Hindawi
Discrete Dynamics in Nature and Society
Volume 2020, Article ID 9260479, 7 pages
https://doi.org/10.1155/2020/9260479

Research Article

Due-Window Assignment and Resource Allocation Scheduling with Truncated Learning Effect and Position-Dependent Weights

Shan-Shan Lin

School of Business and Management, Fujian Jiangxia University, Fuzhou 350108, China

Correspondence should be addressed to Shan-Shan Lin; 1257290990@qq.com

Received 19 July 2020; Revised 17 August 2020; Accepted 15 September 2020; Published 8 October 2020

Academic Editor: Chin-Chia Wu

This paper studies single-machine due-window assignment scheduling problems with truncated learning effect and resource allocation simultaneously. Linear and convex resource allocation functions under common due-window (CONW) assignment are considered. The goal is to find the optimal due-window starting (finishing) time, resource allocations and job sequence that minimize a weighted sum function of earliness and tardiness, due window starting time, due window size, and total resource consumption cost, where the weight is position-dependent weight. Optimality properties and polynomial time algorithms are proposed to solve these problems.

1. Introduction

Scheduling models and problems with learning effects (see Biskup [1]; Lu et al. [2]; Azzouz et al. [3]; Wang et al. [4]) and/or resource allocations (see Shabtay and Steiner [5]; Yang et al. [6]) have become popular topics for scheduling researchers in recent years. Scheduling with learning effects and resource allocations simultaneously was introduced by Wang et al. [7], who focused on single-machine scheduling problems. Lu et al. [8] studied single-machine due-date assignment scheduling with learning effects and resource allocations. They proved that several problems can be solved in polynomial time. Wang and Wang [9] and Li et al. [10] considered common and slack due-window assignment problems with learning effects and resource allocations. Wang and Wang [11] considered single-machine scheduling problems with learning effects and convex resource allocation function. For the scheduling criterion (the total resource compression criterion) minimization subject to the constraint that the total resource compression criterion (the scheduling criterion) is less than or equal to a fixed constant, they proved that the problems can be solved in polynomial time. Wang et al. [12] and Liu and Jiang [13] considered due-date assignment scheduling with job-dependent learning effects and resource allocation. Liu and Jiang [14] considered flow shop due-date assignment scheduling with resource allocation and learning effect. Shi and Wang [15] considered flow shop due-window assignment scheduling with resource allocation and learning effect.

In recent years, many researchers focused on the study of scheduling with due-window, where a time interval is assumed, such that jobs completed within this interval are not penalized (Janiak et al. [16] and Wang et al. [17]). Wang et al. [18] considered the single-machine due-window scheduling problems with position-dependent weights. For the weighted sum of earliness and tardiness, due window starting time, and due window size, where the weight only dependent on its position in a sequence (i.e., a position-dependent weight), they proved that the problems can be solved in polynomial time. In this study, we continue the work of Wang et al. [18], i.e., we consider the due-window assignment scheduling problems with learning effect and resource allocation in the single-machine environment. The goal is to find the optimal due-window starting (finishing) time, resource allocations, and job sequence such that a sum of scheduling cost (including weighted sum function of earliness and tardiness, due window starting time, due window size, where the weight is position-dependent weight) and total resource consumption cost is minimized. The contributions of this paper are given as follows. (1) The structural properties of single-machine scheduling problems are derived. (2) For the linear resource allocation, we proved that the sum of scheduling cost and total resource consumption cost can be solved in polynomial time. For the convex resource allocation, three versions of scheduling cost and total resource consumption cost can be solved in polynomial time respectively. (3) It is further extended the model to the case with slack due-window (SLKW) assignment model.

The rest of the article is organized as follows: In Section 2, we introduce the problem. In Sections 3 and 4, we provide some properties to optimally solve these problems under linear and convex resource allocation. In Section 5, we conclude the paper.

2. Problem Formulation

We study a scheduling problem consisting of a set of n independent jobs $N = \{J_1, J_2, \ldots, J_n\}$ that need to be processed on a single machine. For the linear resource allocation, the actual processing time of job J_j is

$$P_j^A = \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u_j, \quad (1)$$

where $\widetilde{p}_j$ is the basic processing time of job J_j (i.e., the processing time without any resource allocation and truncated learning effect), $\alpha_j \le 0$ is the job-dependent learning rate (Mosheiov and Sidney [19]) of job J_j, $0 < \delta < 1$ is a truncation parameter (Wang et al. [20]), β_j is the compression rate of job J_j, and $u_j$is the amount of resource allocated to job J_j and satisfies $0 \le u_j \le \overline{u}_j \le ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/\beta_j)$.

