Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds.

SHOP SCHEDULING USING TABU SEARCH AND SIMULATION

Daniel A. Finke
D. J. Medeiros

Department of Industrial & Manufacturing Engineering
310 Leonhard Building
The Pennsylvania State University
University Park, PA 16802, U.S.A.

Mark T. Traband

Applied Research Laboratory
The Pennsylvania State University
P.O. Box 30
State College, PA 16804, U.S.A.

ABSTRACT

An important goal in scheduling products through a manufacturing facility is to assure that the work is completed as close as possible to its due date. Work that is late creates downstream delays, while early completion can be detrimental if storage space is limited. This paper reports initial results in developing a scheduling procedure for an automated steel plate fabrication facility. The approach uses Tabu search combined with simulation to schedule product through a set of machines. Performance of the procedure is evaluated by comparison to the optimal solution for small problem instances, and to a good heuristic for larger problems. Results show that the Tabu search method works well for this problem. Combining Tabu search with simulation allows the incorporation of more realistic constraints on system operation.

1 INTRODUCTION

General Dynamics Electric Boat has recently completed the construction of an advanced steel processing facility. This multi-million dollar, state of the art facility will support the production of cut plate and extruded shapes for ships. The primary goal of the facility is to reduce the costs associated with the production of structural subassemblies for vessels (Williams et al. 2001).

Raw plate and extruded shapes enter at one end of the building and pass through several automated machines which perform blasting, marking, and cutting operations. The components produced may go through several secondary operations such as forming and bending, then are assembled into kits which exit the building and move to the assembly area.

An important goal in scheduling the facility is to ensure that kits are completed just-in-time. Late kits can delay the assembly operations. Items that are completed early occupy limited storage space.

Simulation models of the facility and a scheduling approach were developed in parallel with facility construction and equipment installation and activation. Production data was not available, so these activities utilized sample data sets. This paper describes our approach to scheduling work so that it finishes close to its due date, also known as the earliness-tardiness (ET) problem (Li et al. 2000).

Our development assumes that the jobs have individual due dates and common penalties for earliness and tardiness. The processing time and machine assignment for every job is deterministic and known in advance. The machine setup time for a job is included in the processing time. All jobs are available for immediate processing and no preemption is allowed. These assumptions are representative of anticipated operation of the facility, with the exception of penalties which are at present unknown.

The machines in this system are arranged in groups or tiers (Figure 1). Each job is processed by one machine in a tier before it can be processed in the subsequent tier. This type of environment is described as the flow shop with parallel machines (FSPM) (Nowicki and Smutnicki, 1998). In our case, however, machine assignments at each tier are known. This is because the machines in a tier are not identical, although they perform similar operations.

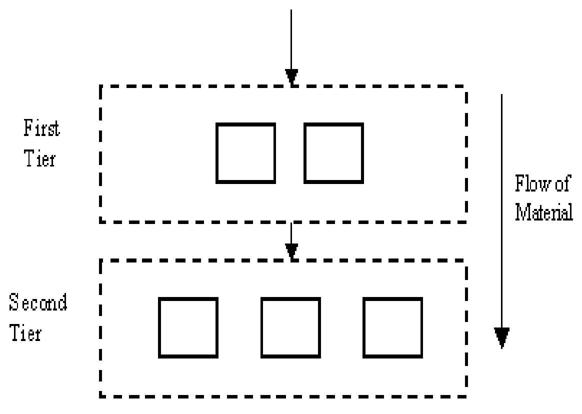

Figure 1: System Structure

2 METHODOLOGY

ET scheduling problems are notoriously difficult to solve because of their combinatorial nature. Heady and Zhu (1998) state that the multiple machine, multiple job ET problem is NP-hard. The multiple machine case can be separated even further into two categories: serial and parallel machines. The multiple, identical, parallel machine case has been studied in great detail, and many solution procedures have been presented for individual cases of this problem. Tamaki, Nishino, and Abe (1999) present a solution method that uses a genetic algorithm in the framework of multiple objective optimization. Tamaki, Komori, and Abe (1999) use a list scheduling technique to find the start times on each machine. Pandelis and Teneketzis (1993) present a model to solve the multiple machine case using queuing theory. This model has merit in the dynamic case where the set of jobs to be processed is not known ahead of time.

Little is known about the flexible flow line cases of the multiple machine ET problem. Lashine, Foote, and Ravindran (1991) propose a nonlinear mixed integer goal-programming model for the two-machine closed flow shop case. Their model includes inventory levels, costs, and planning in addition to scheduling.

