Hindawi
Discrete Dynamics in Nature and Society
Volume 2020, Article ID 9746538, 7 pages
https://doi.org/10.1155/2020/9746538

Research Article

Optimization for Due-Window Assignment Scheduling with Position-Dependent Weights

Li-Yan Wang,[1] Dan-Yang Lv,[1] Bo Zhang,[1] Wei-Wei Liu ,[2] and Ji-Bo Wang [1]

[1]*School of Science, Shenyang Aerospace University, Shenyang 110136, China*
[2]*Department of Science, Shenyang Sport University, Shenyang 110102, China*

Correspondence should be addressed to Wei-Wei Liu; liuww_2010@163.com and Ji-Bo Wang; wangjibo75@163.com

Received 14 May 2020; Accepted 16 June 2020; Published 22 July 2020

Academic Editor: Chin-Chia Wu

This paper considers a single-machine due-window assignment scheduling problem with position-dependent weights, where the weights only depend on their position in a sequence. The objective is to minimise the total weighted penalty of earliness, tardiness, due-window starting time, and due-window size of all jobs. Optimal properties of the problem are given, and then, a polynomial-time algorithm is provided to solve the problem. An extension to the problem is offered by assuming general position-dependent processing time.

1. Introduction

Conventionally, in scheduling theory, due-windows are job-dependent either if they are dictated by the customer (i.e., given constants) or they are decision variables (i.e., due-window assignment). A due-window for job J_i is defined by a due-window starting time d_i^1 and a due-window finishing time d_i', i.e., the due-window $[d_i^1, d_i']$, and the due-window size is $D_i = d_i' - d_i^1$. In the just-in-time (JIT) production methodology and scheduling theory, setting proper due-windows is challenging (see Gong et al. [1], Janiak et al. [2], and Geng et al. [3]). In the literature, three very popular due-window assignment methods are studied:

Common due-window (CON-DW) assignment method (Liman et al. [4, 5]): all jobs are assigned a common due-window, i.e., all the jobs have a common due-window $[d^1, d']$, where $d_i^1 = d^1$, $d_i' = d'$, the due-window size of all the jobs is $D = d' - d^1$, and both d^1 and D are decision variables. In the literature, most studies considered the CON-DW assignment method, e.g., Mosheiov and Sarig [6] addressed a minmax CON-DW assignment problem, the objective of which is to minimise the largest cost among earliness, tardiness, due-window starting time, and due-window size. They proved that the single-machine and two-machine flow-shop problems can be solved in polynomial time. They also proved that the cases of parallel identical machines and uniform machines are NP-hard. Yin et al. [7] considered the batch delivery scheduling problem with an assignable common due-window on a single machine. Yin et al. [8] studied the single-machine scheduling problem with CON-DW assignment and batch delivery cost. Liu et al. [9] considered the single-machine CON-DW assignment scheduling problem with deteriorating jobs. For the weighted sum of earliness, tardiness, and due-window location penalty minimization, they proposed a polynomial-time algorithm to solve the problem. Wang and Wang [10] considered the single-machine resource allocation scheduling problem with learning effect and CON-DW assignment.

Slack due-window (SLK-DW) assignment method (Mosheiov and Oron [11]) is $d_i^1 = p_i + q^1$, $d_i' = p_i + q'$, and $D_i = d_i' - d_i^1 = q' - q^1 = D$, where p_i is the normal processing time of job J_i and q^1 and D are decision variables. Wang et al. [12] considered the single-machine SLK-DW assignment scheduling problem with deteriorating jobs and learning effect. Ji et al. [13] considered the single-machine SLK-DW assignment scheduling problem with group technology. Yin et al. [14], Yin et al. [15], and Wang et al. [16] considered SLK-DW assignment scheduling problems with resource allocation (controllable processing time). Mor and Mosheiov [17] considered SLK-DW assignment proportionate flow-shop scheduling problems.

Different due-windows' (DIF-DW) assignment method: it is assumed that the job J_i has a due-window $[d_i^1, d_i']$, where $d_i^1 \geq 0$ and $d_i' \geq 0\,(d_i^1 \leq d_i')$ denote the starting time and finishing time of the due-window, respectively. The due-window size of the job J_i is $D_i = d_i' - d_i^1$, and both d_i^1 and D_i are decision variables. Wang et al. [12] considered DIF-DW assignment scheduling problems with deteriorating jobs and learning effect.

