

主动队列管理机制中 PI 算法的一种参数配置方法*

刘 明, 窦文华, 张鹤颖, 张 镒

(国防科技大学 计算机学院, 湖南 长沙 410073)

摘 要 :研究了主动队列管理机制中 PI 控制器的参数配置方法。AQM 中 PI 算法默认的参数以及一些文献提出的参数配置方法主要是从稳定性考虑,没有保证系统的动态性能指标。在控制论中,“二阶最优模型”是具有较好的动态性能和稳态性能的二阶系统。提出了一种 PI 控制器的参数配置方法,可以使 TCP/AQM 控制模型的开环传递函数具有“二阶最优模型”的形式。NS 仿真结果表明采用该方法配置的参数可以提高 PI 控制器的动态性能并保证系统的稳态性能。最后指出 PI 控制器参数配置对网络状态的敏感性,以及今后的研究方向。

关键词 :拥塞控制 ;主动队列管理 ;控制理论 ;二阶最优模型

中图分类号 :TP393 **文献标识码** :A

Study on the Parameter Setting of PI Controller in Active Queue Management

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

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

Abstract :Active Queue Management (AQM) is an effective method to improve the performance of end-to-end congestion control. PI controller is a popular AQM scheme having good steady-state properties. We present the guidelines for setting parameters of PI controller using the “second-order optimization model” instead of the margin of stability to improve transient behavior. The performance of PI controller with new parameters is verified and compared with that of default ns parameters using ns simulations. The results show that the dynamic behavior of PI controller is improved. We present several future directions in the end.

Key words :congestion control ;active queue management ;control theory

当前网络中的许多应用要求低延迟、低丢失率的服务,现有的端到端拥塞控制机制难以满足这些性能要求, IETF 建议在路由器中采用主动队列管理机制 (AQM)。AQM 的目的是设计一种有效的拥塞早期检测机制,将队列长度控制在较低的水平,避免缓冲区溢出,并减少报文丢失。目前,已经提出了多种 AQM 算法,如 RED^[1],PI^[2],REM,AVQ 等。许多 AQM 算法依赖于直觉进行设计,没有系统的运用理论工具。PI 算法在这方面做了探索性的研究,它借助于控制论进行设计,有效地消除了稳态误差,使队列长度保持稳定。文献 [3] 的结论表明 PI 算法能够加速 Web 应用的响应。但是 PI 控制器按照它的默认参数配置存在动态性能差的问题。文章将从新的角度解释 PI 控制器提出的理论依据,并提出综合考虑系统稳定性和动态性能的参数配置方法。

1 主动队列管理机制中的 PI 算法

在文献 [4] 中,采用 TCP 流分析建立了基于拥塞 AQM 路由器的 TCP 动态模型:

$$\begin{cases} \dot{W} = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))} \mu(t-R(t)) \\ \dot{q}(t) = \frac{W(t)}{R(t)} N(t) - C \end{cases}$$

* 收稿日期:2004-11-09
基金项目:国家自然科学基金项目资助(90104001)
作者简介:刘明(1977—),男,博士生。

该模型作线性化处理, TCP/AQM 组成的反馈控制系统框图如图 1 所示。图中 e^{-sR_0} 为延时环节,

$P_{tcp}(s) = (\frac{R_0 C^2}{2N^2})(s + \frac{2N}{R_0^2 C})$, $P_{queue}(s) = (\frac{N}{R_0})(s + \frac{1}{R_0})$ (R_0 为 TCP 连接的回返时间, C 为链路带宽, N 为激活的 TCP 连接数目)

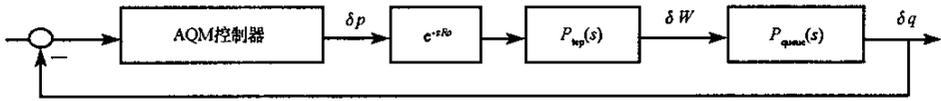


图 1 TCP/AQM 组成的反馈控制系统框图

Fig.1 Block diagram of a TCP/AQM control system

忽略延时, TCP 系统模型为串联的两个惯性单元, 它的传递函数为:

$$G(s) = \frac{K_m}{(T_1 s + 1)(T_2 s + 1)} \tag{1}$$

$$K_m = \frac{(R_0 C)^3}{4N^2}, \quad T_1 = \frac{R_0^2 C}{2N}, \quad T_2 = R_0 \tag{2}$$

