Applied Mathematical Modelling 34 (2010) 837–841

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Short communication

Single machine scheduling with general job-dependent aging effect and maintenance activities to minimize makespan ☆

Chuan-li Zhao *, Heng-yong Tang

School of Mathematics and Systems Science, Shenyang Normal University, Shenyang, Liaoning 110034, People's Republic of China

ARTICLE INFO

Article history:
Received 6 March 2009
Received in revised form 7 June 2009
Accepted 22 July 2009
Available online 28 July 2009

Keywords:
Scheduling
Aging effect
Single machine
Makespan

ABSTRACT

This paper considers single machine scheduling with an aging effect in which the processing time of a job depends on its position in a sequence. It is assumed that aging ratios are job-dependent and machine can be maintained some times in a schedule. After a maintenance activity, machine will be restored to its initial condition. The processing of jobs and the maintenance activities of machine are scheduled simultaneously. The objective is to schedule the jobs and the maintenance activities, so as to minimize the makespan. We provide a polynomial time algorithm to solve the problem.

1. Introduction

In the classical scheduling problems it is assumed that the processing times of jobs are constant. However, there are many situations where the processing times of the job may be dependent on their positions in the sequence [1]. This phenomenon is called *learning effect* or *aging effect*. In a learning environment, the later a given job is scheduled in the sequence, the shorter its processing time; while in an aging environment, the later a given job is scheduled in the sequence, the longer its processing time.

The learning effect on scheduling problem was first introduced by Biskup [2]. Since then, scheduling problem with learning effect have received increasing attention [3–6]. An updated survey of the results on scheduling problems with learning effect was provided by Biskup [7]. Mosheiov [4] first studied scheduling problem with an aging effect. Gordon et al. [8] considered various single machine scheduling problems in which the processing time of a job depends either on its position in a processing sequence or on its start time. They identified several situations in which an objective function for a scheduling problem under a certain job processing time deterioration or learning scenario is priority-generating. Gordon and Strusevich [9] given polynomial time dynamic programming algorithms for solving single machine scheduling and due date assignment problems, provided that the processing times of the jobs are positionally dependent. Mosheiov [6] considered single machine total completion time minimization problem. He proved that the optimal schedule is *V-shaped* with respect to job processing times. Zhao and Tang [10] studied some single machine scheduling problems with an aging effect. They showed that the scheduling problems with an aging effect have some interesting features and solved two resource constrained problems.

On the other hand, in most of manufacturing situations, machines may need to be maintained periodically. Therefore, a more realistic scheduling model should take into account associated machine maintenance. There are several papers which

☆ This work was supported by National Natural Science Foundation of China (NNSFC) 10471096 and Science Foundation of Liaoning Province Education Department 2008z192.

* Corresponding author.
E-mail addresses: zhaochuanli@synu.edu.cn, czhao58@yahoo.com.cn (C.-l. Zhao).

0307-904X/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.apm.2009.07.002

consider scheduling problem with maintenance activities. Graves and Lee [11] discussed problems where the machine maintenance activity can happen at most one or two times during the planning horizon. Lee and Chen [12] considered parallel machine scheduling where each machine must be maintained once during the planning horizon. Qi et al. [13] investigated single machine scheduling with multiple maintenance activities which are to be scheduled jointly with jobs. Lee and Leon [14] and Zhao et al. [15] studied scheduling with a rate-modifying activity, it can be considered as a special type of scheduling with maintenance activity.

The concept of aging effect and maintenance activity has been extensively studied independently in the literature. However, to the best of our knowledge, apart from the recent paper of Kuo and Yang [16], there are no research results on scheduling models considering the aging effects and maintenance activity at the same time, although the phenomena can be found in real-life situation [16]. Kuo and Yang studied a single machine scheduling problem with the cyclic process of an aging effect. For two different aging effect models, they provided the polynomial time algorithm to solve the single machine makespan minimization problem.

