基于累积效应的光交换机分组调度算法

熊庆旭 冯金鑫 王凤周

(北京航空航天大学 电子信息工程学院, 北京 100083) (北京遥感设备研究所,北京 100854)

摘 要: 光交换机中的光路重置需要较长的时间,分组调度必须是基于帧的. 为提高传输效率帧长通常较大,从而导致了光交换机中特有的分组累积效应. 针对该效应,提出一种新的分组调度算法 p-LQF(Longest Queue First)+Hold. 通过对业务需求矩阵分解得到置换矩阵, 选择匹配占用最大的置换矩阵建立端口匹配. 在 4×4 端口的 VOQ(Virtual Output Queue)结构的光交换机中,分别对均匀、非均匀以及对角线的 Bernoulli 和 ON/OFF 业务进行了仿真. 仿真结果显示该算法对于符合强大数定理的可接入业务能获得 100% 的通过率,其平均分组时延接近甚至低于主流的 LQF+Hold 算法,但复杂度更低. 该算法能够提供带宽保证以及短期公平性,同时能适应分组的动态到达.

关 键 词: 光交换机; 网络排队; 调度; 分组交换

中图分类号: TP 393

文献标识码: A **文 章 编 号**: 1001-5965(2006)03-0311-05

Novel scheduling algorithm based on packet aggregation effect for optical switches

Xiong Qingxu Feng Jinxin

(School of Electronics and Information Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100083, China)

Wang Fengzhou

(Institute of Remote-Sensing Equipment, Beijing 100854, China)

Abstract: The packet scheduling in optical switches has to be frame based because the configuration consumes much time. The frame size is usually selected to be large enough in order to improve transmission efficiency, leading packet aggregation effect. With consideration of the packet aggregation effect, a novel packet scheduling algorithm called permutation-based longest queue first+hold (p-LQF+Hold) was proposed. A set of permutations were calculated by using of the traffic requirement matrix. The permutation with the largest matched occupancy was used to find the matching on each frame boundary. Simulation was made on a 4×4 virtual output queue(VOQ) optical switch under uniform, nonuniform and diagonal Bernoulli and ON/OFF traffic patterns, respectively. The simulation results show p-LQF+Hold can achieve 100% throughput under any admissible traffic that satisfies the strong law of large numbers, and has close to or even lower average packet delay than the popular LQF+Hold algorithm. p-LQF+Hold can guarantees bandwidth requirement and short-term fairness and efficiently accommodate the dynamical packet arrivals.

Key words: optical switches; queuing networks; scheduling; packet switching

光网络因其巨大的传输容量是未来网络发展的方向之一,而光交换是光网络的核心技术. 由于光存储技术还不实用,无法彻底实现全光交换. 因此实用的光交换机在相当一段时间里将采用电存储光交换的光电混合结构. 考虑到光交换机主要用于高速干线网中,因此通常用虚拟输出排队

收稿日期: 2005-06-07

作者简介: 熊庆旭(1965-),男,江西南昌人,副教授,qxxiong@ieee.org.

VOQ (Virtual Output Queue)排队结构[1]. VOQ 结构是为克服传统的先进先出(first in first out)输入排队中头分组(head of line)阻塞而提出的一种输入排队结构. 每个输入端口的缓存物理或逻辑地分为 N 个缓存器,分别为其去往每一个输出端口的分组维持一个队列. 光交换机中由于光路的重配置时间较长,导致了与电交换不同的特点[2]. 为提高传输效率及降低内部工作速率,输入输出端口之间的匹配关系必须固定维持一个较长的时间. 但这将使得未获得匹配的 VOQ 在较长时间里只有分组到达而又无法传输,分组大量累积显著改变了分组的到达过程,从而产生了分组累积效应.

现有光交换机中的调度算法主要采用电交换机中的方法,通过仿真选取性能较好的. 算法 LQF (Longest Queue First)+Hold[3] 及 i-LQF (iterative LQF)+Hold[4] 是目前 2 种典型的算法,前者采用最大权重匹配,后者采用流行的 3 步迭代极大权重匹配法[5]. 但这些算法没有充分考虑光交换机的特点. 本文基于光交换机中的分组累积效应,提出了一种新的调度算法 p-LQF(permutation-based LQF)+Hold. 该算法首先分解业务需求矩阵(requirement matrix),获得有限个置换矩阵(permutation matrix),选择其中匹配占用(matched occupancy)最大的建立端口匹配. p-LQF+Hold 算法对符合强大数定理的可接入业务均能获得 100% 的通过率,平均分组时延接近甚至低于目前公认为最好的 LQF+Hold 算法,远低于 i-LQF+Hold 算法.

