Web请求分配和选择的综合方案与性能分析*

单志广[1], 戴琼海[2], 林 闯[3], 杨 扬[1]

[1](北京科技大学 信息工程学院,北京 100083);

[2](清华大学 自动化系,北京 100084);

[3](清华大学 计算机科学与技术系,北京 100084)

E-mail: Shanzhiguang@ustb.edu.cn

http://www.ustb.edu.cn

摘要: Internet的服务模式正由传统的通信与信息浏览向电子交易与服务转变,这就要求WWW服务器既支持电子商务类具有优先级的请求,同时也要维护各类Web应用的公平性.以实现系统负载均衡和满足不同请求的WebQoS需求及公平性为目标,讨论并提出了并行WWW服务器集群系统HTTP请求分配和选择的综合方案,并提供了这些方案的随机高级Petri网模型.为解决模型状态空间爆炸问题,还提出了一种可以显著简化模型求解复杂性的近似性能分析技术;给出了综合方案的数值分析结果和性能评价,建议了适合电子商务类应用的、实现高性能集群系统的综合方案.

关键词: WWW服务器集群;请求分配;请求选择;性能分析;随机高级Petri网

中图法分类号: TP393 **文献标识码**: A

随着Internet上WWW应用的迅猛发展和用户访问频率的提高,目前Web流在Internet总流量中所占的比例已超过50%.客户HTTP请求的指数性增长使得Internet上的许多热门站点日益面临服务器超载的问题.集群技术是目前解决该问题的一种有效途径.WWW服务器集群是由多台WWW服务器(同构的或异构的)通过高速局域网联结而成的,对外部具有单一网络地址的体系结构,通过并行处理来扩展系统的实时性和吞吐量等性能.请求分配是指由一台特殊的计算机(请求分配器)集中接收所有到达的HTTP请求,然后按照特定的分配方案将客户的请求均衡、透明地分配到集群中的各台服务器上.请求选择是指服务器根据请求的不同类型和要求,按照特定的选择方案从缓冲队列中挑选请求实施服务.

伴随电子商务的兴起,Internet的服务模式正由传统的通信与浏览向电子交易与服务转变,这使得WWW服务器成为支持电子商务的核心设施.与传统的TCP/IP和HTTP服务的平均主义哲学不同,电子商务通常要求对用户或服务进行区分优先级别的处理,例如,通过网上在线公司进行股票投资交易的请求要比简单的浏览或下载具有更加严格的实时要求.但目前大多数UNIX内核的WWW服务器采用FIFO(first-in-first-out)调度策略,在超载的情况下不加区别地丢弃高优先级的请求分组,使得通过网络集成服务(integrated services)或区分服务(differentiated services)所实现的QoS改进受到严重损害.因此,仅靠网络QoS机制并不能完全解决端到端的QoS问题,

* 收稿日期: 2000-03-27; 修改日期: 2000-06-15

基金项目: 国家重点基础研究发展规划资助项目(G1999032707);国家自然科学基金资助项目(69873012);国家863高科技发展计划资助项目(863-306-ZT05-01-02,863-300-05-04-02-00)

作者简介: 单志广(1975-),男,黑龙江省哈尔滨人,博士生,主要研究领域为计算机网络,系统性能评价;戴琼海(1964-),男,上海人,博士,教授,博士生导师,主要研究领域为多媒体通信;林闯(1948-),男,山东栖霞人,教授,博士生导师,主要研究领域为系统性能评价,计算机网络,随机Petri网;杨扬(1955-),男,河北隆化人,教授,博士生导师,主要研究领域为多媒体通信.

© 中国科学院软件研究所 http://www.jos.org.cn

WWW 服务器也应具备建立和支持 QoS 的机制与策略[1]. 本文研究的请求分配和选择方案,将考虑不同请求的 WebQoS 需求,并兼顾服务的公平性.

请求分配和选择方案及其性能分析是集群技术的重要研究内容. 目前已提出了一些请求分配算法,例如,转轮(round-robin)[2,3]、最少连接优先(least connections first)[2,4]、快速反应优先(faster response precedence)[2,4]和选择加权百分率(selected weighted percentage)[2,5]等. 请求选择方案有:随机均衡选择(random selecting)、列优先级(head of line)和队列长度阈值(queue length threshold)等[6]. 文献[6]已经给出了若干请求分配和选择方案,并利用随机高级 Petri 网(stochastic high level Petri net,简称 SHLPN)[7]模型技术进行了性能分析. 本文在此基础上,提出一种新的请求选择方案——加权优先级(weighted priority)方案,兼顾请求的 QoS 要求及服务的公平性;并提出请求分配和选择的综合方案,目标是综合实现 WWW 服务器集群的负载均衡及其 QoS 控制,给出了综合方案的 SHLPN 模型及其近似分析方法,通过数值结果分析和评价了不同综合方案的性能差异,建议了有效保证负载均衡的、支持请求优先级并兼顾公平性的综合方案.

1 请求分配和选择的 SHLPN 模型与方案

SHLPN 是图形的、数学的模型和分析的工具,已广泛应用于计算机科学、通信网络、管理与控制等领域,其并行、并发、资源共享的描述特性以及模型分解和压缩技术更适合于对系统资源管理、请求调度方案和模型的研究. SHLPN 的详细理论和应用可参阅文献[7].

1.1 请求分配和选择的 SHLPN 模型

在我们的 SHLPN 模型中,请求的到达和接受服务可由时间变迁来表示,到达和服务的速率与系统的状态相关;请求进入缓冲队列和共享互斥区由瞬时变迁表示,不占用处理时间,并可联系随机开关(即实施概率). 瞬时变迁比时间变迁有更高的实施优先级. 缓冲队列由位置来表示,它们的占有程度由位置的标识(marking)表示. 允许变迁的实施条件用变迁的可实施谓词规定,当谓词条件不能满足时,变迁不能实施. 模型中的标记(token)表示请求或资源,不同类型的请求或资源使用不同颜色的标记表示.

