Journal of Nan Kai, Vol. 10, No. 1, pp.35-39 (2013)

A Study of Unrelated Parallel-Machine Scheduling with Deteriorating Maintenance Activities to Minimize the Total Completion Time

Suh-Jenq Yang, Jia-Yuarn Guo , Hsin-Tao Lee

Department of Industrial Management, Nan Kai University of Technology

通訊作者：楊肅正
聯絡地址：南投縣草屯鎮中正路 568 號
電子郵件：sjyang@nkut.edu.tw
投稿日期：2012 年 11 月
接受日期：2013 年 5 月

Abstract

Production scheduling with maintenance planning to improve the production efficiency of machines has received considerable attention over the years. In two recent papers, Wang, Wang, and Liu (2011) and Wang, Huang, Ji, and Feng (2012) studied the scheduling problem in which the n jobs must be scheduled on *m* parallel unrelated machines with deteriorating maintenance activities to minimize the total completion time. Wang et al. (2011) showed that the problem could be solved in $O(n^{2m+3})$ time and Wang et al. (2012) showed that the problem could be solved in $O(n^{m+3})$ if $0 < \alpha_{ij} \le 1$, where $\alpha_{ij} > 0$ was the modifying rate of job *j*, if it was assigned to machine *i* after the maintenance activity was performed. In this note, we proposed a more efficient algorithm and showed that both problems studied by Wang et al. (2011) and Wang et al. (2012) could be solved in $O(n^{m+3})$ time even though for the case of $\alpha_{ij} > 1$.

Keywords: scheduling, maintenance activity, unrelated parallel-machine, total completion time

1. Introduction

Maintenance is important in production as it helps sustain machine efficiency or product quality. Scheduling with maintenance activity has been widely studied in the past decade. For research results on scheduling models with maintenance activities and different machine environments, a reader can refer to Lee and Leon (2001), Wu and Lee (2003), Ji, He, and Cheng (2006), Lee and Wu (2008), Kuo and Yang (2008), Low, Hsu, and Su (2008), Hsu, Low, and Su (2010), Mosheiov and Sidney (2010), Zhao, Tang, and Cheng (2009), Lodree and Geiger (2010), Yang and Yang (2010a; 2010b; 2010c), Zhao and Tang (2010), and Yang (2011; 2012).

In a recent paper, Wang, Wang, and Liu (2011) studied scheduling problems with a deteriorating maintenance activity on an identical parallel-machine setting. They assumed that at most one maintenance activity is allowed on each machine during the planning horizon to change its production rate. They further assumed that the maintenance activity can be performed immediately after completing the processing of any job, and the duration of the maintenance activity on a machine is dependent on its running time (Kubzin & Strusevich, 2006). If job J_j is processed before the maintenance activity, its actual processing time is p_j (i.e., its normal processing time), and if job J_j is processed after the maintenance activity, its actual processing time is $\alpha_j p_j$, where $\alpha_j > 0$ is the modifying rate of job J_j. The objective was to minimize the total completion time. They first showed that the two identical parallel-machine scheduling problem can be solved in $O(n^6)$ time. They then extended the problem to the case with *m* unrelated parallel-machine and show that it can be solved in $O(n^{2m+3})$ time. Later, Wang, Huang, Ji, and Feng (2012) investigated the same problem proposed by Wang et al. (2011) on an unrelated-parallel machine setting. They showed that the total completion time minimization problem can be solved in $O(n^{m+3})$ if $0 < \alpha_{ij} \le 1$, where $\alpha_{ij} > 0$ is the modifying rate of job J_j, if it is assigned to machine M_i after the maintenance activity is performed.

In this note, we extend the problem studied by Wang et al. (2011) and Wang et al. (2012) and show that it can be solved in $O(n^{m+3})$ time even though for the case of $\alpha_{ij} > 1$.

