European Journal of Operational Research 211 (2011) 612–622

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Innovative Applications of O.R.

A genetic algorithm for the unrelated parallel machine scheduling problem with sequence dependent setup times

Eva Vallada *, Rubén Ruiz

Grupo de Sistemas de Optimización Aplicada, Instituto Tecnológico de Informática, Universidad Politécnica de Valencia, Edificio 7A, Camino de Vera S/N, 46021 Valencia, Spain

ARTICLE INFO

Article history:
Received 25 June 2009
Accepted 4 January 2011
Available online 9 January 2011

Keywords:
Parallel machine
Scheduling
Makespan
Setup times

ABSTRACT

In this work a genetic algorithm is presented for the unrelated parallel machine scheduling problem in which machine and job sequence dependent setup times are considered. The proposed genetic algorithm includes a fast local search and a local search enhanced crossover operator. Two versions of the algorithm are obtained after extensive calibrations using the Design of Experiments (DOE) approach. We review, evaluate and compare the proposed algorithm against the best methods known from the literature. We also develop a benchmark of small and large instances to carry out the computational experiments. After an exhaustive computational and statistical analysis we can conclude that the proposed method shows an excellent performance overcoming the rest of the evaluated methods in a comprehensive benchmark set of instances.

1. Introduction

In the unrelated parallel machine scheduling problem, there is a set $N = \{1,\ldots,n\}$ of n jobs that have to be processed on exactly one machine out of a set $M = \{1,\ldots,m\}$ of m parallel machines. Therefore, each job is made up of one single task that requires a given processing time. Machines are considered unrelated when the processing times of the jobs depend on the machine to which they are assigned to. This is the most realistic case which is also a generalisation of the uniform and identical machines cases. Moreover, the consideration of setup times between jobs is very common in the industry. The setup times considered in this paper are both sequence and machine dependent, that is, setup time on machine i between jobs j and k is different than setup time on the same machine between jobs k and j. In addition, setup time between jobs j and k on machine i is different than setup time between jobs j and k on machine i'.

The most studied optimisation criterion is the minimisation of the maximum completion time of the schedule, a criteria that is known as makespan or $C_{\max}$. Summing up, in this paper we deal with the unrelated parallel machine scheduling problem with sequence dependent setup times denoted as $R/S_{ijk}/C_{\max}$ (Pinedo, 2008). We propose and evaluate a genetic algorithm that includes a fast local search and a local search enhanced crossover operator among other innovative features that, as we will see, result in a state-of-the-art performance for this problem.

* Corresponding author. Tel.: +34 96 387 70 07x74911; fax: +34 96 387 74 99.
E-mail addresses: evallada@eio.upv.es (E. Vallada), rruiz@eio.upv.es (R. Ruiz).

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.01.011

The remainder of this paper is organised as follows: in Section 2 we review the literature on this problem. In Section 3 a Mixed Integer Programing (MIP) model formulation is presented. In Section 4 we describe in detail the proposed genetic algorithm and preliminary computational results. In Section 5, a Design of Experiments approach is applied in order to calibrate the genetic algorithm. Results of a comprehensive computational and statistical evaluation are reported in Section 6. Finally, conclusions are given in Section 7.

2. Literature review

Parallel machine scheduling problems have been widely studied in the past decades. However, the case when the parallel machines are unrelated has been much less studied. Additionally, the consideration of sequence dependent setup times between jobs has not been considered until recently. In Allahverdi et al. (2008) a recent review of scheduling problems with setup times is presented, including the parallel machine case. In this section we focus our attention on the available algorithms for the parallel machine scheduling problems considering setup times.

In the literature, we can find several heuristic and metaheuristic algorithms for the mentioned problem. However, most of them are focused on the identical parallel machine case. In Guinet (1993), a heuristic is proposed for the identical parallel machines case with sequence dependent setup times and the objective to minimise the makespan. A tabu search algorithm is given in Franca et al. (1996) with the objective to minimise the total completion time. A three phase heuristic is proposed by Lee and Pinedo (1997) for the same problem with sequence dependent setup times (independent of the machine) and the objective to minimise the sum of weighted tardiness of the jobs. In Kurz and Askin (2001), the authors proposed several heuristics and a genetic algorithm to minimise makespan. Other heuristics for the same problem are those proposed by Gendreau et al. (2001) and Hurink and Knust (2001). In both cases the objective is to minimise the makespan and in the latter case precedence constraints are also considered. In Eom et al. (2002) and Dunstall and Wirth (2005) heuristics are proposed for the family setup times case. In Tahar et al. (2006) a linear programming approach is proposed where job splitting is also considered. Anghinolfi and Paolucci (2007) and Pfund et al. (2008) present heuristic and metaheuristic methods for the same problem, respectively.