In a recent paper, Wang et al. [18] considered CON-DW and SLK-DW assignment methods with position-dependent weights, i.e., the weight does not correspond with the job but with the position in which some job is scheduled. They proved that both these due-window assignment methods with position-dependent weights can be solved in polynomial time, respectively. *"The scheduling with due-window assignment has many real-world applications. For example, the due-window might reflect an assembly environment in which the components of the product should be ready within a time interval in order to avoid staging delays or a shop where several jobs constitute a single customer's order. It is clear that a wide due-window increases the supplier's production and delivery flexibility. However, a large due-window and delaying job completion reduce the supplier's competitiveness and customer service level"* (Yang et al. [19]). It is natural and interesting to continue the work of Wang et al. [18] but study the DIF-DW assignment scheduling problem with position-dependent weights. The contributions of this paper are given as follows: (1) the structural properties of scheduling problems are derived; (2) the total weighted penalty of earliness, tardiness, due-window starting time, and due-window size of all jobs' minimization can be solved in polynomial time; and (3) it is further extended the model to the case with general position-dependent processing time. We refer the reader to the survey of Janiak et al. [2] on the scheduling problems with (CON-DW, SLK-DW, and DIF-DW) due-windows.

The remainder of the paper is organized as follows. In Section 2, we formulate the problem. Section 3 gives some results and an optimal policy for the proposed problem. An extension of the proposed problem is given in Section 4. Finally, the conclusion and future work are given.

2. Problem Description

A set of n jobs $\tilde{N} = \{J_1, J_2, \ldots, J_n\}$ needs to be processed on a single machine. All the independent jobs are available at time zero, and preemption is not allowed. For a given sequence, we assume that job J_i has a due-window $[d_i^1, d_i']$, where $d_i^1 \geq 0$ ($d_i' \geq 0$) denote the starting time (finishing time) of the due-window, $d_i^1 \leq d_i'$. The due-window size of job J_i is defined by $D_i = d_i' - d_i^1$, and d_i^1 and D_i of all jobs are decision variables. The normal processing time of job J_i is denoted by p_i (i.e., the processing time without being influenced by any factor), $i = 1, 2, \ldots, n$. For a given sequence, let C_i be the completion time of job J_i. The aim is to find the optimal starting time of the due-windows, the size of the due-windows, and the sequence of jobs δ such that the following measure is minimized:

$$Z\left(d_i^1, D_i, \delta(i)\right) = \sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}, \tag{1}$$

where $\delta(i)$ denotes the job scheduled in the ith position, $\psi_i > 0$ $(i = 1, 2, \ldots, n)$ denote a position-dependent weight (i.e., weight ψ_i does not correspond with the job but with the position in which some job is scheduled), ψ_0 (ψ_{n+1}) is the unit cost of $d_{\delta(i)}^1$ ($D_{\delta(i)}$), $L_{\delta(i)}$ is the earliness-tardiness of job $J_{\delta(i)}$ $(i = 1, 2, \ldots, n)$, and

$$L_{\delta(i)} = \begin{cases} d_{\delta(i)}^1 - C_{\delta(i)}, & \text{for } d_{\delta(i)}^1 > C_{\delta(i)}, \\ 0, & \text{for } d_{\delta(i)}^1 \leq C_{\sigma(i)} \leq d_{\delta(i)}', \\ C_{\delta(i)} - d_{\delta(i)}', & \text{for } C_{\delta(i)} > d_{\delta(i)}'. \end{cases} \tag{2}$$

Using the three-field notation (Graham et al. [20]), the problem studied here is $1|\mathrm{DIF-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$. Wang et al. [18] considered single-machine scheduling problems with common due-window (CON-DW) and slack due-window (SLK-DW) assignments. They proved that the problems $1|\mathrm{CON-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d^1 + \psi_{n+1} D$ and $1|\mathrm{SLK-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 q^1 + \psi_{n+1} D$ can be solved in $O(n \log n)$ time, respectively.

3. Main Results

Obviously, there exists an optimal sequence δ^* without any machine idle time between the processing of jobs, and the first job in the sequence starts at time zero.

Lemma 1. *There exists an optimal sequence such that $d_{\delta(i)}^1 \leq d_{\delta(i)}' \leq C_{\delta(i)}$.*

Proof. We consider two cases that contradict this optimal property:

Case i: if $d_{\delta(i)}^1 \leq C_{\delta(i)} < d_{\delta(i)}'$, then the total cost for job $J_{\delta(i)}$ is

$$z_{\delta(i)} = \psi_0 d_{\delta(i)}^1 + \psi_{n+1}\left(d_{\delta(i)}' - d_{\delta(i)}^1\right). \tag{3}$$

We shift $d_{\delta(i)}'$ to the left such that $d_{\delta(i)}' = C_{\delta(i)}$, and we have

$$\tilde{z}_{\delta(i)} = \psi_0 d^1_{\delta(i)} + \psi_{n+1}\left(C_{\delta(i)} - d^1_{\delta(i)}\right) < z_{\delta(i)}. \quad (4)$$

Hence, Case i is not an optimal due-window assignment.

