运筹学与控制论

部分工件必须不误工的误工排序问题*

彭洪洁[1]，苏永英[1]，唐国春[1,2]
(1. 重庆师范大学 数学与计算机科学学院，重庆 400047；2. 上海第二工业大学 管理工程研究所，上海 201209)

摘要 排序论中使误工工件的个数为最少的单台机器排序问题，称为误工问题，是排序论中最基本的问题之一。1973 年，Sidney 研究在工件的一个子集 T 中的工件必须不误工的条件下，使误工工件的个数为最少的误工排序问题 $1 \mid T \mid \sum U_j$，并且给出该问题复杂性为 $O(n \log n)$ 的多项式算法——Sidney 算法。本文把 Sidney 算法改写成比较简洁的算法 1，1) 步骤 1 设 $E_0 = T, J - E_0 = \{j_1, j_2, \cdots, j_m\}$，$j_1 < j_2 < \cdots < j_m$，$m = n - |T|$，令 $k = 1$；2) 步骤 2 若 $k = m + 1$，算法终止，$(E_m, J - E_m)$ 就是最优排序；若 $k < m + 1$，转入步骤 3；3) 步骤 3 设 $F_k = E_{k-1} \cup \{j_k\}$，计算 E_k 如下：如果 F_k 是不误工子集，令 $E_k = E_{k-1} \cup \{j_k\}$；否则，如果 F_k 不是不误工子集，令 $E_k = F_k \setminus \{j_r\}$。其中工件 j_r 的加工时间为 $p_r = \max\{p_i \mid j_i \in F_k \setminus T\}$。$E_k$ 中的工件是按 EDD 序排列。$k = k + 1$，转入步骤 2。并用数学归纳法证明算法 1 产生的排序是该误工问题的最优解。

关键词 排序；误工；算法；最优性
中图分类号：O223 **文献标识码**：A **文章编号**：1672-6693(2009)02-0018-04

排序论中使误工工件的个数为最少的单台机器排序问题，称为误工问题[1]，是排序论中最基本的问题之一，具有重要的理论意义和实用价值。1968 年 Moore[2] 提出解决这个误工排序问题的算法。这个算法经过 Hodgson 的修改，称为 Moore-Hodgson 算法，可以在时间 $O(n \log n)$ 内得到误工问题的最优解。对于 Moore-Hodgson 算法最优性的证明也已经有许多改进，其中最简洁的证明是文献[3]给出的。同时误工排序问题也有不少推广。例如，要求某些工件是必须不误工的[4]，或者工件的就绪时间不相同、但是与交货期有“一致性”关系的[5]，或者工件的加工时间与工件的权有反向“一致性”关系的[6]等等。文献[7]对误工排序问题的部分成果进行了总结、介绍。

本文研究工件的一个子集 T 中的工件必须不误工的条件下，使误工工件的个数为最少的误工排序问题 $1 \mid T \mid \sum U_j$。1973 年，Sidney 给出该问题复杂性为 $O(n \log n)$ 的多项式算法——Sidney 算法。本文把 Sidney 算法改写成比较简洁的算法 1，并用数学归纳法证明算法 1 产生的排序是该误工问题的最优解。

1 问题的描述

设有 n 个工件 $J_1, J_2, \cdots, J_n$ 要在一台机器上加工。记这 n 个工件的集合为 J，即 $J = \{J_1, J_2, \cdots, J_n\}$。在不至于混淆的情况下，也用工件 J_j 的下标 j 来表示这个工件。因此，这 n 个工件的集合也可以写成 $J = \{1, 2, \cdots, n\}$。已知工件 j 的加工时间是 p_j，交货期是 d_j。这些工件已经全部准备就绪，可以进行加工。这台机器每次只能加工一个工件，并且加工不允许中断，只有当一个工件加工完成后才能加工其他工件。并且要求某些指定的工件必须不误工，以 T 表示指定必须不误工的工件的集合，试问：如何安排一个加工次序，使得那些指定的工件都不误工，又使误工工件的个数为最少，这就是使某些工件必须不误工条件下的误工排序问题，用三参数表示为 $1 \mid T \mid \sum U_j$。

