mathematics

Article

Branch-and-Bound and Heuristic Algorithms for Group Scheduling with Due-Date Assignment and Resource Allocation

Hongyu He [1,2], Yanzhi Zhao [1,*], Xiaojun Ma [1,3], Zheng-Guo Lv [4] and Ji-Bo Wang [4]

[1] School of Economics, Shenyang University, Shenyang 110096, China; hehongyu@syu.edu.cn (H.H.); maxiaojun@dufe.edu.cn (X.M.)
[2] Institute of Carbon Neutrality Technology and Policy, Shenyang University, Shenyang 110044, China
[3] School of Statistics, Dongbei University of Finance and Economics, Dalian 116025, China
[4] School of Science, Shenyang Aerospace University, Shenyang 110136, China; lvzhengguo@stu.sau.edu.cn (Z.-G.L.); wangjibo@sau.edu.cn (J.-B.W.)
* Correspondence: zyzhi@syu.edu.cn

Abstract: Green scheduling that aims to enhance efficiency by optimizing resource allocation and job sequencing concurrently has gained growing academic attention. To tackle such problems with the consideration of scheduling and resource allocation, this paper considers a single-machine group scheduling problem with common/slack due-date assignment and a controllable processing time. The objective is to decide the optimized schedule of the group/job sequence, resource allocation, and due-date assignment. To solve the generalized case, this paper proves several optimal properties and presents a branch-and-bound algorithm and heuristic algorithms. Numerical experiments show that the branch-and-bound algorithm is efficient and the heuristic algorithm developed based on the analytical properties outruns the tabu search.

Keywords: scheduling; single machine; resource allocation; group technology; due-date assignment

MSC: 90B35

check for updates

Citation: He, H.; Zhao, Y.; Ma, X.; Lv, Z.-G.; Wang, J.-B. Branch-and-Bound and Heuristic Algorithms for Group Scheduling with Due-Date Assignment and Resource Allocation. *Mathematics* **2023**, *11*, 4745. https://doi.org/10.3390/math11234745

Academic Editors: Fajun Yang and Chunjiang Zhang

Received: 25 October 2023
Revised: 17 November 2023
Accepted: 21 November 2023
Published: 23 November 2023

1. Introduction

Due to the reflection on the balance between resource allocation costs and efficiency, the scheduling problem with a controllable processing time (CPSP) has received a considerable amount of attention. In contrast to the conventional scheduling problem with a constant processing time, the controllable processing time varies according to the allocated resources, especially those represented by energy. Since the essential objective of the green scheduling problem (GSP) was to maximize the environmental benefits by deciding the energy usage allocation and schedule, the CPSP could be extended to deal with the GSP (Foumani and Smith [1]). Early research on the CPSP introduced the idea that the processing time often varied and could be reduced with the cost of more allocations of production resources (Shabtay and Steiner [2]; Manier and Bloch [3]; Kuntay et al. [4]). Uruk et al. [5] considered flexible operations and resource allocation in a two-machine flowshop environment. Mor and Mosheiov [6] integrated batch scheduling into the CPSP. The jobs inside a certain batch were modeled as identical jobs. For the CPSP with scenario-based demands, Akhoondi and Lotfi [7] developed heuristic algorithms for master production scheduling. Li and Wang [8] combined the deteriorating effects with the CPSP. The problem of minimizing the weighted sum of the makespan and resource cost was proved to be polynomially solvable. The CPSP with learning effects was considered by Sun et al. [9]. For a two-machine flowshop CPSP with common due-date assignment and no-wait constraints, they proved that the proposed problem could be solved in polynomial time. Sun et al. [9]

studied the CPSP with slack due-date assignment and no-wait constraints and proved that the irregular objective can be polynomially solvable.

Group technology and due-date assignment are concurrently integrated into the CPSP, which is widely reflected in real manufacturing environments and attracts an increasing amount of academic attention (Shabtay et al. [10]; Zhu et al. [11]). For green scheduling with the consideration of carbon-emission supervision and reduction, the group of jobs is usually divided according to periodical carbon-emission demand and endowed with different workloads. In terms of group scheduling, Webster and Baker [12] were among the pioneers that introduced the idea of group technology to the single-machine scheduling problem. Li et al. [13] integrated the due-date assignment problem into the group scheduling problem and proved that irregular minimization can be solved in polynomial time. Liu et al. [14] considered a single-machine group scheduling problem with deteriorating effects and developed composite solution methods for the objective of makespan minimization. The due-date assignment problem is usually proposed to integrate with group scheduling and was recently covered by Yang et al. [15] and Yin et al. [16–19]. As for scheduling with the combined considerations of the controllable processing time and group technology, Shabtay et al. [10] dealt with the single-machine CPSP with group technology and due-date assignment. Yan et al. [20] studied the integrated problem with learning effects and a total resource limitation. For the special cases, the problem was proved to be polynomially solvable. Liu and Wang [21] considered a new group scheduling model with due-date assignment and a controllable processing time. They proved the special cases where the job numbers of different groups were identical were polynomially solvable.

In light of the significance of the CPSP with group technology in real manufacturing environments, this paper continues the study of an integrated model of group scheduling with a controllable processing time under CON/slack due-date assignment (Liu and Wang [21]) and extends the work to a general case where the job numbers of different groups are variable. This paper considers an integrated solution method to tackle the open problem of Liu and Wang [21]. The objective is to minimize the weighted earliness and tardiness. Our main contributions include the following: (1) the incorporation of the generality of the integrated CPSP model; (2) the proposition of optimal properties; (3) the analysis of the lower bound strategy; and (4) composite solution algorithms to solve the general case of the problem.

The remainder of this paper is organized as follows. Section 2 makes notations and assumptions. Section 3 presents several preliminary properties and shows the lower bound analysis for general cases. Section 4 proposes the solution algorithms of an exact method and heuristics. Section 5 displays numerical experiment results. Finally, Section 6 makes the summary.

2. Problem Formulation

In this section, notations used throughout this whole paper will first be introduced as follows (see Abbreviation section).

Investigate a set of $\ddot{O}$ jobs grouped into Q groups $F_1, F_2, \ldots, F_Q$. All the jobs are to be processed on a single machine and are available at time zero. Let the number of jobs in F_g be O_g; it follows that $O_1 + O_2 + \ldots + O_Q = \ddot{O}$. A setup time s_g has to be required before the processing of jobs in group F_g. Let $G_{g,k}$ denote the kth job in F_g, where $g = 1, 2, \ldots, Q; k = 1, 2, \ldots, O_g$. As in Liu and Wang [21], the processing time of $G_{g,k}$ is

$$p^A_{g,k} = \left(\frac{\varpi_{g,k}}{u_{g,k}} \right)^{\sigma}, \tag{1}$$

where $\varpi_{g,k}$ is the workload of $G_{g,k}$, $\sigma > 0$ is a constant, and $u_{g,k}$ is the amount of resources allocated to $G_{g,k}$.

Let $E_{g,k} = \max\{0, d_{g,k} - C_{g,k}\}$ (resp. $T_{g,k} = \max\{0, C_{g,k} - d_{g,k}\}$) be the earliness (resp. tardiness) of $G_{g,k}$ in F_g, where $d_{g,k}$ (resp. $C_{g,k}$) is the due date (resp. completion time) of $G_{g,k}$. Under the CON assignment, it is assumed that $d_{g,k} = d_g(g = 1, \ldots, Q, k = k =$

