

基于服务质量需求的存储系统动态优先权调度策略

李中¹ 王刚² 刘璟²¹(天津航海仪器研究所 天津 300131)²(南开大学信息技术科学学院 天津 300071)

(createform@163.com)

Quality of Service Aware Dynamic Priority Scheduling Scheme for the Mixed Class Multimedia Workloads in the Storage Systems

Li Zhong¹, Wang Gang², and Liu Jing²¹(Tianjin Navigation Instrument Research Institute, Tianjin 300131)²(College of Information Technology Science, Nankai University, Tianjin 300071)

Abstract QADPS (QoS-aware dynamic priority scheduling scheme), a new scheduling scheme for meeting the different QoS requirements of multimedia applications is introduced in this paper. When retrieving data from the storage systems, multimedia applications need the guarantee of QoS. The fraction of requests whose response time exceeds a specific delay limit must be below a certain proportion. Different multimedia applications may have different QoS requirements in the mixed multimedia workloads. Based on the history of a multimedia application receiving service, its QoS failed-distance can be calculated. The basic idea of QADPS is to assign higher priority to the multimedia applications that have shorter failed-distance. Using this scheme, the requests of the multimedia applications with stricter QoS requirements can achieve more chances of service applied by the storage systems. The effectiveness of this approach is evaluated through simulation. Compared with the conventional scheme, QADPS substantially increases the number of the concurrent multimedia applications.

Key words storage system; multimedia applications; quality of service; dynamic priority; scheduling scheme

摘要 多媒体应用要求存储系统提供保证服务质量的数据访问服务,响应时间超过延迟上限的访问请求必须小于一定的比例。基于服务质量需求的动态优先权调度策略(QADPS)根据多媒体应用接受存储系统服务的历史信息,计算多媒体应用的服务质量失败距离;并为失败距离较小的多媒体应用分配较高的调度优先权。QADPS根据多媒体应用的不同服务质量需求,实现了有差别的调度:多媒体应用的服务质量需求越严格,它的访问请求被调度的机会越高。QADPS能够在保证服务质量的前提下,支持更多的并发多媒体应用。

关键词 存储系统;多媒体应用;服务质量;动态优先权;调度策略

中图分类号 TP333

多媒体应用要求存储系统提供保证服务质量的数据访问服务:一方面希望存储系统在特定的响应时间上限内完成对访问请求的服务;另一方面又能够忍受小于一定比例的访问请求的响应时间超过

收稿日期:2005-01-19;修回日期:2005-11-04

基金项目:国家自然科学基金项目(60273031);高等学校博士点专项科研项目(20020055021);天津市科技发展计划重点基金项目(043800311)

上限,同时要求这些超时的访问请求均匀分布.这就是多媒体应用的统计学服务质量需求^[1].具有不同服务质量需求的多媒体应用并发访问存储系统,要求存储系统实现针对不同服务质量需求的调度策略.

磁盘阵列是存储系统的主流设备.磁盘系统自身的调度算法,如SCAN,LOOK,SSTF等,无法感知不同的服务质量需求^[2];实时调度策略,如EDF(earliest deadline first)^[3]和LST(latest start time)^[4]等,也不能保证服务质量.

一类适用于多媒体应用的存储系统调度策略基于循环调度(cycle-based scheduling),如GSS(ground sweeping scheme)^[5],T-SCAN(transformation-scan)^[6],BSCAN(batched SCAN)^[7],BIDS(buffer-inventory-based dynamic scheduling)^[8]和BubbleUP^[9]等,这类策略要求所有多媒体应用的访问周期和响应时间的上限必须等于调度周期,且所有应用必须具有相同的统计学服务质量需求,缺乏灵活性.

另一类调度策略基于公平调度(fair scheduling),如Cello^[10]的WFQ(weighted fair queueing),APEX^[11]的令牌桶(token bucket),MARS^[12]的DRR(deficit round robin fair queueing)和Eclipse^[13]的YFQ.这类策略允许多媒体应用具有不同的访问周期,但仅能根据多媒体应用对服务资源的需求(如数据带宽)实现调度,而没有考虑统计学服务质量需求.