Case ii: if $C_{\delta(i)} < d^1_{\delta(i)} \le d'_{\delta(i)}$, then the total cost for job $J_{\delta(i)}$ is

$$z_{\delta(i)} = \psi_i\left(d^1_{\delta(i)} - C_{\delta(i)}\right) + \psi_0 d^1_{\delta(i)} + \psi_{n+1}\left(d'_{\delta(i)} - d^1_{\delta(i)}\right). \quad (5)$$

We shift $d^1_{\delta(i)}$ and $d'_{\delta(i)}$ to the left such that $d^1_{\delta(i)} = d_{\delta(i)}{}' = C_{\delta(i)}$, and we have

$$\tilde{z}_{\delta(i)} = \psi_0 C_{\delta(i)} < z_{\delta(i)}. \quad (6)$$

Hence, Case ii is not an optimal due-window assignment.

To summarise, we have $d^1_{\delta(i)} \le d'_{\delta(i)} \le C_{\delta(i)}$. □

Lemma 2. *For a given sequence δ, the optimal due-window locations $d^1_{\delta(i)}$ and $d_{\delta(i)}{}'$ for job $J_{\delta(i)}$ can be obtained as follows:*

(1) *When* $\min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_i$, *then set* $d^1_{\delta(i)} = d'_{\delta(i)} = 0$

(2) *When* $\min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_0$, *then set* $d^1_{\delta(i)} = d'_{\delta(i)} = C_{\delta(i)}$

(3) *When* $\min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_{n+1}$, *then set* $d^1_{\delta(i)} = 0$ *and* $d'_{\delta(i)} = C_{\delta(i)}$

Proof

(1) When $\min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_i$ and $d^1_{\delta(i)} = d'_{\delta(i)} = 0$, we have $z_{\delta(i)} = \psi_i C_{\delta(i)}$

From Lemma 1, we consider the following two cases:

Case i: if $d^1_{\delta(i)} \le C_{\delta(i)} \le d'_{\delta(i)}$, then the total cost for job $J_{\delta(i)}$ is $\tilde{z}_{\delta(i)} = \psi_0 d^1_{\delta(i)} + \psi_{n+1}(d'_{\delta(i)} - d^1_{\delta(i)}) \ge \psi_i d^1_{\delta(i)} + \psi_i(d'_{\delta(i)} - d^1_{\delta(i)}) = \psi_i d'_{\delta(i)} \ge \psi_i C_{\delta(i)} = z_{\delta(i)}$

Case ii: if $d^1_{\delta(i)} \le d'_{\delta(i)} \le C_{\delta(i)}$, then the total cost for job $J_{\delta(i)}$ is $\tilde{z}_{\delta(i)} = \psi_i(C_{\delta(i)} - d'_{\delta(i)}) + \psi_0 d^1_{\delta(i)} + \psi_{n+1}(d'_{\delta(i)} - d^1_{\delta(i)}) \ge \psi_i(C_{\delta(i)} - d'_{\delta(i)}) + \psi_i d^1_{\delta(i)} + \psi_i(d'_{\delta(i)} - d^1_{\delta(i)}) = \psi_i C_{\delta(i)} = z_{\delta(i)}$

To summarise, if $\min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_i$, then set $d^1_{\delta(i)} = d'_{\delta(i)} = 0$.

Similarly, cases (2) and (3) can be proved. □

Lemma 3. *For a given sequence δ, the optimal due-window locations $d^1_{\delta(i)}$ and $d'_{\delta(i)}$ for job $J_{\delta(i)}$ can be obtained as follows:*

(1) *When* $\psi_i = \psi_0 < \psi_{n+1}$, *then set* $d^1_{\delta(i)} = d'_{\delta(i)} = C_{\delta(j)}$, *where* $j = 0, 1, \ldots, i$

(2) *When* $\psi_i = \psi_{n+1} < \psi_0$, *then set* $d^1_{\delta(i)} = 0, d'_{\delta(i)} = C_{\delta(j)}$, *where* $j = 0, 1, \ldots, i$

(3) *When* $\psi_0 = \psi_{n+1} < \psi_i$, *then set* $d^1_{\delta(i)} = C_{\delta(j)}$ *and* $d'_{\delta(i)} = C_{\delta(i)}$, *where* $j = 0, 1, \ldots, i$

