International Journal of Innovative Computing, Information and Control
Volume **20**, Number **4**, August **2024**

ICIC International ©2024 ISSN 1349-4198
pp. 1033–1044

GROUP SCHEDULING WITH SIMULTANEOUS CONSIDERATIONS OF READY TIMES AND DETERIORATING JOBS

Hongbin Bei and Jibo Wang*

School of Science
Shenyang Aerospace University
No. 37, Daoyi South Avenue, Shenbei New Area, Shenyang 110136, P. R. China
beihongbin@stu.sau.edu.cn; *Corresponding author: wangjibo@sau.edu.cn

Received October 2023; revised February 2024

Abstract. *In this paper, we focus on the group scheduling problem with ready times and deteriorating jobs on a single machine. Under the case of time dependent processing times and setup times, we assume that the setup times of the groups and processing times of the jobs are increasing functions of their starting times, i.e., the setup times of the groups and processing times of the jobs are linear proportional functions of their starting times. Our objective is to minimize the total completion time of all jobs with ready times. Since the problem considered is NP-hard, a heuristic algorithm and a branch-and-bound algorithm are proposed to deal with the problem of this article.*
Keywords: Group scheduling, Single machine, Ready times, Branch-and-bound algorithm, Heuristic algorithm

1. **Introduction.** In classical scheduling models, the processing times of the jobs are usually given constants. However, the processing times of the jobs may change due to the deteriorating effect (Blazewicz et al. [1], Sun and Geng [2], Gawiejnowicz [3], Sun et al. [4] and Pei et al. [5]). Wu and Lee [6] investigated the group scheduling problems with deterioration effect on a single machine where the deterioration rate was the same for all jobs, similarly, the deterioration rate was the same for all groups, and they settled makespan minimization and total completion time minimization problems. Wang et al. [7] examined the general case of above problem, i.e., the deterioration rates of jobs and groups were not the same. They showed that the makespan minimization problem can be tackled in polynomial time. Cheng et al. [8] proposed a sum-of-processing-times-based deteriorating model. Cheng et al. [9] concentrated on the scheduling problems with deteriorating jobs and past-sequence-based setup times. They introduced optimal solutions to handle the problems. Zhao [10] analyzed a single machine scheduling problem with deteriorating jobs. They proved that the weighted number of tardy jobs minimization problem is NP-hard. They proposed several algorithms to deal with this problem. Cheng et al. [11] took setup times into consideration and the objective was to minimize the maximum tardiness. This problem was also NP-hard and a branch-and-bound algorithm was established to deal with the problem. Wang and Wang [12] and Wang et al. [13] addressed the single-machine scheduling problems with deterioration effects and convex resource dependent processing times.

On the other hand, ready time (release time) also needs to be considered for some scheduling problems. Liu et al. [14] concentrated on a single machine group scheduling problem with deterioration effects and release dates, where the processing times and the setup times were both simple linear functions of their starting times. Wang et al. [15]

DOI: 10.24507/ijicic.20.04.1033

supposed the setup times were fixed constants, and they settled the makespan minimization problem with release dates. Lu et al. [16] aimed at a single machine group scheduling problem with release times, where the setup times and processing times were both decreasing linear functions of their starting times. They proved that the makespan problem can be tackled in polynomial time. Wang et al. [17] extended the group scheduling problem, where the setup times were given constants. They proved that several special cases of makespan minimization problem can be solved in polynomial time. Wang et al. [18] considered a group scheduling problem with deterioration effect, where the actual processing times and setup times were both proportional linear functions of their starting times. They showed that the makespan minimization and total weighted completion time minimization problems were polynomially solvable. Based on the deterioration model, Wang and Wang [19] studied the problem of minimizing makespan with ready times. They proved that the problem was polynomially solvable. Xu et al. [20] and Liu et al. [21] concentrated on a group makespan minimization problem, where the setup times were given constants. Xu et al. [20] proved that some special cases were polynomially solvable. For the general case, Liu et al. [21] established a heuristic algorithm and a branch-and-bound (B&B) algorithm.

However, in the actual industrial productions, not all jobs can be immediately put into processing, a certain amount of ready (preparation) time is necessary. Besides, the wear and tear of the machine, as well as the environment and other factors will lead to the prolonged processing times of the jobs, i.e., the deterioration effect. Given that deteriorating jobs and ready times are both simultaneously important, we will continue the work of Wang and Wang [19], but we study the total completion time minimization. The main contributions of this paper are summarized as follows:

- We settle the group scheduling problem for minimizing total completion time with ready times on a single machine.
- We establish a heuristic algorithm to deal with the problem. By comparing with the exact branch-and-bound (B&B) algorithm, we verify the effectiveness of the heuristic algorithm.
- Based on the computational experiments, we summarize the impact of different parameters on the results. Under large-scale instances, it is verified that the heuristic algorithm is suitable for this problem.

The rest of the paper is organized as follows. In Section 2, the scheduling model and problem description are given. In Section 3, some basic results and solution algorithms are given, followed by the computational experiments. The last section presents the conclusions.

2. Problem Description.

First, the notations are defined in Table 1.