In this paper, we extend the model of Kuo and Yang [16] to the case of job-dependent aging effect. We consider single machine makespan minimization problem. By converting the problem to an weighted-bipartite matching problem, we show that the problem remains polynomially solvable.

2. Problem statement

The problem under consideration can be described as follows.

There are n jobs $J_1, J_2, \ldots, J_n$ to be processed on a single machine. All jobs are non-preemptive and available for processing at time zero. Let p_j denote the normal processing time of job J_j. Assume machine can be maintained some times in a schedule and the maintaining time is t. During the maintenance activity the machine is turned off and production is stopped. After a maintenance, machine will be restored to its initial condition. Therefore, a schedule π contains a sequence of jobs and the maintenance activities inserted in job sequence. In a schedule, jobs processed continuously form a group, denoted as G. Thus a schedule π can be denoted as $\pi = [G_1, M, G_2, M, \ldots, M, G_{k+1}]$, where M is the maintenance activity and k is the number of maintenance activities (k is a decision variable in our problem). The actual processing time of J_j when scheduled in position r in a group is given by:

$$p_j^r = p_j r^{a_j}, \tag{1}$$

where $a_j \geqslant 0$ is the aging ratio of job J_j.

The objective is to minimize the makespan. That is, we have to determine the frequency (k) of the maintenance activity (M) and the sequence of all jobs to minimize the makespan. Following the notation used by Kuo and Yang [16], we denote the problem by $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$.

For a given schedule π, if $k = k_0$, jobs are divided into $k_0 + 1$ groups $G_1, G_2, \ldots, G_{k_0}, G_{k_0+1}$. Suppose that $\pi = [G_1, M, G_2, M, \ldots, M, G_{k_0+1}]$, and $G_i = [J_{i1}, J_{i2}, \ldots, J_{i,n_i-1}, J_{i,n_i}]$ $(i = 1, \ldots, k_0 + 1)$. Let job $J_{ir} (r = 1, \ldots, n_i)$ be the rth job in the group G_i, the normal processing time and aging ratio of J_{ir} is p_{ir} and a_{ir}, respectively. We denote the completion time of job $J_{ir} (i = 1, \ldots, k_0 + 1, r = 1, \ldots, n_i,)$ by C_{ir}. Then

$$C_{11} = p_{11},$$

$$C_{12} = \sum_{r=1}^{2} p_{1r} r^{a_{1r}},$$

$$\ldots, \ldots,$$

$$C_{1j} = \sum_{r=1}^{j} p_{1r} r^{a_{1r}},$$

$$\ldots, \ldots,$$

$$C_{1\,n_1} = \sum_{r=1}^{n_1} p_{1r} r^{a_{1r}},$$

$$\ldots, \ldots,$$

$$C_{2j} = \sum_{r=1}^{n_1} p_{1r} r^{a_{1r}} + t + \sum_{r=1}^{j} p_{2r} r^{a_{2r}}, \quad (j = 1, \ldots, n_2),$$

$$\ldots, \ldots,$$

$$C_{ij} = \sum_{l=1}^{i-1} \sum_{r=1}^{n_l} p_{lr} r^{a_{lr}} + (i-1)t + \sum_{r=1}^{j} p_{ir} r^{a_{ir}}, \quad (j = 1, \ldots, n_i);$$

$$\ldots, \ldots,$$

$$C_{k_0+1,j} = \sum_{l=1}^{k_0}\sum_{r=1}^{n_l} p_{lr}r^{a_{lr}} + k_0t + \sum_{r=1}^{j} p_{k_0+1,r}r^{a_{k_0+1,r}}, \quad (j = 1, \ldots, n_{k_0+1});$$

$\ldots, \ldots,$

$$C_{k_0+1,n_{k_0+1}} = \sum_{l=1}^{k_0+1}\sum_{r=1}^{n_l} p_{lr}r^{a_{lr}} + k_0t.$$

3. Main results

The makespan of the $1||C_{\max}$ problem is sequence-independent. This property does not hold for the $1|p_j^r = p_jr^{a_j}, M = k|C_{\max}$ problem.

