基于收益驱动请求分类的多目标动态优先请求调度

陈梅梅

(东华大学旭日工商管理学院电子商务与物流系 上海 200051)

摘 要 请求调度通常需要在充分利用现有服务器资源的基础上满足响应时间最小化和系统吞吐量最大化的目标,但对于以盈利为目的的电子商务网站来说,关键还是要提高交易请求和VIP用户发起请求的达成率。针对电子商务网站请求调度的多重目标,首先提出了收益驱动的请求分类多维标准,在此基础上定义了请求优先级和调度优先级的概念,给出了基于请求分类的多目标动态优先调度算法MODP,并引入了基于事前过载判断而非负载测量的调度机制以避免控制延迟,有利于电子商务网站在多变的负载条件下自适应地实现差别服务和QoS保障。仿真实验证明了MODP机制与算法的有效性,将其与传统FCFS调度方法进行对比研究,结果表明:服务器无论在高载还是低载情况下,MODP调度策略在实现收益最大化、平均响应时间最小化的目标方面都具有明显的优势。

关键词 请求分类,请求优先级,请求调度,调度优化,商务网站

中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.8.040

Multi-objective Dynamic Priority Request Scheduling Based on Reward-driven Request Classification

CHEN Mei-mei

(Department of Electronic Business, Glorious Sun School of Business and Management, Donghua University, Shanghai 200051, China)

Abstract The general target of request scheduling is to maximize throughput and minimize response time under the condition that existing system resource is in full use. But for a busy business Web system with the goal of revenue generation, it is crucial to increase the completion rate of transaction related requests and requests from VIP. Aiming at the multiple objectives of request scheduling for business Web server, the multi-dimension criterion for reward-driven request classification was firstly presented. Then, based on the definitions of request priority and scheduling priority, the algorithm of multi-objectives dynamic priority scheduling was proposed, which can provide DiffServ and QoS guarantee adaptively for business Web system under the variety workload. At the same time, the dynamic scheduling mechanism was introduced based on one-step-ahead overload estimation instead of the workload measurement to avoid the control delay. Simulation experiment shows the validity of this scheduling mechanism and algorithm. Through the comparison of the completion rate of transaction requests as well as the average response time with that of the traditional method FCFS, MODP proves its preferential principle under not only lower workload but also overload condition.

Keywords Request classification, Request priority, Request scheduling, Scheduling optimization, Business Web server

1 引言

基于精心的容量规划而构建的电子商务系统中,服务器多数情况下处于低载状态[1],由于事件营销导致的可预期的过载可以暂时利用云服务技术有效应对,但由于服务器负载的异步性和突发性等特点导致的瞬时过载却是不可避免且难以预测的[2],而采用扩容或者在服务器集群中增加同质服务器以实现负载均衡等方法来解决瞬时过载问题,对于多数企业来说显然并非经济可行的方案。

电子商务网站,特别是零售型网站,在交易高峰时段的HTTP请求到达率至少超过平均值8~10倍,致使系统吞吐量下降、响应时间过长,影响用户体验,甚至因系统崩溃导致商务网站直接的经济损失[3]。而Web系统的服务器目前普遍采用尽力而为的服务模式和先来先服务(First Come First Service, FCFS)的调度方法,不具备任何过载控制机制和相应的QoS保障能力[4]。因此,对于准许进入系统等待处理的请求,研究有效的请求调度机制对提高系统性能具有重大现实意义。

一般地,请求调度的目标是:在充分利用系统现有资源的条件下,确保系统吞吐量最大且用户请求的响应时间最小。但在随机突发的高峰访问时段,单纯追求性能目标最优会以部分地牺牲企业盈利目标为代价,从而导致诸如交易请求被系统放弃而优先处理冲浪者的浏览请求等情况。而对于一个繁忙的商务网站,更重要的是要提高交易请求的达成率,以满

到稿日期:2015-05-06 返修日期:2015-08-16

陈梅梅(1970—),女,博士,副教授,主要研究方向为电子商务系统性能智能优化与评价、网络消费行为、神经营销学。

足电子商务业务的盈利目标。因此本文提出收益驱动的多目标请求调度思想，其核心是：在服务器过载的情况下，对于已准入系统排队等待处理的用户请求，根据一系列调度原则确定每一请求的最优序列解，不计较某一资源或个体用户响应时间的得失，在提高服务器整体性能的同时使网站收益最大化。