(4) *When* $\psi_0 = \psi_{n+1} = \psi_i$, *then set* $d^1_{\delta(i)} = C_{\delta(j_1)}$ *and* $d'_{\delta(i)} = C_{\delta(j_2)}$, *where* $j_1 = 0, 1, \ldots, i$ *and* $j_2 = j_1, j_1 + 1, \ldots, i$

Proof. The proof is similar to the proof of Lemma 2. □

Lemma 4. *The optimal sequence of the problem* $1|DIF - DW|\sum_{i=1}^n \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)}$ *can be obtained by sequencing the jobs in a nondecreasing order of* p_i, *i.e., the smallest processing time (SPT) first rule.*

Proof. From Lemmas 1–3, the objective function $\sum_{i=1}^n \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)}$ can be transformed into the following three cases: (1) $\sum_{i=1}^n \psi_i C_{\delta(i)}$; (2) $\psi_0 C_{\delta(i)}$; and (3) $\psi_{n+1} C_{\delta(i)}$. For all the three cases, it is easy to verify (by the pairwise interchange method) that sequencing the jobs in a nondecreasing order of p_i is optimal.

Let $A = \{i \mid \min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_i, \quad i = 1, 2, \ldots, n\} \cup \{i \mid \psi_i = \psi_0 < \psi_{n+1}, \quad i = 1, 2, \ldots, n\} \cup \{i \mid \psi_i = \psi_{n+1} < \psi_0, \quad i = 1, 2, \ldots, n\} \cup \{i \mid \psi_i = \psi_0 = \psi_{n+1}, \quad i = 1, 2, \ldots, n\}$, $B = \{i \mid \min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_0, \quad i = 1, 2, \ldots, n\} \cup \{i \mid \psi_0 = \psi_{n+1} < \psi_i, \quad i = 1, 2, \ldots, n\}$, and $C = \{i \mid \min\{\psi_i, \psi_0, \psi_{n+1}\} = \psi_{n+1}, \quad i = 1, 2, \ldots, n\}$; then,

$$\begin{aligned} Z\left(d^1_i, D_i, \delta\right) &= \sum_{i=1}^n \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)} \\ &= \sum_{i\in A} \psi_i \sum_{j=1}^i p_{\delta(j)} + \sum_{i\in B} \psi_i \sum_{j=1}^i p_{\delta(j)} + \sum_{i\in C} \psi_i \sum_{j=1}^i p_{\delta(j)} \\ &= \sum_{i=1}^n \psi'_i \sum_{j=1}^i p_{\delta(j)} \\ &= \sum_{i=1}^n p_{\delta(i)} \sum_{j=i}^n \psi'_i \\ &= \sum_{i=1}^n \lambda_i p_{\delta(i)}, \end{aligned} \quad (7)$$

where

$$\psi_i' = \begin{cases} \psi_i, & i \in A, \\ \psi_0, & i \in B, \\ \psi_{n+1}, & i \in C. \end{cases} \quad (8)$$

And

$$\lambda_i = \sum_{j=i}^{n} \psi_i'. \quad (9)$$

□

Remark 1. Obviously, $\lambda_i = \sum_{j=i}^{n} \psi_i'$ is a decreasing function on i; from Hardy et al. [21], the optimal sequence can be obtained by the SPT rule, and it is the same as Lemma 4.

From Lemmas 1–4, a polynomial-time algorithm can be proposed for the $1|\mathrm{DIF-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ problem.

Theorem 1. *Algorithm 1 solves the problem* $1|\mathrm{DIF-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ *in* $O(n \log n)$ *time.*

Proof. Optimality can be guaranteed by Lemmas 1–4. In Algorithm 1, Step 1 needs $O(n \log n)$ time by the SPT rule; Steps 2 and 3 can be performed in $O(n)$ time. Thus, the total time for Algorithm 1 is $O(n \log n)$.

In order to illustrate Algorithm 1 for the problem $1|\mathrm{DIF-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$, we present the following instance. □

Example 1. The data are as follows: $n = 10$, $p_1 = 15$, $p_2 = 20$, $p_3 = 26$, $p_4 = 24$, $p_5 = 17$, $p_6 = 28$, $p_7 = 21$, $p_8 = 25$, $p_9 = 27$, $p_{10} = 14$, $\psi_0 = 14$, $\psi_1 = 7$, $\psi_2 = 20$, $\psi_3 = 12$, $\psi_4 = 24$, $\psi_5 = 14$, $\psi_6 = 22$, $\psi_7 = 15$, $\psi_8 = 8$, $\psi_9 = 19$, $\psi_{10} = 12$, and $\psi_{11} = 50$.

