

文章编号: 1001-2486(2006)05-0047-05

# 大延时网络中的主动队列管理机制\*

刘明, 窦文华, 张鹤颖

(国防科技大学 计算机学院, 湖南长沙 410073)

**摘要:**研究了网络延时对路由器主动队列管理机制的影响, 分析了几种典型主动队列管理算法在大延时网络中的性能。在介绍了基于内模补偿的 DC-AQM 算法的优缺点之后, 根据 PID 控制器延时补偿的 Ziegler-Nichols 设定方法, 提出了 ZNDC(Ziegler-Nichols delay compensation) AQM 算法并进行了仿真实验验证, 实验结果表明算法达到了预期的目标。

**关键词:**拥塞控制; 主动队列管理; 控制理论

**中图分类号:** TP393 **文献标识码:** A

## AQM Algorithms in Large-delay Networks

LIU Ming, DOU Wen-hua, ZHANG He-ying

(College of Computer, National Univ. of Defense Technology, Changsha 410073, China)

**Abstract:** The study was made to research into the effect of the network delay upon the AQM mechanism and to analyze the properties of several typical AQM algorithms in delay networks, which was put forward recently. With a discussion of the advantage and disadvantage of the DC-AQM, this paper, according to the Ziegler-Nichols approach, in which PID controller may compensate for the time delay, proposed an AQM algorithm using Ziegler-Nichols Method that can reduce the negative impact caused by the large delay. Finally, the performance of AQM with Ziegler-Nichols Delay Compensation is evaluated through NS simulations. Results from the simulation reached the objective expected.

**Key words:** congestion control; AQM( active queue management); control theory

AQM 是路由器中支持拥塞控制的主要机制, 它的主要思想是在路由器的缓冲区没有溢出之前, 根据网络的拥塞状况, 以一定的概率丢弃报文, 克服舍弃队列的缺陷。1998 年 IETF 建议在路由器中采用 AQM 机制(RFC2309), 并推荐 RED 作为候选算法。但是 RED 很难找到适用于网络条件大范围变化的参数设置方法。因此设计新的主动队列管理算法成为计算机网络研究的一个热点, 相继提出了 ARED 算法、PI 算法、AVQ 算法、REM 算法、PIP 算法、PD 算法等。

近年来, 无线局域网、无线自组网络(ad hoc)、卫星网络等无线网络得到了快速发展和广泛应用, 使得因特网在结构和传输介质方面的异构性越来越显著。这些网络的共同特点是传输延时大, 而上述 AQM 算法没有充分考虑网络延时对算法性能的影响, 在提出这些算法的文献中也没有验证它们在大延时网络中的性能。文献[1]采用控制论中的内模补偿原理设计用于 AQM 的延时补偿控制器, 提出了 DC-AQM 算法, 它的稳定性有待验证。

### 1 大延时网络环境下典型 AQM 算法的性能仿真

已有的大多数关于 AQM 机制的文献在验证算法的性能时都设定网络的延时为几十 ms, 为了验证它们在大延时网络环境下的性能, 采用图 1 所示的拓扑结构进行 NS 仿真实验, 在源节点  $s_i$  和目的节点  $d_i$  之间建立连接,  $s_i$  与  $r_1$  间的延时为 delay1,  $d_i$  与  $r_2$  间的延时为 delay2。路由器  $r_1$  和  $r_2$  之间的链路是所有连接的唯一瓶颈链路, 带宽为 100Mb/s, 链路延时为 10ms, 路由器的缓冲区容量为 800 个报文。端节

\* 收稿日期: 2006-08-01

基金项目: 国家自然科学基金资助项目(90104001)

作者简介: 刘明(1977—), 男, 博士生。

点与路由器之间的链路带宽均为 100Mb/s,平均报文大小为 512 字节,负载  $N = 200$ ,参考队列长度为 100。我们将验证 NS 中发布的 ARED,PI,REM,AVQ 等算法的性能。

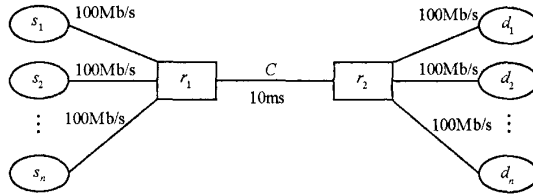


图 1 仿真实验网络拓扑

Fig.1 Simulation topology

上述几种算法在各自的文献中有较详细的小延时条件下的性能仿真实验,它们的队列长度能够较好地收敛到目标值附近,可以保证报文通过路由器有较小的排队延时,同时它们有较高的链路利用率和较低的报文丢弃率。当延时增大时,实验结果(实验 1)如图 2 所示。图 2 给出了 ARED, PI, REM, AVQ 算法在  $\text{delay1} = \text{delay2} = 100\text{ms}$  时的队列长度。

