

## Research Article

# Single Machine Scheduling and Due Date Assignment with Past-Sequence-Dependent Setup Time and Position-Dependent Processing Time

Chuan-Li Zhao,<sup>1</sup> Chou-Jung Hsu,<sup>2</sup> and Hua-Feng Hsu<sup>2</sup>

<sup>1</sup> College of Mathematics and Systems Sciences, Shenyang Normal University, Shenyang, Liaoning 110034, China

<sup>2</sup> Department of Industrial Management, Nan Kai University of Technology, Nantou 542, Taiwan

Correspondence should be addressed to Chou-Jung Hsu; [jrsheu@nktu.edu.tw](mailto:jrsheu@nktu.edu.tw)

Received 23 July 2014; Accepted 14 August 2014; Published 27 August 2014

Academic Editor: Dehua Xu

Copyright © 2014 Chuan-Li Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper considers single machine scheduling and due date assignment with setup time. The setup time is proportional to the length of the already processed jobs; that is, the setup time is past-sequence-dependent (p-s-d). It is assumed that a job's processing time depends on its position in a sequence. The objective functions include total earliness, the weighted number of tardy jobs, and the cost of due date assignment. We analyze these problems with two different due date assignment methods. We first consider the model with job-dependent position effects. For each case, by converting the problem to a series of assignment problems, we proved that the problems can be solved in  $O(n^4)$  time. For the model with job-independent position effects, we proved that the problems can be solved in  $O(n^3)$  time by providing a dynamic programming algorithm.

## 1. Introduction

In many realistic scheduling environments, a job's processing time may be depending on its position in the sequence [1]. Two well-known special cases of this stream of research are (i) *positional deterioration (aging effect)*, where the processing time of a job increases as a function of its position in a processing sequence and (ii) *learning effect*, where the processing time of a job decreases as a function of its position in a processing sequence. Biskup [2] and Cheng and Wang [3] independently introduced the learning concept to scheduling research. Other studies include Mosheiov and Sidney [4], Mosheiov [5, 6], Wu et al. [7, 8], and Yin et al. [9, 10]. Biskup [11] presented an updated survey of the results on scheduling problems with the learning effect. Mosheiov [6] first mentioned the *aging effect* in scheduling research. Other studies include Mosheiov [12], Kuo and Yang [13], Janiak and Rudek [14], Zhao and Tang [15], and Rustogi and Strusevich [16], among others. Moreover, some studies consider scheduling problems with general position-dependent processing time. Mosheiov [17] considered a scheduling problem with general position-dependent processing time. The polynomial

algorithm is derived for makespan minimization on an m-machine proportionate flow shop. Zhao et al. [18] studied scheduling and due date assignment problem. They provided a unified model for solving the single machine problems with rejection and position-dependent processing time. Rustogi and Strusevich [19] presented a critical review of the known results for scheduling models with various positional effects.

Koulamas and Kyparisis [20] first introduced a scheduling problem with past-sequence-dependent (p-s-d) setup time. They assumed that the job setup time is proportional to the sum of processing time of all already scheduled jobs. It is proved that the standard single machine scheduling with p-s-d setup time can be solvable in polynomial time when the objectives are the makespan, the total completion time, and the total absolute differences in completion time, respectively. Wang [21] studied the single machine scheduling problems with time-dependent learning effect and p-s-d setup time considerations. He showed that the makespan minimization problem, the total completion time minimization problem, and the sum of the quadratic job completion time minimization problem can be solved in polynomial time, respectively. Yin et al. [22] considered a single machine scheduling model

with p-s-d setup time and a general learning effect. They showed that the single machine scheduling problems to minimize the makespan and the sum of the  $k$ th power of completion time are polynomially solvable under the proposed model. Hsu et al. [23] presented a polynomial-time algorithm for an unrelated parallel machine scheduling problem with setup time and learning effects to minimize the total completion time. Lee [24] proposed a model with the deteriorating jobs, the learning effect, and the p-s-d setup time. He provided the optimal schedules for some single machine problems. Huang et al. [25] considered some single machine scheduling problems with general time-dependent deterioration, position-dependent learning, and p-s-d setup time. They proved that the makespan minimization problem, the total completion time minimization problem, and the sum of the  $\mu$ th power of job completion time minimization problem can be solved by the SPT rule.

Meeting due dates is one of the most important objectives in scheduling (Gordon et al. [26]). In some situations, the tardiness penalties depend on whether the jobs are tardy, rather than how late they are. In these cases, the number of tardy jobs should be minimized (Yin et al. [27]). Kahlbacher and Cheng [28] considered scheduling problems to minimize costs for earliness, due date assignment, and weighted number of tardy jobs. They presented nearly a full classification for the single and multiple machine models. Shabtay and Steiner [29] studied two single machine scheduling problems. The objectives are to minimize the sum of weighted earliness, tardiness, and due date assignment penalties and minimize the weighted number of tardy jobs and due date assignment costs, respectively. They proved that both problems are strongly NP-hard and give polynomial solutions for some important special cases. Koulamas [30] considered the second problem of Shabtay and Steiner [29]. He presented a faster algorithm for a due date assignment problem with tardy jobs. Gordon and Strusevich [31] addressed the problems of single machine scheduling and due date assignment problems in which a job's processing time depends on its position in a processing sequence. The objective functions include the cost of the due dates, the total cost of discarded jobs that cannot be completed by their due dates, and the total earliness of the scheduled jobs. They presented polynomial-time dynamic programming algorithms for solving problems with two due date assignment methods, provided that the processing time of the jobs is positionally deteriorating. Hsu et al. [32] extended part of the objective functions proposed by Gordon and Strusevich [31] to the positional weighted earliness penalty and showed that the problems remain solvable in polynomial time.

## 2. Problem Formulation and Preliminaries

This paper studies the single machine scheduling problems with simultaneous consideration of due date assignment, p-s-d setup time, and position-dependent processing time.

The problem can be described as follows.