$1,2,\ldots,O_g)$, where d_g is a decision variable. Under the SLK assignment, it is assumed that $d_{g,k} = p^A_{g,k} + q_g(g = 1,\ldots,Q, k = k = 1,2,\ldots,O_g)$, where q_g denotes the common flow allowance in group F_g and q_g is a decision variable. Denote $[z]$ as some job (or group) scheduled in the zth position; the objective is to determine a group schedule χ and an internal job schedule ψ_g within F_g, a set of $\mathbf{d} = \{d_g|g = 1,\ldots,Q\}$ ($\mathbf{q} = \{q_g|g = 1,\ldots,Q\}$) and a set of $\mathbf{u} = \{u_{g,k}|g = 1,\ldots,Q; k = 1,\ldots,O_g\}$, such that the optimization objective

$$\widetilde{OF}(CON) = \sum_{g=1}^{Q}\sum_{k=1}^{O_g}(\alpha_{g,k}E_{g,[k]} + \beta_{g,k}T_{g,[k]} + \xi d_g) + \sum_{g=1}^{Q}\sum_{k=1}^{O_g} v_{g,k}u_{g,k} \tag{2}$$

$$\widetilde{OF}(SLK) = \sum_{g=1}^{Q}\sum_{k=1}^{O_g}(\alpha_{g,k}E_{g,[k]} + \beta_{g,k}T_{g,[k]} + \xi q_g) + \sum_{g=1}^{Q}\sum_{k=1}^{O_g} v_{g,k}u_{g,k} \tag{3}$$

is minimized, where $\alpha_{g,k}$ (resp. $\beta_{g,k}$) is a position-dependent weight for the earliness (resp. tardiness) cost, $v_{g,k}$ is the unit consumption cost, and $\xi \geq 0$ is a given constant. As in Liu and Wang [21], the problem can be denoted by

$$1\left|X, p^A_{g,k} = \left(\frac{\omega_{g,k}}{u_{g,k}}\right)^{\sigma}, \widetilde{GT}\right|\widetilde{OF}(X) \tag{4}$$

where $X \in \{CON, SLK\}$ and $\widetilde{GT}$ denotes group technology. For a special case, i.e., where the number of jobs in F_g is identical, Liu and Wang [21] proved that the problem can be solved in $O(\ddot{O}^3)$ time. This paper will consider how to solve the general problem $1\left|X, p^A_{g,k} = \left(\frac{\omega_{g,k}}{u_{g,k}}\right)^{\sigma}, \widetilde{GT}\right| \widetilde{OF}(X)$ $(X \in \{CON, SLK\})$.

3. Preliminary Properties

From Liu and Wang [21], the following results are given:

Lemma 1 (Lemma 2, Liu and Wang [21])**.** *For a given job sequence ψ_g within F_g $(g = 1,\ldots,Q)$, under a CON (resp. SLK) assignment, there exists an optimal $d_g = C_{g,[h_g]}$ (resp. $q_g = C_{g,[h_g-1]}$) where h_g satisfies the following inequality: $\sum_{l=h_g+1}^{O_g}\beta_{g,l} - \sum_{l=1}^{h_g}\alpha_{g,l} \leq \xi O_g \leq \sum_{l=h_g}^{O_g}\beta_{g,l} - \sum_{l=1}^{h_g-1}\alpha_{g,l}$.*

Lemma 2 (Lemma 6, Liu and Wang [21])**.** *Under the given group and job sequences*

$$1\left|X, p^A_{g,k} = \left(\frac{\omega_{g,k}}{u_{g,k}}\right)^{\sigma}, \widetilde{GT}\right|\widetilde{OF}(X),$$

the optimal resource allocation $\mathbf{u}^*(\chi, \psi_g|g = 1,\ldots,Q)$ is

$$u^*_{[g],[k]} = \left(\frac{\sigma\left(B_{[g]k} + \xi\sum_{r=g+1}^{Q}O_{[r]}\right)}{v_{[g],[k]}}\right)^{\frac{1}{\sigma+1}} \times \left(\omega_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}, \tag{5}$$

where, for the CON assignment,

$$B_{[g],k} = \begin{cases} \sum_{l=1}^{k-1}\alpha_{[g],l} + \xi O_{[g]}, & k = 1,2,\ldots,h_g, \\ \sum_{l=k}^{O_g}\beta_{[g],l}, & k = h_g+1, h_g+2,\ldots,O_g, \end{cases} \tag{6}$$

and for the SLK assignment,

$$B_{[g],k} = \begin{cases} \sum_{l=1}^{k} \alpha_{[g],l} + \xi O_{[g]}, & k = 1, 2, \ldots, h_g - 1, \\ \sum_{l=k+1}^{O_g} \beta_{[g],l}, & k = h_g, h_g + 1, \ldots, O_g - 1, \\ 0, & k = O_g. \end{cases} \tag{7}$$

As in Liu and Wang [21], it follows that

$$\begin{aligned} & \widetilde{OF}(X)(\chi, \psi_g | g = 1, \ldots, Q, \mathbf{u}^*) \\ = & \left(\sigma^{\frac{-\sigma}{\sigma+1}} + \sigma^{\frac{1}{\sigma+1}}\right) \sum_{g=1}^{Q} \sum_{k=1}^{O_{[g]}} \left(B_{[g],k} + \xi \sum_{z=g+1}^{Q} O_{[z]}\right)^{\frac{1}{\sigma+1}} \left(v_{[g],[k]} \varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}} \\ & + \xi \sum_{g=1}^{Q} \left(O_{[g]} \times \sum_{z=1}^{g} s_{[z]}\right) \end{aligned} \tag{8}$$

Lemma 3 (Lemma 7, Liu and Wang [21])**.** *Given group order χ, the optimal job sequence ψ_g^* ($g = 1, \ldots, Q$) within F_g can be obtained by matching the smallest (resp. second smallest) $B_{g,k}$ to the job with the largest (resp. second largest) $v_{g,k}\varpi_{g,k}$, and so on, where, for the CON assignment,*

$$B_{g,k} = \begin{cases} \sum_{l=1}^{k-1} \alpha_{g,l} + \xi O_g, & k = 1, 2, \ldots, h_g, \\ \sum_{l=k}^{O_g} \beta_{g,l}, & k = h_g + 1, h_g + 2, \ldots, O_g, \end{cases} \tag{9}$$

and for the SLK assignment,

$$B_{g,k} = \begin{cases} \sum_{l=1}^{k} \alpha_{g,l} + \xi O_{[g]}, & k = 1, 2, \ldots, h_g - 1, \\ \sum_{l=k+1}^{O_g} \beta_{g,l}, & k = h_g, h_g + 1, \ldots, O_g - 1, \\ 0, & k = O_g. \end{cases} \tag{10}$$

By using the group interchanging technique, the following results can be obtained.

Lemma 4. *The term $\sum_{g=1}^{Q}\left(O_{[g]} \times \sum_{z=1}^{g} s_{[z]}\right)$ is minimized if $\frac{O_{[1]}}{s_{[1]}} \geq \frac{O_{[2]}}{s_{[2]}} \geq \ldots \geq \frac{O_{[Q]}}{s_{[Q]}}$.*

Proof. This is proved using the group interchanging technique. □

If the optimal job sequence ψ_g^* ($g = 1, \ldots, Q$) within F_g is given (by Lemma 3), the term $\sum_{g=1}^{Q} \sum_{k=1}^{O_{[g]}} \left(B_{[g],k} + \xi \sum_{z=g+1}^{Q} O_{[z]}\right)^{\frac{1}{\sigma+1}} \left(v_{[g],[k]} \varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}$ can not be minimized by the non-increasing (LPT) order of O_g and the non-decreasing (SPT) order of O_g.

Example 1. $Q = 2$, $O_1 = 3, O_2 = 2$, $\sigma = \xi = 1$, $B_{1,1} = 40, B_{1,2} = 42, B_{1,3} = 43$, $B_{2,1} = 40$, $B_{2,2} = 50$, $v_{1,1}\varpi_{1,1} = 23, v_{1,2}\varpi_{1,2} = 29, v_{1,3}\varpi_{1,3} = 13, v_{2,1}\varpi_{2,1} = 13, v_{2,2}\varpi_{2,2} = 13$.

According to Lemma 3, the optimal job sequence within F_1 (resp. F_2) is $G_{1,1} \to G_{1,2} \to G_{1,3}$ (resp. $G_{2,1} \to G_{2,2}$). According to the LPT order of O_g (i.e., $F_1 \to F_2$), it follows that

$$\begin{aligned} & \sum_{g=1}^{Q} \sum_{k=1}^{O_{[g]}} \left(B_{[g],k} + \xi \sum_{z=g+1}^{Q} O_{[z]}\right)^{\frac{1}{\sigma+1}} \left(v_{[g],[k]} \varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}} \\ = & (40+2)^{0.5} * 23^{0.5} + (42+2)^{0.5} * 29^{0.5} + (43+2)^{0.5} * 13^{0.5} + 40^{0.5} * 13^{0.5} + 50^{0.5} * 13^{0.5} \\ = & 139.2871. \end{aligned}$$

If the group order is $F_2 \to F_1$, the following formula can be obtained.

$$\begin{aligned}
&\sum_{g=1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
=\ & (40+3)^{0.5}*13^{0.5}+(50+3)^{0.5}*13^{0.5}+40^{0.5}*23^{0.5}+42^{0.5}*29^{0.5}+43^{0.5}*13^{0.5}\\
=\ & 138.7665.
\end{aligned}$$

Therefore, the LPT order of O_g is not an optimal group schedule.

