G Model
ASOC 3684 1–13
ARTICLE IN PRESS
Applied Soft Computing xxx (2016) xxx–xxx
Contents lists available at ScienceDirect
Applied Soft Computing
journal homepage: www.elsevier.com/locate/asoc

Robust multiobjective optimisation for fuzzy job shop problems

Q1 Juan José Palacios [a], Inés González-Rodríguez [b,*], Camino R. Vela [a], Jorge Puente [b]

[a] Dept. of Computing, University of Oviedo, Campus de Viesques S/N, 33204 Gijón, Spain
[b] Dept. of Mathematics, Statistics and Computing, University of Cantabria, Av. Los Castros S/N, 39005 Santander, Spain

A R T I C L E I N F O

Article history:
Received 16 July 2015
Received in revised form 13 May 2016
Accepted 1 July 2016
Available online xxx

Keywords:
Scheduling
Fuzzy sets
Metaheuristics
Robustness
Multiobjective optimisation

A B S T R A C T

In this paper we tackle a variant of the job shop scheduling problem with uncertain task durations modelled as fuzzy numbers. Our goal is to simultaneously minimise the schedule's fuzzy makespan and maximise its robustness. To this end, we consider two measures of solution robustness: a predictive one, prior to the schedule execution, and an empirical one, measured at execution. To optimise both the expected makespan and the predictive robustness of the fuzzy schedule we propose a multiobjective evolutionary algorithm combined with a novel dominance-based tabu search method. The resulting hybrid algorithm is then evaluated on existing benchmark instances, showing its good behaviour and the synergy between its components. The experimental results also serve to analyse the goodness of the predictive robustness measure, in terms of its correlation with simulations of the empirical measure.

1. Introduction

Q2 Operations scheduling is one of the most critical issues in manufacturing and production systems, as well as in information processing environments. The importance of scheduling as a research topic is undeniable, both as a source of interesting complex combinatorial optimisation problems and as a field with multiple real applications in industry, finance, welfare, etc. In particular, shop problems in their multiple variants—for instance, incorporating flexibility, operators or energy efficiency—can model many situations which naturally arise in manufacturing environments [30,34,52,60].

Unfortunately, classical scheduling cannot model many practical situations due to the fact that project decisions usually have to be made in advance, when activity durations are still highly uncertain. A great variety of approaches have been considered to deal with these real-life situations, as can be seen for instance in [36]. Fuzzy sets have contributed to enhancing the applicability of scheduling, helping to bridge the gap between classical techniques and real-world user needs. They have been used both for handling flexible constraints and uncertain data [20,65,69]. They are also demonstrating to be an interesting tool for improving solution robustness, a much-desired property in real-life applications [43,56].

When there is uncertainty in some of the input data, robustness becomes an important factor to be taken into account. The better-known approaches to robustness, based on min-max or min-max regret criteria, aim at constructing solutions having the best possible performance in the worst case motivated by practical applications where an anticipation of the worst case is crucial [2,43,68]. However, this kind of approach may be deemed as too conservative in some cases where the worst case is not that critical and instead an overall acceptable performance is preferred [42]. Here, we take an approach that might be more adequate in these situations.

Clearly, when the improvement in robustness must not be obtained at the cost of loosing performance quality in the solutions, we face a bi-objective scheduling problem. In general there is a growing interest in multiobjective optimisation for scheduling and, given its complexity, in the use of metaheuristic techniques to solve these problems, as can be seen in [5,14,26,37] among others. In particular, the multiobjective fuzzy job shop problem is receiving an increasing attention, mostly to optimise objective functions related to makespan and due-date satisfaction. Existing proposals include genetic algorithms [27,64], differential evolution algorithms [38], or hybrid strategies like the genetic simulated-annealing algorithm from [68]. Interestingly, the latter contemplates finding both robust and satisfactory schedules, although the robustness optimisation criterion is based on the worst-case approach.

