

基于自适应模糊控制的 Web 带宽区分服务

高昂¹ 慕德俊¹ 胡延苏¹ 潘文平²

(西北工业大学自动化学院 西安 710072)¹ (南京航空航天大学自动化学院 南京 210016)²

摘要 基于线程/进程分配的 Web 区分服务策略仅仅关注连接延迟,而在服务器带宽受限的情况下,由于处理延迟成为客户端总延迟的主导部分,因此区分效果很差。提出的基于带宽调节的区分服务策略,通过两级自适应模糊控制,调整服务于不同优先级请求的虚拟主机的带宽配额,从而控制处理延迟,实现比例延迟保证。经稳定性分析与实验验证,这种方法取得了良好的效果,相对于静态模糊控制,其延迟比与期望值的方差减少了 40%。

关键词 区分服务,自适应模糊控制,比例延迟保证

中图法分类号 TP393 文献标识码 A

Bandwidth Differentiated Service in Web Server Based on Self-tuning Fuzzy Control

GAO Ang¹ MU De-jun¹ HU Yan-su¹ PAN Wen-ping²

(College of Automation, Northwest Polytechnical University, Xi'an 710072, China)¹

(College of Automation, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)²

Abstract The strategy of differentiated service based on process allocation only concerns connection delay, but when the bandwidth of Web server is over-subscribed, such method is ineffective because the processing delay becomes the dominant factor of client-perceived delay. This paper proposed a QoS strategy using two-level fuzzy control, by adjusting the bandwidth of virtual hosts that sever the different priority requests, the self-tuning fuzzy controller could affect the processing delay and achieve the proportional delay guarantee. The stability analysis and experimental results demonstrate that the proposed approach achieves a better performance in terms of the proportional delay variance to the target. The self-tuning fuzzy controller outperforms the static fuzzy controller by 40% on average.

Keywords Differentiated service, Self-tuning fuzzy control, Proportional delay guarantee

区分服务(DiffServ)实际上是对瓶颈资源的调度问题^[1],对于 Web 服务器,最大带宽与最大并发线程个数往往是整个系统的瓶颈,也是影响延迟的主要因素。现有基于反馈理论的 Web QoS 研究和仿真都是采用线程/进程分配(Process allocation)的策略^[2-5],而忽视了带宽对延迟的影响。

本文在带宽受限(over-subscribed)的情况下,根据服务品质协议(SLA, service Level Agreement)和客户的延迟比,调整相应虚拟主机的带宽配额,保证高优先级用户低延迟而不同优先级用户的延迟比保持不变。

1 线程/进程分配的 QoS 策略的缺陷

图 1(1)所示为 HTTP1.1 协议 Pipeline 方式下客户端和服务器的交互过程,①是 TCP 连接建立过程(实验中小于 0.5ms);②是连接延迟,即 HTTP 请求进入队列,等待空闲服务线程/进程的时间;③是处理延迟,主要是页面传输时间,包括 URL 的解析时间以及组成页面的多个嵌入文件(Embedded objects)的传输时间;④TCP 连接关闭。直接影响用户延迟的主要是连接延迟和处理延迟,即 $l_i = w_i + s_i$,其中 l_i , w_i , s_i 分别表示第 i 类请求的总延迟、连接延迟(即排队时间)

和处理延迟(即服务时间)。目前使用的反馈控制^[2-5]及基于排队论的预测控制^[6,7]都是采用线程/进程分配的 QoS 策略,即调整服务不同优先级请求的线程/进程数量,从而控制连接延迟,达到比例延迟保证。但是当服务器带宽受限时,页面的传输时间将大大增加,此时系统的处理延迟相对于连接延迟不可忽略。

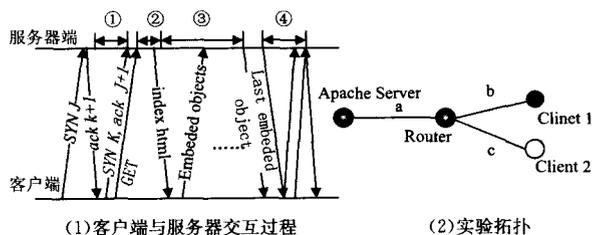


图 1 客户端与服务器交互过程与实验拓扑

首先通过实验说明基于线程/进程分配的 QoS 策略的缺陷。图 1(2)所示为实验拓扑,Apache 2(Httpd -ver. 2. 2. 63)为 client 1 和 client 2 两类客户服务(在本文以下的实验中,均根据 IP 对客户分类),负载由 SURGE^[8]产生,通过 PI 控制器调整服务不同客户的线程配额^[2]。为说明带宽对其的影响,

