

Web集群中基于控制论的分布式QoS量化控制^{*}

王晓川[†], 金士尧, 夏明波

(国防科学技术大学 并行与分布处理国家重点实验室, 湖南 长沙 410073)

Distributed Quantitative QoS Control Based on Control Theory in Web Cluster

WANG Xiao-Chuan[†], JIN Shi-Yao, XIA Ming-Bo

(National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, China)

+ Corresponding author: Phn: +86-731-4512376, E-mail: mrwangxc@163.com, http://www.nudt.edu.cn

Wang XC, Jin SY, Xia MB. Distributed quantitative QoS control based on control theory in Web cluster. Journal of Software, 2007,18(11):2810-2818. <http://www.jos.org.cn/1000-9825/18/2810.htm>

Abstract: This paper proposes a quantitative QoS control mechanism for Web cluster based on control theory. It's featured with distributed control, effective and flexible systematical design, good extensibility, high availability and deployment convenience. Implementation and experiments validate the feasibility and effectiveness of the proposal.

Key words: cluster; Web server; quantitative QoS control

摘要: 基于控制理论提出一种分布式 Web 集群 QoS 量化控制机制,具有分布控制、灵活、有效的系统化设计,高可扩展,高可用及易于部署等特点.系统的实现及实验验证了该方案的可行性和有效性.

关键词: 集群; Web 服务器; 量化 QoS 控制

中图法分类号: TP393 **文献标识码:** A

随着电子商务等大量关键业务在 Web 平台上的广泛部署, Web 服务的 QoS 控制得到越来越多的关注. 单纯以软硬件资源扩张换取高性能服务的做法不仅代价高昂, 而且也不能完全满足需求. 为了充分挖掘软、硬件资源潜力, 获取最大效益, 量化的 QoS 控制具有重大的理论和现实意义. QoS 的量化控制主要是指为特定类型的用户或者服务提供量化的服务性能, 如平均响应时间或者数据传输速率. 量化控制并不等同于提供最好的服务性能, 而在于对性能的定量控制和管理. 灵活、有效的量化 QoS 控制有利于提高资源使用效益, 改善用户体验, 也有利于应用新颖的服务模式, 提高服务效益. 但是, 服务内容的多样性、用户访问的多变性使得 Web 系统具有典型的自相似、非线性、时变等特征^[1], 使得量化的 QoS 控制非常困难. 在大规模、异构 Web 集群中, 这一问题尤其突出.

1 相关工作

近年来, 人们围绕 Web 服务的 QoS 控制进行了大量研究, 提出了资源划分、请求调度、接纳控制、服务内容区分、基于理论模型的控制等典型控制方法. 现有方案面临的主要问题在于:

- 缺乏灵活、有效的量化控制. 很多方案在资源使用份额、调度优先级、请求准入比例或者服务内容编码

^{*} Supported by the Innovation Fund For Technology Based Firms under Grant No.04C26213201155 (国家中小型科技企业创新基金)

Received 2006-05-23; Accepted 2006-10-10

方式等方面,通过控制不同类型请求的系统资源用量控制其服务性能,较高优先级的用户请求获得较大比例的系统资源^[2-4]、较高的调度优先级^[5-7]、较大的请求准入比例^[8-10]或者更为优质的服务内容^[11,12]。由于系统资源的使用量往往缺乏与量化性能指标的直接联系,且通常依靠管理员静态配置,这类方法的适应性较差,不能有效地对服务质量参数进行量化控制。

- 模型失配.与基于直觉和经验的QoS控制方案相比,基于服务系统理论模型的控制方案^[13-17]具有能够进行理论化的分析和推演、易于量化等优点,在近年来的研究中得到了更多的关注.但是,由于Web应用的复杂性和人们对其认识的局限性,这类方案基本上都基于大幅度简化和理想化了的系统模型,经常见到的包括排队论中的G/G/N等排队模型及泊松分布、正态分布等常见分布^[13,14],又如经典控制论中常用的线性系统模型^[15-18];模型的建立严重依赖于研究所使用的环境,与实际当中的系统并不能较好地匹配^[1],有的甚至存在较为严重的失配,影响到实际应用中的效果和适应性。
- 参数设置困难.QoS控制的执行效果与参数设置密切相关,确定一组合适的运行参数非常困难,需要对应用系统的深刻理解和丰富的实际经验,或者需要进行离线的系统识别与参数整定.当应用类型或者负载发生变化时,原有的配置很可能不能满足要求,需要再次配置,非常麻烦,增大了应用的难度。
- 实际应用困难.很多已有方案的一个突出问题是,需要对相关软件进行修改甚至推倒重来,这在实际当中是不现实的,给系统的维护与升级带来了巨大的困难。

