systems

Article

Two Due-Date Assignment Scheduling with Location-Dependent Weights and a Deteriorating Maintenance Activity

Wei Wu, Dan-Yang Lv and Ji-Bo Wang *

School of Science, Shenyang Aerospace University, Shenyang 110136, China
* Correspondence: wangjibo@sau.edu.cn

Abstract: This paper investigates single-machine scheduling with a deteriorating maintenance activity, where the processing time of a job depends on whether it is handled before or after the maintenance activity. Under common and slack due date assignments, the aim is to find the optimal job schedule, position of the maintenance activity, and optimal value of the common due date (flow-allowance) so that the linear weighted sum of earliness, tardiness and common due date (flow-allowance) value is minimized, where the weights are location-dependent (position-dependent) weights. Through a series of optimal properties, a polynomial time algorithm is proposed and it is then proven that the problem is polynomially solvable.

Keywords: scheduling; maintenance activity; common due date; common flow-allowance; slack due date; location-dependent weights

check for updates

Citation: Wu, W.; Lv, D.-Y.; Wang, J.-B. Two Due-Date Assignment Scheduling with Location-Dependent Weights and a Deteriorating Maintenance Activity. *Systems* **2023**, *11*, 150. https://doi.org/10.3390/systems11030150

Academic Editors: Mahmoud Efatmaneshnik, Shraga Shoval and Larissa Statsenko

Received: 5 February 2023
Revised: 8 March 2023
Accepted: 9 March 2023
Published: 15 March 2023

1. Introduction

Traditional scheduling problems consider that the machine is always available, while actually, if a maintenance (rate-modifying) activity is carried out on the machine, it becomes unavailable, and it reverts to its original condition after the maintenance activity is finished (see Lee and Leon [1]; Ma et al. [2]; Strusevich and Rustogi [3]). In 2006, Mosheiov and Oron [4] considered single-machine scheduling with a rate-modifying activity (denoted rma). Under the common due-date (denoted CON) assignment, the goal is to minimize the total earliness, tardiness and due-date cost. They proved that the problem can be solved in polynomial time. In 2010, Wang and Wang [5] addressed single-machine scheduling with rma and slack due-date (denoted SLK) assignment. They showed that the non-regular objective minimization is polynomially solvable. In 2012, Yin et al. [6] studied single-machine batch delivery scheduling with the rma and CON assignment, with the goal of minimizing the sum of earliness, tardiness, due-date, holding and delivery cost. They proved that some special cases of the problem can be solved in polynomial time. In 2014, Bai et al. [7] studied the single-machine problem with deteriorating jobs and rma. Under the SLK allocation, the objective is to minimize the weighted sum of earliness, tardiness and the common flow allowance cost. They showed that the problem is polynomially solvable. In 2018, Cheng et al. [8] discussed single-machine batch problems with variable maintenance activities. For the minimization of makespan (total completion time), they demonstrated that the problem can be solved in polynomial time in a special case. In 2019, Detti et al. [9] addressed the single-machine problem with a flexible maintenance activity. In 2021, Wang et al. [10] considered the single-machine problem with rma and a common due-window. For the generalized earliness/tardiness penalties, they showed that the problem is polynomially solvable. In 2022, Zhao et al. [11] studied the single-machine resource allocation problem with an aging effect as well as rma and SLK assignment. In 2023, Sun et al. [12] delved into single-machine scheduling with worsening effects. With the maintenance activity, they showed that some regular objective minimizations are polynomially solvable. For total weighted completion time minimization, they suggested several heuristic and branch-and-bound algorithms. In addition, in 2014, Ji et al. [13] and Fan and Zhao [14] considered due-date assignment scheduling with a deteriorating maintenance activity (denoted *dma*), i.e., the maintenance duration (time) is a linear deteriorating function of the starting time of this maintenance activity. In 2015, Mor and Mosheiov [15] studied single-machine scheduling with *dma*. Under the due-window assignment, they proved that some non-regular minimizations can be solved in polynomial time. In 2017, Li and Chen [16] investigated the single-machine problems with job-rejection and *dma*, and they showed that the makespan, total completion time and due-date assignment minimizations are polynomially solvable. Zhu et al. [17] explored single-machine scheduling with a general (including worsening and resource-dependent) maintenance activity. They showed that the problem can be solved in polynomial time for a number of regular and non-regular objectives. Wang et al. [18] addressed identical parallel machine scheduling with *dma*. They illustrated that the total completion time minimization can be solved in polynomial time. More recent papers which have studied scheduling problems with *rma* and *dma* include He et al. [19], Jia et al. [20], liu et al. [21] and Zou et al. [22].

At the same time, in modern manufacturing, more and more industries are adopting due-date assignment systems (see Gordon et al. [23,24], Qian and Zhan [25], Lu et al. [26], Wang [27]) and location-dependent (position-dependent) weights (Brucker [28], Liu et al. [29], Wang et al. [30], Wang et al. [31]). Recently, Jiang et al. [32] investigated proportionate flow-shop problems with location-dependent weights. Under the *CON* and *SLK* assignments, they showed that the non-regular objective minimization is polynomially solvable (i.e., time is $O(n^2 \log n)$, where n is the number of jobs). Liu et al. [33] studied single-machine scheduling with resource allocation, worsening jobs and location-dependent weights. Under the *CON* and *SLK* assignments, they provided a bi-criteria analysis for the scheduling cost (including the weighted sum of the absolute value in lateness and common due date (flow-allowance) cost) and the resource consumption cost. They proved that three versions of these two costs can be solved in polynomial time. In 2021, Lv and Wang [34] considered the same problems as Jiang et al. [32], and they demonstrated that both problems can be addressed by using a lower-order algorithm (i.e., time is $O(n \log n)$). Wang et al. [35] and Wang et al. [36] investigated single-machine scheduling with setup times associated with past sequences and weights related to location. Under the *CON* and *SLK* assignments, Wang et al. [35] demonstrated that the weighted sum of earliness, tardiness and due-date cost minimization is polynomially solvable. Wang et al. [36] demonstrated that the weighted sum of lateness, number of early and delayed jobs, and due-date cost minimization is also polynomially solvable.

In real production processes, the location-dependent (position-dependent) weights can be found in services and logistics systems (e.g., in Didi taxi dispatching, Sun et al. [37]). Hence, the work of *CON* (*SLK*) assignment and *dma* will be continued by considering the location-dependent weights, i.e., we investigate single-machine due-date assignment scheduling with *dma* and location-dependent weights. Under the *CON* and *SLK* assignments, the aim of the article is to find the job schedule, location of the maintenance activity, and optimal value of common due-date (flow-allowance) so as to minimize the linear weighted sum of the earliness, tardiness and due date assignment costs, where the weights are related to the location. We demonstrate that the problem can be solved in polynomial time, i.e., time complexity is $O(n^4)$. For a comparison with other similar papers (see Table 1; the related symbols are given later), this article extends the results of Mosheiova and Oron [4], Wang and Wang [5], Brucker [28], and Liu et al. [29], by scrutinizing a more general scheduling model. The rest of this article is structured as below. In Section 2, we introduce the problem. In Sections 3 and 4, we explore the main details of common due-date and slack due-date assignments. In Section 5, an example and computational experiments are reported to verify the effectiveness of the algorithms. Section 6 presents the conclusion of the problem.

Table 1. Summary of *CON* and *SLK* results.

Problem	Complexity	Ref.
$1\|CON, b_i = p_i\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \mu_i \widetilde{L_{[i]}} + \omega_0 d_{opt})$	$O(n \log n)$	Brucker [28]
$1\|SLK, b_i = p_i\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \mu_i \widetilde{L_{[i]}} + \omega_0 q_{opt})$	$O(n \log n)$	Liu et al. [29]
$1\Big\|S_{psd}, CON, b_i = p_i\Big\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \mu_i \widetilde{L_{[i]}} + \omega_0 d_{opt})$	$O(n \log n)$	Wang et al. [36]
$1\Big\|S_{psd}, SLK, b_i = p_i\Big\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \mu_i \widetilde{L_{[i]}} + \omega_0 q_{opt})$	$O(n \log n)$	Wang et al. [36]
$1\|rma, CON, b_i = (p_i, \varepsilon_i p_i)\| \sum_{i=1}^{n} (\mu_0 \widetilde{E_{[i]}} + \nu_0 \widetilde{L_{[i]}} + \omega_0 d_{opt})$	$O(n^4)$	Mosheiova and Oron [4]
$1\|rma, SLK, b_i = (p_i, \varepsilon_i p_i)\| \sum_{i=1}^{n} (\mu_0 \widetilde{E_{[i]}} + \nu_0 \widetilde{L_{[i]}} + \omega_0 q_{opt})$	$O(n^4)$	Wang and Wang [5]
$1\|dma, CON, b_i = (p_i, \varepsilon_i p_i)\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i d_{opt})$	$O(n^4)$	Theorem 1
$1\|dma, SLK, b_i = (p_i, \varepsilon_i p_i)\| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$	$O(n^4)$	Theorem 2

2. Problem Description

In this article, the problem can be described as: n independent jobs $\{\check{T}_1, \check{T}_2, \ldots, \check{T}_n\}$ can be usable at time 0 for handling on a single-machine and they are not preemptive. The machine stops working when a maintenance activity is being carried out, if the job $\check{T}_i$ is handled before the maintenance activity, the corresponding normal processing time (i.e., basic processing time that is not affected by other factors) is defined as p_i, and its actual processing time is $b_i = p_i$. If the job $\check{T}_i$ is processed after the maintenance activity, the corresponding actual processing time is defined as $b_i = \varepsilon_i p_i$, where ε_i is the modified rate of the job $\check{T}_i$, and it satisfies $0 < \varepsilon_i \leq 1$. Meanwhile, the machine only has a deteriorating maintenance activity, and its maintenance time is: $\varphi_{(t)} = t_0 + \alpha S_t$, where t_0 is the basic maintenance time, $\alpha > 0$ is the rate of deterioration, and S_t is the starting time of dma. We consider two different types of due-dates that include CON and SLK. As for the CON model, all the jobs have the same due date, i.e., $d_i = d_{opt}$, in which d_{opt} is a decision variable. For the SLK model, the due date d_i of job $\check{T}_i$ is equivalent to the sum of the actual processing time b_i and the common flow-allowance q_{opt}, i.e., $d_i = b_i + q_{opt}$, and the common flow-allowance q_{opt} is a decision variable. For a known schedule σ, the completion (resp. starting) time of the job $\check{T}_i$ is C_i (resp. S_i), the earliness (resp. tardiness) cost of the job $\check{T}_i$ is $\widetilde{E_i} = \max\{0, d_i - C_i\}$ (resp. $\widetilde{L_i} = \max\{0, C_i - d_i\}$). Let $[i]$ be the job being arranged at the ith position in the schedule, then our goal is: (1) to determine the optimal job schedule; (2) to determine where to carry out the maintenance activity of the machine; (3) to determine the value of the common due date and the common flow-allowance; (4) to minimize the objective function: $\dot{Z} = \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i d_{opt})$ and $\ddot{Z} = \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$, where μ_i, ν_i, ω_i are weights of the ith location in a schedule, i.e., the weights are the location-dependent weights rather than the weights connected with the job $\check{T}_i$. With three-field notation, the scheduling can be expressed as:

$$1|dma, CON, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i d_{opt}); \tag{1}$$

$$1|dma, SLK, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt}). \tag{2}$$

3. Results of *CON*

Assuming that the maintenance activity occurs exactly before the job $\check{T}_{[i]}$, its starting time is equivalent to the completion time of the job $\check{T}_{[i-1]}$, i.e., $S_t = C_{[i-1]}$. For the sake of simplicity, let j be the location of the maintenance activity.