到稿日期:2009-08-05 返修日期:2009-10-09

高昂(1984-),男,博士生,主要研究方向为网络 QoS 控制、网络化控制, E-mail: snailgao@gmail.com;慕德俊(1963-),男,教授,博士生导师,主要研究方向为计算机网络、信息安全;胡延苏(1985-),女,博士生,主要研究方向为网络化控制;潘文平(1980-),男,博士,主要研究方向为网络拥塞控制、网络 QoS 控制。

实验进行两次,分别对应链路 a 带宽 50Mbps 和 2Mbps 的情况, b, c 为非瓶颈链路, 带宽 100Mbps。

1) 当服务器带宽为 50Mbps 时, 服务器端并发 100 个服务线程, 负载为 200 个模拟客户 (UE, User Equivalents), 结点 1, 2 各 100 个, 此时, 系统的瓶颈资源是服务线程, 控制器分配给高优先级更多的线程配额, 使其获得更小的延迟。如图 2(1) 所示, 控制器在 220s 开启, 当服务于 Class 1 的线程配额增加到 65 时 (见图 2(1)(b)), 两者的延迟比增加为 2 并稳定 (图 2(1)(a)), 此时两类客户的处理延迟基本相同 $s_1 = s_2 = 100\text{ms}$ (见图 2(1)(c)), 而对应的连接延迟差别明显, $w_1 = 50\text{ms}, w_2 = 250\text{ms}$ (图 2(1)(d)), 因此总延迟比 $l_2/l_1 = 350/150 \approx 2.3$ 。可见基于线程/进程分配的 QoS 策略是有效的;

2) 当服务器带宽为 2Mbps 时, 服务器端并发 200 的服务线程, 负载仍为 200 个模拟客户, 结点 1, 2 各 100。此时无论两类线程的配额如何变化 (见图 2(2)(b)), 其处理延迟大大增加 $s_1 = s_2 = 1500\text{ms}$ (见图 2(2)(c)), 而连接延迟几乎为零 $w_1 = w_2 = 0\text{ms}$ (见图 2(2)(d))。由于带宽是瓶颈资源, 延迟主要是服务线程/进程进行文件传输的处理延迟, 而不再是等待空闲线程/进程的连接延迟。此时线程/进程分配的 QoS 策略失效 (见图 2(2)(a))。

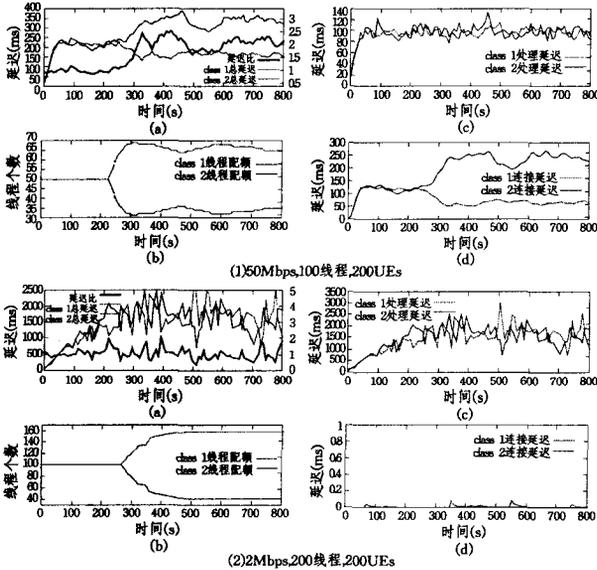


图 2 10M 和 2M 带宽对比试验结果

由此可见, 尽管基于线程/进程分配的 QoS 策略可以实现比例延迟保证, 但它仅仅是对连接延迟进行调整, 对处理延迟没有控制能力。而在实际应用中, 往往由于巨型文件的传输消耗了大量的带宽并且长时间占用线程/进程资源, 使得带宽成为系统的瓶颈, 此时总延迟中处理延迟贡献较大, 针对连接延迟进行控制的策略已经无法影响总延迟, 因此本文对带宽受限的 Web 服务器上的 QoS 问题进行研究, 通过两级模糊控制器, 在系统模型不可知的情况下, 实现基于带宽调节的比例延迟区分服务。

2 基于带宽调节的区分服务

一般而言, 处理延迟与页面大小成正比, 与带宽成反比^[9], 即 $s_i \propto \text{size}_i / b_i$, 图 3(1) 为基于带宽调节的区分服务模型。根据 SLA 将用户请求在终端路由器分为 $N (N \geq 2)$ 类, 并重定向到 Web 服务器不同的端口, 每个端口对应一台虚拟