Example 2. $Q=2$, $O_1=3, O_2=2$, $\sigma=\xi=1$, $B_{1,1}=40, B_{1,2}=42, B_{1,3}=43$, $B_{2,1}=40$, $B_{2,2}=50$, $v_{1,1}\varpi_{1,1}=23, v_{1,2}\varpi_{1,2}=29, v_{1,3}\varpi_{1,3}=13, v_{2,1}\varpi_{2,1}=33, v_{2,2}\varpi_{2,2}=33$.

According to Lemma 3, the optimal job sequence within F_1 (resp. F_2) is $G_{1,1}\to G_{1,2}\to G_{1,3}$ (resp. $G_{2,1}\to G_{2,2}$). According to the SPT order of O_g (i.e., $F_2\to F_1$), it follows that

$$\begin{aligned}
&\sum_{g=1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
=\ & (40+3)^{0.5}*13^{0.5}+(50+3)^{0.5}*13^{0.5}+40^{0.5}*23^{0.5}+42^{0.5}*29^{0.5}+43^{0.5}*13^{0.5}\\
=\ & 168.3652.
\end{aligned}$$

If the group order is $F_1\to F_2$, the function is given as follows:

$$\begin{aligned}
&\sum_{g=1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
=\ & (40+2)^{0.5}*23^{0.5}+(42+2)^{0.5}*29^{0.5}+(43+2)^{0.5}*13^{0.5}+40^{0.5}*33^{0.5}+50^{0.5}*33^{0.5}\\
=\ & 167.9405.
\end{aligned}$$

Therefore, the SPT order of O_g is not an optimal group schedule.

4. Solution Methods

4.1. Lower Bound Analysis

For a special case (i.e., $O_1=O_2=\ldots=O_Q=\overline{O}$), Liu and Wang [21] demonstrated that $1\Big|X, O_g=\overline{O}, p^A_{g,k}=\left(\frac{\varpi_{g,k}}{u_{g,k}}\right)^{\sigma}, \widetilde{GT}\Big|\widetilde{OF}(X)$ can be solved in $O(\ddot{O}^3)$ time. For the general case of

$$1\left|X, p^A_{g,k}=\left(\frac{\varpi_{g,k}}{u_{g,k}}\right)^{\sigma}, \widetilde{GT}\right|\widetilde{OF},$$

the complexity is an open question. To solve the general case of this problem, some heuristic and branch-and-bound (B and B) algorithms will be proposed.

From Lemma 3 (Lemma 7, Liu and Wang [21]), the optimal job sequence ψ^*_g (g = $1,\ldots,Q$) within F_g can be obtained. Let $\chi=(\chi^s,\chi^u)$ be a group sequence, where χ^s (resp. χ^u) is the scheduled (resp. unscheduled) part, and suppose there are r groups in χ^s. From Equation (8), the following formula can be obtained.

$$
\begin{aligned}
&\widetilde{OF}(X)(\chi^s,\chi^u)\\
=\;& \left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=1}^{r}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
&+\left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=r+1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
&+\xi\sum_{g=1}^{r}\left(O_{[g]}\times\sum_{z=1}^{g}s_{[z]}\right)+\xi\sum_{g=r+1}^{Q}\left(O_{[g]}\times\left(\sum_{z=1}^{r}s_{[z]}+\sum_{z=r+1}^{g}s_{[z]}\right)\right)
\end{aligned}
\tag{11}
$$

From Equation (11), the terms $\xi\sum_{g=1}^{r}\left(O_{[g]}\times\sum_{z=1}^{g}s_{[z]}\right)$, $\sum_{z=1}^{r}s_{[z]}$, and $\left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=1}^{r}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\times\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}$ are constants. $\xi\sum_{g=r+1}^{Q}\left(O_{[g]}\times\left(\sum_{z=1}^{r}s_{[z]}+\sum_{z=r+1}^{g}s_{[z]}\right)\right)$ can be minimized by Lemma 4. Then the following inequality can be obtained: $\left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=r+1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\geq\left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=r+1}^{Q}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}$. Hence, the lower bound is given as follows:

$$
\begin{aligned}
&LB(\widetilde{OF})\\
=\;& \left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=1}^{r}\sum_{k=1}^{O_{[g]}}\left(B_{[g],k}+\xi\sum_{z=g+1}^{Q}O_{[z]}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
&+\left(\sigma^{\frac{-\sigma}{\sigma+1}}+\sigma^{\frac{1}{\sigma+1}}\right)\sum_{g=r+1}^{Q}\sum_{k=1}^{O_{<g>}}\left(B_{[g],k}\right)^{\frac{1}{\sigma+1}}\left(v_{[g],[k]}\varpi_{[g],[k]}\right)^{\frac{\sigma}{\sigma+1}}\\
&+\xi\sum_{g=1}^{r}\left(O_{[g]}\times\sum_{z=1}^{g}s_{[z]}\right)+\xi\sum_{g=r+1}^{Q}\left(O_{<g>}\times\left(\sum_{z=1}^{r}s_{[z]}+\sum_{z=r+1}^{g}s_{<z>}\right)\right)
\end{aligned}
\tag{12}
$$

where $\frac{O_{<r+1>}}{s_{<r+1>}}\geq\frac{O_{<r+2>}}{s_{<r+2>}}\geq\ldots\geq\frac{O_{<Q>}}{s_{<Q>}}$.

4.2. Upper Bound Algorithms

Algorithm 1: Upper bound

From the above analysis and Nawaz et al. [22], the following Algorithm 1 can be proposed as an upper bound for $1\left|X,p_{g,k}^{A}=\left(\frac{\varpi_{g,k}}{u_{g,k}}\right)^{\sigma},\widetilde{GT}\right|\widetilde{OF}(X)$.

Phase 1

Step (a1). Sequence groups in non-decreasing order of s_g.

Step (a2). Sequence groups in non-increasing order of $\frac{O_g}{s_g}$.

Step (a3). Sequence groups in non-increasing order of O_g.

Step (a4). Choose the better solution from Steps (a1), (a2), and (a3).

Phase 2

Step (b1). Let χ^0 be the group sequence obtained from Phase 1.

Step (b2). Set $\lambda=2$. Select the first two groups from the sorted list and select the better of the two possible sequences. Do not change the relative positions of these two jobs with respect to each other in the remaining steps of the algorithm. Set $\lambda=3$.

Step (b3). Pick the job in the λth position of the list generated in Step (b1) and find the best group sequence by placing it at all possible λ positions in the partial sequence found in the previous step, without changing the relative positions to each other of the already assigned groups. The number of enumerations at this step equals λ.

Step (b4). If $\lambda = Q$, STOP; otherwise, set $\lambda = \lambda + 1$ and go to Step (b3).

Algorithm 2: Tabu search

As in Noman et al. [23], a tabu search (TS) algorithm is an effective method for the difficult scheduling problems; hence, the TS algorithm incorporating the analytical properties of the CPSP is designed to reach the near-optimal solution of $1\left|X, p^A_{g,k} = \left(\frac{\omega_{g,k}}{u_{g,k}}\right)^\sigma, \widetilde{GT}\right|\widetilde{OF}(X)$. The initial group sequence of the TS algorithm is decided in the non-decreasing order of s_i, and the maximum number of iterations for the TS algorithm is set at $200Q$ (as in Yan et al. [24], in general, the maximum number of iterations is 2000; here, it is set to $200Q$).

Step (1). Let the tabu list be empty and the iteration number be zero.

Step (2). Set the initial group sequence of the TS algorithm, calculate its objective cost (by Equation (8)), and set the current group sequence as the best solution χ^*.

Step (3). Search the associated neighborhood of the current group sequence and resolve if there is a group sequence χ^{**} with the smallest objective cost in the associated neighborhood and it is not in the tabu list.

Step (4). If χ^{**} is better than χ^*, then let $\chi^* = \chi^{**}$. Update the tabu list and the iteration number.

