

基于改进网络模型的大时滞网络拥塞控制算法

王晓曦^{1,3}, 王永吉^{1,2}, 周津慧¹, 王秀丽^{1,3}

- (1. 中国科学院软件研究所互联网软件技术实验室, 北京 100080;
2. 中国科学院计算机科学重点实验室, 北京 100080;
3. 中国科学院研究生院, 北京 100039)

摘 要: 随着对网络拥塞控制的深入研究, 出现了许多有关网络流量的控制理论及网络模型. Misra 等人于 2000 年基于流体流 (fluid flow) 理论提出的模型被研究人员广为采用. 但模型在推导过程中的一些近似却使得模型对网络行为描述不精确. 本文对此从理论和实验结果两方面进行了详细分析, 并对原模型进行了改进. 基于改进的模型, 把一种 PID (Proportional Integral Differential) 及类 PID 设计方法用于 AQM (Active Queue Management) 控制器的设计. 对比仿真结果表明, 该算法具有更好的性能.

关键词: 网络模型; 拥塞控制; 主动队列管理; PID 控制; 大时滞; D 稳定域

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2005) 05-0842-05

Congestion Control Algorithm Based on Improved Model in Large-Delay Networks

WANG Xiao-xi^{1,3}, WANG Yong-ji^{1,2}, ZHOU Jin-hui¹, WANG Xiu-li^{1,3}

- (1. Laboratory for Internet Software Technologies, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China;
2. Key Laboratory for Computer Science, Chinese Academy of Sciences, Beijing 100080, China;
3. Graduate School, Chinese Academy of Sciences, Beijing 100039, China)

Abstract: With further research on the network congestion control, many flow control theories and network models have appeared. Based on fluid flow theory, Misra developed a dynamic model of the TCP congestion window in 2000, which has been accepted by many researchers. However, some flaws are inevitable due to its approximation in the model evolvement. This paper gives a detailed analysis of its shortcoming from both the theory and experimental results. Then it presents an improved model. Based on this model, a new PID (Proportional Integral Differential) and PID-like controller design method is applied to AQM controller. Experimental results show that this algorithm outperforms other algorithms.

Key words: network model; congestion control; active queue management; PID control; large delay; D stability

1 引言

网络拥塞避免与控制对保证 Internet 的稳定起着至关重要的作用, 是目前的重要研究课题^[1]. 在 Internet 发展初期, 拥塞控制主要通过 TCP 协议中基于滑动窗口的流量控制完成^[1]. TCP 的流量控制算法也经过了 Tahoe, Reno, New Reno, SACK 和 Vegas 等多个版本的改进. 随着技术的发展和应用的丰富, 人们把注意力转向路由器等网络中间节点设备, 期望通过增强它们的功能来实现主机端无法达到的目标^[2]. 由此出现了 RED (Random Early Detection)^[3]、PI (Proportional Integral)^[4]、REM (Random Exponential Marking)^[5]等主动队列管理算法.

网络拥塞控制属于自动控制和计算机科学两个学科的交叉点. 目前越来越多的控制论专家投身其中, 出现了许多关于

网络流量的控制理论及网络模型. V Misra 等人于 2000 年基于流体流 (fluid flow) 理论提出的网络模型较具影响力^[6]. 该模型建立了 AQM (Active Queue Management) 作用下 TCP 连接拥塞窗口的动态模型, 较为恰当的描述了 TCP 传输流的行为^[7], 为研究人员广为采用. 依据该模型, 提出了许多主动队列管理算法^[4,8,9]. 但该模型在推导中的一些化简也带来了问题.