为使本文研究的 WWW 服务器集群系统具有一般性,我们作如下约定($1 \leqslant i \leqslant n; 1 \leqslant j \leqslant m$):

(1) 系统包含 m 个服务器,接受 n 类请求. 其中,第 i 类请求记作 r_i,第 j 个服务器记作 s_j.

(2) 每个服务器包含一个缓冲队列,s_j 的队列由标识符 q_j 表示,其缓冲空间的容量为 b_j.

(3) 任一类请求的到达为泊松(poisson)过程. 请求 r_i 到达速率为 λ_i,r_i 可以被分配到 m 个队列中的任一队列. 当所有 m 个队列都满时,请求的接纳过程就会中断.

(4) 每个服务器服务不同请求可有不同的服务时间. s_j 的服务速率为 μ_j,服务速率是独立的、指数分布的.

必须说明,虽然目前国际上许多研究工作指出:在很多网络环境中网络传输的分组和连接的到达存在着自相似性[8],但性能分析在什么条件下必须将自相似性考虑进来,仍是一个活跃的研究领域,关于自相似模型的应用范围也没有一致的意见. 在 SIGMETRICS'95 会议上对此问题讨论的结果[9]也清楚地表明:自相似效应在某些网络环境中很重要,而在另外的环境下对性能没有多大影响. 文献[10]指出,应用级(如 Web)自相似传输应该与网络级自相似传输区别对待,而且应用级自相似传输的行为在很大程度上与传输网络的情况无关,其自相似性可以通过接纳控制和资源分配的方式得到有效的控制. Catledge 和 Cunha 等人通过分析一些著名 Web 站点的日志文件发现,

© 中国科学院软件研究所 http://www.jos.org.cn

Web 访问中会话组的到达规律基本上服从负指数分布[11]. 文献[12,13]中的理论分析和仿真实验结果均表明,在相邻 Web 请求之间,时间间隔的分布基本上符合一种分阶段的负指数分布. 综上所述,与网络级传输较明显的自相似性不同,对于应用级的 Web 请求,上述(3)中泊松到达的假设是可行的.

图 1 给出了一个 WWW 服务器集群系统请求分配和选择的 SHLPN 模型. 其中,对应于 n 类不同客户的请求,我们将每个 WWW 服务器分解成 n 个独立的逻辑服务器,每个逻辑服务器对应于一个逻辑缓冲队列. 在请求接受服务时,每个服务器的 n 个逻辑队列共享该服务器. 图 1 中主要变迁和位置的含义如下($1\leqslant i\leqslant n, 1\leqslant j\leqslant m$):

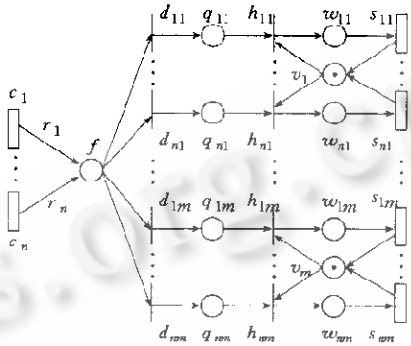


Fig. 1 A SHLPN model of request dispatching and selecting for Web server clusters

图1 WWW服务器集群系统请求分配和选择的SHLPN模型

q_{ij}:表示服务器 s_j 接收请求 r_i 的逻辑缓冲队列. 标识符的第 1 个下标表示接收请求的类型,第 2 个下标表示队列所属的服务器. q_{ij} 的容量限定为 b_{ij},q_{ij} 中请求 r_i 的标识数量记为 $M(q_{ij})$.

c_i:表示请求 r_i 到来的时间变迁,它有实施速率 λ_i.

d_{ij}:表示将请求 r_i 分配到服务器 s_j 的瞬时变迁,分配方案由其所联系的可实施谓词和随机开关表达.

f:表示请求分配时进行判断的位置(即请求分配器),它瞬时保留到来的请求,根据 d_{ij} 联系的可实施谓词和/或随机开关以决定到来的请求放入哪一队列.

s_{ij}:表示服务器 s_j 对队列 q_{ij} 中的请求 r_i 实施服务的变迁,它的实施速率 $\mu_{ij}=\mu_j/r_i$,s_j 对请求 r_i 的服务时间表达为 $T_{ij}=r_i/\mu_j$. 其中 r_i 既代表请求类型,又表示服务时间要求的权重.

v_j:表示共享服务器的位置,它包含一个服务器标记.

图 1 中,假定 n 个逻辑队列按实时调度优先级排列. 队列 q_{1j} 的实时调度优先级最高,q_{nj} 最低. 每个逻辑队列包含相同类型的请求,有相同的实时要求,按照 FIFO 机制接受服务. TH_{ij} 是 q_{ij} 中用户请求的实时调度阈值. 当 q_{ij} 中请求标识的数量达到或超过 TH_{ij},而当在 $q_{kj}(1\leqslant k<i)$ 中请求标识的数量小于 TH_{kj} 时,q_{ij} 中的请求得到服务. 通常,请求实时调度的优先级越高,它的实时调度阈值就越小.

SHLPN 模型的性能分析有两种方法:一种是根据模型构造对应的马尔可夫链(Markov chain),对整个模型直接求解;另一种是对模型进行分解、迭代的近似求解. 一般情况下,图 1 模型是一个 $m\times n$ 维的马尔可夫链,随着 b_{ij},m 和 n 的增大,马尔可夫链的状态空间呈指数增长. 当状态的数量超过一定限制后,目前一般计算机的存储和计算能力无法容忍,而使问题不可实际求解. 所以可行的解法是分解模型和迭代求解子模型之间相互关系的近似求解.