常见的 Web 服务器请求调度算法有很多，但都存在着各自的缺陷。Cherkasova 提出的最短工作优先调度（Shortest Job First，SJF）可有效缩短静态内容请求的平均响应时间[1]，但会导致“大”请求的“饥饿”现象。Schroeder 等证明了最短剩余进程时间优先法（Shortest Remaining Processing Time First，SRPT）能有效实现平均响应时间最小化[5]，但其只适用于宽带网络条件下瞬时超载时静态请求的调度。Jordi Guitart 等提出基于会话的动态负载相互平衡算法（Dynamic Weighted Fairing Sharing，DWFS）用于确保对高优先权任务提供高水平服务[6]，但这一歧视性措施最终导致完整会话数目增加条件下的低系统吞吐量。文献[4]考虑了商务网站的赢利目标，提出了基于请求区分的混合优先级调度策略即给予交易请求绝对优先级，采用传统的 FIFO（First In First Out）调度策略以提高带来收益请求的达成率，并在过载时对浏览请求采用后进先出 LIFO（Last In First Out）的调度机制以提高系统吞吐量，仿真实验表明该策略在过载情况下能大大提高交易请求的达成率。但 Doshi 和 Heffes 研究认为：LIFO 在响应时间方面波动性较大，从而导致其请求丢失率比 FIFO 的更大[8]。文献[8]没有考虑平均响应时间和请求丢失率两项重要的性能指标，并且过载判断也只是基于 CPU 利用率单一指标的实际测量值。此外，文献[7]对于如何确定基于请求分类的优先级和表示队列中请求重要程度的效用值等关键问题也没有提出有效的方法。

本文针对商务网站最优调度的多重目标，首先提出了基于收益驱动的请求分类多维标准；其次提出了基于多维请求分类标准的多目标动态优先请求调度算法，同时引入了基于事前过载判断的调度机制以期解决调度延迟的问题；最后通过仿真实验验证了该请求调度算法与机制的有效性。

2 收益驱动的请求分类

文献[9]提出了基于请求内容优先级和请求到达时间与处理时间的调度策略，但该调度策略没有考虑重要用户发出请求和交易相关动态请求的优先级，不适用于解决商务网站请求的调度问题。文献[10]提出了根据请求发起用户的角色重要性和请求操作类型的二维请求分类标准，并实现了应用获益的 Web Service 请求调度策略。但作为以实现交易请求完成率最大为目标的商务网站来说，请求操作类型不能仅笼统地分为关键请求和浏览请求，需要根据请求内容进一步细分，这样才有利于具有交易转化可能性的浏览请求的优先调度。本文采用多维标准对请求进行分类，具体如下。

2.1 基于请求操作类型的区分标准

根据事务处理性能委员会（Transaction Processing Performance Council）发布的第一个以支持电子商务活动的评估为目标的基准测试程序 TPC-W，准许进入系统等待处理的请求首先被区分为基本的两大类：浏览请求和交易请求。浏览请求主要指非交易过程中提交的请求，包括用户进入主页、分类浏览、站内查询、浏览商品详细信息等请求。而交易请求是指用户从进入登录页面开始，围绕交易过程，最后以“确定”按钮作为结束，整个过程中所提交的一系列请求，一般包括登录、购物车、购买请求和购买确认、支付等动态请求。

对于交易请求，系统给予绝对的优先权；而对于浏览请求，则需要根据发起请求的用户类型、请求的文件类型和请求的内容进一步细分，以确定请求的优先级。

2.2 基于请求发起用户的区分标准

各用户类型划分标准和静态优先权如表 1 所列[11]。

表 1 用户类型及其优先级权重系数

请求发起的用户类型	划分标准	优先级权重系数
重要用户（Primary user）	交易频率高、交易量大、利润贡献率高的 VIP 用户	3
普通用户（Ordinary user）	一般的注册用户	2
匿名用户（Anonymous user）	以浏览为主要目的的非注册用户	1

2.3 基于请求文件类型的区分标准

Martin F 等人将请求按文件类型分为 7 种，并对来自于 Saskatchewan、NASA、ClarkNet 等教育、科研和商务领域的 6 组服务器日志文件 1 个月到 1 年的统计数据进行了分析[10]，统计数据如表 2 所列。