Lemma 1. *If $C_{[i]} \geq d_{opt}$, then $C_{[i+1]} \geq d_{opt}$, if $C_{[i]} \leq d_{opt}$, then $C_{[i-1]} \leq d_{opt}$.*

Proof. For a given schedule σ, if $C_{[i]} \geq d_{opt}$, we can get $C_{[i]} \geq d_{opt} \Leftrightarrow C_{[i+1]} - b_{[i+1]} \geq d_{opt} \Rightarrow C_{[i+1]} \geq d_{opt}$. Similarly, if $C_{[i]} \leq d_{opt}$, we can also get $C_{[i]} \leq d_{opt} \Leftrightarrow C_{[i-1]} + b_{[i]} \leq d_{opt} \Rightarrow C_{[i-1]} \leq d_{opt}$. □

From Lemma 1, it can be seen that the jobs before the job $\check{T}_{[i]}$ are the early jobs, and the jobs after the job $\check{T}_{[i]}$ are the late jobs.

Lemma 2. *For a known schedule $\sigma = (\check{T}_{[1]}, \check{T}_{[2]}, \ldots, \check{T}_{[n]})$, the optimal value of the common due date d_{opt} is equal to the completion time of the hth job, i.e., $d_{opt} = C_{[h]}$, where h satisfies both*

$$\left(\sum_{i=1}^{h} \mu_i - \sum_{i=h+1}^{n} \nu_i + \sum_{i=1}^{n} \omega_i\right) \geq 0 \text{ and } \left(\sum_{i=1}^{h-1} \mu_i - \sum_{i=h}^{n} \nu_i + \sum_{i=1}^{n} \omega_i\right) \leq 0.$$

Proof. See Appendix A. □

Remark 1. *For a given schedule, if h satisfies both the above inequalities, the optimal common due date can be determined by Lemma 2. But h may not meet the above both inequalities, so we need to set $d_{opt} = 0$.*

Lemma 3 (Hardy et al. [38]). *For $\sum_{i=1}^{n} \xi_i \eta_i$, if the sequence $\{\xi_1, \ldots, \xi_n\}$ is in non-decreasing order and the sequence $\{\eta_1, \ldots, \eta_n\}$ is in non-increasing order, or vice versa, it is the smallest.*

Next, we consider the problem $1|dma, CON, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i d_{opt})$. Under Lemmas 1–3, the following cases should be considered.

Case 1. If $j \leq h$, the schedule is shown in Figure 1.

$\check{T}_{[1]}$	$\check{T}_{[2]}$	$\ldots$	$\check{T}_{[j-1]}$	dma	$\ldots$	$\check{T}_{[h]}$	$\ldots$	$\check{T}_{[n]}$
$p_{[1]}$	$p_{[2]}$	$\ldots$	$p_{[j-1]}$	$\varphi(t)$	$\ldots$	$\varepsilon_{[h]}\, p_{[h]}$	$\ldots$	$\varepsilon_{[n]}\, p_{[n]}$

Figure 1. If $j \leq h$.

Where $d_{opt} = C_{[h]} = \sum_{i=1}^{j-1} p_{[i]} + \varphi(t) + \sum_{i=j}^{h} \varepsilon_{[i]}\, p_{[i]}$ and $\varphi(t) = t_0 + \alpha \sum_{i=1}^{j-1} p_{[i]}$. Then the objective function is expressed as

$$
\begin{aligned}
&\dot{Z}([j]) = \sum_{i=1}^{n} (\mu_i \, \widetilde{E_{[i]}} + \nu_i \, \widetilde{L_{[i]}} + \omega_i \, d_{opt}) \\
=\; & \sum_{i=1}^{h} \mu_i \Big(C_{[h]} - C_{[i]} \Big) + \sum_{i=h+1}^{n} v_i \Big(C_{[i]} - C_{[h]} \Big) + \sum_{m=1}^{n} \omega_m \cdot C_{[h]} \\
=\; & t_0 \sum_{m=1}^{j-1} \mu_m + \alpha \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{j-1} \mu_m \right) + \sum_{i=j}^{h} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{j-1} \mu_m \right) + \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m \right) \\
& + \sum_{i=j}^{h} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=j}^{i-1} \mu_m \right) + \sum_{i=h+1}^{n} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=i}^{m} \nu_m \right) + \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) \\
& + t_0 \sum_{m=1}^{n} \omega_m + \alpha \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) + \sum_{i=j}^{h} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) \\
=\; & \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=1}^{j-1} \mu_m + \sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m + \alpha \sum_{m=1}^{n} \omega_m \right) + \sum_{i=j}^{h} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m \right) \\
& + \sum_{i=h+1}^{n} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=i}^{n} v_m \right) + t_0 \left(\sum_{m=1}^{j-1} \mu_m + \sum_{m=1}^{n} \omega_m \right) \\
=\; & \dot{Z}_1([j]) + f_1(t_0)
\end{aligned} \tag{3}
$$

where $f_1(t_0) = t_0 \left(\sum\limits_{m=1}^{j-1} \mu_m + \sum\limits_{m=1}^{n} \omega_m \right)$ is a constant (if j is given),

$$
\begin{aligned}
\dot{Z}_1([j]) = & \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=1}^{j-1} \mu_m + \sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m + \alpha \sum_{m=1}^{n} \omega_m \right) \\
& + \sum_{i=j}^{h} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m \right) + \sum_{i=h+1}^{n} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=i}^{n} v_m \right)
\end{aligned} \tag{4}
$$

is only related to the location j of the maintenance activity. Therefore, for a known j, minimizing $\dot{Z}([j])$ is equivalent to minimizing $\dot{Z}_1([j])$. Let

$$
\lambda_{il} = \begin{cases}
p_i \left(\alpha \sum\limits_{m=1}^{j-1} \mu_m + \sum\limits_{m=1}^{l-1} \mu_m + \sum\limits_{m=1}^{n} \omega_m + \alpha \sum\limits_{m=1}^{n} \omega_m \right) & 1 \le l \le j-1 \\
\varepsilon_i \, p_i \left(\sum\limits_{m=1}^{l-1} \mu_m + \sum\limits_{m=1}^{n} \omega_m \right) & j \le l \le h \\
\varepsilon_i \, p_i \sum\limits_{m=l}^{n} v_m & h+1 \le l \le n
\end{cases} \tag{5}
$$

$\dot{Z}_1([j])$ can be minimized by solving the next assignment problem:

$$
\begin{aligned}
& Min \quad \dot{Z}_1([j]) = \sum_{i=1}^{n} \sum_{l=1}^{n} \lambda_{il} \, \beta_{il} \\
& s.t \quad \sum_{i=1}^{n} \beta_{il} = 1 \qquad l = 1, \ldots, n \\
& \qquad \sum_{l=1}^{n} \beta_{il} = 1 \qquad i = 1, \ldots, n \\
& \qquad \beta_{il} = 0 \; or \; 1 \qquad 1 \le i, l \le n
\end{aligned} \tag{6}
$$

where β_{il} is a 0 or 1 variable, if the job $\check{T}_i$ is at location l, $\beta_{il} = 1$, otherwise $\beta_{il} = 0$.

Case 2. If $j > h$, the schedule is shown Figure 2.

$\check{T}_{[1]}$	$\check{T}_{[2]}$	$\ldots$	$\check{T}_{[h]}$	$\ldots$	$\check{T}_{[j-1]}$	dma	$\ldots$	$\check{T}_{[n]}$
$p_{[1]}$	$p_{[2]}$	$\ldots$	$p_{[h]}$	$\ldots$	$p_{[j-1]}$	$\varphi(t)$	$\ldots$	$\varepsilon_{[n]}\, p_{[n]}$

Figure 2. If $j > h$.

Where $d_{opt} = C_{[h]} = \sum_{i=1}^{h} p_{[i]}$, and $\varphi(t) = t_0 + \alpha \sum_{i=1}^{j-1} p_{[i]}$. Then the objective function is expressed as

$$
\begin{aligned}
\dot{Z}([j]) &= \sum_{i=1}^{n}(\mu_i\, \widetilde{E_{[i]}} + v_i\, \widetilde{L_{[i]}} + \omega_i\, d_{opt}) \\
&= \sum_{i=1}^{h} \mu_i \Big(C_{[h]} - C_{[i]}\Big) + \sum_{i=h+1}^{j-1} v_i \Big(C_{[i]} - C_{[h]}\Big) + \sum_{i=j}^{n} v_i \Big(C_{[i]} - C_{[h]}\Big) + \sum_{i=1}^{n} \omega_i \left(\sum_{i=1}^{h} p_{[i]}\right) \\
&= \sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m\right) + \sum_{i=h+1}^{j-1} p_{[i]} \left(\sum_{m=i}^{j-1} v_m\right) + \sum_{i=h+1}^{j-1} p_{[i]} \left(\sum_{m=j}^{n} v_m\right) \\
&\quad + t_0 \sum_{m=j}^{n} v_m + \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=j}^{n} v_m\right) + \sum_{i=j}^{n} \varepsilon_{[i]}\, p_{[i]} \left(\sum_{m=i}^{n} v_m\right) + \sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{n} \omega_m\right) \\
&= \sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m + \alpha \sum_{m=j}^{n} v_m + \sum_{m=1}^{n} \omega_m\right) + \sum_{i=h+1}^{j-1} p_{[i]} \left(\sum_{m=i}^{n} v_m + \alpha \sum_{m=j}^{n} v_m\right) \\
&\quad + \sum_{i=j}^{n} \varepsilon_{[i]}\, p_{[i]} \left(\sum_{m=i}^{n} v_m\right) + t_0 \sum_{m=j}^{n} v_m \\
&= \dot{Z}_2([j]) + f_2(t_0)
\end{aligned} \tag{7}
$$

where $f_2(t_0) = t_0 \sum_{m=j}^{n} v_m$ is only related to t_0 and it is a constant,

$$
\begin{aligned}
\dot{Z}_2([j]) = &\sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m + \alpha \sum_{m=j}^{n} v_m + \sum_{m=1}^{n} \omega_m\right) \\
&\sum_{i=h+1}^{j-1} p_{[i]} \left(\sum_{m=i}^{n} v_m + \alpha \sum_{m=j}^{n} v_m\right) + \sum_{i=j}^{n} \varepsilon_{[i]}\, p_{[i]} \left(\sum_{m=i}^{n} v_m\right)
\end{aligned} \tag{8}
$$

is only related to the location j of the maintenance activity. Similarly to Case 1 (i.e., $j \leq h$), let

$$
\delta_{il} = \begin{cases}
p_i \left(\sum_{m=1}^{l-1} \mu_m + \alpha \sum_{m=j}^{n} v_m + \sum_{m=1}^{n} \omega_m\right) & 1 \leq l \leq h \\
p_i \left(\sum_{m=l}^{n} v_m + \alpha \sum_{m=j}^{n} v_m\right) & h+1 \leq l \leq j-1 \\
\varepsilon_i\, p_i \sum_{m=l}^{n} v_m & j \leq l \leq n
\end{cases} \tag{9}
$$

then, the problem can be translated into the assignment problem below:

$$
Min \quad \dot{Z}_2([j]) = \sum_{i=1}^{n} \sum_{l=1}^{n} \delta_{il}\, \gamma_{il}
$$

$$
s.t \qquad \sum_{i=1}^{n} \gamma_{il} = 1 \qquad l = 1, \ldots, n
$$

$$\sum_{l=1}^{n} \gamma_{il} = 1 \qquad i = 1, \ldots, n \tag{10}$$
$$\gamma_{il} = 0 \; or \; 1 \qquad 1 \le i, l \le n$$

where γ_{il} is a 0 or 1 variable, when the job $\check{T}_i$ is at position l, $\gamma_{il} = 1$, otherwise $\gamma_{il} = 0$.

