Int J Adv Manuf Technol (2011) 56:1125–1138
DOI 10.1007/s00170-011-3244-3

ORIGINAL ARTICLE

A novel objective function for job-shop scheduling problem with fuzzy processing time and fuzzy due date using differential evolution algorithm

Yanmei Hu · Minghao Yin · Xiangtao Li

Received: 16 August 2010 / Accepted: 27 February 2011 / Published online: 5 April 2011

Abstract The job-shop scheduling problems with fuzzy processing time and fuzzy due date are investigated in this paper. The ranking concept among fuzzy numbers based on possibility and necessity measures which are developed in fuzzy sets theory is introduced. And on the basis of two consistent measures in this concept, several novel objective functions are proposed. The purpose of our research is to obtain the optimal schedules based on these objective functions. A modified DE algorithm will be designed to solve these objective functions. Several jop-shop scheduling problems with fuzzy processing time and fuzzy due date are experimented to show the efficiency and comparability of our approach. Through the experimental results, the potential application of the possibility and necessity theory in the real world is shown.

Keywords Fuzzy job-shop scheduling · Fuzzy processing time · Fuzzy due date · Differential evaluation algorithm · Ranking concept

1 Introduction

The job-shop scheduling problem, abbreviated as JSSP, is well-known as one of the popular problem in scheduling, which plays an important role not only in manufacturing systems, but also in industrial process for improving resource utilization. Since JSSP was firstly formulated, many approximation methods have been proposed, such as priority dispatch rules [1, 2], constraint satisfaction approaches [3, 4], and local search methods, since it belongs to the class of decision problems which are NP-hard. Among those methods, local search methods have been of concern all the time; many researches have been proposed, such as [5–8]. The genetic algorithm (GA) was applied as the search algorithm in all of these papers. Stated in a simple way, in all these JSSP problems, various factors, such as processing time and due date and so on, have precisely been fixed at some crisp values.

However, in the real world, when it comes to scheduling problems, most factors involved in are often only imprecisely or ambiguously known. For instance, the processing time of a job often can't be measured precisely and sometimes it's tolerable that the completion of a job timeouts in a certain degree. In such situations, it is more reasonable to consider the fuzziness. For example, "the processing time is around 5 h" may be a more reasonable statement. It is more often that the phrase "around 5 h" is modeled as a fuzzy number $\widetilde{5}$. And those scheduling problems with fuzzy factors are recognized as the so-called fuzzy scheduling problems [9].

The scheduling problems with fuzzy factors have been investigated by many researchers since it was firstly proposed by Ishii et al. [10]. However, there are not many results obtained for fuzzy job-shop scheduling problems. Sakawa and Mori [11] presented an efficient GA by incorporating the concept of similarity among individuals to search the best schedule with fuzzy processing time and fuzzy due date, and also compared it with the simulated annealing algorithm. Next, Sakawa and Kubota [12] applied the same GA to solve multi-objective fuzzy job scheduling problem (FJSSP). Li et al. [13] proposed a GA to solve FJSSP with alternative machines by adopting two-chromosome presentation and the extended Giffer–Thompson Procedure. Lei [14] proposed an efficient Pareto archive particles swarm optimization. It has two phases. Firstly, FJSSP is converted into a continuous optimization problem, and then the proposed

Y. Hu · M. Yin (✉) · X. Li
College of Computer Science, Northeast Normal University, Changchun 130117, People's Republic of China
e-mail: ymh@nenu.edu.cnymh@nenu.edu.cn

algorithm is used to search a set of Pareto-optimal solutions of the continuous problems based on three objectives. Those papers all use the concept of agreement index to formulate the objective functions. Moreover, Lei also considered other measures, including maximize fuzzy completion time and the mean fuzzy completion time.

Under these circumstances, in this paper, we propose a different approach by using the concept of possibility and necessity to formulate the objective functions. In Itoh and Ishii [15], possibility measure was firstly introduced to the fuzzy job-shop scheduling problems. They use the possibility to measure the tardiness of each job, and then to search a schedule which minimizes the number of λ–tardy jobs. The approach in this paper is totally different from that of [15]. In order to explore more suitable model, here we adopt the ranking concept of fuzzy numbers and uses the possibility and necessity measures that were proposed by Dubios and Prade [16]. For making the results consistent, we are going to formulate the objective functions based on two indices, which are consistent with each other, that are certainly different from that in other papers [11, 15, 17, 18]. At the same time, we define a kind of special fuzzy number as the supposed type of fuzzy processing time and fuzzy due date, and then we also derive the analytic formulas for these objective functions for simplifying the computation. Furthermore, we design a differential evolution (DE) algorithm to solve these objective functions.

In Section 2, we introduce the concept of fuzzy numbers, and some arithmetics among fuzzy numbers, which are based on the "Extension Principle" in fuzzy sets theory. In Section 3, we introduce the fuzzy job-shop scheduling problem, and present several types of objective functions, which are consistent, based on the possibility and necessity measures. Also, a special type of fuzzy number is defined, and the analytic formulas based on it for these objective functions are derived in order to simplify the computation. In Section 4, a modified DE algorithm is designed to solve this fuzzy job-shop scheduling problem. In Section 5, some numerical examples are provided and solved to evaluate our discussion and to show that there is a potential future to apply the possibility and necessity measures to the fuzzy job-shop scheduling problem. In the final section, the conclusions are given.

2 Fuzzy number

Let X be a set of points, with a generic element of X denoted by x, thus $X=\{x\}$. From [19], we can define a fuzzy set $\widetilde{A}$ in X as follows:

Define 1 If $\mu_{\widetilde{A}}$ is a map from X to [0, 1], that is a function $\mu_{\widetilde{A}} : X \to [0,1]$, then $\widetilde{A}$ is a fuzzy subset of X.

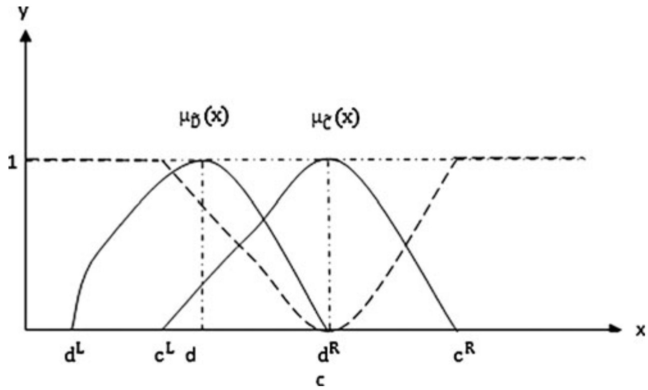

Fig. 1 $d_R \le c$ with $\mathrm{PSD}\left(\widetilde{D},\widetilde{C}\right) = 0$

From definition 1, it's easy to know that $\widetilde{A}$ is characterized by a membership function $\mu_{\widetilde{A}}(x)$ which associates with each point in X a real number in [0, 1], with the value of $\mu_{\widetilde{A}}(x)$ at x representing the "grade of membership" of x in $\widetilde{A}$. Here, we can conclude that the fuzzy subset $\widetilde{A}$ of R is defined by a membership function $\mu_{\widetilde{A}} : R \to [0,1]$. Let $\widetilde{A}_a$ denote the α-level set of $\widetilde{A}$, then $\widetilde{A}_\alpha = \left\{x \in R : \mu_{\widetilde{A}}(x) \ge \alpha\right\}$. The 0-level set $\widetilde{A}_0 = \left\{x \in R : \mu_{\widetilde{A}}(x) \ge 0\right\}$ is the closure of the union of all level sets, i.e. $\widetilde{A}_0 = cl\left(\bigcup_{\alpha>0}\widetilde{A}_\alpha\right)$. A fuzzy set $\widetilde{A}$ of R is said to be a fuzzy number if it satisfy the conditions below:

1. $\widetilde{A}$ is normal, i.e., there exits an $x \in R$ such that $\mu_{\widetilde{A}}(x) = 1$;
2. There exits an $x \in R$ satisfies:
   (a) $\forall x_1 \le x_2 \le x, \mu_{\widetilde{A}}(x_1) \le \mu_{\widetilde{A}}(x_2) \le \mu_{\widetilde{A}}(x)$;
   (b) $\forall x_1 \ge x_2 \ge x, \mu_{\widetilde{A}}(x) \le \mu_{\widetilde{A}}(x_2) \le \mu_{\widetilde{A}}(x_1)$;
3. $\forall x_1 \le x_0 \le x_2 \in R$, $\mu_{\widetilde{A}}(x_0) \ge \min\left\{\mu_{\widetilde{A}}(x_1), \mu_{\widetilde{A}}(x_2)\right\}$;
4. $\forall \alpha \in (0,1)$, $\widetilde{A}_\alpha = \left\{x \in R : \mu_{\widetilde{A}}(x) \ge \alpha\right\}$ is a closed and bounded subset of R.

Since condition d, we see that if $\widetilde{A}$ is a fuzzy number of R, then $\widetilde{A}_a$ is a closed and bounded interval of R. Therefore, we can denote it with $\left[\widetilde{A}^L_\alpha, \widetilde{A}^U_\alpha\right]$.