If there is no maintenance activity (i.e., $k = 0$), similar to the single machine makespan minimization problem with general learning effect [5], the $1|p_j^r = p_jr^{a_j}, M = k|C_{\max}$ problem can be formulated as an assignment problem.

Lemma 1. *If there is no maintenance activity, the $1|p_j^r = p_jr^{a_j}, M = 0|C_{\max}$ problem can be solved in $O(n^3)$ time.*

For the special case $a_j = a$, Kou and Yang [16] presented a *group balance principle*. In the following, we will show that the solution of *group balance principle* remains valid for the $1|p_j^r = p_jr^{a_j}, M = k|C_{\max}$ problem.

For a given schedule, if the machine is maintained k times, then the jobs are divided into $(k+1)$ groups. The group balance principle is to make the number of jobs to be as equal as possible in each group.

Group balance principle: Suppose that the machine is maintained k times in a schedule, the jobs are divided into $(k+1)$ groups: $G_1, \ldots, G_{k+1}$, G_i has n_i jobs $(i = 1, \ldots, k+1)$, then $\lfloor n/(k+1)\rfloor \leqslant n_i \leqslant \lfloor n/(k+1)\rfloor + 1$.

Lemma 2. *For the $1|p_j^r = p_jr^{a_j}, M = k|C_{\max}$ problem, if k fixed, then there exists an optimal solution such that the numbers of jobs in groups satisfy Group balance principle.*

Proof. Consider an optimal schedule π, assume that the machine is maintained k times, then there are $(k+1)$ groups of jobs in π. Let $G_1, G_2, \ldots, G_k, G_{k+1}$ denote the groups of jobs. Suppose that the numbers of jobs in groups does not satisfy Group balance principle, then there must be at least two groups, say G_i and G_j, where G_i has n_i jobs and G_j has n_j jobs, such that $n_i > n_j$ and $n_i - n_j > 1$. Let $\pi = [\pi_1, G_i, \pi_2, G_j, \pi_3]$. π_1, π_2 and π_3 are partial schedules of π. Assume $G_i = [J_{i1}, J_{i2}, \ldots, J_{i,n_i-1}, J_{i,n_i}]$ and $G_j = [J_{j1}, J_{j2}, \ldots, J_{j,n_j}]$. For convenience, let job J_{i,n_i} be job J_j. By moving job $J_{i,n_i}(J_j)$ from group G_i to group G_j last position, we get a new schedule $\tilde{\pi} = [\pi_1, \tilde{G}_i, \pi_2, \tilde{G}_j, \pi_3]$, where $\tilde{G}_i = [J_{i1}, J_{i2}, \ldots, J_{i,n_i-1}]$ and $\tilde{G}_j = [J_{j1}, J_{j2}, \ldots, J_{j,n_j}, J_{i,n_j+1}(J_j)]$.

The actual processing time of job J_j is $p_jn_i^{a_j}$ in π, while it is $p_j(n_j+1)^{a_j}$ in $\tilde{\pi}$. Since $n_i - n_j > 1$ and $a_i \geqslant 0$, then $p_jn_i^{a_j} > p_j(n_j+1)^{a_j}$.

Hence, the makespan of $\tilde{\pi}$ is strictly smaller than that of π. This contradicts the optimality of π and proves the lemma.

