

文章编号: 1001-0920(2013)10-1531-05

基于模型算法控制的主动队列管理算法

徐 琴, 孙金生

(南京理工大学 自动化学院, 南京 210094)

摘 要: 通过深入分析 TCP/AQM 系统的动态特性, 提出一个新的预测模型. 基于该模型, 结合模型算法控制 (MAC) 提出一种新的主动队列管理算法 (MACAQM), 并给出了 MACAQM 的详细设计过程和参数选取的原则. 大量不同网络环境的仿真实验表明了 MACAQM 算法的有效性. 与 PI, RaQ 和 REM 等算法相比较, MACAQM 具有收敛速度快、队列抖动小的优点. 同时, MACAQM 的采样间隔相对较大, 算法实现简单, 所以计算量较小, 占用的路由器资源也较少.

关键词: 网络拥塞控制; 主动队列管理; 预测模型; 模型算法控制

中图分类号: TP273

文献标志码: A

Active queue management algorithm based on model algorithmic control

XU Qin, SUN Jin-sheng

(School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China. Correspondent: SUN Jin-sheng, E-mail: jssun67@yahoo.com.cn)

Abstract: Through the analysis of the TCP/AQM system dynamics, a new predictive model is proposed. New active queue management (AQM) algorithm based on model algorithmic control (MACAQM) is proposed. The design details and the principles of choosing parameters are also presented. The results of simulations under different network environments show that MACAQM is effective in controlling the queue length to the desired value. In comparison to other AQM algorithms, such like PI, RaQ and REM, MACAQM has faster response and lower queue fluctuations. In the same time, MACAQM has larger interval with simple implementation, so it uses less resource on the routers.

Key words: network congestion; active queue management; predictive model; model algorithmic control

0 引 言

网络的爆炸式增长吞噬着网络的有限资源, 使网络的拥塞现象日益严峻. 为了缓解此现象, 网络拥塞控制成为人们研究的焦点. 网络拥塞控制算法根据实施的位置不同分为端算法和链路算法. 端算法是指在用户端实施的拥塞控制算法, 如 TCP Reno 等. 链路算法是指在中间节点 (如路由器) 实施的拥塞控制算法, 如 Droptail、RED^[1]、PI^[2]、REM^[3]、RaQ^[4]、DEM^[5]和 AVQ^[6]等.

Droptail 是目前网络中广泛使用的算法, 它只在缓冲队列满时丢弃到达的数据包, 以减缓网络的拥塞现象, 但这样可能造成全局同步, 使有效吞吐量较低. 为了避免该现象, Floyd 等提出了 RED 算法, 同时主动队列管理算法 (AQM) 的概念随之诞生. AQM 算法在队列未满时便主动地丢弃到达的数据包, 以通知用

户端网络的拥塞情况. 随着显式拥塞通知 (ECN)^[7]的提出, 包标记逐渐代替包丢弃成为 AQM 的主流执行方式. ECN 在缓冲队列没有溢出之前不丢弃数据包, 提高了网络的有效吞吐量, 同时也提高了用户对拥塞信息的反应速度.

由于预测控制对模型精度要求不高, 具有良好的鲁棒性, 对于模型不精确的网络拥塞控制而言, 预测控制是理想的选择. Wang 等^[8]将流体流模型线性化后看作反馈控制模型, 利用模型预测 (MP) 控制思想, 设计状态反馈观测器和反馈控制器. Marami 等^[9]利用模型预测控制 (MPC) 分析网络预测未来的输出并计算最优控制律. Bigdeli 等^[10]将流体流模型在平衡点附近线性化, 结合预测函数控制的思想, 分别设计预测模型、参考轨迹和自补偿、控制率 3 个部分, 提出了 PFC-AQM 算法. 但以上 AQM 算法的实现较为复

收稿日期: 2012-06-23; 修回日期: 2012-09-20.

基金项目: 国家自然科学基金项目(60974129, 70931002); 江苏省自然科学基金项目(BK2009388).