Case 3. If $j = n + 1$, the schedule is shown Figure 3.

$\check{T}_{[1]}$	$\check{T}_{[2]}$	$\ldots$	$\check{T}_{[h]}$	$\ldots$	$\check{T}_{[n]}$	dma
$p_{[1]}$	$p_{[2]}$	$\ldots$	$p_{[h]}$	$\ldots$	$p_{[n]}$	$\varphi(t)$

Figure 3. If $j = n + 1$.

where $d_{opt} = C_{[h]} = \sum_{i=1}^{h} p_{[i]}$ and $\varphi(t) = t_0 + \alpha \sum_{i=1}^{n} p_{[i]}$. This means that there is no maintenance activity on the machine. In this case, the objective function is

$$\begin{aligned}
&\dot{Z}([n+1]) = \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + v_i \widetilde{L_{[i]}} + \omega_i d_{opt}) \\
= &\sum_{i=1}^{h} \mu_i \left(C_{[h]} - C_{[i]} \right) + \sum_{i=h+1}^{n} v_i \left(C_{[i]} - C_{[h]} \right) + \sum_{i=1}^{n} \omega_i \left(\sum_{i=1}^{h} p_{[i]} \right) \\
= &\sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m \right) + \sum_{i=h+1}^{n} p_{[i]} \left(\sum_{m=i}^{n} v_m \right) + \sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) \\
= &\sum_{i=1}^{h} p_{[i]} \left(\sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m \right) + \sum_{i=h+1}^{n} p_{[i]} \left(\sum_{m=i}^{n} v_m \right) \\
= &\sum_{i=1}^{n} p_{[i]} \, \pi_i
\end{aligned} \tag{11}$$

where

$$\pi_i = \begin{cases} \sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m & 1 \le i \le h \\ \sum_{m=i}^{n} v_m & h+1 \le i \le n \end{cases} \tag{12}$$

The optimal schedule σ as well as the minimum value of $\dot{Z}$ can be acquired by matching the minimum weight π_i with the maximum processing time p_i by using HLP rule (see Lemma 3).

Case 4. if $j = 1$, the schedule is shown Figure 4.

dma	$\check{T}_{[1]}$	$\check{T}_{[2]}$	$\ldots$	$\check{T}_{[h]}$	$\ldots$	$\check{T}_{[n]}$
$\varphi(t)$	$\varepsilon_{[1]} \, p_{[1]}$	$\varepsilon_{[2]} \, p_{[2]}$	$\ldots$	$\varepsilon_{[h]} \, p_{[h]}$	$\ldots$	$\varepsilon_{[n]} \, p_{[n]}$

Figure 4. If $j = 1$.

where $d_{opt} = C_{[h]} = \sum_{i=1}^{h} \varepsilon_{[i]} \, p_{[i]} + \varphi(t)$, and $\varphi(t) = t_0$. This means that the maintenance activity occurs before the jobs are processed. In this case, the objective function is

$$
\begin{aligned}
\dot{Z}([1]) &= \sum_{i=1}^{n}(\mu_i\,\widetilde{E_{[i]}}+v_i\,\widetilde{L_{[i]}}+\omega_i\,d_{opt}) \\
&= \sum_{i=1}^{h}\mu_i\Big(C_{[h]}-C_{[i]}\Big)+\sum_{i=h+1}^{n}v_i\Big(C_{[i]}-C_{[h]}\Big)+\sum_{i=1}^{h}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=1}^{n}\omega_m\right)+t_0\sum_{m=1}^{n}\omega_m \\
&= \sum_{i=1}^{h}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=1}^{i-1}\mu_m\right)+\sum_{i=h+1}^{n}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=i}^{n}v_m\right)+\sum_{i=1}^{h}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=1}^{n}\omega_m\right)+t_0\sum_{m=1}^{n}\omega_m \\
&= \sum_{i=1}^{h}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=1}^{i-1}\mu_m+\sum_{m=1}^{n}\omega_m\right)+\sum_{i=h+1}^{n}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=i}^{n}v_m\right)+t_0\sum_{m=1}^{n}\omega_m \\
&= \sum_{i=1}^{n}\varepsilon_{[i]}\,p_{[i]}\,\pi_i+f_3(t_0) \qquad (13)
\end{aligned}
$$

where $f_3(t_0) = t_0 \sum_{m=1}^{n} \omega_m$ is only related to t_0 and it is a constant. Similarly, the optimal schedule of this case can be obtained through Lemma 3.

From the above cases, the optimal solution of the problem $1|dma, CON, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i\widetilde{E_{[i]}} + \nu_i\widetilde{L_{[i]}} + \omega_i d_{opt})$ can be obtained by using the following algorithm.

Theorem 1. *The optimal solution of* $1|dma, CON, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i\widetilde{E_{[i]}} + \nu_i\widetilde{L_{[i]}} + \omega_i d_{opt})$ *can be obtained in* $O\big(n^4\big)$ *time by using Algorithm 1.*

Proof. The correctness of Algorithm 1 follows from the above analysis. For each j, the complexity of the assignment problem is $O\big(n^3\big)$, and it takes $n-1$ times. So the time complexity of Algorithm 1 is $O\big(n^4\big)$. □

Algorithm 1: Solution of $1|dma, CON, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i\widetilde{E_{[i]}} + \nu_i\widetilde{L_{[i]}} + \omega_i d_{opt})$

Initialization: Let $\dot{Z} = \infty$, $\dot{\sigma}^* = 0$, $d^*_{opt} = 0$ and $j^* = 0$.
Step 1: Calculate h from Lemma 2.
Step 2: For $j = 1 \to n+1$
  If $j \le h$, then
    obtain the minimum value $\dot{Z}([j])$ and the schedule $\dot{\sigma}$ by using (3)–(6);
      If $\dot{Z}([j]) < \dot{Z}$, then
      let $\dot{Z} = \dot{Z}([j])$, $j^* = j$, $d^*_{opt} = d_{opt}$ and $\dot{\sigma}^* = \dot{\sigma}$;
  If $j > h$, then
    obtain the minimum value $\dot{Z}([j])$ and the schedule $\dot{\sigma}$ by using (7)–(10);
      If $\dot{Z}([j]) < \dot{Z}$, then
      let $\dot{Z} = \dot{Z}([j])$, $j^* = j$, $d^*_{opt} = d_{opt}$ and $\dot{\sigma}^* = \dot{\sigma}$;
  If $j = n+1$, then
    acquire the optimal value of $\dot{Z} = \dot{Z}([n+1])$ and the schedule $\dot{\sigma}^*$ by the *HLP* rule;
    Calculate d^*_{opt};
  If $j = 1$, then
    obtain the optimal value of $\dot{Z} = \dot{Z}([1])$ and the schedule $\dot{\sigma}^*$ by the *HLP* rule;
    Calculate d^*_{opt}.
Step 3: Choose the minimum value $\dot{Z}^* = \min\{\dot{Z}[j],\ j = 1, 2, \ldots, n+1\}$, and obtain the corresponding schedule $\dot{\sigma}^*$, d^*_{opt} and j^*.

---

4. Results of *SLK*

Lemma 4. *If $C_{[i]} \geq d_{[i]}$, then $C_{[i+1]} \geq d_{[i+1]}$, if $C_{[i]} \leq d_{[i]}$, then $C_{[i-1]} \leq d_{[i-1]}$.*

Proof. For a given schedule σ, under the SLK model, we have $d_{[i]} = b_{[i]} + q_{opt}$. If $C_{[i]} \geq d_{[i]}$, we can get $C_{[i]} \geq d_{[i]} \Leftrightarrow C_{[i-1]} + b_{[i]} \geq b_{[i]} + q_{opt} \Rightarrow C_{[i-1]} + b_{[i]} \geq q_{opt} \Leftrightarrow C_{[i-1]} + b_{[i]} + b_{[i+1]} \geq q_{opt} + b_{[i+1]} \Leftrightarrow C_{[i]} + b_{[i+1]} \geq d_{[i+1]}$, i.e., $C_{[i+1]} \geq d_{[i+1]}$. Similarly, if $C_{[i]} \leq d_{[i]}$, we can also get $C_{[i]} \leq d_{[i]} \Leftrightarrow C_{[i-1]} + b_{[i]} \leq b_{[i]} + q_{opt} \Leftrightarrow C_{[i-1]} \leq q_{opt} \Rightarrow C_{[i-1]} \leq q_{opt} + b_{[i-1]}$, i.e., $C_{[i-1]} \leq d_{[i-1]}$. □

Lemma 5. *For a given schedule $\sigma = (\check{T}_{[1]}, \check{T}_{[2]}, \ldots, \check{T}_{[n]})$, the optimal value of the common flow-allowance q_{opt} is decided by the start time of the hth job, i.e., $q_{opt} = S_{[h]}$, where h satisfies both*

$$\left(\sum_{i=1}^{h} \mu_i - \sum_{i=h+1}^{n} \nu_i + \sum_{i=1}^{n} \omega_i\right) \geq 0 \text{ and } \left(\sum_{i=1}^{h-1} \mu_i - \sum_{i=h}^{n} \nu_i + \sum_{i=1}^{n} \omega_i\right) \leq 0.$$

Proof. Similar to the proof of Lemma 2. □

Remark 2. *Similarly, if h does not meet the above both inequalities, we need to set $q_{opt} = 0$.*

Next, we investigate the problem $1|dma, SLK, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$. Under Lemmas 3–5, the following cases will be considered.