Let $\pi = [G_1, M, G_2, M, \ldots, M, G_{k+1}]$, the jobs sequence in group G_i is $G_i = [J_{i1}, J_{i2}, \ldots, J_{i,n_i-1}, J_{i,n_i}]$, then the makespan of π is:

$$C_{\max} = \sum_{j=1}^{n_1} p_{1j}j^{a_{1j}} + \sum_{j=1}^{n_2} p_{2j}j^{a_{2j}} + \ldots + \sum_{j=1}^{n_{k+1}} p_{k+1,j}j^{a_{k+1,j}} + kt = \sum_{i=1}^{k+1}\sum_{j=1}^{n_i} p_{ij}j^{a_{ij}} + kt.$$

We need to determine the jobs sequence and the value of k to minimize $C_{\max}$. In order to solve the case with $k = 1, \ldots, n-1$, we define

$$A(k) = \sum_{i=1}^{k+1}\sum_{j=1}^{n_i} p_{ij}j^{a_{ij}}.$$

Since $C_{\max} = A(k) + kt$, for a given k, minimize $C_{\max}$ is equivalent to minimizing $A(k)$.

From Lemma 2, when the machine is maintained k times, the positions of jobs can be divided into $\lfloor n/(k+1)\rfloor + 1$ kinds: let $n = (k+1)m + l$ (where $l < k+1$), then there are $(k+1)$ positions 1, $(k+1)$ positions 2, …, $(k+1)$ positions m and l positions $m+1$. The jobs are divided into $(k+1)$ groups: $G_1, \ldots, G_{k+1}$, G_i has n_i jobs $(i = 1, \ldots, k+1)$, where $n_i = m+1$ if $1 \leqslant i \leqslant l$ and $n_i = m$ if $l+1 \leqslant i \leqslant k+1$. If job J_j $(j = 1, \ldots, n)$ is scheduled in position r $(r = 1, \ldots, n_i)$ in group G_i $(i = 1, \ldots, k+1)$, then its contribution for makespan is $p_jr^{a_j}$ (independent of G_i).

Based on above results, for a given k, the problem can be formulated as a weighted-bipartite matching problem (or assignment problem). We define a bipartite graph $G(V, E)$, $V = V_1 \cup V_2$, $V_1 \cap V_2 = \emptyset$ and $|V_1| = |V_2| = n$. The vertices in set V_1 represent jobs, and the vertices in set V_2 represent positions in the sequence. The vertices in set V_2 can be divided into $m+1$ kinds: $(k+1)$ vertices represent position $r (r = 1, \ldots, m)$, l vertices represent position $m+1$. The edges in set E connect each vertex in V_1 with all vertices in V_2. Associated with edge (j, r), $j \in V_1$ and $r \in V_2$, there is a weight $w_{jr} = p_jr^{a_j}$. The matching with minimum weight specifies the optimal value of $A(k)$. Equivalent, the problem can also be formulated as following assignment problem:

$$\begin{aligned}
\min \quad & \sum_{j=1}^{n}\sum_{i=1}^{k+1}\sum_{r=1}^{n_i} w_{jir}x_{jir},\\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{jir} = 1, \quad i=1,2,\ldots,k+1, \quad r=1,2,\ldots,n_i,\\
& \sum_{i=1}^{k+1}\sum_{r=1}^{n_i} x_{jir} = 1, \quad j=1,2,\ldots,n,\\
& x_{jir} = 0 \text{ or } 1, \quad j=1,2,\ldots,n, \quad i=1,2,\ldots,k+1, \quad r=1,2,\ldots,n_i,
\end{aligned}$$

where $w_{jir} = p_j r^{a_j}$.

From a solution of the assignment problem, we can get an optimal schedule: $x_{jir} = 1$ means that job $J_j (j = 1,\ldots,n)$ is scheduled in position $r (r = 1,\ldots,n_i)$ in group $G_i (i = 1,\ldots,k+1)$.

We now to give a polynomial time algorithm for the problem $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$.

Algorithm 1

Step 1. Let $\mathscr{K} = \{0, 1, \ldots, n-1\}$.
Step 2. For each $k \in \mathscr{K}$, create a weighted-bipartite matching problem with weight $w_{jr} = p_j r^{a_j}$. Solve the weighted-bipartite matching problem and let the corresponding total cost be $A(k)$. Set $C_{\max}(k) = A(k) + kt$.
Step 3. The solution is the best one: $C^*_{\max}(\tilde{k}) = \min\{C_{\max}(k), (k = 0, 1, \ldots, n-1)\}$.

