

基于再励学习的主动队列管理算法*

张雁冰⁺, 杭大明, 马正新, 曹志刚

(清华大学 电子工程系 微波与数字通信国家重点实验室,北京 100084)

A Robust Active Queue Management Algorithm Based on Reinforcement Learning

ZHANG Yan-Bing⁺, HANG Da-Ming, MA Zheng-Xin, CAO Zhi-Gang

(State Key Laboratory on Microwave and Digital Communications, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

+ Corresponding author: Phn: +86-10-62782690, E-mail: ybzhang01@mails.tsinghua.edu.cn, <http://www.tsinghua.edu.cn>

Received 2003-07-23; Accepted 2003-10-14

Zhang YB, Hang DM, Ma ZX, Cao ZG. A robust active queue management algorithm based on reinforcement learning. *Journal of Software*, 2004,15(7):1090~1098.

<http://www.jos.org.cn/1000-9825/15/1090.htm>

Abstract: From the viewpoint of decision theory, AQM (active queue management) can be considered as an optimal decision problem. In this paper, a new AQM scheme, Reinforcement Learning Gradient-Descent (RLGD), is described based on the optimal decision theory of reinforcement learning. Aiming to maximize the throughput and stabilize the queue length, RLGD adjusts the update step adaptively, without the demand of knowing the rate adjustment scheme of the source sender. Simulation demonstrates that RLGD can lead to the convergence of the queue length to the desired value quickly and maintain the oscillation small. The results also show that the RLGD scheme is very robust to disturbance under various network conditions and outperforms the traditional REM and PI controllers significantly.

Key words: congestion control; active queue management; reinforcement learning

摘要: 从最优决策的角度出发,将人工智能中的再励学习方法引入主动队列管理的研究中,提出了一种基于再励学习的主动队列管理算法 RLGD(reinforcement learning gradient-descent).RLGD 以速率匹配和队列稳定为优化目标,根据网络状态自适应地调节更新步长,使得队列长度能够很快收敛到目标值,并且抖动很小.此外,RLGD 不需要知道源端的速率调整算法,因而具有很好的可扩展性.通过不同网络环境下的仿真显示,RLGD 与 REM,PI 等 AQM 算法相比,具有更好的性能和鲁棒性.

关键词: 拥塞控制;主动队列管理;再励学习

中图法分类号: TP393 **文献标识码:** A

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2001AA121062 (国家高技术研究发展计划(863))

作者简介: 张雁冰(1978—),男,甘肃兰州人,硕士生,主要研究领域为通信网络最优控制;杭大明(1977—),男,博士生,主要研究领域为多媒体网络,卫星通信;马正新(1969—),男,副教授,主要研究领域为宽带接入网,IP QoS,星地一体网;曹志刚(1939—),男,教授,博士生导师,主要研究领域为卫星通信,移动通信,数字信号处理.

端到端的拥塞控制机制作为 QoS 保证的主要手段,一直是网络研究的热点.目前应用最为广泛的 TCP 协议是一种反馈式的拥塞控制机制,即源端根据检测到的网络拥塞状况来调整发送窗口大小,从而达到拥塞避免的目的^[1].TCP 协议虽然取得了巨大的成功,但也逐渐暴露出不少问题.比如,在一个 TCP 网络中,如果路由器采用通常的尾部丢弃(drop-tail)策略,可能造成多个 TCP 源的同步,即它们同时增大或减小发送窗口,引起路由器队列振荡,端到端的延时无法保证.因此,人们意识到仅靠源端的流量控制机制很难提供可靠的 QoS 保证,链路上的路由器必须要提供一定的支持,这就是主动队列管理(active queue management,简称 AQM).

一般认为,AQM 有如下 3 个目标:(1) 维持队列长度在一个较小的目标值附近,且尽量稳定.目标值的选取应是吞吐量和延时的折衷(较短的队列长度可以减小延时,但丢弃率相应增大,即吞吐量降低).稳定的队列长度可以有效地消除延时抖动(delay jitter);(2) 具有较强的鲁棒性,即对环境变化不敏感;(3) 保证一定的公平性.Floyd 等人提出的随机早期检测(random early detection,简称 RED)^[2]是最早的一种 AQM 算法.它不是等队列满时才丢弃分组,而是在队列长度达到一定值时就按一定概率来丢弃.丢弃概率随队列长度的增加而增大.RED 试图用一种“提前通知”的手段来避免网络进入拥塞的状态,从而提高网络的性能.虽然研究表明 RED 算法的性能对参数十分敏感,但在某些情况下仍然会造成队列振荡^[3],研究者也提出了很多 RED 的改进算法,如 SRED,FRED,BLUE 等^[4-9],但通过以一定概率丢弃数据包来“提前通知”源端改变发送速率的方法成为以后几乎所有 AQM 算法的核心思想.不同的 AQM 算法主要区别就在于用不同的方法来估计网络的拥塞状态,并计算随机丢包的概率.

