Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2011, Article ID 151563, 19 pages
doi:10.1155/2011/151563

Research Article

Due-Window Assignment and Scheduling with Multiple Rate-Modifying Activities under the Effects of Deterioration and Learning

Zhanguo Zhu,[1, 2] Linyan Sun,[1] Feng Chu,[3] and Ming Liu[4]

[1] *School of Management, State Key Laboratory for Mechanical Manufacturing Systems Engineering and The Key Laboratory of the Ministry of Education for Process Control and Efficiency Engineering, Xi'an Jiaotong University, Shaanxi Province, Xi'an 710049, China*
[2] *Institut Charles Delaunay, Université de Technologie de Troyes and FRE CNRS 2848, Laboratoire d'Optimisation des Systémes Industriels (LOSI), 12 rue Marie Curie, BP 2060, 10010 Troyes Cedex, France*
[3] *Laboratoire d'Informatique, Biologie Intégrative et Systèmes Complexes (IBISC), FRE CNRS 3190, Université d'Evry Val d'Essonne, 40 rue du Pelvoux, CE1455 Courcouronnes, 91020 Evry Cedex, France*
[4] *School of Economics and Management, Tongji University, Shanghai 200092, China*

Correspondence should be addressed to Zhanguo Zhu, zhanguo.zhu@utt.fr

Received 3 September 2010; Accepted 31 January 2011

Academic Editor: Paulo Batista Gonçalves

This paper discusses due-window assignment and scheduling with multiple rate-modifying activities. Multiple types of rate-modifying activities are allowed to perform on a single machine. The learning effect and job deterioration are also integrated concurrently into the problem which makes the problem more realistic. The objective is to find jointly the optimal location to perform multiple rate-modifying activities, the optimal job sequence, and the optimal location and size of the due window to minimize the total earliness, tardiness, and due-window-related costs. We propose polynomial time algorithms for all the cases of the problem under study.

1. Introduction

With the complexity of the manufacturing activities more researchers focus on variants of classical scheduling problems that reflect the reality, such as learning effect, rate-modifying activity, deteriorating effects, and due-window assignment.

The phenomenon of the actual job processing times decreasing due to repetition of tasks by workers is known as the learning effect. Learning effect has received considerable attention in management science since it is first discovered by Wright [1]. However, the analysis of scheduling problems with learning effects is relatively recent. Biskup [2] and Cheng and Wang [3] were among the pioneers. Biskup [2] proposed a learning effect formulation which implied that learning primarily takes place as a result of repeating "processing time independent" operations and proved that with the introduction of learning to job processing times some cases of scheduling problems remain polynomially solvable. Mosheiov and Sidney [4] extended the setting of learning effect to the case of being job dependent. They proposed a new learning model in which the actual processing time of job j is $p_{jr} = p_j r^{a_j}$ if it is scheduled in position r, where a_j is a job-dependent negative parameter and p_j is the normal processing time. They also provided polynomial time solutions for several classical objective functions based on this realistic assumption. Koulamas and Kyparisis [5] studied single-machine and two machine flowshop scheduling with general learning functions and obtained some results on single-machine and special cases of two-machine. Wang et al. [6] studied single machine scheduling problem considering both learning effect and discounted costs. Kuo and Yang [7] introduced a time-independent learning effect into the single-machine group scheduling problems and provided two polynomial time algorithms to solve the problems of two different objectives.

More researchers focus on the topic of rate-modifying activity (RMA) since Lee and Leon [8] first presented this model. In scheduling problems, production rate can be changed by inserting this activity into the job sequence and no jobs are processed during the duration of this activity. Zhao et al. [9] studied two parallel machines scheduling problems in which each machine has a rate-modifying activity. They provided a polynomial algorithm for the total completion time minimization problem and a pseudopolynomial time dynamic programming for the total weighted completion time minimization problem under agreeable ratio condition. Lodree and Geiger [10] addressed a scheduling problem with a rate-modifying activity under simple linear deterioration and proposed an optimal policy to schedule the RMA in the middle of the task sequence under certain conditions. Ji and Cheng [11] studied scheduling with multiple rate-modifying activities. Different from the above literature, they discussed the case that there are multiple different types of rate-modifying activities on each machine. They proved that all the cases of the problem are polynomially solvable. S.-J. Yang and D.-L. Yang [12] analyzed scheduling problems with several maintenance activities. However, they considered three types of aging/deteriorating effects, respectively, and the objective is to minimize the total completion time.

Scheduling with deteriorating jobs, first introduced by J. N. D. Gupta and S. K. Gupta [13], and Browne and Yechiali [14], has received extensive attention in recent years. Deterioration discussed here means the actual job processing time is dependent on its normal processing time and actual starting time. Mosheiov [15] first investigated scheduling problem with the simple linear deteriorating jobs. Ng et al. [16] discussed three scheduling problems with deteriorating jobs to minimize the total completion time. Mosheiov [17], Lee et al. [18], and Sun et al. [19] studied job-shop scheduling problem with deteriorating jobs in different settings of environment. Gawiejnowicz [20] studied two scheduling problems with proportionally deteriorating jobs and they showed that these problems are both NP complete in ordinary sense or strong sense. Ji and Cheng [21] considered parallel machine scheduling problem with simple linear deterioration assumption. They proposed a polynomial time approximation scheme for the objective of minimizing total completion time. Wang and Sun [22] discussed the linear deterioration of job processing times and setup time in the context of group scheduling. Moreover, many studies devoted to scheduling problems with deteriorating jobs and learning effects such as Lee [23], Wang and Cheng [24], Cheng et al. [25], and Yang and Kuo [26].

As an important issue in modern manufacturing system, due window assignment has also received increasing attention. Distinct from due-date assignment (please see Gordon et al. [27], and Biskup and Simons [28]), due-window assignment allows a time internal and no penalized cost are incurred if the jobs are completed within this internal. Otherwise, related earliness and tardiness are taken into account according to the positions of jobs before/after due-window. Liman et al. [29] considered the single machine scheduling problem with common due-window which is an extension of former earliness-tardiness scheduling problem. They proposed a polynomial algorithm to find the optimal size, location of the window, and an optimal sequence to minimize the cost function. Mosheiov and Sarig [30] studied a single machine scheduling problem with due-window and a maintenance activity. They introduced a polynomial time solution to schedule the jobs, the due-window and the maintenance activity. Yang et al. [31] considered due-window assignment and scheduling with job-dependent aging effects and deteriorating maintenance. In their study, they proposed a model with a deteriorating maintenance and provided polynomial time solutions. Zhao and Tang [32] investigated due-window assignment and scheduling with a rate-modifying activity under the assumption of deteriorating jobs that the processing time of a job is a linear function of its starting time. They proposed an $O(n^4)$ algorithm to solve the problem optimally, where n is the number of jobs.

In this paper, we discuss single-machine scheduling problem with due-window assignment and multiple rate-modifying activities which is an extension of the work by Mosheiov and Sarig [30], Ji and Cheng [11], and Yang et al. [31]. In addition, learning effect, and job deterioration are also integrated concurrently into the problem which makes the problem more realistic. To our best knowledge, it is the first work that integrates due-window assignment, multiple rate-modifying activities, learning effect and job deterioration simultaneously. This paper is organized as follows. The problem is formulated in Section 2. Section 3 provides preliminary results related. An optimal policy is given in Section 4. The last section concludes this paper.

