Research Article

Group Scheduling Problems with Time-Dependent and Position-Dependent DeJong’s Learning Effect

Li Sun, Lei Ning, and Jia-zhen Huo

1Business School, Shanghai Dianji University, Shanghai 201306, China
2School of Economics and Management, Tongji University, Shanghai 201804, China

Correspondence should be addressed to Lei Ning; ninglei1985@126.com

Received 16 November 2019; Accepted 11 January 2020; Published 17 February 2020

Academic Editor: Xiaoheng Chang

Copyright © 2020 Li Sun 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.

In this paper, we introduce a group scheduling model with time-dependent and position-dependent DeJong’s learning effect. The objectives of scheduling problems are to minimize makespan, the total completion time, and the total weighted completion time, respectively. We show that the problems remain solvable in polynomial time under the proposed model.

1. Introduction

In classical scheduling problems, the scheduling models routinely assume that job processing times are known and fixed throughout the period of production process. However, this assumption may be unrealistic in many situations that the processing time of jobs may be shortened due to learning effect over time. Production scheduling problems with learning effect have been paid much attention in recent years. To the best of our knowledge, there is little research that considers position-dependent and time-dependent processing time group scheduling model. Besides, the existing learning effect scheduling model suffers the drawback that when a job’s position or the starting processing time is sufficiently large in a schedule, its actual processing time is close to zero (infinity). This paper is to introduce a new scheduling model with position-dependent and time-dependent processing time, which overcomes the above shortcomings and is more general and realistic than the models existing in the literature.

The remaining part of this paper is organized as follows. In Section 2, we present a brief review of the existing scheduling model with learning effect. In Section 3, a precise formulation of the problem is given. Section 4 considers several single-machine scheduling problems with position-dependent and time-dependent DeJong’s learning effect to minimize makespan, the total completion time, and the total weighted completion time, respectively. The last section contains some conclusions of our model.

2. Review of Existing Models

Wright [1]; Biskup [2]; and Cheng and Wang [3] are among the pioneers that brought the learning effect into the field of scheduling research. Since then, learning effect has been widely employed in management science. To overcome the shortcoming associated with Wright’s [1] definition of learning effect that the improvement of learning effect is infinite, DeJong [4] introduced a new learning scheduling model, which is more realistic. Wang et al. [5] studied time-dependent DeJong’s learning effect model \( p_{j} = p_{j} + (a(a + \beta)) \), where \( a \geq 0, \beta \geq 0, \text{ and } 0 < a \leq 1 \) are the parameters obtained empirically and \( a + \beta = 1 \). Other studies have validated DeJong’s learning model, e.g., Okolowski and Gawiejnowicz [6] and Ji et al. [7, 8]. More recent papers which have considered scheduling problems with learning effect include studies of Wang and Wang [9, 10]; Wang et al. [11]; Xu et al. [12]; Chen et al. [13]; Toksari and Arik [14]; Bai et al. [15]; Mustu and Eren [16]; Pei et al. [17]; Zhang et al. [18]; Wang and Wang [19, 20]; Wang et al. [21], and Przybylski [22].

In many production processes, the efficiency can be improved by grouping various parts and products with
similar designs. This phenomenon is known as group technology in the literature. Many advantages have been claimed through the wide applications of group technology, for instance, Ji et al. [23] and Wang and Liu [24]. For the latest results on group scheduling problems, we refer the reader to the studies of Zhang et al. [18, 25]; Wang and Wang [26]; and Lu et al. [27] among others.

3. Problem Formulation

We first define the notations which are used throughout this paper, followed by the descriptions of the problems:

- \( m \): the number of groups (\( m \geq 2 \))
- \( n_i \): the number of jobs in group \( G_i \), \( i = 1, 2, \ldots, m \)
- \( n \): the total number of jobs, i.e., \( n_1 + n_2 + \cdots + n_m = n \)
- \( G_i \): group \( G_i \), \( i = 1, 2, \ldots, m \)
- \( G_{[i]} \): group scheduled in the \( i \)th position in a sequence, \( i = 1, 2, \ldots, m \)
- \( J_{ij} \): the job scheduled in the \( j \)th position in group \( G_i \), \( p_{ij} \): the normal processing time of job \( J_{ij} \), \( i = 1, 2, \ldots, m \) \( j = 1, 2, 3, \ldots, n_i \)
- \( p^\bullet_{[i][j]} \): the normal processing time of job \( J_{[i][j]} \) scheduled in the \( j \)th position of group \( G_i \) in a sequence, \( i = 1, 2, \ldots, m \) \( j = 1, 2, 3, \ldots, n_i \)
- \( s_i \): the sequence-independent setup time of group \( G_i \), \( i = 1, 2, \ldots, m \)
- \( P^\bullet_{[i][j]} \): the actual processing time of job \( J_{[i][j]} \) scheduled in the \( j \)th position of the \( i \)th group \( G_{[i]} \) in a sequence, \( i = 1, 2, \ldots, m \) \( j = 1, 2, 3, \ldots, n_i \)