The problem can be described as follows: there are $\hat{n}$ jobs to be divided into $\hat{m}$ groups and to be processed on a single machine. The machine can only process one job at a time and interruption is not allowed. A setup time is required if the machine switches from one group to another and all setup times of groups can start at time $t_0 \geq 0$. Let $\hat{J}_{ij}$ represent the jth job in the ith group $\hat{\Xi}_i$, $\hat{r}_{ij} > 0$ denote the ready time of job $\hat{J}_{ij}$, i.e., the job $\hat{J}_{ij}$ cannot be processed before its ready time $\hat{r}_{ij}$, $i = 1, 2, \ldots, \hat{m}$, $j = 1, 2, \ldots, \hat{n}_i$, where $\hat{n}_i$ represents the number of jobs within the ith group $\hat{\Xi}_i$ (i.e., $\hat{n}_1 + \hat{n}_2 + \cdots + \hat{n}_{\hat{m}} = \hat{n}$). As in Wang and Wang [19], we consider the following proportional deterioration model, i.e., the actual processing time of the job $\hat{J}_{ij}$ is

$$\hat{p}_{ij} = \hat{\alpha}_{ij}(\lambda + \mu t), \tag{1}$$

the actual setup time of group $\hat{\Xi}_i$ is

$$\hat{s}_i = \hat{\beta}_i(\lambda + \mu t), \tag{2}$$

where t is the starting time of job $\hat{J}_{ij}$ (group $\hat{\Xi}_i$), λ and μ are given positive constants. The objective is to find the optimal group sequence and the job sequence to minimize the total completion time (i.e., $\sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}$). By using three-field notation proposed by Graham et al. [22], the problem can be expressed as follows:

$$1 \left| \hat{r}_{ij}, \hat{p}_{ij} = \hat{\alpha}_{ij}(\lambda + \mu t), \hat{s}_i = \hat{\beta}_i(\lambda + \mu t), GT \right| \sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}. \tag{3}$$

Table 1. Symbols

Symbol	Meaning
$\hat{n}$	the total number of jobs
$\hat{m}$	the number of groups
$\hat{\Xi}_i$	the ith group
$\hat{n}_i$	the number of jobs in $\hat{\Xi}_i$, i.e., $\hat{n}_1 + \hat{n}_2 + \cdots + \hat{n}_{\hat{m}} = \hat{n}$
$\hat{J}_{ij}$	the jth job in $\hat{\Xi}_i$
$\hat{p}_{ij}$	the actual processing time of J_{ij}
$\hat{s}_i$	the actual setup time of $\hat{\Xi}_i$
$\hat{\alpha}_{ij}$	the deteriorating rate of job $\hat{J}_{ij}$
$\hat{\beta}_i$	the deteriorating rate of group $\hat{\Xi}_i$
$\hat{C}_{ij}$	the completion time of $\hat{J}_{ij}$
$[j]$	the jth position in a sequence
TCT	the total completion times of all jobs, i.e., $\sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}$

3. Main Results.

From Lenstra et al. [23], the problem $1 \left| \hat{r}_j \right| \sum_{j=1}^{n} \hat{C}_j$ is strongly NP-hard; hence, the general problem

$$1 \left| \hat{r}_{ij}, \hat{p}_{ij} = \hat{\alpha}_{ij}(\lambda + \mu t), \hat{s}_i = \hat{\beta}_i(\lambda + \mu t), GT \right| \sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}$$

is also strongly NP-hard.

In this paper, we consider the following special case: The ready times and the deteriorating rates of jobs $\hat{J}_{ij}$ and $\hat{J}_{ik}$ are agreeable, i.e., $\hat{r}_{ij} \leq \hat{r}_{ik}$ if and only if $\hat{\alpha}_{ij} \leq \hat{\alpha}_{ik}$. Then, we have

Lemma 3.1. *For the problem*

$$1 \left| \hat{r}_{ij}, \hat{p}_{ij} = \hat{\alpha}_{ij} \left(\lambda + \mu t \right), \hat{s}_i = \hat{\beta}_i \left(\lambda + \mu t \right), GT \right| \sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij},$$

if the ready times and deteriorating rates of jobs are agreeable, i.e., $\hat{r}_{ij} \leq \hat{r}_{ik}$ implies $\hat{\alpha}_{ij} \leq \hat{\alpha}_{ik}$ for all j and k in the ith group $\hat{\Xi}_i$ ($i = 1, 2, \ldots, \hat{m}$), the optimal job sequence π_i^* within group $\hat{\Xi}_i$ can be obtained by sequencing the jobs in the nondecreasing order of $\hat{r}_{ij}$, i.e., the smallest ready time (SRT) first rule.

Proof: Suppose that there is an optimal job sequence that does not follow SRT rule, i.e., we assume that $\pi^* = \left[S_1, \hat{J}_{ij}, \hat{J}_{ik}, S_2\right]$ is an optimal sequence, where $\hat{r}_{ij} \geq \hat{r}_{ik}$, S_1 and S_2 are partial sequence (notice that S_1 and S_2 may be empty). Furthermore, let $\pi' = \left[S_1, \hat{J}_{ik}, \hat{J}_{ij}, S_2\right]$ be another job sequence and the completion time of the last job in S_1 be A. Then, the completion times of jobs $\hat{J}_{ij}$ and $\hat{J}_{ik}$ under π^* are

$$\begin{aligned}
\hat{C}_{ij}(\pi^*) &= \max\{A, \hat{r}_{ij}\} + \hat{\alpha}_{ij}(\lambda + \mu \max\{A, \hat{r}_{ij}\}) \\
&= \max\left\{\left(A + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ij}), \left(\hat{r}_{ij} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ij})\right\} - \frac{\lambda}{\mu}, \\
\hat{C}_{ik}(\pi^*) &= \max\left\{\hat{C}_{ij}(\pi^*), \hat{r}_{ik}\right\} + \hat{\alpha}_{ik}\left(\lambda + \mu \max\left\{\hat{C}_{ij}(\pi^*), \hat{r}_{ik}\right\}\right) \\
&= \max\left\{\left(A + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ij})(1 + \mu\hat{\alpha}_{ik}), \left(\hat{r}_{ij} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ij})(1 + \mu\hat{\alpha}_{ik}),\right. \\
&\qquad \left.\left(\hat{r}_{ik} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ik})\right\} - \frac{\lambda}{\mu}.
\end{aligned}$$