不失一般性，假设工件已经按照交货期的从小到大的次序，即 EDD(earliest due date)序进行编号，所以

* 收稿日期：2008-12-15
资助项目：国家自然科学基金(No. 70731160015)；运筹学与系统工程重庆市市级重点实验室(No. YC200802)
作者简介：彭洪洁，男，硕士研究生，研究方向为组合最优化；通讯作者：唐国春，E-mail：gtang@sh163.net。

有 $d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$。这时工件 j 的下标是工件在 EDD 序中的序号。对于工件集合 J 的子集 σ,在不至于混淆的情况下,σ 也表示这些工件按照下标从小到大排成的排序,即工件集合 σ 的 EDD 序。

下面引理 1 到引理 3 及其证明与文献[8] 中是类似的,在此略去证明了。

定义 1[8] 如果排序 σ 中每一个工件都是不误工的,把子集 σ 称为是不误工子集。

引理 1 如果排序 $\sigma = \{t_1, t_2, \cdots, t_m\}$ 中的工件都是不误工的,那么排序 σ 的 EDD 序中的工件也都是不误工的,反之亦然。

定义 1 是根据 EDD 序来定义不误工子集,可由引理 1 把定义 1 推广到一般。

定义 2[8] 对于工件集 J 的子集 σ,如果在子集 σ 的一个排序中这些工件都是不误工的,那么这个子集称为是不误工子集。

引理 2 推广的误工问题 $1 \mid T \mid \sum U_j$ 有最优解的必要条件是 T 必须是不误工子集。

引理 3 推广的误工问题 $1 \mid T \mid \sum U_j$ 存在这样的最优解,其可以分为前后两部分,1) 不误工的工件的全体 E 按 EDD 序排在前面;2) 误工工件集 L,以任意次序排在 E 的后面。

2 Sidney 算法及其最优性

下面是 J. B. Sidney 对于推广的误工排序问题 $1 \mid T \mid \sum U_j$ 给出的算法 ——Sidney 算法。

步骤 1 把工件按交货期从小到大的次序编号,若有工件的交货期相同,则把其中必须不误工的工件排在后面,并且把必须不误工的工件都标上 $*$ 号。

步骤 2 以 T 表示带 $*$ 号所有工件构成的集合,令 m 表示 T 中的最大标号。计算新的交货期 $d'_1, d'_2, \cdots, d'_n$ 如下。先计算 d'_n,再依次计算 $d'_{n-1}, \cdots, d'_1$。当 $i \geqslant m$ 时,令 $d'_i = d_i$;当 $i \leqslant m-1$ 时,若 r 是 i 后面第一个属于 T 的标号,也即工件 J_r^* 是必须不误工的工件中紧排在 J_i 后面的工件,则令:$d'_i = \min\{d_i, d'_r - p_r\}$。

步骤 3 定义工件集 $E_0, E_1, \cdots, E_n$,其中 $E_0 = \varnothing$;若 $E_{k-1}(k \geqslant 1)$ 已经定义,而且 $C(E_{k-1}) + p_k \leqslant d'_k$,那么定义 $E_k = E_{k-1} \cup \{k\}$;否则,如果 $C(E_{k-1}) + p_k > d'_k$,那么定义 $E_k = E_{k-1} \cup \{k\} \setminus \{r\}$,其中 $J_r \in E_{k-1} \cup \{k\} \setminus T$,并且满足 $p_r = \max\{p_j \mid j \in E_{k-1} \cup \{k\} \setminus T\}$。这里 $C(E_{k-1})$ 表示 E_{k-1} 中所有工件的加工时间之和。当得到 E_n 后,算法终止,$(E_n, J - E_n)$ 就是最优排序,显然 $T \subset E_n$。

实际上,上面的算法是可以不计算 d'_i 的,此时的算法可以改写为如下比较简洁的算法 1 。

算法 1

步骤 1 设 $E_0 = T, J - E_0 = \{j_1, j_2, \cdots, j_m\}, j_1 < j_2 < \cdots < j_m, m = n - |T|$,令 $k = 1$;

步骤 2 若 $k = m+1$,算法终止,$(E_m, J - E_m)$ 就是最优排序;若 $k < m+1$,转入步骤 3;