1 光交换机中的分组调度

MEMS(Micro-Electro-Mechanical Systems)被认为是最有发展前途的光电混合交换结构[4]. 该交换结构是在 VOQ 的交换结构(switch fabric)中采用光交换,在 VOQ 缓存后进行电光转换,在输出缓存前进行光电转换,见图 1,图 1 中 VOQ[1 N]表示到达输入端口 1 去往输出端口 N 的分组队列. 由于光路重配置的原因,光交换机中的分组调度是基于帧的(frame-based),一帧分为重配置和分组传输 2 个阶段. 在重配置阶段没有任何分组离开交换机,因此为保证交换机的稳定,传输阶段的加速比(speedup)需大于等于 $F/(F-C)$,其中 F 表示帧长,C 为重配置时间,如图 2 所示.

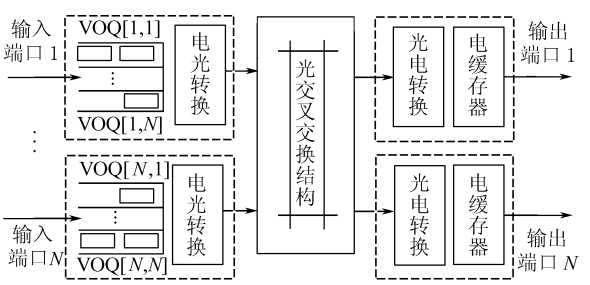


图 1 MEMS 光电混合光交换机结构

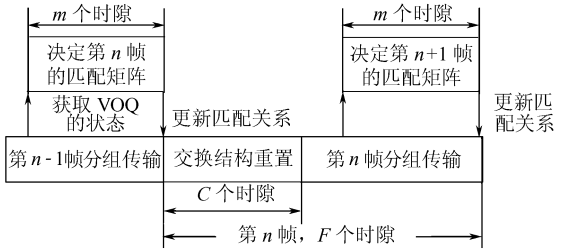


图 2 光电混合结构中固定帧长的分组调度过程

2 p-LQF+Hold 算法

p-LQF+Hold 算法包括离线业务需求矩阵分解和在线置换矩阵调度 2 个部分.

2.1 业务需求矩阵分解

对于 $N\times N$ 交换机,令 r_{ij} 表示从输入端口 i 到输出端口 j 分组的平均速率,则令 $\boldsymbol{R}=(r_{ij})_{N\times N}$ 表示速率矩阵. 当分组到达同时满足式(1)和式(2)时,称为可接入业务. 若式(1)和式(2)中的等号成立,则称 $\boldsymbol{R}$ 为双随机(doubly stochastic)矩阵.

$$\sum_{i=1}^{N} r_{ij} \leqslant 1 \qquad j=1,2,\cdots,N \tag{1}$$

$$\sum_{j=1}^{N} r_{ij} \leqslant 1 \qquad i=1,2,\cdots,N \tag{2}$$

由 Birkhoff 定理可知[6],任意双随机矩阵均可分解为 $k(\leqslant N^2-2N+2)$ 个加权置换矩阵之和. 本文在 $\boldsymbol{R}$ 分解中,首先将 $\boldsymbol{R}$ 各个元素通分为整数,构造相应的业务需求矩阵 $\boldsymbol{M}=[m(i,j)]_{N\times N}$,使得各行各列之和为一常数 p,即

$$\sum_{i=1}^{N} m(i,j)=p \qquad j=1,2,\cdots,N \tag{3}$$

$$\sum_{j=1}^{N} m(i,j)=p \qquad i=1,2,\cdots,N \tag{4}$$

式中,p 为调度长度;$\boldsymbol{M}$ 等于所有置换矩阵之和.