Step (5). If there is not a group sequence in the associated neighborhood but it is not in the tabu list or the maximum number of iterations is reached (i.e., $200Q$), then output the final group sequence. Otherwise, update the tabu list and go to Step (3).

4.3. *Exact Method*

From the lower bound (see Equation (10)) and upper bound (see Algorithm 1), the following branch-and-bound (B and B) algorithm can be proposed to solve $1\left|X, p^A_{g,k} = \left(\frac{\omega_{g,k}}{u_{g,k}}\right)^\sigma, \widetilde{GT}\right|\widetilde{OF}(X)$ optimally.

Algorithm 3: B and B algorithm

Step (1) (find the upper bound). Obtain an initial solution (upper bound) using Algorithm 1.

Step (2) (bounding). Calculate the lower bound (see Equation (10)) for the node. If the lower bound for an unfathomed partial group sequence is larger than or equal to the objective value of the initial solution (see Equation (8)), eliminate the node and all the nodes following it in the branch. Calculate the objective value of the completed group sequence (see Equation (8)). If it is less than the initial solution, replace it as the new solution; otherwise, eliminate it.

Step (3) (termination). Continue until all nodes have been explored.

Remark 1. *For small-sized instances, the B and B algorithm (i.e., Algorithm 3) is an exact algorithm and Algorithms 1 and 2 are heuristic algorithms. For large-sized instances, the B and B algorithm is disabled.*

5. Numerical Study

The algorithms (i.e., Algorithms 1, 2, and 3) were executed in Microsoft Visual Studio Professional 2019 (6.11.24) and carried out on a HUAWEI PC with an Inter core i5-8250U 1.4 GHz CPU and 8.00 GB of RAM. This section considers the *CON* assignment, where ξ=10, and other parameters are given as follows:

(1) $\ddot{O} = 100, 120, 140, 160, 180, 200$;
(2) Q= 10, 12, 14, 16 (each group must contain at least one job);
(3) s_g were drawn from a discrete uniform distribution in [1, 10];
(4) $v_{g,k}\omega_{g,k}$ were drawn from a discrete uniform distribution in [1, 50], [50, 100], and [1, 100];
(5) $\alpha_{g,k}$ and $\beta_{g,k}$ were drawn from a discrete uniform distribution in [1, 50];
(6) $\sigma = 1, 3, 5$.

To avoid the contingency, each problem instance is conducted 15 times. To analyze the effectiveness of Algorithms 1 and 2, they are compared with the B and B algorithm. The error of the solution produced by Algorithms 1 and 2 is calculated as

$$\frac{\widetilde{OF}(H) - \widetilde{OF}^*}{\widetilde{OF}^*} \times 100\%, \tag{13}$$

where $H \in \{\text{Algorithm 1, Algorithm 2}\}$, $\widetilde{OF}^*$ is the optimal objective value (see Equation (8)) generated by Algorithm 3. The running time (ms) of Algorithms 1-3 is defined. The results are summarized in Tables 1–3.

From Tables 1–3, it is found that Algorithm 1 based on the analytical properties of the problem apparently performs better than Algorithm 2 for each scale of the numerical experiments, and the maximum relative error percentage of Algorithm 1 is less than 0.214% with $\ddot{O} \leq 200$. The performance of the B and B algorithm is shown to be efficient, leading to the optimal solvency in terms of the cases with a large job number scale. It is also presented that the coefficient of σ has a noticeable impact on the complexity of the problem, implying that the smaller σ tends to generate more complex cases. For the smaller σ, the B and B algorithm needs more CPU time. However, for any σ, the gap in the CPU time between Algorithms 1 and 2 is not too big.

In Table 4, statistical hypothesis tests are conducted to compare the mean percentage errors of Algorithm 1 and Algorithm 2. For a representative display, the instances where $\ddot{O} = 100, 120, 140, 160, 180, 200$, $\sigma = 3$, and $\varpi_{g,k} \in [1, 100]$ are examined. The $t-$test is used for the tests:

$$t = \frac{\overline{Error_{Algorithm1}} - \overline{Error_{Algorithm2}}}{S_w\sqrt{1/n_{Algorithm1} + 1/n_{Algorithm2}}},$$

where $S_w^2 = \frac{(n_{Algorithm1}-1)S^2_{Algorithm1} + (n_{Algorithm2}-1)S^2_{Algorithm2}}{n_{Algorithm1}+n_{Algorithm2}-2}$ and $\overline{Error}$ denotes the mean error percentage. As the results in Tables 1–3 potentially show that the effectiveness performances of the two algorithms are ranked as Algorithm1>Algorithm3, the corresponding statistical hypothesis test is set as $H_0 : \mu_{Algorithm2} > \mu_{Algorithm1}, H_1 : \mu_{Algorithm2} \leq \mu_{Algorithm1}$. Type I error of 1% is used, and thus $t_{critical} = 2.5$. Further experiment results in Table 4 show that for all instances the hypothesis that $H_0 : \mu_{Algorithm2} > \mu_{Algorithm1}$ with a type I error of 1% is supported.

Table 1. Results for $v_{g,k}\varpi_{g,k} \in [1, 50]$.

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
Ö	**Q**	**σ**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
100	10	1	175.83	257	10,627.67	14,886	3.67	5	**0**	**0**	2272.83	2295	6.22	8.71
		3	7.9	22	482.7	1384	3.6	4	**0**	**0**	2203.5	2352	9.89	26.39
		5	10.17	20	598.5	1298	3.67	5	**0**	**0**	2499.17	2530	8.09	13.147
100	12	1	2037.17	6671	119,460.33	401,171	5.5	7	**0.01**	**0.037**	4291.5	4319	9.29	12.55
		3	88.5	386	4671.33	19,941	5.83	7	**0**	**0**	4773.5	4835	10.94	15.38
		5	30	84	1530.5	4356	5.5	6	**0**	**0**	4596.33	4768	10.1	21.27
100	14	1	64,659.4	269,953	3,365,621.6	14,105,756	8.4	10	**0.00035**	**0.0017**	8621.8	8693	9.15	16.2
		3	1536	2981	78,535	152,347	9	10	**0**	**0**	8639.8	8846	10.26	18.78
		5	338.4	973	16,728.4	48,324	8.4	9	**0.0002**	**0.0007**	8313.2	8723	13.66	19.14
100	16	1	199,804.1	1,210,619	9,729,114.6	59,289,573	12.2	14	**0.002**	**0.013**	69,604.5	70,978	9.8	14.32
		3	5518	21,077	296,168.8	1,051,335	11.6	13	**0.004**	**0.03**	69,253.5	70,416	13.09	17.76
		5	401.8	967	18,365.5	43,727	12.2	14	**0.00008**	**0.0008**	71,538.2	75,239	15.98	25.17
120	10	1	216.17	563	11,452.5	29,391	4	4	**0.008**	**0.048**	2450.83	2488	3.35	7.59
		3	13.8	54	729.6	2843	3.8	4	**0**	**0**	2348.9	2385	8.51	12.64
		5	11	34	582.17	1695	4.17	6	**0**	**0**	2683.17	2756	6.12	9.39
120	12	1	2899.67	7265	147,748.5	375,004	6.33	7	**0.024**	**0.122**	4651.5	4690	8.66	9.96
		3	68.5	132	3220.17	6246	6.67	8	**0**	**0**	5039.67	5143	9.82	16.13
		5	31.17	59	1550.5	3073	6.83	8	**0.0008**	**0.0053**	5054.83	5122	10.64	19.2