本文介绍的基于服务质量需求的动态优先权调度策略QADPS(QoS-ware dynamic priority scheduling scheme)实现了对混合负载中不同类型多媒体应用的差别调度.该算法根据每一个多媒体应用的服务质量需求和它接受服务的历史信息,确定其需要接受服务的紧迫性;急需服务的多媒体应用被赋予较高的调度优先权,优先接受存储系统的服务.它的适应性超过了目前已知的调度策略,能够在保证多媒体应用服务质量的前提下支持更多的并发访问.

1 混合类型多媒体负载访问存储系统的过程模型

1.1 定义时间窗口约束条件

N 个多媒体应用并发访问存储系统,形成混合类型多媒体负载 Π .为满足连续播放的需要, Π 中的一个多媒体应用 τ_i ($1 \leq i \leq N$)按一定的周期 T_i 向存储系统发出数据访问请求,每次从存储设备上读取一定大小的数据块,形成一个访问请求序列. τ_i 有一个响应时间上限 d_i ,要求每个访问请求从到

达存储系统直至完成数据访问的时间间隔小于或等于 d_i ,即响应时间符合 d_i .另一方面,由于多媒体应用自身的特点(如缓冲机制),并不严格要求所有访问请求的响应时间都必须符合 d_i ,通常允许小于一定比例上限 P_i 的访问请求超时完成.

那些响应时间超过 d_i 的访问请求在访问请求序列中的分布是非常重要的,连续超时的情况往往是不能忍受的.比如一个多媒体应用能够忍受10%的访问请求的响应时间超过 d_i ,可能意味在每10个访问请求中最多1个请求超时;也可能意味着在连续100个请求超时后,连续900个请求符合 d_i .用户往往能够忍受前者的服务质量,而对后者却是不能容忍的. P_i 仅仅描述了一段较长的时间间隔内的平均信息,无法准确描述多媒体应用对超时访问请求的忍耐能力.

本文引入时间窗口约束条件(time windows constrained condition) $\lambda_i = (n_i | m_i)$ 精确描述多媒体应用对超时访问请求的忍耐能力.它要求在多媒体应用的访问请求序列中,任意 m_i 个连续的访问请求构成的窗口中,最少 n_i 个访问请求的响应时间符合 d_i ,并且限制连续超时的访问请求不超过 $m_i - n_i$.

1.2 访问过程模型

由于多媒体应用的VBR(variable bit rate)特性和磁盘定位延迟的随机性,存储系统为一个访问请求提供的服务时间 S_i 是一个随机变量.多媒体应用 τ_i 对存储系统的服务质量需求可以用一个4元组 $(T_i, d_i, \lambda_i, S_i)$ 定义.不同的多媒体应用可能因为4元组中参数的不同而具有不同的服务质量需求.利用服务质量需求的定义,混合类型多媒体负载可以描述为

$$\Pi = \{\tau_i = (T_i, d_i, \lambda_i, S_i)\}_{1 \leq i \leq N}.$$

存储系统为 Π 中的每个访问请求提供服务,这一过程可以抽象为图1所示的模型. Π 中的每一个

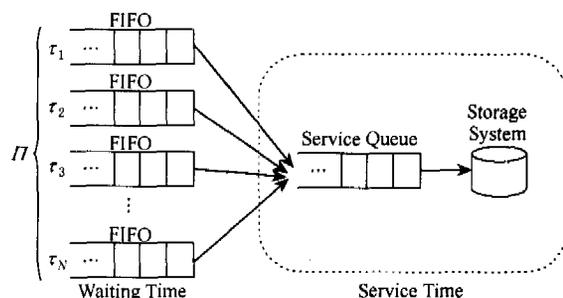


Fig. 1 The process model of mixed multimedia workload retrieving data from the storage systems.

图1 混合类型多媒体负载访问存储系统的过程模型

多媒体应用都对应一个独立的 FIFO 队列, FIFO 队列保证属于同一个多媒体应用的访问请求按照到达系统的顺序接受存储系统的服务,同时也保证只有 FIFO 队列头部的访问请求才可能被调度并接受服务.

如何从这些候选的访问请求中选择适当的访问请求加入服务队列由存储系统的调度策略决定. 存储系统调度的本质是为所有 FIFO 队列的头部访问请求分配优先权,具有较高优先权的访问请求优先接受存储系统的服务.