2. Problem Formulation

The problem we study can be stated as follows. There are given n independent and nonpreemptive jobs to be processed on a single machine. Each job j is available for processing at time 0 and has a normal processing time p_j, for $j = 1, 2, \ldots, n$. $J_{[r]}$ $(r = 1, 2, \ldots, n)$ denotes the job scheduled in the rth position. Similar to Yang and Kuo [26], we assume the model of learning and deteriorating effect is a combination of the job-dependent learning effect model by Mosheiov and Sidney [4] and the linear deterioration model by Mosheiov [33]. So the actual processing time of job j with learning effect and deteriorating effect if it is scheduled in the rth position in a sequence is given by $p_j^A = p_j(r)^{a_j} + bs_j$, for $j, r = 1, 2, \ldots, n$, where $a_j \leq 0$ is the job-dependent learning index of job $j, b > 0$ is the deterioration rate. s_j is the starting time of job j. In addition, we assume multiple rate-modifying activities are allowed on the machine to improve its production efficiency throughout the whole scheduling horizon. The lth rate-modifying activity with constant duration t_l is in i_l, if it is scheduled immediately after the completion of $J_{[i_l]}$, $l = 1, 2, \ldots, u$, as in Figure 1. If job j is processed in position r just after any rate-modifying activity l, its actual processing time becomes $p_j^A = \theta_{jl} p_j(r)^{a_j} + bs_j$, where $0 < \theta_{jl} \leq 1$ is job-dependent modifying rate. For a given schedule π, $C_j = C_j(\pi)$ denotes the completion time of job j, $j = 1, 2, \ldots, n$. In our problem all jobs are assumed to have a common due window. Let d_1 and d_2 denote the starting time and the finishing time of the due window, respectively. Let $D = d_2 - d_1$ denote the due-window size. $E_j = \max\{0, d_1 - C_j\}$

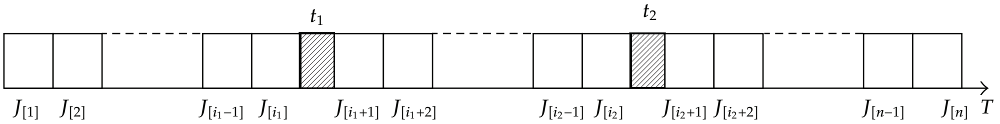

Figure 1: Structure of a schedule containing n jobs and two rate-modifying activities.

denotes the earliness of job j, $j = 1, 2, \ldots, n$. $T_j = \max\{0, C_j - d_2\}$ denotes the tardiness of job j, $j = 1, 2, \ldots, n$. Further, let $\alpha > 0$, $\beta > 0$, $\gamma > 0$ and $\delta > 0$ be the per time unit penalties for earliness, tardiness and due-window starting and due-window size, respectively. The objective is to determine the optimal due-window starting time d_1, the due-window size, the position to schedule multiple rate-modifying activities and to find a schedule π which minimizes the following cost function:

$$f(d_1, D, \pi) = Z = \sum_{j=1}^{n} \left(\alpha E_j + \beta T_j + \gamma d_1 + \delta D\right). \tag{2.1}$$

We focus on the situation that there are two rate-modifying activities ($l = 1, 2$) first, and then extend it to multiple rate-modifying activities ($l \geq 3$).

Following the three-field notation of Graham et al. [34], we denote our problems as $1|\text{DJLE}, 2\text{RM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ with two rate-modifying activities (2RM) and $1|\text{DJLE}, \text{MRM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ with multiple rate-modifying activities ($l \geq 3$) (MRM), where DJLE means "deteriorating jobs and learning effect".

3. Preliminary Works

In this section, some useful preliminary works are given. For the situation there are two rate-modifying activities, we assume the position of the lth rate-modifying activity on machine is i_l, $l = 1, 2$ and they satisfy $1 \leq i_1 \leq i_2 \leq n$. So for any schedule (see Figure 1), we have the following actual processing times for jobs $J_{[1]}, J_{[2]}, \ldots, J_{[n]}$ which are discussed in three parts: $(J_{[1]}, J_{[2]}, \ldots, J_{[i_1]})$, $(J_{[i_1+1]}, J_{[i_1+2]}, \ldots, J_{[i_2]})$, and $(J_{[i_2+1]}, J_{[i_2+2]}, \ldots, J_{[n]})$.

Note that the starting time of $J_{[1]}$, $s_{[1]}$ is equal to 0, the stating time of $J_{[2]}$, $s_{[2]}$ is equal to the completion time of $J_{[1]}$ which is just $p^A_{[1]}$, and the stating time of $J_{[3]}$, $s_{[3]}$ is equal to the completion time of $J_{[2]}$ which is just $p^A_{[1]} + p^A_{[2]}$,

$$\begin{aligned}
p^A_{[1]} &= p_{[1]}(1)^{a_{[1]}}, \\
p^A_{[2]} &= p_{[2]}(2)^{a_{[2]}} + bs_{[2]} \\
&= p_{[2]}(2)^{a_{[2]}} + bC_{[1]} \\
&= p_{[2]}(2)^{a_{[2]}} + bp_{[1]}(1)^{a_{[1]}}, \\
p^A_{[3]} &= p_{[3]}(3)^{a_{[3]}} + bs_{[3]} \\
&= p_{[3]}(3)^{a_{[3]}} + bC_{[2]} \\
&= p_{[3]}(3)^{a_{[3]}} + b\left(p^A_{[1]} + p^A_{[2]}\right) \\
&= p_{[3]}(3)^{a_{[3]}} + bp_{[2]}(2)^{a_{[2]}} + b(1+b)p_{[1]}(1)^{a_{[1]}},
\end{aligned}$$

$$\begin{aligned} p^A_{[4]} &= p_{[4]}(4)^{a_{[4]}} + bp_{[3]}(3)^{a_{[3]}} + b(1+b)p_{[2]}(2)^{a_{[2]}} + b(1+b)^2 p_{[1]}(1)^{a_{[1]}}, \\ &\vdots \end{aligned} \tag{3.1}$$

From above analysis, we obtain the following general expression of actual processing time for jobs $J_{[1]}, J_{[2]}, \ldots, J_{[i_1]}$:

$$p^A_{[j]} = p_{[j]}(j)^{a_{[j]}} + b\sum_{h=1}^{j-1}(1+b)^{j-1-h}p_{[h]}(h)^{a_{[h]}}, \quad j = 1, \ldots, i_1. \tag{3.2}$$

For further analysis of actual processing times of other jobs, we provide the actual processing time of job $J_{[i_1]}$ as

$$p^A_{[i_1]} = p_{[i_1]}(i_1)^{a_{[i_1]}} + b\sum_{h=1}^{i_1-1}(1+b)^{i_1-1-h}p_{[h]}(h)^{a_{[h]}}. \tag{3.3}$$