For the convex resource allocation, the actual processing time of job J_j is

$$P_j^A = \left(\frac{\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}}{u_j}\right)^{\eta}, \quad (2)$$

where $\eta > 0$ is a constant, i.e., P_j^A is a convex decreasing function of resource u_j.

Let $[d_1, d_2]$be the common due-window for all jobs, where $d_1 \ge 0\,(d_2, d_1 \le d_2)$ denotes the starting (finishing) time of the common due window. The length of the due-window is $D = d_2 - d_1$. Both d_1 and $d_2$are decision variables in this paper. The goal of this paper is to find jointly the optimal due-window location, the optimal resource allocation and sequence π such that the following objective function is minimized:

$$Z(d_1, d_2, \pi) = \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}, \quad (3)$$

where $[j]$ denotes the job scheduled in jth position, $w_j\,(j = 0, 1, 2, \ldots, n, n+1)$ denotes a position- dependent weight, $L_{[j]}$ is the earliness-tardiness of job $J_{[j]}$, and

$$L_{[j]} = \begin{cases} d_1 - C_{[j]}, & \text{for} \quad d_1 > C_{[j]}, \\ 0, & \text{for} \quad d_1 \le C_{[j]} \le d_2, \\ C_{[j]} - d_2, & \text{for} \quad C_{[j]} > d_2, \end{cases} \quad (4)$$

where $C_{[j]}$ is the completion time of job $J_{[j]}$, $j = 1, 2, \ldots, n$. Using the three-field classification, the problem can be denoted as $1|\mathrm{CONW}, P_j^A| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}$, where $P_j^A \in \{\widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u_j, ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^{\eta}\}$ (Graham et al. [21]), where CONW denotes the common due-window assignment. Wang et al. [18] considered single-machine scheduling problems with CONW and slack due-window (SLKW) assignments problems $1|\mathrm{CONW}| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$ and $1|\mathrm{SLKW}| \sum_{j=1}^{n} w_j L_{[j]} + w_0 q' + w_{n+1} D$; for the SLKW model, $[d_j', d_j'']$ is the due-window of job J_j such that $d_j' \le d_j''$, where $d_j' = P_j^A + q', d_j'' = P_j^A + q''$, $j = 1, 2, \ldots, n \lim_{x \to \infty}$, q' and q'' are decision variables and $D = q'' - q'$. Wang et al. [18] proved that these both problems can be solved in $O(n \log n)$ time, respectively.

3. Linear Resource Allocation

Lemma 1 [Wang et al. [18]]. *For any given sequence π, there exists an optimal sequence in which $d_1 = C_{[k]}$ for some k and $d_2 = C_{[l]}$ for some $l, l \ge k$, where $\sum_{i=0}^{k-1} w_i \le w_{n+1} \le \sum_{i=0}^{k-1} w_i$, and $\sum_{i=l+1}^{n} w_i \le w_{n+1} \le \sum_{i=l}^{n} w_i$.*

Lemma 2. *The objective function of the problem $1|CONW, P_j^A| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}$ can be written as*

$$\sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]} = \sum_{j=1}^{n} \xi_j P_{[j]}^A + \sum_{j=1}^{n} v_{[j]} u_{[j]}, \quad (5)$$

where

$$\xi_j = \left\{ \sum_{h=0}^{j-1} w_h, \quad \text{for } j = 1, 2, \ldots, k; w_{n+1}, \quad \text{for } j = k+1, k+2, \ldots, l; \sum_{h=0}^{j-1} w_h, \quad \text{for } j = l+1, l+2, \ldots, n. \right. \quad (6)$$