Similarly, under π', we have

$$\begin{aligned}
\hat{C}_{ik}(\pi') &= \max\left\{\left(A + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ik}), \left(\hat{r}_{ik} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ik})\right\} - \frac{\lambda}{\mu}, \\
\hat{C}_{ij}(\pi') &= \max\left\{\left(A + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ik})(1 + \mu\hat{\alpha}_{ij}), \left(\hat{r}_{ik} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ik})(1 + \mu\hat{\alpha}_{ij}),\right. \\
&\qquad \left.\left(\hat{r}_{ij} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{ij})\right\} - \frac{\lambda}{\mu}.
\end{aligned}$$

Due to $\hat{r}_{ij} \geq \hat{r}_{ik}$, it is obvious that $\hat{C}_{ij}(\pi^*) \geq \hat{C}_{ik}(\pi')$ and $\hat{C}_{ik}(\pi^*) \geq \hat{C}_{ij}(\pi')$. Hence, we have $\sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}(\pi^*) \geq \sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}(\pi')$ which contradicts that π^* is the optimal sequence. Thus, we conclude that an optimal job sequence π_i^* within group $\hat{\Xi}_i$ can be obtained by sequencing the jobs in nondecreasing order of $\hat{r}_{ij}$. □

3.1. **Special case.** If $\hat{r}_{ij} = \hat{r} \neq 0$, the problem can be expressed as

$$1\left|\hat{r}_{ij} = \hat{r}, \hat{p}_{ij} = \hat{\alpha}_{ij}(\lambda + \mu t), \hat{s}_i = \hat{\beta}_i(\lambda + \mu t), GT\right| \sum_{i=1}^{\hat{m}} \sum_{j=1}^{\hat{n}_i} \hat{C}_{ij}.$$

According to Lemma 3.1, the optimal job sequence can be obtained by sequencing the jobs in nondecreasing order of $\hat{\alpha}_{ij}$ within each group. It is obvious that the starting times of all jobs except the first job are larger than $\hat{r}$. The completion of the first job is

$$\hat{C}_{[1][1]} = \max\left\{\left(t_0 + \frac{\lambda}{\mu}\right)\left(1 + \mu\hat{\beta}_{[1]}\right)\left(1 + \mu\hat{\alpha}_{[1][1]}\right), \left(\hat{r} + \frac{\lambda}{\mu}\right)\left(1 + \mu\hat{\alpha}_{[1][1]}\right)\right\} - \frac{\lambda}{\mu},$$

if the ready time $\hat{r}$ is larger, i.e., for all groups $\left(t_0 + \frac{\lambda}{\mu}\right)\left(1 + \mu\hat{\beta}_i\right) \leq \hat{r} + \frac{\lambda}{\mu}$ is satisfied, according to the algorithm proposed by Wang et al. [18], the optimal group sequence can be obtained by sequencing the groups in the non-decreasing order of $\rho\left(\hat{\Xi}_i\right)$, otherwise, we choose the group with the smallest deteriorating rate of the group (i.e., $\min\left\{\hat{\beta}_i \middle| i = 1, 2, \ldots, \hat{m}\right\}$) as the first group and for the remained groups, the optimal group sequence can be obtained by sequencing the groups in the non-decreasing order of $\rho\left(\hat{\Xi}_i\right)$, where

$$\rho\left(\hat{\Xi}_i\right)=\frac{\left(1+\mu\hat{\beta}_i\right)\prod_{l=1}^{n_i}\left(1+\mu\hat{\alpha}_{il}\right)-1}{\left(1+\mu\hat{\beta}_i\right)\sum_{k=1}^{n_i}\prod_{l=1}^{k}\left(1+\mu\hat{\alpha}_{il}\right)}. \tag{4}$$

Then, the problem $1\left|\hat{r}_{ij}=\hat{r},\hat{p}_{ij}=\hat{\alpha}_{ij}(\lambda+\mu t),\hat{s}_i=\hat{\beta}_i(\lambda+\mu t),GT\right|\sum_{i=1}^{\hat{m}}\sum_{j=1}^{\hat{n}_i}\hat{C}_{ij}$ can be solved by the following algorithm (i.e., Algorithm 1):

Algorithm 1

Step 1 The optimal job sequence can be obtained by sequencing the jobs in the non-decreasing order of $\hat{\alpha}_{ij}$ in each group, i.e., $\hat{\alpha}_{i[1]}\leq\hat{\alpha}_{i[2]}\leq\cdots\leq\hat{\alpha}_{i[n_i]}$.

Step 2 Calculate:

$$\rho\left(\hat{\Xi}_i\right)=\frac{\left(1+\mu\hat{\beta}_i\right)\prod_{l=1}^{n_i}\left(1+\mu\hat{\alpha}_{il}\right)-1}{\left(1+\mu\hat{\beta}_i\right)\sum_{k=1}^{n_i}\prod_{l=1}^{k}\left(1+\mu\hat{\alpha}_{il}\right)}.$$

Step 3 Choose the group with the smallest deteriorating rate of the group as the first group and sequence the remained groups in the non-decreasing order of $\rho\left(\hat{\Xi}_i\right)$.

Step 4 Sequence all groups in the non-decreasing order of $\rho\left(\hat{\Xi}_i\right)$.

Step 5 The optimal group sequence $\pi^*_{\hat{\Xi}}$ can be obtained from Step 3 and Step 4.