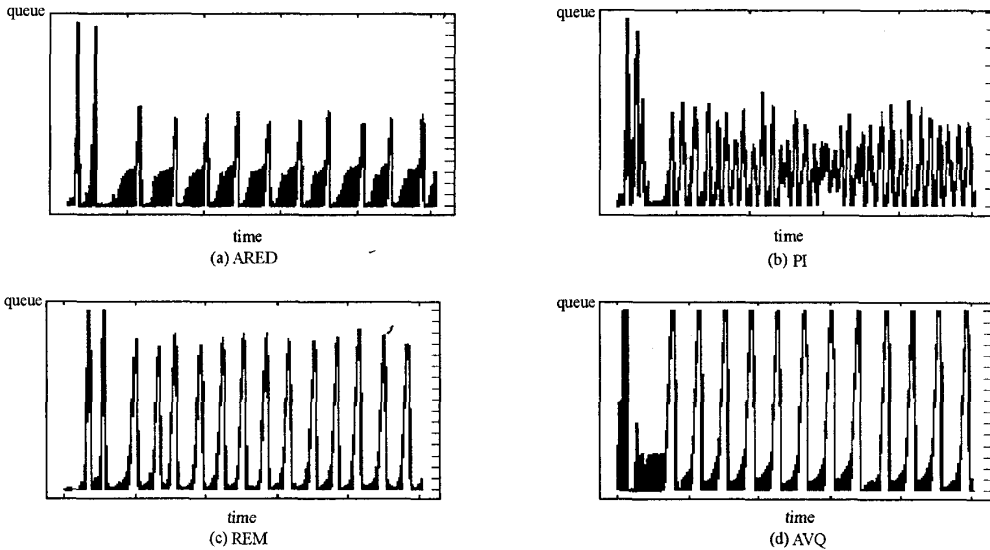


图 2 实验 1 中典型 AQM 算法队列长度随时间的变化曲线

Fig.2 Evolution of the queue length with FTP flows in simulation 1

在实验中,几种算法在延时较大的情况下存在一些共同的特点,首先是缓冲区队列的大幅震荡,造成了很大的端到端延时抖动;其次是链路利用率低,尤其是 ARED,降低了近 20%;最后是队列长度不能收敛到目标值附近。可见随 NS 发布的几种 AQM 算法均不能适应延时较大的网络环境,因此迫切需要能够对延时进行补偿的 AQM 算法。

## 2 延时补偿算法设计

### 2.1 Ziegler-Nichols 整定方法

为了延时补偿算法设计的简化,将 TCP 二阶时滞模型进行一阶时滞模型的拟合:

$$G(s) = \frac{K_m e^{-sR_0}}{(T_1 s + 1)(T_2 s + 1)}, \quad \widehat{G}(s) = \frac{K_m e^{-s(T_1 + T_2 + R_0 - \sqrt{T_1^2 + T_2^2})}}{\sqrt{T_1^2 + T_2^2} s + 1} \quad (1)$$

令  $L = T_1 + T_2 + R_0 - \sqrt{T_1^2 + T_2^2}$ ,  $T_0 = \sqrt{T_1^2 + T_2^2}$ , 并使用一阶 Pade 近似表示时滞环节

$$\widehat{G}(s) = \frac{K_m(1 - Ls/2)}{(T_0s + 1)(1 + Ls/2)}$$

经典控制理论中,对如(1)式的一阶加纯滞后模型,在设计控制器时如果  $L/T_0 > 0.7$ ,采用 Smith 补偿器对滞后加以补偿;而当  $L/T_0 < 0.7$  时,采用 Ziegler-Nichols 方法进行延时补偿。当前的网络应用环境中通常满足后一种条件,因此使用 Ziegler-Nichols 整定公式来设计控制器克服延时环节的影响。控制器如(2)式:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s\right), \quad K_p = 1.2T_0/(K_m L), \quad T_i = 2L, \quad T_d = 0.5L \quad (2)$$