Note that $bs_{[i_1]} = p^A_{[i_1]} - p_{[i_1]}(i_1)^{a_{[i_1]}}$, $bs_{[i_1+1]} = p^A_{[i_1]+1} - \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}}$ according to the expressions of actual processing times of jobs in Section 2, and the starting time of job $J_{[i_1+1]}$, $s_{[i_1+1]}$ is equal to the sum of the completion time of job $J_{[i_1]}$ and the duration of the first rate-modifying activity, which is $C_{[i_1]} + t_1$. Moreover $C_{[i_1]} = p^A_{[i_1]} + s_{[i_1]}$, so

$$\begin{aligned} p^A_{[i_1+1]} &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bs_{[i_1+1]} \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + b\left(C_{[i_1]} + t_1\right) \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + b\left(t_1 + p^A_{[i_1]} + s_{[i_1]}\right) \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bt_1 + bp^A_{[i_1]} + bs_{[i_1]} \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bt_1 + bp^A_{[i_1]} + p^A_{[i_1]} - p_{[i_1]}(i_1)^{a_{[i_1]}} \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bt_1 + (b+1)p^A_{[i_1]} - p_{[i_1]}(i_1)^{a_{[i_1]}} \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bt_1 + bp_{[i_1]}(i_1)^{a_{[i_1]}} + b(b+1)\sum_{h=1}^{i_1-1}(1+b)^{i_1-1-h}p_{[h]}(h)^{a_{[h]}} \\ &= \theta_{[i_1+1]1}p_{[i_1+1]}(i_1+1)^{a_{[i_1+1]}} + bp_{[i_1]}(i_1)^{a_{[i_1]}} + b\sum_{h=1}^{i_1-1}(1+b)^{i_1-h}p_{[h]}(h)^{a_{[h]}} + bt_1, \end{aligned}$$

$$\begin{aligned}
p^A_{[i_1+2]} &= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b s_{[i_1+2]} \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b C_{[i_1+1]} \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b\Big(p^A_{[i_1+1]} + s_{[i_1+1]}\Big) \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b p^A_{[i_1+1]} + b s_{[i_1+1]} \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b p^A_{[i_1+1]} + p^A_{[i_1+1]} - \theta_{[i_1+1]1} p_{[i_1+1]} (i_1+1)^{a_{[i_1+1]}} \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + (b+1) p^A_{[i_1+1]} - \theta_{[i_1+1]1} p_{[i_1+1]} (i_1+1)^{a_{[i_1+1]}} \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b\theta_{[i_1+1]1} p_{[i_1+1]} (i_1+1)^{a_{[i_1+1]}} + b(b+1) p_{[i_1]} (i_1)^{a_{[i_1]}} \\
&\quad + b(b+1)\sum_{h=1}^{i_1-1} (1+b)^{i_1-h} p_{[h]} (h)^{a_{[h]}} + b(b+1)t_1 \\
&= \theta_{[i_1+2]1} p_{[i_1+2]} (i_1+2)^{a_{[i_1+2]}} + b\theta_{[i_1+1]1} p_{[i_1+1]} (i_1+1)^{a_{[i_1+1]}} + b(b+1) p_{[i_1]} (i_1)^{a_{[i_1]}} \\
&\quad + b\sum_{h=1}^{i_1-1} (1+b)^{i_1+1-h} p_{[h]} (h)^{a_{[h]}} + b(b+1)t_1, \\
&\ \ \vdots
\end{aligned} \tag{3.4}$$

We obtain the following general expression of actual processing time for jobs $J_{[i_1+1]}, J_{[i_2+2]}, \ldots, J_{[i_2]}$:

$$\begin{aligned}
p^A_{[j]} &= \theta_{[j]1} p_{[j]} (j)^{a_{[j]}} + b\sum_{h=1}^{i_1} (1+b)^{j-1-h} p_{[h]} (h)^{a_{[h]}} \\
&\quad + b\sum_{h=i_1+1}^{j-1} (1+b)^{j-1-h} \theta_{[h]1} p_{[h]} (h)^{a_{[h]}} + b(1+b)^{j-i_1-1} t_1, \quad j = i_1+1, \ldots, i_2, \\
p^A_{[i_2]} &= \theta_{[i_2]1} p_{[i_2]} (i_2)^{a_{[i_2]}} + b\sum_{h=1}^{i_1} (1+b)^{i_2-1-h} p_{[h]} (h)^{a_{[h]}} \\
&\quad + b\sum_{h=i_1+1}^{i_2-1} (1+b)^{i_2-1-h} \theta_{[h]1} p_{[h]} (h)^{a_{[h]}} + b(1+b)^{i_2-i_1-1} t_1, \\
p^A_{[i_2+1]} &= \theta_{[i_2+1]2} p_{[i_2+1]} (i_2+1)^{a_{[i_2+1]}} + b t_2 + b\theta_{[i_2]1} p_{[i_2]} (i_2)^{a_{[i_2]}} \\
&\quad + b\sum_{h=1}^{i_1} (1+b)^{i_2-h} p_{[h]} (h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2-1} (1+b)^{i_2-h} \theta_{[h]1} p_{[h]} (h)^{a_{[h]}} + b(1+b)^{i_2-i_1} t_1, \\
&\ \ \vdots
\end{aligned} \tag{3.5}$$

Similarly, the general expression of actual processing time for jobs $J_{[i_2+1]}, J_{[i_2+2]}, \ldots, J_{[n]}$ is as follows:

$$\begin{aligned} p^A_{[j]} &= \theta_{[j]2} p_{[j]}(j)^{a_{[j]}} + b\sum_{h=1}^{i_1}(1+b)^{j-1-h} p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2}(1+b)^{j-1-h}\theta_{[h]1} p_{[h]}(h)^{a_{[h]}} \\ &\quad + b\sum_{h=i_2+1}^{j-1}(1+b)^{j-1-h}\theta_{[h]2} p_{[h]}(h)^{a_{[h]}} + b(1+b)^{j-i_1-1}t_1 + b(1+b)^{j-i_2-1}t_2, \quad j = i_2+1,\ldots,n. \end{aligned} \tag{3.6}$$

We present that some properties of an optimal solution for the common due-window assignment problem proved by Mosheiov and Sarig [30] still hold for the problem discussed in this paper.

Lemma 3.1. *An optimal schedule exists in which the due-window starts and finishes at certain job completion times.*

Proof. For any given job sequence ($\pi = J_{[1]}, J_{[2]}, \ldots, J_{[k]}, \ldots, J_{[k+m]}, \ldots, J_{[n]}$), we set $C_{[k]} < d_1 < C_{[k+1]}$ and $C_{[k+m]} < d_2 < C_{[k+m+1]}$, where kth and $(k+m)$th are positions in sequence π and satisfy $0 \le k \le (k+m) \le n$ (see Figure 2). Considering the relative location of due-window and two rate-modifying activities, there are six cases altogether, that is, $i_1 \le i_2 \le k$, $k \le i_1 \le i_2 \le k+m$, $k+m \le i_1 \le i_2$, $i_1 \le k \&\& k+m \le i_2$, $i_1 \le k \&\& k \le i_2 \le k+m$, and $k \le i_1 \le k+m \&\& k+m \le i_2$. For simplification of description, we only investigate the case $i_1 \le i_2 \le k$ in this part, and the proofs of other cases are similar. In addition, we set $\varphi_1 = d_1 - C_{[k]}$, $\varphi_2 = d_2 - C_{[k+m]}$ and clearly $0 \le \varphi_1 \le p^A_{[k+1]}$ and $0 \le \varphi_2 \le p^A_{[k+m+1]}$.

