ELSEVIER

European Journal of Operational Research 109 (1998) 137–141

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Theory and Methodology

Benchmarks for shop scheduling problems

Ebru Demirkol, Sanjay Mehta, Reha Uzsoy *

School of Industrial Engineering, Purdue University, West Lafayette, IN 47907, USA

Received 12 February 1996; accepted 17 July 1996

Abstract

In this paper we present extensive sets of randomly generated test problems for the problems of minimizing makespan (C_{max}) and maximum lateness (L_{max}) in flow shops and job shops. The 600 problems include three different types of routings, four different due date configurations and a variety of problem sizes. The problems, as well as the best existing solution and a lower bound on the optimal value are available on the world-wide web. © 1998 Elsevier Science B.V.

Keywords: Combinatorial optimization; Scheduling; Benchmarks

1. Introduction

Job shop and flow shop scheduling problems have been addressed extensively in the literature (Pinedo, 1995). Most studies have focused on minimizing makespan (C_{max}), which is strongly NP-hard for more than two machines in both environments (Garey and Johnson, 1979). Hence a number of research efforts have focused on developing enumerative algorithms to obtain exact solutions (Applegate and Cook, 1991; Balas, 1969; Carlier and Pinson, 1989). While the most commonly used heuristics for these problems in practice have been dispatching rules (Bhaskaran and Pinedo, 1991), a variety of other heuristic approaches such as decomposition (Adams et al., 1988; Balas et al., 1995), tabu search (Taillard, 1994), simulated annealing (Matsuo et al., 1988; Van Laarhoven et al., 1992) and local search (Aarts et al., 1994) have been employed.

A central component of all of these studies is computational experiments to evaluate the performance of the algorithms proposed from the points of view of solution quality and computational requirements. Most of these experiments, however, have been carried out for the $J//C_{max}$ problem and have used relatively small sets of benchmark problems from the literature. These problem sets include a set of 66 problems with C_{max} as performance measure available from Cook (Applegate and Cook, 1991) and 20 $J//C_{max}$ problems due to Storer et al. (1992). Taillard (1993) provides 120 flow shop, 80 job shop and 60 open shop problems with C_{max} as performance measure. To the best of our knowledge there are no benchmark problems currently available for shop scheduling problems with due date based performance measures.

We present an extensive set of randomly generated shop scheduling problems with minimizing makespan (C_{max}) and maximum lateness (L_{max}) as performance measures. The problems are large (up to 1000 operations) and include job shop scheduling

* Corresponding author. Tel.: + 1-765 494 0829.

PII S0377-2217(97)00019-2

problems with two different routing structures as well as flow shop problems. In the first set of job shop problems we consider, which we shall refer to as $J//C_{max}$ or $J//L_{max}$ depending on the performance measure of interest, the routing of each job is a random permutation of all the machines in the shop. In the second set of job shop problems the set of machines is partitioned into two sets. All jobs must visit all machines in the first set in some random order before visiting any machine in the second set. We shall denote these problems by $J/2SETS/C_{max}$ and $J/2SETS/L_{max}$. In this note we provide a description of the mechanism for generating the problems. The problems themselves, together with lower bounds on the optimal solution value and the best solution value we have obtained to date, are provided in Demirkol et al. (1996b) or on the world-wide web site http://www.ecn.purdue.edu/labs/uzsoy. The web site also provides the code and problem parameters used to generate the problems, which will hopefully allow users to correlate problem characteristics with algorithm performance. A total of 600 problems (120 for minimizing C_{max} and 480 for minimizing L_{max}) are provided.

In Section 2, we describe the approach used for generating the test problems. Section 3 briefly describes the procedures used to obtain the solution values given. We conclude the paper with a summary in Section 4.

2. Generating benchmark problems

Table 1 shows the experimental design used to generate the test problems. We consider six basic shop scheduling problems, $F//C_{max}$, $F//L_{max}$, $J//C_{max}$, $J//L_{max}$, $J/2SETS/C_{max}$ and $J/2SETS/L_{max}$. For each type of problem, all jobs are available at time zero and the operation processing times are generated from a discrete uniform distribution between 1 and 200. We consider four values for the number of jobs (n = 20, 30, 40, 50) and two for the number of machines (m = 15, 20). Hence the total number of operations varies from 300 to 1000. The ratio of jobs to machines varies between 1 and 3.3. These combinations yield a problem set which extends the currently available set of benchmark problems without being based on a specific application. For the problems with L_{max} as performance measure, the due dates are determined using two parameters T and R. T determines the expected number of tardy jobs (and hence the tightness of the due dates) and R is the due date range parameter. The job due dates are generated from the distribution

$$\text{Uniform}\left[\mu - \frac{\mu R}{2}, \mu + \mu\frac{R}{2}\right]$$

where $\mu = (1.0 - T)nP$ and P is the expected operation processing time (100.5 for the problems presented in this paper). This expression assumes that all machines will be utilized equally and will have no idle time, which will tend to make it an optimistic estimate of the actual makespan. We consider two possible values of T (0.3 and 0.6) and two possible values of R (0.5 and 2.5). For each parameter combination (number of jobs, number of machines, T and R), we generated 10 problems, yielding a total of 80 problems each for $J//C_{max}$, $J/2SETS/C_{max}$ and $F//C_{max}$ and 320 problems each for $J//L_{max}$, $J/2SETS/L_{max}$ and $F//L_{max}$.

