Applied Soft Computing 13 (2013) 4750–4762

Contents lists available at ScienceDirect

Applied Soft Computing

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

A particle swarm optimization for a fuzzy multi-objective unrelated parallel machines scheduling problem

S.A. Torabi [a,*], N. Sahebjamnia [a], S.A. Mansouri [b], M. Aramon Bajestani [a]

[a] School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran
[b] Brunel Business School, Brunel University, Uxbridge, Middlesex, UK

ARTICLE INFO

Article history:
Received 21 November 2012
Received in revised form 27 June 2013
Accepted 30 July 2013
Available online 30 August 2013

Keywords:
Fuzzy mathematical programming
Unrelated parallel machine scheduling
Secondary resource constraints
Multiple objective particle swarm optimization

ABSTRACT

This paper proposes a novel multi-objective model for an unrelated parallel machine scheduling problem considering inherent uncertainty in processing times and due dates. The problem is characterized by non-zero ready times, sequence and machine-dependent setup times, and secondary resource constraints for jobs. Each job can be processed only if its required machine and secondary resource (if any) are available at the same time. Finding optimal solution for this complex problem in a reasonable time using exact optimization tools is prohibitive. This paper presents an effective multi-objective particle swarm optimization (MOPSO) algorithm to find a good approximation of Pareto frontier where total weighted flow time, total weighted tardiness, and total machine load variation are to be minimized simultaneously. The proposed MOPSO exploits new selection regimes for preserving global as well as personal best solutions. Moreover, a generalized dominance concept in a fuzzy environment is employed to find locally Pareto-optimal frontier. Performance of the proposed MOPSO is compared against a conventional multi-objective particle swarm optimization (CMOPSO) algorithm over a number of randomly generated test problems. Statistical analyses based on the effect of each algorithm on each objective space show that the proposed MOPSO outperforms the CMOPSO in terms of quality, diversity and spacing metrics.

1. Introduction

Parallel machine scheduling problem (PMSP) is concerned with allocating a set of jobs to a number of parallel machines in order to meet customer's requirements. In the literature, the studies on PMSP can be generally classified into the three categories [1]: identical, uniform and unrelated parallel machine scheduling problem. Among these categories, unrelated PMSP (UPMSP) represents a generalization of the other two categories in which different machines perform the same function but have different processing capabilities or capacities. However, dealing with real-life UPMSPs is a major challenge for researchers and practitioners, due not only to the fact that they are mostly NP-hard except for the objective of minimizing flow time (see [2]), but also more importantly to their special characteristics/requirements in practice. This paper focuses on an UPMSP; which has been addressed much less than the identical and uniform PMSPs in the literature especially when setup times are taken in account (see [3,4]).

Kamath [5] present a survey on UPMSP involving makespan considerations. Li et al. [6] and Chyu and Chang [7] examine the UPMSP to minimize the mean flow time, while Cao et al. [8] study the same problem by considering the total cost functions. Lin et al. [9] address the problem to minimize the total tardiness by including the machine-dependent sequence-dependent setup times.

Researchers have addressed the more practical versions of UPMSPs by considering other features of real scheduling problems such as secondary resource constraints, non-zero ready times and so on. Chen [10] develops a heuristic method to minimize makespan in an UPMSP with different die types as a secondary resource constraint. Chen and Wu [11], and Chen [12], solve an UPMSP with auxiliary equipment constraints as secondary resource constraints. Lamothe et al. [13] propose a new model in order to minimize total tardiness by considering specific constraints such as secondary resources.

As another feature, Bang [14] develops an algorithm for UPMSP with sequence-dependent set-ups and distinct ready times while minimizing total tardiness. Chen [15] develops an iterated local search to minimize the total weighted number of late jobs on UPMSP without preemption, with sequence dependent setup times and ready times.

Chang et al. [16] have proven that a single-machine, total weighted tardiness minimization problem with static job releases and static machine availability and weights of all jobs being equal, is strongly NP-hard. Clearly, the single machine case considered by them is a special case of the sequence-dependent unrelated parallel machine-scheduling problem considered in this paper. Therefore, the problem investigated in this paper is also strongly NP-hard.

* Corresponding author. Tel.: +98 21 88335605.
E-mail address: satorabi@ut.ac.ir (S.A. Torabi).

1568-4946/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2013.07.029

Amiri and Khanmohammadi [17] classified the proposed methods for solving such problems into two categories as classic and intelligent algorithms (IA). While the Dynamic programming [18] and Lagrangian relaxation [19] are classified as former categories, genetic algorithm (GA) [20], PSO [21], ant colony optimization (ACO) [22], neural network (NN) [23–26] and various hybrid IAs [27] are categorized as the later one.

