Bicriteria Two-Machine Flowshop Scheduling using Metaheuristics

S. Afshin Mansouri
CREST, Department of Computer Science,
King's College London,
London WC2R 2LS, UK
Afshin.Mansouri@kcl.ac.uk

S. Hamed Hendizadeh
Department of Mechanical and Manufacturing Engineering,
University of Manitoba,
Winnipeg, Manitoba,
Canada R3T 5V6
umhendiz@cc.umanitoba.ca

Nasser Salmasi
Department of Industrial Engineering,
Sharif University of Technology,
Tehran, Iran
nsalmasi@sharif.edu

ABSTRACT

A bicriteria two-machine flowshop scheduling problem is addressed to minimize the number of setups and makespan. The jobs have sequence dependent setup times. Scheduling problems considering either of these objectives are $\mathcal{NP}$-hard. Two multi-objective metaheurisctics based on genetic algorithms (MOGA) and simulated annealing (MOSA) are proposed to find approximations of Pareto-optimal frontiers. The performances of these approaches are compared with lower bounds for small problems. In larger problems, their performance is compared with each other. Experimentations revealed that both algorithms perform very similar on small problems and that MOGA outperforms MOSA in terms of the quality of solutions on larger problems.

Categories and Subject Descriptors

J.2 [**Physical Sciences and Engineering**]: Engineering

General Terms

Algorithms, Experimentation, Performance

Keywords

Multicriteria scheduling, Sequence-dependent setups, Flowshop, Pareto-optimal frontier, Genetic Algorithms, Simulated Annealing

1. INTRODUCTION

In this paper we address a flowshop sequence dependent job scheduling problem. The term "sequence dependent" implies that the setup times depend on the sequence in which the jobs are processed on the machines. It is assumed that n jobs are to be processed on two machines in the same order. Each job J_i is characterized by two attributes. The attribute of job J_i on machine k is denoted by $a_{i,k}$. Let A_1 and A_2 denote the sets of all possible attributes on machines M_1 and M_2, respectively. If job J_j is processed immediately after job J_i, a setup time $s_{ij,k}$ is required on machine k, if $a_{i,k} \neq a_{j,k}$. The setup times as in many real world scheduling problems are sequence dependent. The process time of job J_i on machine k is shown by $p_{i,k}$.

The goal is to schedule the set of jobs in order to minimize the number of setups and the *makespan* (or C_{max}). The first objective is usually favored by the production managers to reduce cost and complexity of the production plan while the second one is mostly considered by the customers as a measure of service. There seems to be a natural conflict between these two objectives so one must consider the set of Pareto-optimal solutions equally favorable if preferences of the decision-maker are not known *a priori*. Using the standard three-field notation of multicriteria scheduling problems, the bicriteria problem addressed in this research can be referred to as $F_2|S_{sd}|Setups, C_{max}$.

2. THE SOLUTION APPROACHES

We employ multi-objective genetic algorithm and simulated annealing metaheuristics (MOGA and MOSA respectively) to find approximations of the Pareto-optimal sets.

3. COMPUTATIONAL EXPERIMENTS

Two groups of test problems were generated at random to evaluate the performance of proposed solution approaches. The first group includes 11 small size problems with 9 to 38 jobs. The second group includes 24 larger problems with 40 to 1000 jobs.

Parameters of both algorithms were tuned empirically. For small size problems, the quality of frontiers are measured with reference to lower bounds. For large size problems, the union of the final frontiers of MOGA and MOSA is used as the reference set. Both MOGA and MOSA algorithms were run 20 times on each problem on the same platform. Average and standard deviation of these runs for small and large problems are calculated. Significance of the difference between the two algorithms were examined using t-Test at $\alpha = 0.05$ level of significance.

4. CONCLUSION

Statistical experimentations show that the quality of both algorithms are equivalent on small size problems. It was also observed that MOGA performs better than MOSA in finding approximations of Pareto-optimal fronts in large size instances at $\alpha = 0.05$ level of significance.

5. ACKNOWLEDGMENT

S. Afshin Mansouri is supported in part by EPSRC grant EP/D050863/1.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
ACM 978-1-59593-697-4/07/0007.