

文章编号:1671-9352(2007)02-0046-05

波分复用星形单跳网中3信道的传输调度问题

戴珍香^{1,2}, 李曙光^{1,2}, 亓兴勤³

(1. 山东大学 数学与系统科学学院, 山东 济南 250100; 2. 烟台大学 数学与信息科学学院, 山东 烟台 264005;
3. 山东大学 威海分校应用数学系, 山东 威海 264209)

摘要:考虑波分复用星形单跳网中的数据包传输调度问题, 假定诸发送机频率可调, 而接收机频率固定. 当 $m \geq 2$ 时, 这一调度问题是 NP-完备的, m 表示所拥有的信道数目. 对目前所知最好的一个 2-近似算法进行了精细的分析, 证明了 $m = 3$ 时, 该算法近似比为 $\frac{7}{4}$, 并通过实例说明此结果为最佳可能.

关键词:波分复用; 星形网; 单跳; 数据包传输调度; 调谐时延; 近似算法; 最坏情形分析

中图分类号: O157.5; TP301 **文献标识码:** A

Three-channel transmission scheduling in WDM star single-hop networks

DAI Zhen-xiang^{1,2}, LI Shu-guang^{1,2} and QI Xing-qin³

(1. School of Math. and System Sci., Shandong Univ., Jinan 250100, Shandong, China;
2. School of Math. and Info. Sci., Yantai Univ., Yantai 264005, Shandong, China;
3. Department of Applied Math., Shandong University at Weihai, Weihai 264209, Shandong, China)

Abstract: The problem of scheduling packet transmissions in WDM star single-hop networks with tunable transmitters and fixed-tuned receivers is considered. It is NP-complete for any fixed $m \geq 2$, where m is the number of available channels. By rigorous analysis, the worst-case performance ratio of the current approximation known as the best algorithm for this problem is reduced to $\frac{7}{4}$ for three channels. An example is presented to show that this result is the best possible.

Key words: WDM; star networks; single-hop; packet transmission scheduling; tuning delay; approximation algorithms; worst-case analysis

0 引言

波分复用(wavelength division multiplexing, WDM)能充分利用光纤的巨大带宽, 降低对电子处理速率的要求, 是通信研究的一大热点. WDM单跳(single-hop)网中, 每个用户配一个接收机和一个发送机. 接收机和发送机是全频带可调的, 这样任意两个用户都能直接通信^[1]. 网络采用包交换(packet switching)技术, 所有数据包长度相同, 传输一个数据包需要一个单位时间. WDM网络中, 一个重要的参数是“调谐时延”(tuning delay). 调谐时延是指发送机(或接收机)从一个波长调到另一个波长所用的时间.

收稿日期:2005-10-27

基金项目:国家自然科学基金资助项目(60373025)

作者简介:戴珍香(1965-), 女, 硕士研究生, 研究方向:经济数学.

本文考虑 WDM 单跳网中的数据包的传输调度问题(以下简称传输调度问题). 网络结构为星形, 有 n 个用户和 m 个信道, 每个信道使用一个特定频率的波长. 假定发送机是全频带可调的, 而接收机频率固定. 在任一时刻, 一个发送机至多在一个信道上发送, 一个信道至多携带一个数据包. 问题的目标是用最短时间来实现所有的传输请求(数据包).

WDM 网络中可供使用的波长数十分有限, 总是远远小于网络用户数, 因此研究数据包的传输调度, 对于提高网络吞吐量有重要意义. 受现有技术条件的限制, 可调光器件的调谐时延比较大, 且在今后较长一段时间之内都不能忽略不计^[2]. 因此在设计算法时, 必须考虑到调谐时延的影响.

传输调度问题已经引起了很多人的兴趣^[2~6]. 对于均匀传输模式(用户之间传输相同数目的数据包), 文献[3]给出了分组传输算法, 文献[2]证明了这一算法是最优的. 当传输模式满足一定条件时, 文献[4]给出了一个多项式时间精确算法. 对于非均匀传输模式(用户之间传输任意数目的数据包), 文献[2]给出了近似比为 2 的列表调度算法(List Scheduling Algorithm), 文献[5]证明了 $m = 2$ (两个信道)时, 该算法近似比可降低为 $\frac{3}{2}$.

