即只用一个比特位就可以将链路费用反馈给用户，虽然给网络带来的负荷非常小，但是仍然需要链路算法的参与，需要 Internet 网络的路由器做出相应的改变，从而增加了实现的难度。因此我们在 Morcs 机制的基础上进一步提出了 Morcs 机制的用户自适应实现方式 aMorcs。该机制不需要链路算法的参与，完全在用户端实现，具有非常好的可实现性。

在 aMorcs 机制中，用户端采用发送探测包的方式来测量源端到目的端之间的环回时延 (round trip time, rtt) 的值 rtt，并计算得出单向排队时延  $d = (rtt - rtt_{min})/2$ ，其中  $rtt_{min}$  表示 rtt 的最小值，显然用户流在网络中不排队时的 rtt 是最小的，因此 rtt 与  $rtt_{min}$  之差可用来近似表示用户流在穿越网络时所经历的所有链路的排队时延之和，由于是环回时延，因此需要减半。

得到了单向排队时延  $d$  就可以计算得到用户的网络费用信息  $p^s(t)$ ，推理如下：

$$\begin{aligned}
 p^s(t) &= \sum_{l \in L_s} p_l(t) = \sum_{l \in L_s} [p_l^0 + \gamma d_l(t)] \\
 &= \sum_{l \in L_s} p_l^0 + \gamma \sum_{l \in L_s} d_l(t) \\
 &= \gamma \cdot d + \sum_{l \in L_s} p_l^0
 \end{aligned}
 \tag{3}$$

式(3)中右边第二项是一个不变的常数，因此得到了单向排队时延  $d$ ，用户就可以自己计算出费用函数  $p^s(t)$ ，并进行相应的速率调整。实际上此时费用函数反映了网络的拥塞状况，用户单向排队时延越大，说明网络越拥塞，因此用户费用  $p^s(t)$  越大，此时用户根据成本最小的原则来调节用户速率，从而缓解网络拥塞，从而起到拥塞控制的作用。

在计算单向排队时延  $d$  的时候，我们用  $d_{new}$  表示最近测量得到的单向排队时延：

$$d_{new} = (rtt - rtt_{min})/2$$

并采取一次滤波的方式来得到平均单向排队时延：

$$d = \delta \cdot d + (1 - \delta)d_{new}, \quad 0 \leq \delta \leq 1$$

aMorcs 的用户算法：在每个 ACK 包到达时：

(1) 测量计算单向排队时延

$$d_{new} = (rtt - rtt_{min})/2$$

$$d = \delta \cdot d + (1 - \delta)d_{new}, \quad 0 \leq \delta \leq 1$$

(2) 计算用户单位带宽总费用： $p^s(t) = \sum_{l \in L_s} p_l^0 + \gamma \cdot d$ ；

(3) 计算  $J_s(t) = \partial J_s / \partial x_s = p^s(t) - U'_s |_{x_s(t)}$ ，根据式(2)计算  $t+1$  周期的发送速率

$$x_s(t+1) = \begin{cases} [x_s(t) + \alpha \cdot J_s]_{m_s}^{M_s}, & J_s < 0 \\ [x_s(t) - \beta \cdot J_s]_{m_s}^{M_s}, & J_s \geq 0 \end{cases}$$

其中  $[z]_a^b = \min\{\max\{z, a\}, b\}$ ， $\alpha, \beta$  为调整系数，且满足  $0 < \alpha \leq \beta$ ；

(4) 调整发送速率为  $x_s(t+1)$ ，并进入下一个 rtt 周期。

### 4 仿真结果

我们通过 OPNET 仿真来验证 Morcs 和 aMorcs 机制的收敛性和稳定性。仿真采用的网络拓扑如图 1 所示，网络由三个路由器(n1, n2 和 n3)和两条链路(link1 和 link2)组成，三个用户 src1, src2 和 src3 分别通过不同的链路发送数据到相应的目的端(d1, d2 和 d3)。link1 的容量为 50 包/毫秒 (pkts/ms)，link2 的容量为 20pkts/ms，其它链路容量为 100pkts/ms。数据包长度满足负指数分布，平均包长 1 kbyte。Link1 和 link2 的传播时延均为 1ms，接入链路的传播时延为 2ms，src1, src2 和 src3 的开始时间分别为 1ms, 1000ms 和 2000ms。链路的缓存管理采用简单的 FIFO 方式，并假设缓存足够大，不会产生丢包。

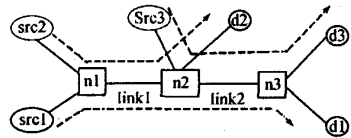


图 1 仿真拓扑 1

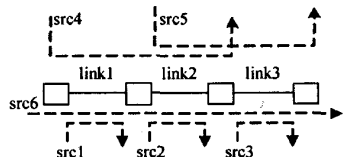


图 2 仿真拓扑 2

仿真中采用的用户效用函数为常用的对数函数  $U_s(x_s) = w_s \ln x_s$ ， $w_s$  表示用户效用的权重，其中

$w_2 = w_3 = 12$ 。Morcs 用户算法的调节参数设为  $\alpha = 0.3, \beta = 2, \phi = 1.07$ 。链路算法的参数设置如下： $\lambda = 0.05, \delta = 0.1$ ，链路固定费用与容量成反比： $p_1^0 = 0.02, p_2^0 = 0.05$ 。