被调度策略选择的访问请求加入到一个服务队列中,这些访问请求访问磁盘数据的物理操作还将被底层的磁盘调度算法(如 SCAN 算法)再次调度. 在本文的研究中,访问请求的服务时间 S_i 包含了由于磁盘调度算法而产生的服务队列等待时间,在研究存储系统的调度策略时将不再考虑磁盘调度算法的影响. 根据过程模型描述,访问请求的响应时间由在 FIFO 队列中的等待时间和服务时间组成.

2 QADPS 调度策略

QADPS 调度策略根据多媒体应用的服务质量需求和多媒体应用接受服务的历史信息,确定多媒体应用需要接受服务的紧迫程度;并赋予急需接受服务的多媒体应用以较高的优先权.

2.1 多媒体应用接受服务的历史信息

由于多媒体应用 τ_i 的服务质量需求仅仅关注访问请求的响应时间是否超过响应时间上限 d_i ,因此 τ_i 的访问请求序列中一个访问请求的状态信息可以用一个比特位(bit)表示,其中 0 表示访问请求的响应时间超时,1 表示访问请求的响应时间符合 d_i . 进而 τ_i 的历史信息能够用一个比特序列 $\mu_i = (\delta_1^i \delta_2^i \delta_3^i \dots \delta_{p_i}^i)$ 描述. μ_i 中比特位的排列顺序表示访问请求被服务的顺序,其中 $\delta_{p_i}^i$ 描述距离当前最近一个完成服务的访问请求. μ_i 的长度 p_i 随着访问请求不断被服务而不断地增长.

在比特序列 μ_i 中,定义有限长度的连续比特位为一个字(word). 由于多媒体应用 τ_i 的时间窗口约束条件 $\lambda_i = (n_i | m_i)$ 要求在任意 m_i 个连续访问请求中,至少 n_i 个访问请求的响应时间符合 d_i ,因此调度策略只需要关注 μ_i 中所有长度为 m_i 的字. 由于 τ_i 的访问请求按照 FIFO 的顺序接受服务,因此 μ_i 中任意一个长度为 m_i 的字都曾经是在某一个时刻 μ_i 的最后 m_i 个比特位. 据此分析,由 μ_i 的最后

个比特位所构成的字 $\mu_i^{m_i} = \delta_{p_i - m_i + 1}^i \dots \delta_{p_i}^i$ 包含了与多媒体应用 τ_i 的服务质量相关的全部历史信息. $\mu_i^{m_i}$ 的取值随着服务的进行而变化,形成滑动服务窗口(sliding service windows).

$\mu_i^{m_i}$ 的最高位为 $\delta_{p_i - m_i + 1}^i$,最低位为 $\delta_{p_i}^i$,只要当 $p_i \geq 1$ 时 $\delta_{p_i}^i$ 就有实际意义,而当 $p_i < m_i$ 时 $\mu_i^{m_i}$ 将无法有效的定义. 为了便于描述所有可能的情况,需要定义 p_i 为任意整数时的 $\mu_i^{m_i}$,因此假设当 $j \leq 0$ 时 $\delta_j^i = 0$.

$\mu_i^{m_i}$ 被保存在一个长度为 m_i 位的移位寄存器中,如图 2 所示. 当多媒体应用 τ_i 的一个访问请求完成服务时,滑动服务窗口将向前滑动一位. 根据该访问请求是否超时,一个 0 或 1 将从寄存器的右端(最低位)移入,而寄存器的左端(最高位)将溢出, $\mu_i^{m_i}$ 的值被更新,多媒体应用接受服务的历史信息也被更新.

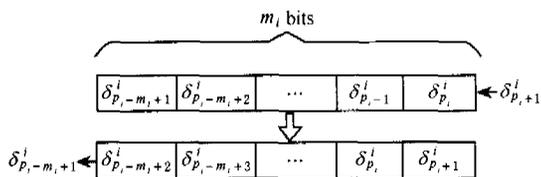


Fig. 2 Using the sliding service windows to update the history information of multimedia applications.