传输调度问题类似于熟知的开放式工作站调度问题(Open Shop Scheduling Problem). 事实上, 如果调谐时延为 0, 我们就得到了后者. 当机器数目 $m \geq 3$ 时, 开放式工作站调度问题是 NP-完备的^[7]; 文献[8]给出了一个 2-近似算法; 文献[9]证明了 $m = 3$ 时, 该算法的近似比是 $\frac{5}{3}$. 当 $m = 2$ 时, 开放式工作站调度问题是多项式时间之内可解的^[7]. 从这一点上来说, 传输调度问题要更难一些, 因为 $m \geq 2$ 时, 它总是 NP-完备的^[6].

本文对文献[2]中的列表调度算法进行了细致的分析, 证明了 $m = 3$ 时, 该算法近似比可降为 $\frac{7}{4}$, 并通过实例说明这一结果是最佳可能的.

全文组织如下: 第 1 节定义了传输调度问题并介绍了列表调度算法. 第 2 节证明了 $m = 3$ 时列表调度算法是 $\frac{7}{4}$ -近似算法, 并给出一个实例说明 $\frac{7}{4}$ 是最佳可能的. 第 3 节是结束语.

1 预备知识

给定一个极小化问题, 若对于该问题的任意实例, 算法 A 都能在多项式时间之内找到一个目标值不超过 ρ 倍最优值的近似解, 则称算法 A 是该极小化问题的一个 ρ -近似算法. ρ 称为算法 A 的近似比.

本文所考虑的传输调度问题, 其定义如下:

设有 m 个信道(波长) C_1, C_2, \dots, C_m 和 n 个用户 $1, 2, \dots, n$. 用户 i 配备一个全频带可调发送机 t_i 和一个频率固定(已预先调到某个波长上)的接收机 r_i . 设 $p(t_i, r_k)$ 表示由 t_i 发送到 r_k 的数据包的数目 ($1 \leq i, k \leq n$). 由于 r_k 已预先调到某个特定的信道 C_j 上, 因此所有 r_k 接收的数据包必须通过信道 C_j 传输. 另一方面, 由于发送机是全频带可调的, 因此可以通过任一信道发送数据包. 设 $p(t_i, C_j)$ 表示由 t_i 通过信道 C_j 发送的数据包的数目 ($1 \leq i \leq n, 1 \leq j \leq m$). 在任一时刻, 一个发送机至多在一个信道上发送, 一个信道至多携带一个数据包. 传输一个数据包需要 1 个单位时间. 发送机调到某一信道, 要花费 δ 个单位时间(调谐时延). 初始时, 发送机没有调到任何特定信道. 给定一组数据包, 我们要设计一个调度方案, 使得所有的数据包得到传输. 调度的长度定义为至少有一个信道忙于接收数据包的最迟时间(调度的开始时间为 0). 问题的目标是寻找长度最小的调度.

为叙述方便, 我们约定如下术语: 当数据包通过某个信道传输, 我们称该信道为忙碌的, 否则称为闲置的. 注意到在初始调频阶段(时间区间 $[0, \delta)$ 内), 任一信道不能携带数据包, 因此约定这一段上信道也是忙碌的. 当一个发送机正调往某一信道或正在发送数据包, 我们称该发送机为忙碌的, 否则称为闲置的. 设 $\text{Busy}(C_j)$ 和 $\text{Idle}(C_j)$ 分别表示信道 C_j 的总忙碌时间和总闲置时间, $\text{Busy}(t_i)$ 和 $\text{Idle}(t_i)$ 分别表示发送机 t_i 的总忙碌时间和总闲置时间. 设 $p(t_i)$ 为由 t_i 发送的数据包的总数目, $c(t_i)$ 为 t_i 在发送过程中所用到的信道数, $p(C_j)$ 为通过信道 C_j 传输的数据包的总数目.

设 L_{OPT} 为最优调度的长度. 文献[2]给出了 L_{OPT} 的两个下界.