表 2 6 组 Web 服务器访问日志统计数据

Data Sets	NASA		ClarkNet		Waterloo		Calgary		Saskatvhewan		NCAS	
Types	p_{1j}	l_{1j}	p_{2j}	l_{2j}	p_{3j}	l_{3j}	p_{4j}	l_{4j}	p_{5j}	l_{5j}	p_{6j}	l_{6j}
HTTP	30.7	18.8	19.9	15	38.7	35	47.1	13.2	55.6	50.7	51.1	51.1
Images	63.5	48.8	78	76.6	50.1	18.9	50.3	50.2	36.5	36.5	48.1	36
Sound	0.2	1.1	0.2	2.4	0.01	0.1	0.1	1.3	0.1	1.5	0.2	3.5
Video	1	29.7	0.007	2.4	0.0006	0.1	0.3	11.4	0.004	2.6	0.1	6.2
Dynamic	2.6	0.3	1.2	0.8	0.3	0.2	0.04	0.01	6.7	4.4	0.01	0.06
Format	0.01	0.07	0.01	0.04	3.7	25.2	1	21.7	0.02	0.1	0.006	0.2
Other	1.99	1.93	0.683	2.76	7.184	20.5	1.16	2.19	1.076	4.1	0.484	2.94

各类型文件请求在用户请求流中的重要程度可根据各文件类型请求占所有请求的概率 p_{ij} 和导致负载的比重 l_{ij} 两项指标计算得到，具体算法如下。

Step1 分别计算每组数据中各类型请求的权重系数 $w_{ij}=p_{ij}/l_{ij}$，$i=1,2,\cdots,n$，$j=1,2,\cdots,m$；

Step2 得到各类型请求的权重系数 $w_j=\sum_{i=1}^{n}w_{ij}/n$，$n=6$；

Step3 经归一化得到不同文件类型请求的静态权重系数 $c_j=w_j/\sum_{j=1}^{m}w_j$，$m=7$。

可得 7 种文件类型请求的权重系数如表 3 所列。

表3　7种请求文件类型及优先级权重系数

请求类型	文件类型	w_i	c_j
HTTP	.html,.htm,.shtml,.map 等	1.62	2.47
Images	.gif,.jpg,.jpeg,.xbm,.bmp 等	1.39	2.11
Sound	.au,.snd,.wav,.mid,.midi,.lha,.aif 等	0.09	0.14
Video	.mov,.move,.avi,.qt,.mpeg,.mpg 等	0.01	0.02
Dynamic	.cgi,.pl,.asp,.jsp,.php 等	2.89	4.4
Format	.ps,.eps,.pdf,.doc,.dvi 等	0.14	0.21
Other	其余文件类型	0.43	0.65

2.4　基于请求内容的区分标准

对于HTTP请求，还可根据请求的内容对其重要性进行判断，如浏览商品信息、购物指南等内容的请求相对于浏览参考页的请求应该被给予更高的优先处理权，支付请求相对于交易确认请求应该被给予更高的优先处理权。对于动态请求，除了交易请求外还有查询请求，相对于分类浏览，查询请求针对性更强，交易转化的可能性更大，因而应该被优先处理。

3　多目标请求优化调度的原则

原则一：带来收益的重要请求优先，提高交易请求达成率。

确保那些已经在购买进程中的交易请求得到及时的处理而不出现丢弃的情况，尤其是在过载情况下更应确保用户顺利完成整个交易。所以，应给予交易请求绝对的优先级，除非当交易请求将导致CPU过载而磁盘空间还有空闲时。而对于交易请求，再根据请求的文件类型和内容进一步判断其优先级，越是接近交易确认的请求，其优先权重越高。

原则二：交易转化可能性高的浏览请求优先，提高系统整体吞吐量。

浏览请求的及时处理有利于源源不断地输送潜在的交易请求，这也是整体吞吐量最大化目标的要求。但如果仅仅给予交易请求静态的绝对优先级，则可能会导致低优先级的浏览请求几乎得不到处理。因此对于浏览请求，可按转化为交易请求的可能性确定其优先级，确保那些已有明显购买意向或向交易请求转化的可能性较大的浏览请求得到及时处理。

原则三："小"请求优先，缩短系统平均响应时间。

如果采用最短作业优先(Shortest Job First，SJF)调度方法，由于"小"请求按SJF策略在"大"请求之前得到Web服务器的响应处理，可高效地缩短平均响应时间。如图1所示的两种调度方法中，采用FCFS调度策略请求队列的平均响应时间是(3000+3100+3130)/3≈3077(时间单位/每个请求)；而采用SJF调度策略请求队列的平均响应时间是(30+1300+3130)/3≈1097(时间单位/每个请求)。