Proof. From Lemma 1, we have

$$
\begin{aligned}
Z(d_1, d_2, \pi) &= w_0 C_{[k]} + \sum_{j=1}^{k-1} w_j \left(C_{[k]} - C_{[j]}\right) + \sum_{j=l+1}^{n} w_j \left(C_{[j]} - C_{[l]}\right) + w_{n+1}\left(C_{[l]} - C_{[k]}\right) + \sum_{j=1}^{n} v_{[j]} u_{[j]} \\
&= w_0 \sum_{j=1}^{k} P_{[j]}^{A} + \sum_{j=1}^{k-1} w_j \left(\sum_{h=j+1}^{k} P_{[h]}^{A}\right) + \sum_{j=l+1}^{n} w_j \left(\sum_{h=j+1}^{l} P_{[h]}^{A}\right) + w_{n+1}\left(\sum_{h=k+1}^{l} P_{[h]}^{A}\right) + \sum_{j=1}^{n} v_{[j]} u_{[j]} \\
&= \sum_{j=1}^{k} P_{[j]}^{A} \left(\sum_{h=0}^{j-1} w_h\right) + \sum_{j=l+1}^{n} P_{[j]}^{A} \left(\sum_{h=j}^{n} w_h\right) + w_{n+1}\left(\sum_{j=k+1}^{l} P_{[j]}^{A}\right) + \sum_{j=1}^{n} v_{[j]} u_{[j]} \\
&= \sum_{j=1}^{n} \xi_j P_{[j]}^{A} + \sum_{j=1}^{n} v_{[j]} u_{[j]},
\end{aligned}
\tag{7}
$$

where ξ_j $(j = 1, 2, \ldots, n)$ are given by (6).

From Lemma 2, we have

$$
\begin{aligned}
Z(d_1, d_2, \pi) &= \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]} \\
&= \sum_{j=1}^{n} \xi_j \left(\widetilde{p}_{[j]} \max\{r^{\alpha_j}, \delta\} - \beta_{[j]} u_{[j]}\right) + \sum_{j=1}^{n} v_{[j]} u_{[j]} \\
&= \sum_{j=1}^{n} \xi_j \widetilde{p}_{[j]} \max\{r^{\alpha_{[j]}}, \delta\} + \sum_{j=1}^{n} \left(v_{[j]} - \xi_j \beta_{[j]}\right) u_{[j]}.
\end{aligned}
\tag{8}
$$

From (8), for a given sequence, the optimal resource allocation $u_{[j]}^{*}$ with $v_{[j]} - \xi_j \beta_{[j]} < 0$ should be $\overline{u}_{[j]}$; otherwise, $u_{[j]}^{*} = 0$, i.e., the optimal resource allocation of job $J_{[j]}$ is

$$
u_{[j]}^{*} = \begin{cases} 0, & \text{if,} \quad v_{[j]} - \xi_j \beta_{[j]} \geq 0, \\ \overline{u}_{[j]}, & \text{if,} \quad v_{[j]} - \xi_j \beta_{[j]} < 0. \end{cases} \tag{9}
$$

For a given sequence, from (8), we can obtain the optimal resource allocation. In order to determine the optimal sequence, let $x_{jr} = 1$ if job $J_j\,(j = 1, 2, \ldots, n)$ is scheduled at position r $(r = 1, 2, \ldots, n)$, and $x_{jr} = 0$, otherwise. Then, the problem $1|\text{CONW}, P_j^A = \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}$ can be solved by the following assignment problem:

$$
\text{Min} \sum_{r=1}^{n} \sum_{j=1}^{n} \Psi_{jr} x_{jr}, \tag{10}
$$

$$
\text{S.T.} \sum_{r=1}^{n} x_{jr} = 1, \quad j = 1, 2, \ldots, n, \tag{11}
$$

$$
\sum_{j=1}^{n} x_{jr} = 1, \quad r = 1, 2, \ldots, n, \tag{12}
$$

$$
x_{jr} = 0 \text{ or } 1, \; j = 1, 2, \ldots, n, \tag{13}
$$

where

$$
\Psi_{jr} = \begin{cases} \xi_r \widetilde{p}_j \max\{r^{\alpha_j}, \delta\}, & \text{if } v_j - \xi_r \beta_j \geq 0, \; j, r = 1, 2, \ldots, n, \\ \xi_r \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} + \left(v_j - \xi_r \beta_j\right) \overline{u}_j, & \text{if } v_{[j]} - \xi_r \beta_j < 0, \; j, r = 1, 2, \ldots, n. \end{cases} \tag{14}
$$

And ξ_r $(r = 1, 2, \ldots, n)$ are given by (6).

Based on the above analysis, the problem $1|\text{CONW}, P_j^A = \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}$ can be optimally solved by the following algorithm. □

Algorithm 1

Step 1. Calculate the indices k and l according to Lemma 1.