例如,4×4 端口的 $\boldsymbol{M}$ 如下所示:

$$\boldsymbol{M}=\begin{bmatrix}3&2&2&1\\2&0&2&4\\0&4&2&2\\3&2&2&1\end{bmatrix} \tag{5}$$

式中,$p=8$. 则可把 $\boldsymbol{M}$ 分解为以下 8 个置换矩阵:

$$\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \qquad \begin{bmatrix}0&0&0&1\\1&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&1&0&0\\0&0&1&0\end{bmatrix}\quad\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&0&1&0\\0&1&0&0\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&0&0&1\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\1&0&0&0\end{bmatrix}$$

$$\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&1&0&0\\1&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&1&0&0\\1&0&0&0\end{bmatrix}$$

其中,第 7 与第 8 个置换矩阵相同,则

$$\boldsymbol{M}=\boldsymbol{P}^1+\boldsymbol{P}^2+\boldsymbol{P}^3+\boldsymbol{P}^4+\boldsymbol{P}^5+\boldsymbol{P}^6+2\boldsymbol{P}^7 \quad (6)$$

式中,$\boldsymbol{P}^k$ 表示分解出的第 k 个匹配矩阵.

2.2　矩阵分解算法

在矩阵分解过程中,可以采用如下分解步骤,将 $\boldsymbol{R}$ 分解为若干个匹配矩阵之和.

1) 根据 $\boldsymbol{M}$ 构造观察矩阵 $\underline{\boldsymbol{M}}$. 若 $\boldsymbol{M}$ 中 $m(i,j)>0$,则令 $\underline{\boldsymbol{M}}$ 中的 $\underline{m}(i,j)$ 为 1,否则为 0. 例如式(5) $\boldsymbol{M}$ 的观察矩阵为

$$\underline{\boldsymbol{M}}=\begin{bmatrix}1&1&1&1\\1&0&1&1\\0&1&1&1\\1&1&1&1\end{bmatrix} \quad (7)$$

2) 在 $\underline{\boldsymbol{M}}$ 中寻找一个不为 0 的元素,该元素所在的行和列中不为 0 的元素个数最少,然后将 $\underline{\boldsymbol{M}}$ 中该元素所在的行与列的所有元素置 0. 重复上面过程,直到寻找出所有 N 个元素为止. 该 N 个元素构成一个 $N\times N$ 置换矩阵. 例如在式(7)中找到的相应元素分别为 $\underline{m}(1,2)$,$\underline{m}(2,1)$,$\underline{m}(3,3)$,$\underline{m}(4,4)$,得到置换矩阵:

$$\boldsymbol{P}^1=\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \quad (8)$$

用 $\boldsymbol{M}$ 减去所得置换矩阵,得到的矩阵再次构造观察矩阵,循环执行 1) 和 2) 的操作,直到最后得到的 $\boldsymbol{M}$ 所有元素都为 0,跳出循环.

3) 所得置换矩阵中若有相同者,保留其中一个,其余的删除.

2.3　在线调度算法

在分解出置换矩阵后,令 Q_{ij} 表示一帧开始时输入端口 i 去往输出端口 j 的 VOQ 中等待传输的分组队长,其中 $1\leqslant i,j\leqslant N$. 则 $\boldsymbol{P}^k$ 的匹配占用为

$$\boldsymbol{O}^k=\sum_{i,j}^{N}Q_{i,j}p_{i,j}^k \quad (9)$$

选择 $\boldsymbol{P}^k[\boldsymbol{P}^k\mid\boldsymbol{O}^k\geqslant\boldsymbol{O}^n(n\in1,2,\cdots,N,\text{且 }n\neq k)]$ 作为该帧的匹配阵. 例如上面的 4×4 端口交换机,假设 VOQ 队长矩阵为

$$\boldsymbol{Q}=\begin{bmatrix}150&70&40&30\\20&0&30&100\\0&90&40&30\\70&40&30&10\end{bmatrix} \quad (10)$$

计算各置换矩阵的匹配占用,可以得 $\boldsymbol{O}^k=$[140 170 370 330 250 200 300],式中,$\boldsymbol{O}^3$ 最大,则选择 $\boldsymbol{P}^3$.

3　数值仿真

本文分别讨论 p-LQF+Hold,i-LQF+Hold 以及 LQF+Hold 3 种算法在相同业务下的通过率及平均分组时延特性.

3.1　仿真参数

所有输入端口的分组到达为独立相同分布过程. 本文讨论随机及突发业务中典型的 Bernoulli 及 ON/OFF 业务.