目前 AQM 主要通过两种参数来估计网络的拥塞状态:(平均)队列长度和(平均)包到达率.比如,RED 及其改进算法都是根据平均队列长度来计算丢包概率的.而 AQ,VAQ^[13,15]等算法则是利用到达率来更新一个虚拟链路(virtual link)的带宽,从而决定丢包概率.Sanjeewa 等人提出的 REM(random exponential marking)^[14]算法由于同时采用了队列长度和到达率这两种参数来计算丢包概率,因而取得了更好的效果.早期的 AQM 主要以仿真和实验为主要研究手段,丢包率的计算方法和参数的选择都缺乏理论依据.自从 Vishal Misra 等人在文献[12]中提出了 TCP 的非线性模型以后,经典的控制理论成为分析、设计 AQM 控制器的主要方法.比如,Hollot 等人用经典的线性控制理论分析了 RED 的稳定性^[10],并为 AQM 设计了比例积分(proportional-integrator,简称 PI)控制器^[11].仿真显示,PI 与 RED 相比的确增强了对队列的控制能力,但其参数严重依赖于网络环境,比如连接数目、往返时间(RTT)大小等.这使得 PI 控制器不能适应动态变化的网络环境,鲁棒性较差.

REM 算法通过到达率和队列长度两个参数来调节丢包概率,虽然取得了较好的效果,但由于调整步长固定,使得队列长度在目标值附近仍然有较剧烈的振荡.

再励学习(reinforcement learning,简称 RL)是人工智能领域中的一种最优决策算法.由于它无须知道系统模型,可以只通过反馈信号(称为报酬)来进行强化学习,并随着系统的运行收敛到最优,因而具有广泛的用途.目前已有不少研究者将 RL 方法用于解决通信网络中的接纳控制^[19]、无线网络中的动态信道分配^[20]等问题.

本文将 AQM 归纳为一个最优决策问题,采用再励学习的思想,结合梯度下降法提出了一种新的 AQM 算法 RLGD(reinforcement learning gradient-descent).RLGD 以链路速率匹配和队列长度稳定为优化目标,通过自适应地调整丢包概率的更新步长和方向,使队列长度能够快速收敛于目标值,并且抖动很小.由于 RL 是一种与模型无关的学习方法,因此 RLGD 不必关心源端的速率调整方法,只要求有一定的反馈控制机制即可,这使得 RLGD 具有很强的可扩展性.仿真显示,RLGD 比 PI 和 REM 控制器具有更好的性能和鲁棒性.

1 基于再励学习的 AQM 算法

1.1 再励学习

再励学习是人工智能领域中的一种无监督学习算法,它通过反馈的报酬值来指导策略的改进,最终收敛于最优.在再励学习中,进行学习并执行策略的实体称为 Agent,Agent 以外的所有实体集合称为 Environment.再励学习就是 Agent 和 Environment 的一个不断交互的过程.图 1 给出了这一过程的图示.

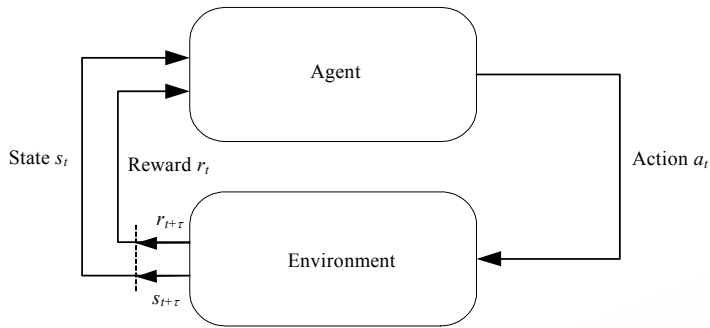


Fig.1 Reinforcement learning
图 1 再励学习

在每一时刻 t , Agent 观察得到 Environment 当前的状态 $s_t \in \mathcal{S}$, 这里 \mathcal{S} 是所有可能的状态集合, 在此基础上, Agent 选择执行动作 $a_t \in A(s_t)$, 这里 $A(s_t)$ 是在状态 $s_t \in \mathcal{S}$ 下可行的动作集合. 在下一时刻 $t+1$, Environment 反馈给 Agent 一个报酬值 $r_{t+1} \in \mathcal{R}$, 并进入下一状态 s_{t+1} . 可见, 在每一时刻, Agent 完成一个从状态到动作的映射, 我们把这一映射称为“策略”, 记作 π_t . $\pi_t(s, a)$ 表示状态 $s_t = s$ 时, 动作 $a_t = a$ 的概率. 再励学习指出了 Agent 根据与 Environment 的交互过程来改进策略的方法, 其目标是取得长期总报酬的最大化, 即 $\text{Maximize} \sum_{t=0}^{\infty} r_t$.