2. Methods

We follow the notation and terminology used by Wang et al. (2012) throughout the paper and will introduce additional notation when needed. The problem under study is described as follows: There are *n* independent jobs $J = \{J_1, J_2, ..., J_n\}$ to be processed on *m* parallel unrelated machines $M = \{M_1, M_2, ..., M_m\}$. Preemption of jobs is not allowed and each machine can process only one job at a time and cannot stand idle until the last job assigned to it has been finished. We assume that at most one maintenance activity is allowed on each machine during the planning horizon to change its production rate.

We denote that the maintenance activity of machine M_i is in position l_i ($0 \le l_i \le n$, $i = 1, 2, ..., m$) if it is scheduled between the completion time of the job that is scheduled in the position $(l_i + 1)$th to the last job and the starting time of the job that is scheduled in the position l_ith to the last job. Let p_{ij} be the normal processing time of job J_j if it is assigned to machine M_i. Then the actual processing time of job J_j if it is scheduled in the position *k*th to the last job on machine M_i is given by:

$$p_{ijk} = \begin{cases} p_{ij}, & k > l_i, \\ \alpha_{ij} p_{ij}, & k \le l_i. \end{cases} \tag{1}$$

for $i = 1, 2, ..., m$, $j = 1, 2, ..., n$, and $0 \le k \le n$, where $\alpha_{ij} > 0$ is the modifying rate of job J_j, if it is assigned to machine M_i after the maintenance activity is performed.

In addition, the duration of the maintenance activity on a machine is dependent on its running time (Kubzin & Strusevich, 2006) and is defined by $T_i = t_i + \delta_i t$ on machine M_i for $i = 1, 2, ..., m$, where $t_i > 0$ is the basic maintenance time, $\delta_i > 0$ is the maintenance factor, and t is the starting time of the maintenance activity.

In what follows we present a 7 jobs instance on the two unrelated parallel-machine scheduling problem under study. Figure 1 illustrates a feasible schedule in which jobs J_1, J_2, J_3 and J_4 are processed on machine M_1, jobs J_5, J_6 and J_7 are processed on machine M_2, and the rate-modifying activity positions are $l_1 = 2$ and $l_2 = 1$.

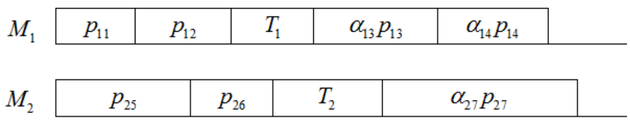

Figure 1 A feasible schedule with $m = 2$, $n = 7$, $l_1 = 2$, and $l_2 = 1$.

Following Wang et al. (2012), we denote our scheduling problem as $Rm \mid T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} \mid \sum C_j$, where rm indicates the rate-modifying activity and $\sum C_j$ denotes the total completion time.

3. Results

In this section, we provide a more efficient algorithm to show that both problems studied by Wang et al. (2011) and Wang et al. (2012) can be solved in $O(n^{m+3})$ time no matter what $0 < \alpha_{ij} \le 1$ or $\alpha_{ij} > 1$. First, the following two lemmas are useful for finding the optimal solution of the problem.

Lemma 1 (Mott, Kandel, & Baker, 1986) The number of non-negative integer solutions to $x_1 + x_2 + ... + x_m = n$ is $C(n+m-1, n) = C(n+m-1, m-1) = \frac{(n+m-1)!}{(m-1)!n!}$.

Lemma 2 (Wang et al., 2012) The number $C(n+m, m)$ is bounded from above by $\frac{(2n)^m}{m!}$.