In the single-objective case, it is common to hybridise evolutionary algorithms with local search to produce memetic algorithms,

* Corresponding author.
E-mail addresses: juanjose.palonso@gmail.com (J. José Palacios), gonzalezri@unican.es (I. González-Rodríguez), crvela@uniovi.es (C.R. Vela), puente@uniovi.es (J. Puente).

http://dx.doi.org/10.1016/j.asoc.2016.07.004

Please cite this article in press as: J. José Palacios, et al., Robust multiobjective optimisation for fuzzy job shop problems, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2016.07.004

which benefit from the synergy between their components to provide a better search capacity. It is possible to find various multiobjective memetic algorithms (MOMAs) in the literature, some of them applied to manufacturing problems [12,13,39,62,51]. However, according to [49], the number of multiobjective local search algorithms proposed so far is still reduced. In fact, the main difficulty in designing multiobjective memetic algorithms is the implementation of the local search, which essentially is a single-objective optimisation technique. Some multiobjective memetic algorithms resort to scalarisation methods to guide the local search towards concrete areas of the Pareto front, for instance [39,40,51]. Other MOMAs use instead acceptance criteria for the local search based on Pareto-dominance [44,47,62]. In most cases, the local search method uses a hill-climbing strategy. On the other hand, in [41] a multiobjective flexible job shop scheduling problem is solved not with an evolutionary algorithm but a combination of tabu search, based on the TSAB from [53] and only for makespan minimisation, and path relinking.

In the following we tackle the fuzzy job shop problem, where uncertainty in task durations is modelled using fuzzy numbers. After introducing the problem in Section 2, in Section 3 we give a precise definition of two robustness measures based on the average behaviour across all possible cases: an a-priori measure to be evaluated in constant time from the predictive schedule, and an a-posteriori measure, to be evaluated at the moment of executing the schedule or, in the absence of a real execution, by a surrogate obtained with Monte-Carlo simulations. In Section 4 we propose a hybrid method to find non-dominated solutions with respect to the makespan—the total time needed to complete all jobs—and the a-priori robustness measure. This algorithm combines a multi-objective evolutionary algorithm with a new dominance-based tabu search as iterative improvement method. In Section 5, we report and analyse results of an experimental study which contemplates the synergy between the method components, the performance of the proposed method and the relation between both robustness measures. Finally, some conclusions are given in Section 6.

2. The fuzzy job shop scheduling problem

The *job shop scheduling problem*, also denoted *JSP*, consists in scheduling a set of jobs $\{J_1, \ldots, J_n\}$ on a set of physical resources or machines $\{M_1, \ldots, M_m\}$, subject to a set of constraints. There are *precedence constraints*, so each job J_i, $i = 1, \ldots, n$, consists of m tasks $\{\theta_{i1}, \ldots, \theta_{im}\}$ to be sequentially scheduled. Also, there are *capacity constraints*, whereby each task θ_{ij} requires the uninterrupted and exclusive use of one of the machines for its whole processing time p_{ij}. A feasible schedule is an allocation of starting times for each task such that all constraints hold. The objective is to find a schedule which is *optimal* according to some criterion, most commonly that the *makespan* is minimal.

2.1. Uncertain durations

In real-life applications, it is often the case that the exact processing time of tasks is not known in advance. However, based on previous experience, an expert may be able to estimate, for instance, an interval of possible values for the processing time or its most typical value, and he/she may even be able to assess whether some values in the interval appear to be more plausible than others. This naturally leads to modelling such durations using fuzzy intervals or fuzzy numbers, which have been extensively studied in the literature (cf. [22]). A *fuzzy interval* A is a fuzzy set on the reals (with membership function $\mu_A : \mathbb{R} \to [0, 1]$) such that its *α-cuts* $A_\alpha = \{u \in \mathbb{R} : \mu_A(u) \geq \alpha\}$, $\alpha \in (0, 1]$, are intervals. A fuzzy interval is a *fuzzy number* if its α-cuts (denoted $[\underline{a}_\alpha, \overline{a}_\alpha]$) are closed, its *support* $A_0 = \{u \in \mathbb{R} : \mu_A(u) > 0\}$ is compact (closed and bounded) and there is a unique modal value u^*, $\mu_A(u^*) = 1$. Clearly, real numbers can be seen as a particular case of fuzzy ones.