作者简介: 徐琴(1987-), 女, 博士生, 从事网络拥塞控制的研究; 孙金生(1967-), 男, 教授, 博士生导师, 从事网络拥塞控制、复杂产品质量控制等研究.

杂,占用的资源多,对变化的网络环境控制效果不理想.

为了更好地解决 TCP 网络中的拥塞控制问题,本文通过对 TCP/AQM 系统动态特性的深入分析,提出了一个新的预测模型,并在该模型的基础上,利用预测控制的思想进行设计,提出一种新的结合模型算法控制的 AQM 算法 (MACAQM). 该算法实现简单,占用的资源较少,同时能够快速有效地控制队列长度,使其稳定在期望值附近.

1 TCP/AQM 系统动态特性

1.1 AQM 拥塞控制机制分析

典型的 AQM 算法都会通过一定的方式计算合理的丢弃概率,以控制路由器上的队列长度. 丢弃概率的更新可以由特殊的事件触发,也可以是一个定时器定时更新. 例如 Blue 算法通过链路空闲和缓冲溢出的状况调整丢弃概率, AVQ 和 RED 在每一个新的数据包到达路由器时更新丢弃概率,它们均属于第 1 种情况;而大多数 AQM 算法是第 2 种情况,这也是当前 AQM 算法发展的主流,如大家熟知的 PI 和 REM 等.

以下着重分析用定时器更新丢弃概率的 AQM 算法的特性,为了简便,如果不特殊说明,则下文中所指的 AQM 算法均属于该类. 此类 AQM 算法有以下 3 个性质: 1) 路由器对每个数据包的处理只有两种结果,标记(如果不采用 ECN 则是丢弃,在不引起混淆的情况下不会特殊标明)或者不标记; 2) 在同一个采样间隔中丢弃概率是不变的; 3) 在每个间隔中数据包的处理是相互独立的. 由此可见,在同一个采样间隔内路由器上数据包的处理服从典型的二项分布.

路由器上队列长度(缓存空间占有情况)的变化较为关键,而路由器数据包到达的情况直接影响着路由器队列长度的变化,所以数据包到达的情况成为本文重点研究对象. 定义路由器上的数据包到达率(简称包到达率)为

$$\lambda(k) = m(k)/\delta. \quad (1)$$

其中: $k \geq 0$ 为第 k 个采样时刻, $m(k)$ 为第 k 个采样间隔内数据包到达的数量, δ 为采样间隔.

假设在第 k 个采样间隔丢弃概率为 $p(k)$, 处理的数据包数量为 $m(k)$, 不标记的一个数据包对应包到达率改变值为 $a(k)$, 标记的一个数据包平均对应包到达率改变值为 $d(k)$. 根据 AQM 算法服从二项分布的特性,该采样间隔包到达率变化的期望为

$$\Delta\bar{\lambda}(k+1) = a(k)(1-p(k))m(k) + d(k)p(k)m(k). \quad (2)$$

路由器将拥塞信息转发到用户端有延时,为了方便分

析,假设采样间隔与系统延时相同. 在第 k 个采样间隔作的处理将在第 $k+1$ 个采样间隔响应. 实际上,不同的 TCP 流有不同的数据包发送率和往返延时,路由器处理不同的 TCP 流的数据包对应着不同的 $a(k)$ 和 $d(k)$. 为了使分析更加合理,下面通过对 TCP 源端的分析,得出统计学上的 $a(k)$ 和 $d(k)$ 的期望值 $\bar{a}(k)$ 和 $\bar{d}(k)$.

1.2 TCP/AQM 系统的动态特性

TCP 拥塞控制通过滑动的拥塞窗口控制发送速率,主要涉及慢起动、拥塞避免、快速重传和快速恢复. 忽略慢起动, TCP 最重要的一个性质是拥塞窗口的变化符合加性增乘性减,在使用 ECN 的情况下,这意味着若每一个往返时延拥塞窗口增加 1,则收到一个标记的数据包拥塞窗口减少一半.