仿真结果如图 3 和图 4 所示，可以看出 Morcs 和 aMorcs 算法是收敛和稳定的，均收敛到系统优化速率分配 (10, 40, 10) 附近，并满足比例公平性。这说明 aMorcs 通过 rtt 的变化完全可以得到用户所经过链路费用信息，从而省掉链路算法。aMorcs 算法的用户速率波动稍大，这是由于 aMorcs 算法利用 rtt 的变化来估计排队时延，而 rtt 的变化会受到其它因素的影响，这在实际网络中更加明显 (如链路非对称性、噪声干扰等)。

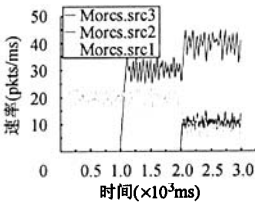


图 3 Morcs 算法的收敛性

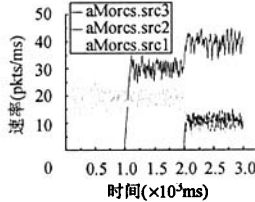


图 4 aMorcs 算法的收敛性

图 5 是在相同条件下 Morcs 和 REM 算法的收敛性和稳定性的比较。可以看出：Morcs 和 REM 算法均收敛到系统优化的速率值 (或附近)，Morcs 的收敛速率稍慢，但是稳定性要远优于 REM。这是由于 (1) Morcs 机制的用户算法采用了类似于 AIMD 的迭代算法；(2) 每条链路的固定费用可以使链路算法中的标记概率  $m_i = 1 - \phi^{-p_i}$  偏离极值 0，从而降低了统计误差 (REM 中用户速率在开始阶段的毛刺现象主要是由该原因造成)。

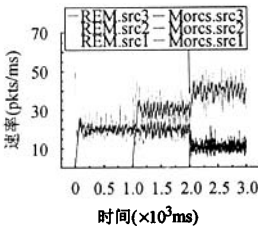


图 5 Morcs 算法与 REM 算法的比较

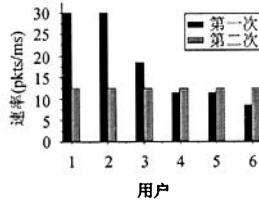


图 6 Morcs 算法的公平性

下面通过仿真来验证 Morcs 和 aMorcs 机制的公平性，仿真采用图 2 所示的网络拓扑，用户数  $n=6$ ，link1, link2 和 link3 的链路容量均为 50pkts/ms，其它链路的容量为 100pkts/ms，用户效用函数  $U_s(x_s) = w_s \ln x_s$ 。在第一次仿真中  $w_3 = 12$ ；第二次仿真中  $w_2 = w_4 = 2.5, w_1 = 7.5, w_3 = 12.5, w_5 = w_6 = 5$ 。链路固定费用均设为 0.2，用户算

法和链路算法的其它参数设置同上。从图 6 中可以看出，第一次用户速率满足比例公平性，即  $\sum_{s \in S} \frac{x_s^* - x_s}{x_s} \leq 0$ ；而第二次仿真中所有的用户速率几乎相等，满足最大最小公平性。这说明通过设置不同的权值 (即用户愿意付出的单位带宽费用)，Morcs 机制可以达到不同的公平性指标。

最后需要指出一点的是，由于优化速率控制机制的前提是价格杠杆的调节作用，即只有经济意义上的费用才能起到约束用户行为的作用，才能促使用户调整速率，以达到拥塞控制和优化系统的目标，因此优化速率控制是与网络计费密切相关的。Internet 优化速率控制和网络计费的关系，将影响优化速率控制机制的可实现性，这是一个值得下一步研究的课题。

### 附录 Morcs 机制收敛性和稳定性的证明

下面证明 Morcs 机制是收敛和稳定的，为此先给出几个前提条件：

C1:  $U_s(x_s)$  在  $I_s$  上是  $x_s$  的连续、递增以及严格的凹函数，并且二次可微分；

C2: 用户的单位带宽费用  $p^s(t)$  在  $I_s$  上是  $x_s$  的连续、递增凸函数，并且二次可微分。

命题 在条件 C1 和 C2 成立的前提下，Morcs 的用户算法和链路算法产生的用户速率向量  $x$  收敛到社会福利最大化的系统优化操作点  $x^*$ ，并且  $x^*$  是全局稳定的。