Case 5. If $j \leq h$, the schedule is identical to Case 1, where $q_{opt} = S_{[h]} = \sum_{i=1}^{j-1} p_{[i]} + \varphi(t) + \sum_{i=j}^{h-1} \varepsilon_{[i]} p_{[i]}$ and $\varphi(t) = t_0 + \alpha \sum_{i=1}^{j-1} p_{[i]}$. Then the objective function is expressed as

$$\begin{aligned}
&\ddot{Z}([j]) = \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt}) \\
= \;& \sum_{i=1}^{h} \mu_i \Big(d_{[i]} - C_{[i]}\Big) + \sum_{i=h+1}^{n} \upsilon_i \Big(C_{[i]} - d_{[i]}\Big) + \sum_{i=1}^{n} \omega_i q_{opt} \\
= \;& t_0 \sum_{m=1}^{j-1} \mu_m + \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=1}^{j-1} \mu_m\right) + \sum_{i=j}^{h-1} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=1}^{j-1} \mu_m\right) + \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{i} \mu_m\right) \\
& + \sum_{i=j}^{h-1} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=j}^{i} \mu_m\right) + \sum_{i=h}^{n} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=i+1}^{n} \nu_m\right) + \sum_{i=1}^{j-1} p_{[i]} \left(\sum_{m=1}^{n} \omega_m\right) \\
& + t_0 \sum_{m=1}^{n} \omega_m + \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=1}^{n} \omega_m\right) + \sum_{i=j}^{h-1} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=1}^{n} \omega_m\right) \\
= \;& \sum_{i=1}^{j-1} p_{[i]} \left(\alpha \sum_{m=1}^{j-1} \mu_m + \sum_{m=1}^{i-1} \mu_m + \sum_{m=1}^{n} \omega_m + \alpha \sum_{m=1}^{n} \omega_m\right) + \sum_{i=j}^{h-1} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=1}^{i} \mu_m + \sum_{m=1}^{n} \omega_m\right) \\
& + \sum_{i=h}^{n} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=i+1}^{n} \upsilon_m\right) + t_0 \left(\sum_{m=1}^{j-1} \mu_m + \sum_{m=1}^{n} \omega_m\right) \\
= \;& \ddot{Z}_3([j]) + f_1(t_0)
\end{aligned} \tag{14}$$

where $f_1(t_0)$ is the same as it in Case 1, and

$$\ddot{Z}_3([j]) = \sum_{i=1}^{j-1} p_{[i]}\left(\sum_{m=1}^{i-1}\mu_m + \alpha\sum_{m=1}^{j-1}\mu_m + \sum_{m=1}^{n}\omega_m + \alpha\sum_{m=1}^{n}\omega_m\right) + \sum_{i=j}^{h-1}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=1}^{i}\omega_m + \sum_{m=1}^{n}\omega_m\right) + \sum_{i=h}^{n}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=i+1}^{n}\nu_m\right) \tag{15}$$

is only related to the position of j. Therefore, for a known j, minimizing $\ddot{Z}([j])$ is identical to minimizing $\ddot{Z}_3([j])$, and $\ddot{Z}_3([j])$ can be minimized by resolving the assignment problem. Let ρ_{il} represent the weight of job $\check{T}_i$ at location l, i.e.,

$$\rho_{il} = \begin{cases} p_i\left(\sum\limits_{m=1}^{l-1}\mu_m + \alpha\sum\limits_{m=1}^{j-1}\mu_m + \sum\limits_{m=1}^{n}\omega_m + \alpha\sum\limits_{m=1}^{n}\omega_m\right) & 1 \le l \le j-1 \\ \varepsilon_i\,p_i\left(\sum\limits_{m=1}^{l}\mu_m + \sum\limits_{m=1}^{n}\omega_m\right) & j \le l \le h-1 \\ \varepsilon_i\,p_i\sum\limits_{m=l+1}^{n}\nu_m & h \le l \le n \end{cases} \tag{16}$$

The problem can be translated into the assignment problem below:

$$\begin{aligned} Min \quad & \ddot{Z}_3([j]) = \sum_{i=1}^{n}\sum_{l=1}^{n}\rho_{il}\,\chi_{il} \\ s.t \quad & \sum_{i=1}^{n}\chi_{il} = 1 \qquad l = 1,\ldots,n \\ & \sum_{l=1}^{n}\chi_{il} = 1 \qquad i = 1,\ldots,n \\ & \chi_{il} = 0 \text{ or } 1 \qquad 1 \le i,l \le n \end{aligned} \tag{17}$$

where χ_{il} is a 0 or 1 variable, if the job $\check{T}_i$ is at position l, $\chi_{il} = 1$, otherwise $\chi_{il} = 0$.

Case 6. If $j > h$, the schedule is the same as Case 2, where $q_{opt} = S_{[h]} = \sum\limits_{i=1}^{h-1} p_{[i]}$, and $\varphi(t) = t_0 + \alpha\sum\limits_{i=1}^{j-1} p_{[i]}$. Then the objective function is stated as

$$\begin{aligned} \ddot{Z}([j]) &= \sum_{i=1}^{n}(\mu_i\,\widetilde{E_{[i]}} + \nu_i\,\widetilde{L_{[i]}} + \omega_i\,q_{opt}) \\ &= \sum_{i=1}^{h}\mu_i\left(d_{[i]} - C_{[i]}\right) + \sum_{i=h+1}^{j-1}\nu_i\left(C_{[i]} - d_{[i]}\right) + \sum_{i=j}^{n}\nu_i\left(C_{[i]} - d_{[i]}\right) + \sum_{i=1}^{n}\omega_i\,q_{opt} \\ &= \sum_{i=1}^{h-1} p_{[i]}\left(\sum_{m=1}^{i}\mu_m\right) + \sum_{i=h}^{j-1} p_{[i]}\left(\sum_{m=i+1}^{j-1}\nu_m\right) + \sum_{i=h}^{j-1} p_{[i]}\left(\sum_{m=j}^{n}\nu_m\right) + t_0\sum_{m=j}^{n}\nu_m \\ &\quad + \sum_{i=1}^{j-1} p_{[i]}\left(\alpha\sum_{m=j}^{n}\nu_m\right) + \sum_{i=j}^{n}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=i+1}^{n}\nu_m\right) + \sum_{i=1}^{h-1} p_{[i]}\left(\sum_{m=1}^{n}\omega_m\right) \\ &= \sum_{i=1}^{h-1} p_{[i]}\left(\sum_{m=1}^{i}\mu_m + \alpha\sum_{m=j}^{n}\nu_m + \sum_{m=1}^{n}\omega_m\right) + \sum_{i=h}^{j-1} p_{[i]}\left(\sum_{m=i+1}^{n}\nu_m + \alpha\sum_{m=j}^{n}\nu_m\right) \\ &\quad + \sum_{i=j}^{n}\varepsilon_{[i]}\,p_{[i]}\left(\sum_{m=i+1}^{n}\nu_m\right) + t_0\sum_{m=j}^{n}\nu_m \\ &= \ddot{Z}_4([j]) + f_2(t_0) \end{aligned} \tag{18}$$

where $f_2(t_0)$ same as it in Case 2, and

$$\ddot{Z}_4([j]) = \sum_{i=1}^{h-1} p_{[i]} \left(\sum_{m=1}^{i} \mu_m + \alpha \sum_{m=j}^{n} \nu_m + \sum_{m=1}^{n} \omega_m \right) + \sum_{i=h}^{j-1} p_{[i]} \left(\sum_{m=i+1}^{n} \nu_m + \alpha \sum_{m=j}^{n} \nu_m \right) + \sum_{i=j}^{n} \varepsilon_{[i]} p_{[i]} \left(\sum_{m=i+1}^{n} \nu_m \right) \tag{19}$$

is only related to the location of the maintenance activity. Similarly to Case 5 (i.e., $j \leq h$), let

$$\Omega_{il} = \begin{cases} p_i \left(\sum\limits_{m=1}^{l} \mu_m + \alpha \sum\limits_{m=j}^{n} \nu_m + \sum\limits_{m=1}^{n} \omega_m \right) & 1 \leq l \leq h-1 \\ p_i \left(\sum\limits_{m=l+1}^{n} \nu_m + \alpha \sum\limits_{m=j}^{n} \nu_m \right) & h \leq l \leq j-1 \\ \varepsilon_i \, p_i \left(\sum\limits_{m=l+1}^{n} \nu_m \right) & j \leq l \leq n \end{cases} \tag{20}$$

The problem can be translated into the assignment problem below:

$$\begin{aligned} Min \quad & \ddot{Z}_4([j]) = \sum_{i=1}^{n} \sum_{l=1}^{n} \Omega_{il} \, \varsigma_{il} \\ s.t \quad & \sum_{i=1}^{n} \varsigma_{il} = 1 \qquad l = 1, \ldots, n \\ & \sum_{l=1}^{n} \varsigma_{il} = 1 \qquad i = 1, \ldots, n \\ & \varsigma_{il} = 0 \; or \; 1 \qquad 1 \leq i, l \leq n \end{aligned} \tag{21}$$

where ς_{il} is a 0 or 1 variable, when the job $\check{T}_i$ is at position l, $\varsigma_{il} = 1$, otherwise $\varsigma_{il} = 0$.

Case 7. If $j = n+1$, the schedule is equal to *Case 3*, where $q_{opt} = S_{[h]} = \sum\limits_{i=1}^{h-1} p_{[i]}$, and $\varphi(t) = t_0 + \alpha \sum\limits_{i=1}^{n} p_{[i]}$. In this case, the objective function is

$$\begin{aligned} \ddot{Z}([n+1]) &= \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt}) \\ &= \sum_{i=1}^{h} \mu_i \left(C_{[h]} - C_{[i]} \right) + \sum_{i=h+1}^{n} \upsilon_i \left(C_{[i]} - C_{[h]} \right) + \sum_{i=1}^{n} \omega_i \left(\sum_{i=1}^{h-1} p_{[i]} \right) \\ &= \sum_{i=1}^{h-1} p_{[i]} \left(\sum_{m=1}^{i} \mu_m \right) + \sum_{i=h}^{n} p_{[i]} \left(\sum_{m=i+1}^{n} \upsilon_m \right) + \sum_{i=1}^{h-1} p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) \\ &= \sum_{i=1}^{h-1} p_{[i]} \left(\sum_{m=1}^{i} \mu_m + \sum_{m=1}^{n} \omega_m \right) + \sum_{i=h}^{n} p_{[i]} \left(\sum_{m=i+1}^{n} \upsilon_m \right) \\ &= \sum_{i=1}^{n} p_{[i]} \, \tau_i \end{aligned} \tag{22}$$

where

$$\tau_i = \begin{cases} \sum\limits_{m=1}^{i} \mu_m + \sum\limits_{m=1}^{n} \omega_m & 1 \leq i \leq h-1 \\ \sum\limits_{m=i+1}^{n} \upsilon_m & h \leq i \leq n \end{cases} \tag{23}$$

The optimal schedule σ and the minimum value of the objective function $\ddot{Z}$ can be found by matching the minimum weight τ_i with the maximum processing time p_i by using Lemma 3.