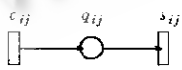


Fig. 2 SHLPN submodel A_{ij} for request dispatching and selecting

图2 请求分配和选择的SHLPN子模型 A_{ij}

我们对图 1 的模型进行分解,将每个逻辑服务器及其逻辑队列分解成一个子模型. 图 2 描述了一个经过精化设计的子模型 A_{ij},共有 $m\times n$ 个类似的子模型. c_{ij} 表示请求 r_i 被分配到 q_{ij} 的时间变

© 中国科学院软件研究所 http://www.jos.org.cn

迁.子模型之间的相互影响关系表现为迭代求解过程中子模型之间输入和输出的参数关系,在变迁 c_{ij} 和 s_{ij} 的实施谓词和速率中加以描述.它们可实施与否,既与该子模型队列状态相关,又与其他子模型队列状态相关.在模型中,稳定状态下的标识概率作为输入、输出参数.$M(q_{ij})$ 的稳定状态概率用 $P[M(q_{ij})]$ 表示.

1.2 请求分配方案

图2中变迁 c_{ij} 可以联系不同的可实施谓词和随机开关,从而决定了不同的请求分配方案.我们考虑如下3种请求分配方案进行模型设计:

(1) 转轮(round-robin,简称RR)方案:到达的请求循环分配给集群中的每一个服务器,各个服务器获得请求的机会均等.

(2) 最少连接优先(least connections first,简称LCF)方案:选择当前集群中等待队列最短的服务器队列作为到达请求分配的目的地.

(3) 最小期望等待时间(shortest expected delay,简称SED)方案:选择当前具有最小期望等待时间(即队列长度与服务器服务时间的乘积函数值为最小)的服务器队列作为到达请求分配的目的地.

在以上3种方案中,变迁 c_{ij} 的可实施谓词和随机开关的表达式详见文献[6],这里从略.

1.3 请求选择方案

图2中变迁 s_{ij} 联系不同的可实施谓词和随机开关就决定了不同的请求选择方案.我们考虑如下3种请求选择方案的模型设计:

(1) 随机均衡选择(random selecting,简称RS)方案

不同类型的请求对于任一服务器具有相同的接受服务概率.

变迁 s_{ij} 没有可实施谓词.变迁 s_{ij} 与标识相关的可实施概率 $x(M(q_{ij}))$ 可以写作

$$x(M(q_{ij}))=\begin{cases}\dfrac{1}{\| RS(M) \|} & \text{if } i\in RS(M)\\ 0 & \text{otherwise}\end{cases}$$

其中 $RS(M)=\{k\mid M(q_{kj})>0,\text{for } 1\leqslant k\leqslant n\}$.

(2) 队列优先级(head of line,简称HOL)方案

具有最高优先级的请求总是最先得到处理.服务器 s_j 中各逻辑队列的实时调度阈值为 $TH_{1j}=TH_{2j}=\ldots=TH_{nj}=1$.

变迁 s_{ij} 的可实施谓词 z_{ij} 表达为 $z_{ij}:[(M(q_{ij})\geqslant TH_{ij})\wedge(\forall\ k,1\leqslant k<i,M(q_{kj})<TH_{kj})]$.

变迁 s_{ij} 与标识相关的可实施概率 $x(M(q_{ij}))$ 可以写作

$$x(M(q_{ij}))=\begin{cases}1 & \text{if } i\in RS(M)\\ 0 & \text{otherwise}\end{cases}.$$

其中 $RS(M)=\{k\mid M(q_{kj})>0 \text{ and } M(q_{lj})=0,\text{for } 1\leqslant k\leqslant n,\forall\ l\neq k,1\leqslant l<k\}$.

(3) 加权优先级(weighted priority,简称WP)方案

根据客户请求的加权重要程度进行选择服务.在我们的模型中,假设到达的请求带有区分重要程度的优先权重.如果某类请求的实时优先权重是另一类请求的两倍,则该类请求得到服务的概率就是另一类请求的两倍.具有高实时优先级的请求会获得高服务概率,但不是绝对优先概率1.在服务器 s_j 中,各逻辑队列的实时调度阈值为 $TH_{1j}=TH_{2j}=\ldots=TH_{nj}=1$.

© 中国科学院软件研究所 http://www.jos.org.cn

变迁 s_{ij}没有可实施谓词.

标识符 α_i 表示请求 r_i 的实时调度权重,变迁 s_{ij}与标识相关的可实施概率 $x(M(q_{ij}))$为

$$x(M(q_{ij}))=\begin{cases}1 & \text{if } i\in RS(M)\\ \beta_i & \text{if } i\in RN(M)\\ 0 & \text{otherwise}\end{cases}.$$

其中 $RS(M)=\{k\mid M(q_{kj})>0 \text{ and } M(q_{ij})=0, \text{for } 1\leqslant k\leqslant n, \forall\ l\neq k, 1\leqslant l\leqslant n\}$, $RN(M)=\{k\mid M(q_{kj})>0 \text{ and } M(q_{lj})>0, \text{for } 1\leqslant k\leqslant n, \exists\ l\neq k, 1\leqslant l\leqslant n\}$, β_i 的表达式为 $\beta_k=\alpha_k\Big/\Big(\alpha_k+\sum_l \alpha_l\Big)$.

2 请求分配和选择的综合方案与近似性能分析

请求分配和选择的综合方案由请求分配方案和请求选择方案联合而成,共有 9 种综合方案. 为叙述方便,可以根据请求选择(或分配)方案将其分成 3 组:基于 RS 的综合方案(RR-RS,LCF-RS 和 SED-RS)、基于 HOL 的综合方案(RR-HOL,LCF-HOL 和 SED-HOL)和基于 WP 的综合方案(RR-WP,LCF-WP 和 SED-WP).

请求分配方案的主要目标是实现负载均衡,而请求选择方案则要满足不同类型请求的 QoS 要求并兼顾公平性.这使得综合方案的设计应考虑请求分配和请求选择前后过程之间的配合,对两个前后独立的、目标不同的方案进行必要的目标一致性修改,使综合方案能够同时兼顾负载均衡和满足不同类型请求的 WebQoS 需求,实现集群系统整体的性能优化.具体而言,基于 HOL 和 WP 的两组综合方案由于在请求选择时区分不同请求的类型并支持 QoS 优先级,所以,在请求分配时必须同时考虑请求的类型及其 QoS 需求,才能与请求选择方案密切配合,取得更好的性能效果.这两组方案的综合是本文研究的重点,而基于 RS 的综合方案由于请求选择过程不考虑请求的类型及 QoS 需求,因而这组综合方案最简单,只是请求分配方案与请求选择方案的简单组合,不需要任何目标一致性的修改,我们视其为一组特殊的综合方案.

