Computers & Industrial Engineering 58 (2010) 610–617

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Fuzzy job shop scheduling problem with availability constraints ☆

Deming Lei *

School of Automation, Wuhan University of Technology, 122 Luoshi Road, Wuhan, Hubei Province, People's Republic of China

A R T I C L E   I N F O

Article history:
Received 2 January 2009
Received in revised form 11 October 2009
Accepted 4 January 2010
Available online 11 January 2010

Keywords:
Fuzzy job shop scheduling
Periodic maintenance
Random key representation
Genetic algorithm

A B S T R A C T

This paper presents the fuzzy job shop scheduling problem with *availability constraints*. The objective is to find a schedule that maximizes the minimum agreement index subject to periodic maintenance, *non-resumable* jobs and fuzzy due-date. A random key genetic algorithm (RKGA) is proposed for the problem, in which a novel random key representation, a new decoding strategy incorporating maintenance operation and discrete crossover (*DX*) are used. RKGA is applied to some fuzzy scheduling problem with *availability constraints* and compared with other algorithms. Computational results show that RKGA performs better than other algorithms.

1. Introduction

Fuzzy job shop scheduling problem (FJSSP) is the extension of job shop scheduling problem (JSSP). FJSSP has become the main research topic of production scheduling and some results have been obtained in the past decade. Kuroda and Wang (1996) discussed the static JSSP and dynamic JSSP with fuzzy information. A branch-and-bound algorithm is used to solve the static JSSP and the methods for dynamic JSSP are also considered. Sakawa and Mori (1999) presented an efficient genetic algorithm (GA) by incorporating the concept of similarity among individuals. Sakawa and Kubota (2000) presented a GA for multi-objective JSSP with fuzzy processing time and fuzzy due-date. Song, Zhu, Yin, and Li (2006) presented a combined strategy of GA and ant colony optimization. They also designed a new neighborhood search method and an improved tabu search to intensify the local search ability of the hybrid algorithm. Niu, Jiao, and Gu (2008) proposed a particle swarm optimization with genetic operators (GPSO) *to minimize fuzzy makespan*. Lei (2008) proposed an efficient Pareto archive particle swarm optimization for FJSSP with three objectives for obtaining a set of Pareto optimal solutions.

FJSSP with alternative process plan also attracts some attentions. Lei and Guo (2008) presented a two-population GA by using two-string representation and two independent populations. Lei (in press) developed an efficient decomposition–integration genetic algorithm for minimizing fuzzy makespan. Li, Zhu, Yin, and Song (2005) proposed a GA by adopting two-chromosome presentation and the extended version of Giffler–Thompson Procedure (Giffler & Thompson, 1960).

Machine availability constraints mean that machines can be unavailable for preventive maintenance, periodic repair or random breakdown. These constraints have been considered in many scheduling problems, *such as single machine* (Chen, 2009; Graves & Lee, 1999), *parallel machines* (Liao, Shyur, & Lin, 2005) *and flow shop* (Allaoui & Artiba, 2006; Allaoui, Lamouri, Artiba, & Aghezzaf, 2008). *With respect to JSSP,* Mauguière, Billaut, and Bouquard (2005) suggested a branch-and-bound algorithm to solve the single machine and JSSP with availability constraints and Gao, Gen, and Sun (2006) presented a hybrid GA to solve flexible JSSP with non-fixed availability constraints.

Machine availability constraints problems have been investigated extensively; however, for FJSSP, literature still assumes that machines are always available and availability constraints are seldom involved. The combination of FJSSP and availability constraints will make the considered problems be more close to the real-world situations. In this study, *FJSSP with preventive maintenance (FJSSP-PM)* is considered and an efficient RKGA is presented, which uses a random key representation and a decoding strategy incorporating maintenance operations. The chromosome of the presentation can be directly converted into an ordered operation list and a schedule is directly built by using the operation list. No special genetic operators are required and no illegal individuals occur in the search process of RKGA.

The remainder of the paper is organized as follows. The operations on fuzzy numbers are introduced in Section 2 and the problem formulation is done in Section 3. Section 4 describes a random key scheduling algorithm. Numerical test experiments on the proposed algorithm are reported in Section 5 and the conclusions are summarized in the final section.

☆ This manuscript was processed by Area Editor T.C. Edwin Cheng, ScD, PhD.

* Tel.: +86 2786534910.
E-mail address: deminglei11@163.com.

0360-8352/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cie.2010.01.002

2. Operations on fuzzy numbers

In fuzzy context, some operations of fuzzy numbers are required to be redefined to build a schedule. These operations involve addition operation and max operation of fuzzy numbers as well as the ranking methods of fuzzy numbers. Addition operation is used to calculate the fuzzy completion time of operation. Max operation is to determine the fuzzy beginning time of operation.