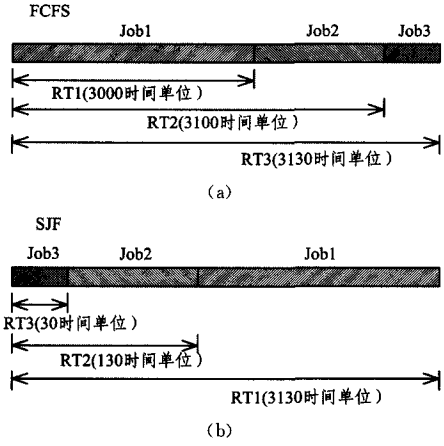

图1　两种不同的请求调度策略的实例

原则四：考虑请求到达时间，避免出现"大"请求"饥饿"现象。

单独使用SJF调度策略会由于"大"请求总是因优先权小而长时间没有机会得到Web服务器的响应处理。为同时满足响应时间最小化、吞吐量最大化和丢失率最小化三重性能目标，调度策略需要同时考虑请求到达时间，避免"大"请求超时。

4　多目标动态优先调度算法

4.1　基于请求分类的请求优先级

针对电子商务系统请求调度的多重目标和主要原则，在收益驱动的请求区分的多维标准的基础上，首先采用双队列策略，经过请求区分之后的浏览请求和交易请求将进入相应的队列并分别进行请求优先级的计算。

定义1　对于任意请求$R(i)$，其请求的文件大小是$FileSize(i)$，请求到达的时间为$TimeArrived(i)$，请求发起用户类型的权重系数是$UserType$，请求文件类型的权重系数是$Filetype$，请求内容的权重系数是$ReqContent$。

据Sameh Elnikety等人的研究结果，一个特定的动态请求导致的系统负载通常是一定的，也就是说，可以通过请求执行的脚本程序如Servlet等得到表示其执行成本的运行时间[4]。则请求$R(i)$优先级的计算公式如下：

$$P_t(i)=\alpha\times UserType+\beta\times FileType+\delta\times ReqContent+\varepsilon\times(T-TimeArrived(i))/\varphi\times RunningTime(i) \quad (1)$$

$$p_b(i)=\alpha\times UserType+\beta\times FileType+\delta\times ReqContent+\varepsilon\times(T-TimeArrived(i))/\omega\times FileSize(i) \quad (2)$$

其中，T为当前系统时钟；$\alpha,\beta,\delta,\varepsilon,\omega$和$\varphi$为权重参数，可根据系统负载预测结果进行调整，以达到动态优先调度的目的。

根据定义1中请求优先级的计算方法可知，当过载将发生时，电子商务系统请求调度的多重目标可通过以下方式实现。

(1)通过上调参数α,β,δ的值，给予VIP用户发出的重要内容和重要文件类型的请求相对更高的优先权；

(2)通过下调参数φ和ω的值，确保运行时间短或访问的文件的"小"请求优先得到处理，以缩短平均响应时间；

(3)同时通过上调参数ε值，使先到达的"大"请求也有被处理的机会，从而避免"小"请求优先导致的"大"请求"饥饿"现象的发生。

系统每一时间间隔都分别对两个请求队列中还没得到处理的请求和新到达的请求按照动态调整的参数重新计算其请求优先级。请求$R(i)$在相应的队列中按请求优先级$p_b(i)$或$p_t(i)$进行排序并等待进一步处理。

4.2　基于负载的动态调度优先级

请求的调度并不是按照请求优先级$p_b(i)$或$p_t(i)$进行的，而是采用二维优先级机制，即根据请求优先级再次计算得到的调度优先级进行处理，目的是方便根据事前过载判断的结果，分别对请求优先级和调度优先级进行动态调整。

定义2　$P_T(i)$和$P_B(i)$分别表示交易请求队列和浏览请求队列按各自的$p_t(i)$或$p_b(i)$排序后得到的请求$R(i)$的优先顺序号。则请求的调度优先级计算公式如下：

$$P(i)=\begin{cases}P_T(i)-k, & 0<k<1, i\in\{T\}\\ P_B(i)\times h, & h=1,2,\cdots,\infty, i=\{B\}\end{cases} \quad (3)$$

其中，$\{T\}$和$\{B\}$分别表示交易请求队列和浏览请求队列，k和h为调度调节因子，参数h表示交易和浏览请求同时存在时处理h个交易请求后处理1个浏览请求。