A set  $N = \{J_1, J_2, \dots, J_n\}$  of  $n$  jobs has to be scheduled on a single machine. All jobs are available for processing at time

zero and preemption is not permitted. Each job  $J_j$  has a basic processing time  $p_j$ . The actual processing time of job  $J_j$ , if scheduled in position  $r$  of a sequence, is given by

$$p_j^A = g(j, r) p_j, \quad (1)$$

where  $g(j, 1), g(j, 2), \dots, g(j, n)$  represent an array of job-dependent positional factors.

Each job  $J_j \in N$  has to be assigned a due date  $d_j$ , by which it is desirable to complete that job. Given a schedule, denote the completion time of job  $J_j$  by  $C_j$ . Job  $J_j$  is called tardy if  $C_j > d_j$ , and it is called nontardy if  $C_j \leq d_j$ . Let  $U_j = 1$  if job  $J_j$  is tardy and let  $U_j = 0$  if job  $J_j$  is nontardy. The earliness of  $J_j$  is defined as  $E_j = d_j - C_j$ , provided that  $C_j \leq d_j$ . In all problems considered in this paper, the jobs in set  $N$  have to be split into two subsets denoted by  $N_E$  and  $N_T$ . We refer to the jobs in set  $N_E$  as "nontardy," while the jobs in set  $N_T$  are termed "tardy." A penalty  $\beta_j$  is paid for the tardy job  $J_j \in N_T$ . Given a schedule  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ , we assumed that the p-s-d setup time of  $J_{[j]}$  is given as Koulamas and Kyriaris [20] did, as follows:

$$s_{[j]} = \delta \sum_{i=1}^{j-1} p_{[i]}^A, \quad j = 2, 3, \dots, n, \quad s_{[1]} = 0, \quad (2)$$

where  $\delta \geq 0$  is a normalizing constant.

The purpose is to determine the optimal due dates and the processing sequence such that the following function is minimized:

$$F(\mathbf{d}, \pi) = \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \varphi(\mathbf{d}), \quad (3)$$

where  $\pi$  is the sequence of jobs,  $\alpha$  is the positive unit earliness cost,  $\mathbf{d}$  is the vector of the assigned due dates, and  $\varphi(\mathbf{d})$  denotes the cost of assigning the due dates that depends on a specific rule chosen for due date assignment. We denote the problem as

$$1 | p_j^A = p_j g(j, r), S_{psd} | \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \varphi(\mathbf{d}). \quad (4)$$

Most of the presented results hold for a general positional effect, that is, for any function  $g(j, r)$  that depends on both position  $r$  and job  $J_j$ . For each individual model, there is a particular rule that defines  $g(j, r)$  and explains how exactly the value of  $p_j$  changes, for example.

(i) *Job-Dependent Learning Effect (Mosheiov and Sidney [4]).* The actual processing time of a job  $J_j$ , if scheduled in position  $r$  of a sequence, is given by

$$p_j^A = p_j r^{a_j}, \quad (5)$$

where  $a_j \leq 0$  is a job-dependent learning parameter (include  $a_j = a$  as a special case, i.e.,  $p_j^A = p_j r^a$ , Biskup [2]).

(ii) *Job-Dependent Aging Effect (Zhao and Tang [15]).* The actual processing time of a job  $j$ , if scheduled in position  $r$  of a sequence, is given by

$$p_j^A = p_j r^{a_j}, \quad (6)$$

where  $a_j \geq 0$  is a job-dependent aging parameter (include  $a_j = a$  as a special case, i.e.,  $p_j^A = p_j r^a$ , Moshieov [17]).

(iii) *Positional Exponential Deterioration* (Wang [33]). The actual processing time of a job  $j$ , if scheduled in position  $r$  of a sequence, is given by

$$p_j^A = p_j a^{r-1}, \quad (7)$$

where  $a \geq 1$  is a given positive constant representing a rate of deterioration, which is common for all jobs.

We study our problem with the two most frequently used due date assignment methods.

(i) *The Common Due Date Assignment Method* (usually referred to as CON). Where all jobs are assigned the same due date, such that is  $d_j = d$  for  $j = 1, 2, \dots, n$  and  $d \geq 0$  is a decision variable.

(ii) *The Slack Due Date Assignment Method* (usually referred to as SLK). Where all jobs are given an equal flow allowance that reflects equal waiting time (i.e., equal slacks), such that is  $d_j = p_j^A + q$  for  $j = 1, 2, \dots, n$  and  $q \geq 0$  is a decision variable.

We first provide some lemmas.

**Lemma 1** (Hardy et al. [34]). *Let there be two sequences of numbers  $x_i$  and  $y_i$  ( $i = 1, 2, \dots, n$ ). The sum  $\sum_{i=1}^n x_i y_i$  of products of the corresponding elements is the least if the sequences are monotonically ordered in the opposite sense.*

It is not difficult to see that the following property is valid for both the variants of our problem.

**Lemma 2.** *There exists an optimal schedule in which the following properties hold: (1) all the jobs are processed consecutively without idle time and the first job starts at time 0 for both the variants of the problem; (2) all the nontardy jobs are processed before all the tardy jobs for both the variants of the problem.*

### 3. The CON Due Date Assignment Method

In the CON model,  $d_j = d$  ( $j = 1, 2, \dots, n$ ). We choose  $\gamma d$  as the cost function  $\varphi(d)$ , where  $\gamma$  is a positive constant. Thus, it follows from function (3) that our problem is to minimize the objective function:

$$F(d, \pi) = \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d. \quad (8)$$

The problem denotes

$$1 | p_j^A = p_j g(j, r), S_{psd}, CON | \alpha \sum_{J_j \in N_E} E_j + \alpha \sum_{J_j \in N_T} \beta_j U_j + \gamma d. \quad (9)$$

Kahlbacher and Cheng [28] provide an  $O(n^4)$  time algorithm for the problem  $1 | CON | \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T}$

$\beta_j U_j + \gamma d$ . In this section, we consider a generalization of the basic model with p-s-d setup times and position-dependent processing times. As a result of Lemma 2, we can restrict our attention to those schedules without idle times and search for the optimal schedule only among the schedules in which one of the jobs is on time.