假设这样一个网络场景: 共有 N 条 TCP 连接,依次为 $TCP_1, TCP_2, \dots, TCP_N$, 其中 $TCP_i (i = 1, 2, \dots, N)$ 在第 k 个采样间隔中平均拥塞窗口大小为 $W_i(k)$, 平均往返时延为 $R_i(k)$, 定义数据包发送率为

$$V_i(k) = W_i(k)/R_i(k). \quad (3)$$

根据 TCP 拥塞控制策略的 AIMD 性质,如果 TCP_i 收到一个正确的 ACK 拥塞窗口增加 $1/W_i(k)$, 则数据包发送速率增加 $1/(V_i(k)R_i^2(k))$; 如果收到一个拥塞通知拥塞窗口减少 $W_i(k)/2$, 则数据包发送速率减少 $V_i(k)/2$.

结合第 1.1 节假设的路由器情况,设这 N 条 TCP 通过该路由器的所有 TCP 连接,当前路由器上处理的数据包来自 TCP_k , 则有

$$\lambda(k) = \sum_{i=1}^N V_i(k), \quad (4)$$

$$a(k) = 1/(V_k(k)R_k^2(k)), \quad (5)$$

$$d(k) = -V_k(k)/2. \quad (6)$$

注 1 由式 (1) 关于包到达率的定义可知,式 (4) 实际是一个近似值,并且忽略了 TCP 端到路由器的延迟,等式两端并不绝对相等,但不影响分析.

注 2 $a(k)$ 和 $d(k)$ 分别对应了 TCP 拥塞控制策略的加性增和乘性减部分,式 (5) 和 (6) 成立. 实际中处理的数据包需要经过系统延时才能达到源端,这里忽略该段延时.

$V_k(k)$ 和 $R_k^2(k)$ 对于路由器均是未知量,下面通过统计学分析求出 $V_k(k)$ 和 $1/(V_k(k)R_k^2(k))$ 的期望值,继而求出 $\bar{a}(k)$ 和 $\bar{d}(k)$. 有

$$\bar{V}_k(k) = \sum_{i=1}^N \frac{V_i(k)V_i(k)}{\lambda(k)} \geq \frac{\left(\sum_{i=1}^N V_i(k)\right)^2}{N\lambda(k)} = \frac{\lambda(k)}{N}, \quad (7)$$

$$\bar{a}(k) = \sum_{i=1}^N \frac{V_i(k)}{V_i(k)R_i^2(k)\lambda} = \sum_{i=1}^N \frac{1}{R_i^2(k)\lambda(k)}. \quad (8)$$

式(7)的等号只有在源端所有拥塞窗口的大小完全相同的情况下才成立, 假设

$$\bar{V}_k(k) = \rho\lambda(k)/N, \quad (9)$$

其中 $1 < \rho < N$. ρ 实际上是随时间变化的, 但是在足够长的时间内会在一个确定值附近变化, 为了分析方便, 本文不考虑 ρ 随时间的变化.

式(8)在每个 TCP 流的信息都被记录才能得到精确的结果, 为了节省计算量, 假设存在 R 满足

$$\sum_{i=1}^N \frac{1}{R_i^2(k)\lambda_i(k)} = \frac{N}{R^2\lambda(k)}. \quad (10)$$

在所有 TCP 连接的往返延时相同的网络中, R 的值与往返延时的值相同. 而实际网络中用户不断变化, R 在实际系统中也是随时间改变的. 为了分析方便, 本文不考虑 R 随时间的变化.

将式(5)~(10)代入(2), 有

$$\Delta\bar{\lambda}(k+1) = \frac{N}{R^2\lambda(k)}(1-p(k))m(k) - \frac{\rho\lambda(k)}{2N}p(k)m(k). \quad (11)$$

结合式(1), 将式(11)改写为

$$\Delta\bar{m}(k+1) = \frac{N\delta^2}{R^2}(1-p(k)) - \frac{\rho}{2N}p(k)m^2(k). \quad (12)$$

式(11)和(12)直观地表达了丢弃概率对将来包到达情况影响的预期, 即可以根据当前的网络状况和丢弃概率预测将来包的到达率.