Now, we can solve the problem $1|\mathrm{DIFW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ according to Algorithm 1 as follows:

Step 1: according to Lemma 4, the optimal sequence is $\delta^* = (J_{10}, J_1, J_5, J_2, J_7, J_4, J_8, J_3, J_9, J_6)$

Step 2: for the optimal sequence $\delta^* = (J_{10}, J_1, J_5, J_2, J_7, J_4, J_8, J_3, J_9, J_6)$, the completion time of all jobs is $C_{10} = 14$, $C_1 = 29$, $C_5 = 46$, $C_2 = 66$, $C_7 = 87$, $C_4 = 111$, $C_8 = 136$, $C_3 = 162$, $C_9 = 189$, and $C_6 = 217$, and the optimal due-window locations $d_{\delta(i)}^1$ and $d_{\delta(i)}'$ for each job are given in Table 1

Step 3: the optimal due-window sizes are $D_{\delta(i)} = 0$ $(i = 1, 2, \ldots, 10)$, $\lambda_1 = 123$, $\lambda_2 = 116$, $\lambda_3 = 102$, $\lambda_4 = 90$, $\lambda_5 = 76$, $\lambda_6 = 62$, $\lambda_7 = 48$, $\lambda_8 = 34$, $\lambda_9 = 26$, and $\lambda_{10} = 12$, and the objective function is $Z = \sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)} = \sum_{i=1}^{n} \lambda_i p_{\delta(i)} = 13202$.

4. An Extension

In this section, the problem $1|\mathrm{DIF-DW}|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ is extended to a setting of general position-dependent processing time. Let p_i^A be the actual processing time of J_i; under the general position-dependent processing time setting, the actual processing time of J_i is $p_i^A = \theta(i, r)$ if it is assigned to position r, $i, r = 1, \ldots, n$. Thus, the input for the problem contains a matrix of $(n \times n)$ job-position values. Biskup [22] introduced a job-independent learning effect model in which $p_i^A = \theta(i, r) = p_i r^{\alpha}$, where $\alpha \leq 0$ is the learning index (see also Wang et al. [23]). Mosheiov and Sidney [24] introduced job-dependent learning effects, i.e., $p_i^A = \theta(i, r) = p_i r^{\alpha_i}$, where $\alpha_i \leq 0$ is the job-dependent learning index of job J_i. Wang et al. [25] introduced truncated job-dependent learning effects, i.e., $p_i^A = \theta(i, r) = p_i \max\{r^{\alpha_i}, \beta\}$, where $0 < \beta < 1$ is a truncation parameter. We refer the reader to the survey of Azzouz et al. [26] on scheduling problems with learning effects.

From (7), we have

$$Z(d_i^1, D_i, \delta) = \sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)} = \sum_{i=1}^{n} \lambda_i \theta(i, r), \quad (10)$$

where λ_i are given by (9).

From (10), the optimal sequence of the problem $1|\mathrm{DIF-DW}, p_i^A = \theta(i, r)|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ can be obtained by solving the following assignment problem:

$$\begin{aligned} \text{Min} \quad & \sum_{i=1}^{n} \sum_{r=1}^{n} \lambda_r \theta(i, r) x_{ir}, \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_{ir} = 1, \quad r = 1, \ldots, n, \\ & \sum_{r=1}^{n} x_{ir} = 1, \quad i = 1, \ldots, n, \end{aligned} \quad (11)$$

where $\lambda_r, r = 1, \ldots, n$, are given by (9), and

$$x_{ir} = \begin{cases} 1, & \text{if job } J_i \text{ is assigned to position } r, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

Based on the above analysis, the solution procedure of the problem $1|\mathrm{DIF-DW}, p_i^A = \theta(i, r)|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ can be summarized as follows.

Theorem 2. *Algorithm 2 solves the problem* $1|\mathrm{DIF-DW}, p_i^A = \theta(i, r)|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$ *in* $O(n^3)$ *time.*

Proof. Optimality is guaranteed by Lemmas 1–3 and the above analysis. In Algorithm 2, Step 1 needs $O(n^3)$ time by the SPT rule; Steps 2 and 3 can be performed in $O(n)$ time. Thus, the total time for Algorithm 2 is $O(n^3)$. In order to illustrate Algorithm 2 for the problem $1|\mathrm{DIF-DW}, p_i^A = \theta(i, r)|\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d_{\delta(i)}^1 + \psi_{n+1} D_{\delta(i)}$, we present the following instance. □

Example 2. The data are as follows: $n = 8$, $\psi_0 = 14$, $\psi_1 = 8$, $\psi_2 = 18$, $\psi_3 = 12$, $\psi_4 = 24$, $\psi_5 = 10$, $\psi_6 = 20$, $\psi_7 = 15$, $\psi_8 = 7$, and $\psi_9 = 21$. The job-dependent processing time is given in Table 2.