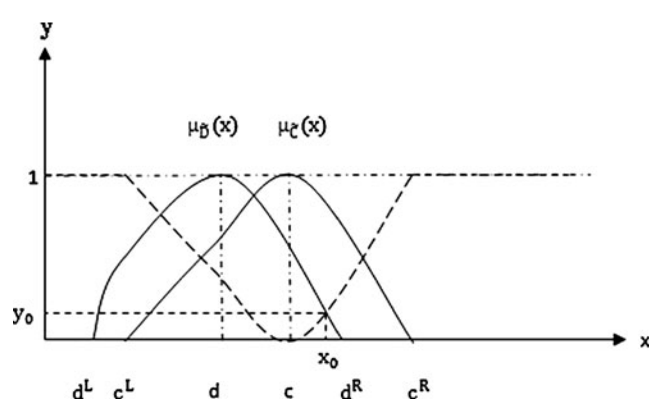

Fig. 2 $d < c_R$ and $d_R < c$ with $\mathrm{PSD}\left(\widetilde{D},\widetilde{C}\right) = y_0$

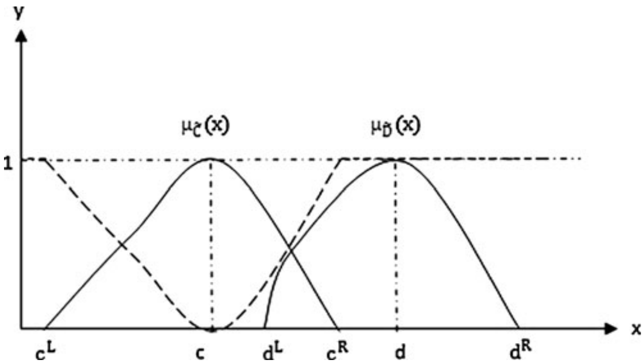

Fig. 3 $d \geq c_R$ with $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = 1$

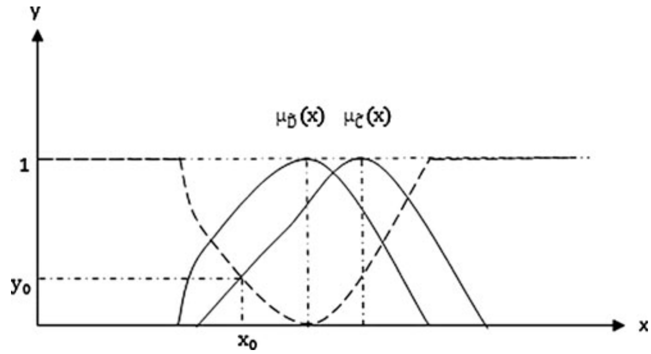

Fig. 5 $c_L < d$ with $c > d_L$ with $\mathrm{ND}\left(\widetilde{D}, \widetilde{C}\right) = y_0$

Using the Extension Principle in Zadeh [19], we obtain that if $\widetilde{A}$ and $\widetilde{B}$ are two fuzzy numbers, then the membership function of $\widetilde{A} \oplus \widetilde{B}$ is defined by

$$\mu_{\widetilde{A} \oplus \widetilde{B}}(z) = \sup_{x+y=z} \min\left\{\mu_{\widetilde{A}}(x), \mu_{\widetilde{B}}(y)\right\}$$

the membership function of the maximum $m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}$ of $\widetilde{A}$ and $\widetilde{B}$ is defined by

$$\mu_{m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}} = \sup_{z=\max(x,y)} \min\left\{\mu_{\widetilde{A}}(x), \mu_{\widetilde{B}}(y)\right\}$$

Then we also have the following well-known results.

Proposition 1 Let $\widetilde{A}$ and $\widetilde{B}$ be two fuzzy numbers, then $\widetilde{A} \oplus \widetilde{B}$ and $m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}$ are also fuzzy numbers [17]. Furthermore, we have

$$\left(\widetilde{A} \oplus \widetilde{B}\right)_\alpha = \left[\widetilde{A}_\alpha^L + \widetilde{B}_\alpha^L, \widetilde{A}_\alpha^U + \widetilde{B}_\alpha^U\right],$$
$$\left(m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}\right)_\alpha = \left[m\widetilde{a}x\left(\widetilde{A}_\alpha^L, \widetilde{B}_\alpha^L\right), \max\left(\widetilde{A}_\alpha^U, \widetilde{B}_\alpha^U\right)\right].$$

In the next section, we will introduce a specified fuzzy number to simplify our research. Obviously, it satisfies all the above properties.

3 JSSP with fuzzy processing time and fuzzy due date

3.1 JSSP problem

In general, an $n \times m$ job-shop scheduling problem is formulated as follows. Let n jobs $J_j (j \in (1, 2, \ldots, n))$ be processed on m machines $M_r (r = 1, 2, \ldots, m)$, and let the operation of job J_j on machine M_r be $O_{j,i,r}$, where $i \in (1, 2, \ldots, m)$ shows the position of the operation in the technological sequence of the job. In other words, Oj,i,r expresses the i_{th} operation of job J_j processed on machine r. It is assumed here that only one operation can be processed on each machine at a time, and that each operation cannot be started if the previous operation is still being processed.

However, different from the classical $n \times m$ job-shop scheduling problem, in this paper, we formulate a job-shop scheduling problem with fuzzy processing time and fuzzy due date, called fuzzy job scheduling problem. Here, in the FJSSP, the processing time and due date of operation $O_{j,i,r}$ are represented by fuzzy numbers, denoted as $\widetilde{P}_{J_j i}$ and $\widetilde{D}_{J_j}$, respectively. For notational convenience, in the following, we denote the FJSSP of n jobs and m machines as $n \times m$ FJSSP.

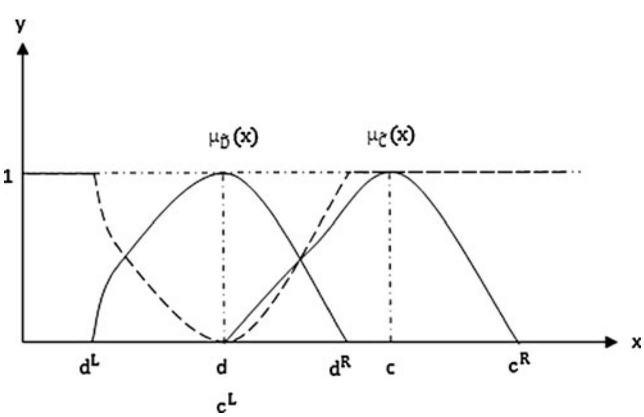

Fig. 4 $c_L \geq d$ with $\mathrm{ND}\left(\widetilde{D}, \widetilde{C}\right) = 0$

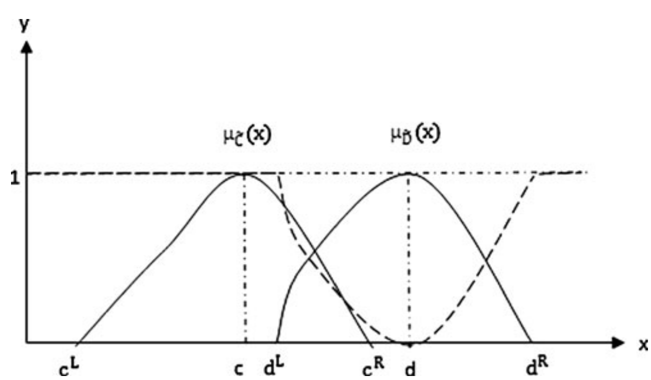

Fig. 6 $c \leq d_L$ with $\mathrm{ND}\left(\widetilde{D}, \widetilde{C}\right) = 1$

Table 1 Numerical example 1 of 6×6 FJSSP

Processing machine (fuzzy processing time)						
Job 1	1 (5 6 13)	5(3 4 5)	2 (1 2 3)	6 (3 4 5)	4(2 3 4)	3 (2 3 4)
Job 2	1 (3 4 5)	2 (2 4 5)	3 (1 3 5)	6 (4 5 6)	4 (5 6 7)	5 (6 7 8)
Job 3	3 (1 2 3)	6 (5 6 7)	5 (4 5 6)	4 (3 4 5)	2 (1 2 3)	1 (1 2 3)
Job 4	6 (2 3 4)	5 (1 2 3)	4 (2 3 4)	2 (2 3 5)	1 (3 4 6)	3 (3 4 5)
Job 5	6 (3 4 5)	5 (2 3 4)	4 (1 2 3)	3 (2 3 4)	2 (4 5 6)	1 (2 3 4)
Job 6	5(6 7 8)	6 (4 5 6)	1 (2 3 4)	2 (3 4 5)	3 (2 3 4)	4 (1 2 3)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Fuzzy due date	30 40	35 40	20 28	32 40	30 35	40 45

3.2 Objective functions based on PSD and ND

In the FJSSP, the processing time and the due date are assumed to be fuzzy numbers. It's easy to know that the completion time of each job J_j is also fuzzy number. Therefore, we proposed our objective functions based on the completion time and due date using the ranking concept of fuzzy numbers. Here, we refer to Dubois and Prade's [16] fuzzy number ranking concept in possibility and necessity theory. Let $\widetilde{A}$ and $\widetilde{B}$ be two fuzzy numbers, Dubois and Prade define the following four measures

$$\mathrm{P\,D}\left(\widetilde{A},\widetilde{B}\right) = \sup_{u\geq v}\min\left\{\mu_{\widetilde{A}}(\mu),\mu_{\widetilde{B}}(v)\right\}, \quad (1)$$

$$\mathrm{PSD}\left(\widetilde{A},\widetilde{B}\right) = \sup_{u}\inf_{\{v:v\geq u\}}\min\left\{\mu_{\widetilde{A}}(u),1-\mu_{\widetilde{B}}(v)\right\}, \quad (2)$$

$$\mathrm{ND}\left(\widetilde{A},\widetilde{B}\right) = \inf_{u}\sup_{\{v:v\leq u\}}\max\left\{1-\mu_{\widetilde{A}}(u),\mu_{\widetilde{B}}(v)\right\}, \quad (3)$$

$$\mathrm{NSD}\left(\widetilde{A},\widetilde{B}\right) = 1-\sup_{u\leq v}\min\left\{\mu_{\widetilde{A}}(u),\mu_{\widetilde{B}}(v)\right\}, \quad (4)$$

where $\mathrm{PD}\left(\widetilde{A},\widetilde{B}\right)$ and $\mathrm{ND}\left(\widetilde{A},\widetilde{B}\right)$ represent the grade of possibility and necessity of $\widetilde{A}$ dominanting $\widetilde{B}$, respectively, that is the grade of possibility and necessity of $\widetilde{A}\geq\widetilde{B}$. $\mathrm{PD}\left(\widetilde{A},\widetilde{B}\right)$ and $\mathrm{NDS}\left(\widetilde{A},\widetilde{B}\right)$ respectively represent the grade of possibility and necessity of $\widetilde{A}$ strictly dominating $\widetilde{B}$, that is the grade of possibility and necessity of $\widetilde{A}\geq\widetilde{B}$. Given the four measures above, we thus obtain four linear orderings when it comes to rank N fuzzy numbers. However, it's not all of the four measures that are consistent with each other, so with different measures, we may gain inconsistent results. Fortunately, the PSD and the ND measures are pairwise consistent due to their tournament relation between two fuzzy numbers, which are required to have continuous membership functions [16]. Therefore, in this paper, we use the ranking measures given in (2) and (3) together, that's PSD and ND, to investigate the relation between fuzzy completion time of job J_j and its fuzzy due date.