Let  $\pi = [J_{[i]}, J_{[2]}, \dots, J_{[n]}]$ ; then

$$C_{[1]} = g([1], 1) p_{[1]},$$

$$C_{[2]} = C_{[1]} + s_{[2]} + g([2], 2) p_{[2]}$$

$$= g([1], 1) p_{[1]} + \delta g([1], 1) p_{[1]} + g([2], 2) p_{[2]}$$

$$= \sum_{k=1}^2 [1 + (2 - k) \delta] g([k], k) p_{[k]},$$

$\vdots$

$$C_{[j]} = \sum_{k=1}^j [1 + (j - k) \delta] g([k], k) p_{[k]},$$

$\vdots$

$$C_{[n]} = \sum_{k=1}^n [1 + (n - k) \delta] g([k], k) p_{[k]}.$$

$$\sum_{j=1}^n C_{[j]} = \sum_{j=1}^n (n - j + 1) \left( 1 + \frac{\delta(n - j)}{2} \right) g([j], j) p_{[j]}. \quad (10)$$

Note that we need only to consider the schedule in which all the nontardy jobs are processed before all the tardy jobs.

**Lemma 3.** *For the problem*

$$1 | p_j^A = p_j g(j, r), S_{psd}, CON | \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d, \quad (11)$$

*if the number of jobs in  $N_E$  is  $h$  (denote  $|N_E| = h$ ), there exists an optimal schedule  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ , such that  $d = C_{[h]}$ .*

*Proof.* Suppose  $\pi = [J_{[i]}, J_{[2]}, \dots, J_{[n]}]$  is an optimal schedule. Since jobs  $J_{[i]}, J_{[2]}, \dots, J_{[h]}$  are nontardy, then  $d \geq C_{[h]}$ . Let  $\Delta = d - C_{[h]}$ . Moving the due date  $\Delta$  units of time to the left such that  $d = C_{[h]}$ , the objective value will be decreasing  $(\alpha j + \gamma) \Delta$ , which is nonnegative. This means we can find a schedule with  $d = C_{[h]}$  that is at least as good as  $\pi$ .  $\square$

As a consequence of Lemma 3, we consider the schedule  $\pi = [J_{[i]}, J_{[2]}, \dots, J_{[n]}]$  with  $d = C_{[h]}$ .

Thus,

$$\begin{aligned}
 F(\mathbf{d}, \pi) &= \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d \\
 &= \alpha \sum_{j=1}^h E_{[j]} + \sum_{j=h+1}^n \beta_{[j]} + \gamma d \\
 &= \alpha \sum_{j=1}^h [C_{[h]} - C_{[j]}] + \sum_{j=h+1}^n \beta_{[j]} + \gamma C_{[h]} \\
 &= (\alpha h + \gamma) \left( \sum_{j=1}^h [1 + (h-j)\delta] g([j], j) p_{[j]} \right) \\
 &\quad - \alpha \sum_{j=1}^h (h-j+1) \left( 1 + \frac{1}{2} \delta (h-j) \right) g([j], j) p_{[j]} \\
 &\quad + \sum_{j=h+1}^n \beta_{[j]} \\
 &= \sum_{j=1}^h \left\{ \alpha (j-1) + \frac{1}{2} \alpha \delta (h-j) (h+j-1) \right. \\
 &\quad \left. + [1 + \delta (h-j)] \gamma \right\} g([j], j) p_{[j]} \\
 &\quad + \sum_{j=h+1}^n \beta_{[j]} \\
 &= \sum_{j=1}^h w_j g([j], j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]},
 \end{aligned} \tag{12}$$

where

$$\begin{aligned}
 w_j &= \left\{ \alpha (j-1) + \frac{1}{2} \alpha \delta (h-j) (h+j-1) \right. \\
 &\quad \left. + [1 + \delta (h-j)] \gamma \right\}, \quad j = 1, 2, \dots, h.
 \end{aligned} \tag{13}$$

Let  $x_{j,r}$  be a binary variable such that  $x_{j,r} = 1$  if job  $J_j$  is scheduled in the  $r$ th position and  $x_{j,r} = 0$ ; otherwise,  $j, r = 1, 2, \dots, n$ . From (12), we can formulate the problem with objective (8) as the following assignment problem  $A1(h)$ , which can be solved in  $O(n^3)$  time:

$$\begin{aligned}
 \text{Min} \quad & \sum_{j=1}^n \sum_{r=1}^n c_{j,r} x_{j,r} \\
 \text{s.t.} \quad & \sum_{r=1}^n x_{j,r} = 1, \quad j = 1, 2, \dots, n, \\
 & \sum_{j=1}^n x_{j,r} = 1, \quad r = 1, 2, \dots, n, \\
 & x_{j,r} \in \{0, 1\}, \quad j = 1, 2, \dots, n, \quad r = 1, 2, \dots, n,
 \end{aligned} \tag{14}$$

where

$$\begin{aligned}
 c_{j,r} &= \begin{cases} w_r g(j, r) p_j & \text{for } r = 1, 2, \dots, h \\ \beta_j & \text{for } r = h+1, h+2, \dots, n, \end{cases} \\
 w_r &= \left\{ \alpha (r-1) + \frac{1}{2} \alpha \delta (h-r) (h+r-1) \right. \\
 &\quad \left. + [1 + \delta (h-r)] \gamma \right\}, \quad r = 1, 2, \dots, h.
 \end{aligned} \tag{15}$$

Note that  $c_{j,r}$  is the cost of assigning job  $J_j$  ( $j = 1, 2, \dots, n$ ) in the  $r$ th ( $r = 1, 2, \dots, n$ ) position in the schedule.

In order to derive the optimal solution, we have to solve the above assignment problem  $A1(h)$  for any  $h = 1, 2, \dots, n$ . We summarize the results of the above analysis and present the following solution algorithm.