Step 1: obtain the optimal sequence by the SPT rule (see Lemma 4)
Step 2: calculate the completion time of each job under the optimal sequence, and determine the optimal due-window locations $d^1_{\delta(i)}$ and $d'_{\delta(i)}$ for each job according to Lemmas 2 and 3
Step 3: obtain the optimal due-window size by setting $D_{\delta(i)} = d'_{\delta(i)} - d^1_{\delta(i)}$ $(i = 1, 2, \ldots, n)$, and calculate the objective function $\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)}$ by equation (7)

Algorithm 1

Table 1: Results of the optimal due-window location.

Job $J_{\delta(i)}$	Job $J_{\delta(i)}$	$d^1_{\delta(i)}$	$d_{\delta(i)}{}'$
$J_{\delta(1)}$	J_{10}	$d^1_{\delta(i)} = 0$	$d_{\delta(1)}{}' = 0$
$J_{\delta(2)}$	J_1	$d^1_{\delta(2)} = C_{\delta(2)} = 29$	$d_{\delta(2)}{}' = C_{\delta(2)} = 29$
$J_{\delta(3)}$	J_5	$d^1_{\delta(3)} = 0$	$d_{\delta(3)}{}' = 0$
$J_{\delta(4)}$	J_2	$d^1_{\delta(4)} = C_{\delta(4)} = 66$	$d_{\delta(4)}{}' = C_{\delta(4)} = 66$
$J_{\delta(5)}$	J_7	$d^1_{\delta(5)} \in \{C_{\delta(i)} \mid i = 0, 1, 2, 3, 4, 5\}$ i.e., $d^1_{\delta(5)} \in \{0, 14, 29, 46, 66, 87\}$	$d_{\delta(5)}{}' = d^1_{\delta(5)}$
$J_{\delta(6)}$	J_4	$d^1_{\delta(6)} = C_{\delta(6)} = 111$	$d_{\delta(6)}{}' = C_{\delta(6)} = 111$
$J_{\delta(7)}$	J_8	$d^1_{\delta(7)} = C_{\delta(7)} = 136$	$d_{\delta(7)}{}' = C_{\delta(7)} = 136$
$J_{\delta(8)}$	J_3	$d^1_{\delta(8)} = 0$	$d_{\delta(8)}{}' = 0$
$J_{\delta(9)}$	J_9	$d^1_{\delta(9)} = C_{\delta(9)} = 189$	$d_{\delta(9)}{}' = C_{\delta(9)} = 189$
$J_{\delta(10)}$	J_6	$d^1_{\delta(10)} = 0$	$d_{\delta(10)}{}' = 0$

Step 1: solve assignment problem (11) to obtain the optimal sequence
Step 2: calculate the completion time of each job under the optimal sequence, and determine the optimal due-window locations $d^1_{\delta(i)}$ and $d'_{\delta(i)}$ for each job according to Lemmas 2 and 3
Step 3: obtain the optimal due-window size by setting $D_{\delta(i)} = d'_{\delta(i)} - d^1_{\delta(i)}$ $(i = 1, 2, \ldots, n)$, and calculate the objective function $\sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)}$ by assignment problem (11)

Algorithm 2

Table 2: Date of Example 2.

J_i	r							
	1	2	3	4	5	6	7	8
J_1	6	7	11	5	6	22	13	21
J_2	8	14	7	8	11	17	12	6
J_3	9	11	13	32	7	10	12	8
J_4	17	22	19	10	5	9	13	7
J_5	16	8	15	14	11	17	13	14
J_6	18	17	31	14	8	23	15	20
J_7	15	12	18	19	8	16	21	13
J_8	13	17	24	16	18	16	13	15

Table 3: Results of the optimal due-window location.