证明 由 C1 和 C2 可知  $J_s(x_s, t)$  是  $x_s$  的严格凸函数，约束条件  $x_s \in I_s$  和  $A^T X \leq C$  均是线性函数，因此用户在线性约束条件下最小化成本函数的问题 (后面称为用户问题) 属于凸规划问题。易证满足链路容量限制条件的  $x_s$  的集合是凸集，又由于  $I_s$  是闭集和凸集，所以用户问题存在最优解。将多个用户最小化成本函数的问题模拟成非协作博弈，则根据文献 [10] 中定理 3.1，可知用户问题的最优解 (纳什均衡)  $x^*$  是唯一的，且满足  $J_s(x^*, t) = 0$ ，即

$$p^s(t) = U'_s(x_s(t)) \tag{A-1}$$

文献 [3] 在定理 1 中证明了在链路拥塞费用函数  $p_l(t) = \gamma(x^l(t) - c_l)$  且用户速率满足 K-T 条件，即式 (A-1) 时，用户问题的最优解也是系统问题的最优解 (即动机一致性，因此只需证明在 Morcs 链路算法的费用函数  $p_l(t) = p_l^0 + \gamma b_l(t) / c_l$  定义下的下降算法，也能收敛到文献 [3] 文中拉格朗日对偶问题的最优解。

定义  $L \times L$  对角阵： $K \triangleq \begin{pmatrix} 1/c_1 & 0 & \dots & 0 \\ 0 & 1/c_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/c_L \end{pmatrix}$ ，则

$p_i(t) = \gamma K(x'(t) - c_i)$ , 文献[3]中对偶问题的负梯度方向记为  $g$ , Morcs 链路算法的下降方向记为  $p$ , 则有:  $p = K \cdot g$ , 由于  $K$  的元素  $1/c_i > 0$ , 因此  $p$  与  $g$  之间的夹角  $\theta < \pi/2$ , 根据文献[9] (p22.的式 2.17) 可知,  $\bar{p}$  充分下降到对偶问题的最优值。因此 Morcs 链路算法费用函数也能收敛到文献[3]中拉格朗日对偶问题的最优解, 从而证明了 Morcs 算法产生的均衡点  $x^*$  也是系统问题 (文献[3]中的原问题) 的最优解, 即满足动机一致性。

Morcs 用户算法属于梯度算法, 根据文献[10]中的定理 4.2, 梯度算法下的纳什均衡是全局稳定的, 因此 Morcs 的用户算法是全局稳定的。

证毕

### 参考文献

- [1] Shenker S. Fundamental design issues for the future Internet. *IEEE J. on Selected Areas in Communications*, 1995, 13(7): 1176 - 1188.
- [2] Kelly F P, Maulloo A, Tan D. Rate control for communication networks: shadow prices, proportional fairness and stability. *J. of the Operational Research Society*, 1998, 49(3): 237 - 252.
- [3] Low S H, Lapsley D E. Optimization flow control, I: Basic algorithm and convergence. *IEEE/ACM Trans. on Networking*, 1999, 7(9): 861 - 874.
- [4] La R J, Anantharam V. Utility-based rate control in the Internet for elastic traffic. *IEEE/ACM Trans. on Networking*, 2002, 10(2): 272 - 286.
- [5] Kelly F P. Charging and rate control for elastic traffic, *European Trans. on Telecommunications*, 1997, 8(1): 33 - 37.
- [6] Kunniyur S, Srikant R. End-to-end congestion control schemes: Utility functions, random losses and ECN marks. In Proceedings of Infocom 2000, TelAviv, Israel, March 2000: 1323 - 1332.
- [7] 朱善利. 微观经济学. 第二版. 北京: 北京大学出版社. 2001: 296 - 313.
- [8] Lapsley D,E, Low S,H. Random early marking for Internet congestion control. In Proceedings of IEEE Globecom'99, Riode Janeiro, Brazil, December 1999: 1747 - 1752.
- [9] 席少霖. 非线性最优化方法. 第一版. 北京: 高等教育出版社. 1992: 22 - 25.
- [10] Alpcan T, Basar T. A game-theoretic framework for congestion control in general topology networks. In Proc. 41st IEEE Conference on Decision and Control, Las Vegas, Nevada, December 2002.

袁刚: 男, 1975年生, 博士生, 研究方向: Internet 优化流量控制、P QoS 控制、流量工程以及 IP 网络管理等。

王玉峰: 男, 1974年生, 博士生, 研究方向: 基于经济模型与理论的移动互联网的资源管理、TCP/IP 协议改进、QoS 保障的策略及其实现机制等。

王文东: 男, 1963年生, 教授, 研究方向: ISDN、ATM 和 IP 网的话音通信技术、高速网络和因特网通信协议、包括实时多媒体业务控制信令、IP QoS 及 IP 网络管理等。

程时端: 女, 1940年生, 教授, 博士生导师, 主要研究方向: ISDN、ATM、ATM 和 IP 网的话音通信技术、协议工程、业务量工程、宽带网络性能和服务质量等。