2 简化的网络模型及其推导

在 2000 年, V Misra 等人提出了 AQM 作用下 TCP 连接拥塞窗口的动态模型. 该模型主要用下面一组非线性微分方程来描述 TCP 流的行为:

$$\begin{aligned} \dot{W}(t) &= \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))} p(t-R(t)) \\ \dot{q}(t) &= \frac{W(t)}{R(t)} N(t) - C \end{aligned} \quad (1)$$

收稿日期: 2004-05-10; 修回日期: 2005-01-16

基金项目: 中国科学院百人计划 (No. BCK35873); 国家自然科学基金 (No. 60373053); 中国科学院与英国皇家学会联合 (No. 20030389, No. 20032006); 国家 863 计划软件重大专项项目 (No. 2003AA1Z2220); 教育部留学回国人员科研启动基金 (No. 教外司留[2003]406 号)

式(1)中各参数物理意义为: 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. 各参数随时间 t 而变化.

其中,第一个微分方程描述的是 TCP 的 AIMD(Additive Increase Multiplicative Decrease) 策略;而第二个方程描述了路由器缓存空间的变化.

用 W_0 表示达到稳态时发送窗口的大小. 进一步考察在稳态情况下式(1)中各参量之间的关系,并在 $W \gg 01$ 时,对式(1)所表示的模型进行近似,可得图 1 所示框图^[7]:

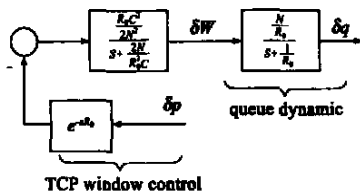


图 1 当 $W_0 \gg 1$ 时对 TCP 连接线性化后的框图

在上图中再加入 AQM 控制算法,最终可以得到如图 2 所示的时滞二阶模型. 其中 $K = \frac{(RC)^3}{4N^2}$, $T_1 = R$, $T_2 = \frac{R^2 C}{2N}$, 各参数物理意义同上. 图中控制器即为具体的 AQM 算法.

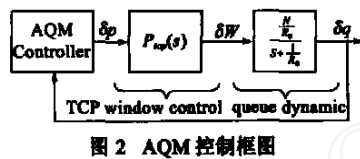


图 2 AQM 控制框图

3 基于简化模型的 PID 反馈控制策略及实验结果

根据图 2 所示模型,我们在文献[8]中,把一种新的 PID 及类 PID 设计方法用于 AQM 控制器的设计,控制闭环系统的理想动态性能. 研究表明,该算法在一般情况下具有更好的性能. 文献[8]详细说明了该算法的设计并进行了大量仿真实验.

然而,通过进一步研究,我们发现该算法在 RTT 往返时间明显增大时,性能有所降低. 在文献[8]的实验中,当往返传输延时增加到 300ms 时,再加上瓶颈节点对包缓存及其他节点处理所耗去的时间,最终的往返时延约在 400ms. 虽然文献[8]中提到,在控制器的参数设计中,针对的是 RTT 为 400ms 时的网络情况,但在上述网络情况下仿真结果如图 3 所示,实验表明,平均队列长度很短,无法提升到期望值 150packets. 这导致了空队列出现的几率大大增加,从而严重降低了链路利用率.

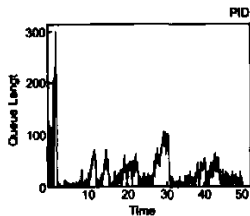


图 3 PID 队列长度 ($d=140ms$)

对大时滞网络下的算法性能进行研究是十分重要的. 现有的大多数策略和算法并没有充分考虑往返时延 (RTT) 对算法性能的影响. 在小时延的 LAN 或 MAN 上适应的算法未必能有效工作在大时滞的 WAN 网络环境下. 文献[10]中有关于 RTT 分布的详尽研究. 在跨洲越洋的 TCP 连接中, RTT 时间往往有上百毫秒. 因此,能够在大时滞网络条件下获得较好的执行效率是十分重要的.

4 对网络模型的改进

网络是一个非线性系统,具有复杂性和不确定性,对其建模具有一定的困难. PID 是一种具有反馈的控制策略,能够较好的根据系统状态做出相应的反应,但前提是系统的模型应该尽量准确.