主机, 通过给每台虚拟主机配置不同的可用带宽 b_i , 来控制不同优先级请求的文件处理时间, 考虑到主机物理带宽为 W , 则有约束条件:

$$\sum_{i=1}^N b_i(k) = W \quad (1)$$

$$\begin{cases} \delta_{i+1} = Y_{i, \text{desire}}, i=1, \dots, N-1 \\ y_i(k) = \frac{l_{i+1}(k)}{l_i(k)} \\ y_i(k) = Y_{i, \text{desire}} \end{cases} \quad (2)$$

式中, δ_i 是由 SLA 保证的固有优先级, δ_i 越小, 优先级越高, $Y_{i, \text{desire}}$ 是由此确定的延迟比的期望值; $y_i(k)$ 是由 QoS 观测器得到的相邻优先级请求的实测延迟 $l_i(k)$ 之比。为了保证式 (2) 成立, 实现比例延迟保证, 本文设计了如图 3(2) 所示的自适应模糊控制器 (STFC, Self-Tuning Fuzzy Controller), 它由带宽分配 (BA, Bandwidth Allocation) 和缩放自适应 (SF, Scaling Factor) 两级模糊控制器组成^[10], 其中带宽分配针对被控模型的不确定性, 通过模糊推理对主机带宽进行调度, 而缩放自适应则用于补偿由页面大小差异导致的处理延迟的抖动。

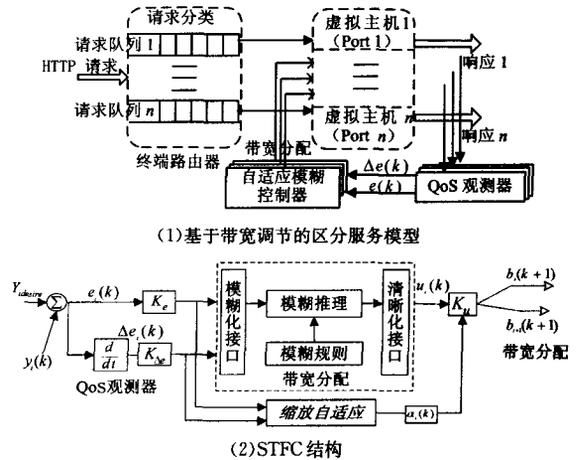


图 3 基于带宽调节的区分服务模型与 STFC 结构

控制器实际的输入为 $K_e e(k)$ 和 $K_{\Delta e} \Delta e(k)$, 实际的输出为 $\alpha_i(k) K_u u_i(k)$, 其中 $K_e, K_{\Delta e}$ 为比例因子, K_u 为量化因子, $\alpha_i(k)$ 为缩放因子, 采样周期为 T , 根据第 k 个采样时刻的偏差 $e_i(k) = Y_{i, \text{desire}} - y_i(k)$, 以及偏差的变化率 $\Delta e_i(k)$, 计算出需要调整的带宽变化量 $\alpha_i K_u u_i(k)$, 因此有:

$$\begin{cases} \bar{b}_i(k+1) = b_i(k) + K_u \alpha_i(k) u_i(k) \\ \bar{b}_{i+1}(k+1) = b_{i+1}(k) - K_u \alpha_{i+1}(k) u_{i+1}(k) \end{cases} \quad (3)$$

式中, $\bar{b}_i(k+1)$ 是控制器给出的下一个采样周期带宽的期望值, $b_i(k+1)$ 表示实际能为其分配的带宽, 由于存在约束式 (1), 并且关心的只是延迟的相对关系, 因此只要求实际分配的带宽之比 $m_i(k+1)$ 与期望值之比一致:

$$\frac{\bar{b}_{i+1}(k+1)}{b_i(k+1)} = \frac{b_{i+1}(k+1)}{b_i(k+1)} = m_i(k+1) \quad (4)$$

将式 (4) 代入式 (1), 有:

$$\left(1 + \prod_{i=1}^{N-1} \prod_{j=1}^{N-1} m_j^{-1}(k+1)\right) b_N(k+1) = W \quad (5)$$

