

SINGLE MACHINE DUE DATE ASSIGNMENT SCHEDULING PROBLEM WITH PRECEDENCE CONSTRAINTS AND CONTROLLABLE PROCESSING TIMES IN FUZZY ENVIRONMENT

J. LI, D. XU AND H. LI

ABSTRACT. In this paper, a due date assignment scheduling problem with precedence constraints and controllable processing times in uncertain environment is investigated, in which the basic processing time of each job is assumed to be the symmetric trapezoidal fuzzy number, and the linear resource consumption function is used. The objective is to minimize the crisp possibilistic mean (or expected) value of a cost function that includes the costs of earliness, tardiness, makespan and resource consumption jointly by scheduling the jobs under precedence constraints and determining the due date and the resource allocation amount satisfying resource constraints for each job. First, the problem is shown to be NP-hard. Furthermore, an optimal algorithm with polynomial time for the special case of this problem is put forward. Moreover, an efficient 2-approximation algorithm is presented based on solving the relaxation of the problem. Finally, the numerical experiment is given, whose results show that our method is promising.

1. Introduction

Just-In-Time(JIT) management system is usually employed by modern enterprises in supply chain management, which demands that the suppliers delivery goods as close to the required dates as possible in order to reduce the earliness-tardiness penalties [3]. One important goal of the JIT scheduling model is to discourage early and tardy jobs so as to minimize the earliness-tardiness penalties [31]. Due date assignment schedule strategies are usually adopted, especially when the goal is to schedule the jobs and determine the due dates of jobs such that the total cost is to be minimized [2, 10, 24, 28, 34].

Owing to the uncertainty inherent in practical applications, the data for the processing times of the jobs are usually recorded or collected under the influence of some unexpected noises. Hence, the processing times are not known exactly in advance. As a result, due date assignment models under uncertain environment are proposed by some researchers, in which the processing times are usually assumed to be random variables or fuzzy numbers [4, 6, 14, 16, 17, 21, 30]. For example, Cheng [6] considers a single machine due date assignment scheduling problem in which the

Received: May 2017; Revised: October 2017; Accepted: March 2018

Key words and phrases: Fuzzy scheduling, Fuzzy number, Possibilistic mean value and variance, Due date assignment scheduling, Precedence constraints, Controllable processing times.

processing times are stochastically independent random variables, and the objective is to determine the optimal due dates and the optimal schedule of jobs that jointly minimize the expected total cost. Soroush [30] investigates a due date assignment scheduling problem with stochastic processing times on a single machine, whose objective is to find the optimal schedule and determine the optimal due dates for jobs such that the expected total earliness and tardiness costs are minimized. Some efficient heuristics are given to find candidates for the optimal sequence, and one of these heuristics is shown to be asymptotically optimal by Portugal and Trietsch [21]. Furthermore, a branch and bound algorithm to find the optimal solution is put forward for this problem in [4]. Iranpoor [14] examines a schedule problem including assigning due dates and scheduling the jobs and maintenance activity on a single machine with stochastic processing times. For the cases that the historical data are not abundant for the processing time of a job to get the probability distribution, and intuition and judgment of the experienced workers play an important role in determining the processing time of a job, the due date assignment scheduling models in fuzzy environment are proposed [16, 17], in which the processing times are presented by fuzzy variables and the objective is to schedule the jobs and determine the due dates to jointly minimize the mean value of the earliness and tardiness penalties.

In practice, the customer may demand that the job is completed before a deadline, which would make the enterprise to allocate extra resources such as labor and/or capital to the job in order to speed up its processing time [15, 19, 27, 32, 33]. If the processing time is speed up by allocating extra resources, it is usually called the controllable processing time in resource allocation scheduling problems. The actual processing time of a job usually has two forms in resource allocation scheduling problems. One is linear resource consumption function, that is, the job J_j 's actual processing time is defined as $p_j = \bar{p}_j - s_j u_j$, where \bar{p}_j is the basic processing time, s_j is the compression rate, and u_j is the quantity of resources allocated for job J_j [35]. The other form is convex resource consumption function, that is, the job J_j 's actual processing time is given by $p_j = (w_j/u_j)^k$, where w_j is the workload along with the quantity u_j of resources allocated to job J_j and the given constant k [18]. A survey of scheduling problems with controllable processing times is given by Shabtay and Steiner [26]. In deterministic environment, the due date assignment scheduling problems with controllable processing times are investigated by many scholars [7, 23, 25, 29].

Note that, in resource allocation scheduling problems, the basic processing time of a job is not known exactly in advance due to some unexpected noises or/and insufficient historical data for the processing time. Always, the basic processing time is estimated by the experienced worker. For example, the basic processing time of a job is usually about 5 hours (denoted as $\bar{5}$ hours) by one worker. If the customer demands that the job must be completed in 3 hours, the firm has to speed up its processing time in order to satisfy the customer's demand. The firm may allocate the extra resources such as other workers to the job. For example, the actual processing time may be about 2.5 hours (denoted as $\tilde{2.5}$ hours) by allocating one more worker to the job. If the uncertain processing time of the job is speed up by

allocating extra resources, it is called the controllable processing time in uncertain environment. Besides, there are usually precedence constraints among jobs in real-life applications [20, 36]. In order to improve the satisfaction of the customers and reduce the inventor costs for the enterprises, it is important to schedule the jobs under the precedence constraints and determine the due date with controllable processing time for each job in uncertain environment. Taking all factors above into account, we study the due date assignment scheduling problem with precedence constraints and controllable processing times in uncertain environment, in which the basic processing time of each job is assumed to be fuzzy number, and the linear resource consumption function is used. The objective is to minimize the crisp possibilistic mean (or expected) value of a cost function that includes the costs of earliness, tardiness, makespan and resource consumption jointly by scheduling the jobs under precedence constraints and determining the due date and the resource allocation amount satisfying resource constraints for each job. First, we show that the problem of this paper is NP-hard. Furthermore, an optimal algorithm with polynomial time for special case of this problem is put forward. Also, an efficient 2-approximation algorithm is presented for the problem with general constraints based on solving a relaxation of the problem. Finally, the numerical experiment is given, whose results show that our method is promising.

The rest of the paper is organized as follows. In next section, some notions of the fuzzy sets theory used in this paper are introduced, and the formal description of the considered problem is given. In section 3, the computational complexity of the problem is discussed. In section 4, an optimal algorithm with polynomial time is proposed for special case of the considered problem. In section 5, a 2-approximation algorithm is presented for the problem with general constraints based on solving a relaxation of this kind of scheduling problems. In section 6 and section 7, the numerical experiment and some conclusions are given, respectively.

2. Preliminary and Problem Formulation

2.1. Preliminary. A fuzzy number \tilde{A} is a fuzzy set of the real line R with a normal, fuzzy convex and continuous membership function of bounded support, and the family of fuzzy numbers of the real line R is denoted by $\mathcal{F}(R)$ [5]. Let $\tilde{A} \in \mathcal{F}(R)$ and $\gamma \in [0, 1]$, the crisp set \tilde{A}_γ defined by $\{t \in R | \tilde{A}(t) \geq \gamma\}$ is called the γ -cut of \tilde{A} [9]. It follows from the definition of fuzzy number that \tilde{A}_γ is an interval, that is $\tilde{A}_\gamma = [a_1(\gamma), a_2(\gamma)]$. In [5], a trapezoidal fuzzy number \tilde{A} , denoted by (a, b, α, β) with peak $[a, b]$, left-width $\alpha > 0$ and right-width $\beta > 0$, is defined with the membership function

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x - a + \alpha}{\alpha}, & x \in [a - \alpha, a); \\ 1, & x \in [a, b]; \\ \frac{x - b - \beta}{-\beta}, & x \in [b, b + \beta]; \\ 0, & \text{otherwise.} \end{cases}$$

When $\alpha = \beta$, the trapezoidal fuzzy number is called the symmetric trapezoidal fuzzy number denoted by (a, b, α) . For the symmetric trapezoidal fuzzy number

(a, b, α) , $(b - a)$ is called peak width, and $\lambda = (b - a)/\alpha$ is called the ratio of peak-width and left-width. In this paper, the symmetric trapezoidal fuzzy number (a, b, α) with the ratio λ of peak-width and left-width is always simplistically called the symmetric trapezoidal fuzzy number (a, b, α) with ratio λ .

The following arithmetic operations of fuzzy numbers are given in [13]. For $\tilde{A}, \tilde{B} \in \mathcal{F}(R)$, let $\tilde{A}_\gamma = [a_1(\gamma), a_2(\gamma)]$ and $\tilde{B}_\gamma = [b_1(\gamma), b_2(\gamma)]$, then

$$\begin{aligned}\tilde{A} + \tilde{B} &= \bigcup_{\gamma \in [0,1]} \gamma[a_1(\gamma) + b_1(\gamma), a_2(\gamma) + b_2(\gamma)]; \\ \tilde{A} - \tilde{B} &= \bigcup_{\gamma \in [0,1]} \gamma[a_1(\gamma) - b_2(\gamma), a_2(\gamma) - b_1(\gamma)]; \\ c\tilde{A} &= \bigcup_{\gamma \in [0,1]} \gamma[ca_1(\gamma), ca_2(\gamma)] \text{ for } c \geq 0; \\ \max\{\tilde{A}, \tilde{B}\} &= \bigcup_{\gamma \in [0,1]} \gamma[a_1(\gamma) \vee b_1(\gamma), a_2(\gamma) \vee b_2(\gamma)].\end{aligned}$$

According to the operations above, for the trapezoidal fuzzy numbers (a_1, b_1, α_1) and (a_2, b_2, α_2) , it is easy to get

$$\begin{aligned}(a_1, b_1, \alpha_1) + (a_2, b_2, \alpha_2) &= (a_1 + a_2, b_1 + b_2, \alpha_1 + \alpha_2); \\ c(a, b, \alpha) &= (ca, cb, c\alpha) \text{ for } c \geq 0.\end{aligned}$$

Definition 2.1. [5] The crisp possibilistic mean value and variance of the fuzzy number \tilde{A} , denoted by $\bar{M}(\tilde{A})$ and $\text{Var}(\tilde{A})$, are defined as

$$\bar{M}(\tilde{A}) = \int_0^1 \gamma(a_1(\gamma) + a_2(\gamma))d\gamma \text{ and } \text{Var}(\tilde{A}) = \frac{1}{2} \int_0^1 \gamma(a_1(\gamma) - a_2(\gamma))^2 d\gamma,$$

respectively. The standard deviation of \tilde{A} is defined by $\sigma_{\tilde{A}} = \sqrt{\text{Var}(\tilde{A})}$.