As described in (2.1), the total cost function includes four parts: the earliness cost, the tardiness cost, the due-window starting time cost, and the due-window size cost.

For job j of a schedule π, we denote the earliness cost by Z^E_j, where $j = k, k-1, \ldots, 1$.

$$\begin{aligned} Z^E_k &= \alpha\varphi_1, \\ Z^E_{k-1} &= \alpha\left[\varphi_1 + p^A_{[k]}\right], \\ Z^E_{k-2} &= \alpha\left[\varphi_1 + p^A_{[k]} + p^A_{[k-1]}\right], \\ &\vdots \\ Z^E_{i_2+1} &= \alpha\left[\varphi_1 + p^A_{[k]} + p^A_{[k-1]} + \cdots p^A_{[i_2+2]}\right], \\ Z^E_{i_2} &= \alpha\left[\varphi_1 + p^A_{[k]} + p^A_{[k-1]} + \cdots p^A_{[i_2+1]} + t_2\right], \\ Z^E_{i_2-1} &= \alpha\left[\varphi_1 + p^A_{[k]} + p^A_{[k-1]} + \cdots p^A_{[i_2+1]} + t_2 + p^A_{[i_2]}\right], \\ Z^E_{i_2-2} &= \alpha\left[\varphi_1 + p^A_{[k]} + p^A_{[k-1]} + \cdots p^A_{[i_2+1]} + t_2 + p^A_{[i_2]} + p^A_{[i_2-1]}\right], \\ &\vdots \end{aligned}$$

$$\begin{aligned}
Z_{i_1+1}^E &= \alpha\Big[\varphi_1 + p_{[k]}^A + p_{[k-1]}^A + \cdots p_{[i_2+1]}^A + t_2 + p_{[i_2]}^A + p_{[i_2-1]}^A + \cdots + p_{[i_1+2]}^A\Big],\\
Z_{i_1}^E &= \alpha\Big[\varphi_1 + p_{[k]}^A + p_{[k-1]}^A + \cdots p_{[i_2+1]}^A + t_2 + p_{[i_2]}^A + p_{[i_2-1]}^A + \cdots + p_{[i_1+1]}^A + t_1\Big],\\
Z_{i_1-1}^E &= \alpha\Big[\varphi_1 + p_{[k]}^A + p_{[k-1]}^A + \cdots p_{[i_2+1]}^A + t_2 + p_{[i_2]}^A + p_{[i_2-1]}^A + \cdots + p_{[i_1+1]}^A + t_1 + p_{[i_1]}^A\Big],\\
&\ \ \vdots,\\
Z_1^E &= \alpha\Big[\varphi_1 + p_{[k]}^A + p_{[k-1]}^A + \cdots p_{[i_2+1]}^A + t_2 + p_{[i_2]}^A + p_{[i_2-1]}^A + \cdots + p_{[i_1+1]}^A + t_1 + p_{[i_1]}^A + \cdots + p_{[2]}^A\Big].
\end{aligned} \tag{3.7}$$

For job j of a schedule π, we denote the tardiness cost by Z_j^T, where $j = k+m+1, k+m+2, \ldots, n$,

$$\begin{aligned}
Z_{k+m+1}^T &= \beta\Big[p_{[k+m+1]}^A - \varphi_2\Big],\\
Z_{k+m+2}^T &= \beta\Big[p_{[k+m+1]}^A + p_{[k+m+2]}^A - \varphi_2\Big],\\
&\ \ \vdots\\
Z_n^T &= \beta\Big[p_{[k+m+1]}^A + p_{[k+m+2]}^A + \cdots + p_{[n]}^A - \varphi_2\Big].
\end{aligned} \tag{3.8}$$

The due-window starting time cost denoted by Z_{d_1} can be expressed as

$$Z_{d_1} = n\gamma\Big[\varphi_1 + p_{[k]}^A + p_{[k-1]}^A + \cdots p_{[i_2+1]}^A + t_2 + p_{[i_2]}^A + p_{[i_2-1]}^A + \cdots + p_{[i_1+1]}^A + t_1 + p_{[i_1]}^A + \cdots + p_{[2]}^A + p_{[1]}^A\Big]. \tag{3.9}$$

The due-window size cost denoted by Z_D can be expressed as

$$Z_D = n\delta\Big[p_{[k+1]}^A + \cdots + p_{[k+m]}^A + \varphi_2 - \varphi_1\Big]. \tag{3.10}$$

For simplifying the total cost function, let

$$w_j = \begin{cases}
\alpha(j-1) + \gamma n, & j = 1, \ldots, i_1,\\
\alpha(j-1) + \gamma n, & j = i_1 + 1, \ldots, i_2,\\
\alpha(j-1) + \gamma n, & j = i_2 + 1, \ldots, k,\\
\delta n, & j = k+1, \ldots, k+m,\\
\beta(n-j+1), & j = k+m+1, \ldots, n.
\end{cases} \tag{3.11}$$

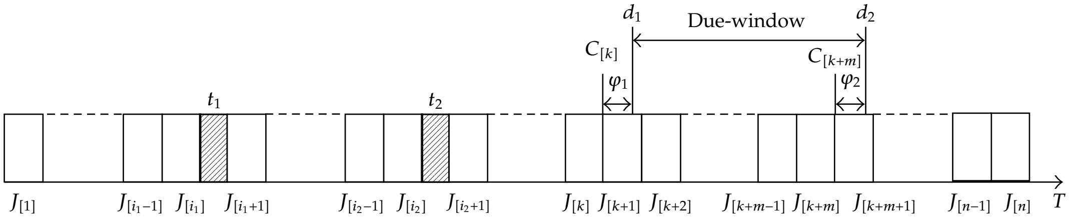

Figure 2: Structure of a schedule considering due-window.

The total cost can be represented as

$$Z = [n\gamma + \alpha k - n\delta]\varphi_1 + [\delta n + \beta(k+m) - \beta n]\varphi_2 + G, \tag{3.12}$$

where $G = [b\sum_{j=i_1+1}^{n} w_j(1+b)^{j-i_1-1} + n\gamma + \alpha i_1]t_1 + [b\sum_{j=i_2+1}^{n} w_j(1+b)^{j-i_2-1} + n\gamma + \alpha i_2]t_2 + \sum_{j=k+m+1}^{n}[\beta(n-j+1) + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}(\beta(n-h+1))]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} + \sum_{j=k+1}^{k+m}[n\delta + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}n\delta]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} + \sum_{j=i_2+1}^{k}[n\gamma + \alpha(j-1) + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}(\alpha(h-1)+\gamma n)]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} + \sum_{j=i_1+1}^{i_2}[n\gamma + \alpha(j-1) + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}(\alpha(h-1)+\gamma n)]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}} + \sum_{j=1}^{i_1}[n\gamma + \alpha(j-1) + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}(\alpha(h-1)+\gamma n)]p_{[j]}(j)^{a_{[j]}}$.

