

基于 D 稳定域和 ITAE 准则的主动队列管理算法*

王秀利^{1,2,3}, 王永吉¹⁺, 周 辉^{1,2}, 蔡开元⁴

¹(中国科学院 软件研究所,北京 100080)

²(中国科学院 研究生院,北京 100049)

³(中央财经大学 信息学院,北京 100081)

⁴(北京航空航天大学 自动化科学与电气工程学院,北京 100083)

Optimal Design of AQM Routers with D -Stable Regions Based on ITAE Performance

WANG Xiu-Li^{1,2,3}, WANG Yong-Ji¹⁺, ZHOU Hui^{1,2}, CAI Kai-Yuan⁴

¹(Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

²(Graduate University, The Chinese Academy of Sciences, Beijing 100049, China)

³(School of Information, Central University of Finance and Economics, Beijing 100081, China)

⁴(School of Automation Science and Electrical Engineering, BeiHang University, Beijing 100083, China)

+ Corresponding author: Phn: +86-10-62661815, Fax: +86-10-62550138, E-mail: ywang@itech.iscas.ac.cn

Wang XL, Wang YJ, Zhou H, Cai KY. Optimal design of AQM routers with D -stable regions based on ITAE performance. *Journal of Software*, 2007,18(12):3092–3103. <http://www.jos.org.cn/1000-9825/18/3092.htm>

Abstract: Active queue management (AQM) is a hotspot in the current studies on network congestion control. Moreover, the feedback control strategy is the most pivotal. This paper applies a optimization method for proportional-integral-differential (PID) controller design with D -stable regions based on the integral of time-weighted absolute error (ITAE) performance to AQM routers that permits the designer to control the desired dynamic performance of a closed-loop system. A set of desired D -stable regions in the complex plane is first specified and then a numerical optimization algorithm based on ITAE performance is run to find the controller parameters such that all the roots of the closed-loop system are within the specified regions. This controller for AQM routers can detect and control the congestion effectively and predictively. Compared with the random early detection (RED) and proportional-integral (PI) algorithms via experimental simulations, the proposed method, called DITAE-PID method, is indeed more efficient and robust in achieving the lower packet loss rate and higher link utilization.

* Supported by the National Natural Science Foundation of China under Grant No.60373053 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant Nos.2004AA1Z2100, 2005AA113140 (国家高技术研究发展计划(863)); the State Education Ministry Scientific Research Foundation for the Returned Overseas Chinese Scholars under Grant No.[2003]406 (国家教育部留学回国人员科研启动基金); the One-Hundred-Talent Program of the Chinese Academy of Sciences (中国科学院“百人计划”); the Chinese Academy of Sciences and Royal Society of United Kingdom for the Joint Research Project under Grant No.20030389 (中国科学院与英国皇家学会国际合作项目)

Received 2006-04-27; Accepted 2006-11-14

Key words: active queue management; congestion control; PID (proportional-integral-differential) control; D stability; ITAE (integral of time-weighted absolute error)

摘要: 主动队列管理(active queue management,简称 AQM)是网络拥塞控制的研究热点之一,其中的关键问题是设计反馈控制策略.提出一种新的基于 D 稳定域和时间乘以误差绝对值乘积积分(integral of time-weighted absolute error,简称 ITAE)性能准则的比例-积分-微分(proportional-integral-differential,简称 PID)优化设计方法(简称 DITAE-PID),并用于 AQM 控制器的设计,控制闭环系统的理想动态性能.首先在复平面上设定一组理想的 D 稳定域,然后以 ITAE 为目标函数,通过数值优化算法求出控制器的参数,使得闭环系统的所有特征根都在 D 稳定域内,以降低排队延时,提高有效吞吐量.对比仿真实验结果表明,该算法能够预先探测和控制拥塞,有较好的鲁棒性,链路利用率更高,丢包率更小,平均队列长度更趋于期望值,同时,趋于期望队列长度的时间更短,其综合性能明显优于典型的随机早期探测(random early detection,简称 RED)和比例-积分(proportional-integral,简称 PI)算法.

关键词: 主动队列管理;拥塞控制;PID 控制; D 稳定域;ITAE

中图分类号: TP393 **文献标识码:** A

近年来,主动队列管理(active queue management,简称 AQM)成为网络拥塞控制研究中的一个技术热点.它通过网络中间节点有控制的分组丢弃机制,实现了较低的排队延时和较高的有效吞吐量.研究人员提出了多种 AQM 算法.算法设计的关键问题是如何生成反馈信息和对反馈信息如何进行响应.比例-积分(proportional-integral,简称 PI)算法^[1]把 AQM 算法设计问题变成了控制理论中的控制器设计问题.但是,使用 PI 控制器虽然可以消除“稳态误差”,但也会减慢系统的反应速度.当网络中的流量发生很大变化时,使用 PI 控制器需要的收敛时间要远远长于使用比例控制器的情况.从控制理论^[2]中得知,微分环节能够反映系统偏差信号的变化率,具有预见性,可预见偏差变化的趋势,因此能产生超前的控制作用,可以改善系统的动态性能.在微分时间选择合适的情况下,可以减小超调,减少调节时间.在 PI 控制器中加入微分环节,就形成了比例-积分-微分(proportional-integral-differential,简称 PID)控制器.

本文提出一种基于 D 稳定域和时间乘以误差绝对值乘积积分(integral of time-weighted absolute error,简称 ITAE)性能准则的 PID 优化设计方法(简称 DITAE-PID),并用于 AQM 控制器的设计. D 域是复平面上的一个区域,可以是连续的区域,如矩形、双曲线的左支、抛物线等,也可以是分散的圆.对于某个系统,如果 D 域位于复平面的左半平面,则该 D 域称为 D 稳定域^[3]. D 稳定域方法的优越性在于:被控对象可以是高阶系统;被控对象可以带有时滞;控制器设计参数可以超过 3 个.而 TCP/AQM 模型恰为带有时滞的高阶系统.对于控制器设计的性能准则,本文采用 ITAE,因为 ITAE 考虑了时间因素,能够降低初始大误差对性能指标的影响,同时强调了过渡过程后期的误差对指标的影响,响应快、超调小.实验结果证明,DITAE-PID 算法的综合性能明显优于典型的随机早期探测(random early detection,简称 RED)和 PI 算法.