Step 2. Calculate the values Ψ_{jr} by using (14).

Step 3. Solve the assignment problem (10)–(13) to determine the optimal job sequence.

Step 4. Calculate the optimal resource allocation by (7).

Step 5. Calculate $d_1 = C_{[k]}$, $d_2 = C_{[l]}$.

Theorem 1. *The problem* $1|\text{CONW}, P_j^A = \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u| \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]}$ *can be solved by Algorithm 1 in* $O(n^3)$ *time.*

Proof. The correctness of Algorithm 1 follows from the above analysis. The time complexity of Step 1 is $O(n)$ time, Step 2 is $O(n^2)$ time, Step 3 is $O(n^3)$ time, Step 4 is $O(n)$, and 5 is $O(n)$ time. Thus, the overall computational complexity of Algorithm 1 is $O(n^3)$.

In order to illustrate Algorithm 1 for $1|\mathrm{CONW}, P_j^A = \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} - \beta_j u| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$, we present the following example. □

Example 1. Data: $n = 7, \delta = 0.6, w_0 = 19, w_1 = 20, w_2 = 12, w_3 = 7, w_4 = 14, w_5 = 24, w_6 = 22, w_7 = 15, w_8 = 22$, and the other corresponding parameters shown in Table 1.

Solution:

Step 1. According to Lemma 1, $k = 1, l = 6$.

Step 2. From (5), $\xi_1 = 19$, $\xi_2 = \xi_3 = \xi_4 = \xi_5 = \xi_6 = 22$, $\xi_7 = 15$, and the values Ψ_{jr} are given in Table 2.

Step 3. Stemming from the assignment problem (8)–(11), the optimal job sequence is $\pi = (J_4, J_7, J_2, J_6, J_1, J_5, J_3)$.

Step 4. From (7), the optimal resource allocation is $u_4 = 1$, $u_7 = 5$, $u_2 = 3$, $u_6 = 4$, $u_1 = 5$, $u_5 = 2$, $u_3 = 0$.

Step 5. Calculate $d_1 = C_{[1]} = C_4 = 7$, $d_2 = C_{[6]} = C_5 = 22.67617$, and $\sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n w_j L_{[j]} = 886.8757$.

Table 1: Data for Example 1.

J_j	J_1	J_2	J_3	J_4	J_5	J_6	J_7
$\widetilde{p}_j$	23	17	19	10	18	16	9
α_j	−0.32	−0.24	−0.33	−0.25	−0.28	−0.3	−0.29
β_j	2	3	1	3	4	2	1
v_j	3	14	30	9	6	15	20
$\overline{u}_j$	5	3	6	1	2	4	5

4. Convex Resource Allocation

4.1. Problem $1|\mathrm{CONW}, P_j^A = (\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}/u_j)| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$. From Lemma 2 and $P_j^A = ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^\eta$, we have

$$\sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]} = \sum_{j=1}^n \xi_j \left(\frac{\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}}{u_j} \right)^\eta + \sum_{j=1}^n v_{[j]} u_{[j]}, \tag{15}$$

where ξ_j $(j = 1, 2, \ldots, n)$ are given by (6).

By taking the first derivative of the objective given by (15) with respect to $u_{[j]}$, equating it to zero and solving it for $J_{[j]}$, we have (16).

Lemma 3. *For a given sequence, the optimal resource allocation of the problem* $1|\mathrm{CONW}, P_j^A = (\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}/u_j)| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$ *is*

$$u_{[j]}^* = \left(\frac{\eta \xi_j}{v_{[j]}} \right)^{(1/\eta+1)} \times \left(\widetilde{p}_{[j]} \max\{ j^{\alpha_{[j]}}, \delta\} \right)^{(\eta/\eta+1)}. \tag{16}$$

By substituting (16) into (15), we have

$$\sum_{j=1}^n \xi_j P_{[j]}^A + \sum_{j=1}^n v_{[j]} u_{[j]} = \left(\eta^{(-\eta/\eta+1)} + \eta^{(1/\eta+1)} \right) \sum_{j=1}^n \left(v_{[j]} \widetilde{p}_{[j]} \max\{ j^{\alpha_{[j]}}, \delta\} \right)^{(\eta/\eta+1)} \left(\xi_j \right)^{(1/\eta+1)}. \tag{17}$$

Let

$$\Psi_{jr} = \left(\eta^{(-\eta/\eta+1)} + \eta^{(1/\eta+1)} \right) \left(v_j \widetilde{p}_j \max\{r^{\alpha_j}, \delta\} \right)^{(\eta/\eta+1)} (\xi_r)^{(1/\eta+1)}, \tag{18}$$

where ξ_r $(r = 1, 2, \ldots, n)$ *are given by (6).*

As in Section 3, for the problem $1|\mathrm{CONW}, P_j^A = (\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}/u_j)| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$, *we can propose the following algorithm:*

Algorithm 2

Step 1. Calculate the indices k and l according to Lemma 1.