图 2 滑动服务窗口更新多媒体应用接受服务的历史信息

2.2 服务质量失败距离

由于 $\mu_i^{m_i}$ 的长度为 m_i ,它的取值有 2^{m_i} 种可能. 根据时间窗口约束条件 λ_i 的要求,在保证服务质量的情况下,在任何时刻 $\mu_i^{m_i}$ 中 1 的数量不应小于 n_i ,此时称为服务质量保证状态. 如果 $\mu_i^{m_i}$ 中 1 的数量少于 n_i ,则多媒体应用 τ_i 的服务质量将无法得到保证,此时称为服务质量失败状态. 当处于服务质量保证状态的多媒体应用 τ_i 的一个访问请求超时完成,一个 0 从 $\mu_i^{m_i}$ 的最低位移入, τ_i 将向服务质量失败状态靠近.

多媒体应用 τ_i 的服务质量失败距离(QoS failed distance) f_i 定义为: τ_i 从当前状态转换为服务质量失败状态,需要访问请求的响应时间连续超时的最少次数;也就是 $\mu_i^{m_i}$ 从当前值转换为恰好包含 $n_i - 1$ 个比特 1 的值时,需要在比特序列 μ_i 的尾部连续添加 0 的个数.

f_i 的方法计算如下:令 $L(n, \mu_i^{m_i})$ 表示从 $\mu_i^{m_i}$ 的最低位开始,第 n 个 1 所在的位置;如果 $\mu_i^{m_i}$ 中 1

的数量少于 n , 则 $L(n, \mu_i^{m_i}) = m_i + 1$. 例如, τ_i 的时间窗口约束条件为 $\lambda_i = (1|3)$, 当 $\mu_i^{m_i} = 101$ 时, $L(1, 101) = 1$ 且 $L(2, 101) = 3$; 而当 $n > 2$ 时, $L(n, 101) = 4$. 则 τ_i 的服务质量失败距离 f_i 为

$$f_i = m_i - L(n_i, \mu_i^{m_i}) + 1. \quad (1)$$

2.3 多媒体应用的优先权

QADPS 调度策略的目的就是要避免服务质量失败状态的产生: 一个多媒体应用距离服务质量失败状态越近, 它的优先权被设置得越高. 因此多媒体应用 τ_i 的下一个访问请求的优先级取值为 τ_i 当前的服务质量失败距离 f_i . f_i 较小的多媒体应用的下一个访问请求将被优先调度, 以增加该应用在符合响应时间上限的情况下完成服务的机会, 从而使它远离服务质量失败状态.

当不同多媒体应用具有相同的 f_i 时, QADPS 优先调度具有较小 m_i 多媒体应用的访问请求; 如果 m_i 仍然相同, 优先调度 deadline 较早的访问请求.

假设多媒体应用 τ_i 的时间窗口约束条件为 $\lambda_i = (2|3)$, 由于 $m_i = 3$, $\mu_i^{m_i}$ 描述了 τ_i 到当前时刻为止最后完成服务的 3 个访问请求的历史信息, 比如 $\mu_i^{m_i} = 110$ 表示最近一个访问请求的响应时间超过了 d_i , 而此前两个访问请求的响应时间符合 d_i . 图 3 为 $\mu_i^{m_i}$ 的状态转换图, 图中每个节点表示 $\mu_i^{m_i}$ 的一种可能状态(取值), 每条边表示一种可能的状态转换. 图中阴影节点表示 $\mu_i^{m_i}$ 的取值中比特 1 少于 2 个, τ_i 处于服务质量失败的状态. QADPS 策略根据 τ_i 的服务质量失败距离 f_i 设置优先权: 当 $\mu_i^{m_i}$ 为阴影状态的值时, τ_i 处于服务质量失败状态, 其 f_i 为 0, 下一个访问请求具有最高优先权; 当 $\mu_i^{m_i}$ 的值为 110 或 101 时, f_i 为 1, 下一个访问请求的优先级设置为 1; 当 $\mu_i^{m_i}$ 的值为 011 或 111 时, f_i 为 2, 下

一个访问请求的优先权最低.

根据下一个访问请求的响应时间是否超过 d_i , $\mu_i^{m_i}$ 能够从图中任意一个状态转换到两个可能状态中的一个; 多媒体应用 τ_i 的服务质量失败距离 f_i 将发生变化, 调度优先权也将发生变化.