前述模型在推导过程中进行了一些近似,这些近似都有可能影响最终结果. 其中,在由式(1)推导至图 1 所示模型的过程中,曾将 TCP 动态窗口中的 e^{-sR_0} 近似为 1,前提条件是 $W_0 \gg 1$,文献[7]在附录中对此进行了详细推导. 的确,当拥塞控制达到稳态时,拥塞窗口应该远远大于 1. 这正是研究人员在这里采用了近似的原因.

然而,如果在系统在达到稳态前,始终无法保证拥塞窗口远大于 1,这种近似就可能出现问题. 甚至在某些特定情况下,比如业务流很多或网络总的负载能力较差,这时即使达到稳态,对每个业务流来说,分配到的带宽也较小. 分配到的带宽变小将导致发送窗口减小. 即在上述情况下,即使系统达到稳态,也有可能无法保证拥塞窗口远大于 1,我们考察前述图 3 所示的仿真结果,发现系统的平均队列一直很小,在这种情况下,就无法保证拥塞窗口远大于 1 这个近似的前提条件.

基于以上考虑,我们不再做上述近似,而是直接对式(1)进一步推导,得到:

$$W = -\frac{1}{s} \left[\frac{N}{R_0^2 C} (1 + e^{-R_0 s}) W + \frac{R_0 C^2}{2N^2} e^{-R_0 s} p \right]$$

对上式进一步整理得到:

$$W = -\frac{\frac{R_0 C^2}{2N^2} e^{-sR_0}}{s + \frac{N}{R^2 C} (1 + e^{-sR_0})} p \quad (2)$$

由此,可以得到图 4 所示框图:

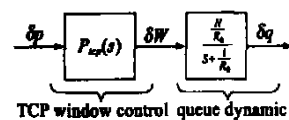


图 4 改进的线性化的 TCP 连接模型框图

其中:

$$P_{cp}(s) = -\frac{\frac{R_0 C^2}{2N^2} e^{-sR_0}}{s + \frac{N}{R^2 C} (1 + e^{-sR_0})} \quad (3)$$

在该图中加入 AQM 控制算法,得到图 5:

这种模型在推行过程中没有进行近似,因此可以更加准确的描述网络系统. 但是模型中引入了时滞环节,阶数增加,系统变得更加复杂.

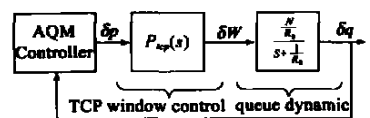


图 5 改进的 AQM 控制模型框图

在后面的章节中,我们应用文献[11]提出的一种控制器设计方法. 该方法的优越性在于被控对象可以为高阶稳定系

统或高阶不稳定系统;被控对象可以带有时滞;控制器设计参数可以超过三个.

5 基于改进模型的 PID 控制器设计方法

我们仍采用 PID 控制器为图 5 中的 AQM Controller. 采用文献[11]提出的控制器设计方法,通过调节特征根的分布来获取具有良好性能的系统控制器参数.

5.1 闭环系统的传递函数与特征多项式

对于图 5 所示系统,设 $G_c(s)$ 为 PID 控制器,其传递函数如下:

$$G_c(s) = \frac{N_c(s)}{D_c(s)} = \frac{K_d s^2 + K_p s + K_i}{s} \quad (4)$$

$G_p(s)$ 为图中的被控对象,其传递函数如下:

$$G_p(s) = P_{tcp}(s) * \frac{N}{R_0} / (S + \frac{1}{R_0}) \quad (5)$$

根据 Pade 级数, e^{-sR_0} 可表示为:

$$e^{-R_0 s} = \left(1 - \frac{R_0 s}{2} \right) / \left(1 + \frac{R_0 s}{2} \right) \quad (6)$$

由式(3)~(6)可得系统的特征多项式为:

$$D(s) = N_c(s) N_p(s) + D_c(s) D_p(s) = a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0 \quad (7)$$