2 MACAQM 算法

与大多数预测控制算法相同, 模型算法控制(MAC)主要由4部分组成: 预测模型、参考轨迹、反馈校正和滚动优化. 本文结合 MAC 算法, 利用前文得出的预测模型设计新的 AQM 算法, 详细设计过程如下.

2.1 预测模型

假设在第 k 个采样时刻, 路由器上队列长度为 $q(k)$, 采用的丢弃概率为 $p(k)$, 则在第 $k+1$ 个采样时刻队列长度为

$$q(k+1) = q(k) + m(k+1) - C\delta = q(k) + m(k) + \Delta m(k+1) - C\delta, \quad (13)$$

其中 C 是瓶颈链路容量. 回顾式(12), 有

$$\Delta\bar{m}(k+1) = \frac{N\delta^2}{R^2}(1-p(k)) - \frac{\rho}{2N}p(k)m^2(k). \quad (14)$$

令 $x_1 = N\delta^2/R^2$, $x_2 = \rho/2N$, 有

$$\Delta\bar{m}(k+1) = x_1(1-p(k)) - x_2p(k)m^2(k). \quad (15)$$

将式(15)带入(13), 得到第 $k+1$ 个采样时刻队列长度

的预测为

$$\hat{q}(k+1) = q(k) + m(k) - C\delta + x_1 - (x_1 + x_2m^2(k))p(k). \quad (16)$$

式(16)即为将要用到的采样模型, x_1 和 x_2 设计为模型参数. 在执行过程中, 由于不能预先测量当前采样时刻数据包到达的数量, $m(k)$ 是未知的, 但可以合理地预测为

$$\hat{m}(k) = m(k-1) + \Delta m(k) = m(k-1) + x_1(1-p(k-1)) - x_2p(k-1)m(k-1)^2. \quad (17)$$

其中: $m(k-1)$ 为上一个采样间隔的到达数据包的数量, $p(k-1)$ 为上一个采样间隔的丢弃概率.

2.2 反馈校正

不同的网络环境会有不同的模型参数, 为了增加算法的鲁棒性, 使其适应更多的网络环境, 本文采用在线校正基础模型的形式对模型进行线性校正. 校正形式如下:

$$\hat{q}_m(k+1) = \hat{q}_m(k) + h_0(q(k) - \hat{q}_m(k)). \quad (18)$$

式(16)为初始模型, 式(18)是算法在执行过程中使用的闭环预测模型.

2.3 参考轨迹

由于 MACAQM 采用单步预测的形式, 参考轨迹不需要设计得太复杂, 本文采用最简单的参考轨迹

$$q_r(k+1) = (1-\alpha)q(k) + \alpha q_{\text{ref}}. \quad (19)$$

其中: $0 < \alpha < 1$, q_{ref} 为期望队列长度.

2.4 滚动优化

在每一个采样时刻, MACAQM 的优化性能指标都是经过一个系统延时之后的时刻, 而不涉及更远的时间, 所以每个采样时刻都要重新计算优化性能指标, 得出合理的控制量. 考虑到反馈校正, 将闭环参考模型改写为

$$\hat{q}(k+1) = q(k) + m(k) - C\delta + x_1 + \beta(k) - (x_1 + x_2m^2(k))p(k), \quad (20)$$

其中 $\beta(k)$ 为模型的校正部分. 在执行过程中, 有如下形式:

$$\beta(k) = \beta(k-1) + h_0(q(k) - \hat{q}_m(k)). \quad (21)$$

由于采用的是单步预测控制, 即预测步长和控制步长都为 1, 可以直接根据期望的输出获得相应的控制量, 避免了复杂的性能指标的设计. 结合参考轨迹和式(20), 得到合理的控制量为

$$p(k) = \frac{q(k) - q_r(k+1) + m(k) - C\delta + x_1 + \beta(k)}{x_1 + x_2m^2(k)}. \quad (22)$$

