

切换时延不为零的光交换调度算法

李季碧,毛期俭,张治中

(重庆邮电学院 通信工程学院,重庆 400065)

摘要:文章针对切换时延不为零的光交换调度提出了一种新算法——2 近似启发算法。算法由两部分组成:选择匹配和决策权重。其中,选择匹配是确定光交叉阵列的切换次数,由贪心算法完成;决策权重是决定各个配置的持续时长,它是通过选择一个值以使剩下的业务矩阵的开销估计最优。2 近似启发算法的近似因子为 2,时间复杂度为 $O(N^2 \log N)$ 。仿真表明这种调度算法更接近最优调度,比 DOUBLE^[1] 和 ADJUST^[2] 算法更能自适应传送来的不同业务模式。

关键词:光交换;调度算法;切换时延

中图分类号: TN915 **文献标识码:** A **文章编号:** 1005- 8788(2006)02- 0005- 03

Scheduling algorithm of optical switches with switching delay

LI Ji-bi, MAO Qi-jian, ZHANG Zhi-zhong

(Communication Engineering Institute, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: A new algorithm—2 approximation heuristic algorithm is presented for the scheduling of optical switches with configuration delay, which works by combining two main operations: matching selection and weight decision. Matching selection is achieved by greedy algorithm, which determines the times of the switch, while weight decision decides the holding time of the configuration selected, which is achieved by choosing a value such that the remaining traffic matrix has the best possible cost estimate. The algorithm has an approximation factor of 2 with a time complexity of $O(N^2 \log N)$. The simulation results demonstrate that this algorithm is close to optimal scheduling and more self-adjustable to various traffic arrival patterns than DOUBLE^[1] and ADJUST^[2] algorithms.

Key words: optical switch; scheduling algorithm; configuration delay

在设计光交换调度算法时,需要综合考虑业务的实际传输时间和光交换矩阵配置时的切换开销。传统的逐时隙调度算法^[3]只是简单假设切换时延为 0 或无穷大,不具有普遍性。由 Towles B 等人提出的 DOUBLE 算法^[1]在考虑了切换时延为一常数时,试图使用折衷的切换次数来均衡业务通过交换机的总开销,但它忽略了切换时延带来的影响。在 DOUBLE 的基础上,Li Xin 等人提出了一种改进算法:ADJUST 算法^[2]。但实际上,当 $|\sqrt{T/\delta N}| > 1$ 时,它比 DOUBLE 算法好,而当 $|\sqrt{T/\delta N}| \leq 1$ 时,它会恶化数据的实际传输时间,性能反而不如 DOUBLE。本文针对 ADJUST 和 DOUBLE 的缺点,提出一个新的调度方法——2 近似启发算法

1 交换模型

图 1 是基于光交换矩阵的光交换系统的结构图。从输入线卡来的数据包首先在电域里进行处理和缓存,然后经光纤传送到光交叉阵列处,并在光域里进行处理。一旦交换后,再经不同的光纤将它们传送到输出线卡,这时数据包再一次在电域里进行处理和缓存。相比于传统的电交换,这种结构的交

换机经济性更好,并具有可升级性好、交换比特率高、功耗低等优点。

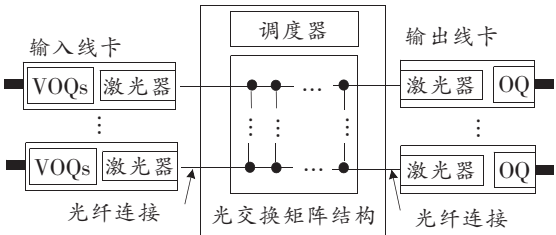


图 1 基于光交叉阵列的多架光交换系统

我们假设交换机的交换矩阵端口数为 $N \times N$,无阻塞,输入排队策略为虚拟输出排队(VOQ),即每个输入端口处为每个输出端口各设置了一个队列。从输入端口来的数据包,大小不一,每个包上都标记了目的输出端口,进入输入端口时自动进入目的端口队列。因假设交换是无阻塞的,任何输入与输出端口一对一匹配随时都能建立。我们根据文献[1~2,4]也采用分组调度方法,采用流水线作业,分 3 个阶段进行:汇聚、调度、传输。每个阶段时长为系统预先定义的常数 T ,也就是说,一个数据包从进入输入端口起,经过 $3T$ 时长后就能通过交换机,送到输出链路上。本文仅考虑允许范围内能调度的

