International Journal of Pure and Applied Mathematics
Volume 18 No. 2 2005, 223-229

SINGLE MACHINE SCHEDULING WITH DETERIORATING JOBS UNDER THE GROUP TECHNOLOGY ASSUMPTION

Ai-Xia Guo[1 §], Ji-Bo Wang[2]

[1]Department of Applied Mathematics
Dalian University of Technology
Dalian, 116024, P.R. CHINA
e-mail: guoaixia.ytu@tom.com

[2]Department of Science
Shenyang Institute of Aeronautical Engineering
Shenyang, 110034, P.R. CHINA
e-mail: wjb7575@sina.com

Abstract: This paper discusses the scheduling problem under the condition that the job processing time is a linear deterioration function of its starting time. Under the group technology assumption, the makespan problem is polynomial time solvable.

AMS Subject Classification: 90B35
Key Words: scheduling, single machine, linear deterioration, makespan

1. Introduction

There is a growing interest in the literature to study scheduling problems of *deteriorating jobs*, i.e., jobs whose processing time is an increasing function of their starting time. Such deterioration appears e.g. in scheduling maintenance jobs or cleaning assignments, where any delay in processing a job is penalized and often implies additional time for accomplishing the job. An extensive

Received: November 18, 2004

§Correspondence author

survey of different models and problems concerning start time dependent job processing times can be found in [1], [4]. Browne and Yechiali [2] consider a scheduling problem in which the processing times of the jobs are not constants over time. n jobs have to be processed on a single machine to minimize the makespan. Job i is characterized by: 1) a "basic" processing time p_i, the length of time required to complete the job if it is scheduled first, i.e., at $t = 0$, and 2) a parameter b_i that jointly with p_i, determines the job's (actual) processing time at $t > 0$, b_i can be interpreted as the growth rate of the processing time of job i. Assuming linear deterioration, i.e., the processing time of the job increases linearly with its starting time t, the actual processing time is $p_i(t) = p_i + b_i t$. This problem can be solved optimally by scheduling jobs in an increasing order of p_i/b_i. Mosheiov [5] considers the problem that all jobs are characterized by a common positive basic processing time. Using this basic assumption, Mosheiov proves that the optimal schedule to minimize flowtime is symmetric and has a V-shaped property with respect to the increasing rates. Mosheiov [6] considers the following objective functions: the makespan, the total flow time, the sum of weighted completion times, the total lateness, the maximum lateness and the maxmum tardiness, and the number of tardy jobs. When the values of the normal processing time equal zero, i.e. $p_i(t) = b_i t$, all these problems can be solved polynomially. Wang et al [10] consider the single machine scheduling problems under the condition that the job processing time is a linear deterioration function of its start time. The makespan problems on the simple linear deterioration and on the general linear deterioration are considered respectively. Zhao et al [11] consider a special type of the actual processing time, which is $p_i(t) = p_i(a + bt)$, where a and b are positive constants. They prove that the single machine scheduling problem of minimizing makespan, sum of weighted completion times, maximum lateness and maximum cost is polynomially solvable, respectively, and the two-machine flow shop scheduling to minimize the makespan, for this case the optimal solution can be obtained by Johnson' rule. Chen [3] and Mosheiov [7] consider scheduling deteriorating jobs in a *multi-machine* setting. They assume a linear deterioration and parallel identical machines. Chen considers minimum flow time and Mosheiov studies makespan minimization. Mosheiov [8] considers makespan minimization on the complexity of flow shop, open shop and job shop problems. Mosheiov introduces a polynomial-time algorithm for the two-machine flow shop and proves NP-hardness when an arbitrary number of machines (three and above) is assumed.

Recently, an important class of scheduling problem is characterized by the group technology assumption, i.e. the jobs are classified into groups by the similar production requirements, no machine setups are needed between two consecutively scheduled jobs from the same group, although an independent setup is required between jobs of different groups. In group technology, it is conventional to schedule continuously all jobs from the same group. Group technology that groups similar products into families helps increase the efficiency of operations and decrease the requirement of facilities. Hence, the scheduling in group technology environment results in a new stream of research (Potts and Van Wassenhove [9]).

In this paper we study single machine scheduling problems with deteriorating jobs under the group technology assumption. The objective function is to minimize makespan.