Step 2. Calculate the values Ψ_{jr} by using (18).

Step 3. Solve the assignment problem (10)–(13) to determine the optimal job sequence.

Step 4. Calculate the optimal resource allocation by (16).

Step 5. Calculate $d_1 = C_{[k]}$, $d_2 = C_{[l]}$.

Theorem 2. *Algorithm 2 solves the problem* $1|\mathrm{CONW}, P_j^A = (\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}/u_j)| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$ *in* $O(n^3)$ *time.*

In order to illustrate Algorithm 2 for $1|\mathrm{CONW}, P_j^A = (\widetilde{p}_j \max\{r^{\alpha_j}, \delta\}/u_j)| \sum_{j=1}^n w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^n v_{[j]} u_{[j]}$, *we present the following example.*

Example 2. Consider $n = 7, \delta = 0.6, \eta = 2, w_0 = 9, w_1 = 8, w_2 = 12, w_3 = 7, w_4 = 14, w_5 = 24, w_6 = 5, w_7 = 15, w_8 = 22$, and the other corresponding parameters shown in Table 3.

Solution:

Step 1. According to Lemma 1, $k = 2, l = 5$.

Step 2. From (5), $\xi_1 = 9$, $\xi_2 = 17$, $\xi_3 = \xi_4 = \xi_5 = 22$, $\xi_6 = 20$, $\xi_7 = 15$, and the values Ψ_{jr} are given in Table 4.

Step 3. Stemming from the assignment problem (8)–(11), the optimal job sequence is $\pi = (J_2, J_1, J_3, J_7, J_6, J_5, J_4)$.

Step 4. From (14), the optimal resource allocation is $u_2 = 10.91533$, $u_1 = 10.71178$, $u_3 = 11.53352$, $u_7 = 5.841858$, $u_6 = 9.501238$, $u_5 = 9.251873$, $u_4 = 5.013141$.

Table 2: Values Ψ_{jr} for Example 1.

	(j/r)	1	2	3	4	5	6	7
$\Psi_{jr=}$	1	262.0000	200.3414	151.0178	119.7068	**98.60000**	98.60000	72.00000
	2	194.0000	160.6827	**131.3178**	112.1496	98.16679	87.28498	66.85148
	3	361.0000	332.5343	290.8883	264.5431	250.8000	250.8000	**171.0000**
	4	**142.0000**	127.9972	110.1639	98.56349	90.12287	83.56748	56.21822
	5	202.0000	162.1420	127.1396	104.6077	88.33854	**75.77993**	54.00000
	6	212.0000	169.9128	137.1665	**116.2334**	101.1959	95.20000	84.00000
	7	171.0000	**151.9446**	133.9793	122.4548	114.1549	108.8000	81.00000

Bold numbers are the optimal solution.

Table 3: Data for Example 2.

J_j	J_1	J_2	J_3	J_4	J_5	J_6	J_7
$\widetilde{p}_j$	13	17	12	10	18	16	9
α_j	−0.32	−0.24	−0.33	−0.25	−0.28	−0.3	−0.29
v_j	3	4	2	9	6	5	8

Table 4: Values Ψ_{jr} for Example 2.

	(j/r)	1	2	3	4	5	6	7
$\Psi_{jr=}$	1	45.20903	**48.20302**	48.17622	45.30844	43.32309	41.96834	38.13077
	2	**65.49197**	72.45895	74.00176	70.67272	68.19401	64.16226	56.87507
	3	32.70817	34.71350	**34.60055**	32.47854	31.34372	30.36358	27.58714
	4	78.94848	86.94426	88.55579	84.40999	81.32839	76.42716	**67.67740**
	5	89.15204	96.82953	97.82774	92.71289	88.93040	**83.26686**	75.19374
	6	72.98642	78.54253	78.92439	74.51153	**71.25929**	67.75459	61.55913
	7	68.03574	73.55408	74.11175	**70.10229**	67.14231	63.15878	57.38356

Bold numbers are the optimal solution.