The simplest model of fuzzy interval is a *triangular fuzzy number* or *TFN*, using an interval $[a^1, a^3]$ of possible values and a modal value a^2 in it. For a TFN A, denoted $A = (a^1, a^2, a^3)$, the membership function takes the following triangular shape:

$$\mu_A(x) = \begin{cases} \dfrac{x - a^1}{a^2 - a^1} & : a^1 \leq x \leq a^2 \\ \dfrac{x - a^3}{a^2 - a^3} & : a^2 < x \leq a^3 \\ 0 & : x < a^1 \quad \text{or} \quad a^3 < x \end{cases} \tag{1}$$

If TFNs are to be used to extend the job shop to handle uncertainty, two issues must be addressed: first, how the arithmetic operations of addition and maximum are to be extended to work with TFNs and, second, the precise meaning of "minimal makespan" when such makespan is a TFN.

2.1.1. Arithmetic of TFNs

In the fuzzy job shop, we essentially need two arithmetic operations on fuzzy numbers, the sum and the maximum. These are obtained by extending the corresponding operations on real numbers using the *Extension Principle*. However, computing the resulting expression is cumbersome, if not intractable; also, the set of TFNs is not always closed under the resulting operation. For the sake of simplicity and tractability of numerical calculations, it is fairly common in the literature, following [25], to approximate the results of these operations by interpolation, evaluating only the operation on the three defining points of each TFN. It turns out that the sum and its approximation coincide, so for any pair of TFNs A and B:

$$A + B = (a^1 + b^1, a^2 + b^2, a^3 + b^3). \tag{2}$$

Regarding the maximum, we have:

$$\max(A, B) \approx \max_I(A, B) = (\max(a^1, b^1), \max(a^2, b^2), \max(a^3, b^3)). \tag{3}$$

The approximation $\max_I$ has been widely used in the scheduling literature, from earlier works [25,46] to more recent ones [68,55], to mention but a few. Additionally, some arguments can be given to support this approximation.

First, for any two fuzzy numbers A and B, if f is a bivariate continuous isotonic function, that is, $f : \mathbb{R}^2 \to \mathbb{R}$ such that for any $u \geq u'$ and $v \geq v'$ it holds that $f(u, v) \geq f(u', v')$, then $F = f(A, B)$ is another fuzzy number such that $F_\alpha = [f(\underline{a}_\alpha, \underline{b}_\alpha), f(\overline{a}_\alpha, \overline{b}_\alpha)]$. Computing $f(A, B)$ is then equivalent to computing f on every α-cut. In particular, the addition and the maximum are continuous isotonic functions, so they can be calculated by evaluating two sums or maxima of real numbers for every value $\alpha \in [0, 1]$. If seems then natural to approximate the maximum by the TFN that results from using linear interpolation, evaluating F_α only for certain values of α (as proposed for 6-point fuzzy numbers in [25]). Given that the defining values (a^1, a^2, a^3) of a TFN A are such that $A_0 = [a^1, a^3]$ and $A_1 = [a^2, a^2]$, $\max_I$ in (3) corresponds to such an interpolation for $\alpha = 0$ and $\alpha = 1$.

Secondly, if $F = \max(A, B)$ denotes the maximum of two TFNs A and B and $G = \max_I(A, B)$ its approximated value, then $F = G$ if A and B do not overlap and, in any case, it holds that $\forall \alpha \in [0, 1]$, $\underline{f}_\alpha \leq \underline{g}_\alpha, \overline{f}_\alpha \leq \overline{g}_\alpha$. The approximated maximum G is thus a TFN which may artificially increase the value of the actual maximum F, while maintaining the support and modal value, that is, $F_0 = G_0$ and $F_1 = G_1$.