According to the arithmetic operations of fuzzy numbers and Definition 2.1, it is easy to verify that

$$\bar{M}(\tilde{A} + \tilde{B}) = \bar{M}(\tilde{A}) + \bar{M}(\tilde{B}) \text{ and } \bar{M}(c\tilde{A}) = c\bar{M}(\tilde{A}) \text{ for } c \geq 0.$$

In [5], it is pointed out that, for the trapezoidal fuzzy number $\tilde{A} = (a, b, \alpha, \beta)$, $\bar{M}(\tilde{A}) = \frac{a+b}{2} + \frac{\beta-\alpha}{6}$ and $\text{Var}(\tilde{A}) = [\frac{b-a}{2} + \frac{\beta+\alpha}{6}]^2 + \frac{(\alpha+\beta)^2}{72}$, respectively. Also, for the symmetric trapezoidal fuzzy number $\tilde{A} = (a, b, \alpha)$ with ratio $\lambda = \frac{b-a}{\alpha}$, it is easy to get $\bar{M}(\tilde{A}) = \frac{a+b}{2}$, $\text{Var}(\tilde{A}) = [\frac{b-a}{2} + \frac{\alpha}{3}]^2 + \frac{\alpha^2}{18} = [\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}]\alpha^2$ and $\sigma_{\tilde{A}} = \alpha\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}$.

2.2. Problem Formulation. In this paper, a single machine due date assignment scheduling problem with precedence constraints and controllable processing times in uncertain environment is investigated, which can be formulated as follows. There is a set of jobs $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ to be processed on a single machine, in which the preemption of the jobs and idle machine time are not allowed, and the machine can process at most one job at a time; there are precedence constraints between

jobs. The precedence constraints are specified in the form of a directed acyclic graph G same as ones used in [1], in which each vertex represents a job. We say that J_i precedes J_j or J_j succeeds J_i , written $J_i \rightarrow J_j$, whenever there is a directed path from vertex J_i to vertex J_j in G ; the additional resources with total amount U_0 will be allotted to jobs; $u_i \geq 0$ denotes the amount of additional resources assigned to job J_i ; if there are no additional resources to jobs J_i , its processing time belongs to $[a_i - \alpha_i, b_i + \alpha_i]$ ($b_i \geq a_i \geq \alpha_i > 0$), and is assumed to be a symmetric trapezoidal fuzzy number (a_i, b_i, α_i) , which is called the basic processing time of job J_i , denoted as $\tilde{p}_i = (a_i, b_i, \alpha_i)$ ($i = 1, \dots, n$); if there are additional resources with amount u_i assigned to job J_i , its actual processing time \tilde{p}_i is assumed to be linear function of the amount of the allocated resources, i.e., the resource consumption function is of the form

$$\tilde{p}_i(u_i) = \tilde{p}_i - s_i u_i,$$

where $\tilde{p}_i = (a_i, b_i, \alpha_i)$ is the basic processing time, s_i is the positive compression rate of job J_i , $0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}$ and $\sum_{i=1}^n u_i \leq U_0$; a schedule is said to be feasible if all jobs can be successfully processed with respect to precedence constraints and resource constraints $\sum_{i=1}^n u_i \leq U_0$ and $0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}$ ($i = 1, \dots, n$); for a feasible schedule π , the deterministic due date of job J_i is denoted by d_i ; fuzzy numbers \tilde{C}_i , $\tilde{E}_i = \max\{d_i - \tilde{C}_i, 0\}$ and $\tilde{T}_i = \max\{\tilde{C}_i - d_i, 0\}$ represent the completion time, the earliness and the tardiness of job J_i , respectively; the per unit earliness penalties and the per unit tardiness penalties are e_i and t_i , respectively. The objective is to determine the due dates $\vec{d} = (d_1, d_2, \dots, d_n)$ and the resource allocation amounts $\vec{u} = (u_1, u_2, \dots, u_n)$ satisfying resource constraints and schedule the jobs under precedence constraints such that the crisp possibilistic mean (or expected) value of a cost function that includes the costs of earliness, tardiness, makespan and resource consumption is minimized. The objective function is denoted as follows:

$$f(\pi, \vec{d}, \vec{u}) = \bar{M}\left(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i)\right) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i,$$

where $\tilde{C}_{\max} = \max\{\tilde{C}_1, \dots, \tilde{C}_n\}$ is the maximal completion time (makespan), $\delta \geq 0$ is the cost of one unit of operation time, and $v_i \geq 0$ is the cost of one unit of resources allocated to job J_i . The problem discussed in this paper can be described by the following optimization problem:

$$\begin{aligned} \min \quad & f(\pi, \vec{d}, \vec{u}) = \bar{M}\left(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i)\right) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i \\ \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}; \\ & \sum_{i=1}^n u_i \leq U_0; \\ & \text{all jobs in } \pi \text{ satisfy the constraints } G. \end{aligned}$$

Extending the standard scheduling notations [11], this problem is denoted as

$$1|prec, \tilde{p}, resource, \vec{d}|\bar{M}\left(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i)\right) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i.$$

Definition 2.2. Let the completion time and the due date of a job be \tilde{C} and d . If the crisp possibilistic mean value of the job's earliness-tardiness penalties

$$\bar{M}(e \max\{d - \tilde{C}, 0\} + t \max\{\tilde{C} - d, 0\})$$

reaches the minimal value when $d = d^*$, then d^* is called the optimal due date with respect to the completion time \tilde{C} .

Definition 2.3. For a feasible schedule π of jobs, assign the due date d_i and the amount u_i of additional resources to $J_i (i = 1, \dots, n)$ and let $\vec{d} = (d_1, \dots, d_n)$ and $\vec{u} = (u_1, \dots, u_n)$. $\langle \pi^*, \vec{d}^*, \vec{u}^* \rangle$ is call the optimal schedule of jobs, if $f(\pi, \vec{d}, \vec{u}) = \bar{M}(\sum_{i=1}^n e_i \tilde{E}_i + \sum_{i=1}^n t_i \tilde{T}_i + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ reaches the minimal value when $\langle \pi, \vec{d}, \vec{u} \rangle = \langle \pi^*, \vec{d}^*, \vec{u}^* \rangle$. For convenience, π is also called an optimal schedule with \vec{d} and \vec{u} , or π is optimal.

3. Computational Complexity Results

In this section, we examine the computational complexity of the problem discussed in this paper, and show that it is NP-hard.

Lemma 3.1. *Let the per unit earliness penalties, the per unit tardiness penalties and the completion time of a job be e, t , and $\tilde{C} = (a, b, \alpha)$ with ratio $\lambda = (b - a)/\alpha$, then the optimal due date d^* of the job with respect to the completion time \tilde{C} is as follows:*

$$d^* = \bar{M}(\tilde{C}) + \frac{k^* - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}},$$

where

$$k^* = \begin{cases} \sqrt{\frac{2t}{t+e}} - 1, & t \leq e; \\ (\lambda + 1 - \sqrt{\frac{2e}{t+e}}), & t > e. \end{cases}$$

Proof. Do linear transform that $d = \bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}}$. By $\bar{M}(\tilde{C}) = \frac{a+b}{2}$, $\sigma_{\tilde{C}} = \sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}} \alpha$ and $\lambda = \frac{b-a}{\alpha}$, the linear transformation can be repressed by

$$d = \bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}} = \frac{a+b}{2} + (k - \frac{\lambda}{2}) \alpha = a + k\alpha.$$

Then, the crisp possibilistic mean value of the job's earliness and tardiness penalties can be expressed as follows:

$$\begin{aligned} & \bar{M}(e \max\{d - \tilde{C}, 0\} + t \max\{\tilde{C} - d, 0\}) \\ & = \bar{M}(e \max\{a + k\alpha - \tilde{C}, 0\} + t \max\{\tilde{C} - (a + k\alpha), 0\}) \triangleq g(k). \end{aligned} \tag{1}$$

Further, we have

$$g(k) = \bar{M}(e \max\{(a + k\alpha) - \tilde{C}, 0\} + t \max\{\tilde{C} - (a + k\alpha), 0\})$$

$$= \begin{cases} t \left(\frac{\lambda}{2} - k\right) \alpha, & k < -1; \\ t \left[\int_0^{1+k} \gamma (\lambda + 1 - k - \gamma) \alpha d\gamma + \int_{1+k}^1 \gamma (\lambda - 2k) \alpha d\gamma \right] \\ + e \left[\int_0^{1+k} \gamma (0 + 1 + k - \gamma) \alpha d\gamma \right], & -1 \leq k < 0; \\ e \left[\int_0^1 \gamma (1 + k - \gamma) \alpha d\gamma \right] + t \left[\int_0^1 \gamma (\lambda + 1 - k - \gamma) \alpha d\gamma \right], & 0 \leq k < \lambda; \\ e \left[\int_0^{1+\lambda-k} \gamma (1 + k - \gamma) \alpha d\gamma + \int_{1+\lambda-k}^1 \gamma (2k - \lambda) \alpha d\gamma \right] \\ + t \left[\int_0^{1+\lambda-k} \gamma (1 + \lambda - k - \gamma) \alpha d\gamma \right], & \lambda \leq k < 1 + \lambda; \\ e\alpha \left(k - \frac{\lambda}{2}\right), & k \geq \lambda + 1. \end{cases}$$

By computation, we can get $g(k) = \alpha h(e, t, \lambda, k)$ where

$$h(e, t, \lambda, k) = \begin{cases} t \left(\frac{\lambda}{2} - k\right), & k < -1; \\ t \left[\frac{(1+k)^3}{6} - k + \frac{\lambda}{2} \right] + e \frac{(1+k)^3}{6}, & -1 \leq k < 0; \\ t \left(\frac{\lambda}{2} - \frac{k}{2} + \frac{1}{6}\right) + e \frac{1+3k}{6}, & 0 \leq k < \lambda; \\ t \frac{(1+\lambda-k)^3}{6} + e \left[\frac{(1+\lambda-k)^3}{6} + k - \frac{\lambda}{2} \right], & \lambda \leq k < \lambda + 1; \\ e \left(k - \frac{\lambda}{2}\right), & k \geq \lambda + 1. \end{cases} \quad (2)$$

It is easy to get that

$$g'(k) = \alpha \begin{cases} -t, & k < -1; \\ t \frac{(1+k)^2}{2} + e \frac{(1+k)^2}{2} - t, & -1 \leq k < 0; \\ \frac{e}{2} - \frac{t}{2}, & 0 \leq k < \lambda; \\ e - t \frac{(1+\lambda-k)^2}{2} - e \frac{(1+\lambda-k)^2}{2}, & \lambda \leq k < \lambda + 1; \\ e, & k \geq \lambda + 1, \end{cases}$$

and $g''(k) \geq 0$. Hence, $g(k)$ reaches the minimum when

$$k = k^* = \begin{cases} \sqrt{\frac{2t}{t+e}} - 1, & t \leq e; \\ (\lambda + 1 - \sqrt{\frac{2e}{t+e}}), & t > e. \end{cases}$$

It follows that the optimal due date is

$$d^* = a + k^* \alpha = \bar{M}(\tilde{C}) + \frac{k^* - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}}.$$

Note that the linear transformation $d = \bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}} = a + k\alpha$ plays an important role in obtaining the optimal due date with respect to $\tilde{C} = (a, b, \alpha)$ with the ratio $\lambda = \frac{b-a}{\alpha}$. In fact, $k = \frac{d-a}{\alpha} = \frac{\lambda}{2} + \sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}} \frac{[d - \bar{M}(\tilde{C})]}{\sigma_{\tilde{C}}}$ is an important □

index which can express the level of the due date minimizing the earliness and tardiness penalties with respect to the completion time in some sense. In practice, k is sometimes called the customer service level of the due date with respect to the completion time in uncertain environment [14, 16].