From (3.12), we know that the total cost includes three items: $[n\gamma + \alpha k - n\delta]\varphi_1$, $[\delta n + \beta(k+m) - \beta n]\varphi_2$, and G. It is easy to find that $G > 0$ based on the expression of G. So the minimization of the total cost depends on the values of $[n\gamma + \alpha k - n\delta]\varphi_1$ and $[\delta n + \beta(k+m) - \beta n]\varphi_2$. Because of φ_1 and φ_2 are independent of the coefficients. So we discuss the minimization problem in the following four different cases.

(1) If $[n\gamma + \alpha k - n\delta] \geq 0$ and $[\delta n + \beta(k+m) - \beta n] \geq 0$, then $\varphi_1 = 0$ and $\varphi_2 = 0$.

(2) If $[n\gamma + \alpha k - n\delta] \leq 0$ and $[\delta n + \beta(k+m) - \beta n] \leq 0$, then $\varphi_1 = p_{[k+1]}^{A} = \theta_{[k+1]2}p_{[k+1]}(k+1)^{a_{[k+1]}} + b\sum_{h=1}^{i_1}(1+b)^{k-h}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2}(1+b)^{k-h}\theta_{[h]1}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_2+1}^{k}(1+b)^{k-h}\theta_{[h]2}p_{[h]}(h)^{a_{[h]}} + b(1+b)^{k-i_1}t_1 + b(1+b)^{k-i_2}t_2$, and $\varphi_2 = p_{[k+m+1]}^{A} = \theta_{[k+m+1]2}p_{[k+m+1]}(k+m+1)^{a_{[k+m+1]}} + b\sum_{h=1}^{i_1}(1+b)^{k+m-h}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2}(1+b)^{k+m-h}\theta_{[h]1}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_2+1}^{k+m}(1+b)^{k+m-h}\theta_{[h]2}p_{[h]}(h)^{a_{[h]}} + b(1+b)^{k+m-i_1}t_1 + b(1+b)^{k+m-i_2}t_2$.

(3) If $[n\gamma + \alpha k - n\delta] \geq 0$ and $[\delta n + \beta(k+m) - \beta n] \leq 0$, then $\varphi_1 = 0$ and $\varphi_2 = p_{[k+m+1]}^{A} = \theta_{[k+m+1]2}p_{[k+m+1]}(k+m+1)^{a_{[k+m+1]}} + b\sum_{h=1}^{i_1}(1+b)^{k+m-h}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2}(1+b)^{k+m-h}\theta_{[h]1}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_2+1}^{k+m}(1+b)^{k+m-h}\theta_{[h]2}p_{[h]}(h)^{a_{[h]}} + b(1+b)^{k+m-i_1}t_1 + b(1+b)^{k+m-i_2}t_2$.

(4) If $[n\gamma + \alpha k - n\delta] \leq 0$ and $[\delta n + \beta(k+m) - \beta n] \geq 0$, then $\varphi_1 = p_{[k+1]}^{A} = \theta_{[k+1]2}p_{[k+1]}(k+1)^{a_{[k+1]}} + b\sum_{h=1}^{i_1}(1+b)^{k-h}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_1+1}^{i_2}(1+b)^{k-h}\theta_{[h]1}p_{[h]}(h)^{a_{[h]}} + b\sum_{h=i_2+1}^{k}(1+b)^{k-h}\theta_{[h]2}p_{[h]}(h)^{a_{[h]}} + b(1+b)^{k-i_1}t_1 + b(1+b)^{k-i_2}t_2$ and $\varphi_2 = 0$.

So, from the analysis, we say that an optimal schedule exists in which the due window starts and finishes at certain job completion times. □

By Lemma 3.1, the the due-window starting time d_1 and finishing time d_2 are denoted with k and $k+m$ as the indices of the jobs completed at them, respectively, that is, $C_{[k]} = d_1$

and $C_{[k+m]} = d_2$. Moreover, we also provide another property of an optimal solution for the scheduling problem with learning effect and multiple rate-modifying activities.

Lemma 3.2. *For the problem* $1|DJLE, 2RM|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$*, there exists an optimal schedule in which* $d_1 = C_{[k]}$ *and* $d_2 = C_{[k+m]}$*, where* $k = \lceil n(\delta - \gamma)/\alpha \rceil$ *and* $(k + m) = \lceil n(\beta - \delta)/\beta \rceil$.

Proof. The proof is similar to that of Mosheiov and Sarig [30]. □

Lemma 3.3. *For the problem* $1|DJLE, MRM|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$*, there also exists an optimal schedule in which* $d_1 = C_{[k]}$ *and* $d_2 = C_{[k+m]}$*, where* $k = \lceil n(\delta - \gamma)/\alpha \rceil$ *and* $(k + m) = \lceil n(\beta - \delta)/\beta \rceil$.

Proof. The proof is similar to Lemma 3.2. □

4. An Optimal Solution Policy

In this section, we show that problems $1|\mathrm{DJLE}, 2\mathrm{RM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ and $1|\mathrm{DJLE}, \mathrm{MRM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ can be both solved in polynomial times.

Theorem 4.1. *The* $1|DJLE, 2RM|\sum_{j}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ *problem can be solved in* $O(n^{2+3})$ *time.*

Proof. For two rate-modifying activities, there are six cases altogether, that is, $i_1 \le i_2 \le k$, $k \le i_1 \le i_2 \le k + m$, $k + m \le i_1 \le i_2$, $i_1 \le k \&\& k + m \le i_2$, $i_1 \le k \&\& k \le i_2 \le k + m$, and $k \le i_1 \le k + m \&\& k + m \le i_2$.

Case 1 $(i_1, i_2 < k)$. If two different rate-modifying activities are performed before the due window, from the preliminary works, the total cost can be given by

$$\begin{aligned}
Z = {} & \left[b \sum_{j=i_1+1}^{n} w_j (1+b)^{j-i_1-1} + n\gamma + \alpha i_1 \right] t_1 + \left[b \sum_{j=i_2+1}^{n} w_j (1+b)^{j-i_2-1} + n\gamma + \alpha i_2 \right] t_2 \\
& + \sum_{j=k+m+1}^{n} \left[\beta(n-j+1) + b \sum_{h=j+1}^{n} (1+b)^{h-j-1} (\beta(n-h+1)) \right] \theta_{[j]2} p_{[j]} (j)^{a_{[j]}} \\
& + \sum_{j=k+1}^{k+m} \left[n\delta + b \sum_{h=j+1}^{n} (1+b)^{h-j-1} n\delta \right] \theta_{[j]2} p_{[j]} (j)^{a_{[j]}} \\
& + \sum_{j=i_2+1}^{k} \left[n\gamma + \alpha(j-1) + b \sum_{h=j+1}^{n} (1+b)^{h-j-1} (\alpha(h-1) + \gamma n) \right] \theta_{[j]2} p_{[j]} (j)^{a_{[j]}} \\
& + \sum_{j=i_1+1}^{i_2} \left[n\gamma + \alpha(j-1) + b \sum_{h=j+1}^{n} (1+b)^{h-j-1} (\alpha(h-1) + \gamma n) \right] \theta_{[j]1} p_{[j]} (j)^{a_{[j]}} \\
& + \sum_{j=1}^{i_1} \left[n\gamma + \alpha(j-1) + b \sum_{h=j+1}^{n} (1+b)^{h-j-1} (\alpha(h-1) + \gamma n) \right] p_{[j]} (j)^{a_{[j]}}.
\end{aligned} \tag{4.1}$$

