Article

Single-Machine Group Scheduling Model with Position-Dependent and Job-Dependent DeJong’s Learning Effect

Jin Qian and Yu Zhan *

Abstract: This paper considers the single-group scheduling models with Pegels’ and DeJong’s learning effect and the single-group scheduling models with Pegels’ and DeJong’s aging effect. In a classical scheduling model, Pegels’ and DeJong’s learning effect is a constant or position-dependent, while the learning effect and aging effect are job-dependent in this paper. Compared with the classical learning model and aging model for scheduling, the proposed models are more general and realistic. The objective functions are to minimize the total completion time and makespan. We propose polynomial time methods to solve all the studied problems.

Keywords: single-machine scheduling; aging effect; learning effect; group technology

MSC: 90B35; 90B36

1. Introduction

In the traditional scheduling field, the processing time of a job is a constant, while in some actual situations, the processing times become shorter. A different approach to decreasing processing times is due to the concept of learning. A steady decline in processing times usually takes place by performing the same task repeatedly. This is called the learning effect. For instance, if a worker repeatedly processes the same jobs, due to the improvement of production technology and the accumulation of experience, the processing times become less. In some cases, the processing times of jobs may also become longer. As the machine ages, the processing times become longer. This is called the aging effect.

Compared with the classical learning model and aging model for scheduling, the proposed models are more general and realistic. In some classical scheduling model, Pegels’ and DeJong’s learning effect is a constant or position-dependent, while the learning effect and aging effect are job-dependent in this paper. This paper considers the single-group scheduling models with Pegels’ and DeJong’s learning effect and the single-group scheduling models with Pegels’ and DeJong’s aging effect. The objective functions are to minimize the total completion time and makespan. We propose polynomial time methods to solve all the studied problems.

2. Literature Review

The learning effect model was first proposed by Biskup [1]. Subsequently, many researchers have done a lot of work on the learning effect and aging effect. Mosheiov [2] explained the aging effect and deteriorating jobs. The SPT rule was no longer applicable to the optimal solution for the aging effect model. Yang [3] studied a single-machine scheduling problem with maintenance times and maintenance activities. This was an aging effect model that considered the linear deterioration and exponential deterioration of processing time. Zhang and Yan [4] considered a single-machine scheduling problem with setup time whose learning effect was based on group and position. The objective
functions were to minimize the total completion time and makespan. They gave polynomial time algorithms. Li et al. [5] considered a single-machine scheduling problem with linear resource or convex resource allocation. The learning effect of the processing time was based on position. They dealt with the problem of the slack due window by an assignment method. Lu et al. [6] considered a single-machine scheduling problem with setup time and convex resource allocation whose learning effect was based on the group position and the position within the group. The optimal sequence was for the jobs in each group to be sorted according to the SPT rule. Mustu and Eren [7] considered a single-machine learning scheduling problem with setup time which was based on position. The objective function was to minimize the total tardiness and it was an NP-hard problem. They proposed some heuristic algorithms to deal with the NP-hard problem. A single-machine group scheduling problem with setup time about linear resource and convex resource allocation was considered by Yin et al. [8], Zhang et al. [9], Zhao et al. [10] and Li et al. [11].

This paper studies the Pegels’ and DeJong’s learning effect scheduling model and Pegels’ and DeJong’s aging effect scheduling model. De Jong [12] proposed a learning scheduling model. Wang et al. [13] considered a single-machine scheduling problem with setup time which was a linear function of the previous actual processing time. A learning effect that was similar to DeJong’s learning effect was based on the sum of previous processing times. When the objective functions were to minimize the total completion time and makespan, they proved that the optimal schedule was sorted according to the SPT rule. Okołowski and Gawiejnowicz [14] studied a parallel-machine scheduling problem whose learning effect was DeJong’s learning effect. The objective function was to minimize the makespan. Since $Pm||C_{\text{max}}$ is a NP-hard problem, this problem was also a NP-hard problem. They proposed two branch-and-bound algorithms to solve this problem. Ji et al. [15] considered the machine scheduling problem with DeJong’s learning effect. For the single-machine scheduling problem, the objective functions were to minimize the total completion time and makespan. They proved that the optimal schedule was sorted according to the SPT rule. Ji et al. [16] studied a parallel-machine deteriorating scheduling problem with DeJong’s learning effect. The objective functions were to minimize the total completion time and makespan. When the objective functions were to minimize the total completion time, the jobs were sorted according to the SPT rule on each machine. Zhang et al. [17] considered both Pegels’ and DeJong’s learning effect scheduling model and Pegels’ and DeJong’s aging effect scheduling model. The objective functions were to minimize the total completion time and makespan. The optimal solution was that the jobs were sorted according to the SPT rule in each group. The group order was related to the objective functions. Sun et al. [18] considered a single-machine scheduling problem. A learning effect similar to DeJong’s learning effect was based on the sum of previous processing times and positions. The objective functions were to minimize the total completion time and makespan. They proved that the jobs in each group were sorted according to the SPT rule. Under different objective functions, they also gave a conclusion on the group order.