在实际的网络应用环境中通常满足  $T_1 \gg T_2$ ,因此取  $T_0 = \sqrt{T_1^2 + T_2^2} \approx T_1 = R_0^2 C/(2N)$ ,  $L = T_1 + T_2 + R_0 - \sqrt{T_1^2 + T_2^2} \approx 2R_0$ ,可得:

$$K_p = 1.2N/(R_0 C)^2, \quad T_i = 4R_0, \quad T_d = R_0$$

(2)式表示为离散增量形式:

$$\begin{aligned} \Delta p(k) &= 1.2N/(R_0 C)^2 \{ [1 + T_i/(4R_0) + R_0/T_i]e(k) - (1 + 2R_0/T_i)e(k-1) + (R_0/T_i)e(k-2) \} \\ &= (1.2/C^2)(N/R_0) \{ [1/R_0 + T_i/(4R_0^2) + 1/T_i]e(k) - (1/R_0 + 2/T_i)e(k-1) + (1/T_i)e(k-2) \} \end{aligned}$$

采用文献[2]中的方法在线估计负载  $N$ ,并根据平衡点计算延时  $R_0$ ,  $R_0 = q_0/C + T_p = \frac{\sqrt{2}N}{C\sqrt{p_0}}$ ,我们

称之为 ZNDC(AQM with Ziegler-Nichols delay compensation)算法。

## 2.2 算法的仿真与性能评价

### 2.2.1 实验 2: 实验 1 环境中 ZNDC 的性能

如图 3 所示,与实验 1 结果比较,ZNDC 在大延时网络环境下性能优于 ARED、PI、REM、AVQ 等算法,它能收敛到目标值附近,并且队列振荡较小,减小了延时对算法的影响。图 4 则显示 ZNDC 的链路利用率在该环境下高于其他几种算法。

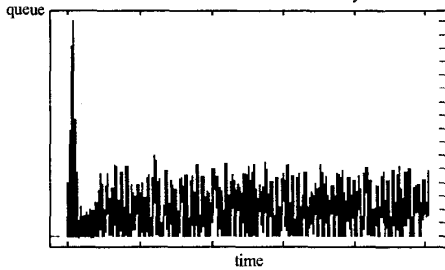


图 3 采用 ZNDC 算法的队列长度  
Fig.3 The queue length using ZNDC

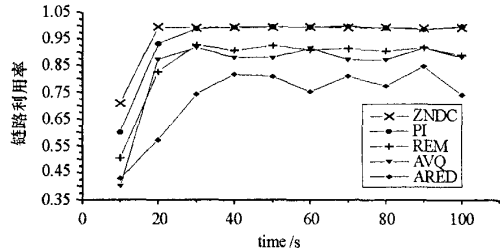


图 4 链路利用率比较  
Fig.4 The link utilization

### 2.2.2 实验 3: 与其他延时补偿算法的比较

分析 AQM 算法在大延时网络中性能的文献比较少,文献[1]在这方面进行了有益的探索,它使用内模控制设计了具有延时补偿能力的 PID 形式的控制器(DCAQM),丢弃率的表达式为:

$$\Delta p(k) = K_p [(1 + T_i/T_i + T_d/T_i)e(k) - (1 + 2T_d/T_i)e(k-1) + (T_d/T_i)e(k-2)]$$

其中  $K_p = \frac{2T_0 + L}{2.6K_m L}$ ,  $K_i = \frac{1}{1.3K_m L}$ ,  $K_d = \frac{T_0}{2.6K_m}$ ,  $T_i = \frac{K_p}{K_i}$ ,  $T_d = \frac{K_d}{K_p}$ ,  $e(k) = q(k) - q_{ref}$ 。

丢弃率计算公式为:

$$p(k+1) = p(k) + 1.34183 \times 10^{-3} e(k) - 2.66093 \times 10^{-3} e(k-1) + 1.31913 \times 10^{-3} e(k-2)$$

在 delay1=5ms, delay2=100ms 时仿真 DCAQM 算法和 ZNDC 算法,结果如图 5 所示。

实验中 DCAQM 具有克服延时的作用,没有出现大幅震荡。但是虽然设置参考队列长度为 100,在负载为 200 时队列长度能收敛到 100,但负载为 400 时队列长度则收敛到 200 左右,负载为 100 时收敛到