The unrelated parallel machines case with sequence dependent setup times has been less studied and only a few papers can be found in the literature. A tabu search algorithm is given in Logendran et al. (2007) for the weighted tardiness objective. Another heuristic for the unrelated parallel machine case with the objective to minimise weighted mean completion time is that proposed by Weng et al. (2001). Kim et al. (2002) proposed a simulated annealing method with the objective to minimise the total tardiness. In Kim et al. (2003) and Kim and Shin (2003) a heuristic and tabu search algorithm were proposed with the objective to minimise the total weighted tardiness and the maximum lateness, respectively. The same problem is also studied in Chen (2005), Chen (2006) and Chen and Wu (2006) where resource constraints are also considered, with the objective to minimise makespan, maximum tardiness and total tardiness, respectively. In Rabadi et al. (2006) a heuristic for the unrelated machine case with the objective to minimise makespan is also presented. Rocha de Paula et al. (2007) proposed a method based on the VNS strategy for both cases, identical and unrelated parallel machines for the makespan objective. In Low (2005) and Armentano and Felizardo (2007) the authors proposed a simulated annealing method and a GRASP algorithm, with the objective to minimise the total flowtime and the total tardiness, respectively.

Regarding the exact methods, there are some papers available in the literature for the parallel machine problem. However, most of them are able to solve instances with a few number of jobs and machines (more details in Allahverdi et al. (2008)).

In this paper, we deal with the unrelated parallel machine scheduling problem in which machine and job sequence dependent setup times are considered, i.e., the setup times depend on both, the job sequence and the assigned machine. We evaluate and compare some of the above methods available in the literature. We also propose a genetic algorithm that shows excellent performance for a large benchmark of instances.

3. MIP mathematical model

In this section, we provide a Mixed Integer Programming (MIP) mathematical model for the unrelated parallel machine scheduling problem with sequence dependent setup times. Note that this model is an adapted version of that proposed by Guinet (1993). We first need some additional notation in order to simplify the exposition of the model.

- p_{ij}: processing time of job j, $j \in N$ at machine i, $i \in M$.
- S_{ijk}: machine based sequence dependent setup time on machine i, $i \in M$, when processing job k, $k \in N$, after having processed job j, $j \in N$.

The model involves the following decision variables:

$$X_{ijk} = \begin{cases} 1, & \text{if job } j \text{ precedes job } k \text{ on machine } i \\ 0, & \text{otherwise} \end{cases}$$

C_{ij} = Completion time of job j at machine i

$C_{\max}$ = Maximum completion time

The objective function is:

$$\min C_{\max}, \tag{1}$$

And the constraints are:

$$\sum_{i \in M} \sum_{\substack{j \in \{0\} \cup \{N\} \\ j \neq k}} X_{ijk} = 1, \quad \forall k \in N, \tag{2}$$

$$\sum_{i \in M} \sum_{\substack{k \in N \\ j \neq k}} X_{ijk} \leqslant 1, \quad \forall j \in N, \tag{3}$$

$$\sum_{k \in N} X_{i0k} \leqslant 1, \quad \forall i \in M, \tag{4}$$

$$\sum_{\substack{h \in \{0\} \cup \{N\} \\ h \neq k, h \neq j}} X_{ihj} \geqslant X_{ijk}, \quad \forall j, \quad k \in N, \quad j \neq k, \quad \forall i \in M, \tag{5}$$

$$C_{ik} + V(1 - X_{ijk}) \geqslant C_{ij} + S_{ijk} + p_{ik}, \quad \forall j \in \{0\} \cup \{N\}, \quad \forall k \in N, \quad j \neq k, \quad \forall i \in M, \tag{6}$$

$$C_{i0} = 0, \quad \forall i \in M, \tag{7}$$

$$C_{ij} \geqslant 0, \quad \forall j \in N, \quad \forall i \in M, \tag{8}$$

$$C_{\max} \geqslant C_{ij}, \quad \forall j \in N, \quad \forall i \in M, \tag{9}$$

$$X_{ijk} \in \{0, 1\}, \quad \forall j \in \{0\} \cup \{N\}, \quad \forall k \in N, \quad j \neq k, \quad \forall i \in M. \tag{10}$$

The objective is to minimise the maximum completion time or makespan. Constraint set (2) ensures that every job is assigned to exactly one machine and has exactly one predecessor. Notice the usage of dummy jobs 0 as X_{i0k}, $i \in M$, $k \in N$. With constraint set (3) we set the maximum number of successors of every job to one. Set (4) limits the number of successors of the dummy jobs to a maximum of one on each machine. With Set (5) we ensure that jobs are properly linked in machine: if a given job j is processed on a given machine i, a predecessor h must exist on the same machine. Constraint set (6) is to control the completion times of the jobs at the machines. Basically, if a job k is assigned to machine i after job j (i.e., $X_{ijk} = 1$), its completion time C_{ik} must be greater than the completion time of j, C_{ij}, plus the setup time between j and k and the processing time of k. If $X_{ijk} = 0$, then the big constant V renders the constraint redundant. Sets (7) and (8) define completion times as 0 for dummy jobs and non-negative for regular jobs, respectively. Set (9) defines the maximum completion time. Finally, set (10) defines the binary variables. Therefore, in total, the model contains n^2m binary variables, $(n + 1)m + 1$ continuous variables and $2n^2m + nm + 2n + 2m$ constraints. This MIP model will be used later in the computational experiments.

4. Proposed genetic algorithm

Genetic algorithms (GAs) are bio-inspired optimisation methods (Holland, 1975) that are widely used to solve scheduling problems (Goldberg, 1989). Generally, the input of the GA is a set of solutions called population of individuals that will be evaluated. Once the evaluation of individuals is carried out, parents are selected and a crossover mechanism is applied to obtain a new generation of individuals (offspring). Moreover, a mutation scheme is