To obtain a more compact and challenging set of test problems, we solved each problem thus generated using the various solution methods discussed in Section 3. The best solution found by any of these methods was taken as the upper bound for each problem. A lower bound for each problem was obtained by relaxing the capacity constraints on all but one machine and solving the resulting single machine problem of minimizing C_{max} with release and delivery times to optimality (Pinedo, 1995). This was performed for each machine and the highest C_{max} value obtained reported as a lower bound for the problem. We then ranked the problems in decreasing order of the percentage gap between upper and lower bounds for each problem parameter combination. In this paper we present only the first five problems for each problem parameter combination. Thus, we provide a total of 120 C_{max} problems and 480 L_{max} problems in the web site.

The C code to generate these problems is provided in the web site. The same random number generator, a prime modulus multiplicative linear congruential generator using parameter values suggested in Law and Kelton (1991), was used in all the

Table 1
Experimental design for problem generation

Problem parameter	Values considered	Total values
Shop configuration	classic job shop; two sets job shop; flow shop	3
Operation processing time	uniform [1, 200]	1
Number of jobs	20, 30, 40, 50	4
Number of machines	15, 20	2
Job due date	$d_j = \text{Uniform}\left[\mu - \frac{\mu R}{2}, \mu + \frac{\mu R}{2}\right]$	
	$\mu = (1.0 - T) *$ number of jobs $* E[P]$	
	T = percentage tardy jobs = 0.3, 0.6	2
	R = due date range = 0.5, 2.5	2

Total problem parameters (C_{max}): 24; C_{max} problems/combination: 10; Total C_{max} problems: 240.

programs and is included in the code. We provide the initial seed, problem parameters, upper bound, lower bound and computation time for each problem in the web site. For all shop configurations, the user must enter the type of shop (flow shop, job shop with random routings or job shop with two-set routings) and an initial seed, as well as the number of jobs and machines when generating the problems. For the L_{max} problems, the user must also enter values for the due date parameters T and R.

For the job shop environment with random routings through all machines, we generate m random numbers for each job, one corresponding to each machine. The machines are then sorted in increasing order of these associated random numbers and the resulting permutation of the machines recorded as the routing for the job. In the job shop environment with two sets of machines, the machines are divided into two equal groups, except for the fifteen machine problems, where the first group consists of seven machines. For each job, we first generate a random number associated with each machine in the first set. We sort the machines in the first set in increasing order of the associated random numbers, obtaining a random permutation of the machines in the first set. We then repeat this procedure for the second set of machines. Thus the routing of a job is a random permutation of the machines in the first set, followed by a random permutation of the machines in the second set. Storer et al. (1992) and Holtsclaw and Uzsoy (1996) found this type of problem to be more difficult that those where the routings are a random permutation of all machines. For the flow shop problems, the routing of each job is taken to be 1, 2, ..., m where m is the number of machines.

3. Solution methods

Since the size of the problems and their complexity render exact solutions impractical, we have used a number of heuristic procedures to obtain the solutions for the problems listed in web site. The most common approach to shop scheduling in practice has been the use of dispatching rules which construct a schedule one operation at a time. We use the first in first out (FIFO), last in first out (LIFO), shortest processing time (SPT) and random (RANDOM) rules for both C_{max} and L_{max} problems. Under the RANDOM rule, each problem is solved ten times using independent random numbers and the best solution over all runs is reported. We consider most work remaining (MWKR) for the C_{max} problems and earliest due date (EDD) and operation earliest due date (OPEDD) for L_{max} problems. In EDD, the operations are prioritized based on the due date of the job to which they belong. OPEDD estimates a due date for the individual operations based on the due date of the job to which they belong and the remaining processing time of the job. It thus gives priority to the operation with largest value of $d_i - K * \text{WKR}_i$ where d_i is the due date of job i and WKR_i is the work remaining (i.e. total processing time of uncompleted operations) for processing on job i, where job i is the job to which the operation belongs. The factor K allows for delays incurred while waiting for processing at the different machines in the routing. We use six different K values (1, 1.5, 2, 2.5, 3 and 4). Bhaskaran and Pinedo (1991) give a detailed discussion of dispatching rules.

In addition to the dispatching rules, we used several versions of the shifting bottleneck (SB) procedure (Adams et al., 1988; Balas et al., 1995) to solve the problems. SB is a heuristic which decomposes the shop problem into single machine subproblems which are prioritized, solved and introduced into the solution one by one until a feasible schedule for the entire shop is obtained. At each iteration, all previously solved subproblems are reoptimized to improve solution quality. We will refer to the criteria used in prioritizing single machine subproblems as bottleneck selection criteria (BSC). We use the SB method proposed by Balas et al. (1995). We consider three different versions of the method with different bottleneck selection criteria described in Demirkol et al. (1996a). For a more detailed discussion of the SB procedure, the reader is referred to Adams et al. (1988), Balas et al. (1995), Pinedo (1995), Ovacik and Uzsoy (1992, 1995), Dauzere-Peres and Lasserre (1993) and Demirkol et al. (1996a).