Next, let $(l_1, l_2, ..., l_m)$ and $C_j(l_1, l_2, ..., l_m)$ be the rate-modifying activity position vector and the completion time of J_j that is processed on one of the m unrelated parallel machines based on $(l_1, l_2, ..., l_m)$. We denote by the subscript $i[r]$ a job scheduled in the rth position to the last job on machine M_i, for $i = 1, 2, ..., m$ and $r = 1, 2, ..., n_i$, where n_i is the number of jobs assigned to be processed on machine M_i and $n = \sum_{i=1}^{m} n_i$. Then, we obtain that

$$\sum C_j(l_1, l_2, ..., l_m) = \sum_{i=1}^{m}\left(\sum_{r=1}^{l_i} r\alpha_{i[r]} p_{i[r]} + \sum_{r=l_i+1}^{n_i}(r+\delta_i l_i) p_{i[r]}\right) + \sum_{i=1}^{m} l_i t_i . \quad (2)$$

Let $w_{i[r]}$ be the weight of job J_j if it is scheduled in the position rth to the last job processed on machine M_i. Then, (2) can be rewritten as follows:

$$\sum C_j(l_1, l_2, ..., l_m) = \sum_{i=1}^{m}\sum_{r=1}^{n_i} w_{i[r]} + \sum_{i=1}^{m} l_i t_i , \quad (3)$$

where

$$w_{i[r]} = \begin{cases} r\alpha_{i[r]} p_{i[r]}, & i = 1, 2, ..., m,\ r \le l_i, \\ (r+\delta_i l_i) p_{i[r]}, & i = 1, 2, ..., m,\ r > l_i. \end{cases} \quad (4)$$

In addition, we define $y_{ijs} = 1$ if J_j is in the position sth to the last job processed on M_i and $y_{ijs} = 0$ otherwise. Then, the $Rm | T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} | \sum C_j$ problem can be formulated as follows:

$$\text{Minimize } \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{s=1}^{n} w_{ijs} y_{ijs} + \sum_{i=1}^{m} l_i t_i . \quad (5)$$

$$\text{subject to } \sum_{j=1}^{n} y_{ijs} = 1,\ i = 1, 2, ..., m,\ s = 1, 2, ..., l_i \quad (6)$$

$$\sum_{j=1}^{n} y_{ijs} \le 1,\ i = 1, 2, ..., m,\ s = l_i + 1, l_i + 2, ..., n \quad (7)$$

$$\sum_{i=1}^{m}\sum_{s=1}^{n} y_{ijs} = 1,\ j = 1, 2, ..., n \quad (8)$$

$$\sum_{j=1}^{n} y_{ij1} \ge \sum_{j=1}^{n} y_{ij2} \ge ... \ge \sum_{j=1}^{n} y_{ijn},\ i = 1, 2, ..., m \quad (9)$$

$$y_{ijs} \in \{0, 1\},\ i = 1, 2, ..., m,\ j = 1, 2, ..., n,\ s = 1, 2, ..., n \quad (10)$$

where

$$w_{ijs} = \begin{cases} s\alpha_{ij} p_{ij}, & i = 1, 2, ..., m,\ j = 1, 2, ..., n,\ s \le l_i, \\ (s+\delta_i l_i) p_{ij}, & i = 1, 2, ..., m,\ j = 1, 2, ..., n,\ s > l_i. \end{cases} \quad (11)$$

Constraint (6) makes sure that each position $(s \le l_i, i = 1, 2, ..., m)$ on each machine is taken by one job. Constraint (7) makes sure that each position $(s \ge l_i + 1, i = 1, 2, ..., m)$ on each machine is taken by at most one job. Constraint (8) makes sure that each job is scheduled exactly once. Constraint (9) ensures that on every machine, the unassigned positions must precede all the assigned positions, so that $y_{ijs} = 1$ if and only if job J_i is in fact the sth to the last job on machine M_i.