2.1 基于 HOL 的综合方案与近似性能分析

如前所述,基于 HOL 的综合方案其请求分配必须同时考虑请求的内容(类型)及其 QoS 需求,才能实现系统负载平衡与 QoS 控制的综合目标.具体而言,在基于 HOL 的综合方案的设计中,当涉及计算队列长度时,应该将优先级高于或等于请求逻辑队列中的标识数量之和记为该服务器队列的"有效长度",并按此"有效长度"实施 LCF 和 SED 请求分配方案,才能实现与 HOL 配合情况下的真正负载均衡.否则,即使是最长的服务器队列,如果包含的都是低优先级的请求,则在使用 HOL 请求选择方案的情况下它仍是轻载的.

下面详细介绍基于 HOL 的综合方案及近似性能分析技术.对于其他组综合方案,该分析过程完全类似.

定义集合 $E_j=\{1,\ldots,j-1,j+1,\ldots,m\}$是不包括服务器 s_j 的服务器索引集合,$E_j(k)$是从 E_j 中挑选 k 个元素的集合,它的补集 $\overline{E}_j(k)=E_j-E_j(k)$.

图 2 模型中变迁 c_{ij}在 3 种基于 HOL 的综合方案中的可实施谓词和实施概率如下:

(1) RR-HOL 综合方案

变迁 c_{ij}的可实施谓词 y_{ij}表达为 $y_{ij}:M(q_{ij})<b_{ij}$.

子模型 A_{ij}中变迁 c_{ij}同其他 k 个子模型 $A_{il}(1\leqslant l\leqslant m, l\neq j)$ 同时可实施的实施概率函数为

© 中国科学院软件研究所 http://www.jos.org.cn

$$g(M(q_{ij}),k)=\frac{1}{k+1}\sum\Big(\prod_{x\in E_j(k)}P[M(q_{ix})<b_{ix}]\prod_{y\in\overline{E}_j(k)}P[M(q_{iy})=b_{iy}]\Big). \quad (1)$$

(2) LCF-HOL 综合方案

"有效长度"$M_i(q_j)$定义为当请求 r_i 到来接受分配时,服务器 s_j 的队列 q_j 中优先级高于或等于 r_i 的请求标识数量:

$$M_i(q_j)=\sum_{k=1}^{i}M(q_{kj}).$$

变迁 c_{ij}的可实施谓词 y_{ij}写作 $y_{ij}:(M(q_{ij})<b_{ij})\wedge(\text{for }\forall\ k\neq j,(M_i(q_j)\leqslant M_i(q_k))\vee(M(q_{ik})=b_{ik}))$.

当服务器 s_j 中的标识数量 $M_i(q_j)=t$ 时,变迁 c_{ij}同其他 k 个子模型 $A_{il}(1\leqslant l\leqslant m,l\neq j)$同时可实施的实施概率函数为

$$g(M_i(q_j),k)=\frac{1}{k+1}\sum\Big(\prod_{(x\in E_j(k))\wedge(M(q_{ix})<b_{ix})}P[M_i(q_x)=t]\cdot \prod_{(y\in\overline{E}_j(k))\wedge(M(q_{iy})<b_{iy})}P[M_i(q_y)>t]\prod_{(y\in\overline{E}_j(k))}P[M(q_{iy})=b_{iy}]\Big). \quad (2)$$

(3) SED-HOL 综合方案

定义对应"有效长度"的最小期望等待时间函数为 $f_i(q_k)=\sum_{l=1}^{i}M(q_{lk})\times T_{lk}$.

变迁 c_{ij}的可实施谓词 y_{ij}写作 $y_{ij}:(M(q_{ij})<b_{ij})\wedge(\text{for }\forall\ k\neq j,(f_i(q_j)\leqslant f_i(q_k))\vee(M(q_{ik})=b_{ik}))$.

变迁 c_{ij}同其他 k 个子模型 $A_{il}(1\leqslant l\leqslant m,l\neq j)$同时可实施的实施概率函数为

$$g(f_i(q_j),k)=\frac{1}{k+1}\sum\Big(\prod_{(x\in E_j(k))\wedge(M(q_{ix})<b_{ix})}P[f_i(q_x)=f_i(q_j)]\cdot \prod_{(y\in\overline{E}_j(k))\wedge(M(q_{iy})<b_{iy})}P[f_i(q_y)>f_i(q_j)]\prod_{(y\in\overline{E}_j(k))}P[M(q_{iy})=b_{iy}]\Big). \quad (3)$$

在上述 3 种基于 HOL 的综合方案中,变迁 c_{ij}的实施概率函数 $g(M(q_{ij}))$一般表达为

$$g(M(q_{ij}))=g(M(q_{ij}),0)-g(M(q_{ij}),1)+\ldots+g(M(q_{ij}),m-1)=\sum_{k=0}^{m-1}g(M(q_{ij}),k). \quad (4)$$

在子模型 A_{ij}中,变迁 c_{ij}的实施速率表达为 $\lambda_i\times g(M(q_{ij}))$.