其中:

$$a_4 = NR_0$$

$$a_3 = 3NR_0 + \frac{1}{2} R_0^2 C^2 K_d$$

$$a_2 = \frac{4N^2}{CR_0} + 2N + \frac{1}{2} R_0^2 C^2 K_p - R_0 C^2 K_d$$

$$a_1 = \frac{4N^2}{R_0^2 C} - R_0 C^2 K_p + \frac{1}{2} R_0^2 C^2 K_i$$

$$a_0 = -R_0 C^2 K_i$$

5.2 闭环系统的特征根与特征多项式系数间关系

设式(7)的 n 个复数特征根(包括可能的实根)为: $s_i = -\alpha_i + j\beta_i, i = 1, 2, \dots, n$

因此,系统的特征多项式 $D(s)$ 还可以表示为:

$$D(s) = \prod_{i=1}^n (s - s_i) \quad (8)$$

将式(8)展开并与式(7)比较,可得:

$$a_4(1^2 3^4 + 1^2 2^3 4^4 - 1^2 3^4 - 1^3 2^4 - 1^4 2^3 - 2^3 1^4 - 2^4 1^3 - 3^4 1^2) = a_0$$

$$a_4(1^2 2^3 + 1^2 4^4 + 1^3 3^4 + 2^3 4^4 - 3^4 1^2 - 4^4 1^2 - 4^4 1^3 - 4^4 2^3 - 1^2 4^4 - 1^3 4^4 - 2^3 4^4 - 2^4 1^3 - 2^4 1^4 - 3^4 2^4) = -a_1$$

$$a_4(1^2 + 1^3 + 1^4 + 2^3 + 2^4 + 3^4 - 1^2 - 1^3 - 1^4 - 2^3 - 2^4 - 3^4) = a_2$$

$$a_4(1 + 2 + 3 + 4) = -a_3$$

$$1^2 3^4 + 1^2 4^4 + 1^3 3^4 - 1^2 3^4 + 2^3 4^4 + 2^3 4^4 - 2^3 1^4 - 2^4 1^3 - 3^4 1^2 = 0$$

$$1^2 3^4 + 1^2 4^4 + 1^3 3^4 + 2^3 4^4 - 1^2 3^4 - 1^2 4^4 - 1^3 3^4 - 2^3 4^4 + 1^3 2^4 + 1^4 2^3 + 1^4 3^4 + 2^4 3^4 + 2^3 1^4 + 2^4 1^3 + 3^4 1^2 + 3^4 2^4 = 0$$

$$1^2 + 2^1 + 1^3 + 3^1 + 1^4 + 4^1 + 2^4 + 4^2 + 2^3 + 3^2 + 3^4 + 4^3 = 0$$
$$1 + 2 + 3 + 4 = 0 \quad (9)$$

本文中系统的特征方程阶数为 4. 对于高阶系统,文献[8,11]给出了根与特征多项式系数关系的递推公式,可供有兴趣的读者进行参考.

5.3 D 稳定域的设计

我们还需要设定一个 D 稳定域,使得当特征根位于 D 稳定域内或其边界上时,系统具有良好的性能. 本文采用的 D 稳定域为:

$$D = \partial D: g(\alpha, \beta) = 4 + \alpha^2 + e = 0 \quad (10)$$

其中 e 的物理意义为:当 e 增大,式(10)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.

$D = \partial D$ 域内的所有根都必须满足:

$$4 + \alpha_i^2 + e = 0, \quad i = 1, 2, 3, 4 \quad (11)$$

令向量 $x = [K_p, K_i, K_d, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \beta_1, \beta_2, \beta_3, \beta_4]^T$. 如果能够找到一个设计向量 x 使得等式(9)和不等式(11)同时满足,那么就可以得到一个满足 D 稳定域要求的控制器. 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. 关于数值优化的理论及方法可参见文献[12,13]. 有关 D 稳定域设计的更详尽理论请参考文献[11].

5.4 求解 PID 控制器参数

假定瓶颈链路的容量为 15Mbps,激活的 TCP 连接数为 60,典型的 RTT 时间为 0.4 秒,由带有约束的数值优化算法,得到的控制器参数 K_p, K_i, K_d 的值为:

$$K_p = 0.4166110697e - 5, \quad K_i = 0.1255578260e - 6, \\ K_d = 0.3716283686e - 5$$

至此,我们得到了具有时滞补偿能力的 AQM 控制器参数. 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:

$$p(k) = K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d [e(k) - e(k-1)] \quad (12)$$

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, $p(k)$ 为 k 时刻的丢包概率.