从上述分析我们也可以看出,一个较好的量化 QoS 控制方案不仅需要较好的有效性和适应性强的系统化设计,而且应该对实际应用友好,具有较强的兼容性,易于部署和使用.值得注意的是,集群网络服务的 QoS 量化控制研究还很少,大规模异构集群的量化 QoS 控制更是具有挑战性的问题。

随着控制理论在工业领域中取得了巨大的成功,人们在最近的研究中也尝试着将控制论,主要是经典控制论的方法引入到Web服务QoS控制当中来^[15-18].虽然已有的研究结果也不同程度地存在着上述问题,但控制理论在复杂工业控制领域的成功启示我们,控制论的理论和方法对于量化QoS控制仍具有重大的指导意义.由于网络应用的复杂性,为网络服务系统建立一个具有较强通用性的解析模型是非常困难的.已有的基于经典控制理论的研究主要基于系统的线性模型,并且要求对系统模型进行显式刻画,对于复杂的网络服务系统,在有效性和灵活性方面仍有不足.此外,复杂的系统识别与参数整定过程也直接影响到了这类方法的易用性.而现代控制论的方法和技术为我们解决网络服务QoS的控制问题提供了一条全新的思路.总体看来,将现代控制理论引入到网络应用QoS定量控制的研究中,将是一个重要的方向。

2 集群中基于控制论的分布式 QoS 量化控制

2.1 分布式系统结构

Web 集群系统通常由两部分组成:前端集群入口(本文中以分发器表述)和后端执行服务器群.前者为用户访问服务的统一入口,后者完成实际的请求处理.现有绝大多数 Web 集群系统仅有一台分发器,为消除入口单点故障,可增加一台备份分发器;为了避免分发器控制算法设计的复杂性,任何时刻仅由主分发器承担入口节点的工作.考虑到单分发器容易成为大规模集群系统的性能瓶颈,本文提出具有多个分发器的 Web 集群结构,如图 1 所示:多个分发器具有独立的外部 IP 地址(通过动态 DNS 解析向外公布)和配置相同的集群服务,可同时接收和处理用户请求;用户请求经由分发器到达执行服务器(箭头 1、箭头 2);为保持集群结构对应用系统的友好性,避免对已有软件系统的改造,服务器响应仍经由分发器返回(箭头 3、箭头 4);分发器两两互为备份,失效一方的 IP 地址将由另一方接管;为实施 QoS 控制,分发器实现为应用层网关,对用户请求进行分类和管理.这样的集群系统结构在保持集群分层结构的同时,不仅消除了系统的单点故障点,也消除了单点性能瓶颈.结合下文给出的分布式 QoS 控制机制,这将是一种完全分布式的集群结构,在可扩展性及可用性方面具有很大优势。

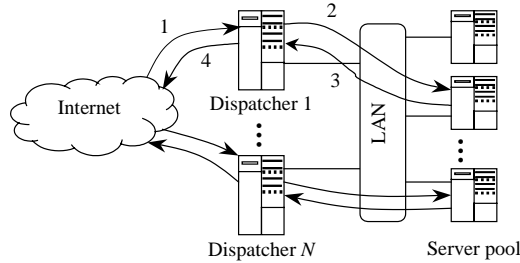


Fig.1 Architecture of distributed Web system

图1 分布式 Web 系统体系结构

2.2 基于控制论的分布式 QoS 控制

传统 Web 集群 QoS 的控制机制一般在前端分发器上实现.在大规模甚至异构集群环境下,精确的 QoS 控制需要不小的计算开销,而分发器要执行应用层网关的工作,负载已十分繁重,这极大地影响到了系统的可扩展性;为了得到更高的执行效率,分发器软件一般都实现为操作系统内核代码甚至使用专门的硬件加速,而在内核中,能够使用的计算设施十分有限,一些内核甚至连浮点操作都不支持,这也极大地阻碍了控制机制的设计与部署.所以,在传统集群 QoS 控制框架下,很难设计和实施以控制论为基础的 QoS 控制.为此,本文提出了全新的以控制论为基础的分布式集群 QoS 控制方案.

从控制论的观点看,一个典型的反馈控制系统包含 3 个基本组成部分:观测器(monitor)、控制器(controller)和激励器(actuator)^[19].观测器负责采集受控系统的运行信息;控制器根据观测器采集的数据、控制模型及控制目标在线推演下一步的控制动作(在实际系统中,还可能包括受控系统的在线估计器);激励器则是控制动作的实施者,直接作用于受控系统.整个反馈控制过程如图 2 所示.