事实 1 $L_{Busy} \geq \max_{1 \leq i \leq n} \text{Busy}(t_i) = \max_{1 \leq i \leq n} \{p(t_i) + \delta \cdot c(t_i)\}.$

事实 2 $L_{OPT} \geq \max_{1 \leq j \leq m} \text{Busy}(C_j) = \max_{1 \leq j \leq m} \{p(C_j) + \delta\}.$

Choi et al.^[2]应用列表调度算法来求解传输调度问题,证明了该算法为 2-近似算法. 我们在此简单介绍这一算法.

列表调度算法

(1) 信道选择规则: 当发送机 t_i 完成在一个信道上的传输之后, 立即调往能使它最早开始下一次传输的那个信道 C_j .

(2) 传输规则: 如果 t_i 已经调到信道 C_j 上, 而 C_j 又恰好闲置, 那么 t_i 马上开始通过 C_j 传输, 并且一直持续到它在 C_j 上完成传输任务为止.

2 主要结果

当只有两个信道时, Dasgupta 和 Palis^[5]证明了列表调度算法的近似比是 $\frac{3}{2}$. 本节中, 我们将证明当有 3 个信道时, 列表调度算法的近似比为 $\frac{7}{4}$, 并给出一个实例说明这一结果是最佳可能的.

定理 1 $m = 3$ 时, 列表调度算法的近似比为 $\frac{7}{4}$.

证明 设 L 和 L_{OPT} 分别表示列表调度算法和最优算法所产生的调度的长度. 不失一般性, 设列表调度算法所产生的调度中, t_n 是最后完成传输任务的发送机, 它依次用到信道 C_1, C_2, C_3 , 通过 3 个信道传输的数据包的数目依次为 a, b, c . 设 t_n 调到 C_j 上以后等待 C_j 出现闲置所用的时间为 i_j (如图 1 所示). 注意调度的开始时间为 0.

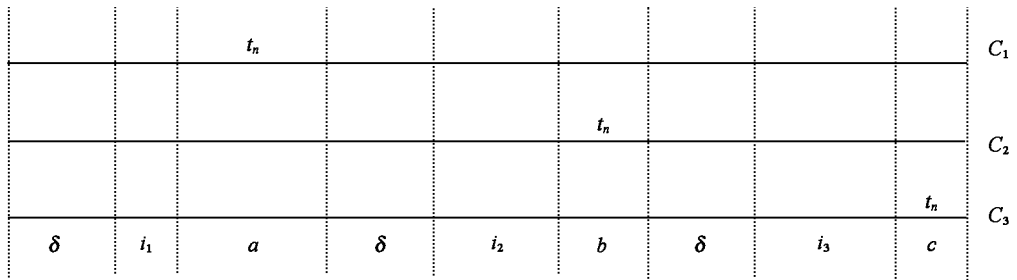


图 1 列表调度算法所产生的调度
Fig. 1 A scheduling produced by list scheduling algorithm

图 1 只显示了一般情形. 有可能出现 $abc = 0$, 从而一个或多个 δ 在图 1 中不出现.

根据列表调度算法, 在时间区间 $[\delta, \delta + i_1]$ 上, C_1, C_2 和 C_3 必定都忙碌. 同理, 在时间区间 $[2\delta + i_1 + a, 2\delta + i_1 + a + i_2]$ 上, C_2 和 C_3 必定都忙碌.

我们要考虑两种情形: (1) $i_3 = 0$, (2) $i_3 > 0$.

情形 1 $i_3 = 0$.