A variety of heuristic procedures could be applied to this problem including Simulated Annealing, Genetic Algorithms, and Tabu Search. Tabu search (TS) has been proven to work well on problems in the ET and MILP classes (Al-Turki et al. 2001, Glover, 1990, James, 1997, James and Buchanan, 1996, 1998, and Løkketangen and Glover, 1998). This fact motivates our use of TS to generate solutions, with simulation used as the evaluation mechanism.

The initial configuration (schedule) is very important to the efficiency of TS. EDD and V-shaped sequences have been shown to be better than random sequences. For the single machine problem, V-shaped sequences have been shown by James (1997) and James and Buchanan (1998) to be the best initial configurations. We tested the performance of these configurations for the multiple machine problem by solving several problems using EDD and V-shaped initial configurations. In all cases, EDD outperformed the V-shaped sequence. The EDD sequence is thus used as the initial configuration, and the earliness/tardiness penalties are computed by simulating the flow of jobs through the facility using the EDD sequence to release new orders to the shop.

After the initial configuration has been input and evaluated a neighborhood of other configurations is generated The neighborhood generation method used in this work is similar to the method proposed by James and Buchanan (1998), in which a predetermined number of sequence positions are generated randomly and the job in each sequence position is interchanged with adjacent jobs within a window. Both the number of generating positions and the window size are parameters of the search. We suspected that appropriate values of these parameters might depend on the tier structure (tall and narrow versus short and wide). We conducted some preliminary experimentation on small problems (10 jobs, 5 machines, and a variety of tier structures) and determined that the number of positions, window size, and tier structure did not significantly affect the performance of the TS procedure, as measured by the iterations required to find the optimal solution.

For each sequence generated, the earliness/tardiness penalty must be computed. For this class of problems, inserting idle time between jobs can result in a lower penalty (by decreasing earliness without making the job tardy). Yano and Kim (1986) showed that, for a known sequence, optimally inserting idle time can be accomplished using linear programming. However, the optimal schedule produced in this manner may not achieve good results when translated to a real world application. In particular, the assumption of unlimited buffer size between machines can result in infeasible schedules. Also, the LP formulation assumes that the initial sequence will be maintained as jobs flow through the system (a permutation schedule) and it may be desirable to rearrange jobs in process. Therefore, we used a simulation model of the system to determine the objective function value for each configuration.

After the neighborhood is generated and all configurations in the neighborhood have been evaluated, TS selects the best of those configurations based on the evaluation value. If the best configuration is on the tabu list, then the next best configuration is chosen, unless a randomly generated value exceeds an aspiration criterion. The list size and aspiration criterion, were set to 10 and 0.95, respectively, for this problem.

The search procedure continues until an overall iteration limit is reached or the TS has made several iterations without finding an improved configuration. We used an overall limit of 100 iterations. Following the suggestion of Kim and Yano (1994), we set the parameter for non-improving iterations to the minimum of the number of jobs and 15.

The simulation model was written using the Quest© software by Delmia. The model represents a tiered shop with limited buffers between machines. It reads a data file containing information on the jobs to be processed (machine assignments and processing times). It then reads a sequence to be evaluated which is output by the Tabu search procedure. The jobs begin processing on the first tier in the desired order. This is accomplished by using a FIFO buffer of capacity 1 in front of the first tier. After completion on their first machine, the jobs flow through the system in first in first out order. No inserted idle time is used, that is, if a job is in a buffer and its next machine is idle the job will immediately begin processing.

3 EXPERIMENTATION AND RESULTS

The purpose of these experiments is three fold. First, an experiment is conducted to determine how Tabu search compares to small optimally solved problems. Then its efficiency is tested on large problems where no optimal solution can be found. Both of these experiments use LP as the evaluation mechanism. Finally, the effect of translating tabu search solutions to a simulated system and using simulation as the evaluation method is examined. Because the target facility for the scheduling procedure is in the startup and test phase of operation, we were unable to obtain actual production data for this experimentation, so our data sets are randomly generated.

3.1 Tabu Search versus Optimization

To test the performance of Tabu Search relative to the optimal solution, a set of 15 small problems (10 jobs, 5 machines, and various tier structures) were generated. Two due date alternatives, tight and loose, were applied to each of the problems, resulting in 30 problem instances.

Each of the 30 problems was solved optimally using the MILP formulation in LINDO©. The optimal solutions were compared to the solutions found by the tabu search procedure using the linear program as the evaluation mechanism. The number of generating positions and window size were set at 2 and 2, respectively. In all cases but one, the tabu search found the optimal solution. The one problem that did not find the optimal solution reached the total iteration limit and the search terminated.