PI 算法就是根据图 1 的控制模型并应用控制论方法设计的 AQM 控制器^[2], 它用比例积分控制器取代 RED^[1]的低通滤波器, 基于瞬时队列长度决定报文丢弃概率。其传递函数为:

$$C_{PI}(s) = K_{PI} \frac{s/\omega_g + 1}{s} \tag{3}$$

它的丢弃概率的计算方法为:

$$p = a \cdot (q - q_{ref}) - b \cdot (q_{old} - q_{ref}) + p_{old} \tag{4}$$

其中, a, b 为常数, q_{ref} 为参考队列长度, q_{old} 为前一次采样的队列长度, p_{old} 为前一次计算的丢弃概率。

文献 2 中提出了用控制论进行 PI 参数配置的方法。

当 $\frac{2N^-}{(R^+)^2 C} \ll \frac{1}{R^+}$ 时, 令

$$\omega_g = \frac{2N^-}{(R^+)^2 C}, \quad K_{PI} = \omega_g \left| \frac{(j\omega_g/P_{queue} + 1)}{(R^+ C)^2 (2N^-)^2} \right| \tag{5}$$

则, 当 $N \geq N^-$ 并且 $R \geq R^+$ 时, PI 控制器 $C_{PI}(s) = K_{PI}(s/\omega_g + 1)/s$ 使图 1 的反馈控制系统稳定, 并且相位稳定裕量约为 $90^\circ - (180^\circ/\pi)\omega_g^2$ 。

2 使用二阶最优模型的 PI 控制器参数配置

2.1 二阶最优模型

设二阶系统的开环传递函数为

$$G(s) = \frac{K}{s(Ts + 1)} \quad (K < \frac{1}{T}) \tag{6}$$

此时系统的动态性能指标与模型参数之间有准确的数学关系, 可以推导出它的主要动态性能指标, 例如阻尼系数 ξ 、自振角频率 ω_n 与模型参数之间的关系如下:

$$\omega_n = \sqrt{K/T}, \quad \xi = \frac{1}{2} \sqrt{1/KT} \tag{7}$$

由此便可求出系统阶跃响应的时域性能指标。反之, 若给定时域性能指标要求, 也可以求出 ω_n 和 ξ 并按 (7) 式确定模型参数 K 和 T 。通常把 $\xi = 0.707$, 即 $KT = 1/2$ 的情况称作“二阶最优模型”, 它的相位稳定裕量 $\gamma = 65.5^\circ$, 截止角频率 $\omega_c = 0.644\omega_n$, 动态时域指标为超调量 $\sigma\% = 4.6\%$, 过渡过程时间 $t_s \approx 6T = 3/K$ 。

2.2 二阶最优模型的 PI 参数配置方法

二阶最优模型校正的具体思路是 通过选择控制器的结构和参数使系统开环传递函数具有 (6) 式的

形式,并满足阻尼系数 $\xi = 0.707$,从而使系统具有良好的动态性能。二阶最优模型校正是经典控制在某些领域中(如简单的电力拖动系统及过程控制等)具有实际应用的串联校正方法。

在图1中,忽略延时环节,TCP窗口调节机制与路由器中的队列变化组成的被控对象为一个二阶系统。其传递函数为(1)式。当 $2NA(R_0^2C) < 1/R_0$ (即 $T_1 > T_2$)时,若使系统开环传递函数具有(6)式的形式,则需要引入如下形式的校正器:

$$G_c(s) = \frac{T_1s + 1}{2K_mT_2s} \quad (8)$$

此时系统的开环传递函数为

$$G(s) = G_c(s)G(s) = \frac{K_m}{(T_1s + 1)(T_2s + 1)} \frac{T_1s + 1}{2K_mT_2s} = \frac{1}{2T_2s(T_2s + 1)} = \frac{1/2T_2}{s(T_2s + 1)} \quad (9)$$

由式(7)可知此时系统的阻尼系数 $\xi = 0.707$,系统开环传递函数为二阶最优模型。

PI控制器的传递函数(8)式可转化为如下形式:

$$C_P(s) = \frac{T_1}{2K_mT_2} + \frac{1}{2K_mT_2s} \quad (10)$$

对(10)式进行z变换得:

$$C_P(z) = \frac{T_1}{2K_mT_2} + \frac{1}{2K_mT_2} \frac{z}{z-1} = \frac{(T_1+1)z - T_1}{2K_mT_2(z-1)} \quad (11)$$