本文第 1 节对相关 AQM 算法进行阐述.第 2 节介绍 ITAE 等 PID 控制器性能准则及 TCP/AQM 模型.第 3 节提出新的 DITAE-PID 主动队列管理算法.第 4 节利用对比仿真实验详细地分析比较了该算法与具有代表性的 RED 算法、PI 算法在多种典型场景下表现出的不同性能.第 5 节总结全文.

1 相关 AQM 算法

为了更好地理解 DITAE-PID 算法并与已有的典型算法进行对比,本节简要阐述 AQM 算法.

Braden 等人提出了主动队列管理 AQM 的研究动议^[4],作为拥塞控制的一种技术手段,期望 AQM 在减小排队时延的同时保证较高的吞吐量.文献[5]对 RED 原型作了两点重大改进:一是采用平均队列长度,而不是实际队列长度作为拥塞程度的指示;二是采用随拥塞程度变化的实际丢包率代替了固定丢包率.RED 符合 AQM 的技术目标,RFC2309 推荐它为主动队列管理的唯一候选算法.但是,RED 对参数设置很敏感,改变参数对性能影响很大^[6].由于 RED 使用“比例控制器+低通滤波器”的方法计算反馈,比例控制器的缺点是控制存在“稳态误

差”,这是平均队列随网络流量增长的主要原因.另外,RED 使用低通滤波器降低了系统的反应速度.

因此,一些研究者又提出了ARED^[7],SRED^[8],MRED^[9],ERED^[10],LARED^[11],BLUE^[12]等算法.ARED 算法根据网络负载的情况调整 \max_p .当平均队列小于 \min_{th} 时,就减小 \max_p ;当平均队列大于 \max_{th} 时,就增大 \max_p .SRED 算法通过估计网络中流的个数来调整报文标记/丢失概率.在 SRED 中,流的个数通过概率统计的方法获得.LARED 算法自适应链路负载.BLUE 算法以“分组丢失率”和“链路有效利用率”作为判别拥塞是否发生的标准.该算法只有一个状态变量:随机丢包率 P 以及两个常量:增大步进值 ΔP_1 和减少步进值 ΔP_2 .采用自适应的计算方法,当队列溢出时,随机丢包率增大 ΔP_1 ,当出现链路资源利用不足时,随机丢包率减少 ΔP_2 .由于采用自适应的丢包率估计方法,从而不可避免地出现振荡等问题.

此后出现了REM(random exponential marking)^[6],PI^[11],Adaptive PI^[13],AVQ(adaptive virtual queue)^[14],PIP^[15]算法等.REM 利用了网络流量优化理论中“价格”的概念来探测和控制网络的拥塞状态.在“价格”的演化算法中,REM 使用了简单的比例关系.单节点上分组的丢弃概率与“价格”之间是指数关系,REM 是梯度寻优的解.PI 中控制的是队列长度.Adaptive PI 通过实时测量链路的报文丢失率获得当前的负载信息,然后动态设置算法中的有关参数.AVQ 中控制的是队列的入口速率.文献[16]提出一种基于稳定裕度的PID控制算法,可使系统瞬态性能优于PI算法,但其参数整定方法相当复杂.

2 PID 控制器性能准则及 TCP/AQM 模型

2.1 PID 控制器性能准则

性能准则主要基于系统误差的测量,系统误差是系统理想输出和实际输出的差值.一般地,PID 控制器设计采用误差绝对值积分(integral of absolute error,简称 IAE)、误差平方积分(integral of square error,简称 ISE)、时间乘以误差平方积分(integral of time-weighted square error,简称 ITSE)、ITAE 等作为性能准则(目标函数),PID 控制器设计的目的就是使目标函数最小^[2].表 1 给出了 IAE,ISE,ITSE,ITAE 等性能准则公式.

Table 1 Performance estimation of PID controller

表 1 PID 控制器性能准则

Name of criterion	Formula
IAE	$IAE = \int_0^{\infty} e(t) dt$
ISE	$ISE = \int_0^{\infty} e(t)^2 dt$
ITSE	$ITSE = \int_0^{\infty} te(t)^2 dt$
ITAE	$ITAE = \int_0^{\infty} t e(t) dt$

由于 IAE 和 ISE 没有考虑时间因素,在得到相对较小超调的同时,需要较长的调节时间.ITSE 虽然考虑了时间因素,但其公式复杂并且费时.相比较而言,ITAE 考虑了时间因素,能够降低初始大误差对性能指标的影响,同时强调了过渡过程后期的误差对指标的影响,响应快、超调小,比 ITSE 简单、省时且具有较好的选择性,因此,ITAE 是一个比较理想的性能准则.

2.2 TCP/AQM 模型

Misra 等人基于流体流理论建立了 AQM 作用下 TCP 连接上拥塞窗口的动态模型^[17],它可以用下面的一组非线性微分方程来描述:

$$\begin{cases} \frac{dW(t)}{dt} = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))} p(t-R(t)) \\ \frac{dq(t)}{dt} = \frac{W(t)}{R(t)} N(t) - C \end{cases} \quad (1)$$

各参数物理意义为:W 表示 TCP 窗口大小,R 表示往返时延,C 表示链路容量,p 表示丢弃概率,q 表示队列长

度, N 表示激活的 TCP 连接数.

为了便于分析系统的稳定性, Hollot 等人用小信号理论在稳态工作点对其进行了局部线性化^[18], 得到了如图 1 所示的时滞二阶模型, 图中控制器即为具体的 AQM 算法. 其中, $K = \frac{(RC)^3}{4N^2}$, $T_1=R$, $T_2 = \frac{R^2C}{2N}$, 各参数物理意义同上.

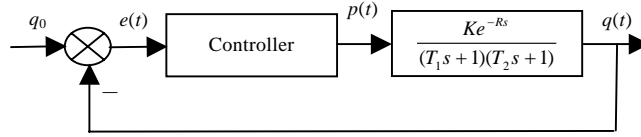


Fig.1 Block diagram of a linearized AQM control system

图 1 线性化后的 AQM 控制框图

3 DITAE-PID 算法设计

本节提出一种 DITAE-PID 控制器优化设计方法, 并把这种方法用于 AQM 控制器的设计, 期望在保持队列稳定的同时, 实现更高的链路利用率和更小的丢包率.

3.1 DITAE-PID 优化设计方法

根据被控对象模型的不同, PID 控制器的设计方法分为很多种. 文献[3]提出在 D 稳定域内根据极点分布来设计带有通用被控对象模型的 PID 控制器, 但控制器设计的目标函数是 PID 控制器 3 个设计参数(比例增益 K_p , 积分环节 K_i , 微分环节 K_d)的平方和, 该目标函数的设置过于简单, 容易陷入局部最优, 由第 2.1 节可知, ITAE 是一个比较理想的性能准则, 因此本文采用 ITAE 性能准则作为控制器设计的目标函数.