Thus, it is unlikely that a polynomial-time algorithm could be developed capable of determining an optimal solution for such UPMSPs like our concerned problem in practice. Hence, many researchers usually apply metaheuristic methods to deal with such problems (see e.g. [28–32]). Among them, Vallada and Ruiz [28] develop a genetic algorithm including a fast local search and a local search enhanced crossover operator. Two versions of the algorithm are proposed after extensive calibrations by using the design of experiments (DOE) approach. They conclude that their methods show an excellent performance when evaluating them over a comprehensive benchmark set of instances. Bozorgirad and Logendran [29] address a sequence-dependent group-scheduling problem on a set of unrelated-parallel machines where the run time of each job differs on different machines. They developed a meta-heuristic algorithm based on Tabu search that could find solutions at least as well as CPLEX but in drastically shorter computational time. Chen [30] considers unrelated parallel machine scheduling with sequence-dependent setup times and unequal ready times aiming to minimize the weighted number of tardy jobs. He proposes an Iterated hybrid meta-heuristic algorithm, which begins with effective initial solution generators to generate initial feasible solutions; then, hybrid meta-heuristics are applied to improve the initial solutions, integrating the principles of the variable neighborhood descent approach and tabu search. Arnaout et al. [31] introduce an enhanced ant colony optimization (ACO) Algorithm and compare its performance to other existing algorithms including ACO I, MetaRaPS, and simulated annealing (SA) on unrelated parallel machines with sequence-dependent setup times. Recently, Ruiz-Torres et al. [32] present a new unrelated parallel machine-scheduling problem with deteriorating effect and the objective of makespan minimization. They design a set of list scheduling algorithms and simulated annealing meta-heuristics and the effectiveness of these approaches are evaluated by solving a large number of benchmark instances.

Motivated by a real case study in a wire and cable manufacturer, this research deals with scheduling of a number of jobs on an unrelated parallel machine system with secondary resource constraints in which each job can only be processed if its required machine and other secondary resources (e.g. labor, tools, etc.) are available. Each job has a due date and requires a single operation with non-zero ready time. Moreover, when different jobs received from customers compete for the same resource (i.e. a set of unrelated parallel machines), in addition to agreeing on a specific due date it is customary to specify a weight or degree of importance based upon the job and the kind of relationship that exists between the customer and the producer. In addition, the setup time required for a job on a machine is dependent upon the degree of similarity or dissimilarity that exists between this job and its immediately preceding job. Thus, the problem we consider here is a sequence-dependent UPMSP aiming to minimize the total weighted flow time, the total weighted tardiness, and the machine load variation of all jobs released during the planning horizon simultaneously.

The main contributions of this paper can be highlighted as follows:

- Proposing a new fuzzy UPMSP model addressing the non-zero ready times, sequence and machine-dependent setup times, and secondary resource constraints for jobs simultaneously to cope with imprecise/ambiguous nature of critical input data as well as some real constraints in practice.
- Proposing a novel MOPSO solution method with a new solution representation modification procedure to reduce the cost of algorithm by joining the secondary resource and machine constraints' presentations, and obtaining discrete permutation from a continuous representation.
- Exploiting new selection regimes within the proposed MOPSO for preserving *gbest* as well as p_{best} solutions.
- Employing a generalized dominance concept in a fuzzy environment to find locally Pareto-optimal frontier.
- Applying a fuzzy distance measure for calculating distance between fuzzy completion times and due dates in order to schedule jobs as close as possible to their due dates.
- Studying the effect of two competent methods (i.e. MOPSO and a conventional MOPSO called CMOPSO) and objective spaces (test problems) on the performance metrics through a novel statistical analysis.

The remainder of this paper is organized as follows. The relevant literature in UPMSP is more elaborated in Section 2 by considering the uncertainty issue. In Section 3, we define our notation, state our assumptions and propose a new fuzzy mixed-integer non-linear programming model for the proposed UPMSP problem. An introduction to particle swarm optimization along with particle swarm optimization-based scheduling literature is given in Section 4. After presenting appropriate solution representation and developing a new formulation to obtain distance between two arbitrary fuzzy numbers, we propose a novel particle swarm optimization algorithm based on the new selection regimes to solve the developed fuzzy UPMSP model in Section 5. The proposed MOPSO is validated through two classes of numerical experiments in Section 6. Performance of the proposed MOPSO is compared against a conventional MOPSO (CMOPSO) over a number of randomly generated test problems in Section 7. Finally, some concluding remarks and further research directions are provided in Section 8.

2. PMSM under uncertainty

The literature review reveals that majority of previous research in this area has focused on developing heuristic methods to find acceptable solutions in a reasonable time relying on the following assumptions:

- All of input parameters (e.g. processing times and due dates) are deterministic;
- Decision variables are deterministic such as jobs' completion times;
- Machines are the only limited resource when processing the jobs.

There are two major drawbacks for making deterministic assumptions: (1) in many real cases, there is no enough historical data for uncertain parameters, thus, we can rarely obtain the actual value of these parameters and (2) due to inherent imprecision (ambiguousness) in the input data, we cannot measure their exact values. Thereby, because of incompleteness and/or unavailability of required data, it seems to be more realistic to consider inherent imprecision/fuzziness in the critical parameters (such as processing times and due dates). Similarly, many researchers have applied fuzzy based approaches to deal with real empirical or practical application (see e.g. [33–45]). In this way, an imprecise processing time with some tolerance values (e.g. a processing time as 40 + - 5 time units) could be modeled as a fuzzy number representing the incompleteness and imprecision of required information.