图 3 给出了 ON/OFF 业务的信源模型. 信源处于 ON 状态时,以峰值码率产生分组;处于 OFF 状态时,不产生分组. 仿真中假设平均突发长度为 16 个分组,每个突发期内的分组去往同一输出端口. 状态转移概率 $a=1/b_s$,$b=\rho/[b_s\times(1-\rho)]$,式中,$b_s$ 为平均突发长度,ρ 为业务强度.

每一种业务均考虑 3 种分组去向分布:均匀、非均匀及对角线. 均匀是指到达分组去往的输出端口分布为均匀分布. 非均匀采用式(11)表示的去向分布. 例如输入端口 1 中到达的分组去往输出端口 1 ~4 的概率分别为 3/8,2/8,2/8 和 1/8.

$$\boldsymbol{M}=\begin{bmatrix}3&2&2&1\\2&0&2&4\\0&4&2&2\\3&2&2&1\end{bmatrix} \quad (11)$$

对角线是公认最扭曲的去向分布,在电交换机中往往用来检验算法性能的最差结果. 对角线业务中分组的去向分布为

$$r_{i,j}=\begin{cases}\delta\times\rho_i & j=i\\(1-\delta)\times\rho_i & j=(i+1)\bmod N\\0 & \text{其他}\end{cases} \quad (12)$$

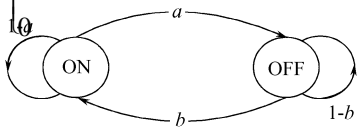


图 3　ON/OFF 信源模型

式中,ρ_i 为输入端口 i 的输入业务强度;mod 表示模运算. 设定所有输入端口的业务强度均为 ρ. 且令 $\delta=2/3$,即公认的最扭曲的对角线业务.

交换机的大小为4×4 端口,仿真时间为50 万时隙, F 和 C 分别为200 时隙和50 时隙.

3.2 平均分组时延

3.2.1 均匀业务

图4 为在均匀 Bernoulli 业务下,3 种算法平均分组时延几乎相同. 但 p-LQF+Hold 时延最小. 从图5 可以看出,在均匀 ON/OFF 业务下,各种算法的平均时延相差较小, p-LQF+Hold 的平均分组时延介于 i-LQF+Hold 和 LQF+Hold 之间.

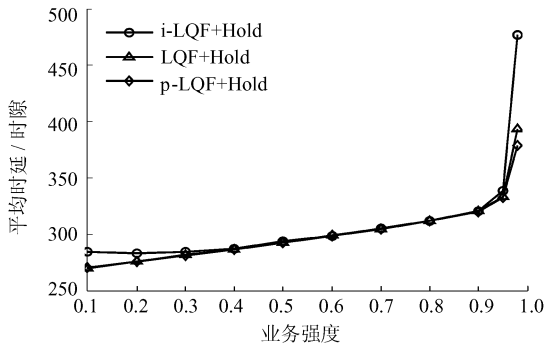


图4 均匀 Bernoulli 业务下 3 种算法的平均分组时延

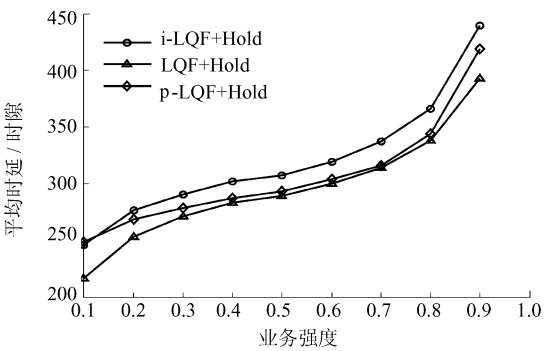


图5 均匀 ON-OFF 业务下 3 种算法的平均分组时延

3.2.2 非均匀业务

由图 6 可知,在非均匀 Bernoulli 业务下, LQF+Hold 和 p-LQF+Hold 算法的平均时延非常接近, 远低于 i-LQF+Hold. 图7 显示对于 $\boldsymbol{M}$ 表示的非均匀 ON/OFF 业务,3 种调度算法的平均分组时延. p-LQF+Hold 平均分组时延介于 i-LQF+Hold 和 LQF+Hold 之间.

3.2.3 对角线分布

图8 显示,在对角线 Bernoulli 分布业务下,p-LQF+Hold 的平均分组时延与 LQF+Hold 几乎相同并且低于 i-LQF+Hold,尤其是当业务强度较大时. 对于对角线 ON/OFF 分布业务,图9 显示, p-LQF+Hold 与 LQF+Hold 平均时延相差无几,且业务强度较大时低于 i-LQF+Hold.