Table 1. *Cont.*

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
Ö	***Q***	**σ**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
120	14	1	10,006.6	19,412	435,290.8	835,108	9.2	10	**0.0008**	**0.0042**	9323.2	9437	8.26	12.21
		3	530.2	977	22,433.6	41,005	9.4	10	**0**	**0**	9080.8	9165	11.89	18.96
		5	66.4	98	3012.6	4404	9.6	11	**0.0024**	**0.0121**	8774	8883	14.31	24.01
120	16	1	843,894.25	2,883,886	35,433,283.5	121,153,291	14.5	17	**0**	**0**	15,476	15,622	9.848	14.03
		3	61,141.5	142,735	1,644,191.5	3,843,414	19.5	20	**0.0021**	**0.0084**	19,312.75	19,635	14.45	18.35
		5	271.08	419	10,762.33	17,094	14.5	17	**0.0007**	**0.002**	15,875.83	15,895	20.06	23.59
140	10	1	455.5	1599	20,615.5	71,768	4.83	5	**0.0004**	**0.0021**	2682	2718	4.01	7.83
		3	10.6	28	502.1	1323	4.3	5	**0**	**0**	2545.1	2600	8.12	13.45
		5	15	32	650.5	1561	4.67	5	**0**	**0**	2094.67	2962	8.16	17.48
140	12	1	4105.83	10,490	180,284.5	465,705	7.33	8	**0.003**	**0.016**	4979.33	4999	4.8	8.7
		3	47.67	73	1968	2973	7.67	9	**0**	**0**	5411.83	5548	9.9	19.9
		5	23.83	30	964	1217	7.33	8	**0**	**0**	5429	5556	10.24	13.75
140	14	1	50,553.4	117,757	1,993,660	4,706,960	10.8	11	**0.007**	**0.034**	9784.2	9811	6.26	10.17
		3	1339.6	3279	52,678	134,425	10.6	11	**0**	**0**	9811.4	10,039	14.3	17.4
		5	64	142	2457.2	5792	10	11	**0**	**0**	9349.4	9740	15.43	20.12
140	16	1	166,960	447,424	6,116,255	16,383,293	16.75	20	**0.006**	**0.023**	16,496.25	16,766	12.31	22.12
		3	2232.5	5255	60,607	144,116	18.75	20	**0**	**0**	18,515.75	18,790	19.1	22.83
		5	332.25	499	11,752	18,263	15.25	16	**0**	**0**	16,794	16,983	15.8	19.51
160	10	1	227.33	373	9236.33	15,118	5.17	6	**0.003**	**0.01**	2878	2913	5.02	6.32
		3	11	34	453.9	1416	5.7	9	**0**	**0**	2717.8	2789	9.23	14.22
		5	9.67	26	365.67	912	4.83	5	**0**	**0**	3078.17	3138	6.68	14
160	12	1	17,965.17	94,241	675,544	3,524,123	8	8	**0.02**	**0.12**	5342.83	5488	7.17	13.31
		3	159.67	384	5693.5	13,875	8	9	**0**	**0**	5705.33	5751	10.51	22
		5	110.17	524	3884.5	18,778	7.67	8	**0**	**0**	5776.5	5833	7.26	10.37
160	14	1	79,640	251,562	2,795,280	8,997,594	12	14	**0.0005**	**0.0029**	10,322.4	10,453	9.68	12.55
		3	4694	17,685	156,976	597,877	11.8	13	**0**	**0**	10,389	10,577	10	12.36
		5	138.8	281	5010	10,482	11	12	**0**	**0**	9447	9533	12.26	14.11
160	16	1	570,198	1,476,613	18,462,293	47,409,312	17	18	**0.0006**	**0.0027**	16,597	17,459	9.05	11.94
		3	5860.75	11,937	180,292	351,636	18.25	19	**0**	**0**	17,415	17,824	14.37	22.25
		5	328.75	601	10,467	19,738	18.25	19	**0**	**0**	17,673	17,814	19.79	30.48
180	10	1	356.17	806	12,843.67	28,269	5.5	7	**0.006**	**0.03**	3068.17	3112	4.58	12.58
		3	12.6	45	462.5	1699	5.3	6	**0**	**0**	2920.6	2964	7.87	13.99
		5	24	30	797	1018	5.5	6	**0**	**0**	3288.5	3334	5.7	9.44
180	12	1	6426.33	12,934	237,213	500,166	8.67	9	**0.009**	**0.05**	5354.9	5658	5.45	6.72
		3	332.5	563	10,763	16,754	9	10	**0**	**0**	6104.1	6217	9.6	16.2
		5	39.33	90	1274.33	2982	9	10	**0**	**0**	6089.3	6179	13.81	18.33
180	14	1	74,008.6	157,651	2,211,136	4,730,406	14	16	**0**	**0**	10,878	11,003	6.23	10.57
		3	1262	2,338	39,928.2	74,721	13.8	15	**0.003**	**0.016**	11,011	11,450	13.76	23.6
		5	248.8	542	7692.6	17,820	13.6	15	**0**	**0**	10,355.8	10,860	13.38	20.15
180	16	1	1,874,526	6,582,648	57,396,522	202,311,764	19.5	20	**0.0004**	**0.0019**	17,192	17,441	10.04	16.74
		3	6445.75	21,324	221,697	727,407	15.75	17	**0.01**	**0.038**	16,556	16,671	15.83	18.75
		5	3281	10,783	94,828	313,272	19.5	22	**0**	**0**	18,737	18,926	14.1	17.05
200	10	1	481.83	766	16,095.5	25,248	6.5	8	**0**	**0**	3227.67	3288	3.49	6.01
		3	12	25	407.5	810	5.9	6	**0**	**0**	3123.5	3190	6.92	13.73
		5	11.17	16	354.33	541	5.83	6	**0**	**0**	3481.5	3580	8.09	12.45
200	12	1	14,715	34,002	505,003	1,171,235	8.83	9	**0**	**0**	5663.5	5829	4.39	6.4
		3	114.67	337	3510.67	10,782	9.83	12	**0**	**0**	6272	6485	10.68	15.74
		5	53.67	99	1612.83	2969	9.5	11	**0**	**0**	6431.17	6513	13.89	21.51
200	14	1	68,228	106,308	1,988,507	3,107,931	14.75	16	**0.001**	**0.0059**	28,108	28,783	10.1	12.3
		3	1867	10,405	63,026	298,133	15.33	19	**0**	**0**	28,039	28,709	12.93	22.43
		5	334.8	1466	9071.7	40,078	15	16	**0**	**0**	11,087	11,404	13.58	23.88
200	16	1	1,033,181	2,735,173	29,163,754	77,121,970	20	21	**0.005**	**0.02**	17,884	18,201	10.65	12.97
		3	2709.5	7421	114,889	314,102	13.25	15	**0**	**0**	14,930	15,927	20.5	28.3
		5	1232.5	2125	32,467.25	55,554	20.25	22	**0**	**0**	18,227	18,930	18.34	21.67

Table 2. Results for $v_{g,k}\varpi_{g,k} \in [50, 100]$.