2. Linear Deterioration under the Group Technology Assumption

Assume that there are n jobs $[J_1, J_2, \ldots, J_n]$ which are grouped into f groups, and these n jobs are to be processed on a single-machine. Each of which is available at time $t_0 \geq 0$, jobs are processed one by one in groups on the machine and a setup time is required if the machine switches from one group to another, we assume that the setup times are sequence independent, and the processing of a job may not be interrupted. We let n_i be the number of jobs belonging to group G_i. Thus, $n_1 + n_2 + \ldots + n_f = n$. In addition, $p_{i,j}(t) = p_{i,j}(a + bt)$ denotes the actual processing time of job $J_{i,j}$ in group G_i if its starting time is t, where a and b are positive constants, $J_{i,j}$ denotes the j-th job in group G_i, $i = 1, 2, \ldots, f; j = 1, 2, \ldots, n_i$, and s_i denotes the setup time required to process a job in group G_i following a job in some other group, $C_{i,j}$ denotes the completion time of job $J_{i,j}$ in group G_i. The basic processing time of job $J_{i,j}$ is $p_{i,j}$ and the deterioration function is $d(x) = a + bx$. The objective is to minimize makespan. The problem is denoted as

$$1|S, GT, p_{ij}(a + bt)|C_{\max}, \tag{1}$$

where $C_{\max}$ denotes makespan, S denotes setup times, GT denotes group technology. When $f = 1$, the problem contains only a single group, and the basic single-machine results can be applied. Unless otherwise noted, we assume that $f > 1$.

For the single machine problem without the group technology assumption, Zhao et al. [11] have proved the following lemma.

Lemma 1. *For the problem $1|p_i(t) = p_i(a + bt)|C_{\max}$, the makespan is sequence independent. If the starting time of the first job is t, the makespan is*

$$C_{\max}(t|J_1, J_2, \ldots, J_n) = t + (a + bt)\sum_{i=1}^{n} b^{i-1} C_n^i(p_1, p_2, \ldots, p_n),$$

where p_i denotes basic processing time of job J_j, $C_n^i(p_1, p_2, \ldots, p_n)$ is sum of C_n^i items, each item is the product of i numbers of $p_1, p_2, \ldots, p_n$.

Lemma 2. *The sum $\sum_{i=1}^{n} \mu_i \prod_{k=i}^{n} \gamma_k$ is minimized when calculated over the permutation ordered by increasing order of $\mu_i\gamma_i/(\gamma_i - 1)$.*

Proof. The proof can be obtained by the method of adjacent pairwise interchange. □

From the classical group technology assumption, problem (1) can be solved by the following algorithm.

Algorithm 1. *Step 1.* Jobs in each group scheduled in any order of $p_{i,j}$, $i = 1, 2, \ldots, f; j = 1, 2, \ldots, n_i$.

Step 2. Calculate $A_i = \sum_{k=1}^{n_i} b^{k-1} C_{n_i}^k(p_{i1}, p_{i2}, \ldots, p_{in_i})$, $i = 1, 2, \ldots, f$.

Step 3. Groups scheduled in increasing order of $s_i(1 + bA_i)/(bA_i)$.

Theorem 1. *For the problem $1|S, GT, p_{ij}(a + bt)|C_{\max}$, Algorithm 1 generates an optimal solution.*

Proof. Without loss of generality, we assume for some permutation, group G_i processing in i-th group, job $J_{i,j}$ processing at j-th position in group G_i. Hence, the completion time of all jobs is as follows:

The first group G_1:

$$
\begin{aligned}
C_{1,1} &= t_0 + s_1 + p_{11}(a + b(t_0 + s_1)), \\
C_{1,2} &= t_0 + s_1 + p_{11}(a + b(t_0 + s_1)) \\
&\quad + p_{12}[a + b(t_0 + s_1 + p_{11}(a + b(t_0 + s_1)))] \\
&= t_0 + s_1 + (a + b(t_0 + s_1))(\sum_{j=1}^{2} p_{1j} + bp_{11}p_{12}) \\
&= t_0 + s_1 + (a + b(t_0 + s_1)) \sum_{k=1}^{2} b^{k-1} C_2^k(p_{11}, p_{12}), \\
&\quad \cdots\cdots \\
C_{1,n_1} &= t_0 + s_1 + (a + b(t_0 + s_1)) \sum_{k=1}^{n_1} b^{k-1} C_{n_1}^k(p_{11}, p_{12}, \ldots, p_{1n_1}) \\
&= t_0 + s_1 + (a + b(t_0 + s_1))A_1 \\
&= (t_0 + s_1)(1 + bA_1) + aA_1.
\end{aligned}
$$