Job $J_{\delta(i)}$	Job $J_{\delta(i)}$	$d^1_{\delta(i)}$	$d_{\delta(i)}{}'$
$J_{\delta(1)}$	J_3	$d^1_{\delta(1)} = 0$	$d_{\delta(1)}{}' = 0$
$J_{\delta(2)}$	J_5	$d^1_{\delta(2)} = C_{\delta(2)} = 17$	$d_{\delta(2)}{}' = C_{\delta(2)} = 17$
$J_{\delta(3)}$	J_2	$d^1_{\delta(3)} = 0$	$d_{\delta(3)}{}' = 0$
$J_{\delta(4)}$	J_1	$d^1_{\delta(4)} = C_{\delta(4)} = 29$	$d_{\delta(4)}{}' = C_{\delta(4)} = 29$
$J_{\delta(5)}$	J_6	$d^1_{\delta(5)} = 0$	$d_{\delta(5)}{}' = 0$
$J_{\delta(6)}$	J_4	$d^1_{\delta(6)} = C_{\delta(6)} = 46$	$d_{\delta(6)}{}' = C_{\delta(6)} = 46$
$J_{\delta(7)}$	J_8	$d^1_{\delta(7)} = C_{\delta(7)} = 59$	$d_{\delta(7)}{}' = C_{\delta(7)} = 59$
$J_{\delta(8)}$	J_7	$d^1_{\delta(8)} = 0$	$d_{\delta(8)}{}' = 0$

5. Conclusion and Future Work

This study addressed the due-window (DIF-DW) assignment scheduling problem under the consideration of position-dependent weights. The goal is to determine the optimal sequence, the optimal due-window location, and size such that the total penalty (including the earliness, tardiness, due-window starting time, and due-window size of all jobs) is minimized. It was proved that the problem can be solved in polynomial time. The proposed model was also extended to the general position-dependent processing time, and the polynomial-time solution was provided. Further extensions are considering the above problems in the setting of m-machine flow-shop and m-identical (unrelated) parallel machines (Hsu and Liao [27]), studying the scheduling with two-agent resource-dependent release time (Liu and Duan [28]), or investigating scheduling with rate-modifying activity under deterioration effect (Xue and Zhang [29]).

Step 1: by (9), we have $\lambda_1 = 93$, $\lambda_2 = 85$, $\lambda_3 = 71$, $\lambda_4 = 59$, $\lambda_5 = 45$, $\lambda_6 = 35$, $\lambda_7 = 21$, and $\lambda_8 = 7$. According to assignment problem (11), the optimal sequence is $\delta^* = (J_3, J_5, J_2, J_1, J_6, J_4, J_8, J_7)$.

Step 2: for the optimal sequence $\delta^* = (J_3, J_5, J_2, J_1, J_6, J_4, J_8, J_7)$, the completion time of all jobs is $C_3 = 9$, $C_5 = 17$, $C_2 = 24$, $C_1 = 29$, $C_6 = 37$, $C_4 = 46$, $C_8 = 59$, and $C_7 = 72$, and the optimal due-window locations $d^1_{\delta(i)}$ and $d_{\delta(i)}{}'$ for each job are given in Table 3.

Step 3: the optimal due-window sizes are $D_{\delta(i)} = 0$ $(i = 1, 2, \ldots, 8)$, and the objective function is $Z = \sum_{i=1}^{n} \psi_i L_{\delta(i)} + \psi_0 d^1_{\delta(i)} + \psi_{n+1} D_{\delta(i)} = 3348$.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Natural Science Foundation of Liaoning Province (2020-MS-233).

References

[1] H. Gong, B. Zhang, and W. Peng, "Scheduling and common due date assignment on a single parallel-batching machine with batch delivery," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 464390, 7 pages, 2015.

[2] A. Janiak, W. A. Janiak, T. Krysiak, and T. Kwiatkowski, "A survey on scheduling problems with due windows," *European Journal of Operational Research*, vol. 242, no. 2, pp. 347–357, 2015.

[3] X.-N. Geng, J.-B. Wang, and D. Bai, "Common due date assignment scheduling for a no-wait flowshop with convex resource allocation and learning effect," *Engineering Optimization*, vol. 51, no. 8, pp. 1301–1323, 2019.

[4] S. D. Liman, S. S. Panwalkar, and S. Thongmee, "Determination of common due window location in a single machine scheduling problem," *European Journal of Operational Research*, vol. 93, no. 1, pp. 68–74, 1996.

[5] S. D. Liman, S. S. Panwalkar, and S. Thongmee, "Common due window size and location determination in a single machine scheduling problem," *The Journal of the Operational Research Society*, vol. 49, no. 9, pp. 1007–1010, 1998.

[6] G. Mosheiov and A. Sarig, "Minmax scheduling problems with a common due-window," *Computers & Operations Research*, vol. 36, no. 6, pp. 1886–1892, 2009.

[7] Y. Yin, T. C. E. Cheng, C.-J. Hsu, and C.-C. Wu, "Single-machine batch delivery scheduling with an assignable common due window," *Omega*, vol. 41, no. 2, pp. 216–225, 2013.