Triangular fuzzy numbers (TFN) is often used to represent processing time of operation, so only the operations of TFNs are described.

For two TFNs $\tilde{s} = (s_1, s_2, s_3)$ and $\tilde{t} = (t_1, t_2, t_3)$, the addition of them is shown by the following formula:

$$\tilde{s} + \tilde{t} = (s_1 + t_1, s_2 + t_2, s_3 + t_3) \quad (1)$$

Generally, the ranking method is to compare the maximum fuzzy completion time, in this paper, the following ranking method (Sakawa & Kubota, 2000) is used to define a new criterion for max operation.

The following three criteria are adopted to rank TFNs:

Criterion 1: The greatest associate ordinary number c_1 as the first criterion is applied to rank them.
Criterion 2: If c_1 cannot rank them, then c_2 is chosen as the second criterion.
Criterion 3: If c_1 and c_2 do not still rank them, the difference of spreads c_3 is used as the third criterion.

For TFN $\tilde{s}$, $c_1(\tilde{s}) = (s_1 + 2s_2 + s_3)/4$, $c_2(s) = s_2$ and $c_3(s) = s_3 - s_1$. If TFNs have same value of $c_1(c_2)$, then $c_1(c_2)$ cannot rank them. These criteria can be used to rank all TFNs. For example, for $\tilde{s}$ and $\tilde{t}$, if $c_1(\tilde{s}) > c_1(\tilde{t})$, then $\tilde{s} > \tilde{t}$; If $c_1(\tilde{s}) = c_1(\tilde{t})$ *and* $c_2(\tilde{s}) > c_2(\tilde{t})$, then $\tilde{s} > \tilde{t}$; if $c_1(\tilde{s}) = c_1(\tilde{t}), c_2(\tilde{s}) = c_2(\tilde{t})$ and $c_3(\tilde{s}) > c_3(\tilde{t})$, then $\tilde{s} > \tilde{t}$.

For $\tilde{s} = (s_1, s_2, s_3)$ and $\tilde{t} = (t_1, t_2, t_3)$, membership function $\mu_{\tilde{s}\vee\tilde{t}}(z)$ of $\tilde{s} \vee \tilde{t}$ is defined as follows:

$$\mu_{\tilde{s}\vee\tilde{t}}(z) = \sup_{z=x\vee y} \min(\mu_{\tilde{s}}(x), \mu_{\tilde{t}}(y)) \quad (2)$$

In this paper, the max of two TFNs $\tilde{s}$ and $\tilde{t}$ is approximated with the following criterion:

$$\text{If } \tilde{s} > \tilde{t}, \text{ then } \tilde{s} \vee \tilde{t} = \tilde{s}; \quad \text{else } \tilde{s} \vee \tilde{t} = \tilde{t} \quad (3)$$

The criterion $\tilde{s} \vee \tilde{t} \approx (s_1 \vee t_1, s_2 \vee t_2, s_3 \vee t_3)$ is used by Sakawa and Mori (1999) and named Sakawa criterion for simplicity. Sakawa criterion has been extensively applied to build a complete *schedule* of the fuzzy problem.

For TFNs $\tilde{s}$ and $\tilde{t}$ *with* $t_2 \neq s_2$, Fig. 1 describe four possible circumstances, in which edge ss_1 may intersect with tt_1 and ss_3 may intersect with tt_3. Fig. 1(2) is first discussed. Fig. 2 shows the comparison between two criteria. For triangle 1 is superfluous and 2 is lost, the approximation error is defined to be the ratio of the area of triangles 1 and 2 to the area of $\tilde{s} \vee \tilde{t}$. Two cases are as follows:

If $\tilde{t} > \tilde{s}$, then $s_1 - t_1 \leqslant t_2 - s_2$ or $s_3 - t_3 \leqslant t_2 - s_2$. If $s_1 - t_1 \leqslant t_2 - s_2$, the area of triangle 1 in Fig. 2(a1) is smaller than or equal to a half of the area of $\triangle tt_1s_1$, that is, triangle 1 in (a1) has smaller area than or identical area with $\triangle tAs_1$, which is the triangle 2 of Fig. 2(b); similarly, if $s_3 - t_3 \leqslant t_2 - s_2$, the area of 2 in Fig. 2(a1) is also smaller than or equal to that of $\triangle tBs_3$, which the 1 of Fig. 2(b).