Step 5. Calculate $d_1 = C_{[2]} = C_1 = 3.370785$, $d_2 = C_{[5]} = C_6 = 6.0368777$, and $\sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D + \sum_{j=1}^{n} v_{[j]} u_{[j]} = 440.6014$.

4.2. Problem $1|CONW, P_j^A = ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^\eta, \sum_{j=1}^{n} v_{[j]} u_{[j]} \le \overline{U} | \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$. In this section, we aim to minimize the following cost function $\sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$ subject to $\sum_{j=1}^{n} v_{[j]} u_{[j]} \le \overline{U}$, $1|\mathrm{CONW}, P_j^A = ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^\eta, \sum_{j=1}^{n} v_{[j]} u_{[j]} \le \overline{U} | \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$ where $\overline{U}$ is a limitation on the total resource consumption cost. Obviously, in an optimal solution for the problem $1|\mathrm{CONW}, P_j^A = ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^\eta, \sum_{j=1}^{n} v_{[j]} u_{[j]} \le \overline{U} | \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$ the constraint will be satisfied as $\sum_{j=1}^{n} v_{[j]} u_{[j]} = \overline{U}$.

Lemma 4. *For a given sequence, the optimal resource allocation of the problem* $1|CONW, P_j^A = ((\widetilde{p}_j \max\{r^{\alpha_j}, \delta\})/u_j)^\eta, \sum_{j=1}^{n} v_{[j]} u_{[j]} \le \overline{U} | \sum_{j=1}^{n} w_j L_{[j]} + w_0 d_1 + w_{n+1} D$ *is*

$$u_{[j]}^* = \frac{(\xi_j)^{(1/\eta+1)} (v_{[j]})^{(-1/\eta+1)} (\widetilde{p}_{[j]} \max\{j^{\alpha_{[j]}}, \delta\})^{(\eta/\eta+1)}}{\sum_{j=1}^{n} (\xi_j)^{(1/\eta+1)} (\widetilde{p}_{[j]} \max\{j^{\alpha_{[j]}}, \delta\})^{(\eta/\eta+1)}} \times \overline{U}, \quad j = 1, 2, \ldots, n, \tag{19}$$

where ξ_j $(j = 1, 2, \ldots, n)$ *are given by (6).*

Proof. For a given sequence $\pi = (J_{[1]}, J_{[2]}, \ldots, J_{[n]})$, the Lagrange function is

$$L(u, \lambda) = \sum_{j=1}^{n} \xi_j P_{[j]}^A + \lambda \left(\sum_{j=1}^{n} v_{[j]} u_{[j]} - \overline{U} \right) = \sum_{j=1}^{n} \xi_j \left(\frac{\widetilde{p}_{[j]} \max\{j^{\alpha_{[j]}}, \delta\}}{u_{[j]}} \right)^\eta + \lambda \left(\sum_{j=1}^{n} v_{[j]} u_{[j]} - \overline{U} \right), \tag{20}$$

where λ is the Lagrangian multiplier. Deriving (20) with respect to $u_{[j]}$ and λ, we have

$$\frac{\partial L(u, \lambda)}{\partial u_{[j]}} = \lambda v_{[j]} - \eta \xi_j \times \frac{(\widetilde{p}_j \max\{j^{\alpha_{[j]}}, \delta\})^\eta}{(u_{[j]})^{\eta+1}} = 0. \tag{21}$$

It follows that

$$u_{[j]}^* = \frac{(\eta \xi_j (\widetilde{p}_{[j]} \max\{j^{\alpha_{[j]}}, \delta\})^\eta)^{(1/\eta+1)}}{(\lambda v_{[j]})^{(1/\eta+1)}}. \tag{22}$$

From $\sum_{j=1}^{n} v_{[j]}u_{[j]} = \overline{U}$, we have

$$\lambda^{(1/\eta+1)} = \frac{\sum(\eta\xi_j)^{(1/\eta+1)}\left(\tilde{p}_{[j]}v_{[j]}\max\{j^{\alpha_{[j]}},\delta\}\right)^{(\eta/\eta+1)}}{\overline{U}}. \quad (23)$$

Finally, inserting (23) into (22), we have (19).