To increase efficiency, similarly designed or processed products are processed in groups. This phenomenon is known as group technology in the literature. Ji et al. considered a single-machine group scheduling and job-dependent due-window assignment problem [19]. Sun et al. considered a single-machine group scheduling with learning effect and resource allocation [20]. There have also been many scholars who have done a lot of work on group technology [21–25].

The problem is described in the Section 3. The proof of the polynomial time algorithm for four problems are given in the Section 4. A summary is given in the Section 5.

3. Notation and Problem Statement

Some notations that are used in this paper are introduced.
When three-region notation, DeJong’s and Pegels’ models can be defined as each job. They are the aging scheduling models.

When the normal processing time of each job. They are the learning scheduling models.

The machine can only process one job at a time. The actual processing time is setup time before each group is processed. The jobs are processed continuously in each group. The number of jobs divided into groups is a constant, \( u \). The DeJong’s aging or learning effect of \( b_{[u][v]} \) is inversely proportional to position \( v \). When \( 0 < K < 1 \), the actual processing time of each job is less than the normal processing time of each job. They are the learning scheduling models. When \( K = 1 \), the actual processing time is equal to the normal processing time. When \( K > 1 \), the actual processing time of each job is more than the normal processing time of each job. They are the aging scheduling models.

Our objectives are to minimize the total completion time and the makespan. By the three-region notation, DeJong’s and Pegels’ models can be defined as

\[
\begin{align*}
1) p_{[u][v]}^a &= p_{[u][v]} \left( K + (1 - K)s_{[u][v]} \right), \quad GT, t_v | C_{\text{max}}, \\
2) p_{[u][v]}^a &= p_{[u][v]} \left( K + (1 - K)\beta_{[u][v]}^{s-1} \right), \quad GT, t_v | C_{\text{max}}, \\
3) p_{[u][v]}^d &= p_{[u][v]} \left( K + (1 - K)s_{[u][v]} \right), \quad GT, t_v \sum_{v=1}^{l_v} \sum_{u=1}^{l_u} C_{[u][v]}, \\
4) p_{[u][v]}^d &= p_{[u][v]} \left( K + (1 - K)\beta_{[u][v]}^{s-1} \right), \quad GT, t_v \sum_{v=1}^{l_v} \sum_{u=1}^{l_u} C_{[u][v]}.
\end{align*}
\]
4. Research Method

The parameters of the traditional Pegels’ and DeJong’s learning effect are constants which are job-independent. This paper considers Pegels’ and DeJong’s models whose parameters are job-dependent. When $0 < K < 1$, they are the learning scheduling models. When $K > 1$, they are the aging scheduling models.

4.1. Makespan Minimization

**Theorem 1.** For the $1|p_u^d|[C_{max}]$, any $t_v|C_{max}$ problem, the optimal solution is that the job sequence within the group is solved by the assignment method and the order of the groups is arbitrary.