By substituting (3.11) into above total cost function again, we have

$$\begin{aligned}
Z = {} & \sum_{j=1}^{i_1}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}} + \sum_{j=i_1+1}^{i_2}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}} \\
& + \sum_{j=i_2+1}^{n}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} + \left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1} + n\gamma + \alpha i_1\right]t_1 \\
& + \left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1} + n\gamma + \alpha i_2\right]t_2.
\end{aligned} \tag{4.2}$$

Case 2 ($k \leq i_1,\ i_2 \leq (k+m)$). if two different rate-modifying activities are performed in the due window, similar to the analysis of Case 1, the total cost can be given by

$$\begin{aligned}
Z &= \sum_{j=1}^{n}\left(\alpha E_j + \beta T_j + \gamma d_1 + \delta D\right) \\
&= \sum_{j=1}^{i_1}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}} \\
&\quad + \sum_{j=i_1+1}^{i_2}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}} \\
&\quad + \sum_{j=i_2+1}^{n}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} \\
&\quad + \left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1} + n\delta\right]t_1 + \left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1} + n\delta\right]t_2.
\end{aligned} \tag{4.3}$$

Case 3 ($(k+m) \leq i_1, i_2$). if two different rate-modifying activities are performed after the due window, similar to the analysis of Case 1, the total cost can be given by

$$\begin{aligned}
Z &= \sum_{j=1}^{n}\left(\alpha E_j + \beta T_j + \gamma d_1 + \delta D\right) \\
&= \sum_{j=1}^{i_1}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}}
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{j=i_1+1}^{i_2}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}}\\
&+\sum_{j=i_2+1}^{n}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}}\\
&+\left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1}+\beta(n-i_1)\right]t_1+\left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1}+\beta(n-i_2)\right]t_2.
\end{aligned}
\tag{4.4}
$$

Case 4 $((i_1 \le k)\&\&(k+m \le i_2))$. If one rate-modifying activities is performed before the due-window and the other is after the due-window, similar to the analysis of Case 1, the total cost can be given by

$$
\begin{aligned}
Z&=\sum_{j=1}^{n}\left(\alpha E_j+\beta T_j+\gamma d_1+\delta D\right)\\
&=\sum_{j=1}^{i_1}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}}\\
&+\sum_{j=i_1+1}^{i_2}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}}\\
&+\sum_{j=i_2+1}^{n}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}}\\
&+\left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1}+n\gamma+\alpha i_1\right]t_1+\left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1}+\beta(n-i_2)\right]t_2.
\end{aligned}
\tag{4.5}
$$

Case 5 $((i_1 \le k)\&\&(k \le i_2 \le k+m))$. if one rate-modifying activities is performed before the due window and the other is in the due window, similar to the analysis of Case 1, the total cost can be given by

$$
\begin{aligned}
Z&=\sum_{j=1}^{n}\left(\alpha E_j+\beta T_j+\gamma d_1+\delta D\right)\\
&=\sum_{j=1}^{i_1}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}}
\end{aligned}
$$

$$\begin{aligned}
&+\sum_{j=i_1+1}^{i_2}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}}\\
&+\sum_{j=i_2+1}^{n}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}}\\
&+\left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1}+n\gamma+\alpha i_1\right]t_1+\left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1}+n\delta\right]t_2.
\end{aligned}\tag{4.6}$$

Case 6 $((k \le i_1 \le k+m)\&\&(k+m \le i_2))$. if one rate-modifying activities is performed in the due window and the other is after the due window, similar to the analysis of Case 1, the total cost can be given by

$$\begin{aligned}
Z&=\sum_{j=1}^{n}\left(\alpha E_j+\beta T_j+\gamma d_1+\delta D\right)\\
&=\sum_{j=1}^{i_1}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}}\\
&\quad+\sum_{j=i_1+1}^{i_2}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}}\\
&\quad+\sum_{j=i_2+1}^{n}\left[w_j+b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}}\\
&\quad+\left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1}+n\delta\right]t_1+\left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1}+n\gamma+\alpha i_2\right]t_2.
\end{aligned}\tag{4.7}$$

In the following discussion, for simplification we still take only one case into consideration, that is, Case 3. we define variables x_{jr}, for $j = 1,2,\ldots,n$, $r = 1,2,\ldots,n$. $x_{jr} = 1$, if job j is scheduled in position r, $= 0$, otherwise.

Let

$$B_{jr}=\begin{cases}\left(w_r+b\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)p_{[j]}(r)^{a_{[j]}}, & r=1,\ldots,i_1,\\ \left(w_r+b\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)\theta_{[j]1}p_{[j]}(r)^{a_{[j]}}, & r=i_1+1,\ldots,i_2,\\ \left(w_r+b\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)\theta_{[j]2}p_{[j]}(r)^{a_{[j]}}, & r=i_2+1,\ldots,n.\end{cases}\tag{4.8}$$

The problem can be formulated as follows:

$$\begin{aligned}
\text{Min}\quad & \sum_{j=1}^{n}\sum_{r=1}^{n} B_{jr}x_{jr}+\left[b\sum_{j=i_1+1}^{n} w_j(1+b)^{j-i_1-1}+\beta(n-i_1)\right]t_1 \\
& +\left[b\sum_{j=i_2+1}^{n} w_j(1+b)^{j-i_2-1}+\beta(n-i_2)\right]t_2 \\
\text{subject to}\quad & \sum_{r=1}^{n} x_{jr}=1,\quad j=1,2,\ldots,n, \\
& \sum_{j=1}^{n} x_{jr}=1,\quad r=1,2,\ldots,n, \\
& x_{jr}=1 \text{ or } 0,\quad j=1,2,\ldots,n,\ r=1,2,\ldots,n.
\end{aligned} \tag{4.9}$$

The first set of constraints guarantees each job j is scheduled only once, the second set of constraints guarantees each position r is taken by only one job, and the third constraints means the variable x_{jr} is binary. For given positions i_1 and i_2, the problem is transferred to the following assignment problem

$$\begin{aligned}
\text{Min}\quad & \sum_{j=1}^{n}\sum_{r=1}^{n} B_{jr}x_{jr} \\
\text{subject to}\quad & \sum_{r=1}^{n} x_{jr}=1,\quad j=1,2,\ldots,n, \\
& \sum_{j=1}^{n} x_{jr}=1,\quad r=1,2,\ldots,n, \\
& x_{jr}=1 \text{ or } 0,\quad j=1,2,\ldots,n,\ r=1,2,\ldots,n.
\end{aligned} \tag{AP}$$