收稿日期:2005-08-22
基金项目:重庆市教委基金资助项目(040502;040504;KJ050504);重庆市科委基金资助项目(2005BB2066)
作者简介:李季碧(1975-),女,四川开江人,硕士,研究方向为光网络通信技术。

数据,即在输入和输出端口没有过量的业务。因此,在积累时间 T 内待传业务矩阵 $\mathbf{D}$ 需满足以下条件:

$\sum_{i=1}^N d_{i,j} \leq T, \sum_{j=1}^N d_{i,j} \leq T$, 即任何一行或一列元素之和不大过 $T, d_{i,j}$ 是指输入端口 i 传向输出端口 j 整数倍时隙的汇聚周期 T 内的分组包。交换结构的调度矩阵 $\mathbf{P}$ 是 0-1 矩阵,每一行与每一列最多只有一个元素。 $P_{i,j}$ 表示输入端口 i 与输出端口 j 相连。交换机定期汇聚前来的数据包,然后将它们分割成定长的分组包,再将这些累积的分组包交换到相应的目的端口。假设调度矩阵 $\mathbf{P}$ 是由 s 个配置矩阵 $\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_s$ 构成,相应的持续时间为 $\phi_1, \phi_2, \dots, \phi_s$,待交换业务矩阵 $\mathbf{D}$ 能够被调度矩阵 $\mathbf{P}$ 覆盖应满足条件: $\forall i, j, \sum_{k=1}^s \phi_k \cdot p_{i,j}^k \geq d_{i,j}, p_{i,j}^k$ 是矩阵 $\mathbf{p}_k$ 的 (i, j) 元素。对于一个既定的业务矩阵 $\mathbf{D}$ 和切换延迟 δ , 业务传送开销的最优化可用下式表示:

$$\begin{cases} \sum_{k=1}^s \phi_k \cdot p_{i,j}^k \geq d_{i,j} & \forall i, j \in 1, 2, \dots, N, \\ \text{minimizing } cost = \sum_{k=1}^s \phi_k + s\delta \end{cases}, \quad (1)$$

上式当 $\delta > 0$ 时已被证明为非多项式完全问题,有兴趣的读者可参考文献[1~2, 4],这就促使我们采用启发式方法来获得较为理想的解决方案。

2 2 近似启发算法

在对算法进行描述之前,我们先给出相关的定理和定义。

定义 1: 令 C_0 是最优调度的时延。如果一个多项式算法的时延总是至多为最优调度时延的 ρ 倍(即 ρC_0),那么我们就称之为 ρ 近似算法。

定义 2: 令 $G=(V, E)$ 为一个二分图,匹配 M 为 G 的子集,是指 G 的边集合 E 中没有任何两条边在 M 有冲突。

根据已知业务矩阵 $\mathbf{D}$ 构造一个 $N \times N$ 的二分权重图 G^D, G^D 的顶点数表示交换机端口数,边的权重表示顶点之间要传送的业务。令 $w(v)$ 表示与顶点 v 相连的边的权重之和, $W(G)$ 表示所有顶点的 $w(v)$ 的最大值, $\Delta(G)$ 表示图 G 的最大度, P^{opt} 为最优调度。我们有如下的定理:

定理 1: 最优调度的总开销至少为 $\delta \cdot \Delta(G) + W(G)$ 。

证明: 已知在一个配置中,没有两条或更多的相邻边进行调度,那么定理就包含两个方面:(1) 总的配置次数不能少于 $\Delta(G)$, 因此,总的切换时延不少

于 $\delta \cdot \Delta(G)$ 。(2) 业务实际传输时间不少于 $W(G)$ 。

定理 2: 多项式时间算法的开销 $cost \leq 2(\delta \cdot \Delta(G) + W(G))$ 。

证明: 此定理可通过根据图 G 构造一个权重完全图 G' 来证明。对于 G' , 多项式时间算法的近似因子为 2。具体参见参考文献[5]。

我们的算法就是从定理 1、2 中受到启发得来的,从图 G 中灵活地选出一个权重为 ϕ 的匹配 M , 使余下图具有最佳开销估计。简单地说,就是从 G 中移走 M , 将 M 的权重 ϕ 加上其切换延迟 δ , 再加上余留矩阵的 2 近似总开销,使总和最小。数学表达为 $\min(2(\delta \cdot \Delta(G) + W(G - M)))$ 。

算法 1: 2 近似启发算法。

输入: $N \times N$ 个非负整数矩阵 $\mathbf{D}$, 正整数 δ 。

输出: 配置矩阵 $\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_s$ 及相应的交换时间 $\phi_1, \phi_2, \dots, \phi_s$, 算法流程描述如下:

(1) 根据 $\mathbf{D}$ 构造一个二分权重图 G^D 。图 G^D 中的顶点表示交叉阵列的端口数, G^D 中的边表示输入输出端口有业务需要传送,各边的权重对应于 $\mathbf{D}$ 的元素值大小; 令 $s = 1$;

(2) 择匹配。使用贪心算法从 $\mathbf{D}$ 中找出最大值的匹配 M , 具体实现参见文献[4];

(3) 决策权重。在 M 中找出权重 ϕ 使 $\phi + \delta + 2 \times (\delta \cdot \Delta(G) + W(G))$ 最小;

(4) 根据 M 构造配置矩阵 $\mathbf{P}_s$, 以及相应的持续时间 $\phi_s = \phi$ 。

(5) $G = G - M$;

IF $G = 0$, EXIT

ELSE $s = s + 1$, go to Step 2。

在计算上述的 $(G - M)$ 时,如果有边的权重为非正数,则将此边从图 G^D 中删去。

定理 3: 2 近似算法为一启发式算法。

证明: 如果我们忽略算法的第 3 步,就变成了典型的贪心算法。但在第 3 步中,我们找出的 ϕ 能使余下图具有最佳开销估计,也就保证此算法能得到比纯贪心算法更小的近似因子。

我们的仿真表明,尽管我们的目的是最小化近似因子为 2 的开销,但算法在最坏情况下的性能比分析的要好得多。算法的执行过程简单描述为:(1) 在业务矩阵中依据贪心算法[4] 选出一匹配矩阵;(2) 对匹配矩阵中的元素进行相应计算,找出使 $\phi + \delta + 2(\delta \cdot \Delta(G) + W(G))$ 值最小的元素;(3) 所选匹配矩阵的配置矩阵的权重就为(2)得出的元素。

2 近似启发算法的时间复杂度由两个主要步骤实现。在第 2 步中选择匹配时的时间复杂度为 $O(N^2 \log N)$ ，第 3 步中决策权重的时间复杂度为 $O(N^2)$ 。因此，2 近似启发总的时间复杂度为 $O(N^2 \log N)$ ，等同于 DOUBLE 算法的时间复杂度，但比 ADJUST 算法的 $O(\lambda N^2 \log N)$ 要小。

3 分析与仿真

图 2 给出了 2 近似启发算法的仿真结果。图中设交换端口数 $N=32$ ，业务汇聚周期 $T=2\ 560$ ，对于不同的切换迟延 δ ，我们用算法得到了相应的开销(比如数据传送开销)。在定理 1 中，我们已知最优调度的总开销至少为 $\delta \cdot \Delta(G) + W(G)$ ，我们将算法中得到的总开销除以 $\delta \cdot \Delta(G) + W(G)$ ，得出近似因子，并与 ADJUST 和 DOUBLE 算法的相应结果进行了比较。通过仿真结果和上面的分析我们可得出 2 近似启发算法：(1)能无限接近于最优调度；(2)在业务传输开销方面优于 ADJUST 和 DOUBLE 算法；(3)时间复杂度比 ADJUST 算法的低。图 2 中每个数点为 100 个随机实例结果的平均值。