*Algorithm 4.*

*Step 1.* For  $h = 0$  ( $d = 0$ , all the jobs are tardy), calculate  $F(0) = \sum_{j=1}^n \beta_j$ .

*Step 2.* For  $h$  from 1 to  $n$ , solve the assignment problem  $A1(h)$  and calculate the corresponding objective value  $F(h)$ .

*Step 3.* The optimal value of the function  $F$  is equal to  $\min\{F(h) \mid h = 0, 1, 2, \dots, n\}$ .

As a result, we obtain the following theorem.

**Theorem 5.** Problem  $1|p_j^A = p_j g(j, r), S_{psd}, \text{CON}|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d$  can be solved in  $O(n^4)$  time.

We demonstrate our approach using the following example.

*Example 6.* Consider the problem  $1|p_j^A = p_j g(j, r), S_{psd}, \text{CON}|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d$ .

Let  $n = 5, h = 3$ . The weights are  $\alpha = 1, \gamma = 2$ , and  $\delta = 0.1$ .

The processing times are  $p_1 = 7, p_2 = 6, p_3 = 5, p_4 = 2$ , and  $p_5 = 1$ .

The tardy penalties are  $\beta_1 = 10, \beta_2 = 8, \beta_3 = 4, \beta_4 = 5$ , and  $\beta_5 = 6$ .

The positional effects are

$$g(j, r) = \begin{bmatrix} 2 & 1 & 3 & 2 & 4 \\ 1 & 3 & 2 & 2 & 3 \\ 2 & 3 & 1 & 4 & 3 \\ 1 & 2 & 3 & 1 & 3 \\ 2 & 1 & 2 & 3 & 4 \end{bmatrix}, \tag{16}$$

$$w_1 = 2.7, \quad w_2 = 3.4, \quad w_3 = 4.$$

The assignment problem A1(3) is

$$\begin{aligned}
 \text{Min} \quad & \sum_{j=1}^n \sum_{r=1}^n c_{j,r} x_{j,r} \\
 \text{s.t.} \quad & \sum_{r=1}^n x_{j,r} = 1, \quad j = 1, 2, \dots, n, \\
 & \sum_{j=1}^n x_{j,r} = 1, \quad r = 1, 2, \dots, n, \\
 & x_{j,r} \in \{0, 1\}, \quad j = 1, 2, \dots, n, \quad r = 1, 2, \dots, n,
 \end{aligned} \tag{17}$$

where

$$\begin{aligned}
 c_{j,r} &= \begin{bmatrix} w_1 g(1,1) p_1 & w_2 g(1,2) p_1 & w_3 g(1,3) p_1 & \beta_1 & \beta_1 \\ w_1 g(2,1) p_2 & w_2 g(2,2) p_2 & w_3 g(2,3) p_2 & \beta_2 & \beta_2 \\ w_1 g(3,1) p_3 & w_2 g(3,2) p_3 & w_3 g(3,3) p_3 & \beta_3 & \beta_3 \\ w_1 g(4,1) p_4 & w_2 g(4,2) p_4 & w_3 g(4,3) p_4 & \beta_4 & \beta_4 \\ w_1 g(5,1) p_5 & w_2 g(5,2) p_5 & w_3 g(5,3) p_5 & \beta_5 & \beta_5 \end{bmatrix} \\
 &= \begin{bmatrix} 37.8 & 23.8 & 84 & 10 & 10 \\ 16.2 & 61.2 & 48 & 8 & 8 \\ 27 & 51 & 20 & 4 & 4 \\ 5.4 & 13.6 & 24 & 5 & 5 \\ 5.4 & 3.4 & 8 & 6 & 6 \end{bmatrix}.
 \end{aligned} \tag{18}$$

The solution is

$$x_{j,r} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{19}$$

The optimal sequence is  $\pi^* = [J_4, J_5, J_3, J_1, J_2]$ ,  $d^* = 8.5$ . The total cost is 46.8.

#### 4. The SLK Due Date Assignment Method

In the SLK model,  $d_j = p_j^A + q$  ( $j = 1, 2, \dots, n$ ). We choose  $\gamma q$  as the cost function  $\varphi(\mathbf{d})$ , where  $\gamma$  is a positive constant. Thus, it follows from function (3) that our problem is to minimize the objective function

$$F(\mathbf{d}, \pi) = \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma q. \tag{20}$$

The problem denotes  $1|p_j^A = p_j g(j, r), S_{psd}, SLK|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma q$ .

Similar to the CON model, if the number of jobs in  $N_E$  is given, we have the following solution.

**Lemma 7.** For the problem  $1|p_j^A = p_j g(j, r), S_{psd}, SLK|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma q$ , if  $|N_E| = h$ , there exists an optimal schedule  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ , such that the slack time  $q = C_{[h-1]} + S_{[h]}$ .

*Proof.* The proof is similar to that of Lemma 3.

Let  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ ,  $q = C_{[h-1]} + S_{[h]}$ . Thus,

$$\begin{aligned}
 q &= \sum_{j=1}^{h-1} [1 + (h-j-1)\delta] g([j], j) p_{[j]} \\
 &\quad + \delta \sum_{j=1}^{h-1} g([j], j) p_{[j]} \\
 &= \sum_{j=1}^{h-1} [1 + (h-j)\delta] g([j], j) p_{[j]},
 \end{aligned}$$

$$E_{[j]} = d_{[j]} - C_{[j]}$$

$$= g([j], j) p_{[j]} + q - C_{[j]}$$

$$= g([j], j) p_{[j]} + \sum_{k=1}^{h-1} [1 + (h-k)\delta] g([k], k) p_{[k]}$$

$$- \sum_{k=1}^j [1 + (h-k)\delta] g([k], k) p_{[k]}$$

$$= g([j], j) p_{[j]} + \sum_{k=j+1}^{h-1} [1 + (h-k)\delta] g([k], k) p_{[k]},$$

$$1 \leq j \leq h-1. \tag{21}$$

Consequently,

$$\begin{aligned}
 \sum_{j=1}^{h-1} E_{[j]} &= \sum_{j=1}^{h-1} \left\{ g([j], j) p_{[j]} \right. \\
 &\quad \left. + \sum_{k=j+1}^{h-1} [1 + (h-k)\delta] g([k], k) p_{[k]} \right\} \\
 &= \sum_{j=1}^{h-1} g([j], j) p_{[j]} \\
 &\quad + \sum_{j=2}^{h-1} (j-1) [1 + (h-j)\delta] g([j], j) p_{[j]}.
 \end{aligned} \tag{22}$$