There are \( n \) independent jobs. All jobs are classified into \( m \) groups \( G_1, G_2, \ldots, G_m \) by the similarities and to be processed on a single machine. All the jobs are available at time zero, and job preemption is not allowed. A group setup time is required if the machine switches to process from one group to another. Jobs in the same group are processed consecutively and need no setup time. The machine can handle one job at a time. Each job \( J_{ij} \) has a normal processing time \( p_{ij} \). The actual processing time of job will be shorter than its normal processing time due to the learning effect.

In this paper, we consider the following new time-dependent and position-dependent model:

\[
P^\bullet_{[i][j]} = p_{[i][j]}(M + (1 - M)a^{t_{[i][j]} / r^b}),
\]

where \( 0 < a < 1, b < 0 \) is the learning index, \( t_{[i][j]} \) is the total normal processing times of all jobs scheduled before \( J_{[i][j]} \) in group \( G_{[i]} \), i.e., \( t_{[i][j]} = \sum_{r=1}^{r-1} p_{[i][j]} \). \( M \) represents “the factor of incompressibility” (0 ≤ \( M \) ≤ 1). If \( M = 0 \), the model simplifies to the classical learning model \( p_{ij} = p_i^r \). If \( M = 1 \), the processing time of job is constant. The objectives are to find the optimal job sequence in each group and the optimal group sequence such that makespan, the total completion time, and the total weighted completion time are minimized, respectively. Using the three-field notation for scheduling problem, we denote our problems as:

\[
\begin{align*}
1 & | \text{GT, } s_1, P^\bullet_{[i][j]} = p_{[i][j]}(M + (1 - M)a^{t_{[i][j]} / r^b}) | C_{\text{max}}, \\
2 & | \text{GT, } s_2, P^\bullet_{[i][j]} = p_{[i][j]}(M + (1 - M)a^{t_{[i][j]} / r^b}) | \sum C_{ij}, \\
3 & | \text{GT, } s_3, P^\bullet_{[i][j]} = p_{[i][j]}(M + (1 - M)a^{t_{[i][j]} / r^b}) | \sum w_{ij}C_{ij}.
\end{align*}
\]

(2)

Before proving the problems, some lemmas are introduced as follows.

**Lemma 1.** Let \( F(x, r) = (1 - a^x(r + 1/r)^b)/x \), where \( r \geq 1 \), \( x > 0 \), \( 0 < a < 1 \), and \( b < 0 \). Then, \( F(x, r) \) is a decreasing function for all \( x > 0 \).

**Proof.** Taking the first derivative of \( F(x, r) \) with respect to \( x \), the following is obtained:

\[
\frac{dF(x, r)}{dx} = \frac{a^x(r + 1/r)^b(1 - x \ln a) - 1}{x^2}.
\]

Let

\[
f(x, r) = a^x(r + 1/r)^b(1 - x \ln a) - 1.
\]

Taking the first derivative of \( f(x, r) \) with respect to \( x \), the following is obtained:

\[
\frac{df(x, r)}{dx} = -a^x(r + 1/r)^b \ln^2 a.
\]

Since \( r \geq 1 \), \( x > 0 \), \( 0 < a < 1 \), and \( b < 0 \), we have \( -a^x(r + 1/r)^b \ln^2 a < 0 \). Hence, \( f(x, r) \) is a decreasing function for all \( x > 0 \). Therefore, \( f(x, r) < f(0, r) \). From \( f(0, r) = (r + 1/r)^b - 1 < 0 \), we obtain \( f(x, r) < 0 \). Hence, \( \frac{df(x, r)}{dx} < 0 \) and \( F(x, r) \) is a decreasing function for all \( x > 0 \).