在上述 3 种基于 HOL 的综合方案中,变迁 s_{ij}的可实施谓词与第 1.3 节中的单独的请求选择方案相同,而 s_{ij}的实施概率函数为

$$X_{ij}=\prod_{i\neq k,k=1}^{n}P[M(q_{kj})=0]+\ldots+\frac{1}{\|Q\|}\prod_{k\in Q}P[M(q_{kj})>0]\times \prod_{k\notin Q}P[M(q_{kj})=0]+\ldots+\frac{1}{n}\prod_{k=1}^{n}P[M(q_{ki})>0]. \quad (5)$$

式(5)仅描述了其他子模型对变迁 s_{ij}的可实施概率的影响.n 个逻辑服务器对一个物理服务器的竞争也将造成变迁 s_{ij}实施速率的降低.在性能等价分析中,变迁 s_{ij}的实施时间 t_{ij}可以通过下列推导获得.

t_{ij}=在条件 X_{ij}下 s_{ij}的服务时间+在条件 X_{ij}下等待其他子系统服务的时间.具体地,在条件 X_{ij}下 s_{ij}的服务时间$=\frac{1}{X_{ij}\mu_{ij}}$;

© 中国科学院软件研究所 http://www.jos.org.cn

在条件 X_{ij} 下等待其他子系统服务的时间=X_{ij}×其他子系统忙的概率×期望剩余时间;

其他子系统忙的概率=$\dfrac{\sum_{k\neq i}T(s_{kj})}{\text{server service rate}}=\dfrac{\sum_{k\neq i}T(s_{kj})}{\mu_{kj}}$;

期望剩余时间=完全的服务时间(更新理论)=$\dfrac{1}{\mu_{kj}}$.

最后,我们有

$$t_{ij}=\frac{1}{X_{ij}\mu_{ij}}+X_{ij}\sum_{k\neq i}\frac{T(s_{kj})}{\mu_{kj}}\times\frac{1}{\mu_{kj}}. \tag{6}$$

所以,在子模型 A_{ij} 中,变迁 s_{ij} 的实施速率表达为 $1/t_{ij}$.

现在对子模型 A_{ij} 进行性能分析.考虑到变迁实施速率指数分布的假定,缓冲队列 q_{ij} 中请求个数的随机过程构成一个马尔可夫链,它仅包含一个缓冲队列,且是一个生死过程(birth-death process),其生和死的速率分别为 $\lambda_i\times g(M(q_{ij}))$ 和 $1/t_{ij}$.

缓冲队列 q_{ij} 中请求个数($y\geqslant 1$)分布的稳定概率具有乘积形式解,相邻两个稳定状态之间的关系容易获得,

$$P[M(q_{ij})=y]=\lambda_i\times g(M(q_{ij})=y-1)\times t_{ij}(M(q_{ij})=y)\times P[M(q_{ij})=y-1]. \tag{7}$$

任一状态稳定概率的乘积形式解表达为

$$P[M(q_{ij})=y]=P[M(q_{ij})=0]\times\prod_{k=0}^{y-1}\lambda_i\times g(M(q_{ij})=k)\times t_{ij}(M(q_{ij})=k+1). \tag{8}$$

其中

$$P[M(q_{ij})=0]=\left[1+\sum_{y=1}^{b_{ij}}\prod_{k=0}^{y-1}\lambda_i\times g(M(q_{ij})=k)\times t_{ij}(M(q_{ij})=k+1)\right]^{-1}. \tag{9}$$

至此,我们通过将集群系统的如图1所示模型分解为如图2所示的 $m\times n$ 个子模型,实现了将一个 $m\times n$ 维具有非乘积解的马尔可夫链分解为 $m\times n$ 个具有乘积解的简单生死过程.进而可通过 $m\times n$ 个子模型之间参数迭代,有效地求解系统模型的性能.在迭代中,可按 $A_{11},\ldots,A_{n1},\ldots,A_{1m},\ldots,A_{nm}$ 的顺序求解.在初始迭代中,首先求解子系统 A_{11},对所有稳定概率的初值可以设置为0和1之间的任一值.

系统的吞吐量是性能分析的一个重要测量指标.在一般SHLPN中,稳定状态下子系统 A_{ij} 的吞吐量(即时间变迁 s_{ij} 的吞吐量)$T(s_{ij})$ 表达为

$$T(s_{ij})=\mu_{ij}\sum_{M\in H}P[M], \tag{10}$$

其中 H 是能使变迁 s_{ij} 实施的所有标识集合,μ_{ij} 代表变迁 s_{ij} 在标识 M 下的实施速率.

WWW服务器集群系统的吞吐量 T 可以表达为

$$T=\sum_{i=1}^{n}\sum_{j=1}^{m}T(s_{ij}). \tag{11}$$

响应时间是系统的另一个重要性能指标.在一般SHLPN中,稳定状态下队列位置 q 的平均标记数量 $D(q)$ 可以表达为

$$D(q)=\sum j\times P[M(q)=j]. \tag{12}$$

子系统 A_{ij} 的响应时间 R_{ij} 为

$$R_{ij}=D(q_{ij})/T(s_{ij}). \tag{13}$$

第 i 类请求 r_i 的响应时间 R_i 可以表达为

© 中国科学院软件研究所 http://www.jos.org.cn

$$R_i=\sum_{j=1}^{m}\frac{T(s_{ij})\times R_{ij}}{\sum_{k=1}^{m}T(s_{ik})}. \tag{14}$$

WWW 服务器集群系统的响应时间为

$$R=\sum_{i=1}^{n}\frac{R_i\times\sum_{k=1}^{m}T(s_{ik})}{T}. \tag{15}$$

另外,选择调度的公平性对客户请求而言是与 QoS 性能同等重要的问题. 我们定义当第 i 类请求 r_i 的吞吐量 T_i 如下式所示时为取得良好的公平性:

$$T_i=T\times\frac{\lambda_i}{\sum_{k=1}^{n}\lambda_k}, \tag{16}$$

其中 λ_k 为第 k 类请求的输入速率.