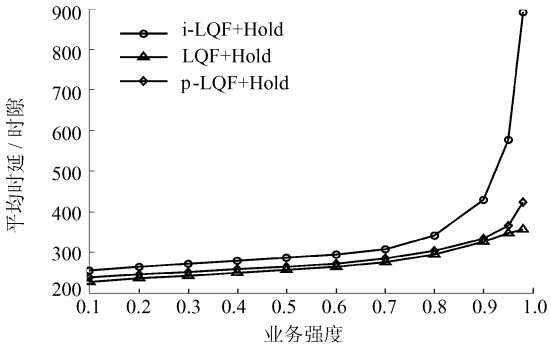


图6 非均匀 Bernoulli 业务下 3 种算法的平均分组时延

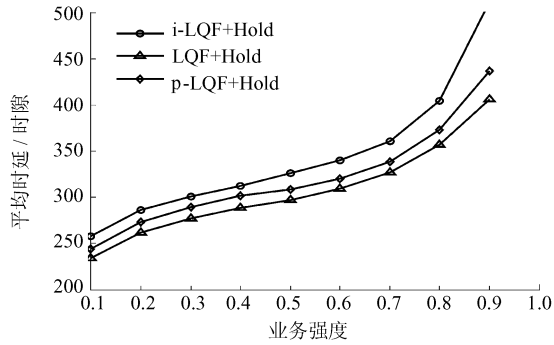


图7 非均匀 ON-OFF 业务下 3 种算法的平均时延比较

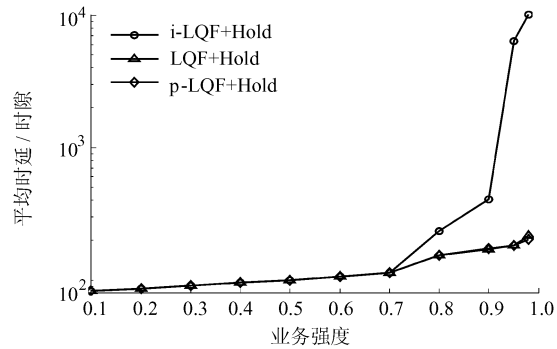


图8 对角线 Bernoulli 业务下 3 种算法的平均分组时延

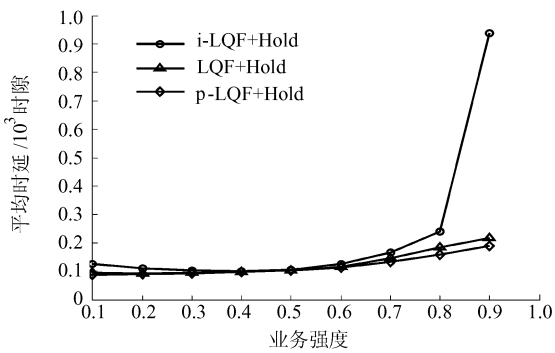


图9 对角线 ON/OFF 业务下 3 种算法的平均分组时延

3.3 通过率

对于均匀业务及 $\boldsymbol{M}$ 表示的非均匀业务,仿真结果表明 3 种算法通过率都能达到 100%. 对于

对角线业务, p-LQF+Hold 和 LQF+Hold 能达到 100% 通过率. 但在对角线 Bernoulli 及 ON/OFF 业务下,i-LQF+Hold 均不能保证达到 100% 通过率.

表 1 和表 2 分别给出了在输入业务强度较大时 3 种算法通过率的比较. 需要指出的是,理论上[3]已经证明,LQF+Hold 算法在加速比不小于 $F/(F-C)$时,对任意可接入业务均能获得 100% 的通过率.