Also, by the linear transformation $d = \bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}}$, the earliness \tilde{E} and tardiness \tilde{T} can also be represented as $\tilde{E} = \max\{d - \tilde{C}, 0\} = \max\{\bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}} - \tilde{C}, 0\}$, and $\tilde{T} = \max\{\tilde{C} - d, 0\} = \max\{\tilde{C} - (\bar{M}(\tilde{C}) + \frac{k - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}}), 0\}$, respectively. Hence, the crisp possibilistic mean value of the job's earliness-tardiness penalties $\bar{M}(e\tilde{E} + t\tilde{T})$ is

$$\bar{M}(e \max\{\bar{M}(\tilde{C}) + \frac{(k - \frac{\lambda}{2})\sigma_{\tilde{C}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} - \tilde{C}, 0\} + t \max\{\tilde{C} - \bar{M}(\tilde{C}) - \frac{(k - \frac{\lambda}{2})\sigma_{\tilde{C}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}}, 0\}).$$

Definition 3.2. For a job with the due date d and the completion time $\tilde{C} = (a, b, \alpha)$ with ratio $\lambda = \frac{b-a}{\alpha}$,

$$k = \frac{\lambda}{2} + \sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}} \frac{[d - \bar{M}(\tilde{C})]}{\sigma_{\tilde{C}}}$$

is called the customer service level of the due date d with respect to the completion time \tilde{C} ; k^* is called the optimal customer service level with respect to the completion time \tilde{C} , if the crisp possibilistic mean value of the job's earliness-tardiness penalties

$$\bar{M}(e \max\{\bar{M}(\tilde{C}) + \frac{(k - \frac{\lambda}{2})\sigma_{\tilde{C}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} - \tilde{C}, 0\} + t \max\{\tilde{C} - \bar{M}(\tilde{C}) - \frac{(k - \frac{\lambda}{2})\sigma_{\tilde{C}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}}, 0\})$$

reaches the minimal value when $k = k^*$.

It is easy to verify that when d^* is the optimal due date with respect to the completion time $\tilde{C} = (a, b, \alpha)$ with $\lambda = (b - a)/\alpha$, $k^* \triangleq (d^* - a)/\alpha = \frac{\lambda}{2} + \sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}} \frac{d^* - \bar{M}(\tilde{C})}{\sigma_{\tilde{C}}}$ is the optimal customer service level with respect to \tilde{C} . On the other hand, if k^* is the optimal customer service level with respect to $\tilde{C} = (a, b, \alpha)$ with $\lambda = (b - a)/\alpha$, then $d^* = a + k^*\alpha = \bar{M}(\tilde{C}) + \frac{k^* - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}}$ is the optimal due date with respect to \tilde{C} .

Let $\vec{k} = (k_1, k_2, \dots, k_n)$ where k_i is the customer service level of the due date d_i with the completion \tilde{C}_i with ratio λ_i . According to Definition 3.2, the objective function $f(\pi, \vec{d}, \vec{u})$ can be represented by \vec{k} as follows:

$$\begin{aligned} f(\pi, \vec{d}, \vec{u}) &= \bar{M}(\sum_{i=1}^n e_i \tilde{E}_i + \sum_{i=1}^n t_i \tilde{T}_i + \delta \tilde{C} \max + \sum_{i=1}^n v_i u_i) \\ &= \bar{M}(\sum_{i=1}^n e_i \max\{d_i - \tilde{C}_i, 0\} + t_i \max\{\tilde{C}_i - d_i, 0\}) + \delta \bar{M}(\tilde{C} \max) + \sum_{i=1}^n v_i u_i \\ &= \sum_{i=1}^n \bar{M}(e_i \max\{\bar{M}(\tilde{C}_i) + \frac{(k_i - \frac{\lambda_i}{2})\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_i^2}{4} + \frac{\lambda_i}{3} + \frac{1}{6}}} - \tilde{C}_i, 0\} \\ &\quad + t_i \max\{\tilde{C}_i - (\bar{M}(\tilde{C}_i) + \frac{(k_i - \frac{\lambda_i}{2})\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_i^2}{4} + \frac{\lambda_i}{3} + \frac{1}{6}}}), 0\}) + \delta \bar{M}(\max\{\tilde{C}_1, \dots, \tilde{C}_n\}) + \sum_{i=1}^n v_i u_i \\ &\triangleq g(\pi, \vec{k}, \vec{u}). \end{aligned}$$

Our objective is also to sequence the jobs under precedence constraints and determine the customer service levels $\vec{k} = (k_1, k_2, \dots, k_n)$ and the amounts $\vec{u} = (u_1, u_2, \dots, u_n)$ of resources assigned to jobs to minimize the function $g(\pi, \vec{k}, \vec{u})$.

Furthermore, without loss of generality, let $\pi = (J_1, J_2, \dots, J_n)$ with the basic processing time $\vec{p}_i = (a_i, b_i, \alpha_i)$ with $\lambda_i = (b_i - a_i)/\alpha_i$ and u_i be the amount of the additional resources allocated to job J_i . Then the completion time of job J_i is $\tilde{C}_i = \sum_{j=1}^i \tilde{p}_i = (\sum_{j=1}^i (a_j - s_j u_j), \sum_{j=1}^i (b_j - s_j u_j), \sum_{j=1}^i \alpha_j)$ ($i = 1, \dots, n$). Given customer service level k_i for J_i , the due date of J_i under k_i is

$$d_i = \bar{M}(\tilde{C}_i) + \frac{k_i - \frac{\lambda_{\tilde{C}_i}}{2}}{\sqrt{\frac{\lambda_{\tilde{C}_i}^2}{4} + \frac{\lambda_{\tilde{C}_i}}{3} + \frac{1}{6}}} \sigma_{\tilde{C}_i} = \sum_{j=1}^i (a_j - s_j u_j) + k_i \sum_{j=1}^i \alpha_j,$$

where $\lambda_{\tilde{C}_i} = \frac{\sum_{j=1}^i (b_j - a_j)}{\sum_{j=1}^i \alpha_j}$. It follows by Lemma 3.1 that

$$\begin{aligned} g(\pi, \vec{k}, \vec{u}) &= \bar{M} \left(\sum_{i=1}^n e_i \max \left\{ \sum_{j=1}^i (a_j - s_j u_j) + k_i \sum_{j=1}^i \alpha_j - \tilde{C}_i, 0 \right\} \right. \\ &\quad \left. + t_i \max \left\{ \tilde{C}_i - \left(\sum_{j=1}^i (a_j - s_j u_j) + k_i \sum_{j=1}^i \alpha_j \right), 0 \right\} \right) + \delta \bar{M}(\tilde{C}_{\max}) + \sum_{i=1}^n v_i u_i \\ &= \sum_{i=1}^n \bar{M} \left(e_i \max \left\{ \sum_{j=1}^i (a_j - s_j u_j) + k_i \sum_{j=1}^i \alpha_j - \tilde{C}_i, 0 \right\} \right. \\ &\quad \left. + t_i \max \left\{ \tilde{C}_i - \left(\sum_{j=1}^i (a_j - s_j u_j) + k_i \sum_{j=1}^i \alpha_j \right), 0 \right\} \right) + \delta \bar{M}(\tilde{C}_n) + \sum_{i=1}^n v_i u_i \quad (3) \\ &= \sum_{i=1}^n \left\{ h(e_i, t_i, \lambda_{\tilde{C}_i}, k_i) \sum_{j=1}^i \alpha_j \right\} + \delta \left[\sum_{i=1}^n \frac{a_i + b_i}{2} - \sum_{i=1}^n s_i u_i \right] + \sum_{i=1}^n v_i u_i \\ &= \sum_{i=1}^n \left\{ h(e_i, t_i, \lambda_{\tilde{C}_i}, k_i) \frac{\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_{\tilde{C}_i}^2}{4} + \frac{\lambda_{\tilde{C}_i}}{3} + \frac{1}{6}}} \right\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i. \end{aligned}$$

Lemma 3.3. *Let π be an optimal schedule of jobs set $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ with $\vec{k} = (k_1, k_2, \dots, k_n)$ and $\vec{u} = (u_1, u_2, \dots, u_n)$, in which $\vec{p}_i = (a_i, b_i, \alpha_i)$ with $\lambda_i = (b_i - a_i)/\alpha_i$ ($i = 1, \dots, n$). If $\lambda_1 = \lambda_2 = \dots = \lambda_n = \lambda$, then*

$$k_i = k_i^* \triangleq \begin{cases} \sqrt{\frac{2t_i}{t_i + e_i}} - 1, & t_i \leq e_i; \\ (\lambda_i + 1 - \sqrt{\frac{2e_i}{t_i + e_i}}), & t_i > e_i, \end{cases} \quad (4)$$

and $\vec{u} = (u_1, u_2, \dots, u_n)$ is a solution of the linear programming as follows:

$$\begin{aligned}
\min \quad & \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\
\text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}; \\
& \sum_{i=1}^n u_i \leq U_0.
\end{aligned} \tag{5}$$

Proof. According to $\lambda_1 = \lambda_2 = \dots = \lambda_n = \lambda$ for the jobs with $\bar{p}_i = (a_i, b_i, \alpha_i)$ ($i = 1, \dots, n$), it is easy to verify that $\lambda_{\bar{C}_i} = \frac{b_{c_i} - a_{c_i}}{\alpha_{c_i}} = \lambda$ for the completion time $\bar{C}_i = (a_{c_i}, b_{c_i}, \alpha_{c_i})$ for job J_i in any feasible schedule of jobs. According to Lemma 3.1 and equation (1), we have $h(e_{i_j}, t_{i_j}, \lambda, k_{i_j}) \geq h(e_{i_j}, t_{i_j}, \lambda, k_{i_j}^*)$. Suppose there exists j_0 such that $k_{j_0} \neq k_{j_0}^*$. Without loss of generality, let $j_0 = 1$. It follows that

$$\begin{aligned}
g(\pi, \vec{k}, \vec{u}) &= \sum_{l=1}^n \{h(e_{i_l}, t_{i_l}, \lambda_{i_l}, k_{i_l}) \sum_{j=1}^l \alpha_{i_j}\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\
&> h(e_{i_1}, t_{i_1}, \lambda_{i_1}, k_{i_1}^*) \alpha_{i_1} + \sum_{l=2}^n \{h(e_{i_l}, t_{i_l}, \lambda_{i_l}, k_{i_l}^*) \sum_{j=1}^l \alpha_{i_j}\} \\
&\quad + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i = g(\pi, \vec{k}^*, \vec{u}),
\end{aligned}$$

where $\vec{k}^* = (k_1^*, k_2^*, \dots, k_n^*)$. A contradiction appears since π is optimal.