**Lemma 2.** If the function \( h(x) \) is a differentiable convex function for \( a < b < c \), then

\[
\begin{align*}
(1) & h(b) - h(a) = h'(\xi_1)(b - a), \quad a < \xi_1 < b, \\
(2) & h(c) - h(a) = h'(\xi_2)(c - a), \quad a < \xi_2 < c, \quad (6)
\end{align*}
\]

(1) \( h'(\xi_1) \leq h'(\xi_2) \).

**Proof.** From the Lagrange mean value theorem, the lemma can be easily obtained.

4. Result of Optimization

In this section, we address the application of our model to solve the single-machine group scheduling problems involving makespan minimization, the total completion time minimization, and the total weighted completion time minimization.

4.1. Makespan Minimization

**Theorem 1.** For the \( 1|\text{GT}, s_1, P^\bullet_{[i][j]} = p_{[i][j]}(M + (1 - M)a^{t_{[i][j]} / r^b})|C_{\text{max}} \) problem, the optimal schedule is obtained: (i)
the job sequence in each group is in the smallest normal processing time order (SPT) and (ii) the groups can be sequenced in any order.

Proof. Suppose a sequence is \{G_{i1}, G_{i2}, \ldots, G_{im}\}. Therefore, the makespan of the sequence is obtained as follows:

\[
C_{\max} = \sum_{i=1}^{m} s_{i[i]} + \sum_{i=1}^{m} \sum_{j=1}^{n_i} p_{ij}[i] (M + (1 - M)a^{(0)\{i[j]\}}b). \tag{7}
\]

The first term of equation (7) is a constant. The value of the second term of equation (7) is determined by the job sequence in each group. Therefore, the makespan minimization of the studied model is independent of the sequence of groups. Hence, the optimal group sequence can be scheduled in any order, so (ii) follows.

Next, we determine the optimal job sequence in each group to minimize by implementing the adjacent job exchange argument. Let \(\pi_i\) and \(\pi_i'\) be two job schedules of a group \(G_i\) where the difference between \(\pi_i\) and \(\pi_i'\) is a pairwise interchange of two adjacent jobs \(J_{ik}\) and \(J_{il}\). That is, \(\pi_i = [S_1, J_{ik}, J_{il}, S_2]\) and \(\pi_i' = [S_1, J_{il}, J_{ik}, S_2]\), where \(S_1\) and \(S_2\) are the partial sequences. The positions of \(J_{ik}\) and \(J_{il}\) in sequence \(\pi_i\) are \(r\) and \(r+1\), respectively, which are reverse in sequence \(\pi_i'\). Furthermore, we assume that \(t\) denotes the completion time of the last job in \(S_1\) and \(p_{ik} \leq p_{il}\).

To prove \(\pi_i\) dominates \(\pi_i'\), it suffices to show that \(C_{G_i}(\pi_i') \leq C_{G_i}(\pi_i)\) and \(C_{G_i}(\pi_i') \leq C_{G_i}(\pi_i)\) for any job \(J_{im}\) in \(S_2\).

By definition, the completion time periods of jobs \(J_{ik}\) and \(J_{il}\) in sequence \(\pi_i\) and \(\pi_i'\) are given by

\[
C_{G_i}(\pi_i) = t + p_{ik} \left( M + (1 - M)a^{(0)\{i[k]\}}b \right),
\]

\[
C_{G_i}(\pi_i') = t + p_{ik} \left( M + (1 - M)a^{(0)\{i[k]\}}b \right) + p_{il} \left( M + (1 - M)a^{(0)\{i[l]\}}b \right) + t, \tag{8}
\]

Thus, we can obtain

\[
C_{G_i}(\pi_i') - C_{G_i}(\pi_i) = (1 - a^{p_{ik}}(r + 1) \frac{b}{p_{ik}} - 1 - a^{p_{il}}(r + 1) \frac{b}{p_{il}}). \tag{9}
\]

From \(p_{ik} \leq p_{il}\) and Lemma 1, it is easy to obtain that \(C_{G_i}(\pi_i') - C_{G_i}(\pi_i) \geq 0\). Hence, we have \(C_{G_i}(\pi_i') \leq C_{G_i}(\pi_i)\).