If $\tilde{t} < \tilde{s}$, then $s_1 - t_1 \geqslant t_2 - s_2$ or $s_3 - t_3 \geqslant t_2 - s_2$. If $s_1 - t_1 \geqslant t_2 - s_2$, triangles 1 and 3 in Fig. 2(a2) has smaller area or identical area with 2 in Fig. 2(b), triangle 1 of Fig. 2(a2) obviously has less area than 2 of (b). Similarly, if $s_3 - t_3 \geqslant t_2 - s_2$, triangle 2 of Fig. 2(a2) has smaller area than triangle 1 of Fig. 2(b).

In each case, at least one of two conditions is met, if two conditions both occur, the approximation error of the new criterion is smaller than that of Sakawa criterion; if only one condition is met, two criteria have the similar approximation error.

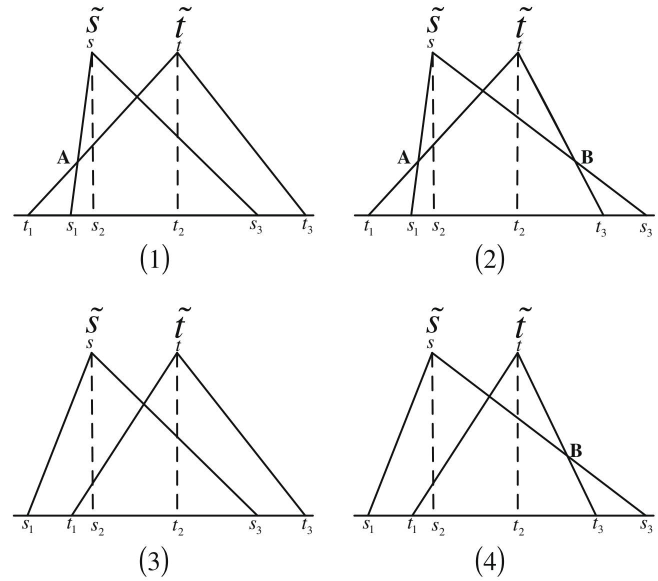

Fig. 1. Four circumstances of two TFNs.

For Fig. 1(1), only crossing point A exists and $s_3 - t_3 \leqslant t_2 - s_2$, if $\tilde{t} > \tilde{s}$ and $s_1 - t_1 \leqslant t_2 - s_2$, the new criterion has smaller approximation error than Sakawa criterion, if $\tilde{t} < \tilde{s}$ or $\tilde{t} > \tilde{s}$ and $s_1 - t_1 > t_2 - s_2$, two criteria have close approximation error. For Fig. 1(4), the similar conclusions also can be obtained. For Fig. 1(3), two criteria have no approximation error.

For TFNs $\tilde{s}$ and $\tilde{t}$ with $t_2 = s_2$, there are also four circumstances. When $s_3 \geqslant t_3$ and $s_1 \geqslant t_1$ or $t_3 > s_3$ and $t_1 > s_1$, two criteria approximate the real max. When $s_3 \geqslant t_3$ and $s_1 < t_1$ or $t_3 > s_3$ and $s_1 \geqslant t_1$, the new criterion has the approximation error and Sakawa criterion does not. When building a schedule, for $\tilde{s}$ and $\tilde{t}$, t_2 is not frequently equal to s_2. The new criterion may have more approximation error than Sakawa criterion in at most one case. The approximation error affects the approximate value of completion times. The less the error of the approximate max, the closer the approximate value of completion time approaches to the real value. Thus, the new criterion can perform better than Sakawa criterion on fuzzy scheduling.

3. Problem description

$n \times m$ FJSSP-PM can be described as follows: given n jobs J_i $(i = 1, 2, \ldots, n)$, each job is composed of several operations processed on machines M_k $(k = 1, 2, \ldots, m)$. Operation o_{ij} indicates the jth operation of J_i and its processing time is represented as TFN $p_{ij} = (a_{ij}^1, a_{ij}^2, a_{ij}^3)$. The due-date of job J_i is trapezoid $d_i = (e_i^1, e_i^2, d_i^1, d_i^2)$. Some maintenance operations occur on each machine during the planning horizon. Each maintenance operation has a fixed predefined time interval. If a job must be reprocessed fully after maintenance if its processing is interrupted by maintenance activity on machine, the job is *non-resumable*. A *resumable* job may continue its processing when machine become available if its processing cannot be completed before the unavailable period. In this study, only *non-resumable* jobs are considered.

FJSSP-PM is the variant of FJSSP and the constraints of FJSSP are still suitable to FJSSP-PM, such as,
Each operation is processed by only one machine,
Only one operation can be processed on its machine at a time,
Each operation cannot be commenced if the precedent operation is still being processed,