步骤 3 设 $F_k = E_{k-1} \cup \{j_k\}$,计算 E_k 如下。如果 F_k 是不误工子集,令 $E_k = E_{k-1} \cup \{j_k\}$;否则,如果 F_k 不是不误工子集,令 $E_k = F_k \setminus \{j_r\}$。其中工件 j_r 的加工时间为 $p_r = \max\{p_i \mid j_i \in F_k \setminus T\}$。$E_k$ 中的工件是按 EDD 序排列,$k = k+1$,转入步骤 2。

定理 1 算法 1 所产生的排序是误工排序问题 $1 \mid T \mid \sum U_j$ 的最优解。

证明 用 $S = (E, L)$ 表示一个排序,其中 E 是不误工工件集,L 是误工工件集。令 b 是算法删除的第一个工件。

首先证明存在一个排序 $S = (E \cup T, L)$,其中 $b \in L$,而且这个排序是问题的最优解。用 k 表示算法第一个误工工件,A 表示集合 $J - T = \{j_1, j_2, \cdots, j_m\}$,并且 $j_1 < j_2 < \cdots < j_m$,记 $p_b = \max\{p_i \mid i = 1, \cdots, k; \forall i \in A\}$。由于在 $\{1, 2, \cdots, k-1\} \cup T$ 中是没有误工工件的,又由于 $p_k \leqslant p_b$,因而在 $\{1, 2, \cdots, b-1, b+1, \cdots, k\} \cup T$ 中是没有误工工件的。

对于问题的一个最优解 $S' = (E' \cup T, L')$,如果 $b \in L'$,那么就不必证明了。现在只需考虑在最优解 $S' = (E' \cup T, L')$ 中 $b \in E'$ 的情况。把排序 (E', L') 记为

$$\pi: \pi(1), \cdots, \pi(a), \pi(a+1) \cdots, \pi(r), \pi(r+1), \cdots, \pi(n)$$

其中

$$L' = \{\pi(r+1), \cdots, \pi(n)\} \tag{1}$$

$$d_{\pi(1)} \leqslant \cdots \leqslant d_{\pi(r)} \tag{2}$$

$$\{\pi(1),\cdots,\pi(a)\} \subseteq \{1,\cdots,k\} \tag{3}$$

$$\{\pi(a+1),\cdots,\pi(r)\} \subseteq \{k+1,\cdots,m\} \tag{4}$$

$$b \in \{\pi(1),\cdots,\pi(a)\} \tag{5}$$

因为$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$,所以总存在满足条件(3)和(4)式的$a$。再者,因为$b \in E' \cap \{1,\cdots,k\}$,所以(5)式也成立。因为$\{\pi(1),\cdots\pi(a)\} \cup T \subseteq E' \cup T$,所以$\{\pi(1),\cdots,\pi(a)\} \cup T$中没有误工工件。另一方面,在任何排序中如果有工件$\{1,\cdots,k\} \cup T$,那么在这个排序中一定有一个工件误工[3],因此有

$$\{\pi(1),\cdots,\pi(a)\} \cup T \subset \{1,\cdots,k\} \cup T$$

这就意味着,有一个工件$h(1 \leqslant h \leqslant k)$是$h \notin \{\pi(1),\cdots,\pi(a)\}$。又因为$\{\pi(a+1),\cdots,\pi(r)\} \subseteq \{k+1,\cdots,m\}$,那么$h$在最优解$S' = (E' \cup T,L')$中是属于误工集$L'$的。从$\{\pi(1),\cdots,\pi(a)\}$里删除工件$b$,并用$h$替代它,把$\{\pi(1),\cdots,\pi(a)\} \cup \{h\} \setminus \{b\} \cup T$中所有工件按 EDD 序排列,因为$\{1,\cdots,k\} \setminus \{b\} \cup T$中的工件是不误工的,并且有　$\{\pi(1),\cdots,\pi(a)\} \cup \{h\} \setminus \{b\} \cup T \subseteq \{1,\cdots,k\} \setminus \{b\} \cup T$