The second group G_2:

$$
\begin{aligned}
C_{2,1} &= [(t_0 + s_1)(1 + bA_1) + aA_1 + s_2] \\
&\quad + p_{21}[a + b((t_0 + s_1)(1 + bA_1) + aA_1 + s_2)], \\
&\quad \cdots\cdots \\
C_{2,n_2} &= [(t_0 + s_1)(1 + bA_1) + aA_1 + s_2] \\
&\quad + [a + b((t_0 + s_1)(1 + bA_1) + aA_1 + s_2)]A_2 \\
&= (t_0 + s_1)(1 + bA_1)(1 + bA_2) + s_2(1 + bA_2) \\
&\quad + a(A_1 + A_2 + bA_1A_2) \\
&= (t_0 + s_1) \prod_{i=1}^{2}(1 + bA_i) + s_2(1 + bA_2) + a \sum_{k=1}^{2} b^{k-1} C_2^k(A_1, A_2).
\end{aligned}
$$

The third group G_3:

$$
\begin{aligned}
C_{3,n_3} &= (t_0 + s_1) \prod_{i=1}^{3}(1 + bA_i) + s_2 \prod_{i=2}^{3}(1 + bA_i) + s_3(1 + bA_3) \\
&\quad + a \sum_{k=1}^{3} b^{k-1} C_3^k(A_1, A_2, A_3). \\
&\quad \cdots\cdots
\end{aligned}
$$

The last group G_f:

$$C_{f,n_f} = (t_0 + s_1)\prod_{i=1}^{f}(1+bA_i) + s_2\prod_{i=2}^{f}(1+bA_i) + \ldots + s_f(1+bA_f) + a\sum_{k=1}^{f} b^{k-1}C_f^k(A_1, A_2, \ldots, A_f)$$

$$= t_0\prod_{i=1}^{f}(1+bA_i) + \sum_{i=1}^{f} s_i \prod_{k=i}^{f}(1+bA_k) + a\sum_{k=1}^{f} b^{k-1}C_f^k(A_1, A_2, \ldots, A_f).$$

Obviously, $A_i = \sum_{k=1}^{n_i} b^{k-1}C_{n_i}^k(p_{i1}, p_{i2}, \ldots, p_{in_i})$ is independent of permutation of jobs in group G_i (Lemma 1). The first item $t_0\prod_{i=1}^{f}(1+bA_i)$ is independent of permutation of groups and jobs in each group. The second item $\sum_{i=1}^{f} s_i \prod_{k=i}^{f}(1+bA_k)$ is minimized by the increasing order of $s_i(1+bA_i)/(bA_i)$ (Lemma 2). The last term $a\sum_{k=1}^{f} b^{k-1}C_f^k(A_1, A_2, \ldots, A_f)$ is independent of permutation of groups and jobs in each group (Lemma 1). Therefore, Algorithm 1 is an optimal permutation for the problem $1|S, GT, p_{ij}(a+bt)|C_{\max}$. $\square$

Obviously, the total time for Algorithm 1 is $O(n\log n)$.

3. Conclusions

In this paper, we investigated the problem of scheduling jobs with start time dependent processing times (deterioration) for the makespan minimization. Under the group technology assumption, this problem is proved to be polynomial time solvable.

References

[1] B. Alidaee, N.K. Womer, Scheduling with time dependent processing processing times: review and extensions, *Journal of the Operational Research Society,* **50** (1999), 711-720.

[2] S. Browne, U. Yechiali, Scheduling deteriorating jobs on a single processor, *Operations Research,* **38** (1990), 495-498.

[3] Z-L. Chen, Parallel machine scheduling with time dependent processing times, *Discrete Applied Mathmatics,* **70** (1996), 81-94.

[4] T.C.E. Cheng, Q. Ding, B.M.T. Lin, A concise survey of scheduling with time-dependent processing times, *European Journal of Operational Research,* **152** (2004), 1-13.

[5] G. Mosheiov, V-shaped policies for scheduling deteriorating jobs, *Operations Research,* **39** (1991), 979-991.

[6] G. Mosheiov, Scheduling jobs under simple linear deterioration, *Computers and Operations Research,* **21** (1994), 653-659.

[7] G. Mosheiov, Multi-machine scheduling with linear deterioration, *INFOR,* **36** (1998), 205-214.

[8] G. Mosheiov, Complexity analysis of job-shop scheduling with deteriorating jobs, *Discrete Applied Mathmatics*, **117** (2002) 195-209.

[9] C.N. Potts, L.N. Van Wassenhove, Integrating scheduling with batching and lot-sizing: A review of algorithms and complexity, *Journal of the Operational Research Society*, **43** (1992), 395-406.

[10] J.-B. Wang, M.-Z. Wang, Z.-Q. Xia, Single-machine scheduling to minimize makespan under linear deterioration, *International Journal of Pure and Applied Mathematics,* **13** (2004), 529-539.

[11] C.L. Zhao, Q.L. Zhang, H.Y. Tang, Schedling problems under linear deterioration, *Acta Automatica Sinica,* **29** (2003), 531-535.