QADPS 的核心思想是使服务质量需求较为宽松的多媒体应用放弃自身获得服务的一些机会, 并将这些服务机会给予那些急需服务的多媒体应用. 该策略的负面影响是可能会导致那些距离服务质量失败状态较远的多媒体应用的一些访问请求超时完成.

3 仿真实验和分析

我们在 DiskSim3.0 仿真软件上实现了基于 QADPS 调度策略的存储系统调度算法^[14]. 仿真软件构造的存储系统是一个由 8 个 HP_C2490A 磁盘组成的磁盘阵列, 并配置为 RAID0.

实验采用两种类型多媒体应用, 一类为 64Kbps 的窄带应用, 另一类为 1024Kbps 的宽带应用, 它们的服务质量需求参数见表 1. 表中 $E(S_i)$ 为访问请求的平均服务时间. 在混合类型多媒体负载中, 两类多媒体应用的数量始终相等, 并且成对逐渐增加.

Table 1 The QoS Requirements of Multimedia Applications
表 1 多媒体应用的服务质量需求

Parameters	Types	
	64Kbps	1024Kbps
T_i (s)	1	1
d_i (ms)	500	500
p_i (%)	10	5
λ_i	(18 20)	(19 20)
$E(S_i)$ (ms)	19.355219	49.075232

参照实验采用 EDF 调度策略和 DRR 公平调度策略. DRR 根据 $E(S_i)$ 的比例为多媒体应用精确保留服务资源.

记录在不同的负载下响应时间超过 500ms 的访问请求所占比例. 如图 4 所示, 当采用 EDF 调度策略时, 两类多媒体应用中超时访问请求的比例几乎相等. 当两类应用的并发数量均为 105 时, 这一比例恰好小于 5%. 因此 EDF 策略能够保证服务质量的并发多媒体应用数量的上限为 105. 如图 5 所示, 当采用 DRR 调度策略时, 保证服务质量的并发应用数量的上限为 104.

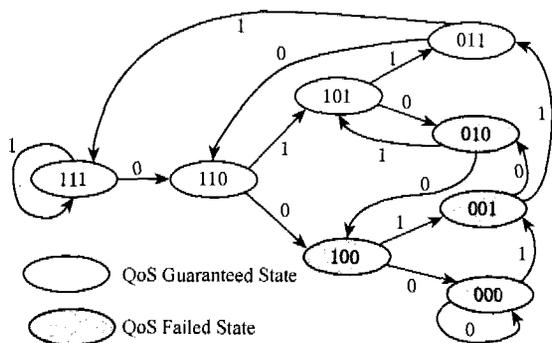


Fig. 3 The state transfers of $\mu_i^{m_i}$, $\lambda_i = (2|3)$.

图 3 当 $\lambda_i = (2|3)$ 时, $\mu_i^{m_i}$ 的状态转换的例子

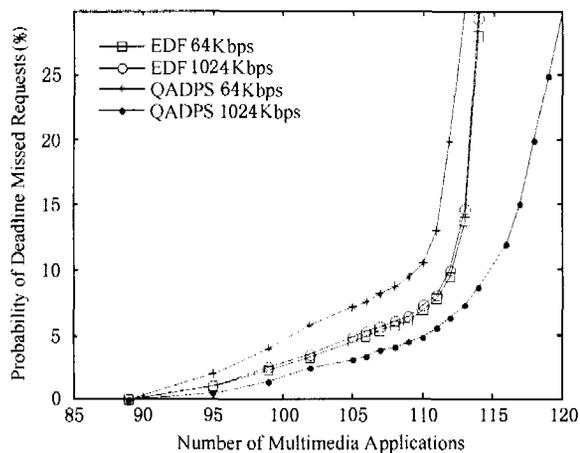


Fig. 4 The probability of requests whose response time is beyond 500ms, EDF vs QADPS.

图4 采用 EDF 和 QADPS 调度算法时,不同多媒体应用中响应时间超过 500ms 的访问请求所占比例

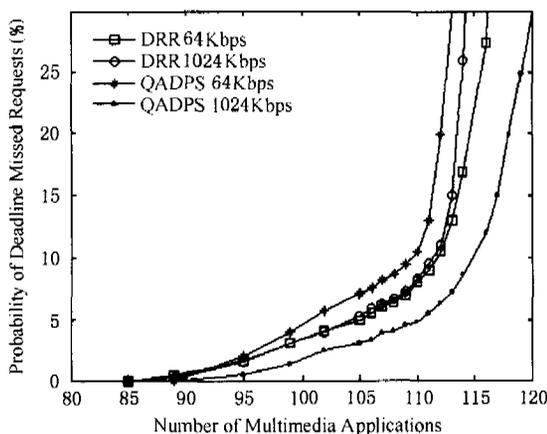


Fig. 5 The probability of requests whose response time is beyond 500ms, DRR vs QADPS.