所以$\{\pi(1),\cdots,\pi(a)\} \cup \{h\} \setminus \{b\} \cup T$中的工件也是不误工的。因为$\{\pi(a+1),\cdots,\pi(r)\} \subseteq E' \cup T$,所以$\{\pi(a+1),\cdots,\pi(r)\}$在$E' \cup T$中是没有误工工件的。如果把$\{\pi(a+1),\cdots,\pi(r)\}$加入集合$\{\pi(1),\cdots,\pi(a)\} \cup \{h\} \setminus \{b\} \cup T$,并把所有工件按 EDD 序排列记为$E' \cup \{h\} \setminus \{b\} \cup T$,由于$d_h \leqslant d_k$,那么工件$\pi(a+1),\cdots,\pi(r)$在这个排序中的位置与在排序$E' \cup T$中的位置一样,并且由$p_h \leqslant p_b$,所以有

$$\sum_{i=1}^{a} p_{\pi(i)} - p_b + p_h \leqslant \sum_{i=1}^{a} p_{\pi(i)}$$

因此$\{\pi(a+1),\cdots,\pi(r)\}$在这个排序中也是不误工的。又$\{\pi(1),\cdots,\pi(a)\} \cup \{h\} \setminus \{b\} \cup T$中的工件是不误工的,从而$E' \cup \{h\} \setminus \{b\} \cup T$中所有工件都是不误工的,并且与最优解$S' = (E' \cup T,L')$中的不误工工件个数相等。这样从$\{\pi(1),\cdots,\pi(a)\}$里删除了工件$b$,并用$h$替代它,相当于把误工工件集$L'$中的$h$与不误工工件集$E'$中而不在$T$的$b$互换得到了一个新的排序。记新的排序为$S = (E \cup T,L)$。从上面证明知排序$S = (E \cup T,L)$的不误工工件的个数与最优解$S' = (E' \cup T,L')$中的不误工工件的个数一样,所以$S = (E \cup T,L)$也是最优解。因此得到了一个最优解$S = (E \cup T,L)$,其中$b \in L$。这种排序的存在性就证明了。

现用数学归纳法来证明算法 1 得到的是最优解。$n = 1$时算法 1 是最优解是很明显的。假设对$n-1$个工件算法 1 得到的是最优解,下面要证明对n个工件算法 1 得到的是最优解。

设$S = (E \cup T,L)$是算法 1 得到的解,记b是算法 1 第一个删除的工件。由上面的证明可知存在最优解$S' = (E' \cup T,L')$,其中$b \in L'$。

用$|\cdot|$表示工件的个数。若有$|E+T| = |E'+T|$,则算法 1 得到的解是最优解。因为$|E+T| = |E| + |T|$,$|E'+T| = |E'| + |T|$,为此,下面只要证明$|E| = |E'|$,就完成了证明。

考虑工件集$\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$。该工件集的工件个数是为$n-1$,由算法 1,对工件集$\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$实行算法时,在步骤 3 中依次执行$j = 1,\cdots,b-1,b+1,\cdots,k$,此时$\{1,\cdots,b-1,b+1,\cdots,k\} \cup T$都不误工,并且都放入$E$集。这时$E$集为$\{1,\cdots,b-1,b+1,\cdots,k\} \cup T$,再从$j = k+1$开始执行余下步骤。对工件集$\{1,\cdots,b-1,b,b+1,\cdots,k,k+1,\cdots,m\} \cup T$实行算法 1,在步骤 3 执行到$j = k$时误工,再从$E$集删除$b$,此时$E$集中工件也为$\{1,\cdots,b-1,b+1,\cdots,k\} \cup T$,并且$L$集为$\{b\}$,再从$j = k+1$开始执行余下步骤。从而两者得到的不误工集$E$的工件和个数都相同,而误工集$L$前者比后者少了工件$b$。因而,对工件集$\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$实行算 1 得到的解是$(E \cup T, L \setminus \{b\})$,由工件个数为$n-1$时的假设知道,$(E \cup T, L \setminus \{b\})$是最优解。