假定带有时滞的被控对象传递函数如下:

$$G_p(s) = \frac{N_p(s)}{D_p(s)} = \frac{a_0 + a_1s + \dots + a_{na-1}s^{na-1} + a_{na}s^{na}}{b_0 + b_1s + \dots + b_{nb-1}s^{nb-1} + s^{nb}} e^{-\tau s} \quad (2)$$

其中, $nb \geq na$, 系数 $a_i (i=0, 1, \dots, na), b_j (j=0, 1, \dots, nb-1)$ 都是实数, τ 是时滞. 该时滞可以用一阶 Pade 近似得到:

$$e^{-\tau s} = \frac{1 - \frac{\tau}{2}s}{1 + \frac{\tau}{2}s} = \frac{1 - Ts}{1 + Ts} \quad (3)$$

其中, $T=0.5\tau$, 把式(3)带入式(2)得:

$$G_p(s) = \frac{c_0 + c_1s + \dots + c_m s^m + c_{na+1}s^{na+1}}{d_0 + d_1s + \dots + d_n s^n + d_{nb+1}s^{nb+1}} = \frac{N_p(s)}{D_p(s)} \quad (4)$$

其中, $c_0=a_0, c_i=a_i - Ta_{i-1} (i=1, 2, \dots, na), c_{na+1}=-Ta_{na}, d_0=b_0, d_j=b_j + Tb_{j-1} (j=1, 2, \dots, nb-1), d_{nb}=1 + Tb_{nb-1}, d_{nb+1}=T$.

注意到高阶 Pade 近似可以提高近似精度, 得到的被控对象仍具有有理传递函数.

设 $G_c(s)$ 表示 PID 控制器的传递函数:

$$G_c(s) = \frac{N_c(s)}{D_c(s)} = \frac{K_d s^2 + K_p s + K_i}{s} \quad (5)$$

其中, K_p 为比例增益, K_i 为积分环节, K_d 为微分环节. 因此, 闭环系统的传递函数 $G(s)$ 为

$$G(s) = \frac{N(s)}{D(s)} = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \quad (6)$$

其特征多项式为

$$\begin{aligned} D(s) &= N_c(s)N_p(s) + D_c(s)D_p(s) = [c_0K_i + (c_0K_p + c_1K_i)s + (c_0K_d + c_1K_p + c_2K_i)s^2 + \dots + c_{na+1}K_d s^{na+3}] + [d_0s + d_1s^2 + \dots + d_{nb+1}s^{nb+2}] \\ &= c_0K_i + (c_0K_p + c_1K_i + d_0)s + \dots + (c_{na+1}K_d + d_{nb+1})s^{nb+2} \end{aligned} \quad (7)$$

给出 n 阶闭环系统的特征多项式的一般形式:

$$D(s, \mathbf{u}) = a_0(\mathbf{u}) + a_1(\mathbf{u})s + a_2(\mathbf{u})s^2 + \dots + a_n(\mathbf{u})s^n \quad (8)$$

其中,未知向量 $\mathbf{u} = [u_1, u_2, \dots, u_m]^T$ 表示控制器的 m 个设计参数.因此要找到合适的 \mathbf{u} 使得特征多项式(8)的 n 个根位于一组特定的 D 稳定域内或其边界上.

设式(8)的 n 个复数特征根(包括可能的实根)为

$$s_i = \sigma_i + j\omega_i, i=1, 2, \dots, n \quad (9)$$

为简化起见,设 $\Sigma_i = [\sigma_1, \sigma_2, \dots, \sigma_i], \Omega_i = [\omega_1, \omega_2, \dots, \omega_i], i=1, 2, \dots, n$.

因此,式(8)可以写成

$$D(s) = \prod_{i=1}^n (s - s_i) = [\alpha_0(\Sigma_n, \Omega_n) + j\beta_0(\Sigma_n, \Omega_n)] + [\alpha_1(\Sigma_n, \Omega_n) + j\beta_1(\Sigma_n, \Omega_n)]s + \dots + [\alpha_{n-1}(\Sigma_n, \Omega_n) + j\beta_{n-1}(\Sigma_n, \Omega_n)]s^{n-1} + s^n \quad (10)$$

其中, $\alpha_i(\Sigma_n, \Omega_n)$ 和 $\beta_i(\Sigma_n, \Omega_n) (i=0, 1, 2, \dots, n-1)$ 都是 $\Sigma_n = [\sigma_1, \sigma_2, \dots, \sigma_n]$ 和 $\Omega_n = [\omega_1, \omega_2, \dots, \omega_n]$ 的函数.

得到下面的递推公式:

当 $n=1$ 时,

$$\alpha_0(\Sigma_1, \Omega_1) = -\sigma_1, \beta_0(\Sigma_1, \Omega_1) = -\omega_1 \quad (11)$$

当 $n>1$ 时,

$$\begin{aligned} \alpha_0(\Sigma_n, \Omega_n) &= -\alpha_0(\Sigma_{n-1}, \Omega_{n-1})\sigma_n + \beta_0(\Sigma_{n-1}, \Omega_{n-1})\omega_n \\ \beta_0(\Sigma_n, \Omega_n) &= -\beta_0(\Sigma_{n-1}, \Omega_{n-1})\sigma_n - \alpha_0(\Sigma_{n-1}, \Omega_{n-1})\omega_n \\ \alpha_j(\Sigma_n, \Omega_n) &= \alpha_{j-1}(\Sigma_{n-1}, \Omega_{n-1}) - \alpha_j(\Sigma_{n-1}, \Omega_{n-1})\sigma_n + \beta_j(\Sigma_{n-1}, \Omega_{n-1})\omega_n, j=1, \dots, n-2 \\ \beta_j(\Sigma_n, \Omega_n) &= \beta_{j-1}(\Sigma_{n-1}, \Omega_{n-1}) - \beta_j(\Sigma_{n-1}, \Omega_{n-1})\sigma_n - \alpha_j(\Sigma_{n-1}, \Omega_{n-1})\omega_n, j=1, \dots, n-2 \\ \alpha_{n-1}(\Sigma_n, \Omega_n) &= \alpha_{n-2}(\Sigma_{n-1}, \Omega_{n-1}) - \sigma_n \\ \beta_{n-1}(\Sigma_n, \Omega_n) &= \beta_{n-2}(\Sigma_{n-1}, \Omega_{n-1}) - \omega_n \end{aligned} \quad (12)$$