首先考虑 $abc = 0$ 的情形. a, b, c 中至少有两个为 0 时证明是平凡的. 故可认为 a, b, c 中恰有一个为 0, 不妨设 $b = 0$. 由第 1 节中 L_{OPT} 的两个下界可得: $L_{OPT} \geq \text{Busy}(t_n) = 2\delta + a + c$, $L_{OPT} \geq \max\{\text{Busy}(C_1), \text{Busy}(C_3)\}$. 另一方面, 我们有: $\text{Idle}(C_1) \leq \delta + c$, $\text{Idle}(C_3) \leq \delta + a$, $\min\{\text{Idle}(C_1), \text{Idle}(C_3)\} \leq \frac{\delta + c + \delta + a}{2} \leq$

$$\frac{1}{2} L_{OPT}. \text{ 因此有: } L = \max\{\text{Busy}(C_1), \text{Busy}(C_3)\} + \min\{\text{Idle}(C_1), \text{Idle}(C_3)\} \leq L_{OPT} + \frac{1}{2} L_{OPT} = \frac{3}{2} L_{OPT}.$$

现在可设 $abc \neq 0$, 故有 $L_{OPT} \geq \text{Busy}(t_n) = 3\delta + a + b + c$.

考虑两种子情形:

子情形 1.1 $a \leq \delta$.

由于 $\text{Idle}(C_2) \leq 2\delta + a + c$, $\text{Idle}(C_3) \leq 2\delta + a + b$, 故 $\min\{\text{Idle}(C_2), \text{Idle}(C_3)\} \leq 2\delta + a + \frac{b+c}{2} < \frac{3}{4}(3\delta + a + b + c) \leq \frac{3}{4}L_{\text{OPT}}$. 因此有 $L = \max\{\text{Busy}(C_2), \text{Busy}(C_3)\} + \min\{\text{Idle}(C_2), \text{Idle}(C_3)\} < L_{\text{OPT}} + \frac{3}{4}L_{\text{OPT}} = \frac{7}{4}L_{\text{OPT}}$.

子情形 1.2 $a > \delta$.

设时间区间 $[\delta + i_1, \delta + i_1 + a + \delta)$ 内, C_2 和 C_3 同时闲置的子区间的总长度为 y , C_2 闲置但 C_3 忙碌的子区间的总长度为 y_2 , C_3 闲置但 C_2 忙碌的子区间的总长度为 y_3 . 显然有 $y_2 + y_3 + y \leq a + \delta$. 若 $y \leq 2\delta$, 我们有: $\text{Idle}(C_2) \leq y + y_2 + \delta + c$, $\text{Idle}(C_3) \leq y + y_3 + \delta + b$. 因此有:

$$\min\{\text{Idle}(C_2), \text{Idle}(C_3)\} \leq \frac{y + y_2 + \delta + c + y + y_3 + \delta + b}{2} \leq \frac{5\delta}{2} + \frac{a + b + c}{2} < \frac{3}{4}(3\delta + a + b + c) \leq \frac{3}{4}L_{\text{OPT}},$$

$$L < L_{\text{OPT}} + \frac{3}{4}L_{\text{OPT}} = \frac{7}{4}L_{\text{OPT}}.$$

若 $y > 2\delta$, 注意到此时必有 $i_2 = 0$. (否则, 设 t_j 是时间区间 $[2\delta + i_1 + a, 2\delta + i_1 + a + i_2)$ 内第一个通过 C_2 传输数据包的发送机. 不论 t_j 是初始时就调到 C_2 还是从 C_1 或者 C_3 调过来, 我们都很容易得到矛盾.) 因此我们有: $\text{Idle}(C_1) \leq 2\delta + b + c$, $\text{Idle}(C_2) \leq 2\delta + a + c$, $\text{Idle}(C_3) \leq 2\delta + a + b$. 故有:

$$\min\{\text{Idle}(C_1), \text{Idle}(C_2), \text{Idle}(C_3)\} \leq \frac{2}{3}(3\delta + a + b + c) \leq \frac{2}{3}L_{\text{OPT}}, L \leq \frac{5}{3}L_{\text{OPT}}.$$

情形 2 $i_3 > 0$.