Suppose \(J_{ij}\) is the first job in \(S_2\), we have

\[
C_{G_j}(\pi_i) = C_{G_j}(\pi_i') + p_{ij} \left( M + (1 - M)a^{(0)\{j[i]\}}b \right) \left( r + 2 \right), \tag{10}
\]

From \(G_j(\pi_i) \leq C_{G_j}(\pi_i')\), we have \(C_{G_j}(\pi_i) \leq C_{G_j}(\pi_i')\) and \(C_{G_i}(\pi_i) \leq C_{G_i}(\pi_i')\) for any job \(J_{im}\) in \(S_2\).

Repeating the job interchange argument for all jobs in each group not sequenced in the SPT order yields part (i) of Theorem 1 and we complete the proof of Theorem 1.

From Theorem 1, the problem \(1|GT, s_j, p_{[0][r]}| = p_{[0][r]}(M + (1 - M)a^{(0)\{i[r]\}}b)|C_{\max}\) can be solved by using Algorithm 1.

Algorithm 1

Step 1. Jobs in each group are scheduled in nondecreasing order of the normal processing time (the SPT order).

Step 2. Groups are scheduled in any order.

We address application of Algorithm 1 in the following example.

Example 1. There are five jobs classified into two groups \(G_1\) and \(G_2\). \(M = 0.5\), \(a = 1/2\), \(b = -1\), \(G_1 = [J_{11}, J_{12}]\), \(s_j = 8\).
Proof. We still use the notation of Theorem 1 to prove the optimal job sequence in each group. From the proof of Theorem 1, we have \( C_{ik}(\pi_1) \leq C_{ij}(\pi'_1) \) and \( C_{ij}(\pi_1) \leq C_{ik}(\pi'_1) \). Hence, it is obtained that \( C_{ik}(\pi_1) + C_{ij}(\pi_1) \leq C_{ik}(\pi'_1) + C_{ij}(\pi_1) \) and \( \pi_1 \) dominates \( \pi'_1 \). Therefore, we complete the proof of part (i).

Next, we determine the optimal group sequence. Let \( \pi_2 \) and \( \pi'_2 \) be two schedules where the difference between \( \pi_2 \) and \( \pi'_2 \) is a pairwise interchange of two adjacent groups \( G_i \) and \( G_j \) and job sequence in each group of \( \pi_2 \) and \( \pi'_2 \) is in SPT order. That is, \( \pi_2 = [\delta_1, G_i, G_j, \delta_2] \) and \( \pi'_2 = [\delta_1, G_j, G_i, \delta_2] \), where \( \delta_1 \) and \( \delta_2 \) are the partial sequences. Furthermore, denote \( t \) as the completion time of the last job in \( \delta_1 \). To prove \( \pi_2 \) dominates \( \pi'_2 \), it suffices to address \( \sum_{i=1}^{n_l} C_{ij}[l](\pi_2) + \sum_{i=1}^{n_j} C_{ij}[l](\pi'_2) \). By definition, the completion times of jobs of groups \( G_i \) and \( G_j \) in \( \pi_2 \) are given by

\[
C_{ij}[l](\pi_2) = t + s_i + P_{[l]}(1),
\]

\[
C_{ij}[l](\pi'_2) = t + s_i + P_{[l]}(1) + P_{[l]}(2)(M + (1 - M)\alpha^{l_1}(1)2^\beta),
\]

\[
C_{ij}[l](\pi_2) = t + s_i + \sum_{i=1}^{n_l} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta),
\]

\[
C_{ij}[l](\pi'_2) = C_{ij}[l](\pi_2) + s_j + P_{[l]}(1),
\]

\[
C_{ij}[l](\pi_2) = C_{ij}[l](\pi_2) + s_j + \sum_{i=1}^{n_l} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta).
\]

(13)

This completes the proof.

From Theorem 2, the problem \( 1|\text{GT}, s, p_{[l]}|_{\pi_2} = P_{[l]}[\pi_2](M + (1 - M)\alpha^{l_1}(1)2^\beta) \Sigma C_{ij} \) can be solved by the following algorithm:

Algorithm 2

Step 1. Jobs in each group are scheduled in a nondecreasing order of the normal processing time (the SPT order).