The results of this experiment show that the tabu search procedure with the parameters we chose performs well on small problems. In several cases, the EDD sequence was the optimal solution, suggesting that EDD is a good choice for generating the initial configuration.

3.2 Tabu Search for Larger Problems

The next experiment conducted was to compare TS to the EDD solution on larger problems where an optimal solution cannot be found. This experiment was conducted on five randomly generated 20-job problems. The number of tiers varied from two to five, and the number of machines per tier ranged from one to five, with an overall limit of 10 machines. Two due date conditions, loose and tight, were applied to each problem, resulting in 10 instances. The number of generating positions and the window size were increased to 3 and 3, respectively, because James (1997) found that these parameters should vary based on the number of jobs.

Each of the ten instances was solved using the tabu search procedure with the linear program as the evaluation mechanism. Results are shown in Table 1. The EDD solutions are improved on average by 18.5% using the TS procedure for large problems where an optimal solution cannot be found.

Table 1: EDD and TS Results

Problem Number	EDD Solution	TS Solution	% Iprovement
1	700	547	21.86
2	616	492	20.13
3	66	65	1.52
4	120	102	15.00
5	225	173	23.11
6	706	610	13.60
7	530	376	29.06
8	278	210	24.46
9	468	323	30.98
10	136	129	5.15

3.3 System with Limited Buffers

The mathematical formulation assumes that there is a buffer with infinite capacity between any two tiers. If the system has limited buffers, solutions found using this approach may not be realistic.

If a system has infinite buffers between each tier the jobs are allowed to finish processing in the previous tier and wait in the buffer. This allows for delaying jobs that are early, and also for selecting any one of the jobs for processing in the next tier. If the buffer space is limited, and constrains the system, only the jobs waiting in the buffer would be allowed to move to the next tier for processing, and it may not be desirable to hold jobs in the buffer if their next machine is free. Therefore, the simulation model was designed to release jobs in the desired sequence, but to process them in FIFO order through the subsequent tiers.

For the ten problems of the previous section, the best solution found with Tabu search and LP was simulated to determine a more realistic value of earliness/tardiness. Then Tabu search using simulation as the evaluation method was applied to the problems. The results are shown in Table 2. The first five problems in the table have loose due dates while the second set have tight due dates.

For all 10 problems, the simulation of the TS-LP solution results in higher penalties than predicted by the TS-LP. In the case of loose due dates (problems 1-5) the penalties are considerably higher, because the simulation does not insert idle time and thus many jobs are finishing before their due date. Results are much closer for tight due dates.

The earliness/tardiness penalty values for the TS-LP solutions are worse, in all cases, than those found by Tabu search using simulation. In some cases Tabu search with simulation produced much better results, as in problems 8 and 10, where the percent improvement is 55% and 23% respectively, but in other cases the improvements were

Table 2: Simulation Results

Prob-lem	Solution Method	Solution
1	Simulation of TS-LP solution	12113
	TS-Simulation	11605
2	Simulation of TS-LP solution	18928
	TS-Simulation	18608
3	Simulation of TS-LP solution	8462
	TS-Simulation	8147
4	Simulation of TS-LP solution	15377
	TS-Simulation	15055
5	Simulation of TS-LP solution	15550
	TS-Simulation	14718
6	Simulation of TS-LP solution	625
	TS-Simulation	613
7	Simulation of TS-LP solution	654
	TS-Simulation	609
8	Simulation of TS-LP solution	629
	TS-Simulation	283
9	Simulation of TS-LP solution	463
	TS-Simulation	388
10	Simulation of TS-LP solution	328
	TS-Simulation	254

minimal. The average percent improvement for the ten problems is 12%.

4 CONCLUSIONS AND FUTURE WORK

Tabu search produces good results in solving earliness-tardiness problems for multiple machine systems in a serial-parallel configuration. It produces improved schedules as compared to simple dispatching rules (such as Earliest Due Date), and in our testing with small problems it often found the optimal solution. Tabu search combined with simulation allows consideration of more realistic constraints such as limited buffer capacity. In addition, an initial shop loading can easily be incorporated into the simulation to provide a more accurate estimation of on-time performance.

Future work includes investigating some alternative neighborhood generation procedures and studying the effect of inserted idle time at the front of the schedule (before job release).

REFERENCES

Al-Turki, U., C. Fedjki, and A. Andijani. 2001. Tabu Search for a Class of Single-Machine Scheduling Problems. *Computers & Operations Research* 28: 1223-1230.