From (11), we see that $w_{ij1} \le w_{ij2} \le \cdots \le w_{ijl_i}$ and $w_{ij(l_i+1)} \le w_{ij(l_i+2)} \le \cdots \le w_{ijn}$ and thus $\sum_{j=1}^{n} y_{ij1} \ge \sum_{j=1}^{n} y_{ij2} \ge ... \ge \sum_{j=1}^{n} y_{ijl_i}$ and $\sum_{j=1}^{n} y_{ij(l_i+1)} \ge \sum_{j=1}^{n} y_{ij(l_i+2)} \ge ... \ge \sum_{j=1}^{n} y_{ijn}$, respectively. Moreover, from constraints (6) and (7), we know that $\sum_{j=1}^{n} y_{ijl_i}(=1) \ge \sum_{j=1}^{n} y_{ij(l_i+1)}(\le 1)$ for $i = 1, 2, ..., m.$ So, the inequality $\sum_{j=1}^{n} y_{ij1} \ge \sum_{j=1}^{n} y_{ij2} \ge ... \ge \sum_{j=1}^{n} y_{ijn}$ holds, for $i = 1, 2, ..., m.$ Hence, constraint (9) can be removed from the formulation without affecting the optimal solution of the problem. As a result, the $Rm | T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} | \sum C_j$ problem can be formulated as the following assignment problem and, therefore, can be solved in $O(mn^3)$ time (Brucker, 2001).

$$\text{Minimize } \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{s=1}^{n} w_{ijs} y_{ijs} + \sum_{i=1}^{m} l_i t_i$$

subject to (6), (7), (8), and (10).

In addition, since there are n jobs to be assigned to m unrelated parallel machines, we have that $0 \le l_1 + l_2 + ... + l_m \le n$. Let $l_{m+1} = n - (l_1 + l_2 + ... + l_m) \ge 0$. This means that $l_1 + l_2 + ... + l_{m+1} = n$. By Lemma 1, the number of nonnegative integer solutions to $l_1 + l_2 + ... + l_{m+1} = n$ is $C(n+m, m)$. By Lemma 2, the number $C(n+m, m)$ is bounded from above by $(2n)^m / m!$. Thus, we conclude that the following theorem holds.

Theorem 1. The $Rm | T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} | \sum C_j$ problem can be solved in $O(n^{m+3})$ time .

Note that if the number of machines m is fixed, then the $Rm | T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} | \sum C_j$ problem is polynomially solvable.

4. Conclusions

In this note, we extended the problem studied by Wang et al. (2011) and Wang et al. (2012). We provided a more efficient algorithm and showed that the $Rm | T_i = t_i + \delta_i t, rm, 0 < \alpha_{ij} | \sum C_j$ problem can be solved in $O(n^{m+3})$ time no matter what $0 < \alpha_{ij} \le 1$ or $\alpha_{ij} > 1$. Further research might be to consider the problem with multi-maintenance activities, other shop settings or optimizing other performance measures.

References

Brucker, P. (2001). *Scheduling algorithms*. New York: Springer-Verlag.

Gawiejnowicz, S. (2007). Scheduling deteriorating jobs subject to job or machine availability constraints. *European Journal of Operational Research*, *180*, 472-478.

Hsu, C. J., Low, C., & Su, C. T. (2010). A single-machine scheduling problem with maintenance activities to minimize make span. *Applied Mathematics and Computation*, *215*, 3929-3935.

Ji, M., He, Y., & Cheng, T. C. E. (2006). Scheduling linear deteriorating jobs with an availability constraint on a single machine. *Theoretical Computer Science*, *362*, 115-126.

Kubzin, M. A., & Strusevich, V. A. (2006). Planning machine maintenance in two-machine shop scheduling. *Operations Research*, *54,* 789-800.

Kuo, W. H., & Yang, D. L. (2008). Minimizing the make span in a single machine scheduling problem with the cyclic process of an aging effect. *Journal of the Operational Research Society, 59*, 416-20.

Lee, C. L., & Leon, V. J. (2001). Machine scheduling with a rate-modifying activity. *European Journal of Operational Research*, *128*, 119-128.

Lee, W. C., & Wu, C. C. (2008). Multi-machine scheduling with deteriorating jobs and scheduled maintenance. *Applied Mathematical Modelling*, *32*, 362-373.