**Proof.** Suppose there are $l_v$ jobs in the $v$th group, $l_1 + \cdots + l_t = l$, $1 \leq v \leq t$. The completion time of each group is

$$C_{[l_v][1]} = t_1 + \sum_{u=1}^{l_1} p_{[u][1]} [K + (1 - K)u^{\alpha_{[u][1]}^v}],$$

$$C_{[l_v][v+1]} = C_{[l_v][v]} + t_{[v+1]} + \sum_{u=1}^{l_{v+1}} p_{[u][v+1]} [K + (1 - K)u^{\alpha_{[u][v+1]}^v}],$$

$$C_{max} = C_{[l_v][t]} = \sum_{v=1}^{t} t_{[v]} + \sum_{v=1}^{t} \sum_{u=1}^{l_v} p_{[u][v]} [K + (1 - K)u^{\alpha_{[u][v]}^v}].$$

It can be seen from $C_{max}$ that the order of the groups is arbitrary, and the job sequence within the group can be solved by the assignment method. So, the problem of sequencing jobs in each group could be solved in polynomial time.

$$\min \sum_{u=1}^{l_v} \sum_{h=1}^{l_v} p_{[u][v]} [K + (1 - K)h^{\alpha_{[u][v]}^v}] e_{u[v] h},$$

$$\text{s.t.} \ \sum_{u=1}^{l_v} e_{u[v] h} = 1, \ v = 1, \ldots, t; \ u = 1, \ldots, l_v,$$

$$\sum_{u=1}^{l_v} e_{u[v] h} = 1, \ v = 1, \ldots, t; \ h = 1, \ldots, l_v,$$

$$e_{u[v] h} = 0, \ or 1, \ v = 1, \ldots, t; \ u = 1, \ldots, l_v; \ h = 1, \ldots, l_v.$$

The algorithm is summarized as follows:

It is easy to show that the total time for Algorithm 1 is $O(n^3)$.

**Algorithm 1** $p_u^d = p_{[u][v]} K + (1 - K)\alpha_{[u][v]}^v, GT, t_v|C_{max}$

**Require:** $t, K, l_v, t_v, p_{[u][v]}, \alpha_{[u][v]}$

**Ensure:** The job sequence within the group and $C_{max}$

1. **First step:** Using the assignment method, the job sequence within the group is solved.
2. **Second step:** The order of the groups is arbitrary.
3. **Last step:** Calculate $C_{max}$.

**Example 1.** If there are five jobs in total, they are divided into two groups. $l = 5$, $t = 2$, $G_1 = \{ job_{11}, job_{21}, job_{31} \}$, $G_2 = \{ job_{12}, job_{22} \}$, $K = 0.5$, $l_1 = 2$, $l_2 = 3$, $p_{11} = 7$, $p_{21} = 5$, $p_{31} = 3$, $p_{12} = 4$, $p_{22} = 2$, $\alpha_{11} = -0.5$, $\alpha_{21} = -0.1$, $\alpha_{31} = -0.6$, $\alpha_{12} = -0.2$, $\alpha_{22} = -0.8$. 
Solution:
The processing time of the jobs in $G_1$ and $G_2$ at different positions are shown in Table 1 and Table 2, respectively.

Table 1. Processing time of the group $G_1$ jobs in different positions.

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>$7(0.5 + 0.5 \times 2^{-0.9})$</td>
<td>$7(0.5 + 0.5 \times 3^{-0.9})$</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>$5(0.5 + 0.5 \times 2^{-0.1})$</td>
<td>$5(0.5 + 0.5 \times 2^{-0.1})$</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>$3(0.5 + 0.5 \times 2^{-0.6})$</td>
<td>$3(0.5 + 0.5 \times 2^{-0.6})$</td>
</tr>
</tbody>
</table>

By the assignment method, the order of the jobs is $job_{31} \rightarrow job_{21} \rightarrow job_{11}$ in $G_1$.

Table 2. Processing time of the group $G_2$ jobs in different positions.

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>$4(0.5 + 0.5 \times 2^{-0.2})$</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>$2(0.5 + 0.5 \times 2^{-0.8})$</td>
</tr>
</tbody>
</table>

By the assignment method, the order of the jobs is $job_{12} \rightarrow job_{22}$ in $G_2$.

$$C_{\text{max}} = \sum_{v=1}^{1} t_v + \sum_{v=1}^{1} \sum_{u=1}^{l_v} p_{[u][v]} |0.5 + 0.5 \times u^{0.1}|$$

$$= 2 + 3(0.5 + 0.5 \times 2^{-0.1}) + 7(0.5 + 0.5 \times 3^{-0.5}) + 3 + 4(0.5 + 0.5 \times 2^{-0.8})$$

$$= 23.924.$$  

**Theorem 2.** For the $1|p^*[u][v]|K + (1 - K)p^*[u][v]|\beta G T, t_v|C_{\text{max}}$ problem, the optimal solution is that the job sequence within the group is solved by the assignment method and the order of the groups is arbitrary.