据此可求得 $b_i(k+1), i=1, \dots, N-1$ 作为 $k+1$ 时刻各个虚拟主机的带宽配额。

3 控制器分析与设计

3.1 带宽分配

语言变量 E, EC, U 分别用来描述 $e_i(k), \Delta e_i(k)$ 和 $u_i(k)$, 都有 7 个语言变量值: $\{NL, NM, NS, ZE, PS, PM, PL\}$, 其具体含义如图 4(1) 所示, 隶属度函数为双端高斯型 (gauss2mf), 基本论域分别为 $[-|e_{\max}|, |e_{\max}|], [-|ec_{\max}|, |ec_{\max}|], [-|u_{\max}|, |u_{\max}|], K_e = \frac{7}{|e_{\max}|}, K_{\Delta e} = \frac{7}{|ec_{\max}|}, K_u = \frac{7}{|u_{\max}|}$. \tilde{B}_i 表示虚拟主机 i 在满足式(2)时应该获得的带宽配额; $\tilde{u}_i(k)$ 表示 \tilde{B}_i 与实际带宽 $b_i(k)$ 之差:

$$\tilde{u}_i(k) = \tilde{B}_i - b_i(k) \quad (6)$$

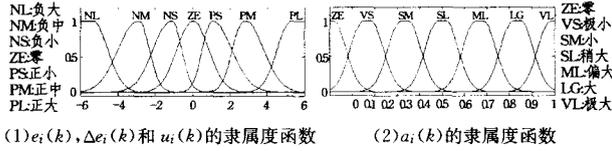


图 4 自适应模糊控制器的隶属度函数

为了实现比例延迟区分, 要求控制器输出 $u_i(k)$, 校正这种偏差, 保证 $b_i(k)$ 收敛于 \tilde{B}_i [11], 所以:

$$U = {}^n u_i(k) = {}^r \tilde{u}_i(k) \quad (7)$$

将系统的脉冲响应划分为图 5(1) 所示的 5 个区域, 根据 $e_i(k), \Delta e_i(k)$ 所处的区域的特性, 确定相应的模糊规则。

1) 在区域①和③, $e_i(k), \Delta e_i(k)$ 异号, 此时偏差 $e_i(k)$ 具有自校正性, $y_i(k)$ 向着 $Y_{i, \text{desire}}$ 的方向移动, $u_i(k)$ 应取较小值, 避免超调;

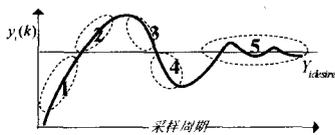
2) 而在区域②和④, $e_i(k), \Delta e_i(k)$ 同号, 此时 $y_i(k)$ 向背离 $Y_{i, \text{desire}}$ 的方向移动, $u_i(k)$ 应取较大值, 抑制这种趋势;

3) 在区域⑤, $|e_i(k)|, |\Delta e_i(k)|$ 都很小, 系统稳定, $u_i(k)$ 微调。

据此, 得到模糊规则表 (如图 5(2) 所示) 及模糊规则基 F :

$$F = \{F^1, \dots, F^L\}, L = 49$$

$$F^l: \text{If } E \text{ is } A_l, \text{ and } EC \text{ is } B_l, \text{ thus } U \text{ is } C_l, l = 1, 2, \dots, L$$



(1) 脉冲响应

U		EC						
		NL	NM	NS	ZE	PS	PM	PL
E	NL	NL	NL	NL	NL	NM	NS	ZE
	NM	NL	NL	NL	NM	NS	ZE	PS
	NS	NL	NL	NM	NS	ZE	PS	PM
	ZE	NL	NM	NS	ZE	PS	PM	PL
	PS	NM	NS	ZE	PS	PM	PL	PL
	PM	NS	ZE	PS	PM	PL	PL	PL
	PL	ZE	PS	PM	PL	PL	PL	PL

(2) 带宽分配规则表

A		EC						
		NL	NM	NS	ZE	PS	PM	PL
E	NL	NL	VL	VL	SM	VS	VS	ZE
	NM	VL	VL	LG	SL	SM	SM	SM
	NS	VL	VL	LG	ML	ZE	NS	NM
	ZE	LG	ML	SL	ZE	SL	ML	LG
	PS	SL	SM	VS	ML	LG	LG	VL
	PM	SM	SM	SM	SL	LG	VL	VL
	PL	ZE	VS	VS	SM	VL	VL	VL

(3) 缩放自适应规则表

图 5 STFC 控制规则

式中, A_l, B_l, C_l 为第 l 条规则中与 E, EC, U 对应的语言值, 假设 $\mu_{C_l}(u^l) = 1$, 即 C_l 的隶属函数在 u^l 点取得最大值。采用单点模糊化、最小值推理和中心平均加权反模糊化的模糊逻辑, 得到精确结果:

$$u(k) = \frac{\sum_{l=1}^L u^l [\min(\mu_{A_l}(e(k)), \mu_{B_l}(\Delta e(k)))]}{\sum_{l=1}^L \min(\mu_{A_l}(e(k)), \mu_{B_l}(\Delta e(k)))} \quad (8)$$