In a FJSSP problem, we always want to finish every job before its due date; so in this paper, we are going to seek for a schedule that more jobs finished before their given due dates. Since the processing time of every operation of a job is fuzzy number, the completion time of each job must be fuzzy number too. For a given job J_j, we can consider the four measures mentioned above between its completion time $\widetilde{C}_{J_j}$ and due date $\widetilde{D}_{J_j}$. Here considering the consistency, we just discuss two of them, $\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ and $\mathrm{ND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$, that is the strictly possibility measure, between $\widetilde{D}_{J_j}$ and .$\widetilde{C}_{J_j}$, and the necessity measure. Moreover, in order to propose the suitable objective functions to obtain the "optimal" schedule, we are going to interpret the meaning of $\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ and $\mathrm{ND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$. From ranking con-

Table 2 Numerical example 2 of 6×6 FJSSP

Processing machine (fuzzy processing time)						
Job 1	4 (9 13 17)	3 (6 9 12)	1 (10 11 13)	5 (5 8 11)	2 (10 14 17)	6 (9 11 15)
Job 2	4 (5 8 9)	2 (7 8 10)	5 (3 4 5)	3 (3 5 6)	1 (10 14 17)	6 (4 7 10)
Job 3	5 (3 5 6)	4 (3 4 5)	3 (2 4 6)	1 (5 8 11)	2 (3 5 6)	6 (1 3 4)
Job 4	6 (8 11 14)	3 (5 8 10)	1 (9 13 17)	4 (8 12 13)	2 (10 12 13)	5 (3 5 7)
Job 5	3 (8 12 13)	5 (6 9 11)	6 (10 13 17)	2 (4 6 8)	1 (3 5 7)	4 (4 7 9)
Job 6	2 (8 10 13)	4 (8 9 10)	6 (6 9 12)	3 (1 3 4)	5 (3 4 5)	1 (2 4 6)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Fuzzy due date	112 121	82 91	49 60	97 102	83 89	54 59

Table 3 Numerical example 3 of 6×6 FJSSP

Processing machine (fuzzy processing time)						
Job 1	6 (5 7 10)	5 (10 14 17)	4 (1 3 5)	3 (1 3 5)	2 (4 6 8)	1 (9 10 11)
Job 2	5 (6 7 8)	1 (9 13 17)	3 (8 12 13)	6 (2 3 4)	4 (10 13 16)	2 (2 3 4)
Job 3	3 (4 5 6)	1 (10 11 12)	5 (9 12 16)	2 (8 12 13)	6 (6 9 12)	4 (4 7 9)
Job 4	4 (1 2 4)	5 (2 4 5)	6 (5 7 8)	3 (5 8 10)	1 (3 5 7)	2 (6 8 10)
Job 5	4 (9 11 15)	1 (4 6 9)	5 (1 2 3)	6 (10 11 15)	2 (4 7 8)	3 (10 11 12)
Job 6	5 (6 7 9)	3 (1 2 4)	2 (6 9 11)	6 (10 14 18)	4 (1 2 3)	1 (9 13 14)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Fuzzy due date	81 88	66 80	89 92	51 60	91 96	75 78

cept, we can easy know that, the value of $\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ is larger, the extent of possibility for $\widetilde{D}_{J_j}$ being bigger than $\widetilde{C}_{J_j}$ is larger. Similarly, the value of $\mathrm{ND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ is larger, the extent of necessity for $\widetilde{D}_{J_j}$ being bigger than $\widetilde{C}_{J_j}$ is larger. And in our FJSSP problem, we hope the values of both $\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ and $\mathrm{ND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ are as larger as possible. Therefore, here we are going to maximize the following objective functions

$$\max_{x\in\Pi} f(\pi)=\sum_{j=1}^{n}\omega_j\mathrm{PSD}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right), \tag{5}$$

$$\max_{x\in\Pi} f(\pi)=\sum_{j=1}^{n}\omega_j\mathrm{ND}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right), \tag{6}$$

where ω_j are the penalty coefficients, Π is the set of all schedules, and π is one of the schedules. Now, in order to consider both two measures simultaneously and since they are consistent, we combine these two measures together and get a new measure defined below

$$\mathrm{PND}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right)=\mathrm{PSD}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right)+\mathrm{ND}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right) \tag{7}$$

Similarly, $\mathrm{PND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$ can be totally interpreted as the degree of $\widetilde{D}_{J_j}$ being bigger than $\widetilde{C}_{J_j}$. Then we can search for a better schedule by maximize the following objective function

$$\max_{x\in\Pi} f(\pi)=\sum_{j=1}^{n}\omega_j\mathrm{PND}\left(\widetilde{D}_{J_i},\widetilde{C}_{J_i}\right) \tag{8}$$

Here, we can also add penalty coefficients for both PSD and ND, to get a more general measure defined below

$$\mathrm{GPND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)=\omega_1\times\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)+\omega_2\times\mathrm{ND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right) \tag{9}$$

where ω_1 and ω_2 are the penalty coefficients respectively imposing upon the possibility measure and necessity measure. And at the same time, we can also maximize a more general objective function as follows

$$\max_{\pi\in\Pi} f(\pi)=\sum_{j=1}^{n}\omega_j\times\mathrm{GPND}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right) \tag{10}$$

Now, there are several objective functions. However, in this paper, in order to get consistent results, we combine PSD and ND, and just consider the objective function given in formula 10.

Table 4 Numerical example 4 of 6×6 FJSSP

Processing machine (fuzzy processing time)						
Job 1	1 (6 7 9)	6 (1 2 4)	4 (4 7 8)	5 (1 2 3)	3 (9 10 13)	2 (2 3 5)
Job 2	1 (10 13 16)	4 (7 11 15)	5 (6 8 11)	6 (2 4 5)	2 (9 12 15)	3 (2 3 4)
Job 3	2 (5 6 9)	6 (9 10 11)	3 (6 7 10)	4 (9 11 14)	1 (8 10 14)	5 (9 11 12)
Job 4	6 (10 11 15)	4 (3 4 6)	1 (9 12 16)	2 (9 12 15)	5 (4 5 7)	3 (5 7 9)
Job 5	4 (2 3 5)	5 (8 12 14)	3 (1 3 5)	2 (3 4 5)	1 (3 4 6)	6 (4 5 6)
Job 6	5 (5 8 10)	3 (7 10 11)	1 (1 3 4)	6 (6 8 9)	4 (4 6 7)	2 (3 4 6)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Fuzzy due date	43 50	96 102	93 103	71 75	49 54	62 70

3.3 Calculation for fuzzy completion time

Given a schedule of the FJSSP, the completion time of a job is the completion time of the last operation of the job. For computing the completion time of an operation, we need to establish its start time, and this is related to its order and the machine it preceeds.

Before this, in order to simplify the calculations, we are planning to introduce a new fuzzy number, called PLR fuzzy number, and just focus on it in the next part of this paper. Now we firstly give the definition of the PLR fuzzy number.

Define 2 Given a fuzzy number $\widetilde{A}$, it is called PLR fuzzy number if it has the form $\widetilde{A} = a_1, a, a, L, R$, distributed by the following membership functions respectively,

$$\mu_{\overline{A}}(x) = \begin{cases} L(x), a_L \leq x \leq a \\ R(x), a \leq x \leq a_R \\ 0, \text{other} \end{cases},$$

where a_L, a, and a_R are non-negative numbers and $a_L \leq a \leq a_R$, L is strictly increasing function and R is strictly decreasing function, which satisfy

$$L(a_L) = R(a_R) = 0 \text{ and } L(a) = R(a) = 1,$$

and both of them are continuous functions.

From definition 2, we can easily know that triangle fuzzy number is a kind of special PLR fuzzy number. Furthermore, we have the following results.

Proposition 2 Let $\widetilde{A} = (a_L, a, a_R, L_R, R_A)$ and $\widetilde{B} = (b_L, b, b_R, L_B, R_B)$ be two PLR fuzzy numbers, then $\widetilde{A} \oplus \widetilde{B}$ and $m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}$ are also PLR fuzzy numbers, and we have

$$\left(\widetilde{A} \oplus \widetilde{B}\right)_\alpha = \left[\widetilde{A}_\alpha^L + \widetilde{B}_\alpha^L, \widetilde{A}_\alpha^U + \widetilde{B}_\alpha^U\right],$$
$$\left(m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}\right)_\alpha = \left[\max\left(\widetilde{A}_\alpha^L, \widetilde{B}_\alpha^L\right), \max\left(\widetilde{A}_\alpha^U, \widetilde{B}_\alpha^U\right)\right].$$

Moreover,

$$\widetilde{A} \oplus \widetilde{B} = \left(a_L + b_L, a + b, a_R + b_R, L_{AB}^+, R_{AB}^+\right),$$
$$m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\} = \left(\max(a_L, b_L), \max(a, b), \max(a_R, b_R), L_{AB}^{\max}, R_{AB}^{\max}\right),$$

where L_{AB}^+ and $L_{AB}^{\max}$ (R_{AB}^+ and $R_{AB}^{\max}$) depends on L_A and L_B (R_A and R_B respectively).

Let $\widetilde{A} = (a_L, a, a_R)$ and $\widetilde{B} = (b_L, b, b_R)$ be two triangle fuzzy numbers, it's easy to obtain that

$$\widetilde{A} \oplus \widetilde{B} = (a_L + b_L, a + b, a_R + b_R)$$

is also a triangle fuzzy number. However, $m\widetilde{a}x\left\{\widetilde{A}, \widetilde{B}\right\}$ is not necessarily a triangle fuzzy number.