再励学习提供了一个解决最优化问题的框架. 它不关心 Environment 的内部细节, 也不考虑报酬的具体形式, 寻求最优解的方法被归纳为动作、状态和报酬 3 个量在 Agent 和 Environment 之间传递的过程. 这种简单、抽象但有效的框架为再励学习打开了广阔的应用空间.

1.2 RLGD算法

文献[16]研究了 TCP/AQM 拥塞控制模型, 给出了如下的平均队列长度 \bar{q} 和链路利用率 u 与丢包概率 p 之间的关系:

$$\bar{q}(p) = \begin{cases} \text{Max}(B, c(T_R^{-1}(p, c/n) - R_0)), & p \leq p_0 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$u(p) = \begin{cases} 1, & p \leq p_0 \\ \frac{T(p, R_0)}{c/r_0}, & \text{otherwise} \end{cases} \quad (2)$$

其中 $T(p, R_0)$ 是 TCP 源的吞吐率, 它取决于丢包率 p 和往返时间 R_0 , B 是缓冲区大小, c 是链路带宽, n 是 TCP 源数目. 从上式可以看出, 队列长度和链路利用率与丢包率的关系十分复杂, 一般的梯度下降法在这里不能直接应用. 因此, 我们利用再励学习结合梯度下降法来求解.

设 t 时刻系统的状态可以表示为矢量 \mathbf{S}_t , $t+1$ 时刻获得的报酬为 r_{t+1} , $Q_t = E \left\{ \sum_{k=0}^{\infty} r_{t+k+1} \mid \mathbf{S}_t = \mathbf{S} \right\}$ 表示 t 时刻系统状态为 \mathbf{S} 时, 此后获得总报酬的期望值, 称为状态值函数(state-value function). 显然, 为取得长期总报酬的最大化, 应选择使 Q_t 最大的动作 a_t . 由于直接的状态值函数很难求, 所以一般用 \mathbf{S}_t 的函数 $Q(\boldsymbol{\theta}, \mathbf{S}_t)$ 来逼近 Q_t . 其中 $\boldsymbol{\theta}$ 是矢量参数, 近似函数 $Q(\boldsymbol{\theta}, \mathbf{S}_t)$ 的形式可以根据具体问题选取, 可以是各种线性或非线性函数形式, 甚至是人工神经网络. 在给出 $Q(\boldsymbol{\theta}, \mathbf{S}_t)$ 的形式以后, 有以下定理:

定理 1. 设 r 是一个有界报酬函数, α 满足标准的随机逼近条件(stochastic-approximation conditions), 在每一时刻 t , Agent 依据当前的状态值函数 Q_t 选择执行的动作, 则 $\sum_{t=0}^{\infty} r_t$ 可以通过下面的迭代取得最大:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha(r_{t+1} + \gamma Q_t - Q_{t-1}) \nabla_{\boldsymbol{\theta}} Q_t \quad (3)$$

$$Q_{t+1} = Q(\boldsymbol{\theta}_{t+1}, \mathbf{S}_{t+1}) \quad (4)$$

这里, α 称为学习因子, γ 称为折扣因子. 下面, 我们给出一个简单的证明, 严格的证明可参见文献[17].

证明:由于 $Q(\theta_t, \mathbf{S}_t)$ 是 Q_t 的近似,我们可调节 θ_t 来使 $Q(\theta_t, \mathbf{S}_t)$ 尽量逼近 Q_t ,即使均方误差 $(Q_t - Q(\theta_t, \mathbf{S}_t))^2$ 最小.由梯度下降法, θ_t 可按照下式迭代:

$$\begin{aligned} \theta_{t+1} &= \theta_t - \frac{1}{2} \alpha \nabla_{\theta} (Q_t - Q(\theta_t, \mathbf{S}_t))^2 \\ &= \theta_t + \alpha (Q_t - Q(\theta_t, \mathbf{S}_t)) \nabla_{\theta} Q(\theta_t, \mathbf{S}_t) \end{aligned} \tag{5}$$

由于真实值 Q_t 很难得到,我们用一阶近似来代替,即 $Q_t = r_{t+1} + \gamma Q(\theta_t, \mathbf{S}_t)$,代入式(5),即得式(3).

基于定理 1,我们构建如下的 RLGD 算法:

设报酬函数 $r = -\omega_1(c - B)^2 - \omega_2(q - q_{ref})^2$.这里, c 是当前的数据到达率, B 是链路带宽, q 是当前的队列长度, q_{ref} 是目标队列长度, ω_1 和 ω_2 分别称为吞吐量权重和延时权重,它们的比反映了吞吐量和延时的折衷.在这一定义下,AQM的目标是,使数据到达率尽量接近链路带宽和使队列长度尽量稳定在目标值附近就相当于最大化 $\sum_{t=0}^{\infty} r_t$.RLGD 算法的结构如图 2 所示.