式中, $\mu_{A_l}(e(k))$ 表示 $e(k)$ 隶属于语言变量值 A_l 的程度。

3.2 缩放自适应

通过调节缩放因子 $\alpha_i(k)$, 缩放自适应对于由于页面大小差异产生的抖动进行补偿。用语言变量 A 描述 $\alpha_i(k)$, 其论域为 $[0, \alpha_{\max}]$, $\alpha_{\max} = 1$, 有 7 个语言变量值 $\{ZE, VS, SM, SL, ML, LG, VL\}$, 其具体含义及隶属度函数 (gauss2mf 型) 如图 4(2) 所示, 其模糊规则表如图 5(3) 所示, 分为 5 个区域。

1) 当 $|e_i(k)|$ 较大时, 如果 $e_i(k)$ 与 $\Delta e_i(k)$ 同号, $\alpha_i(k)$ 取大值加强对偏离趋势的抑制作用 (区域①); 如果 $e_i(k)$ 与 $\Delta e_i(k)$ 异号, $\alpha_i(k)$ 取小值, 在不牺牲带宽分配灵敏度的情况下, 减少超调量和调节时间 (区域②);

2) 由于 Web 页面大小的差异以及负载的高动态特性, 当系统处在平衡点附近而产生扰动时, $|e_i(k)|$ 较小, 如果 $|\Delta e_i(k)|$ 较大且与其同号, 说明系统由平衡点加速偏离期望值 $Y_{i, \text{desire}}$ (区域④), $\alpha_i(k)$ 应取较大值对其补偿; 如果 $|\Delta e_i(k)|$ 较小且与其异号, 说明系统趋向于 $Y_{i, \text{desire}}$, $\alpha_i(k)$ 应取较小值, 避免超调 (区域③);

3) 当 $|e_i(k)|, |\Delta e_i(k)|$ 都很小时, $\alpha_i(k)$ 在零值附近微调 (区域⑤)。

同样采用最小值推理和中心平均加权反模糊化的模糊逻辑, 可以得到 $\alpha(k)$ 的精确结果。

由于模糊规则是通过带宽资源与延迟比之间的关系分析得到的, 避免了对被控对象的精确建模, 因此, 自适应模糊控制可认为是模型无关的。

3.3 稳定性分析与性能评价

以两类客户为例论证系统的稳定性, 即 $N=2$, 由式(3), 式(4)可以得到:

$$\tilde{b}_i(k+1) = b_i(k+1), i = 1, 2 \quad (9)$$

由式(3), 式(6), 式(9)可以得到:

$$\tilde{u}_i(k+1) = \tilde{u}_i(k) - K_u \alpha_i(k) u_i(k) \quad (10)$$

取 Lyapunov 函数 [12]:

$$V[\tilde{u}_i(k)] = \tilde{u}_i^2(k) \quad (11)$$

则:

$$\begin{aligned} \Delta V[\tilde{u}_i(k)] &= V[\tilde{u}_i(k+1)] - V[\tilde{u}_i(k)] \\ &= \tilde{u}_i^2(k+1) - \tilde{u}_i^2(k) \\ &= [\tilde{u}_i(k) - K_u \alpha_i(k) u_i(k)]^2 - \tilde{u}_i^2(k) \\ &= K_u \alpha_i(k) u_i(k) [K_u \alpha_i(k) u_i(k) - 2\tilde{u}_i(k)] \end{aligned} \quad (12)$$

由式(7), $u_i(k)$ 和 $\tilde{u}_i(k)$ 同号, $K_u \alpha_i(k)$ 恒为正, 因为 \tilde{B}_i 和 $\tilde{u}_i(k)$ 在实际中很难精确得到, 取任意小正整数 ϵ , 当 $|\tilde{u}_i(k)| > \epsilon$ 时, 如果

$$|u_i(k)| < 2\epsilon / K_u \alpha_i(k) \quad (13)$$

则 $|u_i(k)| < \frac{2|\tilde{u}_i(k)|}{K_u \alpha_i(k)}$, 式(12)负定, $b_i(k)$ 收敛于 $\tilde{B}_i(k)$ 的 ϵ 邻域。

定义相对方差 [11]:

$$R_i(e) = \frac{\sqrt{\sum_{k=1}^n e_i(k)^2 / n}}{Y_{i, \text{desire}}} \quad (14)$$

评价控制系统的性能, $R_i(e)$ 的值越小, 说明控制器对于

Class $i+1$ 与 Class i 延迟比稳定在 $Y_{i,desired}$ 的控制能力越强。

4 实验验证与分析

同样采用图 1(2)所示的拓扑结构,物理主机带宽设置为 2Mbps,服务器端同样并发 200 的服务线程,负载也各为 100 个模拟客户,服务器端生成大小服从重尾分布 90054 个页面文件。图 6 所示为所有请求页面大小的概率密度分布,可见,绝大部分页面都不超过 10kbyte,且分布具有重尾特性,符合网络通讯实际情况。