(11)式为丢弃概率偏差与瞬时队列长度偏差之间的传递函数,瞬时队列长度偏差 $\delta q = q - q_{ref}$,丢弃概率的参考点 $p_{ref} = 0$,所以 $\delta p = p$ 。(11)式可以写成如下形式:

$$\frac{p(z)}{\delta q(z)} = \frac{(T_1+1)z - T_1}{2K_mT_2(z-1)} \quad (12)$$

转化为

$$p(z)(z-1) = \delta q(z) \left(\frac{T_1+1}{2K_mT_2} z - \frac{T_1}{2K_mT_2} \right) \quad (13)$$

当 $t = kT, T = 1/f_s$ 时,(13)式可以写成差分方程的形式:

$$p(kT) - p((k-1)T) = \frac{T_1+1}{2K_mT_2} \delta q(kT) - \frac{T_1}{2K_mT_2} \delta q((k-1)T) \quad (14)$$

与(4)式比较,可得PI控制器的参数为:

$$a = \frac{T_1+1}{2K_mT_2}, \quad b = \frac{T_1}{2K_mT_2} \quad (15)$$

对于该参数配置方法的稳定性,有如下结论:

如果 $\frac{2N^-}{(R^+)^2C} \ll \frac{1}{R^+}$, 当 $N \geq N^-$ 并且 $R \geq R^+$ 时,PI控制器 $G_c(s) = \frac{T_1s+1}{2K_mT_2s}$ 使图1的反馈控制系统

稳定,并且相位稳定裕量约为 39.5° ,其中 $K_m = (R^+C)^3/4N^{-2}$, $T_1 = R^{+2}C/2N^-$, $T_2 = R^+$ 。

3 仿真结果

文章采用网络仿真器NS2研究PI的队列长度和响应速度。网络拓扑结构如图2所示。端节点 s_i 和 d_i 之间建立连接,路由器 r_1 和 r_2 之间构成瓶颈链路,链路带宽为15Mbps,传输延迟为10ms,其它链路的带宽为100Mbps。发送端采用TCP-Reno,平均报文长度为500字节,TCP连接的传输延迟均匀分布在40~240ms之间。路由器中的缓冲区最多容纳800个报文。用FTP流模拟一直有数据发送的长连接, N 表示激活的FTP流的数目。

使用典型的数据来进行参数配置: $C = 3750 \text{ packets/s}$, $N^- = 20$, $R^+ = 0.246$ 。NS中PI的默认参数为 $a = 1.822 \times 10^{-5}$, $b = 1.816 \times 10^{-5}$ (下文简称参数配置1)。由(5)式可计算出PI的参数: $a = 2.403 \times 10^{-6}$, $b = 2.043 \times 10^{-6}$ (下文简称参数配置2)。由(15)式可计算出使用'二阶最优模型'配置的PI参数:

$a = 2.7638 \times 10^{-5}$, $b = 2.3502 \times 10^{-5}$ (下文简称参数配置3)。丢弃概率的更新计算频率为 160Hz, 参考队列长度为 100。

实验一 考察网络负载 $N = 200$ 时三种参数配置的表现。在时间 $0 \sim 4s$ 内创建 200 个 FTP 连接。图 3 (a)(b)(c) 分别是 $N = 200$ 时在参数配置 1、2、3 情况下的队列长度随时间的变化曲线图。由图中可以看出使用参数配置 1 和 2 分别在 100s 和 20s 时队列长度稳定在参考队列长度附近(图 3(a)(b)), 使用参数配置 3 在小于 5s 时就已经将队列长度稳定在参考队列长度附近(图 3(c))。实验一的三种参数配置稳态链路利用率都接近于 1, 它们在整个过程中的平均队列长度分别为 250、130、102。仿真实验的结果表明, 参数配置 3 的响应最快, 平均队列长度最接近于目标值。