Also, suppose \vec{u} is not a solution of the linear programming (5). Since π is feasible, it is easy to get $0 \leq u_i \leq (a_i - \alpha_i)/s_i$ and $\sum_{i=1}^n u_i \leq U_0$, which implies \vec{u} is in feasible domain of the linear programming (5). Let \vec{u}^* be a solution of the linear programming (5). Then, we have

$$\begin{aligned}
g(\pi, \vec{k}, \vec{u}) &= \sum_{i=1}^n \{h(e_i, t_i, \lambda_{\bar{C}_i}, k_i) \sum_{j=1}^i \alpha_j\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\
&> \sum_{i=1}^n \{h(e_i, t_i, \lambda_{\bar{C}_i}, k_i) \sum_{j=1}^i \alpha_j\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i^* = g(\pi, \vec{k}, \vec{u}^*).
\end{aligned}$$

A contradiction also appears since π is optimal. \square

In the rest of the paper, we use $\vec{k}^* = (k_1^*, k_2^*, \dots, k_n^*)$ to denote the optimal service level vector for an optimal schedule π of the jobs, where k_i^* is obtained by equation (4).

By the lemma above, a reduction based on the problem $1|prepr|\sum_{i=1}^n w_i C_i$ can be obtained.

Theorem 3.4. *The problem $1|prec, \tilde{p}, resource, \tilde{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}max + \sum_{i=1}^n v_i u_i)$ is NP-hard, even if $\lambda_{\bar{p}_i} = (b_i - a_i)/\alpha_i = \lambda$ for the basic processing time $\bar{p}_i = (a_i, b_i, \alpha_i)$ of J_i ($i = 1, \dots, n$).*

Proof. We will show that the NP-hard problem $1|prep|\sum_{i=1}^n w_i C_i$ can be polynomially reducible to the problem $1|prec, \tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$. Let us recall that an instance I of $1|prep|\sum_{i=1}^n w_i C_i$, in which jobs set $\mathcal{J} = \{J_1, \dots, J_n\}$, the precedence constraints between jobs is G , job J_i 's processing time $p_i > 0$, and the weight of job J_i is $w_i (i = 1, \dots, n)$. The corresponding instance I' of the problem $1|prec, \tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ is constructed as follows: let jobs set $\mathcal{J}' = \{J'_1, J'_2, \dots, J'_n\}$ where J'_i corresponds J_i of instance I ; for J'_i , let $\tilde{p}_i = (p_i + 1, 2p_i + 1, p_i)$, $t_i = e_i = \frac{6}{5}w_i$, $\delta = 1, s_i = 1, v_i = 2, 0 \leq u_i \leq 1 (i = 1, \dots, n)$, and $\sum_{i=1}^n u_i \leq n$; let the precedence constraints between jobs in \mathcal{J}' denoted as G' be same as G .

Without of generality, let a sequence of the instance I be $\pi = (J_1, J_2, \dots, J_n)$. Then the corresponding sequence of instance I' is $\pi' = (J'_1, J'_2, \dots, J'_n)$. Since two instances have the same precedence constraints, π is feasible schedule of I if and only if π' is feasible one of I' . Also, for job J_i , it is easy to get the ratio for $\tilde{p}_i = (p_i + 1, 2p_i + 1, p_i)$ is 1; it follows that the ratio of \tilde{C}_i is also 1; by equation (4), we have the optimal customer service level $k_i^* = 0$; it follows that $h(e_i, t_i, 1, 0) = \frac{5e_i}{6} = w_i$. Let \vec{u}^* be a solution of the following linear programming

$$\begin{aligned} \min \quad & \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i = \sum_{i=1}^n u_i + \sum_{i=1}^n \frac{3p_i + 2}{2} \\ \text{s.t.} \quad & 0 \leq u_i \leq \frac{p_i + 1 - p_i}{s_i} = 1; \\ & \sum_{i=1}^n u_i \leq U_0 = n. \end{aligned}$$

It is easy to verify that the above linear programming has only one solution $\vec{u} = (0, \dots, 0)$. Hence, $\vec{u}^* = (0, \dots, 0)$. Then we have

$$\begin{aligned} f(\pi', \vec{d}^*, \vec{u}^*) &= g(\pi', \vec{k}^*, \vec{u}^*) = \sum_{i=1}^n \{h(e_i, t_i, 1, 0) \sum_{j=1}^i p_j\} + \sum_{i=1}^n \frac{3p_i + 2}{2} \\ &= \sum_{i=1}^n w_i (\sum_{j=1}^i p_j) + \sum_{i=1}^n \frac{3p_i + 2}{2} = \sum_{i=1}^n w_i C_i + \sum_{i=1}^n \frac{3p_i + 2}{2}, \end{aligned}$$

which implies that π is an optimal schedule of I if and only if π' is an optimal one of I' . It is easy to verify that instance I' can be obtained from instance I in polynomial time. This means that having a polynomial algorithm for $1|prec, resource, \tilde{p}, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$, we would be able to solve the NP-hard problem $1|prep|\sum_{i=1}^n w_i C_i$ in polynomial time. Therefore, the problem $1|prec, \tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ is NP-hard. \square

4. The Polynomially Solvable Case

In above section, we show that the problem $1|prec, \tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ is NP-hard, even if $\lambda_{\tilde{p}_i} = \lambda (i = 1, \dots, n)$. In this

section, we focus on the problem with no precedence constraints between jobs, which is denoted as $1|\tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$.

In the rest of this section, it is assume that $\lambda_{\tilde{p}_i} = (b_i - a_i)/\alpha_i = \lambda$ for the basic processing time $\tilde{p}_i = (a_i, b_i, \alpha_i)$ of $J_i (i = 1, \dots, n)$. Indeed, we can design **a polynomial time algorithm** to solve the case that $\lambda_{\tilde{p}_i} = \lambda (i = 1, \dots, n)$ as follows:

Step 1: Input the cost of one unit of operation time δ , the total amount of resources U_0 , the basic processing time (a_i, b_i, α_i) , the per unit earliness penalties e_i , the per unit tardiness penalties t_i , the cost of one unit of resources v_i , and the compression rate s_i for job $J_i (i = 1, \dots, n)$.

Step 2: Construct the linear programming induce by the problem $1|\tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ according to (5) as follows:

$$\begin{aligned} \min \quad & \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\ \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i} (i = 1, \dots, n); \\ & \sum_{i=1}^n u_i \leq U_0. \end{aligned}$$

Solve the linear programming above and obtain the solution $\vec{u} = (u_1, \dots, u_n)$. Then compute the actual processing time $\tilde{p}_i = (a_i, b_i, \alpha_i) - s_i u_i = (a_i - s_i u_i, b_i - s_i u_i, \alpha_i)$ for $J_i (i = 1, \dots, n)$.

Step 3: Compute the optimal customer service level k_i according to equation (4), then let $w_i = h(e_i, t_i, \lambda, k_i)$ where the function $h(e, t, \lambda, k)$ is given by (2) and $\lambda = (b_i - a_i)/\alpha_i (i = 1, \dots, n)$.

Step 4: Schedule the jobs according to the nonincreasing order of w_i/α_i . This schedule is denoted as π .

Step 5: Compute the completion time \tilde{C}_i of J_i in the schedule π obtained in step 3, then assign the due date to J_i by $d_i = \bar{M}(\tilde{C}_i) + \frac{k_i - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}_i} (i = 1, \dots, n)$.

Step 6: Return $\pi, \vec{d} = (d_1, d_2, \dots, d_n), \vec{u} = (u_1, u_2, \dots, u_n)$ and $\vec{k} = (k_1, k_2, \dots, k_n)$.

Theorem 4.1. *The algorithm above is an optimal algorithm with polynomial time for the problem $1|\tilde{p}, resource, \vec{d}|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ with $\lambda_{\tilde{p}_i} = (b_i - a_i)/\alpha_i = \lambda$ for the basic processing time $\tilde{p}_i = (a_i, b_i, \alpha_i)$ of $J_i (i = 1, \dots, n)$.*

Proof. It is easy to know that the time spent on the step 1, step 3, step 5 and step 6 of the above algorithm is $O(n)$, and the time spent on step 4 is $O(n \log(n))$. Also, according to [8], the linear programming in step 2 can be solve in polynomial time. Hence, the algorithm above is a polynomial time algorithm.

Assume that there exists an optimal schedule π with \vec{k} and \vec{u} not obtained by the above algorithm. Known from Lemma 3.3, k_i is the same as one obtained by the

above algorithm for J_i . Also, \vec{u} is a solution of the linear programming of the step 2. According to the logic of the algorithm above, it follows that there are at least two adjacent jobs J_j and J_i (J_j before J_i in π) such that $(w_j/\alpha_j) < (w_i/\alpha_i)$. Next, let schedule π' be obtained by interchanging J_j and J_i without changing other jobs. According to equation (3), it is easy to get

$$g(\pi, \vec{k}, \vec{u}) - g(\pi', \vec{k}, \vec{u}) = \alpha_j \alpha_i \left(\frac{w_i}{\alpha_i} - \frac{w_j}{\alpha_j} \right) > 0.$$

Thus, a contradiction appears. \square

5. Approximation Algorithm

Note that the problem $1|prec, \tilde{p}, resource, d|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ is NP-hard even if $\lambda_{\tilde{p}_i} = \frac{b_i - a_i}{\alpha_i} = \lambda$ for the basic processing time $\tilde{p}_i = (a_i, b_i, \alpha_i)$. Motivated by some polyhedral methods for the single machine scheduling problems [12, 22], we can design an approximation algorithm for the problem $1|prec, \tilde{p}, resource, d|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ when $\lambda_{\tilde{p}_i} = \lambda$ based on solving a linear relaxation of the problem, whose worst case ratio is shown to be 2.