2.2 基于 WP 的综合方案与近似性能分析

基于 WP 的综合方案的设计,需要完成与第 2.1 节完全相同的目标一致性修改. 方案综合和性能分析的过程及公式与第 2.1 节几乎完全相同,这里从略. 只是在基于 WP 的综合方案中,变迁 s_{ij} 的可实施概率为

$$X_{ij}=\prod_{i\neq k,k=1}^{n}P[M(q_{kj})=0]+\ldots+\frac{\alpha_i}{\sum_{k\in Q}\alpha_k}\prod_{k\in Q}P[M(q_{kj})>0]\times$$

$$\prod_{k\notin Q}P[M(q_{kj})=0]+\ldots+\frac{\alpha_i}{\sum_{k=1}^{n}\alpha_k}\prod_{k=1}^{n}P[M(q_{kj})>0]. \tag{17}$$

2.3 基于 RS 的综合方案与近似性能分析

基于 RS 的综合方案请求选择过程不考虑请求的实时优先级,因而不需要进行目标一致性修改,只是请求分配与选择方案的简单组合. 其分析过程与第 2.1 节中的不同之处如下:

变迁 c_{ij} 在此 3 种基于 RS 的综合方案中的可实施谓词均为 $M(q_{ij})<b_{ij}$.

(1) LCF-RS 综合方案

当服务器 s_{ij} 的队列标识 $M(q_{ij})=t$ 时,变迁 c_{ij} 同其他 k 个子模型 $A_{il}(1\leqslant l\leqslant m,l\neq j)$ 同时可实施的实施概率函数为

$$g(M(q_{ij}),k)=\frac{1}{k+1}\sum\Big(\prod_{(x\in E_j(k))\wedge(M(q_{ix})<b_{ix})}P[M(q_{ix})=t]\cdot$$

$$\prod_{(y\in E_j(k))\wedge(M(q_{iy})<b_{iy})}P[M(q_{iy})>t]\prod_{(y\in E_j(k))}P[M(q_{iy})=b_{iy}]\Big). \tag{18}$$

(2) SED-RS 综合方案

定义最小期望等待时间函数:

$$f(q_{ij})=M(q_{ij})\times T_{ij}.$$

当服务器 s_{ij} 中的标识为 $f(q_{ij})$ 时,变迁 c_{ij} 同其他 k 个子模型 $A_{il}(1\leqslant l\leqslant m,l\neq j)$ 同时可实施的实施概率函数为

© 中国科学院软件研究所 http://www.jos.org.cn

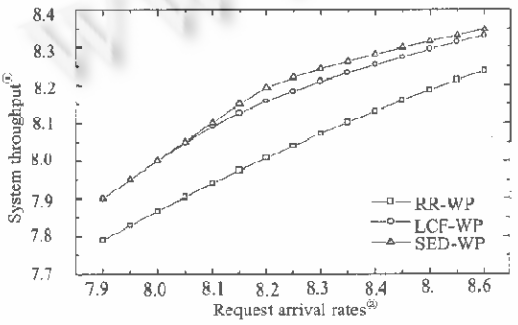
$$g(M(q_{ij}),k)=\frac{1}{k+1}\sum\Big(\prod_{(x\in E_j(k))\wedge(M(q_{ix})<b_{ix})}P[f(q_{ix})=f(q_{ij})]\cdot \prod_{(y\in\overline{E}_j(k))\wedge(M(q_{iy})<b_{iy})}P[f(q_{iy})>f(q_{ij})]\prod_{(y\in\overline{E}_j(k))}P[M(q_{iy})=b_{iy}]\Big). \quad (19)$$

3 数值结果

根据上述近似求解技术,我们使用C语言编写了WWW服务器集群系统性能分析的迭代求解程序,对不同的综合方案进行性能比较和分析,并使用随机Petri网的软件包SPNP[14](stochastic Petri net package)对模型直接精确求解,验证近似求解结果的有效性.

(1) 采用不同请求分配策略的综合方案性能比较

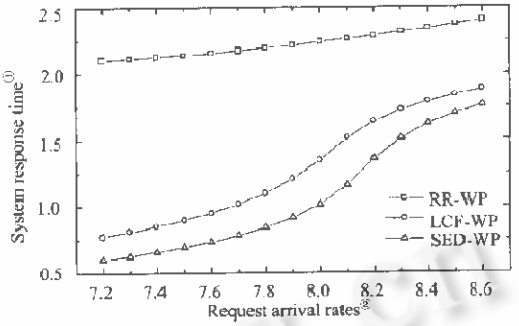
这个例子使用基于WP的综合方案来对比采用不同请求分配策略所带来的系统性能差异.仅考虑WWW服务器集群系统 $n=2,m=2$,共有4个逻辑队列的情况.参数设置如下:请求到达的速率之比 $\lambda_1/\lambda_2=3/2$;服务时间的权重 $r_1=2.0,r_2=3.0$;服务器的服务速率 $\mu_1=6.0,\mu_2=18.0$;请求的实时调度权重分别为 $\alpha_1=3.0,\alpha_2=1.0$;缓冲容量 $b_{11}=b_{12}=b_{21}=b_{22}=8$.这3种综合方案的近似求解的数值结果,包括系统吞吐量和响应时间,分别如图3和图4所示.



①系统吞吐量,②请求到达速率.

Fig. 3 Comparison among the WP-based integrated schemes for difference in throughput

图3 基于WP的综合方案之间系统吞吐量的数值比较



①系统响应时间,②请求到达速率.

Fig. 4 Comparison among the WP-based integrated schemes for difference in response time

图4 基于WP的综合方案之间系统响应时间的数值比较

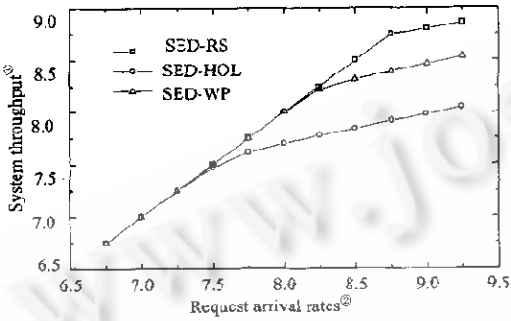
从图3和图4可以看到,在3种基于WP的综合方案中,采用SED分配策略可使系统获得最大的吞吐量和最小的响应时间;采用LCF方案次之;采用RR方案系统的吞吐量最小,而响应时间最大.数值结果证明,RR分配方案未考虑系统的固有特性,导致该方案下系统性能最差;LCF仅考虑了缓冲队列的状态,而未考虑服务器服务速率的影响;而SED兼顾了这两种系统特性,能使集群系统最大程度地并行操作,所以获得了系统性能的较优解.特别是当请求的输入速率接近系统的固有处理能力(输入速率为8.1附近)时,队列充分形成,SED方案的优越性表现得更为明显.采用其他组综合方案可以得到与此类似的结论.可见,SED是实现系统负载均衡的较好的分配方案.