Glover, F. 1990. Tabu Search: A Tutorial. *Interfaces* 20: 74-94.

Heady, R. B., and Z. Zhu. 1998. Minimizing the Sum of Job Earliness and Tardiness in a Multimachine System. *International Journal of Production Research* 36 (6): 1619-1632.

James, R. J. W. 1997. Using Tabu Search to Solve the Common Due Date Early/Tardy Machine Scheduling Problem. *Computers and Operations Research* 24 (3): 199-208.

James, R. J. W., and J. T. Buchanan. 1996. A Neighbourhood Scheme with a Compressed Solution Space for the Early/Tardy Scheduling Problem. University of Canterbury Research Report, Department of Management.

James, R. J. W., and J. T. Buchanan. 1998. Performance Enhancements to Tabu Search for the Early/Tardy Scheduling Problem. *European Journal of Operational Research* 106: 254-265.

Kim, Y-D., and C. A. Yano. 1994. Minimizing Mean Tardiness and Earliness in Single-Machine Scheduling Problems with Unequal Due Dates. *Naval Research Logistics* 41: 913-933.

Lashine, S., B. Foote, and A. Ravindran. 1991. A Nonlinear Mixed Integer Goal Programming Model for the Two-Machine Closed Flow Shop. *European Journal of Operational Research* 55: 57-70.

Li, Y., K. F. Man, K. S. Tang, S. Kwong, and W. H. Ip. 2000. Genetic Algorithm to Production Planning and Scheduling Problems for Manufacturing Systems. *Production Planning and Control* 11 (5): 443-458.

Løkketangen, A., and F. Glover. 1998. Solving Zero-One Mixed Integer Programming Problems Using Tabu Search. *European Journal of Operational Research* 106: 624-658.

Nowicki, E., and C. Smutnicki. 1998. The Flow Shop with Parallel Machines: A Tabu Search Approach. *European Journal of Operational Research* 106: 226-253.

Pandelis, D. G., and D. Teneketzis. 1993. Optimal Stochastic Sequencing with Earliness and/or Tardiness Costs. In *IEEE Proceedings of the 33rd Conference on Decision and Control, Dec.*, 3618-3623.

Tamaki, H., T. Komori, and S. Abe. 1999. A Heuristic Approach to Parallel Machine Scheduling with Earliness and Tardiness Penalties. *IEEE* 5: 1367-1370.

Tamaki, H., E. Nishino, and S. Abe. 1999. A Genetic Algorithm Approach to Multi-Objective Scheduling Problems with Earliness and Tardiness Penalties. *IEEE* 9: 46-52.

Williams, D. L., D. A. Finke, D. J. Medeiros, and M. T. Traband. 2001. Discrete simulation development for a proposed shipyard steel processing facility. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 882-887. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Yano, C. A., and Y-D. Kim. 1986. Algorithms for Single Machine Scheduling Problems and Minimizing Tardiness and Earliness (Tech. Rep. NO. 86-40) . Ann Arbor, Michigan: The University of Michigan, Department of Industrial & Operations Engineering.

AUTHOR BIOGRAPHIES

DANIEL A. FINKE is a graduate student at The Pennsylvania State University. He received an M.S. in Industrial Engineering and Operations Research from Penn State in 2002. He received his B.S. in Industrial Engineering from New Mexico State University in 2000. His current interests include simulation based optimization and decision improvement. His email address is `<daf903@psu.edu>`.

D. J. MEDEIROS is an Associate Professor in the Department of Industrial and Manufacturing Engineering at Penn State University. She holds a B.S.I.E from the University of Massachusetts and M.S.I.E and Ph.D. from Purdue University. She has served as Track Coordinator, Proceedings Editor, and Program Chair for WSC. She is a member of IIE. Her research interests include manufacturing systems control and CAD/CAM. Her email address is `<djm3@psu.edu>`.

MARK T. TRABAND, Ph.D. has been employed as a Research Associate at the Applied Research Laboratory, The Pennsylvania State University(ARL PennState) since 1990. He is currently the head of the Manufacturing Sys-tems Division. Dr. Traband received a B.S. degree in In-dustrial Engineering from Virginia Polytechnic Institute and State University in 1985. He was selected as an Office of Naval Research Graduate Fellow in 1985. He received his M.S. and Ph.D degrees in Industrial Engineering from The Pennsylvania State University in 1987 and 1995. His email address is `<mtt1@psu.edu>` .