3.2. A heuristic algorithm.

For

$$1\left|\hat{r}_{ij},\hat{p}_{ij}=\hat{\alpha}_{ij}(\lambda+\mu t),\hat{s}_i=\hat{\beta}_i(\lambda+\mu t),GT\right|\sum_{i=1}^{\hat{m}}\sum_{j=1}^{\hat{n}_i}\hat{C}_{ij},$$

according to Lemma 3.1, the optimal job sequence can be obtained by sequencing the jobs in nondecreasing order of $\hat{r}_{ij}$ within each group. According to the special case considered above, a group sequence can be obtained, in addition, Liu et al. [21] established a heuristic algorithm for the problem $1\left|\hat{r}_{ij},\hat{p}_{ij}=\hat{\alpha}_{ij}\left(\lambda+\mu t\right),\hat{s}_i,GT\right|C_{\max}$, and then the following heuristic algorithm is proposed according to the special case (i.e., $\hat{r}_{ij}=\hat{r}$) and Liu et al. [21].

Remark 3.1. *The incremental point in Algorithm 2 is that the improvement strategy of this article is different from Liu et al. [21], i.e., our improvement strategy is to exchange the ath group and bth group; however, the improvement strategy of Liu et al. [21] is to place some job at all positions in a partial group sequence.*

3.3. Branch-and-bound (B&B) algorithm.

The sequence of jobs within each group is determined by Lemma 3.1. In order to find the lower bound, we propose the following lemma.

Lemma 3.2. *The sum of product*

$$\sum_{i=1}^{m}x_i\prod_{j=1}^{i}y_j$$

$(y_i>1)$ *is minimized if the sequence*

$$\frac{y_1-1}{x_1y_1},\frac{y_2-1}{x_2y_2},\ldots,\frac{y_m-1}{x_my_m}$$

Algorithm 2 (HA)

Step 1 Sequence the jobs in the nondecreasing order of $\hat{r}_{ij}$ within each group.

Step 2 Schedule groups in non-decreasing order of

$$\varphi\left(\hat{\Xi}_i\right) = \left(\hat{r}_{i1} + \frac{\lambda}{\mu}\right)(1 + \mu\hat{\alpha}_{i1}).$$

Step 3 Schedule groups in non-decreasing order of

$$\rho\left(\hat{\Xi}_i\right) = \frac{\left(1 + \mu\hat{\beta}_i\right)\prod_{l=1}^{\hat{n}_i}(1 + \mu\hat{\alpha}_{il}) - 1}{\left(1 + \mu\hat{\beta}_i\right)\sum_{k=1}^{\hat{n}_i}\prod_{l=1}^{k}(1 + \mu\hat{\alpha}_{il})}.$$

Step 4 Schedule groups in non-increasing order of

$$\theta\left(\hat{\Xi}_i\right) = \left(1 + \mu\hat{\beta}_i\right)\sum_{k=1}^{\hat{n}_i}\prod_{l=1}^{k}(1 + \mu\hat{\alpha}_{il}).$$

Step 5 Choose the best solution as an initial solution from Steps 2-4 by calculating the total completion time of all jobs, i.e.,

$$\sum_{i=1}^{\hat{m}}\sum_{j=1}^{\hat{n}_i}\hat{C}_{ij}.$$

Step 6 Set $a = 1$.

Step 7 Set $b = a + 1$.

Step 8 Exchange the ath group and bth group of the initial solution. Calculate the total completion time of all jobs of the new sequences, and choose the one with smaller result as the new solution.

Step 9 If $b \leq \hat{m}$, $b = b + 1$, go to Step 8, otherwise, go to Step 10.

Step 10 If $a \leq \hat{m} - 1$, $a = a + 1$, go to Step 7, otherwise, go to Step 11.

Step 11 Output the final result.

is in non-decreasing order, i.e.,

$$\frac{y_1 - 1}{x_1 y_1} \leq \frac{y_2 - 1}{x_2 y_2} \leq \cdots \leq \frac{y_m - 1}{x_m y_m}.$$

Proof: Suppose that

$$\pi : \{x_1, \ldots, x_i, x_j, \ldots, x_m\}, \{y_1, \ldots, y_i, y_j, \ldots, y_m\},$$

$$\pi' : \{x_1, \ldots, x_j, x_i, \ldots, x_m\}, \{y_1, \ldots, y_j, y_i, \ldots, y_m\},$$

where $y_i > 1$ $(i = 1, 2, \ldots, m)$. Then we have

$$\begin{aligned}
&\sum_{i=1}^{m} x_i \prod_{j=i}^{m} y_j(\pi) - \sum_{i=1}^{m} x_i \prod_{j=i}^{m} y_j(\pi') \\
&= (x_i y_1 \cdots y_i + x_j y_1 \cdots y_i y_j) - (x_j y_1 \cdots y_j + x_i y_1 \cdots y_j y_i) \\
&= y_1 \cdots (x_j y_j (y_i - 1)) - y_1 \cdots (x_i y_i (y_j - 1)) \\
&= y_1 \cdots (x_i y_i)(x_j y_j)\left(\frac{y_i - 1}{x_i y_i} - \frac{y_j - 1}{x_j y_j}\right).
\end{aligned}$$

Assume that

$$\frac{y_i - 1}{x_i y_i} \leq \frac{y_j - 1}{x_j y_j},$$

hence, we have

$$\sum_{i=1}^{m} x_i \prod_{j=i}^{m} y_j(\pi) \le \sum_{i=1}^{m} x_i \prod_{j=i}^{m} y_j(\pi').$$

We conclude that an optimal schedule can be obtained by sequencing in nondecreasing order of $\frac{y_i-1}{x_i y_i}$. □