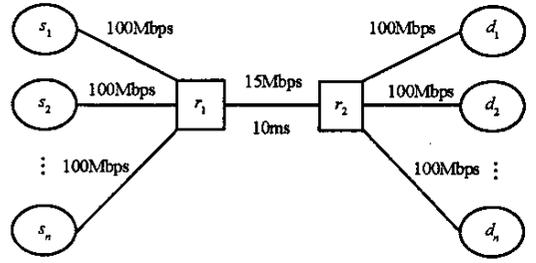


图 2 仿真拓扑

Fig. 2 Simulation topology

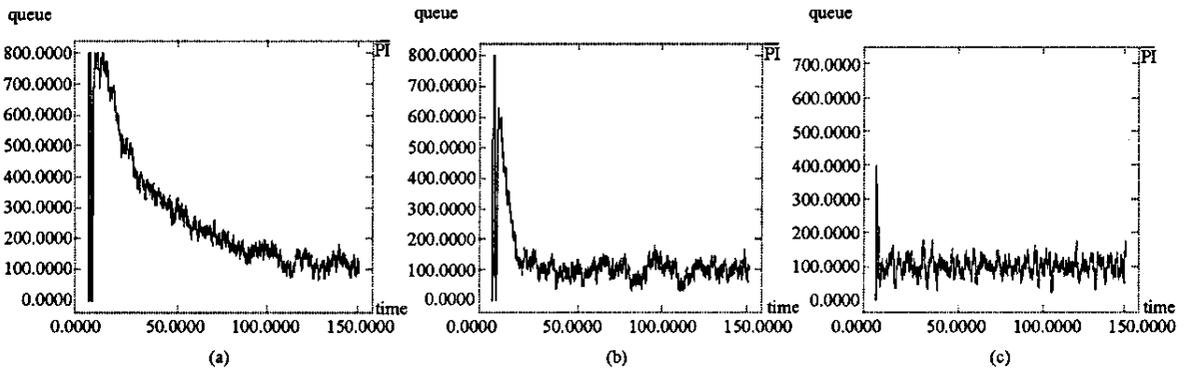


图 3 实验一中参数配置 1、2、3 情况下的队列长度随时间的变化曲线

Fig. 3 Evolution of the queue length with FTP flows in Simulation 1

实验二 考察网络负载变化时三种参数配置的表现。在时间 $0 \sim 4s$ 内创建 200 个 FTP 连接, 在时间 $50 \sim 52s$ 间 100 个 FTP 连接退出, 在时间 $100 \sim 102s$ 间 100 个 FTP 连接重新加入。图 4 (a)(b)(c) 分别是在参数配置 1、2、3 情况下的队列长度随时间的变化曲线图。由图中可以看出: 使用参数配置 1 时, 队列长度收敛得很慢, 始终不在参考点附近(图 4(a)); 使用参数配置 2 时, 队列长度可以在负载变化前收敛到参考点附近(图 4(b)); 使用参数配置 3 时, 队列长度在连接退出和加入时, 一直稳定在参考队列长度附近(图 4(c))。图 5 是三种参数配置在实验二中的链路利用率, 当负载减小时, 前两种参数配置的队列长度经常为空, 链路利用率比较低。仿真实验的结果表明, 使用参数配置 3 提高了 PI 控制器的动态性能, 同时在负载变化时队列长度能较好的稳定在参考点附近, 并保持较高的链路利用率。

实验三 验证 PI 算法在三种参数配置下的报文丢弃率。实验结果表明, 参数配置 3 在试验中的报文丢失率最低, 参数配置 1 的报文丢失率最高, 见表 1 (实验中采用了报文丢弃策略, 如果采用报文标记策略, 这三种参数配置在稳态时对 TCP 流的报文丢弃率接近于零)。