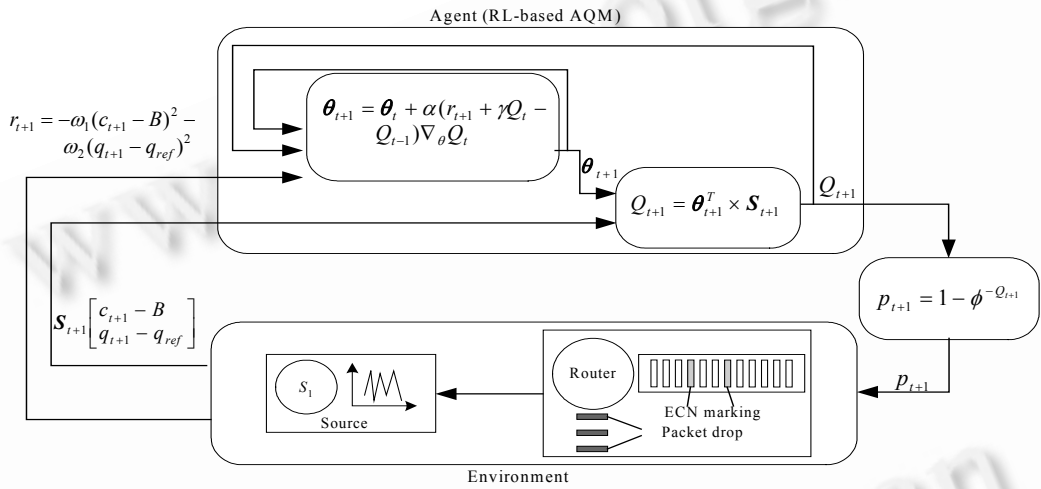


Fig.2 Structure of RLGD algorithm

图 2 RLGD 算法结构

设状态为当前到达率与链路带宽之差和当前队列长度与目标长度之差,将其表示为矢量形式:

$$\mathbf{S} = \begin{bmatrix} c - B \\ q - q_{ref} \end{bmatrix} \tag{6}$$

θ 为可调节的矢量参数,为计算方便,我们选择 Q 为 θ 和 \mathbf{S} 的内积,即

$$\begin{aligned} Q_{t+1} &= Q(\theta_{t+1}, \mathbf{S}_{t+1}) \\ &= \theta_{t+1}^T \times \mathbf{S}_{t+1} \\ &= [\theta_{t+1,1}, \theta_{t+1,2}] \times \begin{bmatrix} S_{t+1,1} \\ S_{t+1,2} \end{bmatrix} \\ &= \theta_{t+1,1}(c_{t+1} - B) + \theta_{t+1,2}(q_{t+1} - q_{ref}) \end{aligned} \tag{7}$$

这样, Q 的梯度为

$$\nabla_{\theta} Q_t = \begin{bmatrix} \frac{\partial Q_t}{\partial \theta_{t,1}} \\ \frac{\partial Q_t}{\partial \theta_{t,2}} \end{bmatrix} = \begin{bmatrix} c - B \\ q - q_{ref} \end{bmatrix} \tag{8}$$

代入式(3)得,

$$\begin{bmatrix} \theta_{t+1,1} \\ \theta_{t+1,2} \end{bmatrix} = \begin{bmatrix} \theta_{t,1} + \alpha(r_{t+1} + \gamma Q_t - Q_{t-1})(c_t - B) \\ \theta_{t,2} + \alpha(r_{t+1} + \gamma Q_t - Q_{t-1})(q_t - q_{ref}) \end{bmatrix} \quad (9)$$

$$Q_{t+1} = \theta_{t+1,1}(c_{t+1} - B) + \theta_{t+1,2}(q_{t+1} - q_{ref}) \quad (10)$$

即 RLGD 按照式(9)和式(10)进行迭代,得到 Q 值.其中 θ_1 和 θ_2 分别根据当前的到达率和队列长度进行调整,且调整的步长自适应变化.

最后,按照下式计算丢包概率:

$$p_{t+1} = 1 - \phi^{-Q_{t+1}} \quad (11)$$

这里, ϕ 是大于 1 的常数.在式(11)中,我们的公式保留了与 REM 算法中相似的形式.在这种形式下,一个数据包经过一条由 L 段链路组成的路径后被丢弃的概率为

$$p_{t+1}^{total} = 1 - \prod_{l=1}^L (1 - p_{t+1}^l) = 1 - \phi^{-\sum_l Q_{t+1}^l} \quad (12)$$

可见,RLGD 算法中的 Q 值与 REM 算法中的“价格”类似,也可以被看做是网络拥塞程度的一种度量.如果我们将其称为“ Q 价格”,则式(12)表示一条由 L 段链路组成的路径上的 Q 价格就等于各段链路路上的 Q 价格之和.