Therefore,

$$\begin{aligned}
 F(\mathbf{d}, \pi, h) &= \alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma q \\
 &= \alpha \sum_{j=1}^{h-1} g([j], j) p_{[j]}
 \end{aligned}$$

$$\begin{aligned}
& + \alpha \sum_{j=2}^{h-1} (j-1) [1 + (h-j)\delta] g([j], j) p_{[j]} \\
& + \sum_{j=h+1}^n \beta_{[j]} + \gamma q \\
& = \alpha \sum_{j=1}^{h-1} g([j], j) p_{[j]} \\
& + \alpha \sum_{j=2}^{h-1} (j-1) [1 + (h-j)\delta] g([j], j) p_{[j]} \\
& + \sum_{j=h+1}^n \beta_{[j]} + \gamma \sum_{j=1}^{h-1} [1 + (h-j)\delta] g([j], j) p_{[j]} \\
& = \sum_{j=1}^h w_j g([j], j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]},
\end{aligned} \tag{23}$$

where

$$w_j = \begin{cases} \alpha + \gamma (1 + (h-j)\delta) & \text{for } j = 1, \\ \alpha + [\alpha (j-1) + \gamma] [1 + \delta (h-j)] & \text{for } j = 2, \dots, h-1, \\ 0 & \text{for } j = h. \end{cases} \tag{24}$$

□

Since the objective functions for CON and SLK due date assignment methods have the same structure, we have the following solution.

Let  $x_{j,r}$  be a binary variable such that  $x_{j,r} = 1$  if job  $J_j$  is scheduled in the  $r$ th position and  $x_{j,r} = 0$ ; otherwise,  $j, r = 1, 2, \dots, n$ . If  $|N_E| = h$ , then we can formulate the problem with objective (20) as the following assignment problem A2( $h$ ), which can be solved in  $O(n^3)$  time:

$$\begin{aligned}
\text{Min} \quad & \sum_{j=1}^n \sum_{r=1}^n c_{j,r} x_{j,r} \\
\text{s.t.} \quad & \sum_{r=1}^n x_{j,r} = 1, \quad j = 1, 2, \dots, n, \\
& \sum_{j=1}^n x_{j,r} = 1, \quad r = 1, 2, \dots, n, \\
& x_{j,r} \in \{0, 1\}, \quad j = 1, 2, \dots, n, \quad r = 1, 2, \dots, n,
\end{aligned} \tag{25}$$

where

$$c_{j,r} = \begin{cases} w_r g(j, r) p_j & \text{for } r = 1, 2, \dots, h, \\ \beta_j & \text{for } r = h+1, h+2, \dots, n, \end{cases}$$

$w_r$

$$w_r = \begin{cases} \alpha + \gamma (1 + (h-r)\delta) & \text{for } r = 1, \\ \alpha + [\alpha (r-1) + \gamma] [1 + \delta (h-r)] & \text{for } r = 2, \dots, h-1, \\ 0 & \text{for } r = h. \end{cases} \tag{26}$$

In order to derive the optimal solution, we have to solve the above assignment problem A2( $h$ ) for any  $h = 1, 2, \dots, n$ .

As a result, we obtain the following theorem.

**Theorem 8.** The problem  $1|p_j^A = p_j g(j, r), S_{psd}, SLK|\alpha \sum_{j \in N_E} E_j + \sum_{j \in N_T} \beta_j U_j + \gamma d$  can be solved in  $O(n^4)$  time.

## 5. Job-Independent Position Effects Case

In this section, we explore the model with job-independent position effects; that is, the actual processing time of job  $J_j$ , if scheduled in position  $r$  of a sequence, is given by  $p_j^A = p_j g(r)$ , where  $g(1), g(2), \dots, g(n)$  represent an array of job-independent positional factors. In Section 4, we have shown that the general version (job-dependent position effects) can be solved in  $O(n^4)$  time. In the following, we present an  $O(n^3)$  time dynamic programming algorithm for solving the special version with job-independent position effects. The main idea that will be used in the development of our algorithm is similar to that of Shabtay et al. [35].

Based on the properties proved in Section 4, we have the following solutions.

For the CON model, if  $|N_E| = h$ ,  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ , and  $d = C_{[h]}$ , then

$$\begin{aligned}
F(\mathbf{d}, \pi, h) &= \sum_{j=1}^h \left\{ \alpha (j-1) + \frac{1}{2} \alpha \delta (h-j) (h+j-1) \right. \\
&\quad \left. + [1 + \delta (h-j)] \gamma \right\} g(j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]} \\
&= \sum_{j=1}^h \bar{w}_j g(j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]},
\end{aligned} \tag{27}$$

where

$$\begin{aligned}
\bar{w}_j &= \left\{ \alpha (j-1) + \frac{1}{2} \alpha \delta (h-j) (h+j-1) \right. \\
&\quad \left. + [1 + \delta (h-j)] \gamma \right\}, \quad j = 1, 2, \dots, h.
\end{aligned} \tag{28}$$