Theorem 1. *Algorithm 1 will find an optimal solution for the problem* $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ *in* $O(n^4)$ *time.*

Proof. As discussed above, we can convert the problem $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ to a weighted-bipartite matching problem and find an optimal solution by Algorithm 1. Step 2 can be solved in $O(n^3)$ time for each k. Step 1 is executed n times. Step 3 can be solved in $O(n\log n)$ time. Consequently, the overall time requirement of Algorithm 1 is $O(n^4)$. □

4. Special case

For the case where $a_j = a$ for all J_j, Kou and Yang [16] provided an algorithm with time complexity $O(n\log n)$ to solve the problem. Now we consider a special case, where jobs satisfy the *agreeable condition*, i.e., $p_i < p_j$ implies that $a_i \leqslant a_j$ for all J_i and J_j. Note that the agreeable condition assumption includes the special case where $a_j = a$.

Theorem 2. *For the* $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ *problem, if* $k = 0$ *and jobs satisfy the agreeable condition, an optimal schedule can be obtained by sequencing jobs in non-increasing order of their normal processing times* p_j *(LPT rule).*

Proof. Consider an optimal schedule π, suppose which is not *LPT* schedule. In schedule π, there must be at least two adjacent jobs, say job J_i followed by job J_j, such that $p_i < p_j$ and $a_i \leqslant a_j$. Assume job J_i is scheduled in the position r and its starting time is t_0. Perform a adjacent pairwise interchange on job J_i and J_j. Call the new schedule $\tilde{\pi}$, Let $C_i(\tilde{C}_i)$ denote the completion time of job J_i in schedule $\pi(\tilde{\pi})$. Then

$$\begin{aligned}
C_i &= t_0 + p_i r^{a_i},\\
C_j &= t_0 + p_i r^{a_i} + p_j (r+1)^{a_j},\\
\tilde{C}_j &= t_0 + p_j r^{a_j},\\
\tilde{C}_i &= t_0 + p_j r^{a_j} + p_i (r+1)^{a_i}.
\end{aligned}$$

Hence

$$\tilde{C}_i - C_j = p_i[(r+1)^{a_i} - r^{a_i}] - p_j[(r+1)^{a_j} - r^{a_j}].$$

Let $g(x) = (r+1)^x - r^x$. Taking the first derivative of $g(x)$, we get $g'(x) = (r+1)^x \log(r+1) - r^x \log r$.

If $x > 0$, then $g'(x) > 0$ for $r = 1,\ldots,n$. Hence $g(x)$ is an increasing function of x, and $g(a_i) \leqslant g(a_j)$ for $a_i \leqslant a_j$, i.e., $[(r+1)^{a_i} - r^{a_i}] \leqslant [(r+1)^{a_j} - r^{a_j}]$.

Since $p_i < p_j$ and $[(r+1)^{a_i} - r^{a_i}] \leqslant [(r+1)^{a_j} - r^{a_j}]$, then $\tilde{C}_i - C_j < 0$. The completion times of the jobs processed before jobs J_i and J_j are not affected by the interchange, the completion times of the jobs processed after jobs J_i and J_j are become earlier. Hence the makespan of $\tilde{\pi}$ is strictly less than that of π. This contradicts the optimality of π and proves the theorem.