将式(12)表示成增量形式:

$$p(k) = K_p \left\{ \left[1 + \frac{T}{T_i} + \frac{T_d}{T} \right] e(k) - \left[1 + \frac{2T_d}{T} \right] a(k-1) + \frac{T_d}{T} e(k-2) \right\} \quad (13)$$

其中 $T_i = \frac{K_p}{K_i}, T_d = \frac{K_d}{K_p}, T$ 为采样时间. 这里采用文献[4,8]中给定的采样频率,即 160Hz,相应的采样时间 T 为 0.00625 秒.

由式(13),定义 $c(k) = c(k-1) + p(k)$ (14)

考虑到实际问题的物理意义,分组丢弃概率应为:

$$P(k) = \begin{cases} 0, & c(k) < 0 \\ c(k), & 0 \leq c(k) \leq 1 \\ 1, & c(k) > 1 \end{cases} \quad (15)$$

6 实验

我们运用 NS2 (Network Simulator 2)^[14,15] 网络模拟器验证算法性能. 首先在 NS2 平台上实现了该 PID 控制算法^[16], 然后采用如图 6 所示的网络拓扑结构进行仿真, 并将实验结果与 RED 和 PI 两种典型算法进行比较.

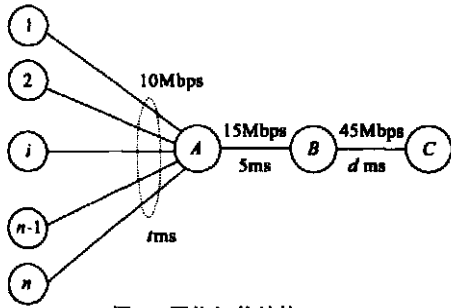


图 6 网络拓扑结构

瓶颈链路位于节点 A 和节点 B 之间, 链路容量 15Mbps, 延时 5ms. 有 n 个业务源, 均为持久性的 FTP 业务源. 它们与节点 A 之间的链路容量均为 10Mbps, 延时 t ms. 节点 B 和节点 C 之间的时延为 d 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 外, 其余各节点队列均为 Drop Tail. 节点 A 的队列将分别由 RED、PI、PID 控制. RED 的高低门限值分别为 100packets 和 200packets, PI 和 PID 的队列长度的期望值为 150packets; 各节点缓存大小均为 300packets.

在以下实验中, 我们将通过队列长度变化曲线图观察比较算法性能, 如平均队列长度、吞吐量、丢包率、链路利用率等. 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 1: 取 n 为 60, 节点 B 和节点 C 之间的时延 d 为 140ms, 所有 FTP 业务源均在 0 时刻启动.

在我们所设计的 PID 控制器参数的假定模型中, 瓶颈链路的容量为 15Mbps, 激活的 TCP 连接数为 60, 典型的 RTT 时间为 0.4 秒. 其中 RTT 时间主要包括传播时延、排队时延等. 当节点 B 到 C 时延为 140ms 时, 往返传播时延为 $(140 + 5 + 5) * 2 = 300$ ms. 排队时延主要在瓶颈节点路由器上, 其时延约为 $150 * 1k * 8 / 15M = 80$ ms. 再加上其他一些延时, 总的往返时延约在 0.4 秒左右. 因此, 我们的 PID 控制器参数即是根据实验 1 中的网络环境参数所设计. 实验仿真结果如图 7 所示.