By substituting the values $u_{[j]}^*$ given in (19) into $\sum_{j=1}^{n}\xi_j P_{[j]}^A = \sum_{j=1}^{n}\xi_j((\tilde{p}_{[j]}\max\{j^{\alpha_{[j]}},\delta\}/u_{[j]}))^{\eta}$, we have

$$\sum_{j=1}^{n}\xi_j P_{[j]}^A = \overline{U}^{-\eta}\left(\sum_{j=1}^{n}\left(v_{[j]}\tilde{p}_{[j]}\max\{j^{\alpha_{[j]}},\delta\}\right)^{(\eta/\eta+1)}\left(\xi_j\right)^{(1/\eta+1)}\right)^{\eta+1}. \quad (24)$$

Similarly to Section 4.1, we have the following. □

Theorem 3. *Problem* $1|CONW, P_j^A = (((\tilde{p}_j\max\{r^{\alpha_j},\delta\})/u_j))^{\eta}, \sum_{j=1}^{n} v_{[j]}u_{[j]} \le \overline{U}|\sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D$ *can be solved in* $O(n^3)$ *time.*

4.3. Problem $1|CONW, P_j^A = ((\tilde{p}_j\max\{r^{\alpha_j},\delta\})/u_j)^{\eta}, \sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D \le \overline{V}|\sum_{j=1}^{n} v_{[j]}u_{[j]}$. In this section, the inverse version' of $1|CONW, P_j^A = ((\tilde{p}_j\max\{r^{\alpha_j},\delta\}/u_j))^{\eta}, \sum_{j=1}^{n} v_{[j]}u_{[j]} \le \overline{U}|\sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D$ will be considered, i.e., the problem of minimizing $\sum_{j=1}^{n} v_{[j]}u_{[j]}$ subject to$\sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D \le \overline{V}$, where $\overline{V}$ is a given real number.

Similarly to Section 4.2, we have.

Lemma 5. *For a given sequence, the optimal resource allocation of the problem*$1|CONW, P_j^A = ((\tilde{p}_j\max\{r^{\alpha_j},\delta\})/u_j)^{\eta}, \sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D \le \overline{V}|\sum_{j=1}^{n} v_{[j]}u_{[j]}$ *is*

$$u_{[j]}^* = \overline{V}^{-(1/\eta)}\left(\xi_j\right)^{(1/\eta+1)}\left(v_{[j]}\right)^{(-1/\eta+1)}\left(\tilde{p}_{[j]}\max\{j^{\alpha_{[j]}},\delta\}\right)^{(\eta/\eta+1)}\left(\sum_{j=1}^{n}\left(\xi_j\right)^{(1/\eta+1)}\left(\tilde{p}_{[j]}v_{[j]}\max\{j^{\alpha_{[j]}},\delta\}\right)^{(\eta/\eta+1)}\right)^{(1/\eta)}, \quad (25)$$

where ξ_j $(j = 1, 2, \ldots, n)$ *are given by (6).*

By substituting the values $u_{[j]}^*$ *given in (25) into* $\sum_{j=1}^{n} v_{[j]}u_{[j]}$, *we have*

$$\sum_{j=1}^{n} v_{[j]}u_{[j]} = \overline{V}^{-(1/\eta)}\left(\sum_{j=1}^{n}\left(v_{[j]}\tilde{p}_{[j]}\max\{j^{\alpha_{[j]}},\delta\}\right)^{(\eta/\eta+1)}\left(\xi_j\right)^{(1/\eta+1)}\right)^{(1/\eta)+1}. \quad (26)$$

Similarly to Section 4.2, we have.

Theorem 4. *Problem* $1|CONW, P_j^A = ((\tilde{p}_j\max\{r^{\alpha_j},\delta\})/u_j)^{\eta}, \sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D \le \overline{V}|\sum_{j=1}^{n} v_{[j]}u_{[j]}$ *can be solved in* $O(n^3)$ *time.*

Remark. Obviously, the CONW model can be extended to the slack due-window (SLKW) assignment model. The objective function $\sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D$ can be replaced by

$$\sum_{j=1}^{n} w_jL_{[j]} + w_0q' + w_{n+1}D, \quad (27)$$

where

$$L_{[j]} = \begin{cases} d'_{[j]} - C_{[j]}, & \text{for } d'_{[j]} > C_{[j]}, \\ 0, & \text{for } d_1 \le C_{[j]} \le d_2, \\ C_{[j]} - d''_{[j]}, & \text{for } C_{[j]} > d''_{[j]}, \end{cases} \quad (28)$$

and $D = d''_j - d'_j = q'' - q'$.