从时刻 $3\delta + i_1 + i_2 + a + b$ 回溯到时刻 $\delta + i_1 + z$ ($1 \leq z \leq 2\delta + i_2 + a + b$), 使得信道 C_3 在时间区间 $[\delta + i_1 + z, 3\delta + i_1 + i_2 + a + b)$ 内忙碌而在时间区间 $[\delta + i_1 + z - 1, \delta + i_1 + z)$ 内闲置. 设 t_j 是从时刻 $\delta + i_1 + z$ 开始通过 C_3 传输数据包的发送机. 令 $k = \min\{c(t_j), c(t_n)\}$. (第1节中约定 $c(t_i)$ 表示 t_i 在整个调度中所用到的信道数.) k 可能为 1, 2, 3. 枚举各种可能性后, 可知在时间区间 $[\delta + i_1, \delta + i_1 + z)$ 内, 至少有一个信道的闲置时间不超过 $(k-1)\delta$. 这意味着:

$$L \leq L_{\text{OPT}} + (k-1)\delta + [3\delta + i_1 + i_2 + i_3 + a + b + c - (\delta + i_1 + z)]. \quad (1)$$

考虑两种子情形.

子情形 2.1 在时间区间 $[\delta + i_1 + z, 3\delta + i_1 + i_2 + i_3 + a + b + c)$ 内, 只有两个发送机, 即 t_j 和 t_n , 通过信道 C_3 传输数据包.

设发送机 t_j 通过信道 C_1, C_2 和 C_3 传输的数据包的数目分别为 a_j, b_j, c_j .

由 (1), 我们得到:

$$\min\{\text{Idle}(C_1), \text{Idle}(C_2), \text{Idle}(C_3)\} \leq (k-1)\delta + c_j + c, \quad (2)$$

由列表调度算法我们有:

$$\text{Idle}(C_3) \leq \min\{a_j + b_j + [c(t_j) - 1] \cdot \delta, a + b + [c(t_n) - 1] \cdot \delta\}, \quad (3)$$

结合 (2) 和 (3), 我们有:

$$\min\{\text{Idle}(C_1), \text{Idle}(C_2), \text{Idle}(C_3)\} \leq \frac{(k-1)\delta + c_j + c + a_j + b_j + [c(t_j) - 1] \cdot \delta + a + b + [c(t_n) - 1] \cdot \delta}{3} \leq \frac{a_j + b_j + c_j + c(t_j) \cdot \delta}{3} + \frac{a + b + c + c(t_n) \cdot \delta}{3} \leq \frac{2}{3}L_{\text{OPT}}.$$

故有: $L \leq \frac{5}{3}L_{\text{OPT}}$.

子情形 2.2 在时间区间 $[\delta + i_1 + z, 3\delta + i_1 + i_2 + i_3 + a + b + c)$ 内, 至少有三个发送机通过信道 C_3 传输数据包.

任取 3 个这样的发送机 t_1, t_2, t_3 . 不妨设 $c(t_1) \leq c(t_2) \leq c(t_3)$. 考虑 5 种情形: $c(t_1) = 1; c(t_1) = c(t_2) = c(t_3) = 2; c(t_1) = c(t_2) = 2, c(t_3) = 3; c(t_1) = 2, c(t_2) = c(t_3) = 3; c(t_1) = c(t_2) = c(t_3) = 3$. 取

$c(t_1) = 2, c(t_2) = c(t_3) = 3$ 为例加以说明, 其它的情形类似. 为叙述方便, 不妨认为 t_3 就是 t_n . 设 t_2 初始时调到 C_2 , 先于 t_3 在 C_2 上传输, t_2 在 C_2 上完成传输之后调到 C_1 ; t_1 初始时调到 C_1 , 先于 t_2 后于 t_3 在 C_1 上传输. 以此为例加以说明, 其它的类似. 此时有:

$$\text{Idle}(C_3) \leq 2\delta + \min\{b, \delta\} \leq \frac{3}{4}(3\delta + a + b + c) \leq \frac{3}{4}L_{\text{OPT}}.$$

因此有: $L \leq \frac{7}{4}L_{\text{OPT}}$.

我们给出下例, 以说明 $\frac{7}{4}$ 是最佳可能的.