Suppose that the job sequence within each group has been determined and the first k groups have been processed and the completion time of the kth group is $\hat{C}_{[k][n_k]}$, we use the notions $[i]$ and $\langle i\rangle$ to denote the scheduled and unscheduled job or group, and then, we have

$$\hat{C}_{\langle k+1\rangle[1]} \ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[1]}\right) - \frac{\lambda}{\mu}$$

$$\vdots$$

$$\hat{C}_{\langle k+1\rangle\hat{n}_{\langle k+1\rangle}} \ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+1\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[q]}\right) - \frac{\lambda}{\mu}$$

$$\hat{C}_{\langle k+2\rangle[1]} \ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+1\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[q]}\right)\left(1+\mu\hat{\beta}_{\langle k+2\rangle}\right)\left(1 + \mu\hat{\alpha}_{\langle k+2\rangle[1]}\right) - \frac{\lambda}{\mu}$$

$$\vdots$$

$$\hat{C}_{\langle k+2\rangle\hat{n}_{\langle k+2\rangle}} \ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+1\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[q]}\right)\left(1 + \mu\hat{\beta}_{\langle k+2\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+2\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+2\rangle[q]}\right) - \frac{\lambda}{\mu}$$

$$\vdots$$

$$\begin{aligned}\hat{C}_{\langle i\rangle[j]} &\ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\frac{\prod_{p=k+1}^{i}\left(1+\mu\hat{\beta}_{\langle p\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle p\rangle}}\left(1+\mu\hat{\alpha}_{\langle p\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right) - \frac{\lambda}{\mu}\\ &= \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\prod_{p=k+1}^{i}\left(1+\mu\hat{\beta}_{\langle p\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle p\rangle}}\left(1+\mu\hat{\alpha}_{\langle p\rangle[q]}\right)\frac{\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)} - \frac{\lambda}{\mu}\end{aligned}$$

$$\vdots$$

$$\hat{C}_{\langle\hat{m}\rangle\hat{n}_{\langle\hat{m}\rangle}} \ge \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right)\prod_{p=k+1}^{\hat{m}}\left(1+\mu\hat{\beta}_{\langle p\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle p\rangle}}\left(1+\mu\hat{\alpha}_{\langle p\rangle[q]}\right) - \frac{\lambda}{\mu}.$$

Hence, we have

$$TCT = \sum_{i=1}^{k}\sum_{j=1}^{\hat{n}_{[i]}}\hat{C}_{[i][j]} + \sum_{i=k+1}^{\hat{m}}\sum_{j=1}^{\hat{n}_{\langle i\rangle}}\hat{C}_{\langle i\rangle[j]}$$

$$\geq \sum_{i=1}^{k}\sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} + \sum_{i=k+1}^{\hat{m}}\sum_{j=1}^{\hat{n}_{\langle i\rangle}} \left(\left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right) \prod_{p=k+1}^{i} \left(1+\mu\hat{\beta}_{\langle p\rangle}\right) \prod_{q=1}^{\hat{n}_{\langle p\rangle}} \left(1 + \mu\hat{\alpha}_{\langle p\rangle[q]}\right) \frac{\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)} - \frac{\lambda}{\mu}\right)$$

$$= \sum_{i=1}^{k}\sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} - \frac{\lambda}{\mu}\sum_{i=k+1}^{\hat{m}} n_{\langle i\rangle} + \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right) \sum_{i=k+1}^{\hat{m}} \left(\prod_{p=k+1}^{i} \left(1+\mu\hat{\beta}_{\langle p\rangle}\right) \prod_{q=1}^{\hat{n}_{\langle p\rangle}} \left(1 + \mu\hat{\alpha}_{\langle p\rangle[q]}\right) \frac{\sum_{j=1}^{\hat{n}_{\langle i\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}\right).$$

It is obvious that $\sum_{i=1}^{k}\sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} - \frac{\lambda}{\mu}\sum_{i=k+1}^{\hat{m}} n_{\langle i\rangle}$ and $\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}$ are constants (k is a given number), and according to Lemma 3.2, the term

$$\sum_{i=k+1}^{\hat{m}} \left(\prod_{p=k+1}^{i} \left(1+\mu\hat{\beta}_{\langle p\rangle}\right) \prod_{q=1}^{\hat{n}_{\langle p\rangle}} \left(1 + \mu\hat{\alpha}_{\langle p\rangle[q]}\right) \frac{\sum_{j=1}^{\hat{n}_{\langle i\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}\right)$$

is minimized by sequencing the unscheduled groups in non-decreasing order of

$$\frac{\left(1+\mu\hat{\beta}_i\right)\prod_{q=1}^{\hat{n}_i}\left(1+\mu\hat{\alpha}_{i[q]}\right) - 1}{\left(1+\mu\hat{\beta}_i\right)\sum_{j=1}^{\hat{n}_i}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{i[q]}\right)} \quad (i = k+1, k+2, \ldots, m),$$

hence, the first lower bound can be obtained as follows:

$$LB_1 = \sum_{i=1}^{k}\sum_{j=1}^{\hat{n}_i} \hat{C}_{[i][j]} - \frac{\lambda}{\mu}\sum_{i=k+1}^{\hat{m}} n_{\langle i\rangle} + \left(\hat{C}_{[k][\hat{n}_k]} + \frac{\lambda}{\mu}\right) \times \sum_{i=k+1}^{\hat{m}} \left(\prod_{p=k+1}^{i} \left(1+\mu\hat{\beta}_{\langle p\rangle}\right) \prod_{q=1}^{\hat{n}_{\langle p\rangle}} \left(1 + \mu\hat{\alpha}_{\langle p\rangle q}\right) \frac{\sum_{j=1}^{\hat{n}_{\langle i\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}{\prod_{q=1}^{\hat{n}_{\langle i\rangle}}\left(1+\mu\hat{\alpha}_{\langle i\rangle[q]}\right)}\right). \quad (5)$$