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
$\ddot{O}$	Q	σ	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
		1	692.67	1837	42,941	112,869	3.67	6	0	0	2101.5	2215	2.36	4.05
100	10	3	68.67	121	3997.17	6986	4	5	0.0009	0.0052	2299.7	2471	5.66	10.47
		5	52.67	148	3078.33	8766	3.93	5	0	0	2258.5	2281	3	5.6
		1	12,155.6	29,096	668,219	1,584,498	5.6	6	0	0	4658	4756	5.37	8.94
100	12	3	1421.2	4734	80,143	269,116	5.8	8	0	0	4449.2	4704	8.84	15.5
		5	210.25	390	11,495.75	21,383	6.25	8	0	0	4640.75	4745	5.79	6.78
		1	212,081	609,272	10,779,171	30,865,791	8.2	9	0.005	0.0235	8636	8692	8.02	15.15
100	14	3	9621.6	27,407	502,216.5	1,471,563	8.2	9	0.0011	0.0057	7882	7917	7.33	11.5
		5	1211.4	2583	59,440	128,597	8.4	10	0	0	8769	8898	9.06	15.87
		1	11,344,238	52,809,989	88,471,297	1,302,004,209	12.6	14	0.0003	0.003	72,779	74,104	7.88	14.96
100	16	3	29,703.8	97,818	1,337,187	4,463,297	11.6	14	0	0	71,341.5	72,994	11.69	17.82
		5	14,915.4	55,930	687,413.2	2,532,746	12.1	14	0	0	69,521.8	72,567	12.2	18
		1	493.67	920	26,844	50,616	4.5	5	0	0	2316.17	2352	3.28	4.89
120	10	3	133	358	6781	18,100	4.16	5	0.001	0.006	2499	2525	4.37	9.34
		5	21.33	52	973.17	2250	4.17	5	0	0	2456.67	2495	5.8	8.99
		1	28,551.8	99,908	1,381,873.4	4,948,904	6.4	7	0	0	4937.2	5128	3.99	6.77
120	12	3	701	1224	32,003.2	54,431	6	7	0.0009	0.0045	4897.6	4949	5.69	7.61
		5	177.75	454	7831	19,741	6	6	0	0	5006	5072	10.1	12.13
		1	366,480	716,135	15,927,240	31,275,408	9.8	11	0.004	0.02	9311	9465	4.68	5.58
120	14	3	30,233.2	89,204	1,372,230	3,969,285	8.6	10	0	0	8449	8514	7.35	15.97
		5	2044.6	5891	85,747	245,514	9.8	11	0	0	9380	9481	10.2	14.2
		1	7,572,497	24,249,379	317,472,513	1,018,104,727	13.6	17	0.008	0.055	77,026	79,544	8.38	20
120	16	3	60,707.88	246,895	2,504,289	10,502,543	16	28	0	0	77,911	79,407	12.43	14.86
		5	12,429.25	44,304	504,877	1,753,666	14	16	0	0	76,225.13	78,184	14.9	27.9
		1	966.33	2528	45,752.67	121,293	4.83	5	0	0	2503	2538	2.38	3.7
140	10	3	63	106	2868	4721	4.83	5	0	0	2715.5	2793	5.56	10.62
		5	71.5	164	3162.17	7601	4.5	5	0	0	2642.33	2801	6.43	10.78
		1	20,983.8	36,950	890,651	1,586,799	6.8	8	0.002	0.01	5433	5555	4.65	6.52
140	12	3	866.2	2444	35,304	99,934	7.2	8	0.015	0.06	5303.8	5337	6.78	9.16
		5	219.75	373	8811.25	14,336	7.5	9	0	0	5328	5414	5.6	11.43
		1	180,479	337,237	7,144,871	13,413,322	10	11	0	0	9954	10,115	4.92	7.12
140	14	3	2341.8	6683	97,818	285,987	9.8	11	0	0	8392.4	8412	10.44	14.15
		5	1285.2	2532	46,758.6	93,674	10.6	11	0	0	9893.2	10,008	10.01	13.25
		1	3,116,283	6,214,264	113,724,235	226,772,280	16.13	18	0.008	0.053	79,908	80,898	6.44	8.4
140	16	3	79,115.5	236,950	2,675,645	8,194,853	16	17	0	0	82,224	83,284	11.66	18.9
		5	10,535.9	30,773	370,875	1,088,717	15.6	16	0	0	81,393	82,736	11.6	14.28
		1	1662.5	3989	69,675.5	164,478	5.33	6	0	0	2697.3	2783	2.72	4.29
160	10	3	432.67	1462	17,212.17	58,095	5.33	6	0	0	2899.67	2939	3.17	6.14
		5	52.67	73	2060.67	2822	5.16	6	0	0	2833.17	2872	5.4	7.4
		1	4559.6	10,545	172,079.8	392,283	8.2	9	0	0	5778	5934	4.09	5.44
160	12	3	1515.4	4496	56,064.8	166,260	7.6	8	0	0	5632.8	5690	5.7	10.22
		5	271.25	742	9671.25	26,713	7.75	8	0	0	5712.5	5893	7.08	9.25
		1	358,318	872,836	12,219,505	29,222,800	12	13	0.006	0.032	10,410	10,566	7.72	10.92
160	14	3	6759.2	16,794	259,990	637,153	11	12	0	0	8906.2	8921	9.29	17.04
		5	1621.4	3155	52,268.6	100,539	11.6	13	0	0	10,228.8	10,369	10.72	14.66
		1	15,875,052	25,972,880	533,903,012	852,604,283	18	19	0	0	80,650	81,123	7.23	7.9
160	16	3	169,494	541,659	5,628,651	18,430,108	17	18	0	0	79,216	80,004	10.2	14.3
		5	77,583.5	351,324	2,594,059	11,773,483	17.6	20	0	0	82,318.4	87,191	14.7	26.25
		1	2453.67	5127	93,880	194,697	5.33	6	0	0	2882.5	2991	3.23	6.79
180	10	3	217.83	364	7643	12,718	5.33	6	0	0	3111	3135	4.07	5.68
		5	112.83	328	4174.3	12,325	5.16	6	0	0	3046.67	3094	5.74	10.89
		1	8435	24,694	284,680	828,457	9	10	0	0	6147.6	6210	4.65	6.66
180	12	3	3219.8	9641	105,989	315,723	8.8	9	0	0	5991	6107	4.44	7.73
		5	135.75	319	4632	11,256	8.75	9	0	0	5980.75	6040	10.71	12.35
		1	245,846	629,740	7,714,735	19,988,982	13.8	15	0	0	10,770	11,037	7.24	11.34
180	14	3	6039.5	11,594	213,514	422,817	12.8	14	0	0	9421.4	9513	7.45	8.65
		5	12,808.6	44,504	355,851	1,181,280	14.8	19	0.0014	0.0071	10,937	11,120	9.94	17.94
		1	1,005,534	8,958,852	30,089,268	88,896,652	20	22	0	0	88,412.2	89,852	10.5	16.5
180	16	3	112,166.7	282,983	3,613,833	9,101,955	18.25	19	0	0	81,956	82,750	14.37	23.7
		5	19,860.2	76,605	575,351	2,151,625	18.8	21	0	0	83,174	84,424	11.8	16.78
		1	1900	3525	66,931	123,429	6.3	7	0	0	3035.5	3088	2.75	4
200	10	3	108.83	222	3596.5	7678	6.17	7	0	0	3286.5	3349	5.73	9.48
		5	58.17	134	1959.5	4291	6	7	0	0	3219	3280	6.17	15

Table 2. *Cont.*

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
Ö	**Q**	σ	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
200	12	1	64,804.8	169,620	1,967,479	5,081,959	9.6	10	**0.002**	**0.012**	6421	6476	3.78	7.52
		3	3398.4	7321	102,159	217,628	9.8	10	**0.003**	**0.014**	6310	6420	7.36	12.5
		5	132.5	384	4049.75	11,995	9.75	10	**0**	**0**	6311.75	6399	6.04	9.76
200	14	1	220,451	293,846	6,409,273	9,303,055	15.5	17	**0**	**0**	11,524	11,638	5	5.63
		3	10,268.8	25,866	319,452	811,783	13.4	14	**0**	**0**	9965.2	10,025	6.24	13.62
		5	2007.6	7062	16,010	196,268	14.8	16	**0.001**	**0.006**	11,582	11,649	11.38	18.63
200	16	1	842,391	1,058,692	24,820,918	156,925,987	24	25	**0**	**0**	87,907	90,125	10.64	15.69
		3	78,060	163,149	2,026,217	4,221,889	20.25	21	**0**	**0**	93,710.5	95,524	10.99	13.41
		5	28,925	48,010	785,538	1,310,844	20.5	21	**0**	**0**	93,008	93,784	15	20.1

Table 3. Results for $v_{g,k}\varpi_{g,k} \in [1, 100]$.