For the *SLK* model, if  $|N_E| = h$ ,  $\pi = [J_{[1]}, J_{[2]}, \dots, J_{[n]}]$ , and  $d = C_{[h-1]} + S_{[h]}$ , then

$$\begin{aligned} F(\mathbf{d}, \pi, h) &= [\alpha + \gamma(1 + (h-j)\delta)] g(1) p_{[1]} \\ &+ \sum_{j=2}^{h-1} \{\alpha + [\alpha(j-1) + \gamma][1 + \delta(h-j)]\} \\ &\times g(j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]} \\ &= \sum_{j=1}^h \bar{w}_j g(j) p_{[j]} + \sum_{j=h+1}^n \beta_{[j]}, \end{aligned} \quad (29)$$

where

$$\bar{w}_j = \begin{cases} \alpha + \gamma(1 + (h-j)\delta) & \text{for } j = 1, \\ \alpha + [\alpha(j-1) + \gamma][1 + \delta(h-j)] & \text{for } j = 2, \dots, h-1, \\ 0 & \text{for } j = h. \end{cases} \quad (30)$$

Using (27) and (29), and with any of the two previously mentioned due date assignments methods, let  $w_j = \bar{w}_j g(j)$ ; the objective function can be formulated as  $F(\mathbf{d}, \pi) = \sum_{j=1}^n w_j p_{\pi(j)}$  for the special case of  $N_E = N$ , where no jobs are tardy. From Lemma 1, the optimal job sequence is obtained by matching the largest  $w_j$  value to the job with the smallest  $p_j$  value, the second largest  $w_j$  value to the job with the second smallest  $p_j$  value, and so on. The index of the  $w_j$  matched with  $p_j$  specifies the position of job  $j$  in the optimal sequence. For example, first renumber the jobs in the LPT order such that  $p_1 \geq p_2 \geq \dots \geq p_n$ , and then reorder the positional weights such that  $w_{i_1} \leq w_{i_2} \leq \dots \leq w_{i_n}$  ( $i_1, i_2, \dots, i_n$  is a permutation of  $1, 2, \dots, n$ ), schedules job  $j$  in the position  $i_j$  ( $j = 1, 2, \dots, n$ ).

We now consider the due date assignment problem to minimize the objective function (3). Since the objective functions for all two due date assignment methods have the same structure, we provide a generic algorithm to solve these problems with two due date assignment methods. If set  $N_E$  is given, ( $|N_E| = h$ ), then we can reorder the positional weights such that  $w_{i_1} \leq w_{i_2} \leq \dots \leq w_{i_h}$ . Thus, an optimal job sequence of  $N_E$  is obtained in  $O(h \log h)$  time. However, in order to find the optimal solution for the due date assignment problem, the contribution of the total cost of the tardy jobs must be taken into account. Below, we present a new dynamic programming algorithm. For a given  $h$ , the idea of a dynamic programming algorithm to minimize the function (3) is as follows. We define the states of the form  $(i, r)$ , where  $i$  means that jobs  $J_1, J_2, \dots, J_i$  have been considered and  $r$ , ( $1 \leq r \leq \min\{i, h\}$ ), represents how many of these jobs have been sequenced as nontardy jobs. A state  $(i, r)$  is associated with  $f(i, r)$ , the smallest value of the objective function in the class of partial schedules for processing  $i$  jobs, provided that  $r$  of the these jobs has been sequenced nontardy. This method works by either each job tardy or nontardy. Next, all  $f(i, r)$  values can

be calculated by applying the recursion for  $i = 1, 2, \dots, n$  and  $r \geq \max\{1, h - (n - i)\}$ . The condition is that  $r \geq h - (n - i)$  is necessary to ensure that we do not consider states that might lead to a solution which has fewer than  $r$  jobs in set  $N_E$ : since  $|N_E| = h$  and there are  $r$  jobs that have been sequenced nontardy among the first  $i$  jobs, the remaining  $h - r$  nontardy jobs needed to be selected from the last  $n - i$  jobs. The formal statement of the algorithm is below.

*Algorithm 9.*

*Step 0.* Renumber the jobs in the LPT order such that  $p_1 \geq p_2 \geq \dots \geq p_n$ .

*Step 1.* Calculate positional weights  $w_j = \bar{w}_j g(j)$  where

$$\begin{aligned} \bar{w}_j &= \left\{ \alpha(j-1) + \frac{1}{2} \alpha \delta (h-j)(h+j-1) \right. \\ &\quad \left. + [1 + \delta(h-j)] \gamma \right\}, \quad (j = 1, 2, \dots, h) \end{aligned} \quad (31)$$

for the CON model and  $\bar{w}_1 = \alpha + \gamma[1 + (h-1)\delta]$ ,  $\bar{w}_j = \alpha + [\alpha(j-1) + \gamma][1 + \delta(h-j)]$ , ( $j = 2, \dots, h-1$ ), and  $\bar{w}_h = 0$  for the SLK model. Reorder the positional weights such that  $w_{i_1} \leq w_{i_2} \leq \dots \leq w_{i_n}$ . Initialize  $f(0, 0) = 0$ ,  $f(i, r) = \infty$  for  $r > i$ .

*Step 2.* For  $i$  from 1 to  $n$  calculate

$$f(i, 0) = f(i-1, 0) + \beta_i, \quad (32)$$

$$\begin{aligned} f(i, r) &= \min \{f(i-1, r) + \beta_i, f(i-1, r-1) + w_{i_r} p_{i_r}\}, \\ &\max\{1, h - (n - i)\} \leq r \leq \min\{i, h\}. \end{aligned} \quad (33)$$

*Step 3.* Compute the optimal value of the function  $f^*(h) = f(n, h)$ .

For a given  $h$  value, calculating all possible  $f(n, h)$  values using the above recursion relation requires  $O(nh)$  time. Since the value of positional weights (and the order of positional weights) can be altered by changing the  $h$  value, we must repeat the entire programming procedure for each  $h = 0, 1, 2, \dots, n$ . Thus the minimal objective value,  $F^*$ , is given by

$$F^* = \min_{h=0,1,\dots,n} \{f^*(h)\}. \quad (34)$$

Therefore, the following statement holds.

**Theorem 10.** Both the problems  $1|p_j^A = p_j g(r), S_{psd}, \text{CON}|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d$  and  $1|p_j^A = p_j g(r), S_{psd}, \text{SLK}|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d$  can be solved in  $O(n^3)$  time.