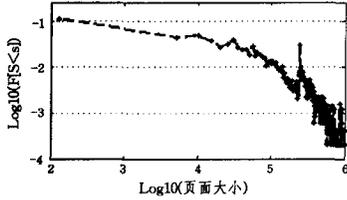


图 6 页面大小概率分布

配置修改过的 mod_bw 模块提供基于带宽的区分服务。 $Y_{1,desired} = \delta_2/\delta_1 = 2$, 所以限制 $e_{max} = 2, ec_{max} = 2$, 取 $u_{max} = 25000$, 即在一个采样周期,带宽最多可以调整 25kbps,为了验证基于带宽调节的区分服务模型以及控制器的有效性,实验共进行 3 次,分别对应自适应模糊控制, $T=15s$; 自适应模糊控制, $T=25s$ 以及静态模糊控制 (SFC, Static Fuzzy Control), $T=25s$ 。以图 7(2)为例加以说明。

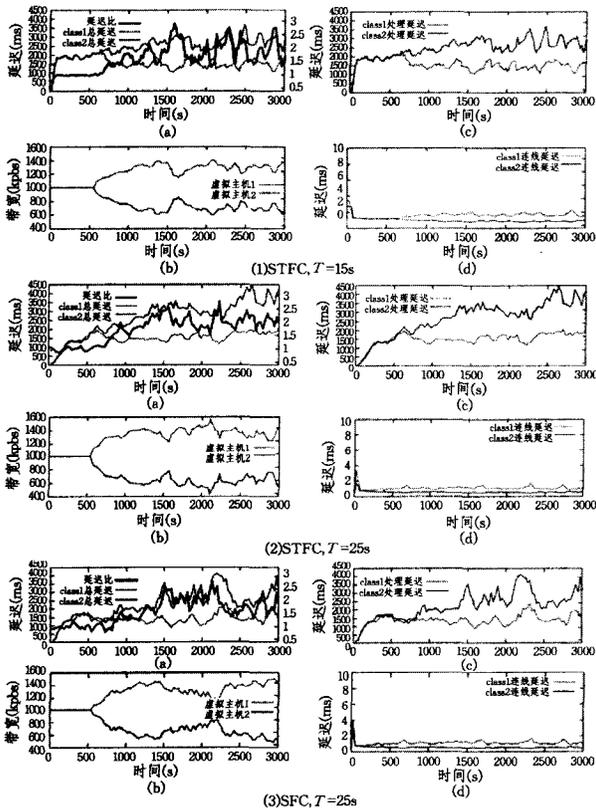


图 7 实验结果对比

1) 前 500s 控制器关闭,两台虚拟主机带宽均为 1Mbps,此时,相应的总延迟基本一致,无区分服务可言;500s 之后,在控制器的作用下,不同虚拟主机的带宽配额随之调整(见图 7(2)(b)),对应的总延迟之比也基本稳定在 $Y_{1,desired}$ 附近(见图 7(2)(a));

2) 造成总延迟差异的主要因素是处理延迟(见图 7(2)(c)),两类请求的连接延迟(见图 7(2)(d))差异不大,均小于 1ms,相对于处理延迟可以忽略。对比第 1 节图 2(2)的实验,在带宽受限的情况下,通过调节带宽控制处理延迟,可以实现比例延迟保证,因此基于带宽调节的区分服务模型是合理有效的;

3) 比较 500s 前后(控制器开启)的处理延迟 s_i^{-500} 和 s_i^{+500} , 显然有:

$$s_2^{-500} - s_2^{+500} \geq s_1^{-500} - s_1^{+500} \quad (15)$$

亦即,低优先级 Class 2 处理延迟增加的程度要大于高优先级 Class 1 处理延迟减少的程度。

考虑到采样周期对于控制器的影响,对比图 7(1)和图 7(2),虽然不同采样周期下,均可实现区分服务,但是在 $T=15s$ 的情况下,延迟比 $y_i(k)$ 抖动明显,按式(15)计算控制器开启后的相对方差, $R_{1,STFC,15s}^2(e) = 0.2362, R_{1,STFC,25s}^2(e) = 0.1870$,这主要是由于个别巨型文件的存在导致了处理延迟抖动,而这种抖动在较小采样周期下表现得更为明显,因此本文在 25s 采样周期下比较自适应模糊控制(见图 7(2))和静态模糊控制(见图 7(3))。当输出自适应关闭时,自适应模糊控制蜕化为静态模糊控制,量化因子 K_u 取为前者的 $1/3^{[12]}$,比例因子不变,即 $K_u^{SFC} = \frac{1}{3} K_u^{STFC}, \alpha^{SFC}(k) \equiv 1, K_e^{SFC} = K_e^{STFC}, K_{\Delta e}^{SFC} = K_{\Delta e}^{STFC}$, 此时依然可以实现比例延迟保证,调解时间也相差不大,但是系统的抖动较明显, $R_{1,STFC,25s}^2(e) = 0.2608, (R_{1,STFC,25s}^2 - R_{1,SFC,25s}^2)/R_{1,STFC,25s}^2 = 40\%$,可见自适应模糊控制在页面大小重尾变化的情况下较好地保持了系统的性能。

结束语 本文通过分析实验指出了基于线程/进程分配的 QoS 策略在带宽受限时的局限性,建立了基于带宽调节的区分服务策略并设计了二阶模糊控制器来自适应调节服务于不同优先级请求的虚拟主机的带宽配额,以实现比例延迟保证。最后,通过实验,验证了模型的有效性。如何将两种控制 QoS 策略结合以适应各种网络状况,将成为下一步工作的重点。

参考文献

- [1] Dovrolis C, Stiliadis D, Ramanathan P. Proportional differentiated services: delay differentiation and packet scheduling[C]// Proceedings of the ACM SIGCOMM, 1999
- [2] Pan W, Mu D, Wu H, et al. Proportional Delay Differentiation Service in Web Application Servers: A Feedback Control Approach[C]// ISECS International Colloquium on Computing, Communication, Control and Management, Vol. 1, 2008; 600-604
- [3] Zhou X, Cai Y, Godavari G K, et al. An Adaptive Process Allocation Strategy for Proportional Responsiveness Differentiation on Web Services[C]// IEEE 2nd Int'l Conf. on Web Services (ICWS), 2004
- [4] Zhou X, Cai Y, Chow E. An integrated approach with feedback control for robust Web QoS design[J]. Computer Communications, 2006, 29(16): 3158-3169
- [5] Wei J, Zhou X, Xu C Z. Robust Processing Rate Allocation for Proportional Slowdown Differentiation on Internet Servers[J]. Computers IEEE Transactions on, 2005, 54(8): 964-977

(下转第 109 页)

算接收到的数据包的可丢率,STT,PLR,POR等参数样本值,根据IDD和STT值,判断网络是否发生拥塞;根据PLR和POR值判断非拥塞状态下的丢包原因;然后在回复的ACK数据包中用3bits携带网络状态信息,以便明确通知发送端,网络是否发生了拥塞。当发送端接收到3个重复的ACK或RTO超时,得知数据包丢失,便通过这些ACK中携带的状态指示,采取适当的控制措施:如果是网络拥塞,便调用拥塞控制算法,否则只重传丢失的数据包;当发生路径中断时,发送端采用固定RTO重传策略。仿真试验验证了此方案是有效的。

本协议的优势在于它仅在发送端和接收端引入简单的计算和判断算法,没有利用过多的开销,不改变网络中其他各层协议,实现简单。但缺陷是本方案中门限值high,low的设置将影响它的性能。因此,设计合理、简单而有效的端到端TCP改进方案将是一个重要研究方向。

参 考 文 献

[1] Fu Z, Meng X, Lu S. How bad tcp can perform in mobile ad hoc networks[C]// Seventh IEEE Symposium on Computers and Communications(ISCC'02). July 2002;298-303

[2] Holland G, Vaidya N. Analysis of tcp performance over mobile ad hoc networks[C]//Proc. of the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking. ACM New York, NY, USA, 1999;219-230

[3] Chandran K, Raghunathan S, Venkatesan S, et al. A feedback-based scheme for improving tcp performance in ad hoc wireless networks[C]// Proc. of the 18th International Conference on Distributed Computing Systems. Washington, DC, USA. IEEE Computer Society, 1998;472

[4] Chen K, Xue Y, Nahrstedt K. On setting tcp's congestion window limit in mobile ad hoc network[C]// IEEE International Conference on Communications, 2003(ICC '03). May 2003(2); 1080-1084

[5] Dyer T D, Boppana R V. A comparison of tcp performance over three routing protocols for mobile ad hoc networks[C]// Proceedings of the 2nd ACM International Symposium on Mobile Ad hoc Networking & Computing. ACM New York, NY, USA, 2001;56-66

[6] Fu Z, Zerfos P, Luo H, et al. The impact of multihop wireless channel on tcp throughput and loss[C]// INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, 30 March-3 April 2003 (3);1744-1753