Table 5 Numerical example 1 of 10×10 FJSSP

Processing machine (fuzzy processing time)										
Job 1	8 (2 3 4)	6 (3 5 6)	5 (2 4 5)	2 (4 5 6)	1 (1 2 3)	3 (3 5 6)	9 (2 3 5)	4 (1 2 3)	7 (3 4 5)	10(2 3 4)
Job 2	10 (2 3 4)	7 (2 3 5)	4 (2 4 5)	6 (1 2 3)	8 (4 5 6)	3 (2 4 6)	2 (2 3 4)	1 (1 3 4)	5 (2 3 4)	9(3 4 5)
Job 3	6 (2 4 5)	9 (1 2 3)	10 (2 3 5)	8 (1 2 4)	1 (3 5 6)	7 (1 3 4)	4 (1 3 5)	2 (1 2 4)	5 (2 4 5)	3(1 3 5)
Job 4	1 (1 2 3)	5 (3 4 5)	8 (1 3 5)	9 (2 4 6)	10 (2 4 5)	6 (1 2 4)	7 (3 4 5)	2 (1 3 5)	4 (1 3 6)	3(1 3 4)
Job 5	2 (2 3 4)	7 (1 3 4)	3 (1 3 4)	5 (1 2 3)	8 (1 3 5)	9 (2 3 4)	10 (3 4 5)	6 (1 3 4)	1 (3 4 5)	4(1 3 4)
Job 6	4 (2 3 4)	2 (2 3 4)	3 (1 2 3)	5 (2 4 5)	6 (1 3 4)	8 (1 3 4)	7 (3 4 5)	9 (1 2 3)	10 (2 4 5)	1(1 3 4)
Job 7	3 (2 3 4)	5 (1 4 5)	4 (1 3 5)	1 (3 4 5)	9 (2 3 4)	7 (3 4 5)	2 (1 2 3)	10 (3 5 6)	8 (3 5 6)	6(1 2 3)
Job 8	7 (3 4 5)	1 (1 2 3)	9 (3 4 5)	6 (2 4 5)	10 (1 3 4)	2 (2 3 4)	5 (1 2 3)	3 (2 4 5)	4 (3 4 5)	8(2 3 5)
Job 9	(3 4 5)	4 (1 3 4)	10 (1 3 5)	2 (2 3 4)	3 (3 5 6)	6 (2 4 5)	8 (1 3 4)	1 (3 4 5)	5 (1 2 3)	7(3 4 5)
Job 10	7 (2 4 5)	5 (1 2 3)	2 (3 4 5)	4 (2 3 4)	1 (1 2 3)	8 (3 4 5)	10 (2 4 5)	6 (3 4 5)	3 (1 2 3)	9(1 2 4)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10
Fuzzy due date	45 60	50 60	50 65	50 65	50 65	45 60	45 60	50 60	45 60	50 60

Table 6 Numerical example 2 of 10×10 FJSSP

Processing machine (fuzzy processing time)										
Job 1	4 (10 13 16)	6 (4 7 9)	7 (10 12 13)	8 (5 6 7)	9 (6 8 9)	2 (7 8 12)	5 (10 12 15)	3 (5 6 7)	1 (2 3 5)	10 (10 14 18)
Job 2	2 (3 5 6)	1 (9 10 13)	3 (5 8 9)	10 (9 12 16)	6 (5 6 9)	7 (7 11 12)	5 (9 13 14)	9 (8 12 16)	8 (2 4 6)	4 (4 7 10)
Job 3	7 (9 12 14)	10 (10 13 14)	9 (5 7 8)	4 (3 4 6)	5 (4 7 8)	2 (3 5 7)	8 (3 4 6)	3 (1 2 4)	1 (5 7 9)	6 (9 11 13)
Job 4	5 (10 12 16)	7 (1 2 4)	10 (6 8 10)	6 (1 3 5)	4 (7 8 11)	1 (5 8 10)	8 (9 10 14)	9 (4 7 8)	2 (4 7 10)	3 (2 3 5)
Job 5	5 (9 12 15)	6 (8 11 14)	10 (10 14 17)	1 (5 7 9)	2 (2 4 5)	4 (1 3 5)	7 (7 8 10)	9 (3 4 6)	8 (9 11 13)	3 (1 2 3)
Job 6	8 (4 7 9)	2 (10 12 15)	3 (3 4 5)	4 (10 14 18)	1 (5 6 9)	5 (10 14 16)	9 (10 12 15)	6 (8 9 12)	10 (5 8 9)	7 (4 7 10)
Job 7	5 (8 12 13)	10 (2 4 6)	8 (10 14 18)	1 (5 7 9)	6 (4 5 8)	9 (4 5 7)	7 (7 10 11)	2 (10 11 12)	3 (10 13 15)	4 (9 12 13)
Job 8	2 (10 12 15)	6 (5 6 9)	3 (1 2 4)	8 (6 9 12)	4 (4 6 9)	1 (7 11 14)	9 (7 11 13)	5 (6 9 11)	10 (8 11 13)	7 (7 9 13)
Job 9	9 (2 4 6)	6 (2 3 5)	2 (2 3 4)	4 (4 6 7)	10 (6 8 9)	3 (8 12 14)	1 (4 7 9)	7 (8 11 14)	8 (1 2 3)	5 (3 5 6)
Job 10	7 (5 8 9)	8 (6 8 9)	3 (8 12 16)	1 (6 9 12)	9 (7 11 13)	5 (10 11 14)	4 (7 10 11)	2 (3 5 7)	10 (3 4 6)	6 (8 11 15)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10
Fuzzy due date	169 184	123 134	100 110	102 105	121 136	167 174	120 130	163 176	79 94	160 163

Table 7 Numerical example 3 of 10×10 FJSSP

Processing machine (fuzzy processing time)										
Job 1	7 (3 4 5)	9 (10 12 13)	2 (4 7 10)	5 (5 8 9)	4 (10 12 16)	1 (10 11 13)	6 (4 7 10)	3 (4 5 8)	8 (4 6 7)	10 (9 12 13)
Job 2	10 (10 11 14)	3 (9 12 14)	5 (7 10 14)	4 (7 11 12)	8 (2 4 6)	1 (8 10 14)	2 (7 11 12)	7 (8 11 14)	6 (6 9 10)	9 (10 14 15)
Job 3	5 (5 7 10)	7 (2 3 5)	4 (1 3 4)	3 (7 10 12)	1 (8 11 12)	8 (2 4 5)	6 (4 5 7)	2 (7 8 10)	9 (9 13 14)	10 (8 12 15)
Job 4	5 (5 8 10)	6 (4 5 8)	4 (7 10 14)	7 (3 5 7)	8 (4 5 6)	2 (8 10 12)	1 (2 3 4)	9 (2 3 5)	3 (6 8 11)	10 (6 8 11)
Job 5	2 (4 7 10)	7 (5 6 7)	4 (9 10 14)	1 (2 3 4)	3 (9 12 13)	5 (5 6 9)	9 (5 7 8)	6 (1 2 4)	8 (3 4 6)	10 (2 4 6)
Job 6	7 (1 3 4)	3 (3 4 5)	8 (3 5 6)	2 (5 7 8)	4 (8 9 13)	9 (9 12 14)	10 (4 7 8)	1 (1 2 4)	6 (2 4 5)	5 (6 9 12)
Job 7	10 (10 11 14)	8 (2 3 4)	6 (9 10 12)	3 (9 10 11)	7 (4 5 6)	2 (3 5 7)	9 (1 3 4)	4 (2 4 5)	1 (8 10 13)	5 (7 10 11)
Job 8	6 (7 11 15)	2 (9 13 15)	8 (5 6 9)	4 (8 9 13)	7 (6 9 12)	3 (6 8 10)	9 (6 9 11)	5 (1 2 4)	10 (8 12 14)	1 (6 9 12)
Job 9	7 (4 7 10)	2 (3 5 6)	8 (6 9 10)	6 (3 5 6)	9 (8 11 12)	4 (5 7 10)	10 (4 6 9)	1 (1 2 4)	3 (3 5 7)	5 (10 12 15)
Job 10	4 (1 2 3)	1 (8 12 13)	9 (7 8 9)	10 (6 9 12)	5 (9 11 15)	2 (7 11 15)	6 (10 14 18)	3 (1 3 5)	8 (1 2 4)	7 (2 3 5)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10
Fuzzy due date	151 156	154 157	106 117	123 138	85 88	86 94	120 135	149 158	117 124	142 148

Table 8 Numerical example 4 of 10×10 FJSSP

Processing machine (fuzzy processing time)										
Job 1	2 (7 11 12)	9 (5 7 9)	10 (4 5 8)	8 (7 11 15)	6 (8 9 13)	3 (7 9 12)	1 (6 7 9)	5 (2 3 4)	7 (4 7 8)	4 (2 4 5)
Job 2	6 (3 5 6)	1 (1 2 3)	5 (1 2 4)	8 (10 11 14)	2 (8 12 14)	4 (2 4 5)	7 (5 8 10)	10 (8 9 11)	3 (2 4 5)	9 (4 6 8)
Job 3	9 (9 11 12)	7 (8 9 13)	3 (7 9 10)	6 (3 5 7)	2 (10 13 17)	5 (2 4 6)	4 (8 9 13)	8 (3 4 5)	1 (1 2 3)	10 (4 5 7)
Job 4	2 (8 11 13)	1 (2 3 5)	8 (4 7 8)	4 (4 6 7)	3 (5 7 8)	9 (6 8 10)	10 (2 3 4)	7 (8 12 14)	6 (5 7 8)	5 (4 6 9)
Job 5	3 (4 6 7)	9 (4 7 9)	7 (6 9 12)	10 (2 4 5)	4 (1 3 5)	6 (9 10 12)	5 (10 11 12)	1 (5 7 9)	2 (8 10 13)	8 (4 6 9)
Job 6	9 (7 8 9)	4 (4 5 8)	1 (4 5 7)	3 (4 6 8)	7 (6 7 9)	10 (3 5 7)	8 (9 11 15)	2 (7 10 12)	5 (6 9 12)	6 (5 7 9)
Job 7	6 (5 6 8)	2 (5 6 9)	9 (4 6 7)	8 (8 9 10)	4 (9 13 14)	3 (4 5 8)	10 (7 10 12)	5 (3 4 6)	1 (9 12 15)	7 (5 8 10)
Job 8	6 (7 11 14)	4 (3 5 6)	3 (10 13 14)	10 (7 9 11)	8 (7 10 13)	1 (5 8 10)	9 (7 11 13)	5 (10 12 16)	7 (10 14 16)	2 (6 7 10)
Job 9	10 (2 3 5)	6 (1 3 4)	7 (4 5 8)	3 (10 14 16)	8 (3 4 6)	5 (1 3 4)	1 (1 2 3)	9 (3 5 6)	2 (5 8 9)	4 (8 11 14)
Job 10	4 (6 8 9)	7 (5 7 9)	5 (5 6 8)	9 (3 5 7)	1 (2 3 5)	2 (1 2 3)	3 (8 10 13)	8 (9 13 16)	10 (4 5 8)	6 (2 4 5)
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10
Fuzzy due date	124 128	81 95	92 99	91 103	109 115	102 107	118 128	170 178	75 86	94 107