又因为式(8)和式(10)具有相同的根,因此,

$$a_n(\mathbf{u})\alpha_j(\Sigma_n, \Omega_n) = a_j(\mathbf{u}), \beta_j(\Sigma_n, \Omega_n) = 0, j=0, 1, \dots, n-1 \quad (13)$$

假定期望的 D 稳定域为 $g(\sigma, \omega) \leq 0$, 则有

$$g(\sigma_i, \omega_i) \leq 0, i=1, 2, \dots, n \quad (14)$$

注意到不等式(14)包含所有的未知根 (Σ_n, Ω_n) , 引入一个向量设计参数 $\mathbf{x} = [\mathbf{u}, \Sigma_n, \Omega_n]$, 由式(13)和式(14)发现,如果能够找到一个设计向量,使得等式(13)和不等式(14)同时满足,则可以得到一个满足 D 稳定域要求的控制器.设定控制器设计的目标函数为 $J = \int_0^\infty t |e(t)| dt$, 因此,求控制器的设计参数就转化成一个带有约束的数值优化问题.

3.2 DITAE-PID主动队列管理控制器

这里将把新的 DITAE-PID 设计方法用于主动队列管理控制器设计.

由 $G_p(s) = \frac{Ke^{-Rs}}{(T_1s+1)(T_2s+1)}$ 及式(3)、式(5),得到闭环系统的特征多项式

$$D(s, \mathbf{u}) = T_1T_2R's^4 + (T_1T_2 + T_1R' + T_2R' - KK_dR')s^3 + (T_1 + T_2 + R' + KK_d - KK_pR')s^2 + (1 + KK_p - KK_iR')s + KK_i \quad (15)$$

其中, $R' = R/2$. 多项式(15)有 4 个根: $s_i = \sigma_i + j\omega_i, i=1, 2, 3, 4$. 根据式(12)和式(13), σ_i, ω_i 和 K_p, K_i, K_d 必须满足等式约束:

$$\begin{aligned} T_1T_2R'(\sigma_1\sigma_2\sigma_3\sigma_4 + \omega_1\omega_2\omega_3\omega_4 - \sigma_1\sigma_2\omega_3\omega_4 - \sigma_1\sigma_3\omega_2\omega_4 - \sigma_1\sigma_4\omega_2\omega_3 - \sigma_2\sigma_3\omega_1\omega_4 - \sigma_2\sigma_4\omega_1\omega_3 - \sigma_3\sigma_4\omega_1\omega_2) &= KK_i, \\ T_1T_2R'(\sigma_1\sigma_2\sigma_3 + \sigma_1\sigma_2\sigma_4 + \sigma_1\sigma_3\sigma_4 + \sigma_2\sigma_3\sigma_4 - \sigma_3\omega_1\omega_2 - \sigma_4\omega_1\omega_2 - \sigma_4\omega_1\omega_3 - \sigma_4\omega_2\omega_3 - \sigma_1\omega_2\omega_3 - \sigma_1\omega_2\omega_4 - \sigma_1\omega_3\omega_4 - \\ \sigma_2\omega_3\omega_4 - \sigma_2\omega_1\omega_3 - \sigma_2\omega_1\omega_4 - \sigma_3\omega_1\omega_4 - \sigma_3\omega_2\omega_4) &= KK_iR' - KK_p - 1, \\ T_1T_2R'(\sigma_1\sigma_2 + \sigma_1\sigma_3 + \sigma_1\sigma_4 + \sigma_2\sigma_3 + \sigma_2\sigma_4 + \sigma_3\sigma_4 - \omega_1\omega_2 - \omega_1\omega_3 - \omega_1\omega_4 - \omega_2\omega_3 - \omega_2\omega_4 - \omega_3\omega_4) &= T_1 + T_2 + R' + KK_d - KK_pR', \\ T_1T_2R'(\sigma_1 + \sigma_2 + \sigma_3 + \sigma_4) &= KK_dR' - T_1T_2 - T_1R' - T_2R', \\ \sigma_1\sigma_2\sigma_3\omega_4 + \sigma_1\sigma_2\sigma_4\omega_3 + \sigma_1\sigma_3\sigma_4\omega_2 - \sigma_1\omega_2\omega_3\omega_4 + \sigma_2\sigma_3\sigma_4\omega_1 - \sigma_2\omega_1\omega_3\omega_4 - \sigma_3\omega_1\omega_2\omega_4 - \sigma_4\omega_1\omega_2\omega_3 &= 0, \\ \sigma_1\sigma_2\omega_3 + \sigma_1\sigma_2\omega_4 + \sigma_1\sigma_3\omega_4 + \sigma_2\sigma_3\omega_4 - \omega_1\omega_2\omega_3 - \omega_1\omega_2\omega_4 - \omega_1\omega_3\omega_4 - \omega_2\omega_3\omega_4 + \sigma_1\sigma_3\omega_2 + \sigma_1\sigma_4\omega_2 + \sigma_1\sigma_4\omega_3 + \sigma_2\sigma_4\omega_3 &= 0 \end{aligned}$$

$$\begin{aligned} &\sigma_2\sigma_3\omega_1+\sigma_2\sigma_4\omega_1+\sigma_3\sigma_4\omega_1+\sigma_3\sigma_4\omega_2=0, \\ &\sigma_1\omega_2+\sigma_2\omega_1+\sigma_1\omega_3+\sigma_3\omega_1+\sigma_1\omega_4+\sigma_4\omega_1+\sigma_2\omega_4+\sigma_4\omega_2+\sigma_2\omega_3+\sigma_3\omega_2+\sigma_3\omega_4+\sigma_4\omega_3=0, \\ &\omega_1+\omega_2+\omega_3+\omega_4=0 \end{aligned} \tag{16}$$

现在设定 D 稳定域为抛物线:

$$D \cup \partial D: g(\sigma, \omega) = 4\sigma + \omega^2 + l \leq 0 \tag{17}$$

其中, l 的物理意义为:当 l 增大时,式(17)表示的边界曲线将沿着实轴向左移动,因此,根的允许区域减小。 $D \cup \partial D$ 域内的所有根都必须满足:

$$4\sigma_i + \omega_i^2 + l \leq 0, i=1,2,3,4 \tag{18}$$

由式(16)和式(18)可以得到 $x=[K_p, K_i, K_d, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \omega_1, \omega_2, \omega_3, \omega_4]^T$ 。