[7] Xu K X, Gerla M, Qi L, et al. Enhancing tcp fairness in ad hoc wireless networks using neighborhood red[C]//Proc. of the 9th Annual International Conference on Mobile Computing and Networking. ACM New York, NY, USA, 2003;16-28

[8] EIRakabawy S M, Klemm A, Lindemann C. TCP with adaptive

pacing for multihop wireless networks[C]// Proc. of the 6th ACM International Symposium on Mobile Ad hoc Networking and Computing. ACM New York, NY, USA, 2005;288-299

[9] Nahm K, Helmy A, Jay Kuo C C. Tcp over multihop 802.11 networks: issues and performance enhancement[C]// Proceedings of the 6th ACM International Symposium on Mobile Ad hoc Networking and Computing. ACM New York, NY, USA, 2005;277-287

[10] Zhai H Q, Chen X, Fang Y. Improving transport layer performance in multihop ad hoc networks by exploiting MAC layer information[J]. IEEE Transactions on Wireless Communications, 2007(6);1692-1701

[11] Ding L, Wang X, Xu Y, et al. Vegas-w: an enhanced tcp-vegas for wireless ad hoc networks[C]// Proc. IEEE Int. Conf. on Communications. Beijing, China, 2008;2383-2387

[12] Fu Z, Greenstein B, Meng X, et al. Design and implementation of a tcp friendly transport protocol for ad hoc wireless networks[C]//The 10th IEEE International Conference on Network Protocols. Paris, France, 2002;212-225

[13] Wang F, Zhang Y. Protocols: Improving tcp performance over mobile ad hoc networks with out-of-order detection and response[C]//Proc. of the 3rd ACM International Symposium on Mobile Ad hoc Networking & Computing. ACM New York, NY, USA, 2002;217-225

[14] Wang Q, Pu J, Liu W. Research of slow-start strategy of tcp congestion control[J]. Microelectronics & Computer, 2007, 24(12);210-212

[15] Wu Q, Wu B, Zhang J. Improving tcp performance for mobile ad hoc networks through employing end-to-end identification[J]. Journal of Hefei University of Technology, 2008, 31(9); 1458-1461

[16] Mirhosseini S M, Torgheh F. Improvement of tcp performance in ad hoc networks using cross layer approach[C]// Proc. of the 2008 Third International Conference on Systems and Networks Communications. Washington, DC, USA; IEEE Computer Society, 2008;322-328

[17] Paxson V. Measurements and analysis of end-to-end Internet dynamics[D]. CA, USA; University of California at Berkeley Berkeley, 1998

[18] Niu Z, Xiang Y. End to end congestion control of udp stream for ad hoc networks[J]. Computer Engineering and Design, 2009, 30(6);1301-1306

[19] JiST user guide and swans user guide [EB/OL]. <http://jist.ece.conell.edu/docs.html>, March 2004

[20] Fu Z, Meng X, Lu S. A transport protocol for supporting multimedia streaming in mobile ad hoc networks[J]. IEEE Journal on Selected Areas in Communications, 2003, 21(10); 1615-1626

[21] Handley M, Floyd S, Padhye J, et al. Tcp friendly rate control (TFRC); Protocol Specification[S]. RFC 3448

(上接第86页)

[6] Henriksson D, Lu Y, Abdelzaher T. Improved prediction for web server delay control[C]// Processings of the 16th Euromicro Conference on Real-Time Systems. 2004;61-68

[7] Lu Y, Abdelzaher T, Lu C, et al. Feedback control with queuing-theoretic prediction for relative delay guarantees in web servers[C]//Proceedings of the 9th IEEE Real-Time and Embedded Technology and Applications Symposium. 2003;208

[8] Barford P, Crovella M. Generating Representative Web Workloads for Network and Server Performance Evaluation[J]. ACM SIGMETRICS Performance Evaluation Review, 1998, 26(1); 151-160

[9] Abdelzaher T F, Bhatti N. Web Server QoS Management by Adaptive Content Delivery[C]// Quality of Service, 1999. IWQoS '99. 1999 Seventh International Workshop on. 1999;216-225

[10] Wei J, Xu C Z. eQoS: Provisioning of Client-Perceived End-to-End QoS Guarantees in Web Servers[J]. IEEE Transactions on Computers, 2006

[11] Wei J, Xu C Z. A Self-tuning Fuzzy Control Approach for End-to-End QoS Guarantees in Web Servers[C]// Quality of Service. IWQoS 2005 vol. 3552, 2005;123-135

[12] Diao Y, Hellerstein J L, Parekh S. Using fuzzy control to maximize profits in service level management-IBM[J]. Journal of Research and Development, 2002