According to Lemma 3.2, all the unscheduled groups satisfy

$$\frac{\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+1\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[q]}\right) - 1}{\left(1+\mu\hat{\beta}_{\langle k+1\rangle}\right)\sum_{j=1}^{\hat{n}_{\langle k+1\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle k+1\rangle[q]}\right)}$$

$$\leq \frac{\left(1+\mu\hat{\beta}_{\langle k+2\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle k+2\rangle}}\left(1+\mu\hat{\alpha}_{\langle k+2\rangle[q]}\right) - 1}{\left(1+\mu\hat{\beta}_{\langle k+2\rangle}\right)\sum_{j=1}^{\hat{n}_{\langle k+2\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle k+2\rangle[q]}\right)}$$

$$\vdots$$

$$\leq \frac{\left(1+\mu\hat{\beta}_{\langle \hat{m}\rangle}\right)\prod_{q=1}^{\hat{n}_{\langle \hat{m}\rangle}}\left(1+\mu\hat{\alpha}_{\langle \hat{m}\rangle[q]}\right) - 1}{\left(1+\mu\hat{\beta}_{\langle \hat{m}\rangle}\right)\sum_{j=1}^{\hat{n}_{\langle \hat{m}\rangle}}\prod_{q=1}^{j}\left(1+\mu\hat{\alpha}_{\langle \hat{m}\rangle[q]}\right)}.$$

However, if the ready times are larger, then this lower bound may not be tight. To overcome this situation, it is necessary to take the ready times into consideration. For convenience, we introduce R_{ij}:

$$R_{ij} = \max_{1 \le p \le j} \left\{ \left(\hat{r}_{ip} + \frac{\lambda}{\mu} \right) \prod_{l=p}^{j} (1 + \mu \hat{\alpha}_{il}) \right\} \quad (p = 1, 2, \ldots, \hat{n}_i),$$

then, we have

$$\hat{C}_{\langle k+1 \rangle [1]} \ge R_{\langle k+1 \rangle [1]}, \ldots, \hat{C}_{\langle i \rangle [j]} \ge R_{\langle i \rangle [j]}, \ldots, \hat{C}_{\langle \hat{m} \rangle \hat{n}_{[\hat{m}]}} \ge R_{\langle \hat{m} \rangle \hat{n}_{[\hat{m}]}}$$

and

$$TCT = \sum_{i=1}^{k} \sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} + \sum_{i=k+1}^{\hat{m}} \sum_{j=1}^{\hat{n}_{\langle i \rangle}} \hat{C}_{\langle i \rangle [j]} \ge \sum_{i=1}^{k} \sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} + \sum_{i=k+1}^{\hat{m}} \sum_{j=1}^{\hat{n}_{\langle i \rangle}} R_{\langle i \rangle [j]}.$$

Hence, the second lower bound is obtained as follows:

$$LB_2 = \sum_{i=1}^{k} \sum_{j=1}^{\hat{n}_{[i]}} \hat{C}_{[i][j]} + \sum_{i=k+1}^{\hat{m}} \sum_{j=1}^{\hat{n}_{\langle i \rangle}} R_{\langle i \rangle [j]}. \tag{6}$$

To make the lower bound tighter, we choose the maximum value of the two lower bounds as the final lower bound, i.e.,

$$LB = \max \{LB_1, LB_2\}. \tag{7}$$

and then a branch-and-bound (B&B) algorithm is established.

Algorithm 3 (B&B)	
Step 1	Use Algorithm 2 to determine an initial solution as the upper bound.
Step 2	Set $k = 1$.
Step 3	In the kth level node, k groups are already scheduled. For the remained $\hat{m}$-k unscheduled groups, calculate the lower bound (see Equation (7)). If the lower bound is larger than or equal to the upper bound, eliminate the node and all the nodes following it in the branch, otherwise, calculate the new upper bound.
Step 4	Continue to search the next level node. If $k \le \hat{m} - 1$, go to Step 3, otherwise, go to Step 5.
Step 5	Output the optimal solution.

Remark 3.2. *The incremental point in B&B algorithm is the upper bound (see Algorithm 2) and lower bound (see Equation (7)).*

3.4. **Computational experiments.** The heuristic algorithm (HA) and branch-and-bound algorithm (B&B) are implemented in C++ language, the testing environment is Visual studio 2022 v17.1.0, and tests are run on a desktop computer with an Intel(R) Core(TM) i5-10500 CPU @ 3.10GHz 3.10 GHz with 8.00 GB of RAM, Windows 10 operating system.

The percentage error of HA is defined as

$$\frac{TCT(HA) - TCT(B\&B)}{TCT(B\&B)} \times 100\%,$$

where $TCT(HA)$ denotes the total completion time of all jobs by using HA and $TCT(B\&B)$ denotes the total completion time of B&B. For the HA and B&B, run 10 randomly generated replicas in each condition and set the maximum CPU time for each instance to 3600 seconds. In order to investigate the impact of different parameters on the accuracy of heuristic algorithm, the following parameters were randomly generated according to