图 3 给出了 RLGD 算法的伪代码.

```

/*RLGD Algorithm*/
T: update period; c: current arrival rate; B: link bandwidth;
q: current queue length; q_ref: expected queue length;
alpha: learning factor; gamma: discount factor; w1: throughput weight; w2: delay weight;
At each packet arrival epoch do
{
  if (now()>last_time+T)
  {
    S[0]=c-B; S[1]=q-q_ref;
    r=-w1*(c-B)^2-w2*(q-q_ref)^2;
    theta[0]=theta[0]+alpha*(r+gamma*Q-Q_old)*(c-B);
    theta[1]=theta[1]+alpha*(r+gamma*Q-Q_old)*(q-q_ref);
    Q_old=Q;
    Q=theta[0]*S[0]+theta[1]*S[1];
    p=1-phi^-Q;
    last_time=now();
  }
  if (p>rand()/MaxRand())
    Mark or drop packet;
  Else
    Put packet into queue;
}

```

Fig.3 Pseudo-Code of RLGD algorithm

图 3 RLGD 算法的伪代码

2 仿真结果

ns-2^[18]是当前网络仿真的主要平台之一.我们使用 Linux 操作系统,在 ns-2 的 2.1b9 版本下对 RLGD 算法进行了仿真.仿真网络的拓扑结构如图 4 所示.节点 A,B 之间为瓶颈链路,我们的 RLGD 算法在节点 A 中实现,其余节点采用尾部丢弃(drop-tail)策略.节点 A 的缓存大小为 200 个数据包,数据包长固定为 1 000byte,仿真时间为 200s.

算法中的参数按下面的值设置:

- 更新周期 $T=2\text{ms}$
- 学习因子 $\alpha=0.0001$
- 折扣因子 $\gamma=0.98$

- 吞吐率权重 $\omega_1=0.01$
- 延时权重 $\omega_2=5$
- $\phi=1.001$

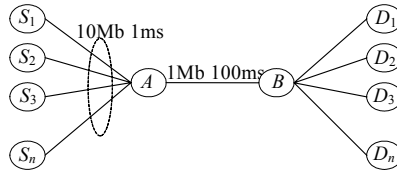


Fig.4 Simulation network topology
图4 网络仿真拓扑结构

为验证 RLGD 算法在各种网络环境下的性能,我们设计了以下 5 组仿真实验.

2.1 实验1

在这一仿真中,主要验证 RLGD 能否满足 AQM 算法的基本要求,即能否在不影响吞吐率的前提下将队列长度稳定在任意的目标值上.为此,我们设计了 80 个 TCP 流在 1s 时同时接入,并一直持续到仿真结束,图 5 显示了目标队列长度分别设为 20,50 和 120 时队列长度的变化情况,可以看出,在 3 种情况下 RLGD 都可以快速、准确地将队列长度稳定在目标值上.

我们还将 RLGD 的性能与 REM 和 PI 控制器作了比较(以下仿真均设队列的目标值为 50packets),结果如图 6 所示.

由图 6 可以看出,我们的 RLGD 算法不但收敛速度大大快于其他两种算法,而且消除了 REM 和 PI 算法在目标值附近振荡的现象,稳定性很好.

图 7 显示的是 RLGD,REM 和 PI 这 3 种算法下的吞吐率随时间变化的情况,可以看出,经过一定时间,3 种算法下的吞吐率都收敛到出口带宽上,但在 RLGD 算法下,收敛的速度最快.

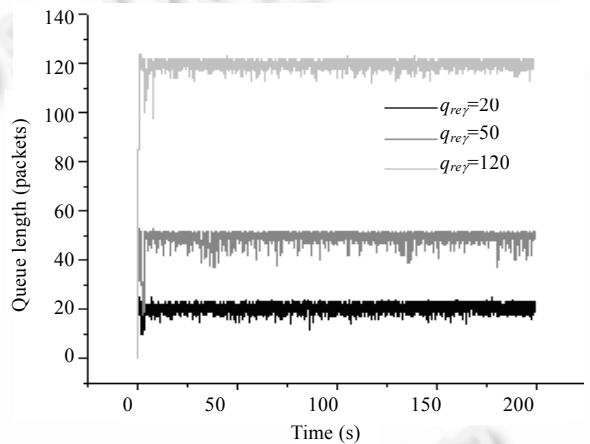


Fig.5 Queue length evolution of RLGD
图5 RLGD 控制下的队列长度变化

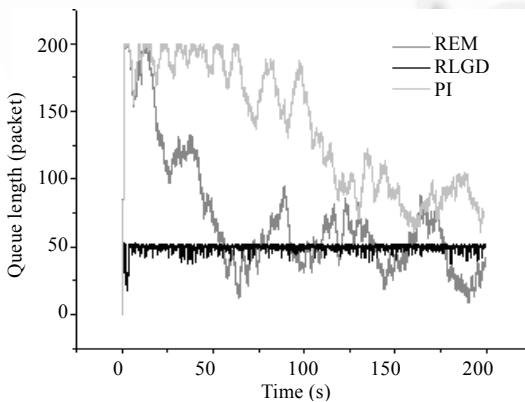


Fig.6 Queue length evolution of RLGD, REM and PI (RTT=100ms)