5. Conclusions

This paper considered the single-machine due-window assignment scheduling problems with learning effect and resource allocation. For the linear and convex resource allocations, we showed that some different models are polynomially solvable, respectively. Future research may focus on the flow shop scheduling problems with learning effect and resource allocation or study the Pareto-optimal solutions with respect to the criterion $\sum_{j=1}^{n} w_jL_{[j]} + w_0d_1 + w_{n+1}D$ and the resource compression cost $\sum_{j=1}^{n} v_{[j]}u_{[j]}$.

Data Availability

No data were used in the study.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

References

[1] D. Biskup, "A state-of-the-art review on scheduling with learning effects," *European Journal of Operational Research*, vol. 188, no. 2, pp. 315–329, 2008.

[2] Y.-Y. Lu, F. Teng, and Z.-X. Feng, "Scheduling jobs with truncated exponential sum-of-logarithm-processing-times based and position-based learning effects," *Asia-Pacific*

Journal of Operational Research, vol. 32, no. 3, pp. 1–17, Article ID 1550026, 2015.

[3] A. Azzouz, M. Ennigrou, and L. Ben Said, “Scheduling problems under learning effects: Classification and cartography,” *International Journal of Production Research*, vol. 56, no. 4, pp. 1642–1661, 2018.

[4] J.-B. Wang, F. Liu, and J.-J. Wang, “Research on m -machine flow shop scheduling with truncated learning effects,” *International Transactions in Operational Research*, vol. 26, no. 3, pp. 1135–1151, 2019.

[5] D. Shabtay and G. Steiner, “A survey of scheduling with controllable processing times,” *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1643–1666, 2007.

[6] 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.

[7] D. Wang, M.-Z. Wang, and J.-B. Wang, “Single-machine scheduling with learning effect and resource-dependent processing times,” *Computers & Industrial Engineering*, vol. 59, no. 3, pp. 458–462, 2010.

[8] Y.-Y. Lu, G. Li, Y.-B. Wu, and P. Ji, “Optimal due-date assignment problem with learning effect and resource-dependent processing times,” *Optimization Letters*, vol. 8, no. 1, pp. 113–127, 2014.

[9] 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, pp. 1–15, Article ID 1450036, 2014.

[10] G. Li, M.-L. Luo, W.-J. Zhang, and X.-Y. Wang, “Single-machine due-window assignment scheduling based on common flow allowance, learning effect and resource allocation,” *International Journal of Production Research*, vol. 53, no. 4, pp. 1228–1241, 2015.

[11] J.-B. Wang and J.-J. Wang, “Research on scheduling with job-dependent learning effect and convex resource-dependent processing times,” *International Journal of Production Research*, vol. 53, no. 19, pp. 5826–5836, 2015.

[12] J.-B. Wang, X.-N. Geng, L. Liu, J.-J. Wang, and Y.-Y. Lu, “Single machine CON/SLK due date assignment scheduling with controllable processing time and job-dependent learning effects,” *The Computer Journal*, vol. 61, no. 9, pp. 1329–1337, 2018.

[13] W. Liu and C. Jiang, “Due-date assignment scheduling involving job-dependent learning effects and convex resource allocation,” *Engineering Optimization*, vol. 52, no. 1, pp. 74–89, 2020.

[14] W.-W. Liu and C. Jiang, “Flow shop resource allocation scheduling with due date assignment, learning effect and position-dependent weights,” *Asia-Pacific Journal of Operational Research*, vol. 37, no. 3, pp. 1–27, Article ID 2050014, 2020.

[15] H.-B. Shi and J.-B. Wang, “Research on common due window assignment flowshop scheduling with learning effect and resource allocation,” *Engineering Optimization*, vol. 52, no. 4, pp. 669–686, 2020.

[16] 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.

[17] L.-Y. Wang, D.-Y. Lv, B. Zhang, W.-W. Liu, and J.-B. Wang, “Optimization for different due-window assignment scheduling with position-dependent weights,” *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–7, Article ID 9746538, 2020.

[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] 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.

[20] 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.

[21] 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.