设有3个信道和9个用户. 通讯矩阵如图2所示, 其中 (t_i, C_j) 项表示由 t_i 发送到固定在信道 C_j 上的接收机的数据包的总数目. 注意这里的 δ 就是调谐时延. 最优调度的长度为 $L_{\text{OPT}} = 4\delta + 3$. 另一方面, 我们给出列表调度算法得到的一个调度, 其长度为 $L = \text{Busy}(t_3) + \text{Idle}(t_3) = 7\delta + 3$. 这样, 对于充分大的 δ , 我们有 $\frac{L}{L_{\text{OPT}}} \rightarrow \frac{7}{4}$. (当然, 这只是理论上的分析. 在实际应用中, δ 总是有限的.) 为简明起见, 我们只给出 $\delta = 3$ 时的两个调度, 如表1所示.

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
C_1	1	1	$\delta+1$	δ	$\delta-2$	0	0	0	0
C_2	2	δ	1	0	0	δ	δ	0	0
C_3	δ	1	1	0	0	0	0	δ	δ

图2 通讯矩阵
Fig.2 Traffic matrix

表1(a) 一个最优调度
Table 1(a) An optimal scheduling

time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
C_1				t_3	t_3	t_3	t_3	t_4	t_4	t_4	t_5			t_2	t_1
C_2				t_2	t_2	t_2	t_6	t_6	t_6	t_1	t_1	t_7	t_7	t_7	t_3
C_3				t_1	t_1	t_1	t_8	t_8	t_8	t_2	t_3	t_9	t_9	t_9	

表1(b) 列表调度算法得到的一个调度
Table 1(b) A scheduling produced by list scheduling algorithm

time	1	2	3	4	5	6	7	8	9	10	11	12
C_1				t_4	t_4	t_4	t_5	t_1	t_2	t_3	t_3	t_3
C_2				t_6	t_6	t_6	t_7	t_7	t_7			t_1
C_3				t_8	t_8	t_8	t_9	t_9	t_9			
time	13	14	15	16	17	18	19	20	21	22	23	24
C_1	t_3											
C_2	t_1			t_2	t_2	t_2		t_3				
C_3	t_2			t_1	t_1	t_1						t_3

3 结束语

本文证明了有3个信道时, 列表调度算法的近似比为 $\frac{7}{4}$, 并且构造了一个实例说明这一结果是最佳可能的. 很自然的一个问题是: $m = 3$ 时列表调度算法的最坏情形分析能否推广到任意的 m , 从而得到好于2的近似比? 希望我们的结果对于这个问题能有所启发.

(上接第50页)

参考文献:

- [1] B Mukherjee. WDM-based local lightwave networks—Part I : Single-hop systems[J]. IEEE Network, 1992, 7(3):12 ~ 27.
- [2] H Choi, H A Choi, M Azizoglu. Efficient scheduling of transmissions in optical broadcast networks[J]. IEEE/ACM Transactions on Networking, 1996, 4(6):913 ~ 920.
- [3] G R Pieris, G H Sasaki. Scheduling transmissions in WDM broadcast-and-select networks[J]. IEEE/ACM Transactions on Networking, 1994, 2(2):102 ~ 110.
- [4] A Dasylya, R Srikant. Optimal WDM schedules for optical star networks[J]. IEEE/ACM Transactions on Networking, 1999, 7(3):446 ~ 456.
- [5] B Dasgupta, M A Palis. Provably good algorithms for transmission scheduling in WDM optical networks[J]. Journal of Parallel and Distributed Computing, 1999, 57:345 ~ 357.
- [6] G N Rouskas, V Sivaraman. Packet scheduling in broadcast WDM networks with arbitrary transceiver tuning latencies[J]. IEEE/ACM Transactions on Networking, 1997, 5(3):359 ~ 370.
- [7] T Gonzalez, S Sahni. Open shop scheduling to minimize finish time[J]. Journal of ACM, 1976, 23(4):665 ~ 679.
- [8] V A Aksjonov. A polynomial-time algorithm of approximate solution of a scheduling problem[J]. Upravlyaemye Sistemy, 1988, 28:8 ~ 11.
- [9] B Chen, V Strusevich. Approximation algorithms for three-machine open shop scheduling[J]. ORSA Journal on Computing, 1993, 5: 321 ~ 326.

(编辑:李晓红)