Step 2. To each group, calculate

\[
\theta_i = \frac{s_i + \sum_{i=1}^{n_l} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta)}{n_i},
\]

\[
i = 1, 2, \ldots, m.
\]

(17)

The completion times of jobs of groups \( G_j \) and \( G_j \) in \( \pi'_2 \) are given by

\[
C_{ij}[l](\pi'_2) = t + s_j + P_{[l]}(1) + P_{[l]}(2)(M + (1 - M)\alpha^{l_1}(1)2^\beta),
\]

\[
C_{ij}[l](\pi_2) = t + s_j + \sum_{i=1}^{n_j} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta),
\]

\[
C_{ij}[l](\pi'_2) = C_{ij}[l](\pi_2) + s_i + P_{[l]}(1),
\]

\[
C_{ij}[l](\pi_2) = C_{ij}[l](\pi_2) + s_i + \sum_{i=1}^{n_j} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta).
\]

(14)

Thus, we have

\[
\sum_{i=1}^{n_j} C_{ij}[l](\pi_2) + \sum_{i=1}^{n_l} C_{ij}[l](\pi_2) - \sum_{i=1}^{n_j} C_{ij}[l](\pi_2) - \sum_{i=1}^{n_l} C_{ij}[l](\pi_2)
\]

\[
= n_j \left( s_j + \sum_{i=1}^{n_j} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta) \right)
\]

\[- n_l \left( s_i + \sum_{i=1}^{n_l} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta) \right).
\]

(15)

We have \( \sum_{i=1}^{n_j} C_{ij}[l](\pi_2) + \sum_{i=1}^{n_l} C_{ij}[l](\pi_2) \leq \sum_{i=1}^{n_j} C_{ij}[l](\pi_2) + \sum_{i=1}^{n_l} C_{ij}[l](\pi_2) \), if and only if

\[
\frac{s_i + \sum_{i=1}^{n_l} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta)}{n_i} \leq \frac{s_j + \sum_{i=1}^{n_j} P_{[l]}(1)(M + (1 - M)\alpha^{l_1}(1)2^\beta)}{n_j}.
\]

(16)

Step 3. Groups are scheduled in the nondecreasing order of \( \theta_i \).

Obviously, it is easy to show that the total time for Algorithm 2 is \( O(n \log n) \).

We show application of Algorithm 2 in the following example.

Example 2. As seen in Example 1, we change the objective to the total completion time minimization. By Algorithm 2, we solve the problem as follows:

Step 1: In group \( G_{11} \), the optimal job sequence is \( J_{11} \rightarrow J_{12} \). In group \( G_{21} \), the optimal job sequence is \( J_{23} \rightarrow J_{12} \rightarrow J_{21} \).

Step 2: To groups \( G_1 \) and \( G_2 \), calculate
\[ \theta_1 = \frac{8 + 2 \times (0.5 + 0.5 \times (1/2)^{0} \times 1^{-1}) + 6 \times (0.5 + 0.5 \times (1/2)^{2} \times 2^{-1})}{2} = 6.69, \]
\[ \theta_2 = \frac{4 + 2 \times (0.5 + 0.5 \times (1/2)^{0} \times 1^{-1}) + 5 \times (0.5 + 0.5 \times (1/2)^{2} \times 2^{-1}) + 10 \times (0.5 + 0.5 \times (1/2)^{7} \times 3^{-1})}{3} = 4.61. \]

Therefore, the optimal group sequence is \( G_2 \rightarrow G_1 \) and the optimal schedule is \( J_{23} \rightarrow J_{22} \rightarrow J_{21} \rightarrow J_{11} \rightarrow J_{12} \).

Hence, the minimum total completion time is \( \sum C_{ij} = 79.65 \).

### 4.3. Total Weighted Completion Time Minimization

In our model, we consider the group scheduling to minimize the total weighted completion time if jobs in each group satisfy an agreeable condition.

**Theorem 3.** For the \( 1|\text{GT}, s_t, p_{t_i}^{a_j} | (M + 1) \alpha_i^{(a_j^{(M + 1)})} \sum w_{ij}C_{ij} \) problem, if jobs in each group have agreeable weighted normal processing time, i.e., \( p_{ij} \leq p_{il} \) implies \( w_{ij} \geq w_{il} \), for all jobs \( i_j \) and \( i_l \) in \( G_i \), \( i = 1, 2, \ldots, m \).