Similar to the results of Kou and Yang [9], we have the following statements. □

Corollary 1. *For the* $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ *problem, suppose jobs satisfy the agreeable condition, let* $C^*_{\max}$ *denote the optimal solution of the problem* $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ *problem in which* $k = 0$*, If* $C^*_{\max} - \sum_{j=1}^{n} p_j \leqslant t$*, then there exists an optimal schedule in which jobs are sequenced in non-increasing order of their normal processing times.*

Theorem 3. *For the $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ problem, if jobs satisfy the agreeable condition, $k = k_0$ and jobs are divided into groups $G_1, G_2, \ldots, G_{k_0}, G_{k_0+1}$ according to the group balance principle, then there exists an optimal schedule in which jobs are sequenced in non-increasing order of their normal processing times and then arranged one by one to each group in turn. That is, the jobs of the sequence are assigned to a schedule from the first position of the first group to the first position of the last group, and then from the second position of the first group to the second position of the last group, ..., and so on.*

Based on Theorems 2 and 3, if jobs satisfy the *agreeable condition*, an optimal schedule for the $1|p_j^r = p_j r^{a_j}, M = k|C_{\max}$ problem can be found in $O(nlogn)$ time.

5. Conclusions

We studied single machine scheduling with job-dependent aging effect. In our model, machine can be maintained some times in a schedule. After a maintenance activity, machine will be restored to its initial condition. The objective of the problem is to make a decision on whether or not and when to schedule the maintenance and the sequence of jobs to minimize the makespan. We show that the problem remains polynomially solvable, although the computational effort required is much larger than that of job-independent version of the problem. Further research includes the investigation of other objectives or multi-machine problems with an aging effect.

Acknowledgement

The authors wish to thank two anonymous referees for their constructive comments and suggestions that improved an early version of this paper.

References

[1] A. Bachman, A. Janiak, Scheduling jobs with position-dependent processing times, J. Oper. Res. Soc. 55 (2004) 257–264.
[2] D. Biskup, Single-machine scheduling with learning considerations, Eur. J. Oper. Res. 115 (1999) 173–178.
[3] G. Mosheiov, Scheduling problems with a learning effect, Eur. J. Oper. Res. 132 (2001) 687–693.
[4] G. Mosheiov, Parallel machine scheduling with a learning effect, J. Oper. Res. Soc. 52 (2001) 1165–1169.
[5] G. Mosheiov, B. Sidney, Scheduling with general job-dependent learning curves, Eur. J. Oper. Res. 147 (2003) 665–670.
[6] G. Mosheiov, A note on scheduling deteriorating jobs, Math. Comput. Modell. 41 (2005) 883–886.
[7] D. Biskup, A state-of-the-art review on scheduling with learning effects, Eur. J. Oper. Res. 188 (2008) 315–329.
[8] V.S. Gordon, C.N. Potts, V.A. Strusevich, J.D. Whitehead, Single machine scheduling models with deterioration and learning: handling precedence constraints via priority generation, J. Schedul. 11 (2008) 357–370.
[9] V.S. Gordon, V.A. Strusevich, Single machine scheduling and due date assignment with positionally dependent processing times, Eur. J. Oper. Res. 198 (2009) 57–62.
[10] Chuan-li Zhao, Heng-yong Tang, Single-machine scheduling problems with an aging effect, J. Appl. Math. Comput. 1–2 (2007) 305–314.
[11] G.H. Graves, C.-Y. Lee, Scheduling maintenance and semiresumable jobs on a single machine, Naval Res. Logist. 46 (1999) 845–863.
[12] C.-Y. Lee, Z.-L. Chen, Scheduling jobs and maintenance activities on parallel machine, Naval Res. Logist. 47 (2000) 145–165.
[13] X. Qi, T. Chen, F. Tu, Scheduling the maintenance on a single machines, J. Oper. Res. Soc. 50 (1999) 1071–1078.
[14] C.-Y. Lee, V.J. Leon, Machine scheduling with a rate-modifying activity, Eur. J. Oper. Res. 128 (2001) 119–128.
[15] Chuan-li Zhao, Heng-yong Tang, Cong-dian Cheng, Two-parallel machines scheduling with rate-modifying activities to minimize total completion time, Eur. J. Oper. Res. 198 (2009) 354–357.
[16] W.-H. Kuo, D.-L. Yang, Minimizing the makespan in a single-machine scheduling problem with the cyclic process of an aging effect, J. Oper. Res. Soc. 59 (2007) 416–420.