In a feasible π , the actual processing time and completion time of J_i ($i = 1, \dots, n$) are \tilde{p}_i and \tilde{C}_i . And, the standard deviations of the actual processing time and the completion time of job J_i are denoted as $\sigma_{\tilde{p}_i}$ and $\sigma_{\tilde{C}_i}$, respectively. By Definition 2.1, for \tilde{p}_i and \tilde{p}_j , we have $\sigma_{\tilde{p}_i} = \sigma_{\tilde{p}_j}$, and $\sigma_{\tilde{p}_i + \tilde{p}_j} = \sigma_{\tilde{p}_i} + \sigma_{\tilde{p}_j}$; furthermore, for \tilde{C}_i and \tilde{C}_j , we also have $\sigma_{\tilde{C}_i + \tilde{C}_j} = \sigma_{\tilde{C}_i} + \sigma_{\tilde{C}_j}$. It follows that $\sigma_{\tilde{C}_i} \leq \sigma_{\tilde{C}_j}$ if J_i is processed before J_j in a feasible schedule π , which means that the schedule π is in accordance with the sequence of $\sigma_{\tilde{C}_1}, \dots, \sigma_{\tilde{C}_n}$ which is sorted from small to large. Also, the precedence constraints G can also be presented as ones of $\sigma_{\tilde{C}_1}, \dots, \sigma_{\tilde{C}_n}$ as follows: $\sigma_{\tilde{C}_i} \geq \sigma_{\tilde{p}_i}$ for $i = 1, \dots, n$; $\sigma_{\tilde{C}_k} \geq \sigma_{\tilde{C}_i} + \sigma_{\tilde{p}_k}$ for each arc $J_i J_k$ in G , and $\sigma_{\tilde{C}_k} \geq \sigma_{\tilde{C}_i} + \sigma_{\tilde{p}_k}$ or $\sigma_{\tilde{C}_i} \geq \sigma_{\tilde{C}_k} + \sigma_{\tilde{p}_i}$ for any two jobs J_k and J_i . According to Lemma 3.1 and equation (3), for the schedule π with the customer service levels \vec{k} and assigned additional resource amounts \vec{u} , the objective function is $g(\pi, \vec{k}, \vec{u}) = \sum_{i=1}^n \{h(e_i, t_i, \lambda_{\tilde{C}_i}, k_i) \frac{\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_{\tilde{C}_i}^2}{4} + \frac{\lambda_{\tilde{C}_i}}{3} + \frac{1}{6}}}\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i$. Hence,

we can formulate $1|prec, \tilde{p}, resource, d|\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i)$ in the following way, where $\sigma_{\tilde{C}_i}$ are used as the variables and the constraints ensure that the variables $\sigma_{\tilde{C}_1}, \dots, \sigma_{\tilde{C}_n}$ specify a feasible set of standard deviations of the completion times:

$$\begin{aligned} \min \quad & \sum_{i=1}^n \{h(e_i, t_i, \lambda_{\tilde{C}_i}, k_i) \frac{\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_{\tilde{C}_i}^2}{4} + \frac{\lambda_{\tilde{C}_i}}{3} + \frac{1}{6}}}\} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\ \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}; \\ & \sum_{i=1}^n u_i \leq U_0; \\ & \sigma_{\tilde{C}_i} \geq \sigma_{\tilde{p}_i} \text{ for } i = 1, \dots, n; \\ & \sigma_{\tilde{C}_k} \geq \sigma_{\tilde{C}_i} + \sigma_{\tilde{p}_k} \text{ for each arc } J_i J_k \text{ in } G; \\ & \sigma_{\tilde{C}_k} \geq \sigma_{\tilde{C}_i} + \sigma_{\tilde{p}_k} \text{ or } \sigma_{\tilde{C}_i} \geq \sigma_{\tilde{C}_k} + \sigma_{\tilde{p}_i} \text{ for any two jobs } J_k, J_i, \end{aligned} \tag{6}$$

where u_1, u_2, \dots, u_n are also variables and $h(e, t, \lambda, k)$ is given in (2).

Note that the equation (6) is not linear inequality. However, for a feasible scheduling $\pi = \{J_{\pi(1)}, J_{\pi(2)}, \dots, J_{\pi(n)}\}$ of jobs set \mathcal{J} , let $S \subset \mathcal{J}$, then we schedule the jobs of S according to the order of the jobs in π , which is denoted as $J_{\pi(s_1)} -$

$J_{\pi(s_2)} - \dots - J_{\pi(s_l)}$. It follows that $\sigma_{\tilde{C}_{\pi(k)}} = \sum_{j=1}^k \sigma_{\tilde{p}_{\pi(j)}}$ and $\sigma_{\tilde{C}_{\pi(s_j)}} = \sum_{i=1}^{s_j} \sigma_{\tilde{p}_{\pi(i)}}$. For convenience, the following notations are used: for any set $S \subset \mathcal{J} = \{J_1, \dots, J_n\}$,

$$\sigma_{\tilde{p}}(S) \triangleq \sum_{J_i \in S} \sigma_{\tilde{p}_i}, \quad \sigma_{\tilde{p}}^2(S) \triangleq \sum_{J_i \in S} \sigma_{\tilde{p}_i}^2 \quad \text{and} \quad \sigma_{\tilde{p}}(S)^2 \triangleq \left[\sum_{J_i \in S} \sigma_{\tilde{p}_i} \right]^2.$$

Then, by $\sigma_{\tilde{p}_i} = \sigma_{\tilde{p}_i}$, we have

$$\begin{aligned} \sum_{J_i \in S} \sigma_{\tilde{p}_i} \sigma_{\tilde{C}_i} &= \sum_{j=1}^l \sigma_{\tilde{p}_{\pi(s_j)}} \sigma_{\tilde{C}_{\pi(s_j)}} = \sum_{j=1}^l \sigma_{\tilde{p}_{\pi(s_j)}} \left(\sum_{i=1}^{s_j} \sigma_{\tilde{p}_{\pi(i)}} \right) \\ &= \sigma_{\tilde{p}_{\pi(s_1)}} [\sigma_{\tilde{p}_{\pi(1)}} + \sigma_{\tilde{p}_{\pi(2)}} + \dots + \sigma_{\tilde{p}_{\pi(s_1)}}] \\ &\quad + \sigma_{\tilde{p}_{\pi(s_2)}} [\sigma_{\tilde{p}_{\pi(1)}} + \sigma_{\tilde{p}_{\pi(2)}} + \dots + \sigma_{\tilde{p}_{\pi(s_1)}} + \dots + \sigma_{\tilde{p}_{\pi(s_2)}}] + \dots \\ &\quad + \sigma_{\tilde{p}_{\pi(s_l)}} [\sigma_{\tilde{p}_{\pi(1)}} + \sigma_{\tilde{p}_{\pi(2)}} + \dots + \sigma_{\tilde{p}_{\pi(s_{l-1})}} + \dots + \sigma_{\tilde{p}_{\pi(s_l)}}] \\ &\geq \sigma_{\tilde{p}_{\pi(s_1)}} [\sigma_{\tilde{p}_{\pi(s_1)}}] + \sigma_{\tilde{p}_{\pi(s_2)}} [\sigma_{\tilde{p}_{\pi(s_1)}} + \sigma_{\tilde{p}_{\pi(s_2)}}] + \dots \\ &\quad + \sigma_{\tilde{p}_{\pi(s_l)}} [\sigma_{\tilde{p}_{\pi(s_1)}} + \sigma_{\tilde{p}_{\pi(s_2)}} + \dots + \sigma_{\tilde{p}_{\pi(s_l)}}] \\ &= \frac{1}{2} \left[\sum_{j=1}^l \sigma_{\tilde{p}_{\pi(s_j)}}^2 + \left(\sum_{j=1}^l \sigma_{\tilde{p}_{\pi(s_j)}} \right)^2 \right] = \frac{1}{2} [\sigma_{\tilde{p}}^2(S) + \sigma_{\tilde{p}}(S)^2]. \end{aligned}$$

Based on the above linear inequalities, using the standard deviations of the completion times as the variables, a linear programming relaxation of the problem

$$1|prec, \tilde{p}, resource, d|\bar{M} \left(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{\max} + \sum_{i=1}^n v_i u_i \right)$$

can be represented as follows:

$$\begin{aligned} \min \quad & \sum_{i=1}^n h(e_i, t_i, \lambda_{\tilde{C}_i}, k_i) \frac{\sigma_{\tilde{C}_i}}{\sqrt{\frac{\lambda_{\tilde{C}_i}^2}{4} + \frac{\lambda_{\tilde{C}_i}}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\ \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}; \\ & \sum_{i=1}^n u_i \leq U_0; \\ & \sigma_{\tilde{C}_i} \geq \sigma_{\tilde{p}_i} \quad \text{for } i = 1, \dots, n; \\ & \sigma_{\tilde{C}_k} \geq \sigma_{\tilde{C}_i} + \sigma_{\tilde{p}_k} \quad \text{for each arc } J_i J_k \text{ in } G; \\ & \sum_{J_j \in S} \sigma_{\tilde{p}_j} \sigma_{\tilde{C}_j} \geq \frac{1}{2} (\sigma_{\tilde{p}}^2(S) + \sigma_{\tilde{p}}(S)^2) \quad \text{for each } S \subset \mathcal{J}. \end{aligned} \tag{7}$$

It is easy to verify that $(\sigma_{\tilde{C}_1}, \sigma_{\tilde{C}_2}, \dots, \sigma_{\tilde{C}_n}, u_1, u_2, \dots, u_n)$ is a solution of (7) if and only if $(\sigma_{\tilde{C}_1}, \sigma_{\tilde{C}_2}, \dots, \sigma_{\tilde{C}_n})$ and (u_1, u_2, \dots, u_n) are the solutions of the following

linear programmings (8) and (9), respectively:

$$\begin{aligned}
 \min \quad & \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\
 \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i}; \\
 & \sum_{i=1}^n u_i \leq U_0.
 \end{aligned} \tag{8}$$

and

$$\begin{aligned}
 \min \quad & \sum_{i=1}^n \left\{ h(e_i, t_i, \lambda \bar{C}_i, k_i) \frac{\sigma_{\bar{C}_i}}{\sqrt{\frac{\lambda^2 \bar{C}_i^2}{4} + \frac{\lambda \bar{C}_i}{3} + \frac{1}{6}}} \right\} \\
 \text{s.t.} \quad & \sigma_{\bar{C}_i} \geq \sigma_{\bar{p}_i} \text{ for } i = 1, \dots, n; \\
 & \sigma_{\bar{C}_k} \geq \sigma_{\bar{C}_i} + \sigma_{\bar{p}_k} \text{ for each arc } J_i J_k \text{ in } G; \\
 & \sum_{J_j \in S} \sigma_{\bar{p}_j} \sigma_{\bar{C}_j} \geq \frac{1}{2} (\sigma_{\bar{p}}^2(S) + \sigma_{\bar{p}}(S)^2) \text{ for each } S \subset \mathcal{J}.
 \end{aligned} \tag{9}$$

Based on the linear programmings above, we can design an **approximation algorithm** for $1|prec, \tilde{p}, resource, \vec{d}| \bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}max + \sum_{i=1}^n v_i u_i)$ with $\lambda_{\tilde{p}_i} = \lambda(1, \dots, n)$ as follows:

Step 1: Input the cost of one unit of operation time δ , the total amount of additional resources U_0 , the precedence constraints G among jobs, the basic processing time (a_i, b_i, α_i) , the per unit earliness penalties e_i , the per unit tardiness penalties t_i , the cost of one unit of resources v_i , and the compression rate s_i for job $J_i (i = 1, \dots, n)$.