图5 采用 DRR 和 QADPS 调度算法时,不同多媒体应用中响应时间超过 500ms 的访问请求所占比例

当采用 QADPS 调度策略时(如图 4、图 5 所示),两类多媒体应用中超时访问请求的比例相差较大.当并发应用的数量为 110 时,64Kbps 和 1024Kbps 应用的超时比例恰好小于 10% 和 5%.因此 QADPS 策略能够保证服务质量的并发多媒体应用数量的上限为 110;这比 EDF 和 DRR 提高了约 5%.QADPS 策略在保证服务质量的前提下能够支持更多的并发多媒体应用.

EDF 和 DRR 调度策略实现公平调度,两类多媒体应用的访问请求被调度的几率相等,超时访问请求的比例几乎相等,其中的少许差异主要是由于 1024Kbps 应用的访问请求需要更多的服务时间.而 QADPS 调度策略是不公平的,64Kbps 应用中超时

访问请求所占的比例大约是 1024Kbps 应用的 2 倍.由于 1024Kbps 应用的服务窗口约束条件比 64Kbps 应用严格,当它们的访问请求同时等待调度时,1024Kbps 应用的访问请求将得到更多的调度机会.QADPS 调度策略根据多媒体应用的服务质量需求实现有差别的调度.

一个值得注意的现象是,采用 QADPS 调度策略时,在包括两种类型多媒体应用的全部访问请求中,超时访问请求的比例大于 EDF 策略,如图 6 所示.QADPS 策略并不能减少超时访问请求的总体数量,但将这些访问请求根据不同多媒体应用的服务质量需求在应用之间进行了重新分配.

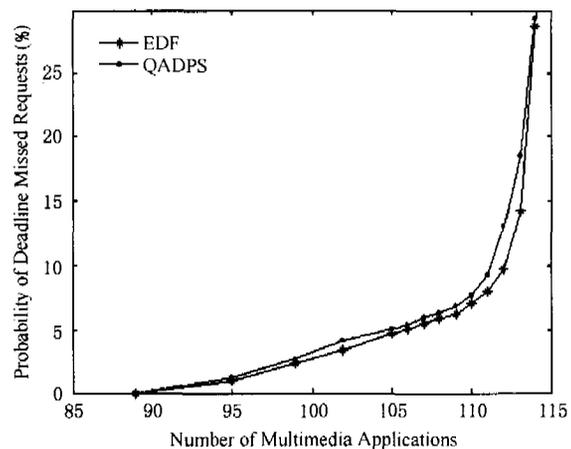


Fig. 6 The probability of mixed requests whose response time is beyond 500ms, EDF vs QADPS.

图6 采用 EDF 和 QADPS 调度算法时,全部访问请求中响应时间超过 500ms 的访问请求所占比例

4 结 论

多媒体应用访问存储系统时,要求存储系统提供保证服务质量的服务.QADPS 调度策略根据多媒体应用接受服务的历史信息计算出服务质量失败距离,并根据服务质量失败距离的大小为多媒体应用动态分配调度优先权,服务质量失败距离越小的多媒体应用被赋予越高的优先权.QADPS 调度策略根据多媒体应用访问存储系统的不同服务质量需求,实现了有差别的调度:多媒体应用的服务质量需求越严格,它的访问请求被调度的机会越高.