2.5 参数选取原则

MACAQM算法需要设置的参数有: 采样间隔 δ , 控制参数 x_1 和 x_2 , 目标队列长度 q_{ref} , 反馈校正参数 h_0 , 参考轨迹参数 α . 由于预测控制要求采样间隔至少是一个系统延时, 采样间隔一般与往返时延相同, 略大或略小影响不大. 理论分析中定义 $x_1 = N\delta^2/R^2$, $x_2 = \rho/2N$, 参数选择在这个附近时的效果是最好的, 但由于算法有反馈校正部分, 没有严格要求. 反馈校正参数 h_0 决定了系统收敛的速度和队列抖动情况, 实际上, h_0 越大, 系统收敛越快, 队列抖动越剧烈, 一般情况下取值范围为 $0 \sim 0.5$. α 的取值范围为 $0 \sim 1$, 大部分情况下, α 取小于0.5的正数. 为了算法效果更好, 在MACAQM中可以让 α 随着队列长度与目标队列长度之差变化, 视具体情况而定.

3 仿真实验

为了表明MACAQM算法的有效性和优越性, 使用NS2网络仿真器进行不同网络环境下的仿真实验. 在仿真中, 瓶颈链路的队列长度是实验重点观察的量, 这是因为若队列长度稳定在期望值, 则意味着具有较高的网络吞吐量、可预测的往返延时和高链路利用率; 同时观察丢弃概率的变化来辅助表明算法的有效性.

使用的网络拓扑结构为哑铃状网络, 如不作特殊说明, 网络参数如下: 瓶颈链路的容量为45 Mb/s, 共500个TCP连接, 每个TCP连接的往返传输时延为0.1 s, TCP数据包的大小为1000 B, 缓冲队列容量为1125个数据包. TCP算法采用TCP Reno, 处理数据包的方式为包标记(ECN).

使用的MACAQM算法参数如下: 采样间隔为0.1 s, x_1 设置为500, x_2 设置为0.1, 目标队列长度 $q_{\text{ref}} = 500$, 反馈校正参数为 $h_0 = 0.3$, 参考轨迹的参数 α 为 $0 \sim 0.5$. 需要特殊说明的是, 为了使控制效果更好, α 根据队列长度与目标队列长度之差 Δq 变化, 具体规则如下: 1) 若 $\Delta q < 50$, 则 α 取0; 2) 若 $50 \leq \Delta q < 100$, 则 α 取0.1; 3) 若 $100 \leq \Delta q < 150$, 则 α 取0.2; 4) 若 $150 \leq \Delta q < 200$, 则 α 取0.3; 5) 若 $200 \leq \Delta q < 250$, 则 α 取0.4; 6) 若 $\Delta q \geq 250$, 则 α 取0.5. 后面仿真实验中的参数不变.

3.1 变化负载

在实际网络中, TCP连接数是时刻变化的. 这组实验将分别观察TCP连接数缓慢变化和剧烈变化下MACAQM的表现.

1) TCP连接数缓慢变化的仿真设置如下: 初始时有500个TCP连接, 从第30s开始, 每0.1s增加1个TCP连接, 共增加500个, 从第120s开始, 每0.1s减

少一个连接, 共减少500个, 实验结果如图1所示. 由图1可见, 随着TCP连接数的缓慢增加或减少, 丢弃概率也缓慢增大或减小, 但是队列长度稳定在期望值附近. 这说明即使TCP连接数缓慢变化, MACAQM算法仍然是有效的.