图6 RLGD,REM 和 PI 控制下的队列长度变化比较 (RTT=100ms)

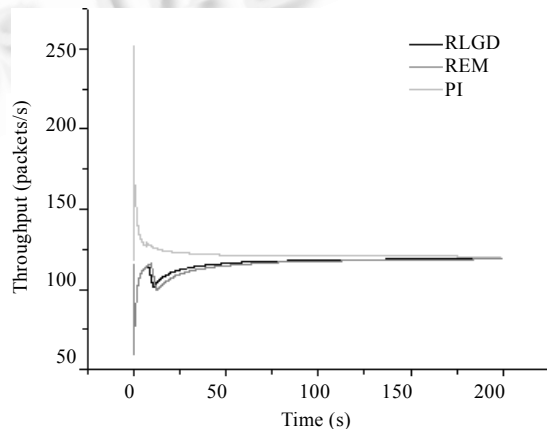


Fig.7 Throughput of RLGD, REM and PI

图7 RLGD,REM 和 PI 控制下的吞吐率

2.2 实验2

往返时间(RTT)是一个影响 AQM 性能很重要的因素.在实验 2 中,我们保持实验 1 的其他条件不变,但将瓶颈链路的 RTT 时间由 100ms 增加为 500ms,仿真结果如图 8 所示.可以看出,我们的 RLGD 算法仍然明显优于 REM 和 PI 算法.

2.3 实验3

在实验 3 中,我们检验 RLGD 在动态环境下的鲁棒性,为此我们设计了 20 个 TCP 流在 1s 时接入,在 75s 和 150s 时又各有 15 个 TCP 流同时接入,以模拟动态的网络环境.3 种算法的性能如图 9 所示.可以看出,在 REM 和 PI 算法下,动态 TCP 流的接入对队列长度的冲击较大,而我们的 RLGD 算法虽然也会造成队列的抖动,但幅度很小,并且很快能重新稳定到目标值.

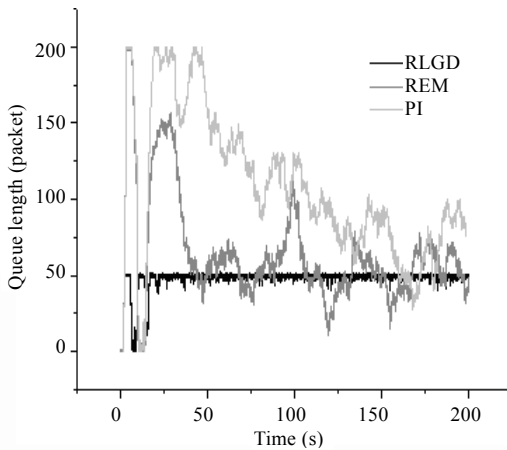


Fig.8 Queue length evolution of RLGD, REM and PI (RTT=500ms)
图 8 RLGD,REM 和 PI 控制下的队列长度变化比较 (RTT = 500ms)

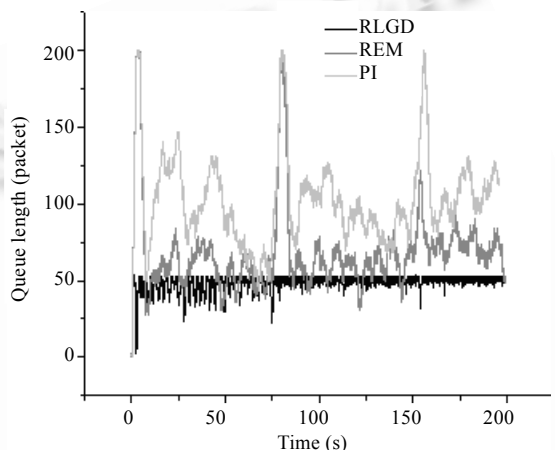


Fig.9 Queue length evolution of RLGD, REM and PI in dynamic environment
图 9 RLGD,REM 和 PI 算法在动态环境下的性能

2.4 实验4

由于 Web 等服务的出现,现在的 Internet 上存在大量持续时间较短的 TCP 流(通常只持续几秒或十几秒).由于 RTT 的影响,TCP 的源端可能刚对它们的存在作出反应就又消失了.因此,短流的存在对 AQM 的性能影响很大.在实验 4 中,我们比较了 REM,PI 和 RLGD 这 3 种算法抗短流冲击的能力.与实验 3 一样,我们设计了 20 个 TCP 流在 1s 时同时接入,在 75s 和 150s 时又各有 15 个 TCP 流同时接入.不同的是,75s 和 150s 时接入的流的持续时间均只有 10s.图 10 显示了 3 种算法的队列长度的变化.显然,对短流的冲击,REM 和 PI 算法控制下的队列都表现出很大的不稳定性,而我们的 RLGD 仍然能够将其稳定在目标值附近.