本文用 Matlab 和 Simulink 来运行优化算法, Simulink 框图如图 2 所示.假定瓶颈链路的容量为 15Mbps (3 750packets/s),激活的 TCP 连接数为 60,典型的 RTT 时间为 0.2s,由基于 ITAE 目标函数的带有约束的数值优化算法,得到控制器参数的值: $K_p=2.5835 \times 10^{-5}$, $K_i=5.7027 \times 10^{-6}$, $K_d=8.6729 \times 10^{-6}$ 。

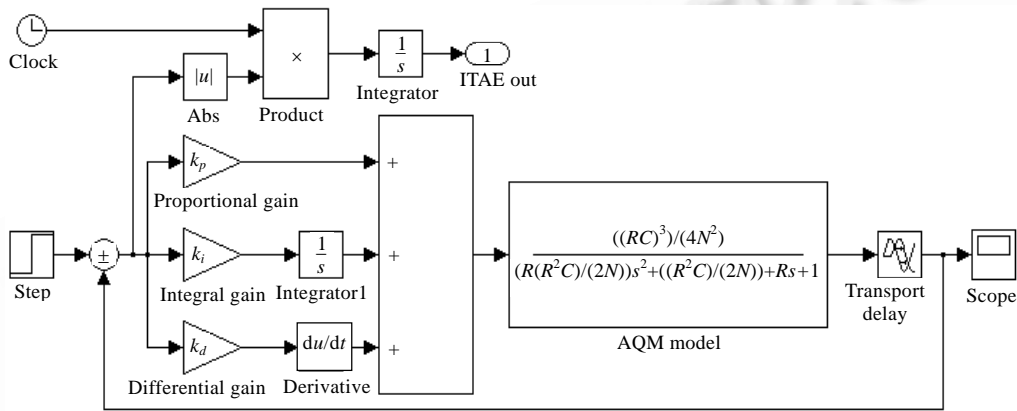


Fig.2 Simulink structure
图 2 Simulink 框图

在实际应用中,采样系统需要控制器的离散形式,以一系列的采样时刻点 kT 代表连续时间 t ,以和式代替积分,以增量代替微分,得到离散的 PID 表达式为

$$p(k) = K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d [e(k) - e(k-1)] \tag{19}$$

其中, $e(k) = q(k) - q_0$, $q(k)$ 是第 k 时刻的队列长度采样值, q_0 为期望队列长度.考虑到实际问题的物理意义,将分组丢弃概率定义为

$$p(k) = \begin{cases} 0, & c(k) < 0 \\ c(k), & 0 \leq c(k) \leq 1 \\ 1, & c(k) > 1 \end{cases} \tag{20}$$

其中, $c(k)$ 为控制器的输出.

将式(19)表示成增量形式:

$$\Delta p(k) = K_p \left\{ \left(1 + \frac{T}{T_i} + \frac{T_d}{T} \right) e(k) - \left(1 + \frac{2T_d}{T} \right) e(k-1) + \frac{T_d}{T} e(k-2) \right\} \tag{21}$$

其中, $T_i = \frac{K_p}{K_i}$, $T_d = \frac{K_d}{K_p}$, T 为采样时间.

为方便比较,这里仍采用文献[1]中给定的采样频率,即 160Hz,相应的采样时间 T 为 0.00625s.通过式(21)得

到控制器的增量形式.

4 性能评价

本文首先用 Matlab 比较采用 PI 和 DITAE-PID 控制器的系统阶跃响应性能指标,如图 3 所示.DITAE-PID 的超调量较小,调节时间较短,其综合性能指标优于 PI.

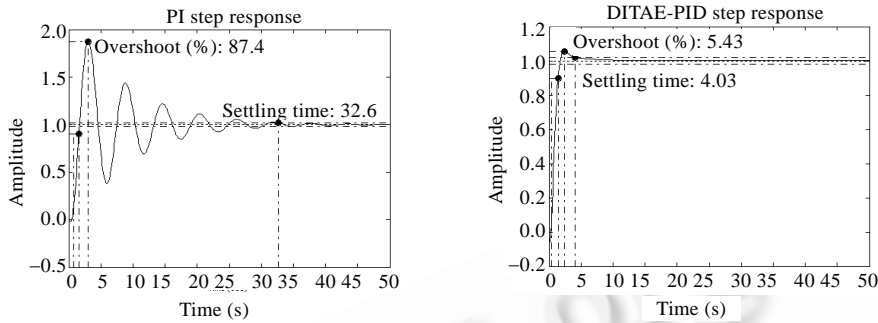


Fig.3 Step response for the PI and DITAE-PID

图 3 PI 和 DITAE-PID 阶跃响应

其次,在 NS-2^[19]平台上实现了 DITAE-PID 算法,用如图 4 所示的网络拓扑结构进行仿真实验.评价平均队列长度、队列抖动、丢包率、链路利用率等指标.

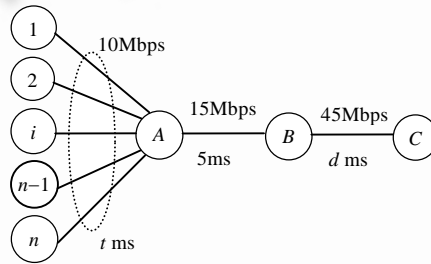


Fig.4 A simple network topology

图 4 网络拓扑结构

瓶颈链路位于节点 A 和节点 B 之间,链路容量 15Mbps(3 750packets/s,分组缺省大小为 500bytes),延时 5ms.所有业务源均为 FTP 业务源.它们与节点 A 之间的链路容量均为 10Mbps,延时 t ms,在默认情况下,t=5,即各业务源节点到 A 节点的延时为 5ms.除节点 A 的队列分别由 RED,PI,DITAE-PID 控制以外,其余队列均为 Drop Tail. RED 的高低门限值分别为 100packets 和 200packets,PI 和 DITAE-PID 的队列长度的期望值为 150packets.节点缓存大小均为 300packets,接收端窗口大小采用默认值 20packets.节点 B 和节点 C 之间的时延为 d ms.

4.1 实验1

取 n=60,节点 B 和节点 C 之间的时延 d=30ms,所有 FTP 业务源均在 0 时刻启动.仿真结果如表 2 和图 5 所示.表 2 中给出了平均队列长度、队列抖动、丢包率、链路利用率.图中横坐标表示时间,单位为秒;纵坐标表示队列长度,单位为包的个数.