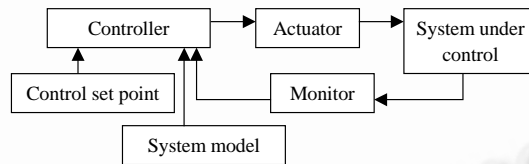


Fig.2 A typical feedback control system

图2 典型的反馈控制系统

基于此框架,分布式的 QoS 控制系统如图 3 所示.为简化起见,我们仅给出了 1 台分发器与 1 台服务器之间的关系图,虚线表示控制流,实线表示虚拟服务的数据流.对于某一虚拟集群服务来说,可能包括物理系统中若干台分发器和服务器:观测器分布于虚拟服务所有的服务器和分发器之上,前一类收集本服务器负载信息并检测本地服务的可用性,后一类收集虚拟服务的 QoS 相关信息;虚拟服务在其每一台服务器上有一个 QoS 控制器,同一服务器上可有多个不同虚拟服务的控制器,各个控制器互相独立,甚至可以采用不同的控制模型或算法;激励器位于虚拟服务的每一台分发器上,根据来自各个控制器的指令作用于应用层网关.分发器与服务器上的控制器之间为多对多的映射关系,一个控制器可对应多台分发器,一台分发器也对应着多台服务器上的控制器.QoS 控制的实施,以三元组(分发器,控制器,虚拟服务用户请求类型)为基本单元:虚拟服务经由分发器 D 到达服务器 S 进行处理的某类用户请求,其 QoS 相关数据受 D 上的观测器监测并报告给 S 上该虚拟服务的 QoS 控制器,控制器结合本地信息为 D 计算出该类请求相应的反馈控制动作并将控制指令发送给 D 上的激励器,激励器通过应用层网关,将反馈动作实施于由 D 到 S 的该类用户请求,从而完成整个反馈回路.由此,全系统各个虚拟服务各类用户请求的 QoS 控制任务分担到了每一个节点上,其中主要的计算负载由各个服务器承担,大幅度提高了 QoS 控制机制的可扩展性和可用性.各个控制器独立性较强,可以针对不同的 QoS 控制目标或者应用类型,设计不同的控制器模型和算法;由于控制器在服务器端以用户空间程序实现,可以方便地使用各类开发工具,有

利于第三方控制软件的集成,这些都大幅度提高了 QoS 控制的灵活性与适应性.上述 QoS 控制框架独立于服务器端的应用软件,现有系统无须更改,大幅度降低了实际部署和应用的难度.

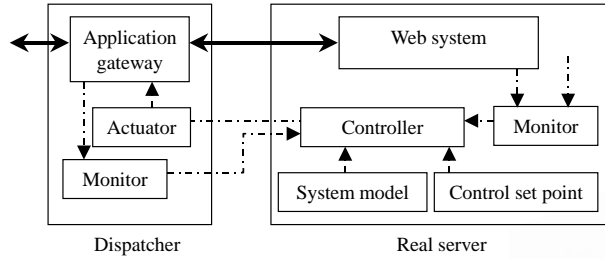


Fig.3 Distributed QoS control system based on control theory
图 3 基于控制论的分布式 QoS 控制系统

分发器和服务器之间周期性的双向控制通信以 UDP 报文来实现,形成自然的双向心跳检测机制,心跳检测失效的服务器将被各个分发器剔除;反之亦然,分发器上不再需要传统的针对服务器的心跳检测.传统由分发器完成的服务可用性检测也由各个服务器上的检测器完成了,如果服务失效,将由相应的控制器通告给各个分发器.结合分发器相互之间的心跳备份机制,即使集群系统出现部分失效,服务可用性仍能得到保证.

2.3 量化的QoS控制

2.3.1 观测器的设计

服务器上的观测器:

- 利用操作系统提供的基础设施,定期采集系统负载信息,主要包括 CPU 利用率、内存使用率.由于 CPU 利用率瞬时波动较大,为了获得较为平滑的整体系统负载信息,避免控制中的振动,使用低通滤波算法对 CPU 负载数据进行预处理,低通滤波器的传递函数为 $G(s)=1/(0.5s+1)$,相应的递推算式为 $y(k+1)=0.998 \times y(k)+0.000999 \times (u(k+1)+u(k))$,其中, k 代表采样时刻, $y(k)$ 为平滑值, $u(k)$ 为实测值.图 4 为使用该低通滤波器过滤输入高频噪声的例子.
- 检测本地服务的可用性,定期通过可配置的脚本访问指定的服务内容,以判断服务可用性.