从实验 3 和实验 4 可以看出,我们的 RLGD 算法在动态变化的网络环境中表现出良好的鲁棒性.

2.5 实验5

由于 UDP 流没有 TCP 流那样的反馈调节机制,即它不管网络的拥塞状况,只按照自己源端的数据率来发送,因此对 AQM 性能的影响更大.在实验 5 中,我们设计了 5 个 TCP 流和 5 个 UDP 流在 1s 时同时开始发送数据,并记录下在 REM,PI 和 RLGD 这 3 种算法下的队列变化情况,如图 11 所示.与前面的结果比较,可以看出,在 UDP 流的影响下,3 种 AQM 算法的性能恶化都最为严重.在 RLGD 算法的控制下,虽然队列抖动比前面的实验都剧烈一些,但性能仍远远好于其他两种算法.

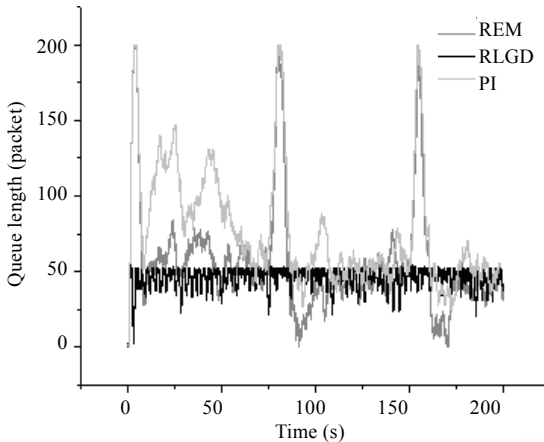


Fig.10 Queue length evolution of RLGD, REM and PI under short flow influence
图 10 短流对 RLGD,REM 和 PI 算法的影响

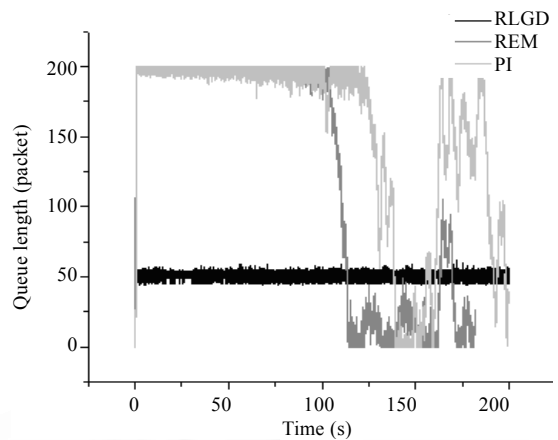


Fig.11 Queue length evolution of RLGD, REM and PI with non-responsive flow
图 11 RLGD,REM 和 PI 算法在 TCP 和 UDP 流共存环境下的性能

由上面的 5 组仿真可以看出,RLGD 算法不但可以将队列长度稳定在目标值附近,而且对各种动态变化的网络环境有较强的适应能力,性能远远优于 REM 和 PI 算法。

3 结 论

作为端到端拥塞控制机制的一部分,链路端的主动队列管理算法是源端流量控制算法的一种有效的补充。与以往的主动队列管理的分析和设计方法不同,我们将 AQM 算法归结为一个最优决策问题,利用人工智能中的再励学习方法,提出了一种新的主动队列管理算法 RLGD。与其他 AQM 算法相比,RLGD 以速率匹配和队列稳定为优化目标,根据当前网络的状态自适应地调整算法的更新步长,避免了队列长度在目标值附近的振荡,从而减小了延时抖动,提供了更好的端到端的 QoS 保证。值得指出的是,RLGD 算法在设计中无须知道源端的流量控制机制,可以适用于更广泛的网络环境,具有很好的可扩展性。

我们还通过大量的仿真对 RLGD 与 REM 和 PI 算法的性能进行了比较,结果显示,RLGD 算法在各种网络环境下的性能都要优于其他两种算法,具有良好的稳定性和鲁棒性。

References:

- [1] Jacobson V, Karels MJ. Congestion avoidance and control. ACM SIGCOMM Computer Communication Review, 1988,18(4):314~329.
- [2] Floyd S, Jacobson V. Random early detection gateways for congestion avoidance. IEEE/ACM Trans. on Networking, 1993,1(4): 397~413.
- [3] Floyd S. A report on some recent development in TCP congestion control. IEEE Communication Magazine, 2001,39(4):84~90.
- [4] Christiansen M, Jeffay K, Ott D, Smith FD. Tuning RED for Web traffic. In: Proc. of the ACM SIGCOMM 2000. Stockholm: ACM Press, 2000. 139~150.
- [5] Ott TJ, Lakshman TV, Wong LH. SRED: Stabilized RED. In: Proc. of the INFOCOM'99. New York: IEEE Communications Society, 1999. 1346~1355.
- [6] Lin D, Morris R. Dynamics of random early detection. In: Proc. of the SIGCOMM'97. Cannes: ACM Press, 1997. 127~137.
- [7] Anjum F, Tassiulas L. Balanced-RED: An algorithm to achieve fairness in Internet. <http://www.isr.umd.edu/CSHCN/>
- [8] Feng W, Kandlur DD, Saha D, Shin KG. A self-configuring RED gateway. In: Proc. of the INFOCOM'99. New York: IEEE Communications Society, 1999. 1320~1328.
- [9] Feng W, Kandlur DD, Saha D, Shin KG. Blue: A new class of active queue management algorithms. Technical Report, UM CSE-TR-387-99, 1999.

- [10] Hollot CV, Misra V, Towsley D, Gong WB. A control theoretic analysis of RED. In: Proc. of the INFOCOM 2001. Anchorage, AK: IEEE Communications Society, 2001. 1510~1519.
- [11] Hollot CV, Misra V, Towsley D, Gong WB. On designing improved controllers for AQM routers supporting TCP flows. In: Proc. of the INFOCOM 2001. Anchorage Alaska: IEEE Communications Society, 2001. 1726~1734.
- [12] 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 2000. Stockholm: ACM Press, 2000. 151~160.
- [13] Kelly FP, Maulloo A, Tan D. Rate control in communication networks; shadow prices, proportional fairness and stability. Journal of the Operational Research Society, 1998,49:237~252.
- [14] Athuraliya S, Lapsley DE, Low SH. Random early marking for Internet congestion control. In: Proc. of the IEEE Globecom'99. Rio de Janeiro: IEEE Communications Society, 1999. 1747~1752.
- [15] Kunniyur S, Srikant R. Analysis and design of an adaptive queue (avq) algorithm for active queue management. In: Proc. of the ACM Sigcomm 2001. San Diego: ACM Press, 2001. 123~134.
- [16] Firoiu V, Borden M. A study of active queue management for congestion control. In: Proc. of the INFOCOM 2000. Tel-Aviv Israel: IEEE Communications Society, 2000. 1435~1444.
- [17] Sutton R, Barto AG. Reinforcement Learning: An Introduction. MIT Press, 1998.
- [18] UCN/LBL/VINT. Network Simulator-NS2. <http://www.mash.cs.berkeley.edu/ns>
- [19] Tong H, Brown TX. Adaptive call admission control under quality of service constraints: A reinforcement learning solution. IEEE Journal of Selected Areas in Communication, 2000,18(2):209~221.
- [20] Nie JH, Haykin S. A Q-learning-based dynamic channel assignment technique for mobile communication systems. IEEE Trans. on Vehicular Technology, 1999,48(5):1676~1687.

第 4 届中国信息和通信安全学术会议(CCICS 2005)

征 文 通 知

中国信息和通信安全学术会议(CCICS)是国际信息和通信安全学术会议(International Conference on Information and Communications Security, 简称 ICICS)的地方版,已成功举办了 3 届,第 1 届由中国科学院信息安全技术工程研究中心主办,于 1999 年 12 月召开;第 2 届由上海交通大学计算机学院主办,于 2001 年 5 月召开;第 3 届由武汉大学计算机学院主办,于 2003 年 3 月召开。该会的规模和影响逐届扩大。第 4 届中国信息和通信安全学术会议(CCICS 2005)拟定于 2005 年 5 月在陕西西安举行。热忱欢迎所有涉及信息安全、通信安全理论和技术方面的研究论文提交本次会议进行交流。会议论文集将由科学出版社出版,会议的优秀论文将被推荐到《软件学报》。

一. 征文要求

论文须为未公开发表并且未向学术刊物和其他学术会议投稿的最新研究成果。文稿使用中文或英文书写,字数一般不超过 6000 字。请将论文(word 文档)全文(注明作者的联系电话和 E-mail 地址)发送到 yangbo@mail.xidian.edu.cn。

二. 重要日期

征文截止日期: 2004 年 7 月 31 日

文章录用通知: 2004 年 9 月 31 日

录用论文定稿: 2004 年 10 月 31 日

三. 联系方式

联系人: 西安电子科技大学通信工程学院 杨波 教授

通信地址: (710071)西安电子科技大学 106 信箱

电话: 029-8203028

E-mail: yangbo@mail.xidian.edu.cn