表 1　业务强度为 0.95 的 Bernoulli 业务通过率　%

业　务	算　法		
	p-LQF+Hold	LQF+Hold	i-LQF+Hold
均匀 Bernoulli 业务	99.938 3	99.939 5	99.939 5
非均匀 Bernoulli 业务	99.931 5	99.933 2	99.900 5
对角线 Bernoulli 业务	99.968 1	99.968 1	97.907 2

表 2　业务强度为 0.9 的 ON/OFF 业务通过率　%

业　务	算　法		
	p-LQF+Hold	LQF+Hold	i-LQF+Hold
均匀 ON/OFF 业务	99.897 6	99.921 9	99.910 2
非均匀 ON/OFF 业务	99.940 0	99.933 6	99.911 0
对角线 ON/OFF 业务	99.970 6	99.970 7	99.652 2

4　分析讨论

仿真结果表明,p-LQF+Hold 在大部分业务类型中的分组时延及通过率性能与 LQF+Hold 十分接近,优于 i-LQF+Hold. 在业务强度较大时,在某些业务类型下平均分组时延甚至低于 LQF+Hold 算法. 由于是通过对业务需求矩阵的分解获得用于匹配的置换矩阵,所以 p-LQF+Hold 能够提供传输带宽的保证,且获得短期公平性,这是另外 2 种算法所不具备的. 但该算法不能提供分组时延的确保,原因是置换矩阵的在线调度决定于分组的到达. 然而,却使得 p-LQF+Hold 对分组到达具有相应的适应能力.

LQF+Hold 和 i-LQF+Hold 算法的复杂度分别为 $O(N^3)$和 $O(N^2)$,前者复杂度太高,不利于高速网络中物理实现,后者的通过率及分组时延特性较差. p-LQF+Hold 算法在线调度时只需查找最大值,无迭代和排序,其复杂度理论上最大为 $O(N^2)$. 当然 p-LQF+Hold 算法需要进行离线矩阵分解,但在干线网中的光交换机中,分组速率需求的变化频率远小于在电交换机中[7]. 另一方面,置换矩阵的在线动态调度,使得当速率需求的变化在一定范围内,原置换矩阵依然有效,无需重新分解速率需求矩阵. 因此,从实际情况来看,p-LQF+Hold 算法的复杂度将远小于 LQF+Hold 算法.

5　结 束 语

本文根据光交换机的分组累积效应,提出一种适应于各种可接入业务的分组调度算法 p-LQF+Hold. 仿真结果显示,p-LQF+Hold 的平均时延和通过率非常接近 LQF+Hold 算法. 但 p-LQF+Hold 在具体实现时无需目前主流算法中的迭代过程,且可以采用分布式并行方式实现,其复杂度远小于基于迭代的 i-LQF+Hold 算法,更低于最大权重的 LQF+Hold 算法,因此易于高速网络中的实际应用.

p-LQF+Hold 算法调度时只需从已有的置换矩阵中选择出匹配占用最大的,无需迭代和排序. 因此从端口迭代的角度来讲,p-LQF+Hold 算法的迭代复杂度为 $O(1)$,i-LQF+Hold 算法的迭代复杂度[4]为 $O(N^2)$. 可见,前者远低于后者,而复杂度是决定算法在高速网络中能否实用的关键.

参考文献 (**References**)

[1] Karol M, Hluchy M, Morgan S. Input versus output queueing on a space-division packet switch [J]. IEEE Trans on Comm, 1987, 35 (12): 1347 ~ 1356

[2] 熊庆旭. 输入排队结构交换机分组调度研究[J]. 通信学报, 2005,26(6):118 ~ 129
Xiong Qingxu. Research on packet scheduling in input-queued switches [J]. Journal of China Institute of Communication, 2005, 26(6):118 ~ 129(in Chinese)

[3] Li Xin, Hamdi M. Design and analysis of scheduling algorithms for switches with reconfiguration overhead [A]. Proc of IEEE HPSR'03[C]. Hong Kong: IEEE Press, 2003. 61 ~ 66

[4] Li Yihan, Panwar S, Chao H J. Frame-based matching algorithms for optical switches[A]. Proc of IEEE HPSR'03[C]. Hong Kong: IEEE Press, 2003. 97 ~ 102

[5] Xiong Qingxu. Providing QoS guarantees in input queued switches: advances and issues [A]. International Conference on Software Telecommunications and Computer Networks-SoftCOM 2003[C]. Split, Croatia: Univ Split, 2003. 311 ~ 315

[6] Chang C S, Chen W J, Huang H Y. Birkhoff-von neumann input-buffered crossbar switches for guaranteed-rate services [J]. IEEE Trans Comm, 2001, 49(7):1145 ~ 1147

[7] Careglio D, Rafel A, Sole Paretat J. Quality of service strategy in an optical packet network with multi-class frame-based Scheduling [A]. Proc of IEEE HPSR'03[C]. Hong Kong: IEEE Press, 2003. 129 ~ 134