实验结果表明:当延时为 30ms 时,RED 和 DITAE-PID 很快稳定到期望队列长度,而 PI 则需要较长时间才能使队列长度趋于期望值.

4.2 实验2

取 n=60,节点 B 和节点 C 之间的时延 d=30ms,FTP 业务源分为 3 组(前 40 个作为一组,后 20 个每 10 个作

为一组),分别在 0s,10s,20s 时刻启动.仿真结果如表 3 和图 6 所示.

Table 2 Performance of AQM algorithms

表 2 AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	142.2	202.6	176.8
Queue length standard deviation	87.6	145.0	116.4
Average packet loss rate (%)	3.59	2.88	2.96
Average link utilization (%)	97.9	97.7	98.6

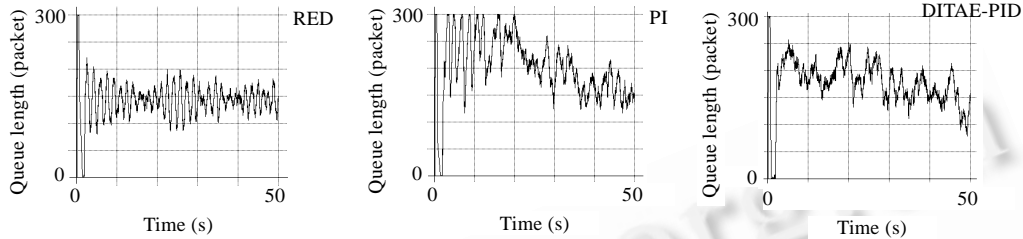


Fig.5 The instantaneous queue length for the RED, PI and DITAE-PID ($n=60, d=30ms$)

图 5 RED,PI,DITAE-PID 队列长度($n=60,d=30ms$,均在 0 时刻启动)

Table 3 Performance of AQM algorithms

表 3 AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	137.4	200.1	169.9
Queue length standard deviation	81.8	142.7	108.0
Average packet loss rate (%)	3.08	2.55	2.43
Average link utilization (%)	97.7	97.5	99.6

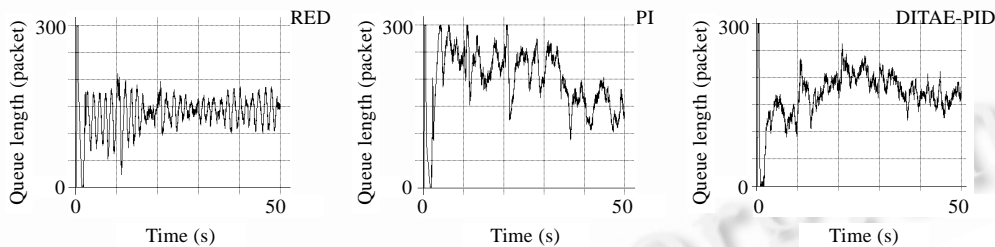


Fig.6 The instantaneous queue length for the RED, PI and DITAE-PID with bursty flows ($n=60, d=30 ms$)

图 6 RED,PI,DITAE-PID 队列长度($n=60,d=30ms$,3 组业务源分别在 0s,10s,20s 时刻启动)

实验结果表明:当延时为 30ms,且不在同一时刻启动时,对 RED,PI,DITAE-PID 都造成了一定程度的冲击. RED的振荡有所加剧,PI 则仍需较长时间才能使队列长度趋于期望值,而 DITAE-PID 所受影响最小,且能很快到达期望队列长度.

4.3 实验3

取 $n=60$,节点 B 和 C 之间的时延 $d=70ms$,所有 FTP 业务源均在 0 时刻启动.仿真结果如表 4 和图 7 所示.

Table 4 Performance of AQM algorithms

表 4 AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	110.7	174.5	139.2
Queue length standard deviation	87.4	122.4	85.0
Average packet loss rate (%)	2.54	2.07	2.06
Average link utilization (%)	97.0	96.8	97.4

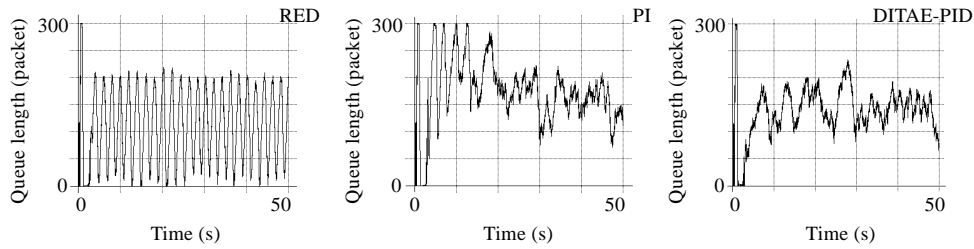


Fig.7 The instantaneous queue length for the RED, PI and DITAE-PID ($n=60, d=70\text{ms}$)

图 7 RED,PI,DITAE-PID 队列长度($n=60,d=70\text{ms}$,均在 0 时刻启动)

实验结果表明:当延时为 70ms 时,RED 的队列出现了大幅振荡.队列的大幅振荡一方面增加了端到端的时延抖动;同时,由于空队列出现的概率加大,导致链路利用率降低,这两点都违背了最初提出的 AQM 设计目标.而 PI 仍存在着需要较长时间才能使队列长度趋于期望值的问题.DITAE-PID 则体现了较好的性能,链路利用率较高、丢包率较小、队列抖动较小.

4.4 实验4

取 $n=60$,节点 B 和节点 C 之间的时延 $d=70\text{ms}$,FTP 业务源分为 3 组(前 40 个作为一组,后 20 个每 10 个作为一个组),分别在 0s,10s,20s 时刻启动.仿真结果如表 5 和图 8 所示.