TABLE 2. Results of algorithms for $\lambda = 1$, $\mu = 1$ and $\hat{n} = 200$

			HA-CPU time (s)		B&B-CPU time (s)		Error percentage of HA (%)		Node number of B&B	
$\hat{\alpha}_{ij}$	$\hat{\beta}_i$	$\hat{m}$	mean	max	mean	max	mean	max	mean	max
		9	0.0004	0.001	1.0301	1.315	0.1377336484	0.2490196553	101733.5	130537
$(0, 0.05)$	$(0, 0.05)$	10	0.0001	0.001	9.0792	12.283	0.1589781533	0.3306463751	867393.3	1161932
		11	0.0001	0.001	59.2151	114.056	0.1242984340	0.1973314696	5659332.3	9563383
		9	0.0002	0.001	1.0131	2.199	0.7081816456	1.4358591210	97173.3	201231
$(0, 0.5)$	$(0, 0.05)$	10	0.0001	0.001	7.5429	24.259	0.8042885930	3.5343885240	718155.6	2292881
		11	0.0001	0.001	69.8512	152.393	0.6204571306	2.1719996270	6794183.7	13955704
		9	0.0001	0.001	0.6631	1.697	0.1167589949	0.2369952011	65735.2	176974
$(0, 0.05)$	$(0, 0.5)$	10	0.0001	0.001	4.7900	5.635	0.1280656881	0.2582161506	452089.5	578839
		11	0.0002	0.001	30.2638	61.173	0.2028386595	0.4199289857	2471944.8	5829476
		9	0.0001	0.001	0.9350	1.690	0.6924522908	1.6448111760	93069.1	167638
$(0, 0.5)$	$(0, 0.5)$	10	0.0002	0.001	8.3903	11.717	0.4377065434	0.8758096449	781294.4	1075701
		11	0.0002	0.001	89.0065	138.563	0.4415966914	1.0544445120	8392369.1	12470603

TABLE 3. Results of algorithms for $\hat{\alpha}_{ij} \in (0, 0.05)$, $\hat{\beta}_i \in (0, 0.05)$ and $\hat{n} = 200$

			HA-CPU time (s)		B&B-CPU time (s)		Error percentage of HA (%)		Node number of B&B	
λ	μ	$\hat{m}$	mean	max	mean	max	mean	max	mean	max
		9	0.0004	0.001	1.0301	1.315	0.1377336484	0.2490196553	101733.5	130537
1	1	10	0.0001	0.001	9.0792	12.283	0.1589781533	0.3306463751	867393.3	1161932
		11	0.0001	0.001	59.2151	114.056	0.1242984340	0.1973314696	5659332.3	9563383
		9	0.0001	0.001	0.4399	0.771	0.0052162055	0.0255437543	43612.3	73206
1	0.01	10	0.0004	0.001	6.7169	8.816	0.0112466061	0.0382105719	668007.1	916848
		11	0.0003	0.001	65.0899	114.274	0.0149275020	0.0432260984	6141125.1	10930308
		9	0.0003	0.001	0.3448	0.999	0.0040174995	0.0062809500	36323.8	103555
0.1	0.01	10	0.0000	0.000	6.2638	11.166	0.0047251900	0.0256720537	640090.3	1155319
		11	0.0002	0.001	55.6383	81.868	0.0055967740	0.0226271690	5423492.3	8223277
		9	0.0002	0.001	0.5451	0.623	0.0016134251	0.0058029788	59490.4	69286
0.01	0.01	10	0.0002	0.001	4.6498	6.509	0.0163961246	0.0588968659	502346.5	709449
		11	0.0003	0.001	36.4580	62.302	0.0067358473	0.0347417966	3860534.9	6858833

Liu et al. [21]. For the parameters $\hat{\alpha}_{ij}$ and $\hat{\beta}_i$, it is assumed that $\lambda = 1$, $\mu = 1$, $t_0 = 1$, and $\hat{r}_{ij}$ are uniformly distributed over $(1, 100)$. From Table 2, we can summarize that the parameter $\hat{\alpha}_{ij}$ influences more than the parameter $\hat{\beta}_i$ on results by comparing the errors in Table 2.

As for the parameters λ and μ, we suppose that $t_0 = 1$, $\hat{r}_{ij} \in (1, 100)$, $\hat{\alpha}_{ij} \in (0, 0.05)$ and $\hat{\beta}_i \in (0, 0.05)$. From Table 3, we can conclude that the parameter μ has a bigger impact on the result. From Tables 2 and 3, we can obtain that the heuristic algorithm is suitable for the problem and the average error percentage is less than 1%.

4. **Conclusions.** In this paper, the group problem with ready times and deteriorating jobs is considered. The objective is to determine the optimal job sequence within each group and the optimal group sequence to minimize the total completion time of all jobs. For a special case (i.e., $\hat{r}_{ij} = \hat{r}$), the problem can be solved in polynomial time. For the

general case, the problem considered is NP-hard, and then, a heuristic algorithm and a branch-and-bound (B&B) algorithm are proposed to deal with the problem. Future research may focus on the group scheduling with ready times and rejection-jobs, two-agent group scheduling with ready times and deteriorating jobs (Zhang et al. [24]). In addition, the problem in this paper can be solved by applying fuzzy TOPSIS-CRITIC method (Wang et al. [25]) or BWM-CRITIC approach (Wang et al. [26]).

REFERENCES

[1] J. Blazewicz, K. Ecker, E. Pesch, G. Schmidt and J. Weglarz, *Handbook on Scheduling*, Springer International Publishing, Cham, 2019.

[2] X. Y. Sun and X. N. Geng, Single-machine scheduling with deteriorating effects and machine maintenance, *International Journal of Production Research*, vol.57, no.10, pp.3186-3199, 2019.

[3] S. Gawiejnowicz, *Models and Algorithms of Time-Dependent Scheduling*, Springer, Berlin, 2020.

[4] X. Y. Sun, T. Liu, X. N. Geng, Y. Hu and J. X. Xu, Optimization of scheduling problems with deterioration effects and an optional maintenance activity, *Journal of Scheduling*, vol.26, pp.251-266, 2023.

[5] J. Pei, Y. Zhou, P. Yan and P. M. Pardalos, A concise guide to scheduling with learning and deteriorating effects, *International Journal of Production Research*, vol.61, no.6, pp.2010-2031, 2023.