**Proof.** First, we show the optimal job sequence in each group. Without loss of generality, we still use the notation of Theorem 1. From Theorem 1 and \( p_{ik} \leq p_{il} \), we have \( w_{ik} \geq w_{il} \). Therefore,

\[
\begin{align*}
& \quad w_{ik}C_{ik}(\pi_i') + w_{il}C_{il}(\pi_i) - w_{ik}C_{ik}(\pi_i) - w_{il}C_{il}(\pi_i) \\
&= (w_{ik} + w_{il})(p_{il} - p_{ik})(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) + w_{il}p_{ik}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\quad - w_{il}p_{il}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\quad - w_{il}p_{il}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) + w_{il}p_{il}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\geq (w_{ik}p_{il} - w_{il}p_{il})(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) + w_{ik}p_{ik}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\quad - w_{ik}p_{ik}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\quad - w_{il}p_{il}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) + w_{il}p_{il}(M + (1 - M)a^{\sum_{l=1}^{i-1}P_{il} + P_{il}}(r + 1)^b) \\
&\quad + (w_{ik}h'(\xi_i') - w_{il}h'(\xi_i') + w_{ik}h'(\xi_i) - w_{il}h'(\xi_i))p_{ik}p_{il}.
\end{align*}
\]

From Lemma 2, it is easy to show \( h'(\xi_i') \geq h'(\xi_i) \). Since \( p_{ik} \leq p_{il} \) and \( w_{ik} \geq w_{il} \), it can be obtained \( u_{ik}C_{ik}(\pi_i) + w_{il}C_{il}(\pi_i) \leq u_{ik}C_{ik}(\pi_i') + w_{il}C_{il}(\pi_i') \). From Theorem 1, we have \( C_{il}(\pi_i) \leq C_{il}(\pi_i') \) for any job \( j_{il} \) in \( S_2 \). Hence, the job completion time of \( S_2 \) in \( \pi_i \) is earlier than the same job completion time of \( S_2 \) in \( \pi_i' \). Thus, the job in each group is in nondecreasing order of \( p_{ij}/w_{ij} \) (the WSPT order) in the optimal schedule. This completes the proof of part (i).
Next, determine the optimal group sequence. We still use the notation of Theorem 2. To prove \( \pi_2 \) dominates \( \pi'_2 \), it suffices to show that

\[
\sum_{i=1}^{n} w_{i[i]} C_{i[i]}(\pi_2) + \sum_{i=1}^{n} w_{j[i]} C_{j[i]}(\pi_2) \leq \sum_{i=1}^{n} w_{j[i]} C_{j[i]}(\pi'_2) + \sum_{i=1}^{n} w_{i[i]} C_{i[i]}(\pi'_2).
\]

For \( j \in \{1, \ldots, n\} \),

\[
\left( \sum_{i=1}^{n} w_{i[i]} \right) \left( s_j + \sum_{i=1}^{n} p_{i[i]} \left( M + (1 - M) a \sum_{i=1}^{n} P_{i[i]} b^{i} \right) \right) - \left( \sum_{i=1}^{n} w_{j[i]} \right) \left( s_j + \sum_{i=1}^{n} p_{i[i]} M + (1 - M) a \sum_{i=1}^{n} P_{i[i]} b^{i} \right).
\]

We have \( \sum_{i=1}^{n} w_{i[i]} C_{i[i]}(\pi_2) + \sum_{i=1}^{n} w_{j[i]} C_{j[i]}(\pi_2) \leq \sum_{i=1}^{n} w_{j[i]} C_{j[i]}(\pi'_2) + \sum_{i=1}^{n} w_{i[i]} C_{i[i]}(\pi'_2) \), if and only if

\[
s_j + \sum_{i=1}^{n} p_{i[i]} \left( M + (1 - M) a \sum_{i=1}^{n} P_{i[i]} b^{i} \right) \leq \frac{s_j + \sum_{i=1}^{n} p_{i[i]} M + (1 - M) a \sum_{i=1}^{n} P_{i[i]} b^{i}}{\sum_{i=1}^{n} w_{j[i]}}.
\]

Hence, the optimal groups can be sequenced in the nondecreasing order of \( \varphi_i \), and this completes the proof of the theorem.

From Theorem 3, if \( w_{i[j]} \) and \( p_{i[j]} \) for jobs in each group have agreeable weighted normal processing time, the problem 1|\( G_T \), \( s_i \), \( p_{A} \) agreeable weighted normal processing time, the problem can be solved by the following algorithm:

**Algorithm 3**

Step 1. Jobs in each group are scheduled in nondecreasing order of \( p_{i[j]} / w_{i[j]} \) (the WSPT order).