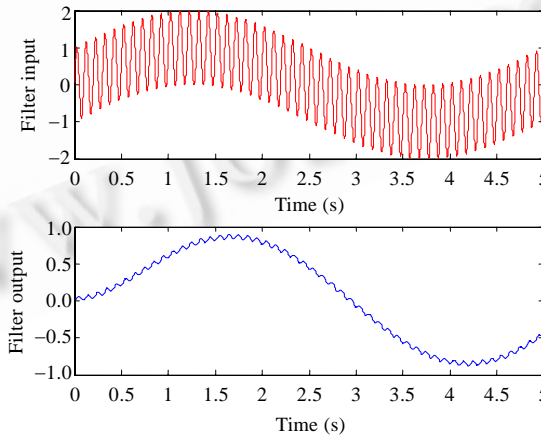


Fig.4 An example of data preprocessing using the low frequency filter
图 4 数据的低通滤波预处理示例

分发器上的观测器通过应用层网关分别为各个控制器收集虚拟服务各类用户请求的 QoS 信息,包括:

- 请求接纳速率:各类请求经由分发器进入服务器的速率.
- 并发请求数:服务器上并发处理的各类请求数.

- 数据吞吐率:各类请求的数据转发率.
- 响应时间:各类请求的平均响应时间.由于响应时间与客户端及网络情况等服务器端不可控因素密切相关,不同类型请求、不同请求对象的响应时间也不相同,同时,考虑到系统负载较大时,用户请求在服务系统的排队时间在响应时间中所占比重较大,因此我们选用初始响应时间作为响应时间指标的测量标准.所谓初始响应时间,即用户请求在分发器应用网关处分析归类完毕,直到服务器返回第 1 个响应报文这段时间.这样,既屏蔽了上述因素的影响,又充分考虑到了应用的特性.

分发器上观测器的数据发送到控制器后,同样进行与前述类似的低通滤波预处理.

2.3.2 激励器的设计

为了保证控制系统对应用系统的透明性,我们选用如下控制量:

- 最大请求接纳速率:通过调节各类请求的最大接纳速率,动态控制其资源使用比率,以满足量化的 QoS 控制目标.
- 最大并发请求数:通过调节各类请求并发处理的最大数量来控制服务器的负载总量.

用户请求在分发器上分析归类完毕后,应用层网关将针对激励器给出的上述控制量实施控制.当阈值条件满足时,应用网关直接选出当前阈值余量最大的服务器,分发用户请求并更新有关参数.当超出其中任何一项阈值时,该类后继请求将在分发器中相应的 FIFO 队列排队等待,直至阈值条件满足.排队请求出队的时机为:控制器发布新的控制量,各项阈值条件均满足要求;有同类请求处理结束,各项阈值条件均满足要求;等待请求所在连接出错或等待超时.

2.3.3 控制器的设计

上述观测器和激励器的设计具有通用性,它们提供了实施 QoS 控制所必需的各类数据和控制手段.而 QoS 控制器的设计则非常灵活,同一虚拟服务在不同服务器上的控制器、同一服务器上不同虚拟服务的控制器,均可独立设计和配置.针对不同的应用类型或者不同的量化 QoS 控制目标,可以设计或定制相应的控制器.典型的量化 QoS 控制目标有:QoS 保证(各类服务请求拥有受保证的量化性能指标,如响应时间或数据吞吐率)、QoS 区分(各类服务请求拥有量化性能区分,如响应时间或数据吞吐率的固定比例),等等.

近年来,以模糊控制、专家控制、神经网络控制、滑模变结构控制等为代表的智能控制技术在复杂工业过程控制领域得到了大量的研究和尝试.在本文中,我们将给出一种基于模糊控制理论设计的量化 QoS 保证控制器,控制目标为平均请求响应时间.

模糊控制(fuzzy control)^[20]是将模糊集合理论运用于自动控制的一种计算机数字控制技术,其本质是以模糊规则及模糊推理表示具有模糊性的人类控制经验或专家知识,是一种基于规则的控制.对于时变非线性的复杂系统,要得到正确而且精密的数学模型相当困难.这些系统有大量重要的定性知识以及仅仅用语言规定的性能指标,由人来控制却往往容易做到.通过把操作人员的控制经验归纳成定性描述的一组条件语句,然后运用模糊集合理论将其定量化,使控制器得以接受人的经验,模仿人的操作策略,这样就得到了模糊控制器.作为智能控制领域中最具实际意义的一种控制方法,模糊控制已经在工业控制领域、家用电器自动化领域和其他很多行业中解决了传统控制方法无法或者是难以解决的问题,取得了令人瞩目的成效.Web服务由于服务内容的多样性、用户请求的多变性、服务处理过程的复杂性,本质上为时变非线性系统,因而难以建立精确且具有广泛适应性的系统数学模型.模糊控制因其在处理复杂系统控制方面的优势,为解决Web的QoS控制问题提供了有力的工具.