[6] C. C. Wu and W. C. Lee, Single-machine group-scheduling problems with deteriorating setup times and job-processing times, *International Journal of Production Economics*, vol.115, no.1, pp.128-133, 2008.

[7] J. B. Wang, W. J. Gao, L. Y. Wang and D. Wang, Single machine group scheduling with general linear deterioration to minimize the makespan, *The International Journal of Advanced Manufacturing Technology*, vol.43, pp.146-150, 2009.

[8] T. C. E. Cheng, W. C. Lee and C. C. Wu, Single-machine scheduling with deteriorating functions for job processing times, *Applied Mathematical Modelling*, vol.34, no.12, pp.4171-4178, 2010.

[9] T. C. E. Cheng, W. C. Lee and C. C. Wu, Single-machine scheduling with deteriorating jobs and past-sequence-dependent setup times, *Applied Mathematical Modelling*, vol.35, no.4, pp.1861-1867, 2011.

[10] C. L. Zhao, An FPTAS for the weighted number of tardy jobs minimization on a single machine with deteriorating jobs, *Journal of Industrial and Management Optimization*, vol.13, pp.587-592, 2017.

[11] T. C. E. Cheng, C. J. Hsu, Y. C. Huang and W. C. Lee, Single-machine scheduling with deteriorating jobs and setup times to minimize the maximum tardiness, *Computers & Operations Research*, vol.38, no.12, pp.1760-1765, 2011.

[12] X. R. Wang and J. J. Wang, Single-machine scheduling with convex resource dependent processing times and deteriorating jobs, *Applied Mathematical Modelling*, vol.37, no.4, pp.2388-2393, 2013.

[13] J. B. Wang, D. Y. Lv, S. Y. Wang and C. Jiang, Resource allocation scheduling with deteriorating jobs and position-dependent workloads, *Journal of Industrial and Management Optimization*, vol.19, no.3, pp.1658-1669, 2023.

[14] L. Liu, Y. T. Xu, N. Yin and J. B. Wang, Single machine group scheduling problem with deteriorating jobs and release dates, *Applied Mechanics and Materials*, vol.513, pp.2145-2148, 2014.

[15] J. B. Wang, X. Huang, Y. B. Wu and P. Ji, Group scheduling with independent setup times, ready times, and deteriorating job processing times, *The International Journal of Advanced Manufacturing Technology*, vol.60, pp.643-649, 2012.

[16] Y. Y. Lu, J. J. Wang and J. B. Wang, Single machine group scheduling with decreasing time-dependent processing times subject to release dates, *Applied Mathematics and Computation*, vol.234, pp.286-292, 2014.

[17] J. B. Wang, L. Liu, J. J. Wang and L. Li, Makespan minimization scheduling with ready times, group technology and shortening job processing times, *The Computer Journal*, vol.61, no.9, pp.1422-1428, 2018.

[18] J. B. Wang, L. Lin and F. Shan, Single-machine group scheduling problems with deteriorating jobs, *The International Journal of Advanced Manufacturing Technology*, vol.39, pp.808-812, 2008.

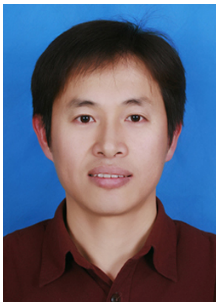
[19] J. B. Wang and J. J. Wang, Single machine group scheduling with time dependent processing times and ready times, *Information Sciences*, vol.275, pp.226-231, 2014.

[20] Y. T. Xu, Y. Zhang and X. Huang, Single-machine ready times scheduling with group technology and proportional linear deterioration, *Applied Mathematical Modelling*, vol.38, no.1, pp.384-391, 2014.

[21] F. Liu, J. Yang and Y. Y. Lu, Solution algorithms for single-machine group scheduling with ready times and deteriorating jobs, *Engineering Optimization*, vol.51, no.5, pp.862-874, 2019.
[22] R. L. Graham, E. L. Lawler, J. K. Lenstra and A. H. G. R. Kan, Optimization and approximation in deterministic sequencing and scheduling: A survey, *Annals of Discrete Mathematics*, vol.5, pp.287-326, 1979.
[23] J. K. Lenstra, A. H. G. R. Kan and P. Brucker, Complexity of machine scheduling problems, *Annals of Discrete Mathematics*, vol.1, pp.343-362, 1977.
[24] L. H. Zhang, D. Y. Lv and J. B. Wang, Two-agent slack due-date assignment scheduling with resource allocations and deteriorating jobs, *Mathematics*, vol.11, 2737, 2023.
[25] P. Wang, Y. Lin and Z. Wang, An integrated multi criteria group decision-making model applying fuzzy TOPSIS-CRITIC method with unknown weight information, *International Journal of Innovative Computing, Information and Control*, vol.18, no.3, pp.815-836, 2022.
[26] P. Wang, Y. Lin and Z. Wang, An integrated BWM-CRITIC approach based on neutrosophic set for sustainable supply chain finance risk evaluation, *International Journal of Innovative Computing, Information and Control*, vol.18, no.6, pp.1735-1754, 2022.

Author Biography

Hongbin Bei is an M.S. graduate student in Shenyang Aerospace University in China. He received his B.S. degree from Northeastern University (China). His research direction is the operations research. His publications include papers in international journals such as Journal of Industrial and Management Optimization, and Mathematics.



Jibo Wang is a professor at the Shenyang Aerospace University in China. He received his B.S. and M.S. degrees from Shenyang Normal University (China), and Ph.D. degree from Dalian University of Technology (China). His research interests are in operations management and operations research. His publications include articles in international journals such as European Journal of Operational Research, Journal of the Operational Research Society, Computers and Operations Research, and Omega.