Lodree Jr., E. J., & Geiger, C. D. (2010). A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration. *European Journal of Operational Research*, *201*, 644-648.

Low, C., Hsu, C. J., & Su, C. T. (2008). Minimizing the make span with an availability constraint on a single machine under simple linear deterioration. *Computers & Mathematics with Applications*, *56*, 257-265.

Mosheiov, G., & Sidney, J. B. (2010). Scheduling a deteriorating maintenance activity on a single machine. *Journal of the Operational Research Society*, *61*, 882-887.

Mott, J. L., Kandel, A., & Baker, T. P. (1986). *Discrete mathematics for computer scientists & mathematicians*. Englewood Cliffs, NJ: Prentice-Hall.

Wang, L. Y., Huang, X.,·Ji, P.,·& Feng, E. M. (2012). Unrelated parallel-machine scheduling with deteriorating maintenance activities to minimize the total completion time. *Optimization Letters*, doi: 10.1007/s11590-012-0472-x.

Wang, J. J., Wang, J. B., & Liu, F. (2011). Parallel machines scheduling with a deteriorating maintenance activity. *Journal of the Operational Research Society*, *62*, 1898-1902.

Wu, C. C., & Lee, W. C. (2003). Scheduling linear deteriorating jobs to minimize make span with an availability constraint on a single machine. *Information Processing Letters*, *87*, 89-93.

Yang, S. J. (2011). Parallel machines scheduling with simultaneous considerations of position-dependent deterioration effects and maintenance activities. *Journal of the Chinese Institute of Industrial Engineers*, *28*, 270-280.

Yang, S. J. (2013). Unrelated parallel-machine scheduling with deterioration effects and deteriorating multi-maintenance activities for minimizing the total completion time. *Applied Mathematical Modelling*, *37*, 2995-3005.

Yang, S. J., & Yang, D. L. (2010a). Single-machine scheduling problems with aging/ deteriorating effect under an optional maintenance activity consideration. *INFOR: Information Systems and Operational Research*, *48*, 171-179.

Yang, S. J., & Yang, D. L. (2010b). Minimizing the make span on single-machine scheduling with aging effect and variable maintenance activities. *Omeg*a, *38*, 528-533.

Yang, S. J., & Yang, D. L. (2010c). Minimizing total completion time in single-machine scheduling with aging/deteriorating effects and deteriorating maintenance activities. *Computers and Mathematics with Applications*, *60*, 2161-2169.

Zhao, C. L., Tang, H. Y., & Cheng, C. D. (2009). Two-parallel machines scheduling with rate-modifying activities to minimize total completion time. *European Journal of Operational Research*, *198*, 354-357.

Zhao, C. L., & Tang, H. Y. (2010). Single machine scheduling with general job-dependent aging effect and maintenance activities to minimize make span. *Applied Mathematical Modelling*, *34*, 837-841.

具退化性維修之非等效平行機台總完工時間最小化生產排程研究

楊肅正、郭家源、李新濤

南開科技大學　工業管理系

摘　　要

過去十年，許多學者投入於具維修策略以改善生產效率之生產排程研究。Wang, Wang, and Liu (2011) 及 Wang, Huang, Ji, and Feng (2012) 分別發表了有關 n 個工作在 m 台非等效平行機台生產環境下，考慮具退化性維修之總完工時間最小化排程問題的研究成果。Wang 等人(2011) 證明本問題能以多項式時間 $O(n^{2m+3})$ 求解，而 Wang 等人(2012) 證明若機器維修後之工作效率能提高，則能以多項式時間 $O(n^{m+3})$ 求解。本研究則證明不論機器維修後之工作效率是否能提高，本問題均能以多項式時間 $O(n^{m+3})$ 求解。

關鍵詞：生產排程、機器維修、非等效平行機台、總完工時間