The assignment problem can be solved in $O(n^3)$ time (see, e.g., Papadimitriou and Steiglitz [35] and Brucker [36]). However, i_1 and i_2 may be any value of $1\cdots n$ for all cases, so the complexity of Case 2 is $O(n^{3+2})=O(n^5)$ and Theorem 4.1 holds. □

Theorem 4.2. *The* $1|JDLE, MRM|\sum_j^n(\alpha E_j+\beta T_j+\gamma d_1+\delta D)$ *problem can be solved in* $O(n^{3+u})$ *time.*

Proof. When there exist u different rate-modifying activities, we take the case $(k+m)\le i_1 < i_2 < i_3\cdots < i_u$ as an example, and the proofs of other cases are similar. We can formulate the problem as follows:

$$
\begin{aligned}
Z &= \sum_{j}^{n}\left(\alpha E_j + \beta T_j + \gamma d_1 + \delta D\right) \\
&= \sum_{j=1}^{i_1}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]p_{[j]}(j)^{a_{[j]}} \\
&\quad + \sum_{j=i_1+1}^{i_2}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]1}p_{[j]}(j)^{a_{[j]}} \\
&\quad + \sum_{j=i_2+1}^{n}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]2}p_{[j]}(j)^{a_{[j]}} \\
&\quad + \cdots + \sum_{j=i_u+1}^{n}\left[w_j + b\sum_{h=j+1}^{n}(1+b)^{h-j-1}w_h\right]\theta_{[j]u}p_{[j]}(j)^{a_{[j]}} \\
&\quad + \left[b\sum_{j=i_1+1}^{n}w_j(1+b)^{j-i_1-1} + \beta(n-i_1)\right]t_1 + \left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_2-1} + \beta(n-i_2)\right]t_2 \\
&\quad + \cdots + \left[b\sum_{j=i_2+1}^{n}w_j(1+b)^{j-i_u-1} + \beta(n-i_u)\right]t_u.
\end{aligned}
\tag{4.10}
$$

Let

$$
Q_{jr} = \begin{cases}
\left(w_r + b\displaystyle\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)p_{[j]}(r)^{a_{[j]}}, & r = 1,\ldots,i_1, \\
\left(w_r + b\displaystyle\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)\theta_{[j]1}p_{[j]}(r)^{a_{[j]}}, & r = i_1+1,\ldots,i_2, \\
\left(w_r + b\displaystyle\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)\theta_{[j]2}p_{[j]}(r)^{a_{[j]}}, & r = i_2+1,\ldots,i_3, \\
\vdots & \\
\left(w_r + b\displaystyle\sum_{h=r+1}^{n}(1+b)^{h-r-1}w_h\right)\theta_{[j]u}p_{[j]}(r)^{a_{[j]}}, & r = i_u+1,\ldots,n.
\end{cases}
\tag{4.11}
$$

The problem can be transformed to the following form after introducing variable x_{jr} defined as above:

$$\begin{aligned}
\text{Min} \quad & \sum_{j=1}^{n}\sum_{r=1}^{n} Q_{jr}x_{jr} + \sum_{l=1}^{u}\left[b\sum_{j=i_1+1}^{n} w_j(1+b)^{j-i_1-1} + \beta(n-i_u)\right] t_l \\
\text{subject to} \quad & \sum_{r=1}^{n} x_{jr} = 1, \quad j = 1,2,\ldots,n, \\
& \sum_{j=1}^{n} x_{jr} = 1, \quad r = 1,2,\ldots,n, \\
& x_{jr} = 1 \text{ or } 0, \quad j = 1,2,\ldots,n, \ r = 1,2,\ldots,n.
\end{aligned} \tag{4.12}$$

For given positions $i_1, i_2, \ldots, i_u$, the last item in the objective function is constant. So the above minimization is equivalent to minimizing the following assignment problem:

$$\begin{aligned}
\text{Min} \quad & \sum_{j=1}^{n}\sum_{r=1}^{n} Q_{jr}x_{jr} \\
\text{subject to} \quad & \sum_{r=1}^{n} x_{jr} = 1, \quad j = 1,2,\ldots,n, \\
& \sum_{j=1}^{n} x_{jr} = 1, \quad r = 1,2,\ldots,n, \\
& x_{jr} = 1 \text{ or } 0, \quad j = 1,2,\ldots,n, \ r = 1,2,\ldots,n.
\end{aligned} \tag{BP}$$

Since $i_1, i_2, \ldots, i_u$ may be any value of $1 \cdots n$, the number of $(1, \ldots, n)$ vectors is bounded by $(n+1)^u$. The complexity of the problem is $O(n^{3+u})$ for all cases and Theorem 4.2 holds. □

The polynomial time algorithm to solve $1|\text{DJLE}, 2\text{RM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ problem optimally is as follows.

Algorithm 1. We have the following steps.

Step 1. Assign the optimal due-window starting time d_1^* and finishing time d_2^* at the completion time of the kth and $(k+m)$th job specifically, where $k = \lceil n(\delta - \gamma)/\alpha \rceil$, $(k+m) = \lceil n(\beta - \delta)/\beta \rceil$.

Step 2. For $(i_2 = 1, i_u \leq n, i_u + +)$ and for $(i_1 = 1, i_1 \leq n, i_1 + +)$.
Calculate the weight B_{jr} with (4.8).
Solve the classical assignment problem (AP) and get the total cost.

Step 3. Obtain the optimal schedule with minimum total cost.

The polynomial time algorithm to solve $1|\text{DJLE}, \text{MRM}|\sum_{j=1}^{n}(\alpha E_j + \beta T_j + \gamma d_1 + \delta D)$ problem optimally is as follows.

Algorithm 2. We have the following steps.

Step 1. Assign the optimal due-window starting time d_1^* and finishing time d_2^* at the completion time of the kth and $(k+m)$th job specifically, where $k = \lceil n(\delta-\gamma)/\alpha \rceil$, $(k+m) = \lceil n(\beta-\delta)/\beta \rceil$.

Step 2. For $(i_u = 1, i_u \leq n, i_u{+}{+})$, for $(i_{u-1} = 1, i_{u-1} \leq n, i_{u-1}{+}{+}), \ldots$, and for $(i_1 = 1, i_1 \leq n, i_1{+}{+})$.
Calculate the weight Q_{jr}, for $j = 1, 2, \ldots, n$; $r = 1, 2, \ldots, n$.
Solve the classical assignment problem (BP) and get the total cost.

Step 3. Obtain the optimal schedule with minimum total cost.

From the above description, it is easy to conclude that Algorithms 1 and 2 take $O(n^5)$ time and $O(n^{3+u})$ time, respectively.

5. Conclusions

In this paper, we consider a single machine scheduling problem with due-window assignment and multiple rate-modifying activities in the settings of learning effect and deteriorating jobs. We introduce an $O(n^{3+u})$ solution algorithm for u different types of rate-modifying activities considering the objective to find jointly the optimal location to perform multiple rate-modifying activities, the optimal job sequence, and the optimal location and size of the due window to minimize the total earliness, tardiness, and due-window-related costs. Further research may investigate problems with multimachine settings and deteriorating rate-modifying activities.