Case 8. If $j = 1$, the schedule is identical with Case 4, where $q_{opt} = S_{[h]} = \sum_{i=1}^{h-1} \varepsilon_{[i]} \, p_{[i]} + \varphi(t)$ and $\varphi(t) = t_0$. In this case, the objective function is

$$\begin{aligned}
&\ddot{Z}([1]) = \sum_{i=1}^{n}(\mu_i \, \widetilde{E_{[i]}} + \nu_i \, \widetilde{L_{[i]}} + \omega_i \, q_{opt}) \\
= \;& \sum_{i=1}^{h} \mu_i \left(d_{[i]} - C_{[i]} \right) + \sum_{i=h+1}^{n} \nu_i \left(C_{[i]} - d_{[i]} \right) + \sum_{i=1}^{n} \omega_i \, q_{opt} \\
= \;& \sum_{i=1}^{h-1} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{i} \mu_m \right) + \sum_{i=h}^{n} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=i+1}^{n} \nu_m \right) + \sum_{i=1}^{h-1} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{n} \omega_m \right) + t_0 \sum_{m=1}^{n} \omega_m \\
= \;& \sum_{i=1}^{h-1} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=1}^{i} \mu_m + \sum_{m=1}^{n} \omega_m \right) + \sum_{i=h}^{n} \varepsilon_{[i]} \, p_{[i]} \left(\sum_{m=i+1}^{n} \nu_m \right) + t_0 \sum_{m=1}^{n} \omega_m \\
= \;& \sum_{i=1}^{n} \varepsilon_{[i]} \, p_{[i]} \, \tau_i + f_3(t_0) \qquad (24)
\end{aligned}$$

where $f_3(t_0)$ is the same as it in Case 4. Similar to Case 7, the minimum weight τ_i with the maximum $\varepsilon_i \, p_i$ can minimize the objective function $\ddot{Z}$ by using Lemma 3.

From above cases, the optimal solution of $1|dma, SLK, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$ can be obtained by the next algorithm.

Theorem 2. *The optimal solution of* $1|dma, SLK, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$ *can be obtained in* $O(n^4)$ *time by using Algorithm 2.*

Proof. Similar to the proof of Theorem 1. □

Algorithm 2: Solution of $1|dma, SLK, b_i = (p_i, \varepsilon_i p_i)| \sum_{i=1}^{n} (\mu_i \widetilde{E_{[i]}} + \nu_i \widetilde{L_{[i]}} + \omega_i q_{opt})$.

Initialization: Let $\ddot{Z} = \infty$, $\ddot{\sigma}^* = 0$, $q_{opt}^* = 0$ and $j^* = 0$.
Step 1: Calculate h from Lemma 5.
Step 2: For $j = 1 \to n + 1$
If $j \leq h$, then
obtain the minimum value $\ddot{Z}([j])$ and the schedule $\ddot{\sigma}$ by using (14)–(17);
If $\ddot{Z}([j]) < \ddot{Z}$, then
let $\ddot{Z} = \ddot{Z}([j])$, $j^* = j$, $q_{opt}^* = q_{opt}$ and $\ddot{\sigma}^* = \ddot{\sigma}$;
If $j > h$, then
obtain the minimum value $\ddot{Z}([j])$ and the schedule $\ddot{\sigma}$ by using (18)–(21);
If $\ddot{Z}([j]) < \ddot{Z}$, then
let $\ddot{Z} = \ddot{Z}([j])$, $j^* = j$, $q_{opt}^* = q_{opt}$ and $\ddot{\sigma}^* = \ddot{\sigma}$;
If $j = n + 1$, then
acquire the optimal value of $\ddot{Z} = \ddot{Z}([n + 1])$ and the schedule $\ddot{\sigma}^*$ by the *HLP* rule;
Calculate q_{opt}^*;
If $j = 1$, then
obtain the optimal value of $\ddot{Z} = \ddot{Z}([1])$ and the schedule $\ddot{\sigma}^*$ by the *HLP* rule;
Calculate q_{opt}^*.
Step 3: Choose the minimum value $\ddot{Z}^* = \min\{\ddot{Z}[j],\ j = 1, 2, \ldots, n + 1\}$, and obtain the corresponding schedule $\ddot{\sigma}^*$, q_{opt}^* and j^*.

---

5. An Example and Computational Experiments

5.1. An Example

Let $n = 7$, $t_0 = 2$, $\alpha = 0.5$, $\mu_i = \omega_i = \nu_i$, respectively are 7, 4, 3, 1, 5, 13, 2, and the remaining data are given in Table 2.

Table 2. The remaining data.

$\check{T}_i$	$\check{T}_1$	$\check{T}_2$	$\check{T}_3$	$\check{T}_4$	$\check{T}_5$	$\check{T}_6$	$\check{T}_7$
p_i	9	11	6	10	15	12	8
ε_i	0.7	0.9	0.8	0.5	0.2	0.3	0.4

Solution: according to Lemmas 2 and 5, we can get $h = 4$ and $\mu_1 = 7$, $\mu_2 = 4$, $\mu_3 = 3$, $\omega_4 = 1$, $\nu_5 = 5$, $\nu_6 = 13$, $\nu_7 = 2$.

First, we consider the case of the common due date d_{opt}.

Case 9. $j = 2$: from Equation (5), we can get

$$\lambda_{il} = \begin{cases} p_i(\alpha * \mu_1 + n * \omega_m + \alpha * n * \omega_m) & l = 1 \\ \varepsilon_i p_i \left(\sum\limits_{m=1}^{l-1} \mu_m + n * \omega_m \right) & l = 2,3,4 \\ \varepsilon_i p_i \sum\limits_{m=l}^{n} v_m & l = 5,6,7 \end{cases} \tag{25}$$

and it is easy to see that

$\lambda_{11} = p_1(\alpha * \mu_1 + n * \omega_4 + \alpha * n * \omega_4) = 9 \times (0.5 \times 7 + 7 \times 1 + 0.5 \times 7 \times 1) = 126$;
$\lambda_{12} = \varepsilon_1 \, p_1(\mu_1 + n * \omega_4) = 0.7 \times 9 \times (7 + 7 \times 1) = 88.2$;
$\lambda_{13} = \varepsilon_1 \, p_1(\mu_1 + \mu_2 + n * \omega_4) = 0.7 \times 9 \times (7 + 4 + 7 \times 1) = 113.4$;
$\lambda_{14} = \varepsilon_1 \, p_1(\mu_1 + \mu_2 + \mu_3 + n * \omega_4) = 0.7 \times 9 \times (7 + 4 + 3 + 7 \times 1) = 132.4$;
$\lambda_{15} = \varepsilon_1 \, p_1(\nu_5 + \nu_6 + \nu_7) = 0.7 \times 9 \times (5 + 13 + 2) = 126$;
$\lambda_{16} = \varepsilon_1 \, p_1(\nu_6 + \nu_7) = 0.7 \times 9 \times (13 + 2) = 94.5$;
$\lambda_{17} = \varepsilon_1 \, p_1 * \nu_7 = 0.7 \times 9 \times 2 = 12.6$.

Similarly, the rest of the values of λ_{il} are given in Table 3,

Table 3. The values of λ_{il}.

$i \setminus l$	1	2	3	4	5	6	7
1	126	**88.2**	113.4	132.3	126	94.5	12.6
2	154	138.6	178.2	207.9	198	148.5	**19.8**
3	**84**	67.2	86.4	100.8	96	72	9.6
4	140	70	90	105	100	**75**	10
5	210	42	54	**63**	60	45	6
6	168	50.4	**64.8**	75.6	72	54	7.2
7	112	44.8	57.6	67.2	**64**	48	6.4

The bold values are the optimal solution.

We can obtain the optimal schedule $\dot{\sigma} = [\check{T}_2, \check{T}_7, \check{T}_1, \check{T}_6, \check{T}_4, \check{T}_3, \check{T}_5]$ by solving the assignment problem (6), and $\dot{Z}_1([2]) = 88.2 + 19.8 + 84 + 75 + 63 + 64.8 + 64 = 458.8$, $\varphi(t) = t_0 + \alpha * p_2 = 2 + 0.5 \times 11 = 7.5$, $d_{opt} = p_2 + \varphi(t) + \varepsilon_7 \, p_7 + \varepsilon_1 \, p_1 + \varepsilon_6 \, p_6 = 11 + 7.5 + 0.4 \times 8 + 0.7 \times 9 + 0.3 \times 12 = 31.6$, $f_1(t_0) = t_0(\mu_7 + n * \omega_4) = 2 \times (7 + 7 \times 1) = 28$, so $\dot{Z}([2]) = \dot{Z}_1([2]) + f_1(t_0) = 458.8 + 28 = 486.8$.

$j = 3$: Similarly, we can easily find the values of λ_{il} from Equation (5), and solve the assignment problem (6) to obtain the optimal schedule $\dot{\sigma} = [\check{T}_1, \check{T}_7, \check{T}_2, \check{T}_6, \check{T}_4, \check{T}_3, \check{T}_5]$, and $\dot{Z}_1([3]) = 562.6$, $\varphi(t) = t_0 + \alpha(p_1 + p_7) = 2 + 0.5 \times (9 + 8) = 10.5$, $d_{opt} = p_1 + p_7 + \varphi(t) + \varepsilon_2 \, p_2 + \varepsilon_6 \, p_6 = 9 + 8 + 10.5 + 0.9 \times 11 + 0.3 \times 12 = 41$, $f_1(t_0) = t_0(\mu_1 + \mu_2 + n * \omega_4) = 2 \times (7 + 4 + 7 \times 1) = 36$, so $\dot{Z}([3]) = \dot{Z}_1([3]) + f_1(t_0) = 562.6 + 36 = 598.6$.

$j = 4$: Similarly, we can easily find the values of λ_{il} from Equation (5), and solve the assignment problem (6) to obtain the optimal schedule $\dot{\sigma} = [\check{T}_1, \check{T}_7, \check{T}_3, \check{T}_6, \check{T}_4, \check{T}_5, \check{T}_2]$, and $\dot{Z}_1([4]) = 754.3$, $\varphi(t) = t_0 + \alpha(p_1 + p_7 + p_3) = 2 + 0.5 \times (9 + 8 + 6) = 13.5$, $d_{opt} = p_1 + p_7 + p_3 + \varphi(t) + \varepsilon_6 p_6 = 9 + 8 + 6 + 13.5 + 0.3 \times 12 = 40.1$, $f_1(t_0) = t_0(\mu_1 + \mu_2 + \mu_3 + n * \omega_4) = 2 \times (7 + 4 + 3 + 7 \times 1) = 42$, so $\dot{Z}([4]) = \dot{Z}_1([4]) + f_1(t_0) = 754.3 + 42 = 796.3$.

Case 10. $j = 5$: from Equation (9), the values of δ_{il} are known, similarly, we can obtain the optimal schedule $\dot{\sigma} = [\check{T}_2, \check{T}_7, \check{T}_4, \check{T}_1, \check{T}_5, \check{T}_6, \check{T}_3]$ by solving the assignment problem (10), and $\dot{Z}_2([5]) = 929.8$, $\varphi(t) = t_0 + \alpha(p_2 + p_7 + p_4 + p_1) = 2 + 0.5 \times (11 + 8 + 10 + 9) = 21$, $d_{opt} = p_2 + p_7 + p_4 + p_1 = 11 + 8 + 10 + 9 = 38$, $f_2(t_0) = t_0(\nu_5 + \nu_6 + \nu_7) = 2 \times (5 + 13 + 2) = 40$, so $\dot{Z}([5]) = \dot{Z}_2([5]) + f_2(t_0) = 929.8 + 40 = 969.8$.