*Example 11.* Consider the problem  $1|p_j^A = p_j g(r), S_{psd}, \text{CON}|\alpha \sum_{J_j \in N_E} E_j + \sum_{J_j \in N_T} \beta_j U_j + \gamma d$ .

Let  $n = 5$  and  $h = 3$ . The weights are  $\alpha = 1$ ,  $\gamma = 2$ , and  $\delta = 0.1$ .

The processing times are  $p_1 = 7, p_2 = 6, p_3 = 5, p_4 = 2$ , and  $p_5 = 1$ .

The tardy penalties are  $\beta_1 = 10, \beta_2 = 8, \beta_3 = 4, \beta_4 = 5$ , and  $\beta_5 = 6$ .

The positional effects are  $g(1) = 8, g(2) = 7, g(3) = 3, g(4) = 4$ , and  $g(5) = 7$ .

Positional weights are  $w_1 = 21.6, w_2 = 23.8$ , and  $w_3 = 12$ .

Positional weights are reordered such that  $w_{i_1} \leq w_{i_2} \leq w_{i_3}$ ; that is,  $w_{i_1} = w_3, w_{i_2} = w_1$ , and  $w_{i_3} = w_2$ :

$$\begin{aligned}
 f(0, 0) &= 0, \\
 i &= 1, \\
 f(1, 0) &= f(0, 0) + \beta_1 = 10, \\
 f(1, 1) &= f(0, 0) + w_{i_1} p_1 = 84, \\
 i &= 2, \\
 f(2, 0) &= f(1, 0) + \beta_2 = 18, \\
 f(2, 1) &= \min \{f(1, 1) + \beta_2, f(1, 0) + w_{i_1} p_2\} \\
 &= \min \{92, 82\} = 82, \\
 f(2, 2) &= f(1, 1) + w_{i_2} p_2 = 203.6, \\
 i &= 3, \\
 f(3, 0) &= f(2, 0) + \beta_3 = 22, \\
 f(3, 1) &= \min \{f(2, 1) + \beta_3, f(2, 0) + w_{i_1} p_3\} \\
 &= \min \{86, 78\} = 78, \\
 f(3, 2) &= \min \{f(2, 2) + \beta_3, f(2, 1) + w_{i_2} p_3\} \\
 &= \min \{207.6, 190\} = 190, \\
 f(3, 3) &= f(2, 2) + w_{i_3} p_3 = 322.6, \\
 i &= 4, \\
 f(4, 0) &= f(3, 0) + \beta_4 = 27, \\
 f(4, 2) &= \min \{f(3, 2) + \beta_4, f(3, 1) + w_{i_2} p_4\} \\
 &= \min \{195, 121.2\} = 121.2, \\
 f(4, 3) &= \min \{f(3, 3) + \beta_4, f(3, 2) + w_{i_3} p_4\} \\
 &= \min \{327.6, 237.6\} = 237.6, \\
 i &= 5, \\
 f(5, 0) &= f(4, 0) + \beta_5 = 33, \\
 f(5, 3) &= \min \{f(4, 3) + \beta_5, f(4, 2) + w_{i_3} p_5\} \\
 &= \min \{243.6, 145\} = 145.
 \end{aligned} \tag{35}$$

Therefore,  $f(3) = f(5, 3) = 145$ .

The optimal sequence is  $\pi^* = [J_4, J_5, J_3, J_1, J_2]$ ,  $d^* = 41.9$ . The total cost is 145.

## 6. Conclusions

Scheduling problems involving position-dependent processing time have received increasing attention in recent years. In this paper, we considered single machine scheduling and due date assignment with setup time in which a job's processing time depends on its position in a sequence. The setup time is past-sequence-dependent (p-s-d). The objective functions include total earliness, the weighted number of tardy jobs, and the cost of due date assignment. The due date assignment methods used in this problem include common due date (CON) and equal slack (SLK). We have presented an  $O(n^4)$  time algorithm for the general case and an  $O(n^3)$  time dynamic programming algorithm for the special cases. In the paper, the model with position-dependent effects is considered. However, in some other situations, a job's processing time may be time-dependent or both position-dependent and time-dependent. Therefore, it is worthwhile for future research to investigate the model in which a job's processing time depends both on its position in a sequence and its start time. It is also interesting for future research to investigate the model in the context of other scheduling settings, including multimachine and job-shop scheduling.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

The authors are grateful to the editor and three anonymous referees for their helpful comments on the earlier version of the paper. This paper was supported in part by the Ministry of Science and Technology of Taiwan under Grant no. MOST 103-2221-E-252-004.

## References

- [1] A. Bachman and A. Janiak, "Scheduling jobs with position-dependent processing times," *Journal of the Operational Research Society*, vol. 55, no. 3, pp. 257–264, 2004.
- [2] D. Biskup, "Single-machine scheduling with learning considerations," *European Journal of Operational Research*, vol. 115, no. 1, pp. 173–178, 1999.
- [3] T. C. E. Cheng and G. Wang, "Single machine scheduling with learning effect considerations," *Annals of Operations Research*, vol. 98, pp. 273–290 (2001), 2000.
- [4] G. Mosheiov and J. B. Sidney, "Scheduling with general job-dependent learning curves," *European Journal of Operational Research*, vol. 147, no. 3, pp. 665–670, 2003.
- [5] G. Mosheiov, "Scheduling problems with a learning effect," *European Journal of Operational Research*, vol. 132, no. 3, pp. 687–693, 2001.