Step 2. Calculate

\[
\varphi_i = \frac{s_j + \sum_{i=1}^{n} p_{i[i]} \left( M + (1 - M) a \sum_{i=1}^{n} P_{i[i]} b^{i} \right)}{\sum_{i=1}^{n} w_{i[j]}}, 
\]

\[i = 1, 2, \ldots, m.\]

**Example 3.** Also turn to the Example 1, change the objective to the total weighted completion time minimization \( (w_{11} = 2, w_{21} = 1, w_{22} = 1, w_{23} = 2, w_{24} = 4) \). Obviously, jobs in each group of this instance have the agreeable weighted normal processing times in Theorem 3. By Algorithm 3, we solve the problem as follows:

Step 1: In group \( G_1 \), the optimal job sequence is \( J_{11} \rightarrow J_{12} \). In group \( G_2 \), the optimal job sequence is \( J_{23} \rightarrow J_{22} \rightarrow J_{21} \).

Step 2: To groups \( G_1 \) and \( G_2 \), calculate

\[
\varphi_1 = \frac{8 + 2 \times (0.5 + 0.5 \times (1/2)^{-1} + 6 \times (0.5 + 0.5 \times (1/2)^{-1} + 2^{-1})}{3} = 4.46.
\]

\[
\varphi_2 = \frac{4 + 2 \times (0.5 + 0.5 \times (1/2)^{-1} + 5 \times (0.5 + 0.5 \times (1/2)^{-1} + 2^{-1}) + 10 \times (0.5 + 0.5 \times (1/2)^{-1} + 3^{-1})}{7} = 1.98.
\]

Therefore, the optimal group sequence is \( G_2 \rightarrow G_1 \) and the optimal schedule is \( J_{23} \rightarrow J_{22} \rightarrow J_{21} \rightarrow J_{11} \rightarrow J_{12} \).

Hence, the minimum total weighted completion time is \( \sum w_{ij} C_{ij} = 122.28 \).
5. Conclusions

We study new group scheduling models with position-dependent and time-dependent learning effect. The actual processing time of a job is a function of the total normal processing time and the total number of jobs scheduled ahead of it, which is motivated by DeJong’s learning schedule model. We show that the models can be solved polynomially when the objectives are the makespan minimization and the total completion time minimization. The total weighted completion time minimization is proved to be solved in polynomial time under an agreeable condition. It is suggested for future research to extend to other more practical learning effect scheduling models in the other machine environment, including multimachine and job-shop settings.

Data Availability

All data generated or analysed during this study are included in this article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (71532015).