Table 9 Parameters of DE

Problem	Population size	Generations	ω_1	ω_2
6×6	100	100	0.5	0.5
10×10	100	100	0.5	0.5

Now we are going to give an example to make the computing of the completion time clear. There is a 2×2 FJSSP below:

Job 1: machine 1 (2, 5, 6), machine 2 (5, 7, 8),
Job 2: machine 2 (3, 4, 7), machine 1 (1, 2, 3).

Here (2, 5, 6) and others are triangle fuzzy numbers and represent the fuzzy processing times of operations. Suppose 2 1 2 1 is a given schedule, then we can see that the start time of job 2 is (0, 0, 0) and the start time of job 1 is also (0, 0, 0). Next, the second operation of job 2 is processed on machine 1, and at the same time, it must proceed after the first operation. So its start time is max {(2,5,6),(3,4,7)}. Here, stated in a simple way, we shall approximate max of triangle fuzzy numbers with the following formula:

$$m\tilde{a}x\left\{\widetilde{A},\widetilde{B}\right\} \cong (\max(a_L, b_L), \max(a, b), \max(a_R, b_R))$$

So we obtain (3, 5, 7) approximately. Easily, we compute the completion time of job 2, the addition of (3, 5, 7) and (1, 2, 3), that is (4, 7, 10).

3.4 Analytic formulas for the objective functions

Here, we assume that the completion time $\widetilde{C}_{J_j}$ of every job J_j is a PLR fuzzy number, so is the due date $\widetilde{D}_{J_j}$. In this case, we can derive the analytic formulas for the objective functions proposed above, which also make the computation neater. Now, we consider the strictly dominate possibility measure $\mathrm{PSD}\left(\widetilde{D}_{J_j},\widetilde{C}_{J_j}\right)$

Table 10 The time and objective function value of each problem

Problem	Time (s)	B_ GPNS	W_GPNS	A_GPNS
6×6 1	1.313	3.32092	3.28299	3.29664
6×6 2	1.328	5.89644	5.73691	5.83721
6×6 3	1.328	5.26347	5.2111	5.22847
6×6 4	1.343	5.17221	5.05369	5.10072
10×10 1	4.093	6.43048	6.34413	6.39046
10×10 2	4.063	8.07412	7.82981	7.93663
10×10 3	4.062	8.61662	8.40187	8.50359
10×10 4	4.031	5.9632	5.79505	5.85307

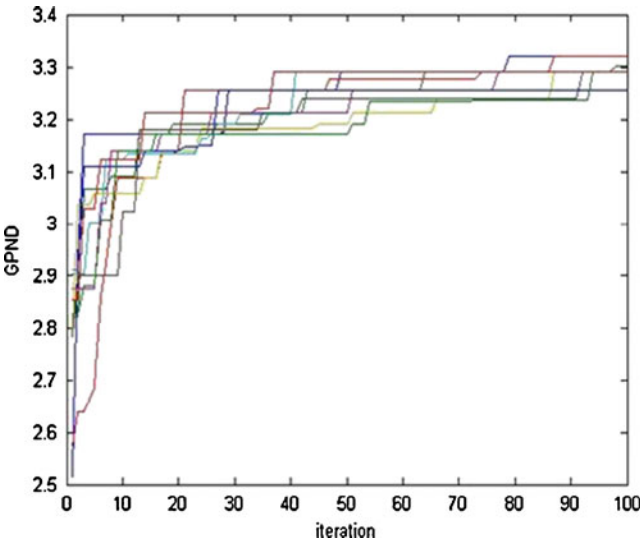

Fig. 7 GPND convergence trends for 6×6 JSSP 1

Proposition 3 Let $\widetilde{C} = (c_L, c, c_R, L_C, R_C)$ and $\widetilde{D} = (d_L, d, d_R, L_D, R_D)$ be two PLR fuzzy numbers, and $R_c^{'}$ denotes the function of $1-R_C$. Then we have the following results:

(1) If $d_R \leq c$ then $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = 0$.
(2) If $d < c_R$ and $d_R > c$, and suppose the point of intersection of the two lines R_D and $R_C^{'}$ is (x_0, y_0), then $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = y_0$.
(3) If $d \geq c_R$ then $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = y_1$.

Proof Firstly, let us see what's the meaning of the PSD from mathematics. From formula 2, we know that $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = \sup_u \inf_{\{v: v \geq u\}} \min\left(\mu_{\widetilde{D}}(u), 1 - \mu_{\widetilde{C}}(v)\right)$. Therefore, we can explain it as follows: assume that the domain of u is $\{u_1, u_2, \ldots, u_n\}$, $\bar{y}_j$ is the value of $\min\left(\mu_{\widetilde{D}}(u_1), 1 - \mu_{\widetilde{C}}(v_j)\right)$ for $\forall v_j \geq u_1$, and $\bar{x}_1 = \min\left\{\bar{y}_j\right\}$, j=1,2,…. Similarly, we can compute $\bar{x}_2, \bar{x}_3, \ldots, \bar{x}_n$. Finally, we obtain a list of $\{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n\}$. The value of $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right)$ is the max value of the list, that is $\mathrm{PSD}\left(\widetilde{D}, \widetilde{C}\right) = \max\{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n\}$.

In order to make our proof neat, we shall apply figures. From Fig. 1, we can see that when $\forall u_i < d_R$, the minimum of all $\bar{y}_j$ is 0, due to $d_R \leq c$. For $\forall u_i \geq d_R$, it's obvious that $\bar{x}_i$ is 0 since $\mu_{\widetilde{D}}(u_i) = 0$. Then we have (1).

Now we pay attention to (2). See Fig. 2, R_D and $R_C^{'}$ intersect at point (x,y). Let u_i=0, it's not difficult to know that $\min\left(\mu_{\widetilde{D}}(u_i), 1 - \mu_{\widetilde{C}}(v_j)\right) = y_0$ for $\forall v_j \geq u_i$. And when $c < u_i < x_0, 1 - \mu_{\widetilde{C}}(v_j) \geq 1 - \mu_{\widetilde{C}}(u_i)$ for $\forall v_j \geq u_i$, so $\min\left(\mu_{\widetilde{D}}(u_i), 1 - \mu_{\widetilde{C}}(v_j)\right) = 1 - \mu_{\widetilde{D}}(u_i) < y_0$. Next, we consider $u_i \leq c$, in this case, the minimum of $1 - \mu_{\widetilde{C}}(v_j) = 0$ for $\forall v_j \geq u_i$ since there exists a value $v_j^{'} = c$ to satisfy $1 - \mu_{\widetilde{C}}\left(v_j^{'}\right) = 0$. Lastly, when $u_i > x_0$, we can