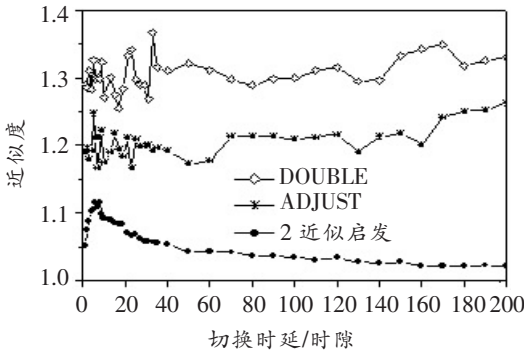


图 2 2 近似启发算法、ADJUST 算法和 DOUBLE 算法比较

为了均衡切换迟延和空白时隙所带来的开销，ADJUST 和 DOUBLE 都将待传业务矩阵 $\mathbf{D}$ 分成粗矩阵 $\mathbf{A}(a_{i,j} = \lfloor d_{i,j} / \alpha \rfloor)$ 和细矩阵 $\mathbf{B}(b_{i,j} = \max(0, d_{i,j} - \lceil \alpha \rceil \times a_{i,j}))$ ，使用简单的边着色算法进行计算。其系数 α 分别为 $\lceil T/N \rceil$ 和 $\lceil T/(\lambda N) \rceil$ ，这说明 DOUBLE 在进行计算时的确忽略了 δ 的影响。如果系统参数 δ 不同，当有相同的数据包到达两个不同

的系统时，DOUBLE 仍会使用相同的调度方式。虽然 ADJUST 将系统参数 δ 纳入了计算，但它仅仅是凭借数据汇聚周期 T 而不是实际到达的数据包进行调度。当有不同的业务到达时，它也不一定能执行得比 DOUBLE 好。比如，根据前面提到的业务矩阵 $\mathbf{D}$ ，若是均匀到达的数据流，系数 $\alpha = \sqrt{80} \delta$ ，如果 $\delta > 79$ ，ADJUST 就不可能将 $\mathbf{D}$ 分成矩阵 $\mathbf{A}$ 和 $\mathbf{B}$ 。也就是说，这时它的业务传送时延并没有对 DOUBLE 算法进行优化。2 近似启发是根据实际到来的业务矩阵 $\mathbf{D}$ 和系统参数 δ 进行调度，因此，它更具有自适应调度的功能。

4 结 论

本文提出的 2 近似启发算法适用于有切换时延的光交换，近似因子为 2，时间复杂度为 $O(N^2 \log N)$ ，仿真结果表明，我们的算法在切换开销方面比 ADJUST 和 DOUBLE 算法要好得多。今后这方面的工作包括(1)找出近似因子 < 2 的启发式算法；(2)能自动更换算法以适应各种更复杂的情况，比如光交叉阵列的开关能异步置换之类。

参考文献：

- [1] Towles B, Dally W J. Guaranteed scheduling for switches with configuration overhead [J]. IEEE Trans. Commun., 2003, 11(5): 835-838.
- [2] Li Xin, Hamd M. On scheduling optical packet switches with reconfiguration delay [J]. IEEE J. Selected Areas in Commun., 2003, 21(9): 1156-1164.
- [3] Gopal I S, Wong C K. Minimizing the number of switching in an SS/TDMA system [J]. IEEE Trans. Commun., 1985, 33(6): 497-501.
- [4] Kesselman A, Kogan K. Non-preemptive scheduling of optical switches [DB/OL]. www.mpi-sb.mpg.de/~akessel/papers/tdma.pdf, 2004.
- [5] Drake D E, Hougrady S. A simple approximation algorithm for the weighted matching problem [J]. Information Processing Letters., 2003, 85(4): 211-213.