我们设计的 QoS 控制器由两个子控制器组成,分别用于确定“最大请求接纳速率”及“最大并发请求数”两大控制量.

“最大请求接纳速率”子控制器与量化的QoS控制直接相关,目前主要实现了平均响应时间、数据吞吐率两类量化控制器.该控制器共有 3 项输入:容量利用率(utilization),表示实际接纳请求数占当前最大请求接纳速率的比率;系统负载(load),表示系统CPU、内存综合利用率;相对性能偏差率(QoS),表示实际QoS指标偏差占期望值的比率.该控制器有一项输出:容量调整比率(capacity),表示本次容量(最大请求接纳速率)调整量的相对比率.

控制器采用“单点模糊、乘积推理、中心平均去模糊化”的Mamdani模糊推理机.平均响应时间控制器的输入、输出的隶属度函数如图 5 所示,共有 29 条模糊推理规则,其控制曲面如图 6 所示.数据吞吐率由于与响应时间成反相关的关系,其控制器设计与图 5、图 6 类似,主要区别是控制动作相反.

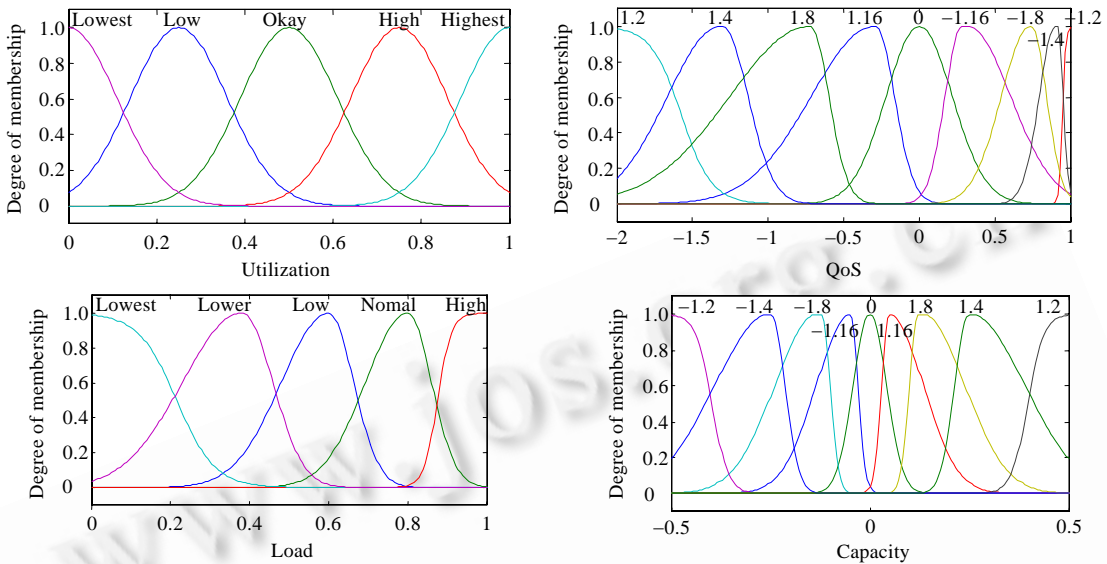


Fig.5 Membership functions for the inputs and output of the fuzzy controller for Ave. time

图 5 平均响应时间模糊控制器输入、输出隶属度函数

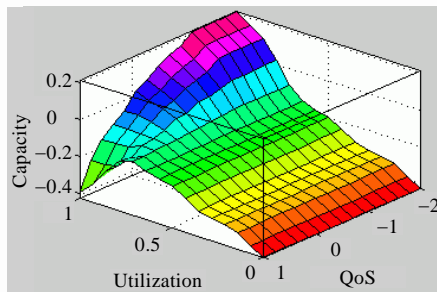


Fig.6 Control surface of the fuzzy controller for Ave. time

图 6 平均响应时间模糊控制器控制曲面

“最大并发请求数”子控制器主要用于控制本地服务器的负载,防止系统超载.该控制器共有两项输入:容量利用率(utilization)表示实际并发处理请求数占当前最大并发请求数的比率,系统负载(load)表示系统CPU、内存综合利用率;一项输出:容量调整比率(capacity),表示本次容量(最大并发请求数)调整量的相对比率.控制器采用具有 9 条模糊推理规则的Mamdani模糊推理机.