从实验结果可以看出, RED 在大时滞中出现了巨幅震荡. PI 虽然最终可以稳定下来, 但是抖动厉害, 调整到稳定状态

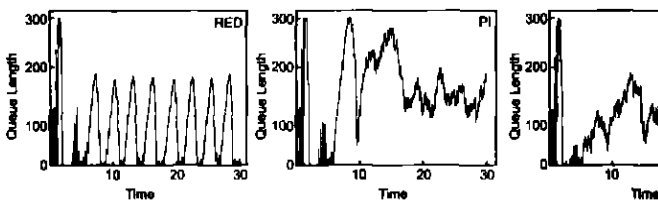


图 7 RED、PI、PID 队列长度 ($d = 140$ ms)

需要的时间较长, 丢包率较高. 相比之下, PID 算法体现了较好的性能.

实验 2: 取 n 为 60, 节点 B 和节点 C 之间的时延 d 为 80ms, 所有的 FTP 业务源均在 0 时刻启动. 各业务源节点到 A 节点的延时 t 分别为 $(2 * n)$ ms, 即从 2ms 到 120ms. 在实际网络中, 各业务源的延时往往是不同的. 本实验改变各业务源到节点 A 的延时, 分别从 2ms 到 120ms. 实验仿真结果如图 8 所示.

虽然各业务流的往返延时各自不同, 但本实验中平均往返延时仍在 400ms 左右. 由于延时的改变, 各算法均受到影响. 相比之下, PID 调整到稳态更快, 抖动小, 空队列出现率和丢包率均较少, 体现了更好的性能.

实验 3: 取 n 为 70, 节点 B 和节点 C 之间的时延 d 为 140ms, 有 60 个 FTP 业务源均在 0 时刻启动. 还有 10 个在 13s 时启动, 发送 100k 字节后停止.

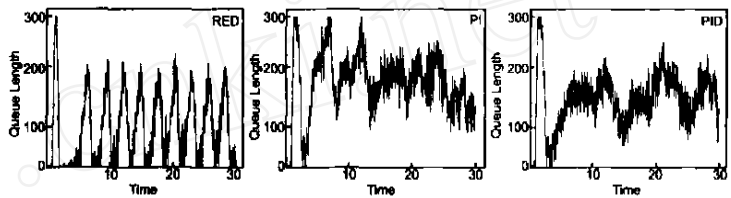


图 8 RED、PI、PID 队列长度 ($d = 80$ ms, t 从 2ms 到 120ms 变化)

实际网络中, 往往会有一些突发的小业务流, 被人们称为 burst. 本实验考查这些突发业务流的冲击对算法的影响. 仿真结果如图 9 所示.

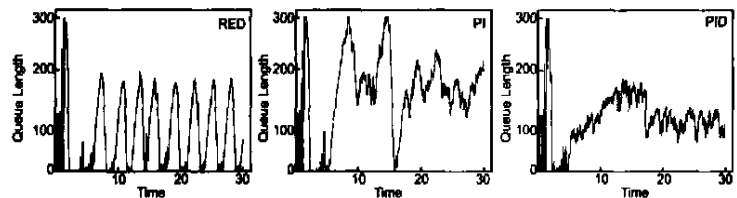


图 9 RED、PI、PID 队列长度 ($d = 140$ ms, 有 10 个节点具有突发业务流)

可以看到, 当引入突发业务流时, 各算法均受到影响, 队列长度有所上升. 而这些突发业务量终止时, 各算法队列也都有所下降. 相比之下, PID 体现了较强的抗干扰能力, 性能较好.

7 结论

本文对文献[6]所提出的网络模型进行了改进, 克服了原模型在一些情况下对网络行为描述不准确的缺陷, 并在此基础上把文献[11]提出的 PID 及类 PID 的控制器设计方法用于 AQM 控制器的设计中, 提出了一种改进的 PID 拥塞控制算法. 本文中, 我们采用 $4 + \frac{2}{e} + e$ 的抛物线区域作为 D 稳定域. 当 e 增大时, 抛物线将向复平面左侧平移, 特征根的选择区域减小. 根据控制理论, 当系统特征根位于该抛物线左侧时, 系统将获得较好的动、静态性能. 仿真实验表明, 新的 PID 算法克服了原有算法存在的缺陷, 在不同情况下均体现了良好的综合性能, 表现为: (1) 链路利用率更高, 平均队

列长度更趋于期望值;(2)趋于期望队列长度的时间更短,队列长度的抖动也更小;(3)具有较好的适应能力,当网络参数有所改变时仍能体现出良好的运行性能.其综合表现优于 RED、PI 等典型 AQM 算法.