Step 2: Construct the linear programming induce by $1|prec, \tilde{p}, resource, \vec{d}| \bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}max + \sum_{i=1}^n v_i u_i)$ according to (8) as follows:

$$\begin{aligned}
 \min \quad & \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\
 \text{s.t.} \quad & 0 \leq u_i \leq \frac{a_i - \alpha_i}{s_i} (i = 1, \dots, n); \\
 & \sum_{i=1}^n u_i \leq U_0.
 \end{aligned}$$

Solve the linear programming above and obtain the solution $\vec{u} = (u_1, \dots, u_n)$. Then compute the actual processing time $\tilde{p} = (a_i, b_i, \alpha_i) - s_i u_i = (a_i - s_i u_i, b_i - s_i u_i, \alpha_i)$ for $J_i (i = 1, \dots, n)$.

Step 3: Compute the optimal customer service level k_i according to equation (4), then let $w_i = h(e_i, t_i, \lambda, k_i)$ where $\lambda = (b_i - a_i)/\alpha_i (i = 1, \dots, n)$.

Step 4: Construct the linear programming induced by $1|prec, \tilde{p}, resource, \vec{d}| \bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}max + \sum_{i=1}^n v_i u_i)$ according to (9) as follows:

$$\begin{aligned}
 \min \quad & \sum_{i=1}^n \left\{ h(e_i, t_i, \lambda, k_i) \frac{\sigma_{\bar{C}_i}}{\sqrt{\frac{\lambda^2 \bar{C}_i^2}{4} + \frac{\lambda \bar{C}_i}{3} + \frac{1}{6}}} \right\} \\
 \text{s.t.} \quad & \sigma_{\bar{C}_i} \geq \sigma_{\bar{p}_i} \text{ for } i = 1, \dots, n; \\
 & \sigma_{\bar{C}_k} \geq \sigma_{\bar{C}_i} + \sigma_{\bar{p}_k} \text{ for each arc } J_i J_k \text{ in } G; \\
 & \sum_{J_j \in S} \sigma_{\bar{p}_j} \sigma_{\bar{C}_j} \geq \frac{1}{2} (\sigma_{\bar{p}}^2(S) + \sigma_{\bar{p}}(S)^2) \text{ for each } S \subset \mathcal{J}.
 \end{aligned}$$

Solve the linear programming above and obtain the solution $(\bar{\sigma}_{\bar{C}_1}, \bar{\sigma}_{\bar{C}_2}, \dots, \bar{\sigma}_{\bar{C}_n})$.

Step 5: Schedule the jobs according to the nondecreasing order of $\bar{\sigma}_{\tilde{C}_j}$, where ties are broken by choosing an order that is consistent with the precedence constraints G . This schedule is denoted as π .

Step 6: Compute the completion time \tilde{C}_i of J_i in the schedule π obtained in step 5, then assign the due date d_i to J_i by $d_i = \bar{M}(\tilde{C}_i) + \frac{k_i - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}_i}$ ($i = 1, \dots, n$).

Step 7: Return π , $\vec{d} = (d_1, d_2, \dots, d_n)$, $\vec{u} = (u_1, u_2, \dots, u_n)$ and $\vec{k} = (k_1, k_2, \dots, k_n)$.

Theorem 5.1. *The worst case ratio of the approximation algorithm above is 2 for $1|prec, \bar{p}, resource, \vec{d}$ $\bar{M}(\sum_{i=1}^n (e_i \tilde{E}_i + t_i \tilde{T}_i) + \delta \tilde{C}_{max} + \sum_{i=1}^n v_i u_i)$ with $\lambda_{\bar{p}_i} = (b_i - a_i)/\alpha_i = \lambda$ for the basic processing time $\tilde{p}_i = (a_i, b_i, \alpha_i)$ of J_i ($i = 1, \dots, n$).*

Proof. Let the schedule $\pi = \{J_{\pi(1)}, J_{\pi(2)}, \dots, J_{\pi(n)}\}$ with \vec{d} , \vec{u} and \vec{k} be obtained by the approximation algorithm above, and $(\bar{\sigma}_{\tilde{C}_1}, \bar{\sigma}_{\tilde{C}_2}, \dots, \bar{\sigma}_{\tilde{C}_n})$ be a solution of the linear programming of the step 4. It follows that $\sum_{J_j \in S} \sigma_{\tilde{p}_j} \bar{\sigma}_{\tilde{C}_j} \geq \frac{1}{2}(\sigma_{\tilde{p}}^2(S) + \sigma_{\tilde{p}}(S)^2)$ for each $S \subset \mathcal{J}$. Additionally, by the step 5, we have $\bar{\sigma}_{\tilde{C}_{\pi(1)}} \leq \bar{\sigma}_{\tilde{C}_{\pi(2)}} \leq \dots \leq \bar{\sigma}_{\tilde{C}_{\pi(n)}}$. Hence, for the set $S = \{J_{\pi(1)}, J_{\pi(2)}, \dots, J_{\pi(i)}\} \subset \mathcal{J}$ ($i = 1, 2, \dots, n$), we have

$$\sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}} \bar{\sigma}_{\tilde{C}_{\pi(j)}} \geq \frac{1}{2}(\sigma_{\tilde{p}}^2(S) + \sigma_{\tilde{p}}(S)^2) \geq \frac{1}{2}\sigma_{\tilde{p}}(S)^2 = \frac{1}{2}\left(\sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}}\right)^2.$$

It follows that $\bar{\sigma}_{\tilde{C}_{\pi(i)}} \sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}} \geq \sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}} \bar{\sigma}_{\tilde{C}_{\pi(j)}} \geq \frac{1}{2}\left(\sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}}\right)^2$, which implies that, for any $i \in \{1, 2, \dots, n\}$,

$$\sigma_{\tilde{C}_{\pi(i)}} = \sum_{j=1}^i \sigma_{\tilde{p}_{\pi(j)}} \leq 2\bar{\sigma}_{\tilde{C}_{\pi(i)}}.$$

Then, we have

$$\begin{aligned} f(\pi, \vec{d}, \vec{u}) &= \sum_{i=1}^n \frac{h(e_{\pi(i)}, t_{\pi(i)}, \lambda, k_{\pi(i)}) \sigma_{\tilde{C}_{\pi(i)}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\ &\leq \sum_{i=1}^n \frac{2h(e_{\pi(i)}, t_{\pi(i)}, \lambda, k_{\pi(i)}) \bar{\sigma}_{\tilde{C}_{\pi(i)}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \\ &\leq 2 \left[\sum_{i=1}^n \frac{h(e_{\pi(i)}, t_{\pi(i)}, \lambda, k_{\pi(i)}) \bar{\sigma}_{\tilde{C}_{\pi(i)}}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \right] \\ &= 2 \left[\sum_{i=1}^n \frac{h(e_i, t_i, \lambda, k_i) \bar{\sigma}_{\tilde{C}_i}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \right]. \end{aligned}$$

On the other hand, for the optimal schedule $\pi^* = \{J_{\pi^*(1)}, J_{\pi^*(2)}, \dots, J_{\pi^*(n)}\}$ with \vec{d}^* , u^* and \vec{k}^* , Lemma 3.1 implies that $\vec{k}^* = \vec{k}$, and u^* is a solution of the linear programming in the step 2, which means that

$$\delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i^* = \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i.$$

Also, $(\sigma_{\tilde{C}_1^*}, \dots, \sigma_{\tilde{C}_n^*})$ satisfies $\sum_{J_j \in S} \sigma_{\tilde{p}_j} \sigma_{\tilde{C}_j^*} \geq \frac{1}{2}(\sigma_{\tilde{p}}^2(S) + \sigma_{\tilde{p}}(S)^2)$ for each $S \subset \mathcal{J}$, which implies that $(\sigma_{\tilde{C}_1^*}, \dots, \sigma_{\tilde{C}_n^*})$ is in the feasible domain of the linear programming of step 4. It follows that

$$\sum_{i=1}^n \left\{ h(e_i, t_i, \lambda, k_i) \frac{\bar{\sigma}_{\tilde{C}_i}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \right\} \leq \sum_{i=1}^n \left\{ h(e_i, t_i, \lambda, k_i^*) \frac{\sigma_{\tilde{C}_i^*}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \right\}.$$

Then, we have

$$\begin{aligned} f(\pi, \vec{d}, \vec{u}) &\leq 2 \left[\sum_{i=1}^n \frac{h(e_i, t_i, \lambda, k_i) \bar{\sigma}_{\tilde{C}_i}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i \right] \\ &\leq 2 \left[\sum_{i=1}^n \frac{h(e_i, t_i, \lambda, k_i^*) \sigma_{\tilde{C}_i^*}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} + \delta \sum_{i=1}^n \frac{a_i + b_i}{2} + \sum_{i=1}^n (v_i - \delta s_i) u_i^* \right] \\ &= 2f(\pi^*, \vec{d}^*, \vec{u}^*). \end{aligned}$$

□

Next, an example is given to show how the approximation algorithm above works.