3 实现及测试

我们实现了如图 2、图 3 所示的分布式 QoS 控制系统.在分发器上,基于 Linux 2.4 版内核实现了集群服务应用层网关及相应的观测器和激励器.应用层网关支持 HTTP 1.1 协议,能够根据配置对用户请求按照各种特征进行分类.服务器上的控制器和观测器实现为用户空间程序,分别实现了 Windows 平台和类 Unix 平台(Linux, FreeBSD, Solaris 等)两大类版本,具有很强的多平台适应性.

测试平台包括 2 台分发器、2 台服务器、4 台客户机.每台机器配置为:单 CPU 2.5GHz Intel P4,512MB 内存及千兆以太网(分发器为双网卡).分发器及客户机运行 Linux 内核 2.4;服务器分别运行 Linux 内核 2.4 及

Windows 2000,其上分别运行 Apache 2.0.46 及 IIS 5.0,最大连接数均为 1 024,连接超时为 15s,无其他任何访问限制设置.

第 1 项测试用以检验简单负载类型下量化 QoS 控制机制对平均响应时间控制的有效性.在测试中只使用了 1 台分发器.使用 sclient^[21]模拟 400 个并发访问用户持续访问 30s,请求同一个大小为 190KByte 的图片.目标(初始)响应时间为 100ms,初始的最大请求接纳速率及最大并发请求数均为 50.测试结果如图 7 所示,可以看出,在简单负载下,控制过程很快趋于平稳,实际响应时间收敛到目标值.

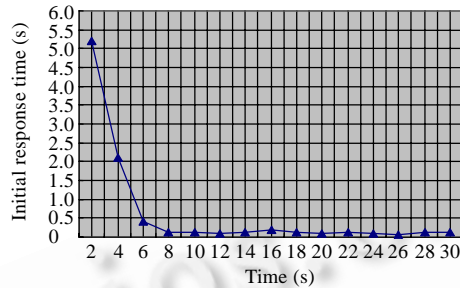


Fig.7 Test results under simple load pattern

图 7 简单负载模式下的测试结果

第 2 项测试用以检验在更为真实的负载类型下,量化 QoS 控制机制对平均响应时间控制的有效性.使用 SURGE 模拟 400 个并发访问用户,访问由 SURGE^[22]自带工具根据某些已知负载统计规律(如文件大小的重尾分布律)生成的文件集.其他设置与第 1 项测试相同.测试结果如图 8 所示,可以看出,虽然有局部偏离发生,控制器仍具有较好的控制效果.

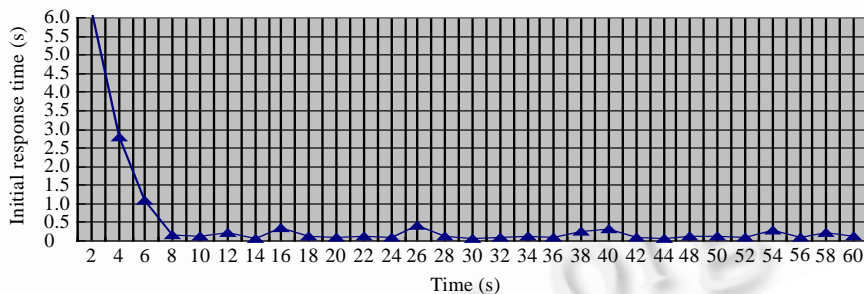


Fig.8 Test results under more realistic load pattern

图 8 较为真实负载模式下的测试结果

第 3 项测试用以检验控制目标变化时量化 QoS 控制机制对平均响应时间控制的有效性.基本设置与第 2 项测试相同,所不同的是目标(初始)响应时间设为 500ms,在控制结果趋于平稳后,在线将控制器的目标响应时间调整为 400ms.Windows 服务器上的性能数据如图 9 所示.

第 4 项测试用以验证量化 QoS 控制对数据吞吐率控制的有效性.基本设置与第 2 项测试相同,所不同的是初始目标数据吞吐率设为 10MByte/s,在控制结果趋于平稳后,在线将控制器的目标数据吞吐率调整为 40MByte/s.Windows 服务器上的性能数据如图 10 所示.可以看出,数据吞吐率的变化较平均响应时间更为平稳,其原因主要在于:数据统计以 MByte/s 为单位,有舍入运算;采样间隔时间平滑了数据吞吐率的统计值.

第 5 项测试用以验证多分发器结构下的量化 QoS 控制的有效性.使用 SURGE 在 3 台客户机上模拟 300 个并发访问用户,访问其中一台分发器;在剩下一台客户机上模拟 100 个并发访问用户访问第 2 台分发器.其他设置与第 2 项测试相同.测试结果与图 8 类似,所不同的是,两台分发器所获得的“最大请求接纳速率”指标的比例大致为 3:1,而在第 2 项测试中,这个比例大致为 1:1.可以看出,分布式的 QoS 控制对于多分发器上负载分布具有自适应性,控制效果不受影响.