显然，根据定义2中调度优先级计算方法可知，电子商务系统请求调度的收益目标是通过以下方式实现的。

(1)相对于浏览请求队列，交易请求队列始终具有绝对的优先权；

(2)设$h=1$，表示交易和浏览请求同时存在时，系统将轮流从两个队列中各取一个请求插入到调度优先队列中依次处理；

(3)当系统过载时，设$h>1$，表示交易和浏览请求同时存在时，先从交易请求队列中依次处理h个交易请求，再处理1个浏览请求，以满足交易达成率最大化目标。

4.3 基于事前过载判断的动态优先调度机制

因为浏览请求大多为静态内容的请求，是一种典型的服务器磁盘重载应用，而以动态内容为主的交易请求则是CPU重载应用[11]，所以需要根据基于预测的事前过载判断的结果，针对系统不同的负载情况，通过动态调整请求优先级计算公式中的相关权重参数$\alpha,\beta,\delta,\varepsilon,\omega$和$\varphi$以及调度优先级计算公式中的调度调节因子$k$和$h$，来取得最优的调度效果。

动态优先调度的原则具体如下：

(1)当预测到将发生CPU过载而磁盘低载或空闲时，则提高浏览请求队列的优先级，同时减少交易请求的处理量，从而避免CPU过载并充分利用空闲的磁盘资源；

(2)当预测到将发生磁盘过载而CPU大部分时间都在等待磁盘操作时，则减少浏览请求处理量，从而避免磁盘过载的发生；

(3)当预测二者均可能过载时，则给予交易请求队列绝对优先权且提高小交易请求优先级，同时反馈系统启动自适应内容降级服务处理浏览请求或控制请求的准入速率，以避免系统过载；

(4)在系统低载时，两队列请求具相同优先级，系统将轮流从两个队列中按优先级依次将请求调入系统处理。

基于请求分类和事前过载判断的电子商务系统的多目标动态优先级请求调度的具体流程如图2所示。

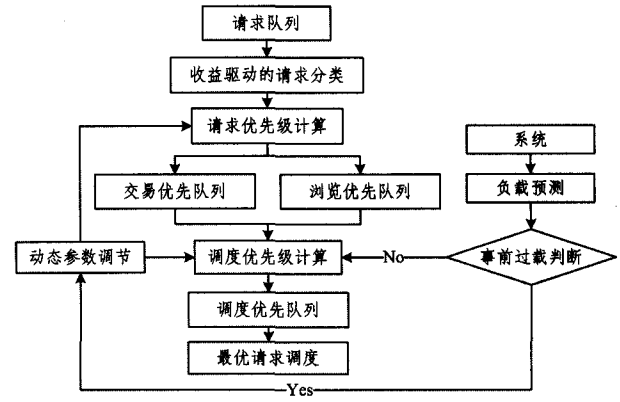

图2 基于请求分类和事前过载判断的多目标动态优先调度的流程

相对于目前各主要请求优先调度机制，本文提出的多目标动态优先调度MODP机制主要具有如下特点：

(1)双队列策略和二维优先级定义有利于实现电子商务系统请求调度的多重目标；

(2)算法简单，有利于提高请求调度实时性；

(3)基于事前过载判断的动态优先级机制有利于提高调度的精确度。

各主要优先调度机制的对比情况如表4所列。

表4 各主要优先调度机制的特点

调度机制	优先级影响因素	特点
FCFS	到达时间	公平性好，响应时间长
SJF	文件大小	适用于静态请求，存在饥饿现象
SRPT	最短剩余运行时间	适用于动态请求，存在饥饿现象
DPSP[7]	请求的文件类型、到达时间、处理时间	①有利于重要文件类型请求及时处理 ②解决了饥饿现象 ③对不同类型请求均采用处理时间的做法欠合理
MODP	请求的操作类型(交易、浏览)、发起用户、文件类型、内容、任务大小(运行时间或文件大小)、到达时间、系统负载情况	①基于多维请求分类，有利于交易请求或重要用户发起的、重要文件类型、重要内容请求及时得到处理，适用于赢利性电子商务系统请求调度 ②小任务优先提高平均响应时间 ③解决了饥饿现象 ④针对动态内容为主的交易请求和静态内容为主的浏览请求，分别采用处理时间和文件大小作为任务大小衡量依据的做法更合理且便于实现；根据负载情况动态调节

5 仿真实验

5.1 实验说明

仿真实验是在PC环境下进行的，利用Matlab7.3模拟实现了各类请求的到达，根据负载判断信号调整参数并计算优先级和基于优先级的请求调度，通过在不同负载情况下与传统的先来先处理(FCFS)请求调度策略进行对比研究，检验基于请求分类和过载判断的多目标动态优先级请求调度算法(MODP)对平均响应时间、吞吐量和交易请求达成率等性能改进的有效性。