(2) 采用不同请求选择策略的综合方案性能比较

本例采用基于SED的综合方案比较不同的请求选择方案给系统带来的性能差异.系统和参数设置同(1).3种综合方案对于系统吞吐量和响应时间的近似求解结果如图5和图6所示.从图中可见,在输入允许负载(非过度负载)的情况下,基于SED的3种综合方案中,采用RS选择策略可

© 中国科学院软件研究所 http://www.jos.org.cn

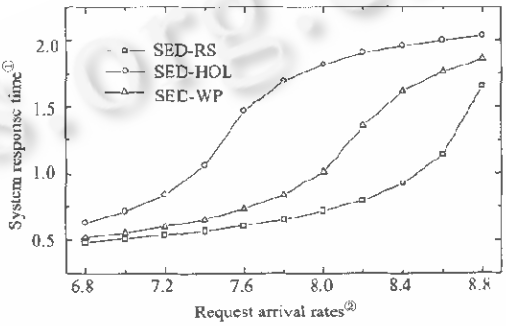
使系统获得最大的吞吐量和最小的响应时间;采用 HOL 策略,系统的吞吐量最小而响应时间最大;采用 WP 方案的情况介于以上二者之间.其他组综合方案可以得到与此类似的结论.数值结果表明,由于 HOL 方案总是优先保证高实时优先级请求的 QoS 要求,严重损害了低优先级请求的性能,从而降低了系统的整体性能指标.但 HOL 选择方案本身就是为了向高优先级请求提供 QoS 而设计的,因此能够为高优先级请求提供较好的实时性能.本例中请求 r_1(优先级最高)在不同选择方案下的响应时间对比情况见图 7.从图 7 可见,使用 HOL 的综合方案能够保证 r_1 的响应时间基本不变,实时性能最好;而使用 RS 的综合方案对高优先级请求的实时性能最差;使用 WP 的情况介于二者之间.



①系统吞吐量,②请求到达速率.

Fig. 5 Comparison among the SED-based integrated schemes for difference in throughput

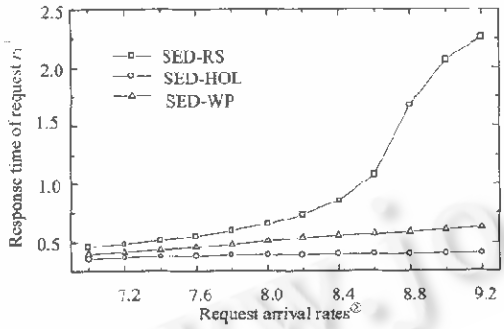
图 5 基于 SED 的综合方案之间系统吞吐量的数值比较



①系统响应时间,②请求到达速率.

Fig. 6 Comparison among the SED-based integrated schemes for difference in response time

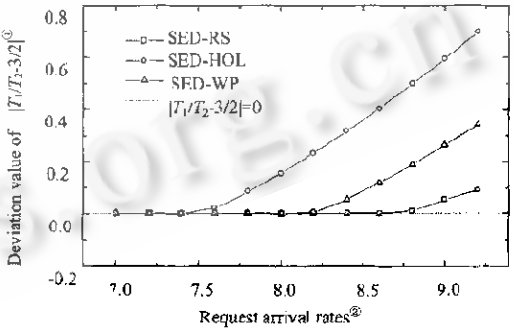
图 6 基于 SED 的综合方案之间系统响应时间的数值比较



①请求 r_1 的响应时间,②请求到达速率.

Fig. 7 Comparison among SED-based integrated schemes for difference in response time of r_1

图 7 基于 SED 的综合方案之间 r_1 响应时间的数值比较



①偏差值 $T_1/T_2-3/2$,②请求到达速率.

Fig. 8 Comparison among the SED based integrated schemes for difference in fairness

图 8 基于 SED 的综合方案之间公平性的数值比较

(3) 采用不同请求选择策略的综合方案的公平性比较

我们仍以基于 SED 的综合方案为例,比较不同的请求选择方案的公平性.根据式(16),公平性的比较准则是,当不同类型请求的吞吐量与其输入速率成正比时,选择方案的公平性较好.系统和参数设置与(1)相同,请求选择方案的公平性比较如图 8 所示.从图中可见,在采用 RS 的综合方案中,两类请求的吞吐量之比 T_1/T_2 与理想公平线 $\lambda_1/\lambda_2=3/2$ 的偏差程度最小,因此公平性较好;

© 中国科学院软件研究所 http://www.jos.org.cn

WP 次之;HOL 方案的公平性最差.不同的选择方案使队列的形成时刻有所不同,使图中曲线具有不同的偏离转折点,显示了不同方案间的公平性能差异.

应该指出,在上述各种方案近似求解的例子中,近似求解时整个系统状态数量是 36.而使用 SPNP 精确求解的系统状态数量是 6561,近似求解显著地简化了求解的复杂性,并明显减少了求解时间.通过与 SPNP 精确求解比较,我们发现近似迭代求解与精确解在系统响应时间上的相对误差最大为 7.5%左右,在系统吞吐量上的相对误差最大为 3.2%左右,可以说明,我们的近似求解结果相当精确.限于篇幅,数值结果的对比不再给出.

4 结 论

本文基于 3 种请求分配方案(RR,LCF 和 SED)和请求选择方案(RS,HOL 和 WP),提出并分析了 9 种面向 WWW 服务器集群系统请求分配和选择的综合调度方案,并提供了这些方案的随机高级 Petri 网模型,提出了一种近似性能分析技术,给出了这些方案的性能分析和数值结果的比较评价.