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
Ö	**Q**	σ	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
100	10	1	329.83	824	19,328.17	48,734	3.67	4	**0.014**	**0.079**	2140.17	2155	4.64	8.26
		3	50.83	87	2968.5	4918	3.67	4	**0**	**0**	2264.33	2319	4.5	8.02
		5	18.83	58	1096	3502	3.5	4	**0**	**0**	2118	2138	3.78	6.28
100	12	1	4367.67	9686	240,672.33	540,128	6	7	**0.01**	**0.043**	4705	4762	5.76	9.03
		3	181.67	399	9496.83	21,206	5.67	6	**0.007**	**0.044**	4523.33	4734	7.16	11.97
		5	46.83	107	2435.5	5610	5.83	6	**0**	**0**	4713.33	4818	8.31	14.59
100	14	1	24,737.4	35,572	1,209,171.6	1,698,917	8.4	9	**0.005**	**0.023**	8615.4	8775	8.7	14.28
		3	1625.2	4206	81,509.4	211,300	8.4	9	**0**	**0**	8503.4	8703	10.47	16.5
		5	526.4	1020	27,985	55,886	8	9	**0**	**0**	8060	8116	11.05	16.53
100	16	1	2,222,533	9,200,131	112,623,096	468,724,389	11.8	14	**0.0005**	**0.005**	68,698	72,010	8.86	19
		3	13,294.2	37,569	614,326	17,99,811	12.7	18	**0**	**0**	71,314	73,597	11.68	16.17
		5	6338	25,535	291,936	1,145,754	11.4	12	**0.002**	**0.03**	71,188.5	72,424	11.87	19
120	10	1	247.67	500	13,054.5	26,234	4	4	**0**	**0**	2326.67	2343	3.32	6.8
		3	41.5	70	2202	3779	4.17	5	**0**	**0**	2482.5	2515	4.64	10
		5	21	45	1114.83	2544	4.33	5	**0**	**0**	2289	2302	5.96	9.2
120	12	1	4983	9675	237,658	461,795	6.83	8	**0.029**	**0.178**	5083	5122	4.73	7.49
		3	720.67	2898	33,675.3	137,482	6.5	7	**0.06**	**0.214**	5122	5197	9.36	13.12
		5	256	1165	12,260.17	56,351	6.5	7	**0**	**0**	5074.67	5144	8.92	14.35
120	14	1	164,821	704,140	7,320,790	31,318,766	9.6	10	**0.013**	**0.027**	9125.2	9352	12.13	16.22
		3	2159.2	5882	94,720	262,132	10	11	**0.001**	**0.005**	9187.6	9385	9.3	12
		5	408	964	18,181.2	42,851	9	9	**0.001**	**0.007**	8608	8656	12.1	14.18
120	16	1	4,849,519	9,598,825	52,989,654	112,559,558	13.2	15	**0**	**0**	15,865	15,997	10.56	16.98
		3	12,306.25	29,416	490,665	1,151,196	13.5	15	**0**	**0**	15,620.2	15,761	12.65	16.87
		5	3802.5	10,468	162,910	448,843	13.25	14	**0.001**	**0.004**	14,382	15,439	16.78	18.53
140	10	1	481	685	22,047	31,800	5	5	**0.004**	**0.023**	2540	2632	3.43	4.68
		3	130.5	416	5846.33	18,224	4.66	5	**0**	**0**	3694.83	2770	4.7	7.76
		5	17.17	29	765.67	1227	4.5	5	**0**	**0**	2483	2479	6.23	11
140	12	1	17,779.17	53,234	745,586.17	2,239,041	7.17	8	**0.008**	**0.029**	5417.67	5509	4.4	6.71
		3	518	1222	20,120	48,454	7.67	9	**0.01**	**0.06**	5590.17	5694	7.55	8.36
		5	101.5	165	4229.83	6889	7.5	8	**0**	**0**	5413.67	5487	7.8	11.7
140	14	1	230,487	820,439	9,048,553.6	32,169,283	11.4	12	**0**	**0**	9604	9685	6.29	9.28
		3	5087	18,172	194,122.4	689,592	10	11	**0**	**0**	9630.2	9722	12.6	16.74
		5	683	2720	28,079.2	112,672	10.6	12	**0**	**0**	9117	9288	11.35	16.13
140	16	1	2,052,347	3,993,328	73,874,865	143,687,994	17	18	**0.0239**	**0.236**	16,395	16,818	7.53	17.59
		3	23,583	34,352	894,455	1,202,496	15	16	**0.003**	**0.01**	16,495	16,802	14.85	17.4
		5	4328.75	8953	147,934	302,143	16	17	**0.004**	**0.015**	16,214	16,248	13.43	24.17
160	10	1	399.83	851	16,425.17	34,669	5.17	6	**0**	**0**	2703	2722	2.7	3.95
		3	162.67	363	6408.5	14,658	5.17	6	**0**	**0**	2895.5	2974	4.48	9
		5	34.83	58	1443.5	2365	5	5	**0**	**0**	2654.67	2674	4.32	9.49
160	12	1	17,007.5	68,790	623,283	2,495,037	7.83	9	**0.002**	**0.01**	5767.83	5854	3.84	5.61
		3	809.17	1142	28,503.83	41,524	8.17	9	**0**	**0**	5939.17	6082	6.15	7.21
		5	296.5	559	10,005.83	18,744	8.17	9	**0.001**	**0.003**	5753.13	5836	6.53	11.7
160	14	1	141,663	343,831	4,923,899.6	11,776,362	11.8	13	**0.025**	**0.057**	10,320.4	10,385	6.9	8.52
		3	6089.8	20,396	205,664	688,699	12.2	15	**0**	**0**	10,127.4	10,591	9.83	14.82
		5	563	2003	19,840.8	71,533	12.2	13	**0.001**	**0.006**	10,234.6	10,594	11.36	17.32
160	16	1	625,439	881,041	19,927,201	28,658,146	17.67	18	**0.01**	**0.034**	16,658	17,092	9.39	10.53
		3	241,470	923,629	7,816,964	29,922,040	18	19	**0**	**0**	41,347	42,603	11.1	13.48
		5	2321.75	3659	75,544.45	112,519	17.5	19	**0**	**0**	41,609	42,607	14.47	18.46

Table 3. *Cont.*

			CPU Time (ms) of Algorithm 3		Node Number of Algorithm 3		CPU Time (ms) of Algorithm 1		Error of Algorithm 1		CPU Time (ms) of Algorithm 2		Error of Algorithm 2	
Ö	**Q**	**σ**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
180	10	1	2058.17	7629	76,555.33	280,949	6	7	**0.03**	**0.126**	2883.17	2934	3.58	6.44
		3	56.33	97	2013.5	3024	5.83	6	**0**	**0**	3072.17	3147	7.15	11.33
		5	92.5	201	3590.33	7950	5.67	7	**0**	**0**	2850.5	2883	5	12.26
180	12	1	34,678.17	100,101	1,148,668.8	3,279,576	9.17	11	**0.001**	**0.009**	6053	6124	6.16	9.66
		3	3099.83	10,978	98,262.5	353,058	9	10	**0**	**0**	6273.5	6361	7.39	11
		5	172.33	551	5601.33	17,282	8.5	9	**0**	**0**	6116.5	6171	11.02	17.11
180	14	1	361,279	1,297,340	11,276,602	4,020,950	13.6	15	**0.005**	**0.03**	10,951	11,032	7.1	11.2
		3	4352.2	10,639	131,942	306,106	15.4	20	**0.002**	**0.012**	10,404	10,446	8.76	10.35
		5	386.4	726	11,725	22,077	13.2	14	**0**	**0**	11,059.2	11,221	12.17	17.82
180	16	1	482,350.67	668,148	15,140,423	21,101,587	20	21	**0**	**0**	44,180	44,475	15.4	20.13
		3	44,507.75	164,236	1,305,440.7	4,818,838	18.75	20	**0**	**0**	44,716	45,632	13.57	18.81
		5	3550.75	4468	104,111.7	125,232	21	24	**0.003**	**0.0122**	43,519	44,053	12.5	20.78
200	10	1	1318.5	3141	44,526.17	106,290	5.83	6	**0**	**0**	2991.5	3067	2.94	5.53
		3	204.67	528	6627	17,212	5.83	6	**0**	**0**	3170.33	3215	6.33	12.04
		5	56.83	152	1995.83	5488	6	7	**0**	**0**	3042.83	3055	5.8	13.07
200	12	1	95,516.3	182,438	2,935,710	5,581,330	9.17	10	**0.01**	**0.063**	6344	6468	3.64	6.82
		3	2094.17	7031	62,325.67	208,191	10	11	**0**	**0**	6470.33	6530	7.08	13.13
		5	400	1214	12,172.17	36,799	10.83	15	**0.003**	**0.02**	6039	6144	8.21	14.75
200	14	1	122,358.4	388,192	3,518,271	11,078,822	15.6	16	**0.004**	**0.02**	11,509	11,714	8.18	11.7
		3	21,235.6	68,264	601,329	1,945,511	14.8	16	**0**	**0**	11,224.8	11,480	9.91	13.44
		5	652.4	1653	18,068.6	45,031	14.8	17	**0**	**0**	11,560.4	11,717	11.29	18.83
200	16	1	3,835,773	7,916,941	116,880,870	236,468,665	20	22	**0.025**	**0.055**	40,431	40,872	10.61	12.29
		3	3614.75	4964	96,473.25	129,904	20.25	22	**0**	**0**	47,094.75	48,138	14.26	16.92
		5	2333.5	3668	60,853	98,033	22	23	**0**	**0**	46,873	47,311	14.06	18.6

Table 4. $t-$values for $v_{g,k}\varpi_{g,k} \in [1, 100]$.