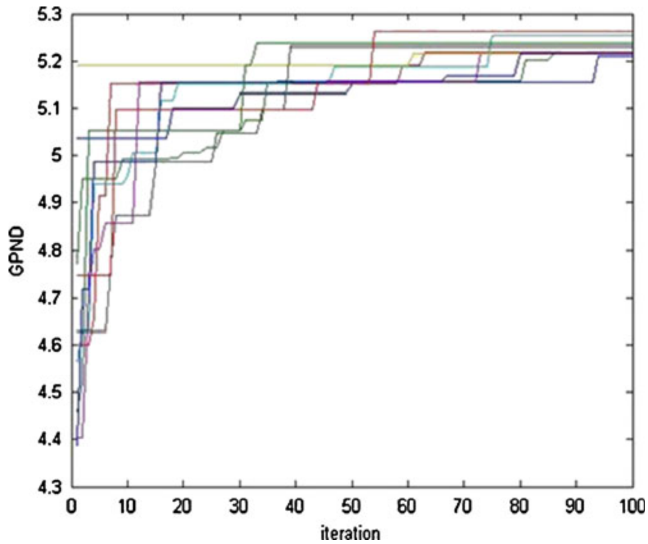

Fig. 9 GPND convergence trends for 6×6 JSSP 3

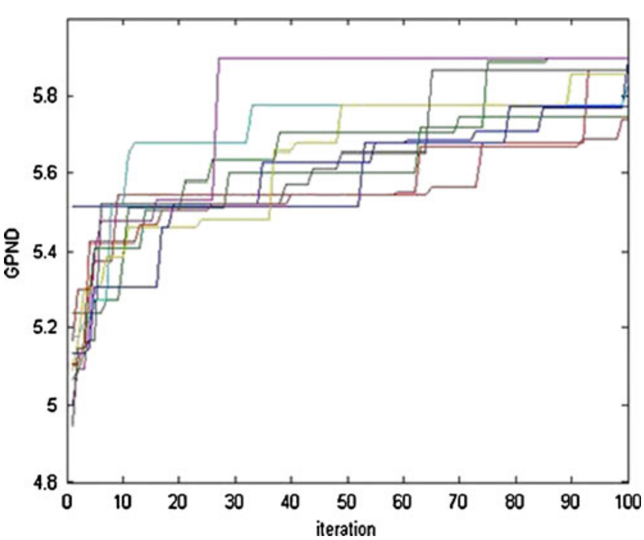

Fig. 8 GPND convergence trends for 6×6 JSSP 2

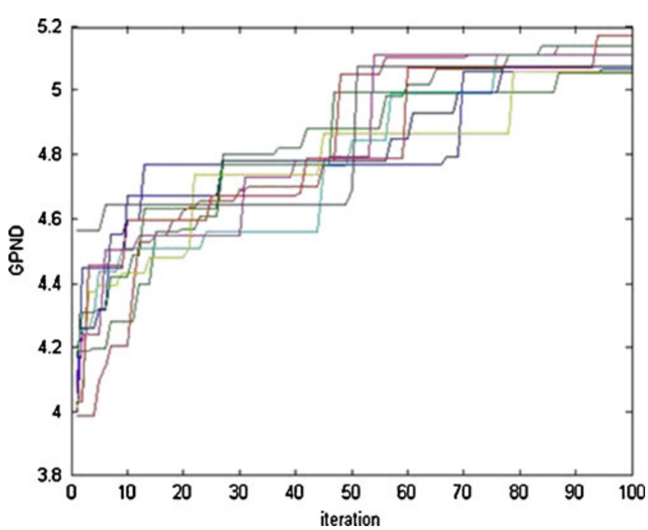

Fig. 10 GPND convergence trends for 6×6 JSSP 4

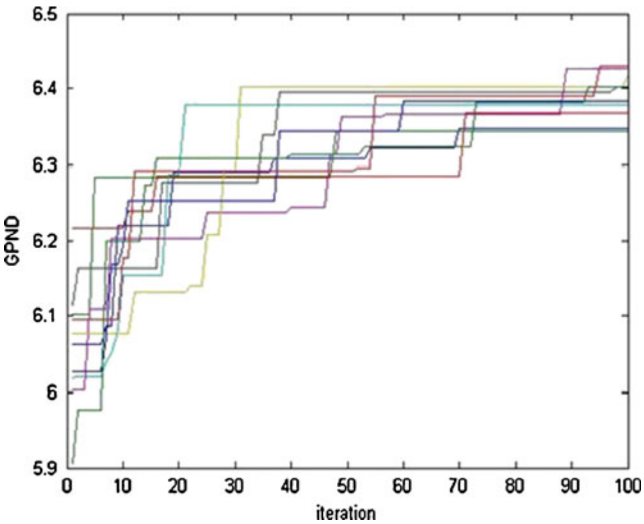

Fig. 11 GPND convergence trends for 10×10 JSSP 1

easily obtain that $\min\left(\mu_{\widetilde{D}}(u_i), 1-\mu_{\widetilde{C}}(v_j)\right) = \mu_{\widetilde{D}}(u_i) < y_0$ for $\forall v_j \geq u_i$. In summary, we conclude (2).

From Fig. 3, it is easy to see that $PSD\left(\widetilde{D},\widetilde{C}\right) = 1$, since $\mu_{\widetilde{D}}(d) = 1-\mu_{\widetilde{C}}(d) = 1$.

We complete the proof.

Proposition 4 Let $\widetilde{C} = (c_L, c, c_R, L_C, R_C)$ and $\widetilde{D} = (d_L, d, d_R, L_D, R_D)$ be two PLR fuzzy numbers, and $L_D^{'}$ denotes the function of $1-L_D$. Then we have the following results:

(1) If $c_L \geq d$, then $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right) = 0$.
(2) If $c_L < d$ and $c > d_L$, and suppose the point of intersection of the two lines $L_D^{'}$ and L_C is (x_0, y_0), then $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right) = y_0$.
(3) If $c \leq d_L$, then $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right) = 1$.

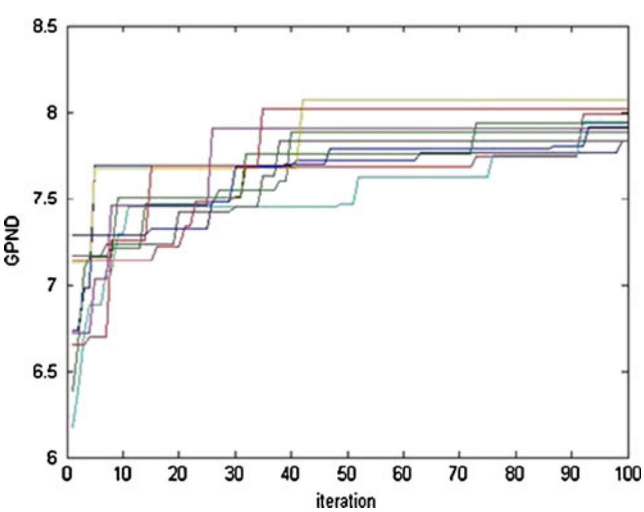

Fig. 12 GPND convergence trends for 10×10 JSSP 2

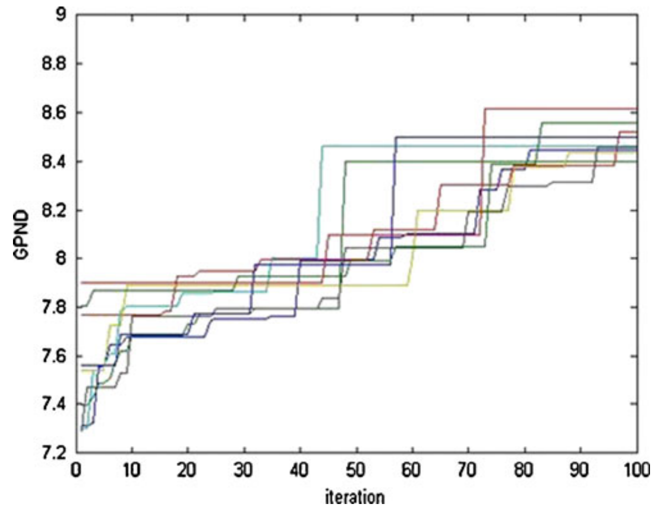

Fig. 13 GPND convergence trends for 10×10 JSSP 3

Proof Similarly, let us start with the mathematic meaning of ND by formula (3). We can easily get that $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right) = \inf_{u} \sup_{\{v:v\leq u\}} \max\left(1-\mu_{\widetilde{D}}(u), \mu_{\widetilde{C}}(v)\right)$, so it can be explained as below: suppose that the domain of u is $\{u_1, u_2, \ldots, u_n\}$, for $\forall v' \leq u_1$, we compute a value by $\max\left(1-\mu_{\widetilde{D}}(u_1), \mu_{\widetilde{C}}(v')\right)$, then we can get a list of values like this. Let $\overline{y}_1$ as the maximum of all the values. Next we can obtain $\overline{y}_2, \ldots, \overline{y}_n$ for u_2, ..., u_n, respectively. Then the value of $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right)$ as the mininimum value of the list of $\{\overline{y}_1, \overline{y}_2, \ldots, \overline{y}_n\}$, that is $\mathrm{ND}\left(\widetilde{D},\widetilde{C}\right) = \min\{\overline{y}_1, \overline{y}_2, \ldots, \overline{y}_n\}$.

Here, we also apply some figures to accomplish our proof. See Fig. 4, there exists u_i=d that makes $\overline{y}_i = 0$ since for $\forall v' \leq u_i \quad \mu_{\widetilde{C}}(v') = 0$ comes into existence. So we have (1).

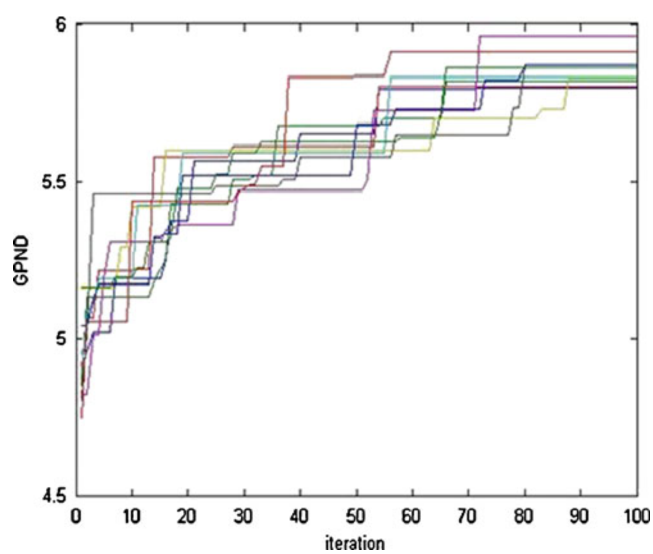

Fig. 14 GPND convergence trends for 10×10 JSSP 4

Table 11 The fuzzy completion time of 6×6 FJSSP 1

Job 1	8 10 18	12 16 23	13 18 26	16 22 31	18 25 35	20 28 39
Job 2	3 4 5	5 8 10	6 11 15	10 16 21	15 22 28	21 29 36
Job 3	12 17 26	21 28 36	25 33 42	28 37 47	29 39 50	30 41 53
Job 4	5 7 9	6 9 12	8 12 16	10 15 21	15 20 32	18 24 37
Job 5	3 4 5	5 7 9	6 9 12	8 12 16	12 17 22	14 20 26
Job 6	9 12 15	13 17 21	15 20 25	18 24 30	20 27 34	21 29 37

For (2), let us shift to Fig. 5. Let the point of intersection of L_D' and L_C is (x_0, y_0). When u=x_0, we can see that for $\forall v' \leq x_0$ the maximum of $\mu_{\widetilde{C}}(v')$ is $\mu_{\widetilde{C}}(x_0) = 1 - \mu_{\widetilde{D}}(x_0) = y_0$. Next we consider the situations of u<x_0 and u>x_0. When u<x_0, we can easily get that $1 - \mu_{\widetilde{D}}(u) > 1 - \mu_{\widetilde{D}}(x_0) = y_0$ and $\mu_{\widetilde{C}}(v') < y_0$ for $\forall v' \leq u$, so it's easy to obtain that $\overline{y} = \max\left(1 - \mu_{\widetilde{D}}(u), \mu_{\widetilde{C}}(v')\right) \geq y_0$, respect to u<x_0. Similarly, we have $\overline{y}' = \max\left(1 - \mu_{\widetilde{D}}(u), \mu_{\widetilde{C}}(v')\right) \geq y_0$, respect to u>x_0. Therefore, we have $\mathrm{ND}\left(\widetilde{D}, \widetilde{C}1\right) = \min\{\overline{y}, y_0, \overline{y}'\} = y_0$, that is (2).