参 考 文 献

- [1] P Halvorsen, C Griwodz, K lund, *et al.* Storage systems support for multimedia application [R]. Department of Informatics, University of Oslo, Tech Rep: 307, 2003

- [2] B L Worthington, G R Ganger, Y N Patt. Scheduling algorithms for modern disk drivers [C]. In: Proc of ACM SIGMETRICS'94. New York: ACM Press, 1994. 241-251
- [3] P Bosch, S J Mullender. Real-time disk scheduling in a mixed-media file system [C]. In: Proc of the 6th IEEE Real Time Technology and Applications Symposium (RTAS2000). Los Alamitos, CA: IEEE Computer Society Press, 2000. 23-33
- [4] P Shenoy, P Goyal, S S Rao, *et al.* Symphony: An integrated multimedia file system [R]. Department of Computer Sciences University of Texas, Tech Rep: TR-97-09, 1997
- [5] P S Yu, M S Chen, D D Kandlur. Design and analysis of a grouped sweeping scheme for multimedia storage management [C]. In: Proc of Int'l Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'92). Heidelberg: Springer-Verlag, 1992. 44-45
- [6] S J Daigle, J K Strosnider. Disk scheduling for multimedia data streams [C]. In: Proc of Conf on High-Speed Networking and Multimedia Computing. New York: ACM Press, 1994. 212-223
- [7] R Deepak, K Hosekote, J Srivastava. I/O scheduling for digital continuous media [J]. ACM/Springer Multimedia System, 1997, 5(4): 213-237
- [8] H Pan, L H Ngoh, A A Lazar. A buffer-inventory-based dynamic scheduling algorithm for multimedia-on-demand servers [J]. ACM/Springer Multimedia Systems, 1998, 6(2): 125-136
- [9] E Chang, H G Molina. Bubbleup: Low latency fast-scan for media servers [C]. In: Proc of ACM Int'l Multimedia Conf. New York: ACM Press, 1997. 87-98
- [10] P J Shenoy, H M Vin. Cello: A disk scheduling framework for next generation operating systems [J]. Real-Time Systems Journal: Special Issue on Flexible Scheduling of Real-Time Systems, 2002, 22(1): 9-47
- [11] K Lund, V Goebel. Adaptive disk scheduling in a multimedia DBMS [C]. In: Proc of the 11th ACM Int'l Conf on Multimedia. New York: ACM Press, 2003. 65-74
- [12] M M Buddhikot. Project mars: Scalable, high performance, Web based multimedia-on-demand (MOD) services and servers; [Ph D dissertation] [D]. Washington: Server Institute of Technology, Department of Computer Science, Washington University, 1998
- [13] J Bruno, J Brustoloni, E Gabber, *et al.* Disk scheduling with quality of service guaranteed [C]. In: Proc of IEEE Int'l Conf on Multimedia Computing and Systems. Los Alamitos, CA: IEEE Computer Society Press, 1999. 400-405
- [14] J S Bucy, G R Ganger. The DiskSim simulation environment version 3.0 reference manual [R]. Carnegie Mellon University, Tech Rep: CMU-CS-03-102, 2003



Li Zhong, born in 1972. Received his Ph. D. degree from Nankai University in 2005. His main research areas include performance analysis of computer systems, and Parallel & distributed system.

李中, 1972年生, 博士, 主要研究方向为存储系统性能分析、并行与分布式系统。



Wang Gang, born in 1974. Associate professor of Nankai University, member of CCF. His main research interests include data layout, mass storage, and parallel & distributed system.

王刚, 1974年生, 副教授, 中国计算机学会会员, 主要研究方向为数据布局、海量存储、并行与分布式系统。



Liu Jing, born in 1942. Professor and Ph. D. supervisor of Nankai University. His main research interests include parallel & distributed system, mass storage, and parallel VLSL algorithm.

刘璟, 1942年生, 教授, 博士生导师, 主要研究方向为并行与分布式系统、海量存储、并行 VLSL 算法。

Research Background

The multimedia applications need storage systems to guarantee their specific QoS requirements. This causes the problem: how to schedule the requests of these concurrent applications properly. Traditional scheduling schemes have not taken different QoS requirements into account.

A systemic research on this problem is conducted in this paper. The QoS parameters of the multimedia applications are clearly defined, and the QADPS scheduling scheme is established. The new scheme supports the concurrent multimedia applications have different QoS requirements, and implements scheduling based on these requirements. Using QADPS, the requests of the applications with stricter QoS requirements can obtain more chances of service applied by the storage systems. Compared with conventional scheme, QADPS substantially increases the number of the concurrent multimedia applications with QoS guarantee.

Our work is supported by the National Natural Science Foundation of China (NSFC-60273031).