4 讨论

一些文献中指出, PI 控制器保证了稳态误差, 但从控制理论的角度分析, 它使系统的响应速度减慢^[5]。文章将对引入比例积分环节后系统的响应速度进行讨论。

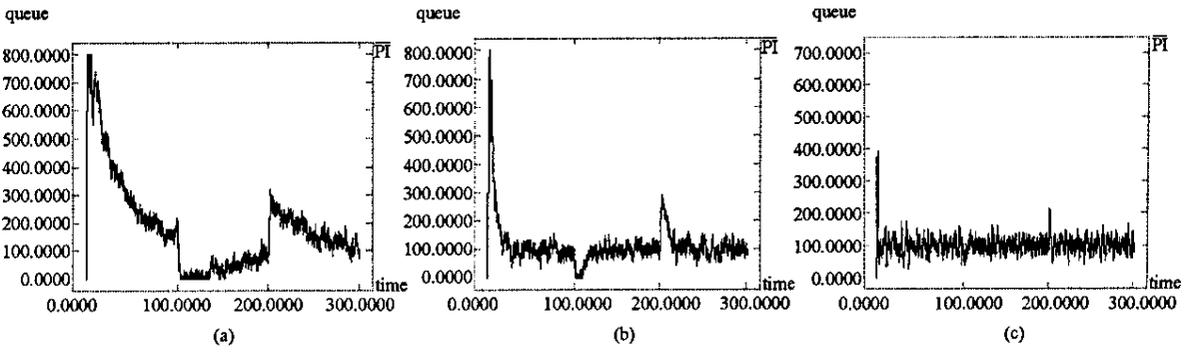


图 4 实验中参数配置 1、2、3 情况下的队列长度随时间的变化曲线

Fig.4 Evolution of the queue length with FTP flows in Simulation 2

表 1 报文丢弃率

Tab.1 Packet drops ratio

	$N = 50$	$N = 200$
参数配置 1	4.59%	12.4%
参数配置 2	4.55%	12.3%
参数配置 3	4.54%	12.3%

在系统校正中单纯引入积分环节, 无论怎么选择参数, 响应都会变慢; 但引入 PI 控制器后, 由于它有比例系数和积分系数两个可调参数, 适当加以选择就可能使系统既稳定又具有较好的

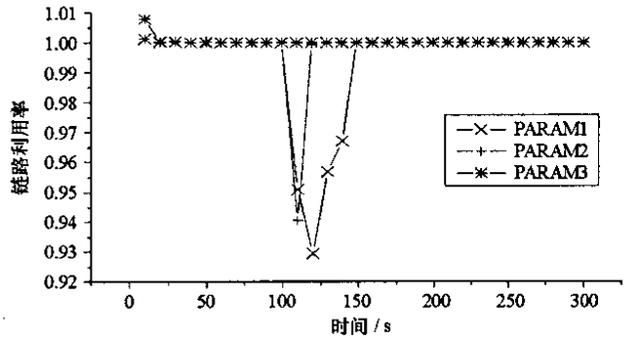


图 5 链路利用率

Fig.5 The link utilization

静态和动态性能, 在很多情况下可以加速系统的响应。例如: 若系统的开环传递函数为 $\frac{K_0}{T_0s + 1}$, 则闭环传递函数为 $H(s) = \frac{K_0}{T_0s + 1 + K_0}$, 当 $K_0 \gg 1$ 时, 系统的过渡时间 $t_s \approx 3T_0/K_0$ 。引入 PI 控制器, $K_p(1 + 1/T_1s)$, 选取 $K_p > 1, K_pK_0 \gg 1, T_1 \gg T_0/(K_pK_0)$, 闭环系统传递函数为: $H(s) \approx \frac{1}{T_0s/(K_pK_0) + 1}$, 此时, 系统在阶跃信号作用下没有静态误差, 并且过渡时间 $t \approx 3T_0/(K_pK_0)$, 小于原过渡时间, 因此, 加入 PI 控制器不一定使响应变慢, 它与参数配置密切相关。

文章中的参数是由网络负载、链路容量、回返时间来计算的, 同时忽略了模型的延时环节。因此, 设计自适应的控制器来降低参数对网络状态的依赖, 以及设计预测控制器克服延时环节的影响是我们今后需要研究的方向。

参考文献:

[1] Floyd S, Jacobson V. Random Early Detection Gateways for Congestion Avoidance[J]. IEEE/ACM Transactions 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[A]. Proceedings of INFOCOM 2001[C], Anchorage, Alaska, 2001: 1726-1734.

[3] Le L, Aikat J, Jeffay K, et al. The Effects of Active Queue Management on Web Performance[A]. Proceedings of ACM SIGCOMM 2003[C], Karlsruhe, Germany, 2003.

[4] 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 ACM SIGCOMM 2000[C], Stockholm, Sweden, 2000.

[5] 章森, 吴建平, 林闯. P²I: 一种新的主动队列管理算法[J]. 计算机学报, 2003, 26(10).

[6] 张明杰, 朱培栋, 卢锡城. DSOCC: 面向区分服务的拥塞控制[J]. 国防科技大学学报, 2004, 26(3).