Example 5.2. Let $\mathcal{J} = \{J_1, \dots, J_{12}\}$ with the basic processing times $\tilde{p}_1 = (3, 4, 2)$, $\tilde{p}_2 = (7, 10, 6)$, $\tilde{p}_3 = (5, 7, 4)$, $\tilde{p}_4 = (9, 13, 8)$, $\tilde{p}_5 = (4, 5, 2)$, $\tilde{p}_6 = (8, 11, 6)$, $\tilde{p}_7 = (6, 8, 4)$, $\tilde{p}_8 = (10, 14, 8)$, $\tilde{p}_9 = (7, 9, 4)$, $\tilde{p}_{10} = (9, 12, 6)$, $\tilde{p}_{11} = (11, 15, 8)$ and $\tilde{p}_{12} = (5, 6, 2)$; the per unit earliness penalty vector for jobs is $(e_1, \dots, e_{12}) = (6, 7, 3, 8, 4, 11, 9, 6, 3, 5, 3, 8)$; the per unit tardiness penalty vector (t_1, \dots, t_{12}) for jobs is $(5, 9, 5, 7, 8, 14, 4, 5, 8, 6, 9, 3)$; the total amount of resources is $U_0 = 50$; the cost of one unit of operation time is $\delta = 2$; the compression rate vector of jobs is $(s_1, \dots, s_{12}) = (0.1, 0.3, 0.2, 0.4, 0.1, 0.3, 0.2, 0.4, 0.2, 0.3, 0.4, 0.1)$; the cost vector of one unit of resources is $(v_1, \dots, v_{12}) = (0.5, 0.7, 0.3, 0.2, 0.1, 0.8, 0.9, 0.5, 0.3, 0.5, 0.9, 1)$; and the precedence constraints between jobs are described as Figure 1.

First, according to the step 2 of the approximation algorithm above, find a solution of the linear programming of the step 2 induced by the problem about the amounts of the resources allocated to jobs as follows:

$$(u_1, \dots, u_{12}) = (0, 0, 3.463, 2.500, 17.318, 0, 0, 5.000, 12.503, 9.216, 0, 0).$$

Then compute the actual processing times for jobs by the resource consumption function $\tilde{p}_i(u_i) = \tilde{p}_i - s_i u_i$ ($i = 1, \dots, 12$), and we get $\tilde{p}_1 = (3, 4, 2)$, $\tilde{p}_2 = (7, 10, 6)$, $\tilde{p}_3 = (4.307, 6.307, 4)$, $\tilde{p}_4 = (8, 12, 8)$, $\tilde{p}_5 = (2.268, 3.268, 2)$, $\tilde{p}_6 = (8, 11, 6)$, $\tilde{p}_7 = (6, 8, 4)$, $\tilde{p}_8 = (8, 12, 8)$, $\tilde{p}_9 = (4.499, 6.499, 4)$, $\tilde{p}_{10} = (6.235, 9.235, 6)$, $\tilde{p}_{11} = (11, 15, 8)$ and $\tilde{p}_{12} = (5, 6, 2)$.

Next, compute the optimal customer service level for each job and find a solution of the linear programming of step 4 in the approximation algorithm; the solution $(\bar{\sigma}_{\tilde{C}_1}, \dots, \bar{\sigma}_{\tilde{C}_{12}})$ is $(6.501, 29.778, 37.749, 5.243, 6.501, 16.987, 19.504, 26.005, 28.522, 35.233, 13.212, 14.471)$. Next, scheduling the jobs according to the nondecreasing order of $\bar{\sigma}_{\tilde{C}_j}$, and we get the schedule

$$\pi = (J_4, J_5, J_1, J_{11}, J_{12}, J_6, J_7, J_8, J_9, J_2, J_{10}, J_3).$$

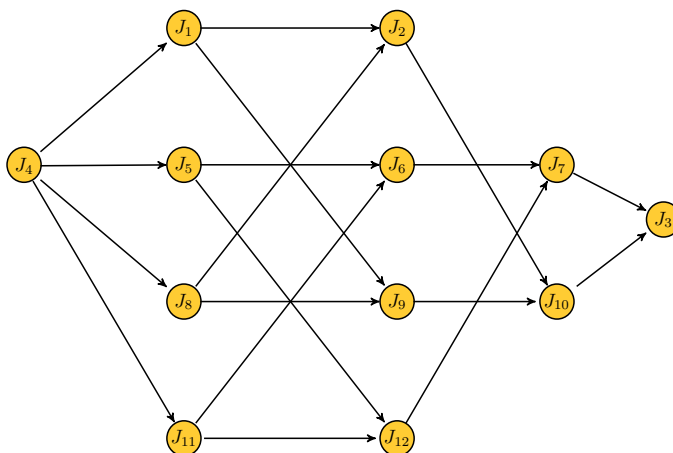


FIGURE 1. The Precedence Constraints for Example 5.2

Finally, compute the completion times for jobs in the schedule π and assign the due date to each job by $d_i = \bar{M}(\tilde{C}_i) + \frac{k_i - \frac{\lambda}{2}}{\sqrt{\frac{\lambda^2}{4} + \frac{\lambda}{3} + \frac{1}{6}}} \sigma_{\tilde{C}_i}$ ($i = 1, \dots, 12$). Then, the due date vector $(d_1, d_2, \dots, d_{12})$ is $(12.710, 90.997, 111.349, 7.729, 17.103, 53.002, 36.371, 49.407, 89.272, 99.609, 40.126, 23.516)$.

Remark 5.3. Note that the constraints of the linear programming (9) induced by the problem is the exponentially large class of constraints. However, as it is pointed in [22], an effective ellipsoid algorithm based on a polynomial-time separation algorithm is given for solving the linear programming with the exponentially large class of constraints. Hence, the solution for the example above can be quickly obtained even the size of the constraints matrix is 4113×12 .

6. Experiment

In [21, 16], two methods are put forward to predict the due date for each job with the uncertain processing time. In [21], a due date assignment model in stochastic environment is proposed in which the job's processing time is assumed as a random variable with normal distribution; the due date for each job is given by $d = E(C) + k\sigma_C$ where $E(C)$ is the mean value of the completion time C , σ_C is the standard variance of the completion time C , and k is the value such that $\Phi(k) = \frac{t}{e+t}$ in which $\Phi(x)$ is the standard normal cumulative distribution function, e is the per unit earliness penalties, and t is the per unit tardiness penalties. Besides, in [16], the due date assignment model in fuzzy environment is given in which the symmetric triangular fuzzy numbers are used to model the uncertain processing times; the due date is assigned to each job by $d = a + k\alpha$ with respect to the completion time (a, α) where k is determined by e and t . Without special statement, the method in [21], the method in [16] and the method of our paper are briefly written as method 1, method 2, and method 3 in the following.

Next, the three methods above will be tested on the same data coming from Example 1 [21] to examine their effectivity by comparing the total costs for the same data, which can be denoted by the the flow diagram in Figure 2.

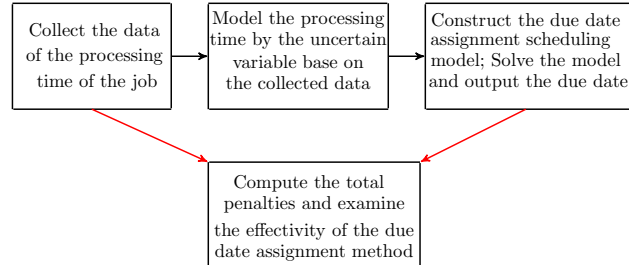


FIGURE 2. The Flow Diagram for Examining the Effectivity of the Due Date Assignment Method

In Example 1 [21], the processing time of the job is the random variable with normal distribution $N(10, 3^2)$; the per unit earliness penalties and tardiness penalties are $e = 3\sqrt{2\pi}$ and $t = 27\sqrt{2\pi}$. Next, we produce 6 samples from the normal distribution $N(10, 3^2)$ same as ones in [16], whose sample sizes are 46, 81, 180, 344, 666 and 1332, respectively. The sampling method for each sample is as follows. First, according to the 3σ principle, determine the interval which the processing time of the job is almost in. It is easy to get that the interval is [1,19]. Second, divide the interval [1,19] into 6 subintervals as [1, 4], [4, 7], [7, 10], [10, 13], [13, 16] and [16, 19]. At last, choose processing times from 6 subintervals accord with the probability of the random variable $N(10, 3^2)$ in each interval.

Next, model the uncertain processing time of the job by the uncertain variable based on the collected data and determine the due dates of the job by use of the three methods. For method 1 [21], the uncertain processing time of the job is modelled by the random variable with normal distribution $N(10, 3^2)$ for each sample, and the due date for the job determined by method 1 is 13.8460; for method 2 [16], the uncertain processing time of this job is modelled by the symmetrical triangle fuzzy number (10, 8.5) for each sample, and the due date determined according to the method 2 is 14.6987 [16]; for method 3, we use the symmetric trapezoidal fuzzy number (10, 10.8095, 8.0952) to model the uncertain processing time for each sample, and the due date for this job is 14.4749 according to the method of our paper.

Then, compute the earliness-tardiness penalties of all jobs in each sample, which are are shown in Figure 3. Compute the total penalties of all jobs in each sample for these three methods, and compute the ratios of the total penalties of jobs for method 3 and the ones for other methods which is given in Table 1 with $r_{i3} = \frac{\text{the total penalties of jobs for method 3}}{\text{the total penalties of jobs for method } i}$ for each sample ($i = 1, 2$); Known from ratios of Table 1, for the same data above, the method proposed in this paper is better than the other two methods.

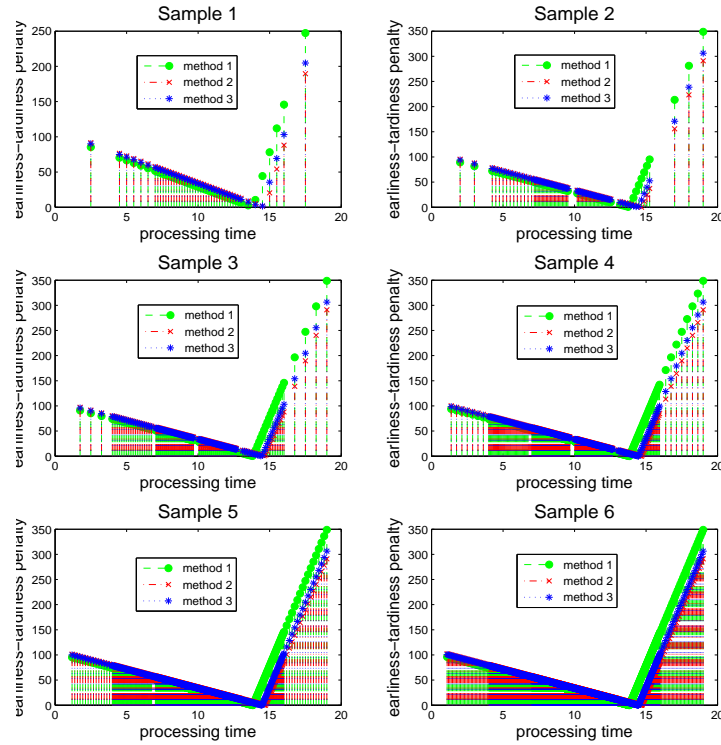


FIGURE 3. The Earliness-Tardiness Penalties of Jobs in Samples1-6

| | Sample 1 46 jobs | Sample 2 81 jobs | Sample 3 180 jobs | Sample 4 344 jobs | Sample 5 666 jobs | Sample 6 1332 jobs |
|----------|---------------------|---------------------|----------------------|----------------------|----------------------|-----------------------|
| r_{13} | 0.9847 | 0.9844 | 0.9788 | 0.9724 | 0.9888 | 0.9921 |
| r_{23} | 0.9888 | 0.9891 | 0.9794 | 0.9725 | 0.9933 | 0.9967 |