Ö	Q	σ	t
100	10	3	4.5484
100	12	3	5.2164
100	14	3	5.0474
100	16	3	4.5728
120	10	3	4.9150
120	12	3	5.2457
120	14	3	4.7943
120	16	3	4.5306
140	10	3	4.6981
140	12	3	4.8486
140	14	3	4.5247
140	16	3	5.1256
160	10	3	4.7880
160	12	3	4.7529
160	14	3	5.0211
160	16	3	4.8972
180	10	3	4.7066
180	12	3	4.9331
180	14	3	4.6207
180	16	3	4.8987
200	10	3	4.9730
200	12	3	5.2078
200	14	3	4.8210
200	16	3	5.1509

6. Conclusions

A group scheduling problem with common/slack due-date assignment and resource allocation was investigated in this paper. Under the generalization of CON/SLK assignments and the job numbers of each group, this paper was intended to decide the job/group sequence, resource allocation, and due-date assignment. **To build systematic solution algorithms, heuristics and a branch-and-bound method incorporating the optimal properties and lower and upper bounds are proposed. Numerical experiments showed that the lower bound developed in this paper is efficient and Algorithm 1 outruns Algorithm 2.** As for future study, a general case of a multi-objective flowshop will be introduced. For tackling the complexity, a well-designed solution framework incorporating an upper bound and lower bound strategy will also be explored.

Author Contributions: Writing—original draft, H.H.; Writing—review & editing, H.H., Y.Z., X.M., Z.-G.L. and J.-B.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Social Science Fund of China (Grant No. 23ATJ006), Liaoning Science and Technology Innovation Think Tank Project (Project No. LNKX2023C03), Liaoning Social Science Fund (Project No. L23BGL013), and National Social Science Fund of China (Grant No. 21&ZD148).

Data Availability Statement: The data used to support the findings of this study are available from the corresponding authors upon request.

Conflicts of Interest: The authors declare there are no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Q	the number of groups ($Q \geq 2$)
$\ddot{O}$	the number of jobs
F_g	gth group ($g = 1, 2, \ldots, Q$)
O_g	the number of jobs in F_g (i.e., $O_1 + O_2 + \ldots + O_Q = \ddot{O}$)
s_g	setup time in group F_g
$G_{g,k}$	the kth job in F_g ($k = 1, 2, \ldots, O_g$)
$\varpi_{g,k}$	the workload of $G_{g,k}$
$u_{g,k}$	the amount of resources allocated to $G_{g,k}$
$E_{g,k}$ (resp. $T_{g,k}$)	the earliness (resp. tardiness) of $G_{g,k}$ in F_g
$d_{g,k}$ (resp. $C_{g,k}$)	the due date (resp. completion time) of $G_{g,k}$
χ	a group schedule within F_g
ψ_g	an internal job schedule within F_g
σ (ξ)	the given constant
$v_{g,k}$	the unit consumption cost
$\alpha_{g,k}$ (resp. $\beta_{g,k}$)	a position-dependent weight for the earliness (resp. tardiness) cost
CON (resp. SLK)	the common (resp. slack) due date
d_g	the common due date in group F_g
q_g	the common flow allowance in group F_g

References

1. Foumani, M.; Smith-Miles, K. The impact of various carbon reduction policies on green flowshop scheduling. *Appl. Energy* **2019**, *249*, 300–315. [CrossRef]
2. Shabtay, D.; Steiner, G. A survey of scheduling with controllable processing times. *Discret. Appl. Math.* **2007**, *155*, 1643–1666. [CrossRef]
3. Manier, M.-A.; Bloch, C. A classification for hoist scheduling problems. *Int. J. Flex. Manuf. Syst.* **2003**, *15*, 37–55. [CrossRef]
4. Kuntay, I.; Xu, Q.; Uygun, K.; Huang, Y. Environmentally conscious hoist scheduling for electroplating facilities. *Chem. Eng. Commun.* **2006**, *193*, 273–292. [CrossRef]
5. Uruk, Z.; Gultekin, H.; Akturk, M.S. Two-machine flowshop scheduling with flexible operations and controllable processing times. *Comput. Oper. Res.* **2013**, *40*, 639–653. [CrossRef]
6. Mor, B.; Mosheiov, G. Batch scheduling of identical jobs with controllable processing times. *Comput. Oper. Res.* **2014**, *41*, 115–124. [CrossRef]
7. Akhoondi, F.; Lotfi, M.M. A heuristic algorithm for master production scheduling problem with controllable processing times and scenario-based demands. *Int. J. Prod. Res.* **2016**, *54*, 3659–3676. [CrossRef]

8. Li, L.; Wang, J.-J. Scheduling jobs with deterioration effect and controllable processing time. *Neural Comput. Appl.* **2018**, *29*, 1163–1170. [CrossRef]
9. Sun, X.; Geng, X.-N.; Wang, J.-B.; Liu, F. Convex resource allocation scheduling in the no-wait flowshop with common flow allowance and learning effect. *Int. J. Prod. Res.* **2019**, *57*, 1873–1891. [CrossRef]
10. Shabtay, D.; Itskovich, Y.; Yedidsion, L.; Oron, D. Optimal due date assignment and resource allocation in a group technology scheduling environment. *Comput. Oper. Res.* **2010**, *37*, 2218–2228. [CrossRef]
11. Zhu, Z.; Sun, L.; Chu, F.; Liu, M. Single-machine group scheduling with resource allocation and learning effect. *Comput. Ind. Eng.* **2011**, *60*, 148–157. [CrossRef]
12. Webster, S.; Baker, K.R. Scheduling groups of jobs on a single machine. *Oper. Res.* **1995**, *43*, 692–703. [CrossRef]
13. Li, S.; Ng, C.T.; Yuan, J. Group scheduling and due date assignment on a single machine. *Int. J. Prod. Econ.* **2011**, *130*, 230–235. [CrossRef]
14. Liu, F.; Yang, J.; Lu, Y.-Y. Solution algorithms for single-machine group scheduling with ready times and deteriorating jobs. *Eng. Optim.* **2019**, *51*, 862–874. [CrossRef]
15. Yang, Y.; Yin, G.; Wang, C.; Yin, Y. Due date assignment and two-agent scheduling under multitasking environment. *J. Comb. Optim.* **2022**, *44*, 2207–2223. [CrossRef]
16. Yin, Y.; Wu, W.-H.; Cheng, T.C.E.; Wu, C.-C. Due date assignment and single-machine scheduling with generalized positional deteriorating jobs and deteriorating multi-maintenance activities. *Int. J. Prod. Res.* **2014**, *52*, 2311–2326. [CrossRef]
17. Yin, Y.; Wang, D.; Wu, C.-C.; Cheng, T.C.E. CON/SLK due date assignment and scheduling on a single machine with two agents. *Nav. Res. Logist.* **2016**, *63*, 416–429. [CrossRef]
18. Yin, Y.; Yang, Y.; Wang, D.; Cheng, T.C.E.; Wu, C.-C. Integrated production, inventory, and batch delivery scheduling with due date assignment and two competing agents. *Nav. Res. Logist.* **2018**, *65*, 393–409. [CrossRef]
19. Yin, Y.; Li, D.; Wang, D.; Cheng, T.C.E. Single-machine serial-batch delivery scheduling with two competing agents and due date assignment. *Ann. Oper. Res.* **2021**, *298*, 497–523. [CrossRef]
20. Yan, J.-X.; Ren, N.; Bei, H.-B.; Bao, H.; Wang, J.-B. Study on resource allocation scheduling problem with learning factors and group technology. *J. Ind. Manag. Optim.* **2023**, *19*, 3419–3435. [CrossRef]
21. Liu, W.; Wang, X. Group technology scheduling with due-date assignment and controllable processing times. *Processes* **2023**, *11*, 1271. [CrossRef]
22. Nawaz, M.; Enscore, E.E., Jr.; Ham, I. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *OMEGA Int. J. Manag. Sci.* **1983**, *11*, 91–95. [CrossRef]
23. Noman, M.A.; Alatefi, M.; Al-Ahmari, A.M.; Ali1, T. Tabu search algorithm based on lower bound and exact algorithm solutions for minimizing the makespan in non-identical parallel machines scheduling. *Math. Probl. Eng.* **2021**, *2021*, 1856734. [CrossRef]
24. Yan, J.-X.; Ren, N.; Bei, H.-B.; Bao, H.; Wang, J.-B. Scheduling with resource allocation, deteriorating effect and group technology to minimize total completion time. *Mathematics* **2022**, *10*, 2983. [CrossRef]