Now look at Fig. 6. When u>c, we can see that $\max\left\{\max\left(1 - \mu_{\widetilde{D}}(u), \mu_{\widetilde{C}}(v')\right)\right\}$ always be 1 for $\forall v' \leq u$, due to that there must exist a v_0=c≤u satisfy $\mu_{\widetilde{C}}(v_0) = 1$. When u≤c, we know that $1 -_{\widetilde{D}}(u) = 1$ since we have $c \geq d_L$. So in this case we always have $\max\left\{\max\left(1 - \mu_{\widetilde{D}}(u), \mu_{\widetilde{C}}(v')\right)\right\} = 1$. Therefore we obtain that $ND\left(\widetilde{D}, \widetilde{C}\right) = 1$.

We complete the proof.

Since triangle fuzzy number is a special type of PLR fuzzy number, all the propositions above can also be applied to it.

4 The DE algorithm

For a given n×m FJSSP problem, here we invoke the DE algorithm proposed by Storn and Price [20] to obtain the best schedule. However, as we know, the DE algorithm is a continuous optimization algorithm, so its original encoding scheme can't be straightly used to solve the scheduling problem. For our purpose, the key issue is to construct a direct relationship between the schedule of the jobs and the individuals in DE. Here, we use the concept of random keys proposed by Bean [21] to generate the job sequence of a specified individual. Suppose we consider a 3×2 FJSSP problem, then the length of individuals is 3×2, and we generate six random numbers in (0, 1) as a vector for each individual. And the job sequence is generated like below: for example, if an individual is (0.21, 0.34, 0.12, 0.17, 0.56, 0.43), we let 1←0.21, 2←0.34, 3←0.12, 4←0.17, 5←0.56 and 6←0.43, then sort the random numbers in ascending order and get a sequence (3,4,1,2,6,5). Since the sequence is not a legal job schedule, we modify each element by the number of jobs, that is 3 here, and also add 1 for avoiding 0. After this, we obtain the final result (1,2,2,3,1,3). It means the following process order: the first operation of job 1, the first operation of job 2, the second operation of job 2, the first operation of job 3, the second operation of job 1, the second operation of job 3. Certainly, some of operation can be operated in parallel, if they don't need the same machine and don't belong to the same job.

Similar to Genetic Algrithm(GA), DE has three evolutionary operator: select, crossover, and mutation. The significant difference from GA is that DE uses distance and direction information from the current population to guide the search process. The crucial idea behind DE is a scheme for producing trial vectors according to the manipulation of target vector and difference vector. In this paper, it works as follows.

For each individual's vector $x_i(G)$, i=1,…N, where N is the number of individuals and G is the G_{th} iteration, we can get a differential vector $v_i(G)$ by mutation:

$$v_{i,G} = x_{r_1,G} + F\left(x_{r_2,G} - x_{r_3,G}\right),$$

where r_1,r_2,r_3∈[1,…,N] are random integers, and $r_1 \neq r_2 \neq r_3 \neq i$, F is the scaling factor controlling the amplification of the differential evolution.

Table 12 The fuzzy completion time of 6×6 FJSSP 2

Job 1	25 34 41	31 43 53	41 55 71	46 63 82	56 77 99	65 88 114
Job 2	5 8 9	15 18 23	18 22 28	21 29 35	31 43 52	35 50 62
Job 3	3 5 6	8 12 14	15 24 29	20 32 40	23 37 46	25 40 50
Job 4	8 11 14	13 20 24	22 33 41	33 46 54	43 58 67	46 63 74
Job 5	8 12 13	14 21 24	24 34 43	28 40 51	31 45 58	37 53 67
Job 6	8 10 13	16 21 24	22 30 36	23 33 40	26 37 45	28 41 51

Table 13 The fuzzy completion time of 6×6 FJSSP 3

Job 1	5 7 10	16 21 27	17 24 32	18 27 37	28 39 47	37 49 58
Job 2	34 46 58	44 59 75	52 71 88	54 74 92	64 87 108	66 90 112
Job 3	4 5 6	14 17 21	26 35 45	34 47 58	40 56 70	44 63 79
Job 4	11 16 24	28 39 50	33 46 58	38 54 68	41 59 75	47 67 85
Job 5	9 11 15	13 17 24	17 23 29	27 34 44	31 41 52	41 52 64
Job 6	6 7 9	7 9 13	13 18 24	23 32 42	24 34 45	33 47 59

Then the trial individual can be produced by the crossover. Let D be the dimension of the vectors, it's operated as:

$$u_{i,j,G} = \begin{cases} v_{i,j,G}, \left(\text{rand}_j \leq CR\right) or (j = j_{\text{rand}}) \\ x_{i,j,G}, \text{otherwise} \end{cases},$$

where j=1,2,…,D, rand$_j$∈[0,1] is a random number, j_{rand}∈[1,2,…,D] is a random index, CR is the crossover rate, $u_{i,G}$ denotes the trial vector of the i_{th} individual at the G_{th} iteration, and $u_{i,j,G}$ is the value of the j_{th} dimension in the trial vector.

Next, select operator is used to produce the offspring by choosing between the trial population and the parent population:

$$x_{i,G+1} = \begin{cases} u_{i,G}, f\left(u_{i,G}\right) > f\left(x_{i,G}\right) \\ x_{i,G}, \text{otherwise} \end{cases},$$

where f is evaluated function, which responses to the objective function here. Suppose that we plan to consider the objective function in formula (10), and then the computational procedure is described as below:

Step 1. Initialize population randomly, initialize the parameters CR, F. And set the current generation $G=0$;

Step 2. Convert every individual to job sequence, and evaluate it by the objective function in formula (10), mark the best individual;

Step 3. Do the mutation, crossover, and select operator for each individual;

Step 4. $G = G + 1$, repeat Step2 to Step5 until the stopping criteria is reached.

In order to get better result, we also incorporate an improvement to the DE. The purpose of this is to exploit a better solution from the neighborhood of a solution. There are several neighborhoods. In this paper, we just use one neighborhood, that's swap [22], to improve the diversity of population and enhance the quality of the solution.

5 Numerical examples

For illustrating the approach we proposed, in this section, we consider some simple examples. Here, we list four 6×6 FJSSPs and 10×10 FJSSPs respectively, they are shown in Tables 1, 2, 3, 4, 5, 6, 7, and 8(the examples we use come from [11, 12]). The term $m(p^L,p,p^R)$ represents the machine and the processing time of operation, and term (d,d^R) represents the due date of job.

Now, we are ready to solve the eight FJSSPs by using the modified DE algorithm for illustrating the proposed objective functions. To be general, we just experiment our method on the bias of formula 10.The parameters of our algorithm are shown in Table 9, and each of them is set by a number of experiments. By the way, all the trials are performed ten times for each problem and all the codes is written with C++, and are run on a PC with an Intel 2.00 GHz CPU.

Table 10 shows some results of our methods. The column of time lists the average executing time of each example. The B_GPND column represents the maximum of GPND (that's formula 10) in ten times of each JSSP, the W_GPND is the worst GPND and the A_GPND is the average GPND. It's obvious that our method is largely efficient. Moreover, in order to show the state of convergence, we also give out the changed values of GPND in each generation for each example. And they are shown in Figs. 7, 8, 9, 10, 11, 12, 13, and 14.

Table 14 The fuzzy completion time of 6×6 FJSSP 4

Job 1	37 49 65	38 51 69	42 58 77	43 60 80	52 70 93	54 73 98
Job 2	10 13 16	17 24 31	23 32 42	31 38 47	40 50 62	42 53 66
Job 3	5 6 9	19 21 26	25 28 36	34 39 50	42 49 64	51 60 76
Job 4	10 11 15	13 15 21	22 28 37	31 40 52	35 45 59	40 52 68
Job 5	2 3 5	13 20 24	14 23 29	17 27 34	23 32 42	29 37 48
Job 6	5 8 10	12 18 21	13 21 25	25 29 35	29 35 42	32 39 48

Table 15 The fuzzy completion time of 10×10 FJSSP 1

Job 1	3 6 9	6 11 15	8 15 20	14 21 28	15 23 31	18 28 37	20 31 42	21 33 45	24 37 50	26 40 54
Job 2	2 3 4	4 6 9	6 10 14	7 12 17	11 17 23	13 21 29	16 24 33	17 27 37	19 30 41	22 34 46
Job 3	2 4 5	4 6 8	6 9 14	7 11 18	10 16 24	11 19 28	12 22 33	17 26 37	19 30 42	20 33 47
Job 4	1 2 3	4 6 8	5 9 13	7 13 19	9 17 24	10 19 28	13 23 33	14 26 38	15 29 44	16 32 48
Job 5	2 3 4	5 10 14	8 14 18	9 16 21	10 19 26	13 22 30	16 26 35	17 29 39	20 33 44	21 36 48
Job 6	3 6 8	6 9 12	9 16 21	11 20 26	12 24 32	14 27 37	18 31 42	19 33 45	23 37 50	24 40 54
Job 7	2 3 4	4 8 10	6 13 18	9 17 23	11 20 27	15 25 33	18 28 39	21 33 45	24 38 51	25 40 54
Job 8	9 17 23	10 19 26	13 23 31	15 27 36	16 30 40	20 33 44	21 35 47	23 39 52	26 43 57	28 46 62
Job 9	3 4 5	4 7 9	5 10 14	7 13 18	10 18 24	12 22 29	13 25 33	16 29 38	17 31 41	21 35 46
Job 10	4 7 10	5 10 13	9 14 18	11 19 27	12 21 30	15 25 35	18 29 40	21 33 45	22 37 50	23 39 54