**Proof.** The completion time of each group is

$$C_{[t]}[1] = t_1 + \sum_{u=1}^{l_1} p_{[u][1]} |K + (1 - K)\beta_{[u][1]}|;$$

$$C_{[t]}[v+1] = C_{[t]}[v] + t_{v+1} + \sum_{u=1}^{l_{v+1}} p_{[u][v+1]} |K + (1 - K)\beta_{[u][v+1]}|;$$

$$C_{\text{max}} = C_{[t]}[t] = \sum_{v=1}^{l_t} t_v + \sum_{v=1}^{l_t} \sum_{u=1}^{l_v} p_{[u][v]} |K + (1 - K)\beta_{[u][v]}|.$$  

It can be seen from $C_{\text{max}}$ that the order of the groups is arbitrary, and the job sequence within the group can be solved by the assignment method. So, the problem of sequencing jobs in each group could be solved in polynomial time.

$$\min \sum_{u=1}^{l_u} \sum_{h=1}^{l_h} p_{[u][h]} |K + (1 - K)\beta_{[u][h]}|$$

s.t. $\sum_{u=1}^{l_u} \sum_{h=1}^{l_h} e_{u[h]} = 1, \quad v = 1, \ldots, t; u = 1, \ldots, l_v,$

$\sum_{u=1}^{l_u} \sum_{h=1}^{l_h} e_{u[h]} = 1, \quad v = 1, \ldots, t; h = 1, \ldots, l_v,$

$e_{u[h]} = 0, or 1, \quad v = 1, \ldots, t; u, h = 1, \ldots, l_v.$
The algorithm is summarized as follows:

It is easy to show that the total time for Algorithm 2 is $O(n^3)$.

**Algorithm 2**

For the Theorem 3.

**Proof.** Suppose there are $(v, p)$ jobs in group $G$, method. Let $v_{l_1} = 1$, $l_2 = 3$, First step: Calculate $l_3 = (v_1 + 1) + \frac{\sum_{u=1}^{b} p[u][K(1 - K)u^{u}[v]}}{l_b}$. The order of the groups is arbitrary. Using the assignment method, the job sequence within the group is solved.

\[
\sum_{v=1}^{t} l_v t_v = \sum_{v=1}^{t} l_v \sum_{k=1}^{u} t_v[k] + \sum_{u=1}^{l} p[v][u][K(1 - K)u^{u}[v]]
\]

**4.2. Total Completion Time Minimization**

**Theorem 3.** For the \( p[v][u][K(1 - K)u^{u}[v]] \) problem, the optimal solution is that the job sequence within the group can be solved by the assignment method and the groups are arranged in nondecreasing order of $\frac{l_b + \sum_{u=1}^{b} p[u][K(1 - K)u^{u}[v]]}{l_b}$.

**Proof.** Suppose there are $l_v$ jobs in the $v$th group, $l_1 + \cdots + l_t = l$, $1 \leq v \leq t$. The completion time of each job is

\[
C[u][1] = t[1] + \sum_{k=1}^{u} p[k][1][K(1 - K)k^{k}[1]], 1 \leq u \leq l_v,
\]

\[
C[u][v+1] = C[v][v] + t[v+1] + \sum_{k=1}^{u} p[k][v+1][K(1 - K)k^{k}[v+1]], 1 \leq u \leq l_{v+1}.
\]

Therefore, the total completion time is

\[
\sum_{v=1}^{t} l_v C[v][v+1] = \sum_{v=1}^{t} l_v t_v + \sum_{v=1}^{t} l_v C[v][v+1] + \sum_{v=1}^{t} \sum_{u=1}^{l} (l_v - u + 1) p[v][u][K(1 - K)u^{u}[v]].
\]

The first term is a constant in the above formula. The second term is

\[
\sum_{v=1}^{t} l_v C[v][v+1] = \sum_{v=1}^{t} l_v \sum_{k=1}^{u} t[v][k] + \sum_{u=1}^{l} p[v][u][K(1 - K)u^{u}[v]].
\]