$j = 6$: Similarly, we can obtain the values of δ_{il} from Equation (9) and tackle the assignment problem (10) to get the optimal schedule $\dot{\sigma} = [\check{T}_3, \check{T}_1, \check{T}_4, \check{T}_2, \check{T}_6, \check{T}_7, \check{T}_5]$, and $\dot{Z}_2([6]) = 1047.2$, $\varphi(t) = t_0 + \alpha(p_3 + p_1 + p_4 + p_2 + p_6) = 2 + 0.5 \times (6 + 9 + 10 + 11 + 12) = 26$, $d_{opt} = p_3 + p_1 + p_4 + p_2 = 6 + 9 + 10 + 11 = 36$, $f_2(t_0) = t_0(\nu_6 + \nu_7) = 2 \times (13 + 2) = 30$, so $\dot{Z}([6]) = \dot{Z}_2([6]) + f_2(t_0) = 1047.2 + 30 = 1077.2$.

$j = 7$: Similarly, we can get the values of δ_{il} from Equation (9) and tackle the assignment problem (10) to acquire the optimal schedule $\dot{\sigma} = [\check{T}_3, \check{T}_2, \check{T}_4, \check{T}_6, \check{T}_7, \check{T}_1, \check{T}_5]$, and $\dot{Z}_2([7]) = 898$, $\varphi(t) = t_0 + \alpha(p_3 + p_2 + p_4 + p_6 + p_7 + p_1) = 2 + 0.5 \times (6 + 11 + 10 + 12 + 8 + 9) = 30$, $d_{opt} = p_3 + p_2 + p_4 + p_6 = 6 + 11 + 10 + 12 = 39$, $f_2(t_0) = t_0 * \nu_7 = 2 \times 2 = 4$, so $\dot{Z}([7]) = \dot{Z}_2([7]) + f_2(t_0) = 898 + 4 = 902$.

Case 11. $j = 8$: From (12), we can obtain $\pi_1 = n * \omega_4 = 7 \times 1 = 7, \pi_2 = \mu_1 + n * \omega_4 = 7 + 7 \times 1 = 14, \pi_3 = \mu_1 + \mu_2 + n * \omega_4 = 7 + 4 + 7 \times 1 = 18, \pi_4 = \mu_1 + \mu_2 + \mu_3 + n * \omega_4 = 7 + 4 + 3 + 7 \times 1 = 21, \pi_5 = \nu_5 + \nu_6 + \nu_7 = 5 + 13 + 2 = 20, \pi_6 = \nu_6 + \nu_7 = 13 + 2 = 15, \pi_7 = \nu_7 = 2$, according to the HLP rule the optimal schedule is $\dot{\sigma} = [\check{T}_6, \check{T}_2, \check{T}_1, \check{T}_3, \check{T}_7, \check{T}_4, \check{T}_5]$, and $\varphi(t) = t_0 + \alpha(p_6 + p_2 + p_1 + p_3 + p_7 + p_4 + p_5) = 2 + 0.5 \times (12 + 11 + 9 + 6 + 8 + 10 + 15) = 37.5$, $d_{opt} = p_6 + p_2 + p_1 + p_3 = 12 + 11 + 9 + 6 = 38$, $\dot{Z}([8]) = p_6 \pi_1 + p_2 \pi_2 + p_1 \pi_3 + p_3 \pi_4 + p_7 \pi_5 + p_4 \pi_6 + p_5 \pi_7 = 12 \times 7 + 11 \times 14 + 9 \times 18 + 6 \times 21 + 8 \times 20 + 10 \times 15 + 15 \times 2 = 866$.

Case 12. $j = 1$: Similar to Case 11, we can know $\pi_1 = 7, \pi_2 = 14, \pi_3 = 18, \pi_4 = 21, \pi_5 = 20, \pi_6 = 15, \pi_6 = 2$, meanwhile $\varepsilon_1 p_1 = 0.7 \times 9 = 6.3, \varepsilon_2 p_2 = 0.9 \times 11 = 9.9, \varepsilon_3 p_3 = 0.8 \times 6 = 4.8, \varepsilon_4 p_4 = 0.5 \times 10 = 5, \varepsilon_5 p_5 = 0.2 \times 15 = 3, \varepsilon_6 p_6 = 0.3 \times 12 = 3.6, \varepsilon_7 p_7 = 0.4 \times 8 = 3.2$, using the HLP rule the optimal schedule is $\dot{\sigma} = [\check{T}_1, \check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_2]$, and $f_3(t_0) = t_0 * n * \omega_4 = 2 \times 7 \times 1 = 14$, $\varphi(t) = t_0 = 2$, $d_{opt} = \varepsilon_1 p_1 + \varepsilon_4 p_4 + \varepsilon_6 p_6 + \varepsilon_5 p_5 + \varphi(t) = 6.3 + 5 + 3.6 + 3 + 2 = 19.9$, so $\dot{Z}([1]) = \varepsilon_1 p_1 * \pi_1 + \varepsilon_4 p_4 * \pi_2 + \varepsilon_6 p_6 * \pi_3 + \varepsilon_5 p_5 * \pi_4 + \varepsilon_7 p_7 * \pi_5 + \varepsilon_3 p_3 * \pi_6 + \varepsilon_2 p_2 * \pi_7 + f_3(t_0) = 6.3 \times 7 + 5 \times 14 + 3.6 \times 18 + 3 \times 21 + 3.2 \times 20 + 4.8 \times 15 + 9.9 \times 2 + 14 = 397.7 + 14 = 411.7$.

Next, we consider the case of the common flow allowance q_{opt}.

Case 13. $j = 2$: Similar to Case 9, the values of ρ_{il} can been gathered from (16), and we can obtain the optimal schedule $\ddot{\sigma} = [\check{T}_6, \check{T}_7, \check{T}_1, \check{T}_5, \check{T}_3, \check{T}_2, \check{T}_4]$ by solving the assignment problem (17), and $\ddot{Z}_3([2]) = 363.4$, $\varphi(t) = t_0 + \alpha * p_6 = 2 + 0.5 \times 12 = 8$, $q_{opt} = p_6 + \varphi(t) + \varepsilon_7 p_7 + \varepsilon_1 p_1 = 12 + 8 + 0.4 \times 8 + 0.7 \times 9 = 29.5$, $f_1(t_0) = 28$, so $\ddot{Z}([2]) = \ddot{Z}_3([2]) + f_1(t_0) = 363.4 + 28 = 391.4$.

$j = 3$: Similarly, we can easily find the values of ρ_{il} from (16), and solve the assignment problem (17) so that we can obtain the optimal schedule $\ddot{\sigma} = [\check{T}_1, \check{T}_7, \check{T}_2, \check{T}_6, \check{T}_3, \check{T}_5, \check{T}_4]$, and $\ddot{Z}_3([3]) = 473$, $\varphi(t) = t_0 + \alpha(p_1 + p_7) = 2 + 0.5 \times (9 + 8) = 10.5$, $q_{opt} = p_1 + p_7 + \varphi(t) + \varepsilon_2 p_2 = 9 + 8 + 10.5 + 0.9 \times 11 = 37.4$, $f_1(t_0) = 36$, so $\ddot{Z}([3]) = \ddot{Z}_3([3]) + f_1(t_0) = 473 + 36 = 509$.

$j = 4$: Similarly, we can obtain the values of ρ_{il} from (16), and tackle the assignment problem (17) to acquire the optimal schedule $\ddot{\sigma} = [\check{T}_1, \check{T}_7, \check{T}_3, \check{T}_6, \check{T}_4, \check{T}_5, \check{T}_2]$, and $\ddot{Z}_3([4]) = 648.5$, $\varphi(t) = t_0 + \alpha(p_1 + p_7 + p_3) = 2 + 0.5 \times (9 + 8 + 6) = 13.5$, $q_{opt} =$

$p_1 + p_7 + p_3 + \varphi(t) = 9 + 8 + 6 + 13.5 = 36.5$, $f_1(t_0) = 42$, so $\ddot{Z}([4]) = \ddot{Z}_3([4]) + f_1(t_0) = 648.5 + 42 = 690.5$.

Case 14. $j = 5$: from (20), the values of Ω_{il} are given, similarly, we can get the optimal schedule $\ddot{\sigma} = [\check{T}_2, \check{T}_7, \check{T}_3, \check{T}_1, \check{T}_5, \check{T}_6, \check{T}_4]$ by solving the assignment problem (21), and $\ddot{Z}_4([5]) = 970.2$, $\varphi(t) = t_0 + \alpha(p_2 + p_7 + p_3 + p_1) = 2 + 0.5 \times (11 + 8 + 6 + 9) = 19$, $q_{opt} = p_2 + p_7 + p_3 = 11 + 8 + 6 = 25$, $f_2(t_0) = 40$, so $\ddot{Z}([5]) = \ddot{Z}_4([5]) + f_2(t_0) = 970.2 + 40 = 1010.2$.

$j = 6$: Similarly, we can obtain the values of Ω_{il} from (20), and tackle the assignment problem (21) to know the optimal schedule $\ddot{\sigma} = [\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_5, \check{T}_6, \check{T}_7, \check{T}_4]$, and $\ddot{Z}_4([6]) = 1088$, $\varphi(t) = t_0 + \alpha(p_2 + p_1 + p_3 + p_5 + p_6) = 2 + 0.5 \times (11 + 9 + 6 + 15 + 12) = 28.5$, $q_{opt} = p_2 + p_1 + p_3 = 11 + 9 + 6 = 26$, $f_2(t_0) = 30$, so $\ddot{Z}([6]) = \ddot{Z}_4([6]) + f_2(t_0) = 1088 + 30 = 1118$.

$j = 7$: Similarly, we can obtain the values of Ω_{il} from (20), and tackle the assignment problem (21) to acquire the optimal schedule $\ddot{\sigma} = [\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_5, \check{T}_7, \check{T}_6, \check{T}_4]$, and $\ddot{Z}_4([7]) = 832$, $\varphi(t) = t_0 + \alpha(p_2 + p_1 + p_3 + p_5 + p_7 + p_6) = 2 + 0.5 \times (11 + 9 + 6 + 15 + 8 + 12) = 32.5$, $q_{opt} = p_2 + p_1 + p_3 = 11 + 9 + 6 = 26$, $f_2(t_0) = 4$, so $\ddot{Z}([7]) = \ddot{Z}_4([7]) + f_2(t_0) = 832 + 4 = 836$.