Table 16 The fuzzy completion time of 10×10 FJSSP 2

Job 1	10 13 16	14 20 25	24 32 38	29 38 45	37 49 62	44 57 74	70 93 111	75 99 118	77 102 123	88 118 142
Job 2	13 17 21	25 32 43	30 40 52	43 59 72	48 65 81	55 76 93	64 89 107	72 101 123	74 105 129	78 112 139
Job 3	14 20 23	24 33 37	29 40 45	32 44 51	36 51 59	39 56 66	42 60 72	43 62 76	48 69 85	57 80 98
Job 4	27 36 44	28 38 48	34 46 58	35 49 63	42 57 74	47 65 84	56 75 98	60 82 106	64 89 116	68 92 121
Job 5	17 24 28	25 35 42	35 49 59	40 56 68	42 60 73	43 63 78	52 71 88	55 75 94	64 86 107	66 89 110
Job 6	4 7 9	14 19 24	17 23 29	27 37 47	32 43 56	42 57 72	52 69 87	60 78 99	65 86 108	69 93 118
Job 7	8 12 13	10 16 19	21 30 37	26 37 46	30 42 54	34 47 61	41 57 72	51 68 84	61 81 99	70 93 112
Job 8	23 29 36	28 35 45	29 38 49	38 51 61	42 57 76	49 68 90	56 79 103	76 102 122	84 113 135	91 122 148
Job 9	2 4 6	6 10 14	28 37 47	32 43 56	38 51 65	46 63 79	51 72 94	60 83 108	61 85 111	79 107 128
Job 10	5 8 9	11 16 18	21 28 36	27 37 48	34 48 61	51 68 82	58 78 93	61 83 100	64 87 106	72 98 121

Table 17 The fuzzy completion time of 10×10 FJSSP 3

Job 1	15 23 31	25 35 44	29 42 54	34 50 63	44 62 79	54 73 92	58 80 102	62 85 110	66 91 117	75 103 130
Job 2	10 11 14	19 23 28	26 33 42	33 44 54	35 48 60	43 58 74	50 69 86	58 80 100	64 89 110	74 103 125
Job 3	5 7 10	11 16 22	12 19 26	19 29 38	27 40 50	29 44 55	33 49 62	40 57 73	49 70 87	57 82 102
Job 4	17 25 34	27 36 48	48 64 86	51 69 93	55 74 99	63 84 111	65 87 115	67 90 120	73 98 131	79 106 142
Job 5	4 7 10	9 13 17	19 24 34	21 27 38	31 42 51	36 50 67	41 57 75	42 59 79	45 63 85	47 67 91
Job 6	12 19 26	43 56 65	46 61 71	51 74 93	59 83 106	68 95 120	72 102 128	73 104 132	75 108 137	81 117 149
Job 7	26 31 40	28 34 44	37 44 56	46 54 67	50 59 73	53 64 80	54 67 84	56 71 91	64 81 104	71 91 115
Job 8	7 11 15	16 25 31	21 31 40	29 40 53	35 49 65	41 57 75	47 66 86	48 68 90	56 80 104	62 89 116
Job 9	4 7 10	7 12 16	13 21 26	16 26 32	25 37 44	31 44 56	35 50 65	37 52 69	46 61 76	56 73 91
Job 10	1 2 3	9 14 16	16 22 25	22 31 37	31 42 52	38 53 67	48 67 85	51 70 90	52 72 94	54 75 99

Table 18 The fuzzy completion time of 10×10 FJSSP 4

Job 1	7 11 12	12 18 21	16 23 29	23 34 44	33 44 57	40 53 69	46 60 78	48 63 82	52 70 90	54 74 95
Job 2	10 16 20	11 18 23	12 20 27	22 31 41	30 43 55	32 47 60	37 55 70	45 64 81	47 68 86	51 74 94
Job 3	25 33 39	33 42 52	40 51 62	43 56 69	53 69 86	55 73 92	63 82 105	66 86 111	67 88 114	71 93 121
Job 4	20 28 34	22 31 39	26 38 47	30 44 54	37 51 62	43 59 72	45 62 76	53 74 90	58 81 98	62 87 107
Job 5	14 19 21	18 26 30	24 35 42	26 39 47	27 42 52	36 52 64	46 63 76	51 70 85	59 80 98	63 86 107
Job 6	12 15 18	16 20 26	20 25 33	24 31 41	30 38 50	33 43 57	45 59 75	52 69 87	58 78 99	63 85 108
Job 7	15 22 28	20 28 37	29 39 46	37 48 56	46 61 70	50 66 78	57 76 90	60 80 96	69 92 111	74 100 121
Job 8	7 11 14	10 16 20	20 29 34	27 38 45	36 48 60	41 56 70	49 69 84	59 81 100	69 95 116	75 102 126
Job 9	2 3 5	16 25 32	34 45 61	61 83 101	64 87 107	65 90 111	66 92 114	69 97 120	74 105 129	82 116 143
Job 10	9 13 15	14 20 24	19 26 32	32 44 53	34 47 58	39 55 68	51 69 85	60 82 101	64 87 109	66 91 114

Furthermore, we also compute the completion time of every FJSSP, and they are shown in Tables 11, 12, 13, 14, 15, 16, 17, and 18, respectively. The term in the i_{th} row and j_{th} column represents the completion time of the jth operation of Job J_i. We can see that the completion times obtained by our method are generally reasonable.

6 Conclusion

In this paper, we introduce the raking concept among fuzzy numbers based on possibility and necessity measures to job-shop scheduling problems with fuzzy processing time and fuzzy due date, and on the bias of this, we propose several novel objective functions. Next, for simplifying the computation, we define the PLR fuzzy number to be the assumed type of fuzzy numbers involved in this paper, and successfully derived the analytic formulas of the objective functions, which are very useful for the implementation of the computer program. In order to obtain better results, we also design a DE algorithm to search the "optimal" schedule that maximizes our objective functions. To illustrate our approach, we list some experimental results, which show that our method is comparable with state-of-the-art methods. And it must be explained that the smallest makespan time of an FJSSP obtained by our method may be not the best, but the completion times of all the jobs are more reasonable due to its consideration of the due date of each job. This shows the potential application of possibility and necessity theory in JSSP.

In the future, to make the computation simpler, we may study some relationship between possibility and necessity measures and other variants of objective functions based on them. On the other hand, we can see that there exist high improvement spaces in the algorithm part from the experiments, especially the convergence; therefore we also may try other heuristic algorithms that widely adopted in the scheduling problems, and apply them to the fuzzy job-shop scheduling problems.

References

1. Blackstone JH, Phillips D, Hogg GL (1982) A state-of-the-art survey of dispatching rules for manufacturing job-shop operations. Int J Prod Res 20(1):27–45
2. Grabot B, Geneste L (1994) Dispatching rules in scheduling: a fuzzy approach. Int J Prod Res 32(4):903–915
3. Erschler JF, Roubellat JP, Vernhes (1976) Finding some essential characteristics of the feasible solutions for a scheduling problem. Oper Res 24:774–783
4. Dubois D, Fargier H, Prade H (1995) Fuzzy constraints in job-shop scheduling. J Intell Manuf 6:215–234
5. L. Davis (1985) Job shop scheduling with genetic algorithms, in: Proceedings of the First International Conference on Genetic Algorithms, pp. 136–140.
6. Wang L, Zheng D-Z (2002) A modified genetic algorithm for job shop scheduling. Adv Manuf Technol 20:72–76
7. Byung Joo Park (2003) Hyung Rim Choi, Hyun Soo Kim, A hybrid genetic algorithm for the job shop scheduling problems. Computer & Industrial Engineering 45:597–613
8. Mahanim Omar, Adam Baharum, Yahya Abu Hasan (2006) A job-shop scheduling problem (JSSP) using Genetic algorithm (GA), Proceedings of the 2nd IMT-GT Regional Conference on Mathematics, Statistics and Applications Universiti Sains Malaysia, Penang, June 13–15, 2006
9. Slowinski R, Hapke M (2000) Scheduling under fuzziness. Physica Verlag, Heidelberg
10. Ishii H, Tada M, Masuda T (1992) Two scheduling problems with fuzzy due-dates. Fuzzy Sets Syst 46:339–347
11. Masatoshi Sakawa, Tetsuya Mori (1999) An efficient genetic algorithm for job-shop scheduling problems with fuzzy processing time and fuzzy due date. Comput Ind Eng 36:325–341
12. Masatoshi Sakawa, Ryo Kubota (2000) Fuzzy programming for multiobjective job shop scheduling with fuzzy processing time and fuzzy due date through genetic algorithms. Eur J Oper Res 120:393–407
13. Li FM, Zhu YL, Yin CW, Song XY (2005) Fuzzy programming for multi-objective fuzzy job shop scheduling with alternative machines through genetic algorithm. In: L. Wang K. Chen Y.S. Ong, Advance in Natural Computation. 992–1004
14. Lei DM (2008) Pareto archive particle swarm optimization for multi-objective fuzzy job shop scheduling problems. Int J Adv Manuf Technol 37:157–165
15. Itosh T, ishii H (1999) Fuzzy due-date scheduling problem with fuzzy processing time, int. Transactions in Operational Research 6:639–647
16. Dubis D, Prade H (1983) Ranking fuzzy numbers in the setting of possibility theory. Inf Sci 20:183–224
17. Chanas S, Kasperski A (2001) Minimizing maximum lateness in a single machine scheduling problem with fuzzy processing times and fuzzy due dates. Eng Appl Artif Intell 14:377–386
18. Chanas S, Kasperski A (2003) on two single machine scheduling problems with fuzzy processing times and fuzzy due dates. Eur J Oper Res 147:281–296
19. Zadeh LA (1965) Fuzzy sets. Inf Control 8:338–353
20. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11:341–359
21. Bean JC (1994) Genetic algorithms and random keys for sequencing and optimization. ORSA J Comput 6:154–160
22. Aldowaisan T, Allahverdi A (2003) New heuristics for no-wait flowshops to minimize makespan. Comput Oper Res 30:1219–1231