参考文献:

- [1] Jacobson V. Congestion avoidance and control[J]. ACM computer communication Review ,1988 ,18(4) :314 ~ 329.
- [2] Braden B ,Braden B ,Clark D ,Crowcroft J ,Davie B ,Deering S. Recommendations on queue management and congestion avoidance in the Internet[S]. RFC2309 ,1998.
- [3] Floyd S ,Jacobson V. Random early detection gateways for congestion avoidance[J]. IEEE/ ACM Transactions on Networking ,1993 ,1(4) :397 - 413.
- [4] Hollot C ,Misra V ,Towsley D ,et al. On designing improved controllers for AQM routers supporting TCP flows[J]. IEEE Computer Society ,2001 :1726 - 1734.
- [5] Sanjeeva A ,Steven HL. REM: Active queue management [J]. IEEE Network ,2001 ,5(3) :48 - 53.
- [6] Misra V ,Gong W B ,Towsley D. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED[A]. Proceedings of the ACM/ SIGCOMM 2000[C]. Stockholm ,2001.
- [7] Hollot C V ,Misra V ,Towsley D ,et al. A control theoretic analysis of RED[A]. Proceedings of the INFOCOM 2000[C]. Tel Aviv ,2000.
- [8] Wang Y J ,Wang X L ,Zhou J H ,Wang X X. Congestion control of AQM routers by an improved PID controller[A]. Proceedings of the 4th International Symposium on Communication Systems , Networks and Digital Signal Processing (CSNDSP) [C]. UK:Newcastle ,2004. 322 - 325.
- [9] 任丰原 ,林闯 ,任勇 ,山秀明. 大时滞网络中的拥塞控制算法[J]. 软件学报 ,2003 ,14(3) :503 - 511.
- [10] Barakat C. TCP/ IP modeling and validation[J]. IEEE Network ,2001 ,15(3) :38 - 47.
- [11] Wang Y J ,Schinkel M ,Schmitt-Hartmann T ,Hunt K J. PID and PID-like controller design by pole assignment within D-stable regions[J]. Asian Journal of Control ,2002 ,4(4) :423 - 432.
- [12] Dennis J R J E ,Schanabel R B. Numerical Methods for Unconstrained Optimization and Nonlinear Equations[M]. Prentice-Hall ,Amsterdam ,1989.
- [13] Leva L ,Colombo A M. Method for optimizing set-point weights in ISA-PID autotuners[J]. IEE Proc Control Theory Appl. 1999 ,146(2) :137 - 146.
- [14] 王秀利 ,王永吉. 一种开放源代码的网络仿真器的原理与实现[J]. 计算机工程与应用 ,2004 ,40(15) :137 - 140.
- [15] UCN/LBL/ VINT. Network simulator-NS2 [DB/ OL]. www.mash. cs. berkeley. edu/ ns ,1995.
- [16] 王晓曦 ,王秀利 ,周津慧 ,王永吉. NS2 网络仿真器功能扩展原理及实现[J]. 小型微型计算机系统 ,2004 ,25(6) :1009 - 1014.

作者简介:



王晓曦 男,1979年2月出生于北京,硕士研究生,主要研究领域为计算机网络和实时系统.



王永吉 男,1962年12月出生于辽宁盖州,研究员,博士生导师,中科院2002年引进海外杰出人才“百人计划”入选者,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、控制理论等. E-mail : ywan @itechs. iscas. ac. cn.



周津慧 女,1957年12月出生于河北省张家口市,副研究员,主要研究领域为智能软件工程、软件过程技术与质量保证.



王秀利 男,1977年12月出生于山东高唐,博士研究生,主要研究领域为计算机网络、嵌入式实时系统.