[8] Y. Yin, T. C. E. Cheng, J. Wang, and C.-C. Wu, "Single-machine common due window assignment and scheduling to minimize the total cost," *Discrete Optimization*, vol. 10, no. 1, pp. 42–53, 2013.

[9] J. Liu, Y. Wang, and X. Min, "Single-machine scheduling with common due-window assignment for deteriorating jobs," *Journal of the Operational Research Society*, vol. 65, no. 2, pp. 291–301, 2014.

[10] J.-B. Wang and M.-Z. Wang, "Single-machine due-window assignment and scheduling with learning effect and resource-dependent processing times," *Asia-Pacific Journal of Operational Research*, vol. 31, no. 5, Article ID 1450036, 2014.

[11] G. Mosheiov and D. Oron, "Job-dependent due-window assignment based on a common flow allowance," *Foundations of Computing and Decision Sciences*, vol. 35, no. 3, pp. 185–195, 2010.

[12] J.-B. Wang, L. Liu, and C. Wang, "Single machine SLK/DIF due window assignment problem with learning effect and deteriorating jobs," *Applied Mathematical Modelling*, vol. 37, no. 18-19, pp. 8394–8400, 2013.

[13] M. Ji, K. Chen, J. Ge, and T. C. E. Cheng, "Group scheduling and job-dependent due window assignment based on a common flow allowance," *Computers & Industrial Engineering*, vol. 68, pp. 35–41, 2014.

[14] Y. Yin, T. C. E. Cheng, C.-C. Wu, and S.-R. Cheng, "Single-machine due window assignment and scheduling with a common flow allowance and controllable job processing time," *Journal of the Operational Research Society*, vol. 65, no. 1, pp. 1–13, 2013.

[15] Y. Yin, D.-J. Wang, T. C. E. Cheng, and C.-C. Wu, "Bi-criterion single-machine scheduling and due-window assignment with common flow allowances and resource-dependent processing times," *Journal of the Operational Research Society*, vol. 67, no. 9, pp. 1169–1183, 2016.

[16] D. Wang, Y. Yin, and T. C. E. Cheng, "A bicriterion approach to common flow allowances due window assignment and scheduling with controllable processing times," *Naval Research Logistics (NRL)*, vol. 64, no. 1, pp. 41–63, 2017.

[17] B. Mor and G. Mosheiov, "Minsum and minmax scheduling on a proportionate flowshop with common flow-allowance," *European Journal of Operational Research*, vol. 254, no. 2, pp. 360–370, 2017.

[18] J.-B. Wang, B. Zhang, L. Li, D. Bai, and Y.-B. Feng, "Due-window assignment scheduling problems with position-dependent weights on a single machine," *Engineering Optimization*, vol. 52, no. 2, pp. 185–193, 2020.

[19] D.-L. Yang, C.-J. Lai, and S.-J. Yang, "Scheduling problems with multiple due windows assignment and controllable

processing times on a single machine," *International Journal of Production Economics*, vol. 150, pp. 96–103, 2014.

[20] 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.

[21] G. H. Hardy, J. E. Littlewood, and G. Polya, *Inequalities*, Cambridge University Press, Cambridge, UK, 1967.

[22] D. Biskup, "Single-machine scheduling with learning considerations," *European Journal of Operational Research*, vol. 115, no. 1, pp. 173–178, 1999.

[23] J.-B. Wang, J. Xu, and J. Yang, "Bi-criterion optimization for flow shop with a learning effect subject to release dates," *Complexity*, vol. 2018, Article ID 9149510, 12 pages, 2018.

[24] G. Mosheiov and J. B. Sidney, "Scheduling with general job-dependent learning curves," *European Journal of Operational Research*, vol. 147, no. 3, pp. 665–670, 2003.

[25] X.-R. Wang, J.-B. Wang, J. Jin, and P. Ji, "Single machine scheduling with truncated job-dependent learning effect," *Optimization Letters*, vol. 8, no. 2, pp. 669–677, 2014.

[26] A. Azzouz, M. Ennigrou, and L. B. Said, "Scheduling problems under learning effects: classification and cartography," *International Journal of Production Research*, vol. 56, no. 3, pp. 1–20, 2017.

[27] C.-L. Hsu and J.-R. Liao, "Two parallel-machine scheduling problems with function constraint," *Discrete Dynamics in Nature and Society*, vol. 2020, Article ID 2717095, 6 pages, 2020.

[28] P. Liu and L. Duan, "A note on two-agent scheduling with resource dependent release times on a single machine," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 503297, 4 pages, 2015.

[29] P. Xue and Y. Zhang, "Single-machine scheduling with upper bounded maintenance time under the deteriorating effect," *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 756251, 6 pages, 2013.