- [6] G. Mosheiov, "Parallel machine scheduling with a learning effect," *Journal of the Operational Research Society*, vol. 52, no. 10, pp. 1165–1169, 2001.
- [7] C. Wu, W. Lee, and T. Chen, "Heuristic algorithms for solving the maximum lateness scheduling problem with learning considerations," *Computers and Industrial Engineering*, vol. 52, no. 1, pp. 124–132, 2007.
- [8] C. Wu, P. Hsu, J. C. Chen, and N. S. Wang, "Genetic algorithm for minimizing the total weighted completion time scheduling problem with learning and release times," *Computers & Operations Research*, vol. 38, no. 7, pp. 1025–1034, 2011.
- [9] Y. Q. Yin, W. H. Wu, T. C. E. Cheng, and C. C. Wu, "Single-machine scheduling with time-dependent and position-dependent deteriorating jobs," *International Journal of Computer Integrated Manufacturing*, 2014.
- [10] Y. Q. Yin, T. C. E. Cheng, and C. C. Wu, "Scheduling with time-dependent processing times," *Mathematical Problems in Engineering*, vol. 2014, Article ID 201421, 2 pages, 2014.
- [11] D. Biskup, "A state-of-the-art review on scheduling with learning effects," *European Journal of Operational Research*, vol. 188, no. 2, pp. 315–329, 2008.
- [12] G. Mosheiov, "A note on scheduling deteriorating jobs," *Mathematical and Computer Modelling*, vol. 41, no. 8–9, pp. 883–886, 2005.
- [13] W.-H. Kuo and D.-L. Yang, "Minimizing the makespan in a single-machine scheduling problem with the cyclic process of an aging effect," *Journal of the Operational Research Society*, vol. 59, no. 3, pp. 416–420, 2008.
- [14] A. Janiak and R. Rudek, "Scheduling jobs under an aging effect," *Journal of the Operational Research Society*, vol. 61, no. 6, pp. 1041–1048, 2010.
- [15] C. L. Zhao and H. Y. Tang, "Single machine scheduling with general job-dependent aging effect and maintenance activities to minimize makespan," *Applied Mathematical Modelling*, vol. 34, no. 3, pp. 837–841, 2010.
- [16] K. Rustogi and V. A. Strusevich, "Single machine scheduling with general positional deterioration and rate-modifying maintenance," *Omega*, vol. 40, no. 6, pp. 791–804, 2012.
- [17] G. Mosheiov, "Proportionate flowshops with general position-dependent processing times," *Information Processing Letters*, vol. 111, no. 4, pp. 174–177, 2011.
- [18] C. Zhao, Y. Yin, T. C. E. Cheng, and C. Wu, "Single-machine scheduling and due date assignment with rejection and position-dependent processing times," *Journal of Industrial and Management Optimization*, vol. 10, no. 3, pp. 691–700, 2014.
- [19] K. Rustogi and V. A. Strusevich, "Simple matching vs linear assignment in scheduling models with positional effects: a critical review," *European Journal of Operational Research*, vol. 222, no. 3, pp. 393–407, 2012.
- [20] C. Koulamas and G. J. Kyparisis, "Single-machine scheduling problems with past-sequence-dependent setup times," *European Journal of Operational Research*, vol. 187, no. 3, pp. 1045–1049, 2008.
- [21] J. Wang, "Single-machine scheduling with past-sequence-dependent setup times and time-dependent learning effect," *Computers and Industrial Engineering*, vol. 55, no. 3, pp. 584–591, 2008.
- [22] Y. Yin, D. Xu, and J. Wang, "Some single-machine scheduling problems with past-sequence-dependent setup times and a general learning effect," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 9–12, pp. 1123–1132, 2010.
- [23] C.-J. Hsu, W.-H. Kuo, and D.-L. Yang, "Unrelated parallel machine scheduling with past-sequence-dependent setup time and learning effects," *Applied Mathematical Modelling*, vol. 35, no. 3, pp. 1492–1496, 2011.
- [24] W. Lee, "Single-machine scheduling with past-sequence-dependent setup times and general effects of deterioration and learning," *Optimization Letters*, vol. 8, no. 1, pp. 135–144, 2014.
- [25] X. Huang, G. Li, Y. Huo, and P. Ji, "Single machine scheduling with general time-dependent deterioration, position-dependent learning and past-sequence-dependent setup times," *Optimization Letters*, vol. 7, no. 8, pp. 1793–1804, 2013.
- [26] V. Gordon, J. Proth, and C. Chu, "A survey of the state-of-the-art of common due date assignment and scheduling research," *European Journal of Operational Research*, vol. 139, no. 1, pp. 1–25, 2002.
- [27] Y. Yin, M. Liu, T. C. E. Cheng, C. Wu, and S. Cheng, "Four single-machine scheduling problems involving due date determination decisions," *Information Sciences. An International Journal*, vol. 251, pp. 164–181, 2013.
- [28] H. G. Kahlbacher and T. C. E. Cheng, "Parallel machine scheduling to minimize costs for earliness and number of tardy jobs," *Discrete Applied Mathematics*, vol. 47, no. 2, pp. 139–164, 1993.
- [29] D. Shabtay and G. Steiner, "Two due date assignment problems in scheduling a single machine," *Operations Research Letters*, vol. 34, no. 6, pp. 683–691, 2006.
- [30] C. Koulamas, "A faster algorithm for a due date assignment problem with tardy jobs," *Operations Research Letters*, vol. 38, no. 2, pp. 127–128, 2010.
- [31] V. S. Gordon and V. A. Strusevich, "Machine scheduling and due date assignment with positionally dependent processing times," *European Journal of Operational Research*, vol. 198, pp. 57–62, 2009.
- [32] C. Hsu, S. Yang, and D. Yang, "Two due date assignment problems with position-dependent processing time on a single-machine," *Computers and Industrial Engineering*, vol. 60, no. 4, pp. 796–800, 2011.
- [33] J. Wang, "Flow shop scheduling jobs with position-dependent processing times," *Journal of Applied Mathematics & Computing*, vol. 18, no. 1–2, pp. 383–391, 2005.
- [34] G. H. Hardy, J. E. Littlewood, and G. Polya, *Inequalities*, Cambridge University Press, London, UK, 1967.
- [35] D. Shabtay, N. Gaspar, and L. Yedidsion, "A bicriteria approach to scheduling a single machine with job rejection and positional penalties," *Journal of Combinatorial Optimization*, vol. 23, no. 4, pp. 395–424, 2012.