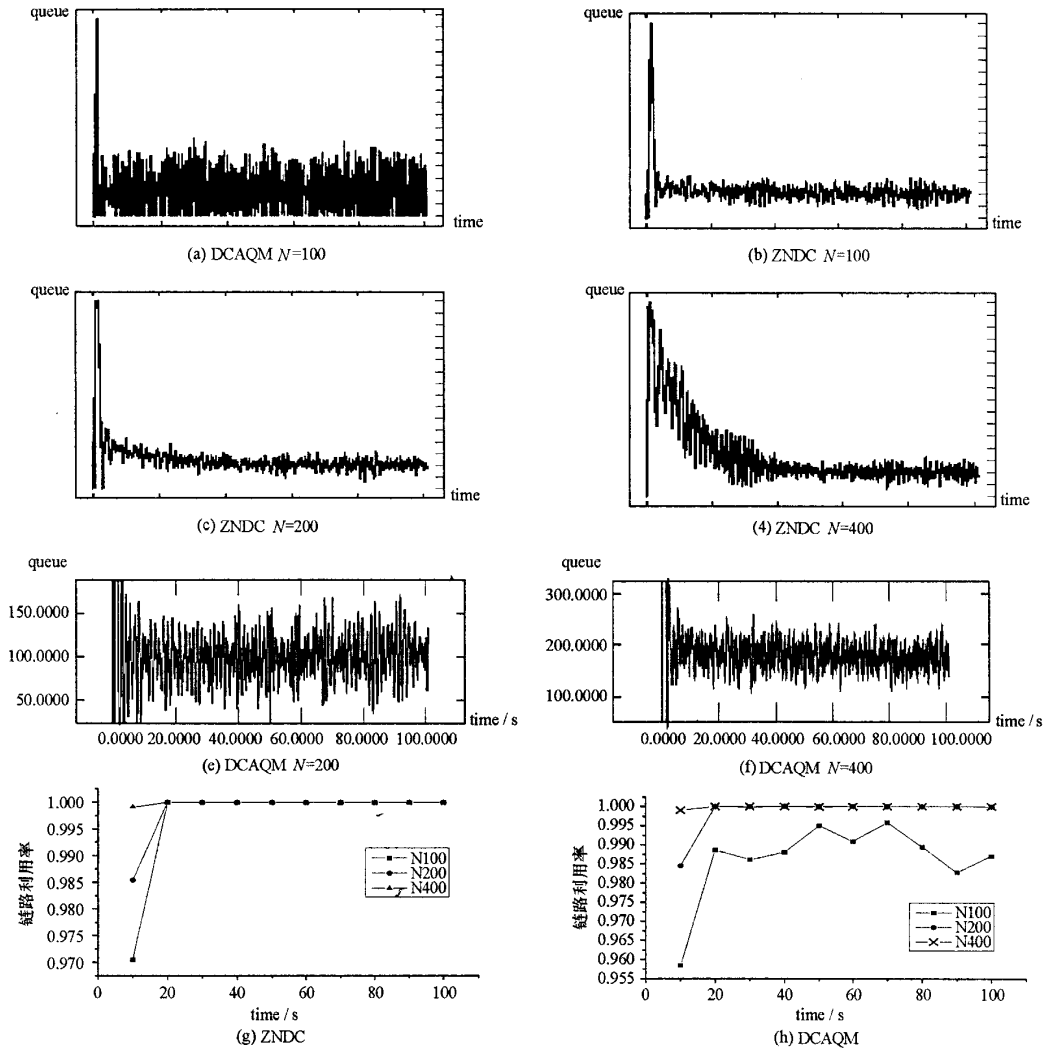


图5 实验3中队列长度随时间的变化曲线以及链路利用率

Fig.5 Evolution of the queue length and link utilization in simulation 3

50左右。实验中发现算法的性能与负载和延时的比值密切相关。从它的丢弃概率计算表达式可以看出它的微分作用很强,积分作用弱,在环境改变时存在稳态误差。该算法不能适应负载和延时的变化,这与文献[1]结论部分的总结相一致。实验中 ZNDC 队列长度的振荡较小,同时在几种负载下都收敛到参考点附近,没有出现 DCAQM 中队列长度与负载或延时的耦合。

### 2.2.3 实验4:综合性能比较

在本实验中验证各种延时的流混合存在并且负载变化时算法的性能。设定回返时间均匀分布于 40~1000s,在 0~1s 时 400 个 TCP 流加入,在 100s 时 200 个 TCP 流退出,在 200s 时 200 个 TCP 流重新加入。

由图 6 的实验结果可以看出,PI 的链路利用率比较低;REM 的队列振荡较大,链路利用率比较低;AVQ 的队列振荡很大,报文丢弃率高;ARED 的队列振荡较大,链路利用率比较低;DCAQM 在负载  $N=400$  时表现出较好的性能,在  $N=200$  时参考队列长度发生偏移,链路利用率低。ZNDC 则表现出较小的队列振荡,较高的链路利用率和较低的报文丢弃率。