We prove that the second term obtains the optimal solution by the adjacent exchange method. Let $S_1 = (\theta_1, G_b, G_c, \theta_2)$ and $S_2 = (\theta_1, G_c, G_b, \theta_2)$ be two job sequences with the same sequence except $G_b$ and $G_c$. $G_b$ is at the $\gamma$th position in the $S_1$ sequence, and $G_c$ is at the $(\gamma + 1)$th position in the $S_1$ sequence. There are $l_b$ jobs in group $G_b$, and there are $l_c$ jobs in group $G_c$.

\[
\sum_{v=1}^{t} l_v C[v][v+1] = \sum_{v=1}^{t} l_v \sum_{k=1}^{u} t[v][k] + \sum_{u=1}^{l} p[v][u][K(1 - K)u^{u}[v]]
\]

\[
= l_b \sum_{v=1}^{t} \{ t[v] + \sum_{u=1}^{l_b} p[v][u][K(1 - K)u^{u}[v]] + l_c \sum_{v=1}^{t} \{ t[v] + \sum_{u=1}^{l_c} p[v][u][K(1 - K)u^{u}[v]]\}
\]

\[
- l_c \sum_{v=1}^{t} \{ t[v] + \sum_{u=1}^{l_c} p[v][u][K(1 - K)u^{u}[v]]\} - l_b \sum_{v=1}^{t} \{ t[v] + \sum_{u=1}^{l_b} p[v][u][K(1 - K)u^{u}[v]]\}
\]

\[
= l_c \{ t_c + \sum_{u=1}^{l_c} p[v][u][K + (1 - K)u^{u}[v]]\} - l_b \{ t_b + \sum_{u=1}^{l_b} p[v][u][K + (1 - K)u^{u}[v]]\}.
\]
If $S_1$ is better than $S_2$, the above formula is less than 0. Then,

$$l_c \{ t_b + \sum_{u=1}^{l_b} p_{u|b}[K + (1 - K)u^{\alpha|u|}] \} < l_b \{ t_c + \sum_{u=1}^{l_c} p_{u|c}[K + (1 - K)u^{\alpha|u|}] \},$$

(17)

$$\frac{t_b + \sum_{u=1}^{l_b} p_{u|b}[K + (1 - K)u^{\alpha|u|}]}{l_b} < \frac{t_c + \sum_{u=1}^{l_c} p_{u|c}[K + (1 - K)u^{\alpha|u|}]}{l_c}.$$  

(18)

The third term can be solved by the assignment method.

$$\text{min} \sum_{u=1}^{l_c} \sum_{h=1}^{l_c} (l_{v|h} - h + 1)p_{u|v|v}[K + (1 - K)h^{\beta|v|}]e_{u|v|h},$$

s.t. $\sum_{h=1}^{l_c} e_{u|v|h} = 1, \quad v = 1, \ldots, t; u = 1, \ldots, l_{v|v},$

$\sum_{v=1}^{l_c} e_{u|v|h} = 1, \quad v = 1, \ldots, t; h = 1, \ldots, l_{v|v},$

$e_{u|v|h} = 0, \text{or} 1, \quad v = 1, \ldots, t; u, h = 1, \ldots, l_{v|v}.$

$\square$

The algorithm is summarized as follows:

It is easy to show that the total time for Algorithm 3 is $O(n^3)$.

**Algorithm 3**

1. **First step**: Using the assignment method, the job sequence within the group is solved.
2. **Second step**: Calculate the order of the groups by nondecreasing order of $\frac{\sum_{u=1}^{l_b} p_{u|b}[K + (1 - K)u^{\alpha|b|}]}{l_b}$.
3. **Last step**: Calculate $\sum_{v=1}^{l_c} \sum_{u=1}^{l_c} C_{u|v|v}$.

**Example 2.** The conditions are the same as in Example 1 and the objective function is changed from $C_{\max}$ to $\sum_{v=1}^{l_c} \sum_{u=1}^{l_c} C_{u|v|v}$.