另一方面,由于$S' = (E' \cup T,L')$,其中$b \in L'$是$\{1,\cdots,b-1,b,b+1,\cdots,k,k+1,\cdots,m\} \cup T$的最优解。工件集$\{1,\cdots,b-1,b,b+1,\cdots,k,k+1,\cdots,m\} \cup T$的工件个数比工件集$\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$的个数多一个,所以它的不误工工件个数不可能小于$|E| + |T|$,即$|E| + |T| \leqslant |E'| + |T|$,即$|E| \leqslant |E'|$。因为$E' \cup T$为不误工工件集,且$b \in L'$,那么在$L'$中删除$b$不会改变最优解$S' = (E' \cup T,L')$的不误工集$E' \cup T$,即$E' \cup T \subset \{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$。那对于工件集$\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,m\} \cup T$,就得到了一个形如$(E' \cup T,L' \setminus \{b\})$这样的排序,其中前面$E' \cup T$是不

误工的工件集,后面 $L'\setminus\{b\}$ 是误工的工件集。由于工件集 $\{1,\cdots,b-1,b+1,\cdots,k,k+1,\cdots,n\}$ 实行算法 1 得到的解 $(E\cup T, L\setminus\{b\})$ 是最优解,有 $|E|+|T|\geqslant|E'|+|T|$,所以 $|E|\geqslant|E'|$,即 $|E'|=|E|$。

证毕

参考文献:

[1] 唐国春,张峰,罗守成,等.现代排序论[M].上海:上海科学普及出版社,2003.

[2] Moore J M. An n-job,one machine sequencing algorithm for minimizing the number of late jobs[J]. Management Science,1968,15:102-109.

[3] 孙叶平,唐万梅,唐国春. Moore-Hodgson 算法最优性的新证明[J].重庆师范大学学报(自然科学版),2007,24(3):4-7.

[4] Sidney J B. An extension of moore 's due date algorithm[C]. Elmaghraby S E. Symposium on the Theory of Scheduling and its Applications[A]. Berlin:Springer,1973,393-398.

[5] Kise H,Ibraki T,Mine,et al. A solvable case of the one-machine scheduling problem with ready and due times[J]. Operations Research,1978,26:121-126.

[6] Lawler E L. Sequencing to minimize the weighted number of tardy jobs[J]. RAIRO,1976,5(S10):27-33.

[7] 唐国春.误工排序问题的研究[J].重庆师范大学学报(自然科学版),2009,26(2):1-6.

[8] 苏永英,蒋宗彩,孙叶平.推广的误工排序问题的最优算法[J].上海第二工业大学学报,2008,25(3):201-206.

Minimizing the Number of Tardy Jobs with a Subset *T* of the Jobs to Be Sure on Time

PENG Hong-jie[1], *SU Yong-ying*[1], *TANG Guo-chun*[1,2]

(1. College of Mathematics and Computer Science, Chongqing Normal University, Chongqing 400047;

2. Institute of Management Engineering, Shanghai Second Polytechnic University, Shanghai 201209, China)

Abstract: The single machine scheduling problem to minimize the number of tardy jobs is one of the most basic scheduling problems in scheduling theory. In 1973, Sidney studied the scheduling problem $1\,|\,T\,|\sum U_j$ to minimize the number of tardy jobs with a subset T of the jobs which must be on time and give a polynomial time algorithm to solve this problem, the optimal could be gotten in time $O(n\log n)$. This algorithm is called Sidney Algorithm by scholars in the following years. In this paper, we have revised Sidney Algorithm as following. Step1, suppose $E_0=T$, $J-E_0=\{j_1,j_2,\cdots,j_m\}$, $j_1<j_2<\cdots<j_m$, $m=n-|T|$, let $k=1$; Step2, if $k=m+1$, stop, then the schedule $(E_m, J-E_m)$ is optimal; Else turn to step3; Step3, arrange the jobs according to the earliest due date (EDD) schedule in E_k, assume $F_k=E_{k-1}\cup\{j_k\}$, if F_k is a tardy subset, let $E_k=E_{k-1}\cup\{j_k\}$; else $E_k=F_k\setminus\{j_r\}$, where the processing time of job j_r satisfying: $p_r=\max\{p_i\,|\,j_i\in F_k\setminus T\}$. Let $k=k+1$, turn to step2. In the end, we use induction to prove that the schedule got from the revised algorithm is also an optimal solution for this problem.

Key words: scheduling; tardy jobs; algorithm; optimality

(责任编辑 黄 颖)