在短时间内，请求的到达时间符合泊松分布，请求类型利用字母组合代码由程序随机产生，其中静态请求占所有请求的90%以上[10]。

对于以静态请求为主的浏览请求，其任务大小由请求的文件大小决定，80%的文件类型请求在[100,100000]范围内，不到10%的请求文件大于100000字节，最大不超过10000000字节[10]。在随机生成该类请求的同时，根据文件大小的区间范围及所占比例的规律，随机给定一个值作为该请求的任务大小$FileSize(i)$，用以根据任务大小计算请求的优先级。

对于以动态请求为主的交易请求，其任务大小由请求的运行时间决定。根据文献[4]所示的13类servlet应用程序在低载和重载情况下的平均执行时间的区间及各类程序执行所占比例，随机给定一个值作为该请求的运行时间*Running*-

$Time$，用以根据任务大小计算请求的优先级。

参数 $\alpha,\beta,\delta,\varepsilon,\omega$ 和 φ 的初始值均设置为 1，$h=1$，$k=0.5$，表示交易和浏览请求同时存在时，轮流从两个队列中按请求优先级依次将请求调入系统处理。根据不同负载状态进行调整，例如当负载判断信号为重载时，$h=2$ 且 $\varphi<1$，$\alpha,\beta,\delta,\varepsilon,\omega=2$ 等，表示交易和浏览请求同时存在时，给予交易请求队列绝对优先权，提高小交易请求优先级，以及重要用户、重要文件类型的优先级。

每一时刻根据负载信号调整相应的参数值，根据式(1)或式(2)重新计算还没处理和新到请求的优先级 $p_b(i)$ 或 $p_t(i)$ 以及排序后得到的请求 $R(i)$ 的优先顺序号 $P_T(i)$ 和 $P_B(i)$；同时根据式(3)计算得到当前队列中每一请求的调度优先级 $P(i)$。

将按规定的时间周期统计完成的请求总量作为吞吐量，计算平均响应时间，根据请求类型代码分析其中交易请求达成率等性能指标。

5.2 实验结果分析

仿真实验结果如表 5 及图 3—图 7 所示。

表 5 仿真实验结果

Comparison Index	Workload	MODP	FCFS	Description
Completion rate of trans. req.	overload	0.681	0.545	+0.136
	lower	0.802	0.727	+0.075
Completed trans. reqs. / total reqs.	overload	0.911	0.506	+0.405
	lower	0.803	0.497	+0.306
Completion rate of bro. req.	overload	0.117	0.534	−0.417
	lower	0.324	0.730	−0.406
Average response time	overload	1.729	1.823	−0.094
	lower	1.673	1.889	−0.216

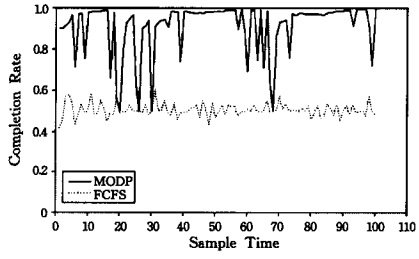

图 3 高载时所有请求中交易请求占比的对比

由图 3 可见，在高载情况下，采用 MODP 策略时已完成的交易请求占所有已完成请求的 91.1%，体现了该策略的优先原则，而采用 FCFS 时则为 50.6%，体现了该策略的公平原则。

如图 4 和图 5 所示，相比 FCFS，高载情况下 MODP 交易请求达成率提高了 13.6%，而低载情况下仅提高了 7.5%。

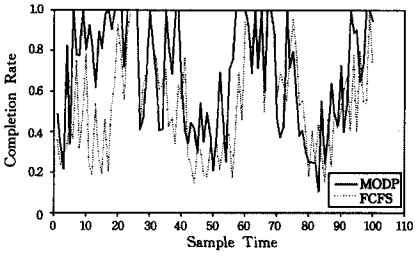

图 4 高载时交易请求达成率的对比

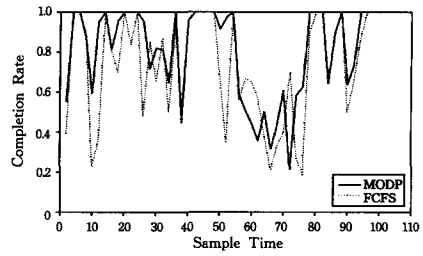

图 5 低载时交易请求达成率的对比

如图 6 所示，采用 MODP 和 FCFS 策略，在高载时浏览请求达成率分别下降了 20.7%和 19.6%；但图 4 显示，采用 MODP 在高载情况下却比采用 FCFS 时交易请求达成率提高了 12.1%；如图 7 所示，采用 MODP 策略比采用 FCFS 时平均响应时间降低，高载时，MODP 比 FCFS 下降约 0.1 个时间单位，低载时下降约 0.3 个时间单位。