**Solution:**

By the assignment method, the order of the jobs is $\text{job}_{31} \rightarrow \text{job}_{21} \rightarrow \text{job}_{11}$ in $G_1$ and the order of the jobs is $\text{job}_{12} \rightarrow \text{job}_{22}$ in $G_2$.

$$f(G_1) = \frac{t_1 + \sum_{u=1}^{l_1} p_{u|1}[K + (1 - K)u^{\alpha|u|}]}{l_1} = \frac{2 + \sum_{u=1}^{3} p_{u|1}[0.5 + 0.5 \times u^{\alpha|u|}]}{3} = 5.12,$$

(20)

$$f(G_2) = \frac{t_2 + \sum_{u=1}^{l_2} p_{u|2}[K + (1 - K)u^{\alpha|u|}]}{l_2} = \frac{3 + \sum_{u=1}^{2} p_{u|2}[0.5 + 0.5 \times u^{\alpha|u|}]}{2} = 4.29.$$  

Therefore, the order of the groups is $G_2 \rightarrow G_1$.

$$\sum_{v=1}^{2} \sum_{u=1}^{l_v} C_{u|v|v} = 3 + 4 + 8.57 + 13.57 + 18.4 + 23.92 = 71.46.$$

(21)

The proof of the $1|p_{u|v|v}|K + (1 - K)\beta_{u|v|v}^{-1}, \text{GT}, t_v|\sum_{v=1}^{l_c} \sum_{u=1}^{l_c} C_{u|v|v}$ Problem is the same as that of Theorem 3.
Theorem 4. For the 1\(|p_u^a|_0 = p[u][v]|K + (1 - K)\beta^{v-1}_{u[v]}|, GT, t_0|\sum_{v=1}^{t_p} \sum_{u=1}^{l_p} C[u][v] problem, the optimal solution is that the job sequence within the group can be solved by the assignment method and the groups are arranged in nondecreasing order of \(t_b + \sum_{u=1}^{l_p} p[u][v][K + (1 - K)\beta^{v-1}_{u[v]}].\)

The algorithm is summarized as follows:

It is easy to show that the total time for Algorithm 4 is \(O(n^3).\)

Algorithm 4 1\(|p_u^a|_0 = p[u][v]|K + (1 - K)\beta^{v-1}_{u[v]}|, GT, t_0|\sum_{v=1}^{t_p} \sum_{u=1}^{l_p} C[u][v]

Require: \(t, K, l_b, t_b, p_{uv}, \beta_{uv}\)

Ensure: The job sequence within the group, group sequence and \(\sum_{v=1}^{t_p} \sum_{u=1}^{l_p} C[u][v]\) problem or polynomial-time solvable problem. After a long period of exploration, the proposed polynomial time methods to solve all the studied problems.

5. Conclusions

This paper considered the single-group scheduling models with Pegels’ and DeJong’s learning effect and the single-group scheduling models with Pegels’ and DeJong’s aging effect. When \(0 < K < 1,\) they are the learning scheduling models. When \(K > 1,\) they are the aging scheduling models. In a classical scheduling model, Pegels’ and DeJong’s learning effect is a constant or position-dependent, while the learning effect and aging effect were job-dependent in this paper. Compared with the classical learning model and aging model for scheduling, the proposed models were more general and realistic. The objective functions were to minimize the total completion time and makespan. We proposed polynomial time methods to solve all the studied problems.

In the future, we can also consider the multimachine Pegels’ and DeJong’s learning scheduling.

6. Limitations

The problem comes from a real production scheduling problem. In some practical single-group scheduling problems, the learning effect and aging effect are job-dependent. We conducted a deep research study on this issue to prove whether it was an NP-hard problem or polynomial-time solvable problem. After a long period of exploration, the present result was achieved.

Author Contributions: The work presented here was performed in collaboration among all authors. Conceptualization, J.Q.; methodology, J.Q.; validation, Y.Z.; investigation, Y.Z.; writing—original draft preparation, J.Q.; writing—review and editing, Y.Z.; supervision, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by J. Qian of the Fundamental Research Funds for the Central Universities grant number N2105020, Y. Zhan of the Fundamental Research Funds for the Central Universities grant number N2105021. and Y. Zhan of the Natural Science Foundation of Liaoning Province Project grant number 2021-MS-102.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank the editor and the anonymous reviewers for their helpful comments and insights that significantly improved our paper.
**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

3. Yang, S.; Yang, D. Minimizing the makespan on single-machine scheduling with aging effect and variable maintenance activities. *Omega* 2010, 38, 528–533. [CrossRef]
12. De Jong, J.R. The effects of increasing skill on cycle time and its consequences for time standards. *Ergonomics* 1957, 1, 51–60. [CrossRef]