Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 182 (2007) 80–94
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Discrete Optimization

A self-adaptive differential evolution heuristic for two-stage assembly scheduling problem to minimize maximum lateness with setup times

Fawaz S. Al-Anzi [a,*], Ali Allahverdi [b,1]

[a] College of Engineering and Petroleum, Department of Computer Engineering, Kuwait University, P.O. Box 5969, Safat, Kuwait
[b] Department of Industrial and Management Systems Engineering, Kuwait University, P.O. Box 5969, Safat, Kuwait

Received 9 December 2005; accepted 7 September 2006
Available online 3 November 2006

Abstract

The two-stage assembly flowshop scheduling problem has been addressed with respect to different criteria in the literature where setup times are ignored. For some applications, setup times are essential to be explicitly considered since they may take considerable amount of time. We address the two-stage assembly flowshop scheduling problem with respect to maximum lateness criterion where setup times are treated as separate from processing times. We formulate the problem and obtain a dominance relation. Moreover, we propose a self-adaptive differential evolution heuristic. To the best of our knowledge, this is the first attempt to use a self-adaptive differential evolution heuristic to a scheduling problem. We conduct extensive computational experiments to compare the performance of the proposed heuristic with those of particle swarm optimization (PSO), tabu search, and EDD heuristics. The computational analysis indicates that PSO performs much better than tabu and EDD. Moreover, the analysis indicates that the proposed self-adaptive differential evolution heuristic performs as good as PSO in terms of the average error while only taking one-third of CPU time of PSO.

Keywords: Assembly flowshop; Scheduling; Maximum lateness; Self-adaptive differential evolution; Setup times; Heuristic

1. Introduction

Consider the two-stage assembly scheduling problem where there is a set of n jobs to be scheduled. Each job requires m parts to be processed at the first stage. There are m first-stage machines where each machine is assigned to process a specific part. These machines are independent of each other. When all m parts of a job are finished at the first stage, a single machine at the second stage assembles the m parts together to finish processing the job. Different applications of this problem have been reported in the literature, e.g., Lee et al. (1993) and Potts et al. (1995). Another application is the production of furniture where the assembly machine

* Corresponding author. Tel.: +965 498 5100; fax: +965 481 7451/484 1603.
E-mail addresses: alanzif@eng.kuniv.edu.kw (F.S. Al-Anzi), allahverdi@kuc01.kuniv.edu.kw (A. Allahverdi).
[1] Fax: +965 481 6137.

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2006.09.011

can be considered to be a single packing machine where all the components of the furniture are to be packed, and therefore, must be available.

Lee et al. (1993) and Potts et al. (1995) addressed the assembly flowshop scheduling problem with respect to makespan minimization and both proved that the problem with this objective function is NP-hard in the strong sense even for the case of having only two machines at the first stage. Lee et al. (1993) discussed a few polynomially solvable cases and presented a branch and bound algorithm. Moreover, they proposed three heuristics and analyzed their error bounds. Potts et al. (1995) showed that the search for an optimal solution may be restricted to permutation schedules. They also showed that any arbitrary permutation schedule has a worst-case ratio bound of two, and they presented a heuristic with a worst-case ratio bound of $2 - 1/m$. Hariri and Potts (1997) also addressed the same problem, developed a lower bound and established several dominance relations. They also presented a branch and bound algorithm incorporating the lower bound and dominance relations. Another branch and bound algorithm was proposed by Haouari and Daouas (1999). Sun et al. (2003) also considered the same problem with the same makespan objective function and proposed heuristics to solve the problem. Koulamas and Kyparisis (2001) generalized the two-stage problem to a three-stage assembly scheduling problem. They proposed several heuristics and analyzed the worst-case ratio bounds of the proposed heuristics for the makespan problem.

Tozkapan et al. (2003) considered the two-stage assembly scheduling problem with the total weighted completion (flowtime) performance measure. They showed that permutation schedules are dominant for the problem with this performance measure. They developed a lower bound and a dominance relation, and utilized the bound and dominance relation in a branch and bound algorithm. They also proposed two heuristics to find an upper bound for their branch and bound algorithm. Al-Anzi and Allahverdi (2006) also considered the problem with respect to total completion time criterion. They proposed an algorithm and showed that the algorithm is optimal under certain conditions. They also proposed several heuristics. Allahverdi and Al-Anzi (2006) addressed the same problem where the objective is to minimize maximum lateness. They formulated the problem and obtained a dominance relation. Moreover, they proposed and evaluated three heuristics for the problem, namely, PSO, tabu search, and EDD heuristics. It was shown that tabu outperforms the PSO and EDD for the case when due dates range is relatively wide. It was also shown that the PSO significantly outperforms the tabu and EDD for tight due dates.

Setup times are assumed to be zero in all of the above mentioned research. While this assumption simplifies the analysis and/or reflects certain applications, it adversely affects the solution quality for many applications which require separate and non-zero treatment of setup times, Allahverdi et al. (1999, 2006). In this paper, we consider the two-stage flowshop scheduling problem with respect to maximum lateness performance measure where setup times are treated as non-zero and separate from processing times, which are sequence independent. Formulation of the problem is presented in the next section. In Section 3, four heuristics are proposed and a dominance relation is developed in Section 4. Comparison of the proposed heuristics is performed in Section 5. Finally, concluding remarks are made in Section 6.

2. Formulation

We assume that n jobs are simultaneously available at time zero and that preemption is not allowed, i.e., any started operation has to be completed without interruptions. Each job consists of a set of $m + 1$ operations. The first m operations are completed at stage one in parallel while the last operation is performed at stage two. Setup times on each machine including the assembly machine are considered as separate from processing times and sequence independent.

Let

$t_{i,j}$	processing time of job i on machine j (at stage one), $i = 1, \ldots, n$, $j = 1, \ldots, m$,
$s_{i,j}$	setup time of job i on machine j (at stage one), $i = 1, \ldots, n$, $j = 1, \ldots, m$,
$t_{[i,j]}$	processing time of the job in position i on machine j (at stage one),
$s_{[i,j]}$	setup time of the job in position i on machine j (at stage one),
p_i	processing time of job i on assembly machine (at stage two),
s_i	setup time of job i on assembly machine (at stage two),