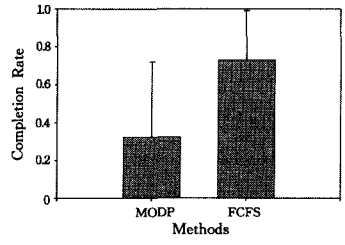

图 6 高载时浏览请求达成率

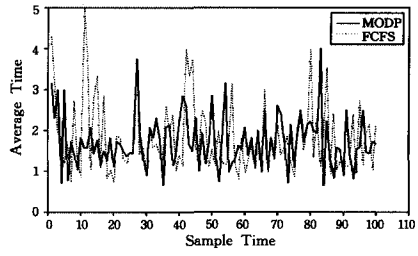

图 7 高载时平均响应时间

5.3 实验结论

通过对比完成的交易请求占所有已完成请求的比例和交易请求达成率，证明了高载时 MODP 的优先原则。

通过浏览请求达成率的对比，在高载时无论采用何种策略浏览请求达成率都下降约 20%，但采用 MODP 策略时交易请求达成率却比采用 FCFS 时提高了 12.1%，这证明了 MODP 策略在实现收益最大化目标方面的有效性；但同时也说明，在高载情况下交易请求的达成率提高是以牺牲浏览请求达成率和吞吐量为代价的。

通过平均响应时间的对比看出，MODP 策略在高载时仍没有出现平均响应时间延长的情况，其反而还略有降低，这充分证明了该调度策略在响应时间最小化目标达成方面的有效性。

结束语 有效的电子商务系统的请求调度方案应该不仅能够避免活锁现象的发生，还应充分利用有限的系统资源，尽可能多地完成电子商务主体认为有价值的请求。本文针对电子商务系统请求调度的多重目标，首先提出了收益驱动的请求区分的多维标准，通过请求类型、用户类型、文件类型和请求内容来区分准入系统的请求，并给出了不同文件类型请求权重计算方法；在界定了请求优先级和调度优先级这两个概

(下转第 222 页)

柴玉梅,张卓,王黎明. 基于频繁概念直乘分布的全局闭频繁项集挖掘算法[J]. 计算机学报,2012,35(5):990-1001

[3] Missaoui G R, Alaoui H. Incremental concept formation algorithms based on Galois (concept) lattices[J]. Computational Intelligence,1995,11(2):246-267

[4] Bêlohlávek R. Fuzzy Galois Connections[J]. Mathematical Logic Quarterly,1999,45(4):497-504

[5] Belohlavek R. Reduction and a simple proof of characterization of fuzzy concept lattices[J]. Fundamenta Informaticae,2001,46(4):277-285

[6] Belohlavek R. Algorithms for fuzzy concept lattice[C]// Proceedings of the 4th International Conference on Recent Advances in Soft Computing. Nottingham, United Kingdom,2002:200-205

[7] Belohlavek R, De Baets B, Outrata J, et al. Computing the Lattice of All Fixpoints of a Fuzzy Closure Operator[J]. IEEE Transactions on Fuzzy Systems,2010,18(3):546-557

[8] Lindig C. Fast Concept Analysis[C]// Working with Conceptual Structures,2000. Aachen: Shaker Verlag,2000:152-161

[9] Pôcs J. Note on generating fuzzy concept lattices via Galois connections[J]. Information Sciences,2012,185(1):128-136

[10] Ghosh P, Kundu K, Sarkar D. Fuzzy graph representation of a fuzzy concept lattice[J]. Fuzzy Sets and Systems, 2010, 161(12):1669-1675

[11] Pei D, Li M Z, Mi J S. Attribute reduction in fuzzy decision formal contexts[C]// International Conference on Machine Learning and Cybernetics (ICMLC). IEEE Press: New York,2011:204-208

[12] Aswanikumar C, Srinivas S. Concept lattice reduction using fuzzy K-Means clustering[J]. Expert Systems with Applications, 2010,37(3):2696-2704

[13] Li L, Zhang J. Attribute reduction in fuzzy concept lattices based on the T implication[J]. Knowledge-Based Systems, 2010, 23(6):497-503

[14] Pang J, Zhang X, Xu W. Attribute Reduction in Intuitionistic Fuzzy Concept Lattices[J]. Abstract and Applied Analysis, 2013,2013(54):1-13