Acknowledgment

This work is partially supported by NSF of China under Grants (70701029) and State-Sponsored Study Abroad Program of China.

References

[1] T. P. Wright, “Factors affecting the cost of airplanes,” *Journal of Aeronautical Sciences*, vol. 3, pp. 122–128, 1936.

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

[3] T. C. E. Cheng and G. Wang, “Single machine scheduling with learning effect considerations,” *Annals of Operations Research*, vol. 98, pp. 273–290, 2000.

[4] 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.

[5] C. Koulamas and G. J. Kyparisis, “Single-machine and two-machine flowshop scheduling with general learning functions,” *European Journal of Operational Research*, vol. 178, no. 2, pp. 402–407, 2007.

[6] J.-B. Wang, L. Sun, and L. Sun, “Single machine scheduling with a learning effect and discounted costs,” *International Journal of Advanced Manufacturing Technology*, vol. 49, no. 9–12, pp. 1141–1149, 2010.

[7] W. H. Kuo and D. L. Yang, “Single-machine group scheduling with a time-dependent learning effect,” *Computers and Operations Research*, vol. 33, no. 8, pp. 2099–2112, 2006.

[8] C.-L. Lee and V.-J. Leon, “Machine scheduling with a ratemodifying activity,” *European Journal of Operational Research*, vol. 128, pp. 129–128, 2001.

[9] C.-L. Zhao, H.-Y. Tang, and C.-D. Cheng, “Two-parallel machines scheduling with rate-modifying activities to minimize total completion time,” *European Journal of Operational Research*, vol. 198, no. 1, pp. 354–357, 2009.

[10] E. J. Lodree, Jr. and C. D. Geiger, “A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration,” *European Journal of Operational Research*, vol. 201, no. 2, pp. 644–648, 2010.

[11] M. Ji and T. C. E. Cheng, “Scheduling with job-dependent learning effects and multiple rate-modifying activities,” *Information Processing Letters*, vol. 110, no. 11, pp. 460–463, 2010.

[12] S.-J. Yang and D.-L. Yang, “Minimizing the total completion time in single-machine scheduling with aging/deteriorating effects and deteriorating maintenance activities,” *Computers & Mathematics with Applications*, vol. 60, no. 7, pp. 2161–2169, 2010.

[13] J. N. D. Gupta and S. K. Gupta, “Single facility scheduling with nonlinear processing times,” *Computers and Industrial Engineering*, vol. 14, no. 4, pp. 387–393, 1988.

[14] S. Browne and U. Yechiali, “Scheduling deteriorating jobs on a single processor,” *Operations Research*, vol. 38, no. 3, pp. 495–498, 1990.

[15] G. Mosheiov, “Scheduling jobs under simple linear deterioration,” *Computers and Operations Research*, vol. 21, no. 6, pp. 653–659, 1994.

[16] C. T. Ng, T. C. E. Cheng, A. Bachman, and A. Janiak, “Three scheduling problems with deteriorating jobs to minimize the total completion time,” *Information Processing Letters*, vol. 81, no. 6, pp. 327–333, 2002.

[17] G. Mosheiov, “Complexity analysis of job-shop scheduling with deteriorating jobs,” *Discrete Applied Mathematics*, vol. 117, no. 1–3, pp. 195–209, 2002.

[18] W. C. Lee, Y. R. Shiau, S. K. Chen, and C. C. Wu, “A two-machine flowshop scheduling problem with deteriorating jobs and blocking,” *International Journal of Production Economics*, vol. 124, no. 1, pp. 188–197, 2010.

[19] L. Sun, L. Sun, K. Cui, and J.-B. Wang, “A note on flow shop scheduling problems with deteriorating jobs on no-idle dominant machines,” *European Journal of Operational Research*, vol. 200, no. 1, pp. 309–311, 2010.

[20] S. Gawiejnowicz, “Scheduling deteriorating jobs subject to job or machine availability constraints,” *European Journal of Operational Research*, vol. 180, no. 1, pp. 472–478, 2007.

[21] M. Ji and T. C. E. Cheng, “Parallel-machine scheduling with simple linear deterioration to minimize total completion time,” *European Journal of Operational Research*, vol. 188, no. 2, pp. 342–347, 2008.

[22] J. B. Wang and L. Sun, “Single-machine group scheduling with linearly decreasing time-dependent setup times and job processing times,” *International Journal of Advanced Manufacturing Technology*, vol. 49, no. 5–8, pp. 765–772, 2010.

[23] W.-C. Lee, “A note on deteriorating jobs and learning in single-machine scheduling problems,” *International Journal of Business and Economics*, vol. 3, pp. 83–89, 2004.

[24] J.-B. Wang and T. C. E. Cheng, “Scheduling problems with the effects of deterioration and learning,” *Asia-Pacific Journal of Operational Research*, vol. 24, no. 2, pp. 245–261, 2007.

[25] T. C. E. Cheng, W.-C. Lee, and C.-C. Wu, “Scheduling problems with deteriorating jobs and learning effects including proportional setup times,” *Computers and Industrial Engineering*, vol. 58, no. 2, pp. 326–331, 2010.

[26] D. L. Yang and W. H. Kuo, “Some scheduling problems with deteriorating jobs and learning effects,” *Computers and Industrial Engineering*, vol. 58, no. 1, pp. 25–28, 2010.

[27] V. Gordon, J.-M. Proth, and C. Chu, “A survey of the state-of-the-art of common due date assignment and scheduling research,” *European Journal of Operational Research*, vol. 139, no. 1, pp. 1–25, 2002.

[28] D. Biskup and D. Simons, “Common due date scheduling with autonomous and induced learning,” *European Journal of Operational Research*, vol. 159, no. 3, pp. 606–616, 2004.

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

[30] G. Mosheiov and A. Sarig, “Scheduling a maintenance activity and due-window assignment on a single machine,” *Computers & Operations Research*, vol. 36, no. 9, pp. 2541–2545, 2009.

[31] S.-J. Yang, D.-L. Yang, and T. C. E. Cheng, “Single-machine due-window assignment and scheduling with job-dependent aging effects and deteriorating maintenance,” *Computers & Operations Research*, vol. 37, no. 8, pp. 1510–1514, 2010.

[32] C. Zhao and H. Tang, "A note to due-window assignment and single machine scheduling with deteriorating jobs and a rate-modifying activity," *Computers and Operations Research*. In press.

[33] G. Mosheiov, "A-shaped policies to schedule deteriorating jobs," *Journal of the Operational Research Society*, vol. 47, no. 9, pp. 1184–1191, 1996.

[34] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Optimization and approximation in deterministic sequencing and scheduling: a survey," *Annals of Discrete Mathematics*, vol. 5, pp. 287–326, 1979.

[35] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1982.

[36] P. Brucker, *Scheduling Algorithms*, Springer, Berlin, Germany, 3rd edition, 2001.