References

[1] T. P. Wright, “Factors affecting the cost of airplanes,” Journal of the Aeronautical Sciences, vol. 3, no. 4, pp. 122–128, 1936.
[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 consideration,” Annals of Operations Research, vol. 98, no. 1–4, pp. 273–290, 2000.
[4] J. R. De Jong, “The effects of increasing skill on cycle time and its consequences for time standards,” Ergonomics, vol. 1, no. 1, pp. 51–60, 1957.
[5] J.-B. Wang, D. Wang, L.-Y. Wang, L. Lin, N. Yin, and W.-W. Wang, “Single machine scheduling with exponential time-dependent learning effect and past-sequence-dependent setup times,” Computers & Mathematics with Applications, vol. 57, no. 1, pp. 9–16, 2009.
[6] D. Okolowski and S. Gawiejnowicz, “Exact and heuristic algorithms for parallel-machine scheduling with DeJong’s learning effect,” Computers & Industrial Engineering, vol. 59, pp. 272–279, 2010.
[7] M. Ji, D. Yao, Q. Yang, and T. C. E. Cheng, “Machine scheduling with DeJong’s learning effect,” Computers & Industrial Engineering, vol. 80, pp. 195–200, 2015.
[8] M. Ji, X. Tang, X. Zhang, and T. C. E. Cheng, “Machine scheduling with deteriorating jobs and DeJong’s learning effect,” Computers & Industrial Engineering, vol. 91, pp. 42–47, 2016.
[9] J.-B. Wang and M.-Z. Wang, “Worst-case behavior of simple sequencing rules in flow shop scheduling with general position-dependent learning effects,” Annals of Operations Research, vol. 191, no. 1, pp. 155–169, 2011.
[10] J.-B. Wang and J.-J. Wang, “Flowshop scheduling with a general exponential learning effect,” Computers & Operations Research, vol. 43, pp. 292–308, 2014.
[11] J.-B. Wang, P. Ji, T. C. E. Cheng, and D. Wang, “Minimizing makespan in a two-machine flow shop with effects of deterioration and learning,” Optimization Letters, vol. 6, no. 7, pp. 1393–1409, 2012.
[12] J. Xu, C.-C. Wu, Y. Yin, C. Zhao, Y.-T. Chiu, and W.-C. Lin, “An order scheduling problem with position-based learning effect,” Computers & Operations Research, vol. 74, pp. 175–186, 2016.
[13] X. Chen, V. Chau, P. Xie, M. Sterna, and J. Blažewicz, “Complexity of late work minimization in flow shop systems and a particle swarm optimization algorithm for learning effect,” Computers & Industrial Engineering, vol. 111, pp. 176–182, 2017.
[14] M. D. Toksari and O. A. Arık, “Single machine scheduling problems under position-dependent fuzzy learning effect with fuzzy processing times,” Journal of Manufacturing Systems, vol. 45, pp. 159–179, 2017.
[15] D. Bai, M. Tang, Z.-H. Zhang, and E. D. Santihanez-Gonzalez, “Flow shop learning effect scheduling problem with release dates,” Omega, vol. 78, pp. 21–38, 2018.
[16] S. Mustu and T. Eren, “The single machine scheduling problem with sequence-dependent setup times and a learning effect on processing times,” Applied Soft Computing, vol. 71, pp. 291–306, 2018.
[17] J. Pei, X. Liu, B. Liao, P. M. Pardalos, and M. Kong, “Single-machine scheduling with learning effect and resource-dependent processing times in the serial-batching production,” Applied Mathematical Modelling, vol. 58, pp. 245–253, 2018.
[18] X. Zhang, L. Liao, W. Zhang, T. C. E. Cheng, Y. Tan, and M. Ji, “Single-machine group scheduling with new models of position-dependent processing times,” Computers & Industrial Engineering, vol. 117, pp. 1–5, 2018.
[19] J.-B. Wang and J.-J. Wang, “Single machine scheduling with sum-of-logarithm-processing-times-based and position-based learning effects,” Optimization Letters, vol. 8, no. 3, pp. 971–982, 2014.
[20] J.-B. Wang and J.-J. Wang, “Research on scheduling with job-dependent learning effect and convex resource-dependent processing times,” International Journal of Production Research, vol. 53, no. 19, pp. 5826–5836, 2015.
[21] J. B. Wang, F. Liu, and J. J. Wang, “Research on m-machine flow shop scheduling with truncated learning effects,” International Transactions in Operational Research, vol. 26, no. 3, pp. 1135–1151, 2019.
[22] B. Przybylski, “A new model of parallel-machine scheduling with integral-based learning effect,” Computers & Industrial Engineering, vol. 121, pp. 189–194, 2018.
[23] M. Ji, K. Chen, J. Ge, and T. C. E. Cheng, “Group scheduling and job-dependent due window assignment based on a common flow allowance,” Computers & Industrial Engineering, vol. 68, pp. 35–41, 2014.
[24] J.-J. Wang and Y.-J. Liu, “Single-machine bicriterion group scheduling with deteriorating setup times and job processing times,” Applied Mathematics and Computation, vol. 242, pp. 309–314, 2014.
[25] X. Zhang, W.-C. Lin, C.-J. Hsu, and C.-C. Wu, “Resource constrained scheduling problems with general truncated sum-of-processing time dependent effect under single machine and
unrelated parallel machines,” *Computers & Industrial Engineering*, vol. 110, pp. 344–352, 2017.

[26] J.-B. Wang and J.-J. Wang, “Single machine group scheduling with time dependent processing times and ready times,” *Information Sciences*, vol. 275, pp. 226–231, 2014.

[27] Y.-Y. Lu, J.-B. Wang, P. Ji, and H. He, “A note on resource allocation scheduling with group technology and learning effects on a single machine,” *Engineering Optimization*, vol. 49, no. 9, pp. 1621–1632, 2017.