[15] Zhang Lei, Zhang Hong-li, Yin Li-hua, et al. Theory and Algorithms of Attribute Decrement for Concept Lattice[J]. Journal of Computer Research and Development, 2013, 50(2):248-259 (in Chinese)
张磊,张宏莉,殷丽华,等. 概念格的属性渐减原理与算法研究[J]. 计算机研究与发展,2013,50(2):248-259

[16] Zhang L, Zhang H, Shen X, et al. An Incremental Algorithm for Removing Object from Concept Lattice[J]. Journal of Computational Information Systems,2013,9(9):3363-3372

[17] Zhang Zhou, Chai Yu-mei, Wang Li-ming, et al. A parallel algorithm generating fuzzy formal concepts[J]. Pattern Recogniton and Artificial Intelligence,2013,26(3):260-269(in Chinese)
张卓,柴玉梅,王黎明,等. 模糊形式概念并行构造算法[J]. 模式识别与人工智能,2013,26(3):260-269

[18] Zhang Zhuo, Du Juan, Wang Li-ming. Load balance-based algorithm for parallelly generating fuzzy formal concepts[J]. Control and Decision,2014,29(11):1935-1942(in Chinese)
张卓,杜鹃,王黎明. 基于负载均衡的模糊概念并行构造算法[J]. 控制与决策,2014,29(11):1935-1942

[19] Frank A, Asuncion A. UCI machine learning repository [EB/OL]. http://www.ics.uci.edu

(上接第 203 页)

念的基础上,采用双优先队列模型分别计算浏览请求和交易请求的优先级,提出了基于请求分类和事前过载判断的动态优先级机制和调度策略,并通过仿真实验将其与传统 FCFS 调度机制进行了对比研究,结果表明,在电子商务服务器高载情况下,提出的多目标动态优先调度算法在有效提高交易请求达成率的前提下平均响应时间有所降低。下一步将深入研究基于预测的事前过载判断。

参 考 文 献

[1] Loh S, Wives L K, de Oliveira J P. Concept-based knowledge discovery in texts extracted from the Web[J]. SIGKDD Explorations,2000,2(1):29-39

[2] Schroeder B, Harchol-Baher M. Web servers under overload: how scheduling can help[J]. ACM Trans. on Internet Technology,2006,6(1):20-52

[3] Menasce D A, Almeida V A F, Riedi R, et al. In search of Invariants for e-business workload[C]// Proceedings of 2000 ACM Conference in E-commerce, Minneapolis. MN,2000:17-20

[4] Elnikety S, Nahum E, Tracey J, et al. A Method for Transparent Admission Control and Request Scheduling in E-commerce Web Sites[C]// Proceedings of the 13th International Conference on World Wide Web. New York, USA: ACM Press, May 2004:276-286

[5] Cherkasova L. Scheduling strategy to improve response time for web applications[C]// Proceedings High Performance Computing and Networking. Amsterdam, Apr. 1998:305-314

[6] Guitart J, Carrera D, Beltran V, et al. Session-based adaptive overload control for secure dynamic web applications[C]// Proceedings of the 2005 International Conference on Parallel Processing. Oslo, Norway, Jun. 2005:341-349

[7] Singhmar N, Mathur V, Apte V, et al. A combined LIFO-priority scheme for overload control of e-commerce Web servers[C]// International Infrastructure Survivability Workshop. Lisbon, Portugal,2006:5-8

[8] Doshi B T, Heffes H. Overload performance of several processor queuing disciplines for the M/M/1 queue[J]. IEEE Transactions on Communications,1986,34(6):538-546

[9] Cao Lin-qi, Xiao Xiao-qiang, et al. DPSP: A client request scheduling policy based on Web content[J]. Journal of Computer Research and Development,2002,139(12):142-147(in Chinese)
曹林奇,肖晓强,等. DPSP:一种基于内容的客户请求调度策略[J]. 计算机研究与发展,2002,139(12):142-147

[10] Martin F A, Carey L W. Web server workload characterization: The search for invariants[C]// Proceedings of ACM SIGM ETR ICS 1996. Philadelphia,1996:126-137

[11] Guan He-qing, Zhang Wen-bo, et al. An application-aware web service requests scheduling strategy [J]. Chinese Journal of Computers,2006,29(7):1189-1198(in Chinese)
官荷卿,张文博,等. 一种应用敏感的 Web 服务请求调度策略[J]. 计算机学报,2006,29(7):1189-1198