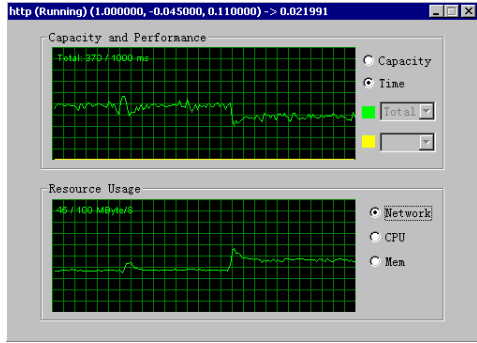


Fig.9 Test results when Ave. time control object changed

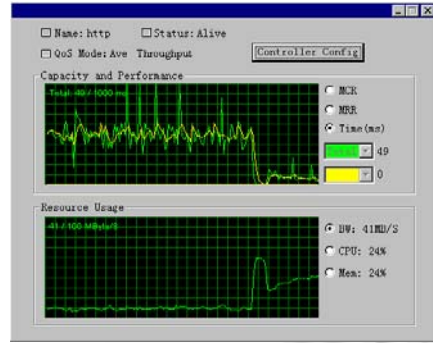


Fig.10 Test results when Ave. throughput control object changed

图 9 平均响应时间控制目标变化时的测试结果 图 10 数据吞吐率控制目标变化时的测试结果

上述测试均表明,基于控制论设计的 Web 集群分布式 QoS 量化控制是有效的,具有较强的适应性.模糊控制器的引入,较好地平滑了难以数学建模的复杂系统模型带来的控制振荡等复杂因素的影响,实现了较好的控制效果.整个控制系统不需要进行复杂的参数整定,完全透明于应用系统,减小了实际应用的难度.

为了观察服务器端控制程序带来的开销,我们在没有任何用户负载的情况下用“top”、“任务管理器”等工具查看其 CPU 使用率.控制程序以 2s 为周期运行,其运行时的 CPU 使用率在 1%左右,对于服务器来说这是微小的开销,但如果数十个这样的控制器运行在像分发器这样的单个节点上,将对性能产生严重的影响.控制系统内部进行周期为 2s 的双向通信,总开销为“单次双向通信开销×请求类型数量×分发器数量×控制器数量”,测试中总共配置了一个 HTTP 虚拟服务,5 类用户请求,通信总开销为 $35 \times 5 \times 2 \times 2 = 700$ Byte.可以看出,在分布式的控制结构下,全系统无单一性能瓶颈点,无单一故障点.

4 总 结

Web 集群 QoS 量化控制具有重要的理论和现实意义,国内外目前的研究与实现还很少.本文提出了基于控制论设计的分布式 QoS 量化控制方案,实现与测试证明了该方案的有效性.这一方案不仅适用于 Web 应用,对邮件等其他多种集群网络服务也有应用价值.目前,该系统已经在金山“剑侠情缘”等门户网站成功投入实用,应用效果良好.

与现有 QoS 控制方案相比,本文提出的分布式 QoS 量化控制方案的创新之处在于:(1) 基于控制论提供了集群 QoS 控制的系统化设计基础,便于根据不同需求设计灵活、有效的控制器;(2) 实现了量化的 QoS 控制;(3) 使用模糊控制较好地解决了模型失配、参数整定困难等传统控制方法的问题(适用的控制方法不仅限于此);(4) 分布式控制结构消除了全局单一性能瓶颈点和单一故障点;(5) 控制系统完全透明于现有的应用系统,实际应用难度和成本大为降低.

在今后的研究中,我们将在本文提出的 Web 集群分布式 QoS 量化控制框架下进一步研究其他类型控制器的应用,并进一步研究平均响应时间、平均数据吞吐率之外的其他 QoS 量化控制目标的实现.

References:

- [1] Crovella ME, Bestavros A. Self-Similarity in World Wide Web traffic: Evidence and possible causes. *IEEE/ACM Trans. on Networking*, 1997,5(6):835-46.
- [2] Li K, Jamin S. A measurement-based admission-controlled Web server. In: *Proc. of the IEEE Infocom 2000*. Tel Aviv: IEEE, 2000.
- [3] Aron M, Druschel P, Zwaenepoel W. Cluster reserves: A mechanism for resource management in cluster-based network servers. In: *Proc. of the ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems*. Santa Clara: ACM, 2000.
- [4] Urgaonkar B, Shenoy P. Dynamic provisioning of multi-tier Internet applications. In: *Proc. of the 2nd IEEE Int'l Conf. on Autonomic Computing (ICAC 2005)*. Seattle: IEEE, 2005.
- [5] Schroeder B, Harchol-Balter M. Web servers under overload: How scheduling can help. *ACM Trans. on Internet Technology*,