Case 15. $j = 8$: From (23), we can get $\tau_1 = \mu_1 + n * \omega_4 = 7 + 7 \times 1 = 14, \tau_2 = \mu_1 + \mu_2 + n * \omega_4 = 7 + 4 + 7 \times 1 = 18, \tau_3 = \mu_1 + \mu_2 + \mu_3 + n * \omega_4 = 7 + 4 + 3 + 7 \times 1 = 21, \tau_4 = \nu_5 + \nu_6 + \nu_7 = 5 + 13 + 2 = 20, \tau_5 = \nu_6 + \nu_7 = 13 + 2 = 15, \tau_6 = \nu_7 = 2, \tau_7 = 0$, according to the *HLP* rule the optimal schedule is $\ddot{\sigma} = [\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_7, \check{T}_4, \check{T}_6, \check{T}_5]$, and $\varphi(t) = t_0 + \alpha(p_2 + p_1 + p_3 + p_7 + p_4 + p_6 + p_5) = 2 + 0.5 \times (11 + 9 + 6 + 8 + 10 + 12 + 15) = 37.5$, $q_{opt} = p_2 + p_1 + p_3 = 11 + 9 + 6 = 26$, $\ddot{Z}([8]) = p_2\,\tau_1 + p_1\,\tau_2 + p_3\,\tau_3 + p_7\,\tau_4 + p_4\,\tau_5 + p_6\,\tau_6 + p_5\,\tau_7 = 11 \times 14 + 9 \times 18 + 6 \times 21 + 8 \times 20 + 10 \times 15 + 12 \times 2 + 15 \times 0 = 776$.

Case 16. $j = 1$: Similar to Cases 12 and 15, we can obtain $\tau_1 = 14, \tau_2 = 18, \tau_3 = 21, \tau_4 = 20, \tau_5 = 15, \tau_6 = 2, \tau_7 = 0$ and $\varepsilon_1\,p_1 = 6.3, \varepsilon_2\,p_2 = 9.9, \varepsilon_3\,p_3 = 4.8, \varepsilon_4\,p_4 = 5, \varepsilon_5\,p_5 = 3, \varepsilon_6\,p_6 = 3.6, \varepsilon_7\,p_7 = 3.2$, and according to the *HLP* rule the optimal schedule is $\ddot{\sigma} = [\check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_1, \check{T}_2]$, and $f_3(t_0) = 14$, $\varphi(t) = t_0 = 2$, $q_{opt} = \varepsilon_4\,p_4 + \varepsilon_6\,p_6 + \varepsilon_5\,p_5 + \varphi(t) = 5 + 3.6 + 3 + 2 = 13.6$, so $\ddot{Z}([1]) = \varepsilon_4\,p_4 * \tau_1 + \varepsilon_6\,p_6 * \tau_2 + \varepsilon_5\,p_5 * \tau_3 + \varepsilon_7\,p_7 * \tau_4 + \varepsilon_3\,p_3 * \tau_5 + \varepsilon_1\,p_1 * \tau_6 + \varepsilon_2\,p_2 * \tau_7 + f_3(t_0) = 5 \times 14 + 3.6 \times 18 + 3 \times 21 + 3.2 \times 20 + 4.8 \times 15 + 6.3 \times 2 + 9.9 \times 0 + 14 = 346.4 + 14 = 360.4$.

After the above calculation, the results are shown in Table 4.

Table 4. The results of the example.

	d_{opt}	q_{opt}	$\dot{Z}([j])$	$\ddot{Z}([j])$	$\dot{\sigma}$	$\ddot{\sigma}$
$j = 1$	19.9	13.6	411.7	360.4	$[\check{T}_1, \check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_2]$	$[\check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_1, \check{T}_2]$
$j = 2$	31.6	29.5	486.8	391.4	$[\check{T}_2, \check{T}_7, \check{T}_1, \check{T}_6, \check{T}_4, \check{T}_3, \check{T}_5]$	$[\check{T}_6, \check{T}_7, \check{T}_1, \check{T}_5, \check{T}_3, \check{T}_2, \check{T}_4]$
$j = 3$	41	37.4	598.6	509	$[\check{T}_1, \check{T}_7, \check{T}_2, \check{T}_6, \check{T}_4, \check{T}_3, \check{T}_5]$	$[\check{T}_1, \check{T}_7, \check{T}_2, \check{T}_6, \check{T}_3, \check{T}_5, \check{T}_4]$
$j = 4$	40.1	36.5	796.3	690.5	$[\check{T}_1, \check{T}_7, \check{T}_3, \check{T}_6, \check{T}_4, \check{T}_5, \check{T}_2]$	$[\check{T}_1, \check{T}_7, \check{T}_3, \check{T}_6, \check{T}_4, \check{T}_5, \check{T}_2]$
$j = 5$	38	25	969.8	1010.2	$[\check{T}_2, \check{T}_7, \check{T}_4, \check{T}_1, \check{T}_5, \check{T}_6, \check{T}_3]$	$[\check{T}_2, \check{T}_7, \check{T}_3, \check{T}_1, \check{T}_5, \check{T}_6, \check{T}_4]$
$j = 6$	36	26	1077.2	1118	$[\check{T}_3, \check{T}_1, \check{T}_4, \check{T}_2, \check{T}_6, \check{T}_7, \check{T}_5]$	$[\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_5, \check{T}_6, \check{T}_7, \check{T}_4]$
$j = 7$	39	26	902	836	$[\check{T}_3, \check{T}_2, \check{T}_4, \check{T}_6, \check{T}_7, \check{T}_1, \check{T}_5]$	$[\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_5, \check{T}_7, \check{T}_6, \check{T}_4]$
$j = 8$	38	26	866	776	$[\check{T}_6, \check{T}_2, \check{T}_1, \check{T}_3, \check{T}_7, \check{T}_4, \check{T}_5]$	$[\check{T}_2, \check{T}_1, \check{T}_3, \check{T}_7, \check{T}_4, \check{T}_6, \check{T}_5]$

We can see that the optimal solution to the question about *CON* is $j = 1$, $\dot{\sigma}^* = [\check{T}_1, \check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_2]$, $d^*_{opt} = 19.9$ and the optimal solution to the question about *SLK* is $j = 1$, $\ddot{\sigma}^* = [\check{T}_4, \check{T}_6, \check{T}_5, \check{T}_7, \check{T}_3, \check{T}_1, \check{T}_2]$, $q^*_{opt} = 13.6$.

5.2. Computational Experiments

To test the validity of Algorithms 1 and 2, the examples are generated randomly. Microsoft visual C++ 2022 was applied to code Algorithms 1 and 2. For every problem size, 20 cases were created and coded on a PC with a 3.10 GHz CPU and 8.00 GB RAM. The features of the examples are listed below:

(1) $n = 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200$, $t_0 = 5$ and $\alpha = 0.1$;
(2) p_i $(i = 1, 2, \ldots, n)$ is evenly distributed over [1, 100];
(3) ε_i $(i = 1, 2, \ldots, n)$ is evenly distributed over [0.5, 0.95];
(4) μ_i, ν_i and ω_i $(i = 1, 2, \ldots, n)$ are evenly distributed over [1, 50].

The computational tests for Algorithms 1 and 2 are given as follows. The minimum (min), average (mean) and maximum (max) CPU times (milliseconds (ms)) are shown in Table 5. From Table 5, we can see that Algorithms 1 and 2 are effective and their CPU times increase moderately as n increases from 30 to 200, and the maximum CPU time is 830,682.20 ms for $n = 200$.

Table 5. CPU times of algorithms.

	Algorithm 1 (ms)			Algorithm 2 (ms)		
Jobs (n)	**Min**	**Mean**	**Max**	**Min**	**Mean**	**Max**
30	96.15	104.12	116.47	105.57	123.42	131.36
40	403.13	447.04	468.66	456.02	490.37	533.89
50	1100.62	1133.92	1207.00	1800.97	1869.21	1969.54
60	2385.21	2415.26	2480.62	2406.82	3211.60	3621.18
70	4764.21	4824.51	4962.15	4427.60	5595.50	6072.50
80	8988.25	9047.69	9174.25	9546.26	10,260.97	11,854.23
90	15,607.89	15,735.94	15,993.65	18,732.12	19,081.05	20,657.20
100	25,808.64	26,019.12	26,248.52	25,948.21	31,214.80	34,985.24
110	40,791.23	40,888.99	41,023.33	44,216.25	47,840.15	50,154.56
120	62,351.40	62,705.79	62,994.49	69,263.57	72,753.31	76,016.59
130	91,992.40	92,142.26	92,376.48	95,431.29	104,475.77	112,460.39
140	131,937.56	132,467.57	134,070.77	153,157.28	159,192.78	165,218.85
150	185,206.31	186,061.38	187,231.45	185,069.15	197,686.43	204,623.32
160	255,436.23	256,559.22	259,944.61	259,783.23	266,256.81	279,325.82
170	342,530.15	342,989.64	344,039.89	382,981.28	389,495.60	391,893.45
180	453,007.52	455,489.67	457,957.65	494,021.26	496,965.56	501,507.23
190	594,550.76	597,809.51	602,660.32	625,462.31	639,839.76	661,165.27
200	763,412.32	780,313.79	791,677.65	812,131.25	822,461.92	830,682.20

6. Conclusions

In this paper, we studied the single-machine due-date assignment problem with a deteriorating maintenance activity. Under the common and slack due-date assignments, the purpose of the problem is to find the optimal job schedule, the position of the maintenance activity, the optimal value of the common due date or the optimal value of the common flow-allowance so that the linear weighted sum of earliness, tardiness and due-date assignment cost is minimized. The problem is proved to be solved in polynomial time. Through computational complexity analysis and experimentation, we demonstrated that the proposed algorithm (approach) performs very well. In the future, other non-regular objectives with deteriorating maintenance activity can be studied, or multi-machine (flexible job shop, see Zhang et al. [39] and Song et al. [40]) due-date assignment problems with deteriorating maintenance activity can be addressed.

Author Contributions: Conceptualization, W.W., D.-Y.L. and J.-B.W.; methodology, W.W. and D.-Y.L.; investigation, J.-B.W.; writing, review and editing, W.W. and D.-Y.L.; supervision, J.-B.W.; writing-original draft preparation, W.W.. All authors have read and agreed to the published version of the manuscript.

Funding: This Work was supported by LiaoNing Revitalization Talents Program (grant no. XLYC2002017).

Data Availability Statement: The data used to support this paper are available from the corresponding author upon request.

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

Appendix A

Proof. When $j = n+1$, we first consider a schedule σ, and the common due date d_{opt} satisfies $C_{[k]} < d_{opt} < C_{[k+1]}$, $\dot{Z}$ is the corresponding value of objective function, let $\dot{x} = d_{opt} - C_{[k]}$, $\dot{y} = C_{[k+1]} - d_{opt}$ ($\dot{x}, \dot{y} \geq 0$), $\dot{Z}'$ and $\dot{Z}''$ are the corresponding objective function values of $d_{opt} = C_{[k]}$ and $d_{opt} = C_{[k+1]}$, respectively, therefore

$$\dot{Z}' = \dot{Z} + \dot{x}\left[\sum_{i=1}^{n}(\nu_i - \omega_i) - \sum_{i=1}^{k}(\mu_i - \nu_i)\right]$$

$$\dot{Z}'' = \dot{Z} - \dot{y}\left[\sum_{i=1}^{n}(\nu_i - \omega_i) - \sum_{i=1}^{k}(\mu_i - \nu_i)\right].$$

Obviously, if $\sum_{i=1}^{n}(\nu_i - \omega_i) \leq \sum_{i=1}^{k}(\mu_i + \nu_i)$, $\dot{Z}' \leq \dot{Z}$, otherwise $\dot{Z}'' \leq \dot{Z}$, this means that d_{opt} is equal to the completion time of some job.