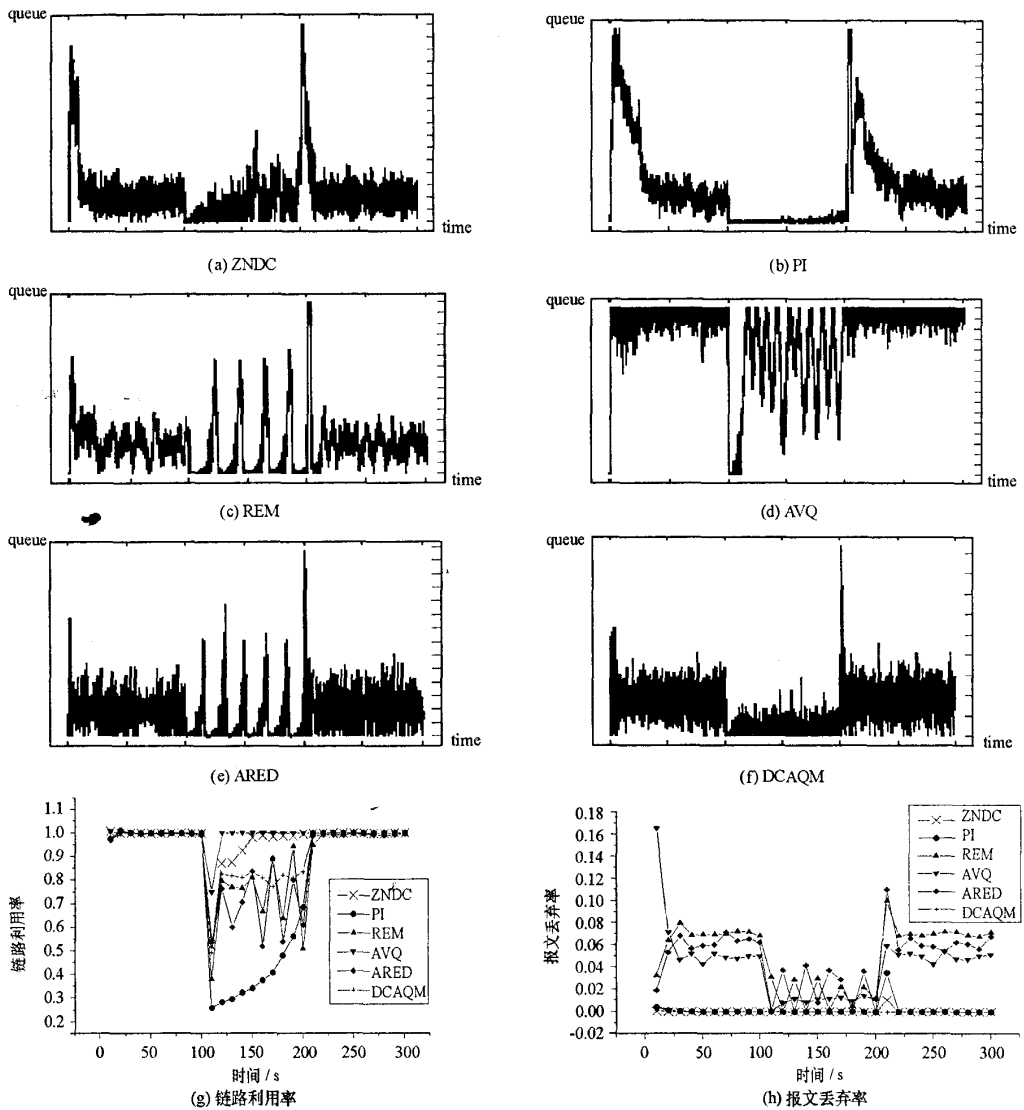


图6 实验4中队列长度随时间的变化曲线以及链路利用率和报文丢弃率  
Fig.6 Evolution of the queue length, link utilization and packet drops ratio in simulation 4

#### 4 结论

分析了几种典型算法在大延时网络中的性能。根据PID控制器延时补偿的Ziegler-Nichols设定算法,提出了ZNDC AQM算法。通过NS仿真实验验证了该算法的性能,实验结果表明算法达到了预期的目标。

#### 参考文献:

- [1] 任丰原,林闯,任勇,等. 大时滞网络中的拥塞控制算法[J]. 软件学报,2003,14(3): 503 - 511.
- [2] Li J S, Leu M S. Network Fair Bandwidth Share Using Hash Rate Estimation[J]. Networks, 2002,40(3):125 - 141.
- [3] 刘明, 蔡文华, 张鹤颖. 主动队列管理机制中PI算法的一种参数配置方法[J]. 国防科技大学学报, 2005(3).
- [4] Dou W H, Liu M, Zhang H Y. A Framework for Designing Adaptive AQM Schemes[C]. Proceedings of ICCNMC 2005, China, 2005.
- [5] 杨洪勇,孔祥新,张福增. 具有通信延时的AQM控制算法的稳定性[J]. 计算机研究与发展, 2005(7).