Thus we solve each $C_{\max}$ problem using five dispatching rules and three SB methods and each $L_{\max}$ problem using eleven dispatching rules and three SB methods. All algorithms were run on a SUN SPARCserver 1000 Model 1104 with four 50 MHz CPUs and 256 MB of RAM, which is a multitasking system running under UNIX. We report only the best solution provided by any of these methods, the computation time taken by the method providing the best solution and the lower bound value. If more than one method finds the best solution, we report the lowest computation time.

4. Summary and conclusions

In this paper we have described a set of 600 test problems for minimizing makespan and maximum lateness in three different shop environments. The problem set is considerably larger than those existing in the literature and contains large problems with different routing structures. To the best of our knowledge this is the first attempt to provide a set of standard benchmark problems for shop scheduling problems with due dates. We hope that these problems, together with those already existing in the literature, will be useful to researchers interested in the computational performance of algorithms for these complex problems.

Acknowledgements

This research was supported by Purdue Research Foundation and by Intel Corporation.

References

Aarts, E.H.L., van Laarhoven, P.J.M., Lenstra, J.K. and Ulder, N.L.J. (1994), "A computational study of local search algorithms for job shop scheduling", *ORSA Journal on Computing* 6, 118–125.

Adams, J., Balas, E. and Zawack, D. (1988), "The shifting bottleneck procedure for job-shop scheduling", *Management Science* 34, 391–401.

Applegate, D. and Cook, W. (1991), "A computational study of the job shop scheduling problem", *ORSA Journal on Computing* 3, 149–156.

Balas, E. (1969), "Machine sequencing via disjunctive graphs: An implicit enumeration approach", *Operations Research* 17, 941–957.

Balas, E., Lenstra, J.K. and Vazacopoulos, A. (1995), "One machine scheduling with delayed precedence constraints", *Management Science* 41, 94–109.

Bhaskaran, K. and Pinedo, M. (1991), "Dispatching", in: G. Salvendy (ed.), *Handbook of Industrial Engineering*, ch. 83, John Wiley, New York.

Carlier, J. and Pinson, E. (1989), "An algorithm for solving the job-shop problem", *Management Science* 35, 164–176.

Dauzere-Peres, S. and Lasserre, J.B. (1993), "A modified shifting bottleneck procedure for job shop scheduling", *International Journal of Production Research* 31, 923–932.

Demirkol, E., Mehta, S.V. and Uzsoy, R. (1996a), "A computational study of shifting bottleneck methods for shop scheduling problems", Research Report, School of Industrial Engineering, Purdue University, West Lafayette, IN 47907.

Demirkol, E., Mehta, S.V. and Uzsoy, R. (1996b), "Benchmarking for shop scheduling problems", Research Report, School of Industrial Engineering, Purdue University, West Lafayette, IN 47907 .

Garey, M.R. and Johnson, D.S. (1979), *Computers and Intractability: A Guide to the Theory of NP Completeness*, W.H. Freeman, San Francisco.

Holtsclaw, H.H. and Uzsoy, R. (1996), "Machine criticality measures and subproblem solution procedures in shifting bottleneck methods: A computational study", *Journal of the Operational Research Society* 47, 666–677.

Law, A.M. and Kelton, W.D. (1991), *Simulation Modeling and Analysis*, McGraw-Hill, New York.

Matsuo, H., Suh, C.J. and Sullivan, R.S. (1988), "A controlled search simulated annealing method for the general jobshop scheduling problem", Working Paper 03-04-88, Department of Management, University of Texas, Austin.

Ovacik, I.M. and Uzsoy, R. (1992), "A shifting bottleneck algorithm for scheduling semiconductor testing operations", *Journal of Electronics Manufacturing* 2, 119–134.

Ovacik, I.M. and Uzsoy, R. (1995), "Decomposition methods for scheduling semiconductor testing facilities", *International Journal of Flexible Manufacturing Systems*, forthcoming.

Pinedo, M. (1995), *Scheduling: Theory, Algorithms and System*, Prentice-Hall.

Storer, R.H., Wu, S.D. and Vaccari, R. (1992), "New search spaces for sequencing problems with application to job shop scheduling", *Management Science* 38, 1495–1509.

Taillard, E.D. (1993), "Benchmarks for basic scheduling problems", *European Journal of Operational Research* 64, 278–285.

Taillard, E.D. (1994), "Parallel taboo search techniques for the job shop scheduling problem", *ORSA Journal on Computing* 6, 108–117.

Van Laarhoven, P.J.M., Aarts, E.H.L. and Lenstra, J.K. (1992), "Job shop scheduling by simulated annealing", *Operations Research* 40, 113–125.