Table 5 Performance of AQM algorithms

表 5 AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	102.1	173.5	96.8
Queue length standard deviation	82.4	120.3	61.9
Average packet loss rate (%)	2.11	2.25	0.01
Average link utilization (%)	95.2	96.3	98.7

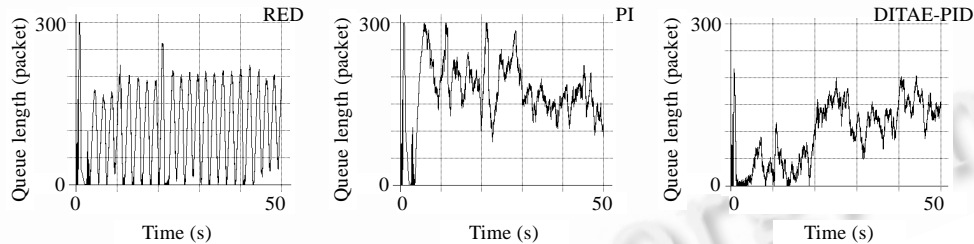


Fig.8 The instantaneous queue length for the RED, PI and DITAE-PID with bursty flows ($n=60, d=70\text{ms}$)

图 8 RED,PI,DITAE-PID 队列长度($n=60,d=70\text{ms}$,3 组业务源分别在 0s,10s,20s 时刻启动)

实验结果表明:当延时为 70ms,且不在同一时刻启动时,再次对 RED,PI,DITAE-PID 造成冲击.RED 的队列仍出现大幅振荡.而 PI 除到达期望队列长度时间较长以外,还出现丢包的情况.相比之下,DITAE-PID 队列长度能较快趋于期望值,且没有丢包现象发生.

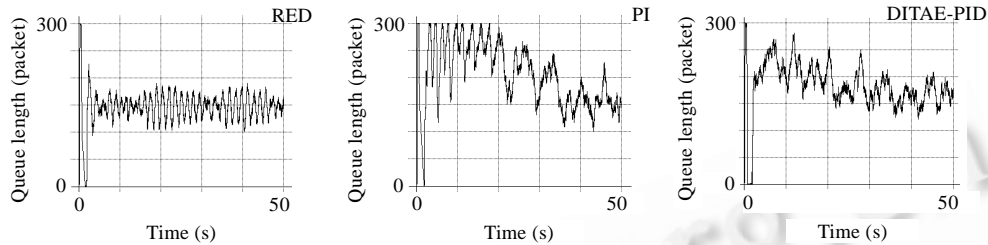
4.5 实验5

取 $n=60$,节点 B 和节点 C 之间的时延 $d=30\text{ms}$,各业务源节点到 A 节点的延时 t 分别为 $(2 \times i)\text{ms}$,即从 2ms~120ms.所有 FTP 业务源均在 0 时刻启动.仿真结果如表 6 和图 9 所示.

在实际网络中,各业务源的延时往往是不同的,因此,本实验改变各业务源到节点 A 的延时,第 i 条链路延时大小为 $(2 \times i)\text{ms}$,即从 2ms~120ms.由实验结果可以看出,由于延时的改变,各算法均受到影响.相比之下,DITAE-PID 已调整到稳态较快、抖动较小、空队列出现率和丢包率均较少,体现了更好的性能.

Table 6 Performance of AQM algorithms**表 6** AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	145.7	204.0	184.1
Queue length standard deviation	88.9	147.5	123.1
Average packet loss rate (%)	3.79	3.95	3.07
Average link utilization (%)	97.9	97.8	98.3

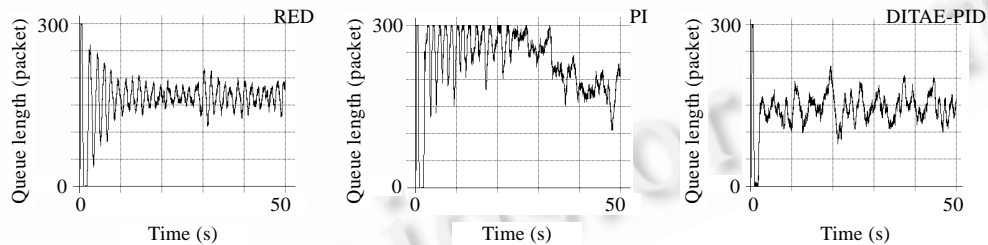
**Fig.9** The instantaneous queue length for the RED, PI and DITAE-PID ($n=60$, $d=30\text{ms}$, $t=(2\times i)\text{ms}$)**图 9** RED,PI,DITAE-PID 队列长度($n=60,d=30\text{ms}$,延时 t 为 $(2\times i)\text{ms}$,即从 $2\text{ms}\sim 120\text{ms}$)

4.6 实验6

取 $n=90$,节点 B 和节点 C 之间的时延 $d=30\text{ms}$,所有 FTP 业务源均在 0 时刻启动.仿真结果如表 7 和图 10 所示.

Table 7 Performance of AQM algorithms**表 7** AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	162.7	233.0	144.1
Queue length standard deviation	105.6	173.4	83.8
Average packet loss rate (%)	5.21	5.76	4.20
Average link utilization (%)	98.0	97.8	99.4

**Fig.10** The instantaneous queue length for the RED, PI and DITAE-PID ($n=90$, $d=30\text{ms}$)**图 10** RED,PI,DITAE-PID 队列长度($n=90,d=30\text{ms}$)

实验结果表明:当网络负载加大时,RED 的队列振荡幅度有所加剧.而 PI 到达期望队列长度的时间仍然较长,同时,丢包现象较为严重.相比之下,DITAE-PID 队列振荡较小,并且很快稳定到期望队列长度.

4.7 实验7

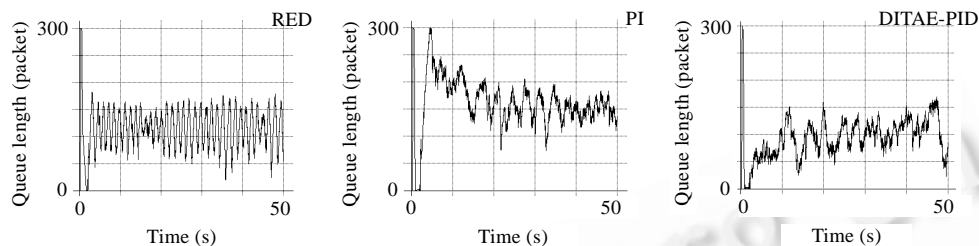
取 $n=30$,节点 B 和节点 C 之间的时延 $d=30\text{ms}$,所有 FTP 业务源均在 0 时刻启动.仿真结果如表 8 和图 11 所示.

实验结果表明:当网络负载减小时,RED 和 PI 的队列振荡幅度仍比 DITAE-PID 要大.PI 到达期望队列长度的时间仍然较长.DITAE-PID 队列振荡较小,并且保持了较高的链路利用率.