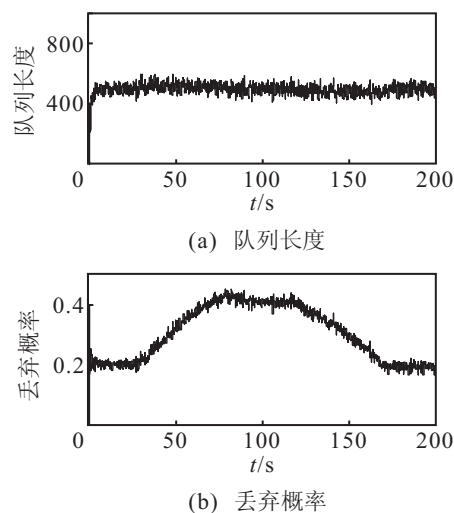


图1 TCP数目缓慢变化时队列长度和丢弃概率的变化

2) TCP连接数剧烈变化的仿真设置如下: 初始时有500个TCP连接, 第30s, 50s, 70s时突然增加100个TCP连接, 第120s, 140s, 160s时突然减少100个TCP连接. 实验结果如图2所示. 由图2可见, 若TCP连接数突然增加, 则丢弃概率随着突然增大, 经过一定时间后稳定, 队列长度在一段时期后也稳定在期望值; 如果TCP连接数突然减少, 则丢弃概率随着突然减小, 队列长度在较短的时间后也稳定在期望值. 但是由于TCP连接数突然变化, 队列在该时刻会有抖动, 特别是在TCP连接数突然增加时, 队列长度抖动较为剧烈, 这是MACAQM算法今后需要改进的地方. 可以看出, 即使TCP连接数剧烈变化, MACAQM算法仍然是有效的.

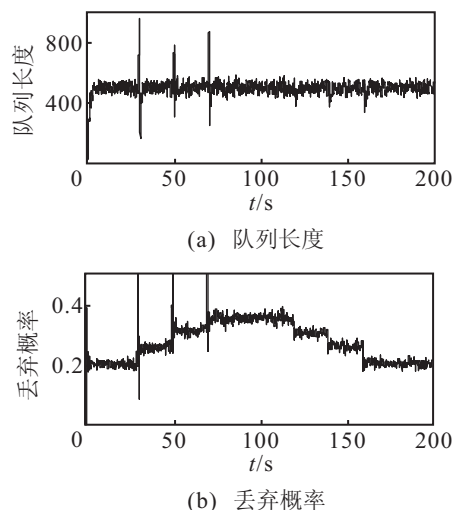


图2 TCP数目剧烈变化时队列长度和丢弃概率的变化

3.2 多瓶颈链路网络

为了观察 MACAQM 在多瓶颈链路下的控制效果, 给出多瓶颈链路网络的实验结果. 设置多瓶颈链路网络结构如图 3 所示, 其中 R_1 和 R_2 、 R_3 和 R_4 之间的链路为瓶颈链路. 共有 300 个 TCP 连接的源端连接在 R_1 上, 其中各有 100 个 TCP 连接的目的端分别连接在 R_2 、 R_3 和 R_4 上; 共有 200 个 TCP 连接的源端连接在 R_2 上, 其中各有 100 个 TCP 连接的目的端分别连接在 R_3 和 R_4 上; 有 100 个 TCP 连接由 R_3 到达 R_4 . 实验结果见图 4. 可以看出, 在多瓶颈链路网络中, MACAQM 也可以有效地控制队列长度, 使其稳定在设定的期望值附近.

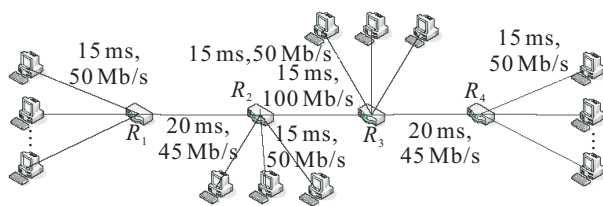
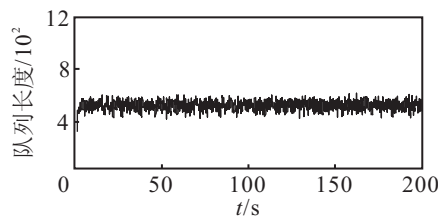
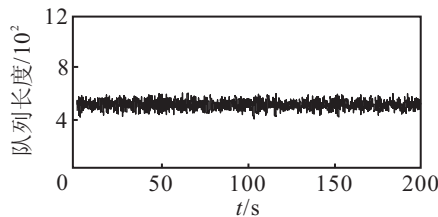


