Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 177 (2007) 135–146
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Discrete Optimization

Scheduling on parallel identical machines to minimize total tardiness

Sang-Oh Shim, Yeong-Dae Kim *

Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, Gusong-dong, Yusong-gu, Daejon 305-701, Republic of Korea

Received 4 March 2005; accepted 23 September 2005
Available online 19 January 2006

Abstract

This paper focuses on the problem of scheduling n independent jobs on m identical parallel machines for the objective of minimizing total tardiness of the jobs. We develop dominance properties and lower bounds, and develop a branch and bound algorithm using these properties and lower bounds as well as upper bounds obtained from a heuristic algorithm. Computational experiments are performed on randomly generated test problems and results show that the algorithm solves problems with moderate sizes in a reasonable amount of computation time.

Keywords: Scheduling; Identical parallel machines; Total tardiness; Branch and bound algorithm

1. Introduction

This paper focuses on the problem of scheduling n independent jobs with different due dates on m identical parallel machines. Generally, there are two decisions to be made in parallel-machine scheduling problems. One is to assign jobs to the machines, and the other is to determine the sequence of the jobs on each machine. In this paper, we develop a branch and bound algorithm for an identical parallel-machine scheduling problem with the objective of minimizing total tardiness of the jobs. Here, the tardiness of a job (i) is defined as $T_i = \max\{0, C_i - d_i\}$, where C_i and d_i are the completion time and due date of job i, respectively. According to the three-field notation of Graham et al. (1979), this problem is referred to as $P_m \| \sum T_i$. It is assumed that all jobs are available at time zero, each machine is continuously available but cannot process more than one job at any time, each job can be processed on only one machine, and no job can be preempted. The problem considered here is at least binary NP-hard (Koulamas, 1994), since special cases of the problem, those with

* Corresponding author. Tel.: +82 42 869 3120; fax: +82 42 869 3110.
E-mail address: ydkim@kaist.ac.kr (Y.-D. Kim).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2005.09.038

$m = 1$ and $m = 2$, are binary NP-hard (Du and Leung, 1990; Lenstra et al., 1997).

Although there have been many studies on parallel-machine scheduling problems with the objective of minimizing makespan or total flow time, not much progress has been made for the problems with the objective of minimizing total tardiness except for certain special cases until recently. Root (1965) and Kovalyov and Werner (2002) propose a solution procedure for minimizing total tardiness in parallel-machine problems in which all jobs have a common due date, and Elmaghraby and Park (1974) deal with problems in which the due date of each job is equal to the processing time of the job. Arkin and Roundy (1991) devise an algorithm for problems in which weights of jobs are proportional to their processing times for the objective of minimizing weighted tardiness. On the other hand, Azizoglu and Kirca (1998) and Yalaoui and Chu (2002) find dominance properties and present branch and bound algorithms using them to obtain optimal solutions in identical parallel-machine scheduling problems with the objective of minimizing total tardiness.

Since it is not easy to obtain optimal solutions for parallel-machine tardiness problems of a practical size, heuristic algorithms have been devised. In most heuristic algorithms, the list scheduling method is employed. In the method, when a machine becomes available for processing a job, one of jobs that can be processed on the machine at the time is selected based on a certain priority rule and scheduled on the machine. Similarly, when a job becomes available for being processed, a machine is selected according to a priority rule among those that can process the job, and then the job is scheduled on the machine. Various dispatching rules have been devised for computation of priorities of the jobs for the problem, or modified from rules for single-machine tardiness problems (Baker, 1973; Dogramaci and Surkis, 1979; Ho and Chang, 1991; Alidaee and Rosa, 1997). On the other hand, Koulamas (1997) suggests a heuristic, called KPM, for the parallel-machine total tardiness problem by extending PSK, the heuristic of Panwalkar et al. (1993) for the single-machine tardiness problem. Also, Lee and Pinedo (1997) and Park (2000) propose a simulated annealing algorithm and a genetic algorithm, respectively, and Armentano and Yamashita (2000); Bilge et al. (2004); Ko et al. (2004) present tabu search algorithms for the problem. Unlike other research, Kim (1987, 1995a) suggests a backward approach, in which jobs are scheduled backward in a reversed time frame.

In this paper, we suggest a branch and bound (BAB) algorithm for the identical parallel-machine scheduling problem with the objective of minimizing total tardiness. We develop dominance properties and lower bounds, which can be used in the BAB algorithm. The remainder of this paper is organized as follows. In the next section, we give dominance properties associated with the problem, and the BAB algorithm and lower bounds used in the algorithm are given in Section 3. The suggested BAB algorithm is tested on randomly generated problem instances and results are shown in Section 4. Finally, Section 5 concludes the paper with a short summary and discussions on possible extensions.

2. Dominance properties

In this section, we present properties of an optimal schedule of the problem considered here. The following notation will be used throughout the paper.

J	set of jobs
M	set of machines
n	number of the jobs ($n = \|J\|$)
m	number of the machines ($m = \|M\|$)
P_i	processing time of job i
d_i	due date of job i
$C_i(\sigma)$	completion time of job i in (partial) schedule σ
$T_i(\sigma)$	tardiness of job i in (partial) schedule σ
$\Gamma(\sigma)$	completion time of the last job on machine k in (partial) schedule σ
$\breve{n}_k(\sigma)$	number of jobs assigned to machine k in (partial) schedule σ
$B_{ij}(\sigma)$	set of jobs scheduled between jobs i and j, which are assigned to the same machine, in (partial) schedule σ
$S(\bullet)$	set of jobs already included in (partial) schedule $\bullet$