Table 8 Performance of AQM algorithms

表 8 AQM 算法性能比较

	RED	PI	DITAE-PID
Average queue length (packets)	114.3	162.0	94.5
Queue length standard deviation	60.3	101.4	44.8
Average packet loss rate (%)	1.46	1.09	1.01
Average link utilization (%)	97.5	97.6	98.3

Fig.11 The instantaneous queue length for the RED, PI and DITAE-PID ($n=30, d=30\text{ms}$)图 11 RED,PI,DITAE-PID 队列长度($n=30,d=30\text{ms}$)

综上所述,DITAE-PID 算法在轻度负载、中度负载、重度负载下一般时延或较大时延的网络、含有突发业务流的网络、含有不同业务源时延的网络等典型场景实验中综合性能更优。

5 结 论

网络拥塞控制算法设计的关键问题是如何生成反馈信息及如何对反馈信息进行响应.本文提出 DITAE-PID 优化设计方法用于 AQM 控制器的设计,介绍了其设计原理,通过基于 D 稳定域和 ITAE 性能准则的数值优化理论及方法得出了控制器参数.实验结果表明,新的 DITAE-PID 算法在几种典型场景下均体现了良好的综合性能:鲁棒性更好、链路利用率更高、丢包率更小、平均队列长度更趋于期望值,同时,趋于期望队列长度的时间更短,队列长度的抖动也 smaller.因此,DITAE-PID 算法整体表现优于具有代表性的 RED 和 PI 算法.

由于 NS-2 等网络模拟器并不能完全代替实际网络本身,因此,把 DITAE-PID 算法用于实际网络中的路由器设计,并进一步改进该算法是以后的研究内容之一;另一方面,本文的研究内容与软件控制论有密切关系.软件控制论旨在探讨软件与控制的原理性交叉应用^[20],如何在软件控制论中应用本文控制算法也是今后应考虑的一个问题.

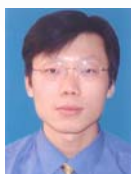
References:

- [1] Hollot CV, Misra V, Towsley D, Gong WB. On designing improved controllers for AQM routers supporting TCP flows. In: Proc. of the IEEE INFOCOM. Anchorage: IEEE Press, 2001. 1726–1734.
- [2] Hu SS. Principle of Automatic Control. 4th ed., Beijing: Science Press, 2005 (in Chinese).
- [3] Wang YJ, Schinkel M, Schmitt-Hartmann T, Hunt KJ. PID and PID-like controller design by pole assignment within D -stable regions. Asian Journal of Control, 2002,4(4):423–432.
- [4] Braden B, Clark D, Crowcroft J, et al. Recommendations on queue management and congestion avoidance in the Internet. RFC2309, 1998. <http://www.rfc.net/rfc2309.html>
- [5] Floyd S, Jacobson V. Random early detection gateways for congestion avoidance. IEEE/ACM Trans. on Networking, 1993,1(4): 397–413.
- [6] Athuraliya S, Li VH, Low SH, Yin QH. REM: Active queue management. IEEE Network, 2001,15(3):48–53.
- [7] Feng WC, Kandlur DD, Saha D, Shin KG. A self-configuring RED gateway. In: Proc. of the IEEE INFOCOM. New York: IEEE Press, 1999. 1320–1328.

- [8] Ott TJ, Lakshman TV, Wong LH. SRED: Stabilized RED. In: Proc. of the IEEE INFOCOM. New York: IEEE Press, 1999. 1346–1355.
- [9] Jiang M, Wu CM, Zhu ML. MRED: A new adaptive active queue management algorithm. Chinese Journal of Electronics, 2004, 13(2):228–232.
- [10] Liu S, Basar T, Srikant R. Exponential-RED: A stabilizing AQM scheme for low-and high-speed TCP protocols. IEEE/ACM Trans. on Networking, 2005,13(5):1068–1081.
- [11] Ji QJ, Dong YQ. A load-adaptive active queue management algorithm. Journal of Software, 2006,17(5):1140–1148 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/17/1140.htm>
- [12] Feng WC, Shin KG, Kandlur DD, Saha D. The BLUE active queue management algorithms. IEEE/ACM Trans. on Networking, 2002,10(4):513–528.
- [13] Lu XC, Zhang MJ, Zhu PD. An adaptive PI active queue management algorithm. Journal of Software, 2005,16(5):903–910 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/16/903.doc>
- [14] Kunniyur S, Srikant R. Analysis and design of an adaptive virtual queue (AVQ) algorithm for active queue management. In: Proc. of the ACM SIGCOMM. San Diego: ACM Press, 2001. 123–134.
- [15] Zhang HY, Liu BH, Dou WH. Design of a robust active queue management algorithm based on feedback compensation. In: Proc. of the ACM SIGCOMM. Karlsruhe: ACM Press, 2003. 277–285.
- [16] Fan YF, Ren FY, Lin C. Design a PID controller for active queue management. In: Proc. of the IEEE ISCC. Antalya: IEEE Press, 2003. 985–990.
- [17] Misra V, Gong WB, Towsley D. Fluid-Based analysis of a network of AQM routers supporting TCP flows with an application to RED. In: Proc. of the ACM SIGCOMM. Stockholm: ACM Press, 2000. 151–160.
- [18] Hollot CV, Misra V, Towsley D, Gong WB. A control theoretic analysis of RED. In: Proc. of the IEEE INFOCOM. Anchorage: IEEE Press, 2001. 1510–1519.
- [19] Fall K, Varadhan K. The ns Manual. The VINT Project: UC Berkeley, LBL, USC/ISI, and Xerox PARC. 2003.
- [20] Cai KY, Cangussu JW, DeCarlo RA, Mathur AP. An overview of software cybernetics. In: Proc. of the STEP. Chicago: IEEE Press, 2004. 77–86.

附中文参考文献:

- [2] 胡寿松.自动控制原理.第4版,北京:科学出版社,2005.
- [11] 纪其进,董永强.一种链路负载自适应的主动队列管理算法.软件学报,2006,17(5):1140–1148. <http://www.jos.org.cn/1000-9825/17/1140.htm>
- [13] 卢锡城,张明杰,朱培栋.自适应 PI 主动队列管理算法.软件学报,2005,16(5):903–910. <http://www.jos.org.cn/1000-9825/16/903.doc>



王秀丽(1977—),男,山东高唐人,博士生,主要研究领域为计算机网络,实时系统,可信计算.



周辉(1980—),男,博士生,CCF 学生会会员,主要研究领域为计算机网络,实时系统,可信计算.



王永吉(1962—),男,博士,研究员,博士生导师,主要研究领域为实时系统,网络优化,智能软件工程,优化理论,机器人,控制理论.



蔡开元(1965—),男,博士,博士生导师,主要研究领域为软件可靠性与测试,智能控制,软件控制论.