从性能分析的数值结果我们看到:综合方案采用 SED 能较好地实现集群系统的负载均衡;使用 RS 选择策略的综合方案具有良好的公平性能,但不支持高优先级请求的 QoS 要求;使用 HOL 选择策略的综合方案支持高优先级请求的 QoS 要求,但却严重损害了低优先级请求的性能,公平性较差.本文提出的加权优先级选择方案 WP 是对 RS 和 HOL 方案的折衷,同时兼顾了公平性和请求的 QoS 需求,实时调度的权重可以根据实际情况进行修改以满足客户的不同要求.由于目前 WWW 服务器集群系统要求能够支持电子商务类区分优先级的服务,并同时维护各类 Web 应用的公平性,根据本文的性能分析,我们认为,请求分配和选择的综合方案 SED-WP 能够较好地实现系统负载均衡,同时兼顾了不同请求的 WebQoS 需求和公平性,是满足电子商务类应用需求的较好的推荐方案.另外,本文提出的 WWW 服务器集群模型、请求分配和选择综合控制方案以及近似性能分析技术还适用于其他类似复杂系统的性能评价.

References:

[1] Bhatti, N., Friedrich, R. Web server support for tiered services. IEEE Network, 1999. 64~71.

[2] Colajanni, M., Yu, P. S., Cardellini, V. Dynamic load balancing in geographically distributed heterogeneous web servers. In: Proceedings of the 18th IEEE International Conference on Distributed Computing Systems (ICDCS'98). Amsterdam IEEE Computer Society, 1998. 295~302. http://dlib.computer.org/conferen/icdcs/8292/pdf/82920295.pdf

[3] Katz, E. D., Butler, M., McGrath, R. A scalable HTTP server: the NCSA prototype. Computer Networks and ISDN Systems, 1994,(28):155~164.

[4] Cisco Systems Inc. Load balancing: a multifaceted solution for improving server avalability. White paper, 2000. http://www.cisco.com/warp/public/cc/pd/cxsr/400/tech/lobal_wp.htm.

[5] Crovella, M. E., Carter, R. L. Dynamic server selection in the Internet. Technical Report, TR-95-014, Department of Computer Science, Bostor University, 1995. http://cs-www.bu.edu/faculty/crovdla/paper-archive/hpcs95/paper.html.

[6] Lin, Chuang. Performance analysis of request dispatching and selecting in web server clusters. Chinese Journal of Computers, 2000,23(5):500~508 (in Chinese).

[7] Lin, Chuang, Marinescu, D. C. Stochastic high-level Petri nets and applications. IEEE Transactions on Computers, 1988, 37(7):815~825.

[8] Paxsom, V., Floyd, S. Wide area traffic: the failure of poisson modeling. IEEE/ACM Transactions on Networking, 1995,3(3):226~244.

[9] Erramilli, A., Willinger, W., Lakshman, T. V., *et al*. Performance impact of self-similarity in traffic. In: Proceedings of

© 中国科学院软件研究所 http://www.jos.org.cn

Sigmetrics'95/Performance'95. Ottawa, Canada, 1995. 265～266. http://www.acm.org/pubs/contents/proceedings/metrics/223587/p265-erramilli.

[10] Ryu, B., Lowen, S. Point process approaches for modeling and analysis of self-similar traffic, Part II-Applications. In: Proceedings of the International Conference on Telecommunication Systems——Modelings, and Analysis, 1997. 62～71.

[11] Cunha, C. A., Bestavros, A., Crovella, M. Characteristics of WWW client-based traces. Technical Report, BU CS-95-010. Department of Computer Science, Boston University, 1995.

[12] Arli, F., Williamson, C. L. Internet web servers: workload characterization and performance implications. IEEE/ACM Transactions on Networking, 1997,5(5):631～645.

[13] Braun, H. W., Claffy, K. C. Web traffic characterization: an assessment of the impact of caching documents from NCSA's Web server. Computer Networks and ISDN Systems, 1995,28:37～51.

[14] Ciaodo, G., Muppala, J., Trivedi, K. S. SPNP: stochastic Petri net package. In: Proceedings of the Petri nets and performance models. Kyoto: IEEE Computer Society, 1989. 142～151.

附中文参考文献:

[6] 林闯. Web服务器集群的请求分配和选择的性能分析. 计算机学报,2000,23(5):500～508.

Integrated Schemes of Web Request Dispatching and Selecting and Their Performance Analysis*

SHAN Zhi-guang[1], DAI Qiong-hai[2], LIN Chuang[3], YANG Yang[1]

([1]*Information Engineering School, Beijing University of Science and Technology, Beijing* 100083, *China*);

([2]*Department of Automation, Tsinghua University, Beijing* 100084, *China*);

([3]*Department of Computer Science and Technology, Tsinghua University, Beijing* 100084, *China*)

E-mail: Shanzhiguang@ustb.edu.cn

http://www.ustb.edu.cn

Abstract: The Internet is undergoing substantial changes from a communication and browsing infrastructure to a medium for conducting business and services. These changes require that Web servers should support the preferential services such as those in E-commerce and also achieve good fairness among different types of requests. In this paper, some integrated schemes of request dispatching and selecting for Web server clusters is provided, with the goal of achieving load balance and meeting the requirements of both WebQoS and fairness. The stochastic high level Petri net (SHLPN) models for them are given. To cope with the well-known state space explosion problem, this paper a novel approximate analysis technique is proposed, which can significantly reduce the complexity of the model solution. Moreover, the numerical results of the performance analysis for those schemes are presented, and the best one among them is recommended that is suited to the preferential services and can achieve high system performance for Web server clusters.

Key words: Web server cluster; request dispatching; request selecting; performance analysis; stochastic high level Petri net (SHLPN)

* Received March 27, 1999; accepted June 15, 2000

Supported by the National Grand Fundamental Research Program of China under Grant No. G1999032707; the National Natural Science Foundation of China under Grant No. 69873012; the National High Technology Development Program of China under Grant Nos. 863-306-ZT05-01-02, 863-300-05-04-02-00

© 中国科学院软件研究所 http://www.jos.org.cn