图 3 多瓶颈链路网络结构



(a) R_1 与 R_2 之间的瓶颈链



(b) R_3 与 R_4 之间的瓶颈链

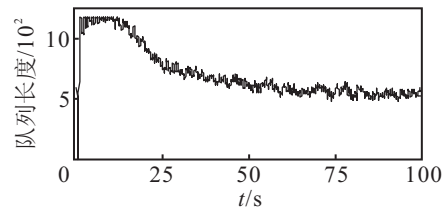
图 4 复杂网络队列长度变化

3.3 与其他算法比较

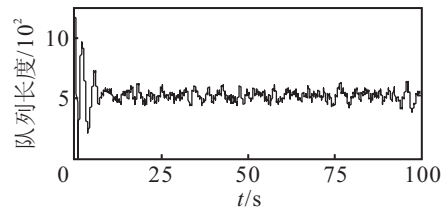
将 MACAQM 与 PI、REM 和 RaQ 等算法进行比较, 网络结构为哑铃状网络, MACAQM 采用前面描述的默认设置. 为了使实验结果更加明显, 仿真时间缩短为 100 s. PI 和 REM 使用 NS2 中的默认参数, RaQ 使用的参数如下: $q_{kp} = 0.0077$, $q_{ki} = 0.0005$, $r_{kp} = 0.0095$, 采用间隔为 0.005 s. 注意到, MACAQM 使用的采样间隔为 0.1 s, 远远大于其他算法的采样间隔, 这意味着大大减少了路由器的计算量, 节省了资源. 如果其他算法的采样间隔设为 0.1 s, 则队列长度需要更长的时间稳定, 队列抖动更大.

实验结果如图 5 所示. 由图 5 可见, PI 和 REM 的收敛速度最慢; RaQ 算法相对快一些, 但稳定之前经

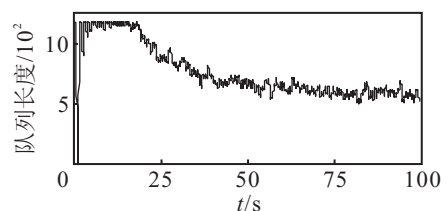
过了几次大幅度的震荡; MACAQM 是其中收敛速度最快、队列抖动最小的算法.



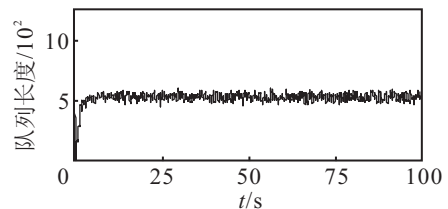
(a) PI



(b) RaQ



(c) REM



(d) MACAQM

图 5 不同算法比较结果

4 结 论

本文通过对 TCP/AQM 系统的动态性进行分析, 提出了新的预测模型, 该模型能够根据当前的网络情况有效地预测缓冲队列长度的变化. 本文利用该模型, 结合预测控制的思想, 提出了一个有效的主动队列管理算法 MACAQM. 该算法具有较好的鲁棒性, 与传统算法相比, 占用的资源更少, 并且有更快的收敛速度和更小的队列抖动.

参考文献(References)

- [1] Floyd S, Jacobson V. Random early detection gateways for congestion avoidance[J]. IEEE/ACM Trans on Networking, 1993, 1(4): 397-413.
- [2] Hollot C, Misra V, Towsley D, et al. On designing improved controllers for AQM routers supporting TCP flows[C]. Proc of 20th Annual Joint Conf of the IEEE Computer and Communications Societies. Alaska: IEEE, 2001, 3: 1726-1734.