We assume that the value of d_{opt} coincides with the completion time of the job at the hth position, i.e., $d_{opt} = C_{[h]}$, then for $\forall\theta \gg 0$, the common due date shifted θ unit to the left, and it is obtained

$$\dot{Z}\left(C_{[h]} + \theta, \sigma\right) - \dot{Z}\left(C_{[h]}, \sigma\right) \geq 0,$$

we have

$$\left(\sum_{i=1}^{h}\mu_i - \sum_{i=h+1}^{n}\nu_i + \sum_{i=1}^{n}\omega_i\right) \geq 0.$$

If the common due date shifted θ unit to the right, we have

$$\dot{Z}\left(C_{[h]} - \theta, \sigma\right) - \dot{Z}\left(C_{[h]}, \sigma\right) \geq 0,$$

and

$$\left(\sum_{i=1}^{h-1}\mu_i - \sum_{i=h}^{n}\nu_i + \sum_{i=1}^{n}\omega_i\right) \leq 0$$

Thus, h satisfies

$$\left(\sum_{i=1}^{h}\mu_i - \sum_{i=h+1}^{n}\nu_i + \sum_{i=1}^{n}\omega_i\right) \geq 0 \; and \; \left(\sum_{i=1}^{h-1}\mu_i - \sum_{i=h}^{n}\nu_i + \sum_{i=1}^{n}\omega_i\right) \leq 0.$$

Similarly, the above result can be obtained when $j = 1, \ldots, n$, i.e., for any given schedule, $d_{opt} = C_{[h]}$. □

References

1. Lee, C.Y.; Leon, V.J. Machine scheduling with a rate-modifying activity. *Eur. J. Oper. Res.* **2001**, *128*, 119–128. [CrossRef]
2. Ma, Y.; Chu, C.; Zuo, C. A survey of scheduling with deterministic machine availability constraints. *Comput. Ind. Eng.* **2010**, *58*, 199–211. [CrossRef]
3. Strusevich, V.A.; Rustogi, K. *Scheduling with Time-Changing Effects and Rate-Modifying Activities*; Springer: Berlin/Heidelberg, Germany, 2017.
4. Mosheiova, G.; Oron, D. Due-date assignment and maintenance activity scheduling problem. *Math. Comput. Model.* **2006**, *44*, 1053–1057. [CrossRef]

5. Wang, X.-Y.; Wang, M.-Z. Single machine common flow allowance scheduling with a rate-modifying activity. *Comput. Ind. Eng.* **2010**, *59*, 898–902. [CrossRef]
6. Yin, Y.; Cheng, T.C.E.; Xu, D.; Wu, C.-C. Common due date assignment and scheduling with a rate-modifying activity to minimize the due date, earliness, tardiness, holding, and batch delivery cost. *Comput. Ind. Eng.* **2012**, *63*, 223–234. [CrossRef]
7. Bai, J.; Li, Z.-R.; Wang, J.-J.; Huang, X. Single machine common flow allowance scheduling with deteriorating jobs and a rate-modifying activity. *Appl. Math. Model.* **2014**, *38*, 5431–5438. [CrossRef]
8. Cheng, M.; Xiao, S.; Luo, R.; Lian, Z. Single-machine scheduling problems with a batch-dependent aging effect and variable maintenance activities. *Int. J. Prod. Res.* **2018**, *56*, 7051–7063. [CrossRef]
9. Detti, P.; Nicosia, G.; Pacifici, A.; De Lara, G.Z.M. Robust single machine scheduling with a flexible maintenance activity. *Comput. Oper. Res.* **2019**, *107*, 19–31. [CrossRef]
10. Wang, J.-B.; Hu, Y.; Zhang, B. Common due-window assignment for single-machine scheduling with generalized earliness/tardiness penalties and a rate-modifying activity. *Eng. Optim.* **2021**, *53*, 496–512. [CrossRef]
11. Zhao, X.; Xu, J.; Wang, J.-B.; Li, L. Bicriteria common flow allowance scheduling with aging effect, convex resource allocation, and a rate-modifying activity on a single machine. *Asia-Pac. J. Oper. Res.* **2022**, *39*, 2150046. [CrossRef]
12. Sun, X.; Liu, T.; Geng, X.-N.; Hu, Y.; Xu, J.-X. Optimization of scheduling problems with deterioration effects and an optional maintenance activity. *J. Sched.* **2023**. [CrossRef]
13. Ji, P.; Li, G.; Huo, Y.; Wang, J.-B. Single-machine common flow allowance scheduling with job-dependent aging effects and a deteriorating maintenance activity. *Optim. Lett.* **2014**, *8*, 1389–1400. [CrossRef]
14. Fan, Y.P.; Zhao, C.-L. Single machine scheduling with multiple common due date assignment and aging effect under a deteriorating maintenance activity consideration. *J. Appl. Math. Comput.* **2014**, *46*, 51–66. [CrossRef]
15. Mor, B.; Mosheiov, G. Scheduling a deteriorating maintenance activity and due-window assignment. *Comput. Oper. Res.* **2015**, *57*, 33–40. [CrossRef]
16. Li, S.-S.; Chen, R.X. Scheduling with rejection and a deteriorating maintenance activity on a single machine. *Asia-Pac. J. Oper. Res.* **2017**, *34*, 1750010. [CrossRef]
17. Zhu, H.; Li, M.; Zhou, Z.J. Machine scheduling with deteriorating and resource-dependent maintenance activity. *Comput. Ind. Eng.* **2015**, *88*, 479–486. [CrossRef]
18. Wang, J.-J.; Wang, J.-B.; Liu, F. Parallel machines scheduling with a deteriorating maintenance activity. *J. Oper. Res. Soc.* **2011**, *62*, 1898–1902. [CrossRef]
19. He, H.; Hu, Y.; Liu, W.-W. Scheduling with deterioration effects and maintenance activities under parallel processors. *Eng. Optim.* **2021**, *53*, 2070–2087. [CrossRef]
20. Jia, X.; Lv, D.-Y.; Hu, Y.; Wang, J.-B.; Wang, Z.; Wang, E. Slack due-window assignment scheduling problem with deterioration effects and a deteriorating maintenance activity. *Asia-Pac. J. Oper. Res.* **2022**, *39*, 2250005. [CrossRef]
21. Liu, W.; Wang, X.; Li, L.; Zhao, P. A maintenance activity scheduling with time-and-position dependent deteriorating effects. *Math. Biosci. Eng.* **2022**, *19*, 11756–11767. [CrossRef]
22. Zou, J.; Sui, Y.-K.; Gao, J.; Zhang, X.-Z. Parallel machines scheduling with deteriorating maintenance activities and job rejection. *Asia-Pac. J. Oper. Res.* **2023**, *40*, 2240013. [CrossRef]
23. Gordon, V.S.; Proth, J.-M.; Chu, C.B. A survey of the state-of-the-art of common due date assignment and scheduling research. *Eur. J. Oper. Res.* **2002**, *139*, 1–25. [CrossRef]
24. Gordon, V.S.; Proth, J.-M.; Chu, C.B. Due date assignment and scheduling: SLK, TWK and other due date assignment models. *Prod. Plan. Control* **2002**, *13*, 117–132. [CrossRef]
25. Qian, J.; Zhan, Y. The due date assignment scheduling problem with delivery times and truncated sum-of-processing-times-based learning effect. *Mathematics* **2021**, *9*, 3085. [CrossRef]
26. Lu, Y.-Y.; Wang, T.-T.; Wang, R.-Q.; Li, Y. A note on due-date assignment scheduling with job-dependent learning effects and convex resource allocation. *Eng. Optim.* **2021**, *53*, 1273–1281. [CrossRef]
27. Wang, W. Single-machine due-date assignment scheduling with generalized earliness/tardiness penalties including proportional setup times. *J. Appl. Math. Comput.* **2022**, *68*, 1013–1031. [CrossRef]
28. Brucker, P. *Scheduling Algorithms*; Springer: Berlin/Heidelberg, Germany, 2006.
29. Liu, W.; Hu, X.; Wang, X. Single machine scheduling with slack due dates assignment. *Eng. Optim.* **2017**, *49*, 709–717. [CrossRef]
30. Wang, S.-H.; Lv, D.-Y.; Wang, J.-B. Research on position-dependent weights scheduling with delivery times and truncated sum-of-processing-times-based learning effect. *J. Ind. Manag. Optim.* **2023**, *19*, 2824–2837. [CrossRef]
31. Wang, Y.-C.; Wang, S.-H.; Wang, J.-B. Resource allocation scheduling with position-dependent weights and generalized earliness-tardiness cost. *Mathematics* **2023**, *11*, 222. [CrossRef]
32. Jiang, C.; Zou, D.; Bai, D.; Wang, J.-B. Proportionate flowshop scheduling with position-dependent weights. *Eng. Optim.* **2020**, *52*, 37–52. [CrossRef]
33. Liu, W.; Yao, Y.; Jiang, C. Single-machine resource allocation scheduling with due-date assignment, deterioration effect and position-dependent weights. *Eng. Optim.* **2020**, *52*, 701–714. [CrossRef]
34. Lv, D.Y.; Wang, J.-B. Study on proportionate flowshop scheduling with due-date assignment and position-dependent weights. *Optim. Lett.* **2021**, *15*, 2311–2319. [CrossRef]

35. Wang, L.-Y.; Huang, X.; Liu, W.-W.; Wu, Y.; Wang, J.-B. Scheduling with position-dependent weights, due-date assignment and past-sequence-dependent setup times. *Rairo-Oper. Res.* **2021**, *55*, S2747–S2758. [CrossRef]
36. Wang, X.; Liu, W.; Li, L.; Zhao, P.; Zhang, R. Due date assignment scheduling with positional-dependent weights and proportional setup times. *Math. Biosci. Eng.* **2022**, *19*, 5104–5119. [CrossRef]
37. Sun, X.; Geng, X.-N.; Liu, T. Due-window assignment scheduling in the proportionate flow shop setting. *Ann. Oper. Res.* **2020**, *292*, 113–131. [CrossRef]
38. Hardy, G.H.; Littlewood, J.E.; Polya, G. *Inequalities*; Cambridge University Press: Cambridge, UK, 1976.
39. Zhang, H.; Qin, C.; Zhang, W.; Xu, Z.; Xu, G.; Gao, Z. Energy-saving scheduling for flexible job shop problem with AGV transportation considering emergencies. *Systems* **2023**, *11*, 103. [CrossRef]
40. Song, L.; Xu, Z.; Wang, C.; Su, J. A new decision method of flexible job shop rescheduling based on WOA-SVM. *Systems* **2023**, *11*, 59. [CrossRef]