TABLE 1. The Ratios of the Total Penalties of the Jobs in Each Sample for Three Methods

Further, according to the approximation algorithm in section 5, our model can deal with the due date assignment problems with the precedence constraints and controllable processing times in uncertain environment, whereas there are no results for this case by the other two ones [16, 21] as far as we know. Moreover, we use

the Example 5.2 to test the approximation algorithm in section 5. All algorithms are implemented using MATLAB language (no explicit use of parallel toolbox), tested in MATLAB 2014a, 8GB memory and Intel core i7-4790 CPU. Then, the schedule with the objective function value 1407.14 is obtained by the approximation algorithm of section 5 in 1 second. On the other side, the optimal schedule $\pi^* = (J_4, J_1, J_5, J_{11}, J_6, J_{12}, J_7, J_8, J_2, J_9, J_{10}, J_3)$ with the objective function value 1402.64 can be obtained by enumeration, but the time is about 1271 seconds due to examining 3628800 permutations. Moreover, the ratio of two objective function values is $\frac{1407.14}{1402.64} = 1.003$, which shows that the approximation algorithm in section 5 is effective.

7. Conclusions

In this paper, we study a due date assignment scheduling problem with precedence constraints and controllable processing times in fuzzy environment, in which the basic processing time of each job is assumed to be the symmetric trapezoidal fuzzy number, and the linear resource consumption function is used. First, the condition for the optimal due date with respect to the completion is given. Then, the properties of the optimal schedule are obtained. Based on these properties, we show that the problem is NP-hard. Furthermore, we identify a polynomially solvable case of the problem. For the general cases, we develop an efficient approximation algorithm based on solving the relaxation of the problem, whose worst case ratio is show to be 2.

However, the approximation algorithm in section 5 is required to solve a kind of linear programming with exponentially large class of constraints. Our work may accordingly focus on designing more effective algorithm for the due date assignment scheduling with general precedence constraints and controllable processing times in uncertain environment.

Acknowledgements. This work was supported by National Natural Science Foundation of China (11401030), Outstanding Young Teacher Training Program of Colleges and Universities in Guangdong Province (2014) and Teacher Research Capacity Promotion Program of Beijing Normal University Zhuhai.

REFERENCES

- [1] L.D. Adolphson, *Single machine job sequencing with precedence constraints*, SIAM Journal on Computing, **6(1)** (1977), 40–54.
- [2] R. Alvarez-Valdes, J.M. Tamarit and F. Villa, *Minimizing weighted earliness-tardiness on parallel machines using hybrid metaheuristics*, Computers & Operations Research, **54** (2015), 1–11.
- [3] P. Assarzadegan and M. Rasti-Barzoki, *Minimizing sum of the due date assignment costs, maximum tardiness and distribution costs in a supply chain scheduling problem*, Applied Soft Computing, **47** (2016), 343–356.
- [4] K.R. Baker, *Minimizing earliness and tardiness costs in stochastic scheduling*, European Journal of Operational Research, **236(2)** (2014), 445–452.
- [5] C. Carlsson and R. Fullér, *On possibilistic mean value and variance of fuzzy numbers*, Fuzzy Sets and Systems, **122(2)** (2001), 315–326.

- [6] T.C.E. Cheng, *Optimal assignment of slack due-dates and sequencing of jobs with random processing times on a single machine*, European Journal of Operational Research, **51(3)** (1991), 348–353.
- [7] T.C.E. Cheng, C. Oğuz and X. Qi, *Due-date assignment and single machine scheduling with compressible processing times*, International Journal of Production Economics, **43(2-3)** (1996), 107–113.
- [8] D. Du, Ker-I Ko and X. Hu, *Design and Analysis of Approximation Algorithms*, Springer Press, New York, 2012.
- [9] R. Fullér and P. Majlender, *On weighted possibilistic mean and variance of fuzzy numbers*, Fuzzy Sets and Systems, **136(3)** (2003), 363–374.
- [10] 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, **139(1)** (2002), 1–25.
- [11] 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, **5** (1979), 287–326.
- [12] L.A. Hall, A.S. Schulz, D.B. Shmoys and J. Wein, *Scheduling to Minimize Average Completion Time: Off-Line and On-Line Approximation Algorithms*, Mathematics of Operations Research, **22(3)** (1997), 513–544.
- [13] M. Hanss, *Applied Fuzzy Arithmetic: An Introduction with Engineering Applications*, Springer Press, Berlin, 2005.
- [14] M. Iranpoor, S.M.T. FatemiGhomi and M. Zandieh, *Due-date assignment and machine scheduling in a low machine-rate situation with stochastic processing times*, Computers & Operations Research, **40(4)** (2013), 1100–1108.
- [15] M. Ji, J.J. Ge, K. Chen and T.C.E. Cheng, *Single-machine due-window assignment and scheduling with resource allocation, aging effect, and a deteriorating rate-modifying activity*, Computers & Industrial Engineering, **66(4)** (2013), 952–961.
- [16] J. Li, K. Sun, D. Xu and H. Li, *Single machine due date assignment scheduling problem with customer service level in fuzzy environment*, Applied Soft Computing, **10(3)** (2010), 849–858.
- [17] J. Li, X. Yuan, E.S. Lee and D. Xu, *Setting due dates to minimize the total weighted possibilistic mean value of the weighted earliness-tardiness costs on a single machine*, Computers & Mathematics with Applications, **62(11)** (2011), 4126–4139.
- [18] L. Liu, J. Wang, F. Liu and M. Liu, *Single machine due window assignment and resource allocation scheduling problems with learning and general positional effects*, Journal of Manufacturing Systems, **43(1)** (2017), 1–14.
- [19] L. Liu, J. Wang and X. Wang, *Single machine due-window assignment scheduling with resource-dependent processing times to minimize total resource consumption cost*, International Journal of Production Research, **54(4)** (2016), 1–10.
- [20] I.N. Lushchakova, *Two machine preemptive scheduling problem with release dates, equal processing times and precedence constraints*, European Journal of Operational Research, **171(1)** (2006), 107–122.
- [21] V. Portougal and D. Trietsch, *Setting due dates in a stochastic single machine environment*, Computers & Operations Research, **33(6)** (2006), 1681–1694.
- [22] M. Queyranne, *Structure of a simple scheduling polyhedron*, Mathematical Programming, **58(1-3)** (1993), 263–285.
- [23] M. Rasti-Barzoki and S. Hejazi, *Pseudo-polynomial dynamic programming for an integrated due date assignment, resource allocation, production, and distribution scheduling model in supply chain scheduling*, Applied Mathematical Modelling, **39(12)** (2015), 3280–3289.
- [24] D. Shabtay, *Due date assignments and scheduling a single machine with a general earliness/tardiness cost function*, Computers & Operations Research, **35(5)** (2008), 1539–1545.
- [25] D. Shabtay, Y. Itskovich, L. Yedidsion and D. Oron, *Optimal due date assignment and resource allocation in a group technology scheduling environment*, Computers & Operations Research, **37(12)** (2010), 2218–2228.

- [26] D. Shabtay and G. Steiner, *A survey of scheduling with controllable processing times*, Discrete Applied Mathematics, **155(13)** (2007), 1643–1666.
- [27] D. Shabtay, G. Steiner, *The single-machine earliness-tardiness scheduling problem with due date assignment and resource-dependent processing times*, Annals of Operations Research, **159(1)**(2008), 25–40.
- [28] D. Shabtay and G. Steiner, *Two due date assignment problems in scheduling a single machine*, Operations Research Letters, **34(6)** (2006), 683–691.
- [29] D. Shabtay, G. Steiner and R. Zhang, *Optimal coordination of resource allocation, due date assignment and scheduling decisions*, Omega, **65** (2016), 41–54.
- [30] H. Soroush, *Sequencing and due-date determination in the stochastic single machine problem with earliness and tardiness costs*, European Journal of Operational Research, **113(2)** (1999), 450–468.
- [31] N. H. Tuong and A. Soukha, *Due dates assignment and JIT scheduling with equal-size jobs*, European Journal of Operational Research, **205(2)** (2010), 280–289.
- [32] D.L. Yang, C.J. Lai and S.J. Yang, *Scheduling problems with multiple due windows assignment and controllable processing times on a single machine*, International Journal of Production Economics, **150** (2014), 96–103.
- [33] Y. Yin, T.C.E. Cheng, C.C. Wu and S.R. Cheng, *Single-machine due window assignment and scheduling with a common flow allowance and controllable job processing time*, Journal of the Operational Research Society, **65(1)** (2014), 1–13.
- [34] Y. Yin, T.C.E. Cheng, D. Xu and C. Wu, *Common due date assignment and scheduling with a rate-modifying activity to minimize the due date, earliness, tardiness, holding and batch delivery cost*, Computers & Industrial Engineering, **63(1)** (2012), 223–234.
- [35] Q. Yue and G. Wan, *Order schedule with controllable processing times, common due date and processing deadline*, Journal of Systems Science and Systems Engineering, **26(2)** (2017), 199–218.
- [36] C. Zhao and H. Tang, *Two-machine flow shop scheduling with deteriorating jobs and chain precedence constraints*, International Journal of Production Economics, **136(1)** (2012), 131–136.

JINQUAN LI*, SCHOOL OF APPLIED MATHEMATICS, ZHUHAI MUNICIPAL KEY LABORATORY OF INTELLIGENT CONTROL, BEIJING NORMAL UNIVERSITY ZHUHAI, ZHUHAI 519087, P.R. CHINA
E-mail address: lj_q_1023@mail.bnu.edu.cn

DEHUA XU, SCHOOL OF SCIENCE, EAST CHINA UNIVERSITY OF TECHNOLOGY, NANCHANG 330013, P.R. CHINA
E-mail address: dhxu@ecut.edu.cn

HONGXING LI, SCHOOL OF ELECTRONIC AND INFORMATION ENGINEERING, DALIAN UNIVERSITY OF TECHNOLOGY, DALIAN 116024, P.R. CHINA
E-mail address: lihx@dlut.edu.cn

*CORRESPONDING AUTHOR