- 2006,6(1):20–52.
- [6] Guitart J, Carrera D. Session-Based adaptive overload control for secure dynamic Web applications. In: Proc. of the 34th Int'l Conf. on Parallel Processing (ICPP 2005). Washington: IEEE, 2005. 341–349.
- [7] Urgaonkar B, Shenoy P. Cataclysm: Policing extreme overloads in internet applications. In: Proc. of the 14th Int'l World Wide Web Conf. (WWW 2005). Chiba: ACM, 2005.
- [8] Breslau L, Knightly EW, Shenker S, Stoica I, Zhang H. Endpoint admission control: Architectural issues and performance. In: Proc. of the ACM SIGCOMM 2000. Stockholm: ACM, 2000. 57–69.
- [9] Bhoj P, Ramanathan S, Singhal S. Web2K: Bringing QoS to Web servers. Technical Report, HPL-2000-61, HP Labs., 2000.
- [10] Elnikety S, Nahum E. A method for transparent admission control and request scheduling in e-commerce Web sites. In: Proc. of the WWW2004. New York: ACM, 2004. 17–22.
- [11] Chandra S, Ellis CS, Vahdat A. Differentiated multimedia Web services using quality aware transcoding. In: Proc. of the IEEE INFOCOM 2000. Tel Aviv: IEEE, 2000.
- [12] Abdelzaher T, Bhatti N. Web server QoS management by adaptive content delivery. In: Proc. of the 7th Int'l Workshop on QoS. 1999. 1563–1577.
- [13] Chen X, Chen H, Mohapatra P. An admission control scheme for predictable server response time for Web accesses. In: Proc. of the 10th World Wide Web Conf. Hong Kong: ACM, 2001. 545–554.
- [14] Li ZD, Xie L. Research on ensuring QoS and its admission control in Web servers. Journal of Computer Research and Development, 2005,42(4):662–668 (in Chinese with English abstract).
- [15] Lu CY, Abdelzaher TF, Stankovic JA, Son SH. A feedback control approach for guaranteeing relative delays in Web servers. In: Proc. of the IEEE Real-Time Technology and Applications Symp. Taipei: IEEE, 2001. 51–62.
- [16] Diao Y, Gandhi N, Hellerstein J, Parekh S, Tilbury D. Using MIMO feedback control to enforce policies for interrelated metrics with application to the apache Web server. In: Network Operations and Management. 2002. 219–234.
- [17] Lu Y, Abdelzaher TF, Saxena A. Design, implementation, and evaluation of differentiated caching services. IEEE Trans. on Parallel and Distributed Systems, 2004,15(5):440–452.
- [18] Lu CY, Lu Y, Abdelzaher TF, Stankovic JA, Son SH. Feedback control architecture and design methodology for service delay guarantees in Web servers. IEEE Trans. on Parallel and Distributed Systems, 2006,17(9):1014–1027.
- [19] Franklin GF, Powell JD, Workman M. Digital Control of Dynamic Systems. 3rd ed., New York: Addison-Wesley, 1998.
- [20] Zhang ZX, Sun CZ, Mizutani E, Wrote; Zhang PA, Gao CH, *et al.*, Trans. Neuro-Fuzzy and Soft Computing. Xi'an: Press of Xian Jiaotong University, 2000 (in Chinese).
- [21] Banga G, Druschel P. Measuring the capacity of a Web server. In: Proc. of the Usenix Symp. on Internet Technologies and Systems. Monterey: USENIX, 1997. 61–71.
- [22] Barford P, Crovella ME. Generating representative Web workloads for network and server performance evaluation. In: Proc. of the ACM SIGMETRICS Conference. Madison: ACM, 1998. 151–160.

附中文参考文献:

- [14] 李振东, 谢立. Web 服务器群的 QoS 确保及其接纳控制研究. 计算机研究与发展, 2005, 42(4): 662–668.
- [20] 张智星, 孙春在, 水谷英二, 著; 张平安, 高春华, 等. 译. 神经-模糊和软计算. 西安: 西安交通大学出版社, 2000.



王晓川(1976—),男,四川内江人,博士生,主要研究领域为分布式系统,Web 系统,系统性能评价.



夏明波(1980—),男,硕士生,主要研究领域为集群技术.



金士尧(1937—),男,教授,博士生导师,CCF 高级会员,主要研究领域为实时系统,并行与分布式系统,复杂系统仿真,系统性能评价,容错系统.