

## DUE DATE BASED SCHEDULING IN A GENERAL FLEXIBLE MANUFACTURING SYSTEM

N. Raman\*  
F. B. Talbot\*\*  
R. V. Rachamadugu\*\*

### EXECUTIVE SUMMARY

Dynamic scheduling of manufacturing systems for due date based objectives has received considerable attention from practitioners and researchers due to the importance of meeting due dates in most industries. Research investigations have focused primarily on the relative effectiveness of various dispatching rules in job shops. These rules operate by prioritizing jobs using a "criticality index" based on job and system status. Jobs are then scheduled from most critical to least, with the indexes typically being updated as the system changes.

This study considers two important issues which have not been addressed previously in the literature. First, we investigate the impact of unequal machine workloads on the relative effectiveness of dispatching rules. This is significant because workloads are likely to be unbalanced in most real systems. While it is clear, intuitively, that this imbalance in machine workloads is likely to deteriorate system performance, it is not obvious whether the superiority of certain dispatching rules established in earlier studies for balanced workloads is carried forward to this case. We show that the performance of different dispatching rules does indeed depend upon the degree of workload imbalance. We also propose and test a scheduling procedure which performs well in both balanced and unbalanced systems.

Next, we develop a scheduling approach which shows promise as being an improved alternative to the use of dispatching rules. This approach decomposes the dynamic scheduling problem into a series of static problems.

---

\*University of Illinois at Urbana—Champaign, Champaign, Illinois

\*\*University of Michigan, Ann Arbor, Michigan

These static problems are then solved using an optimum-seeking method, and the solutions are implemented on a rolling basis. We show through a simulation experiment that adopting this approach over dispatching rules leads to an improvement in the overall solution quality, even in a dynamic environment.

The two very practical implications of our study are: (1) that commonly used dispatching rules in job shops or automated manufacturing systems may not be the best approach when capacity utilization is unbalanced; (2) a job shop or automated manufacturing system would likely benefit from implementing optimal-seeking scheduling rules instead of the traditional job dispatching rules.

## INTRODUCTION

A flexible manufacturing system (FMS) consists of computer numerically-controlled (CNC) machines which are interconnected by an automated material handling system. The overall operation of an FMS is controlled by one or more computers. FMSs can be broadly classified into dedicated and general systems. A dedicated system is designed for repetitive production of few part types in relatively large volumes. A general FMS (GFMS), on the other hand, manufactures a large variety of part types in smaller volumes. These part types usually follow different machine visitation routes through the system, and are produced to specific customer orders. (See Groover (1980) for a detailed discussion on this subject.)

Merchant (1982) observes that GFMSs form a major and growing part of all FMSs because of the increasing need for greater manufacturing flexibility. This study addresses the real time scheduling of a GFMS for due date based scheduling measures. In Panwalker, Dudek and Smith's (1973) study of the industrial scheduling problem, due date based measures were identified as the primary performance criteria used by most practicing managers. A more recent survey by Smith, Ramesh, Dudek and Blair (1986) which specifically addressed FMSs further supports this contention.

We consider a dynamic problem with random job arrival. The primary performance criterion is mean job tardiness. In addition, we also evaluate the generated sequences for the secondary criteria of mean job flow time, proportion of tardy jobs and standard deviation of tardiness.

This problem has received considerable attention in the past in the context of job shops. Prior research (see, for example, Carroll (1965), Baker and Bertrand (1982), Kanet and Hayya (1982), Baker and Kanet (1983), and Baker (1984)) focused largely on the relative effectiveness of dispatching rules under a variety of operating

conditions. One of the major conclusions of these studies is that priority rules based on operation due dates are likely to perform better than those which are based on job due dates.

Operation due dates (ODDs) can be generated from job due dates in two ways. The static approach assigns ODDs at the time of job arrival, and these ODDs remain unchanged till the job is completed. Kanet and Hayya (1982) and Baker (1984) showed the superiority of the Total Work Content (TWK) rule for decomposing job due dates into its operation counterparts under this approach. The second approach sets ODDs dynamically by netting the sum of processing, travel and anticipated waiting times for the remaining operations from the job due date. Since the waiting times are dependent upon the current state of the system (for example, the number of jobs in the system), these ODDs are updated whenever there is a state change. In Vepsalainen and Morton's (1987) study, the waiting times are estimated iteratively.

This study considers two issues which have not been addressed previously. First, we investigate the impact of unequal machine workloads on the relative effectiveness of dispatching rules. Relative machine utilizations are unlikely to be equal in most real systems. While it is clear intuitively that unequal machine workloads are likely to deteriorate system performance, it is not obvious whether the superiority of certain dispatching rules established in earlier studies for balanced workloads is carried forward to this case as well. Unequal machine workloads create bottlenecks in the system, and an average job spends the bulk of its waiting time at the bottleneck machine(s). It may, therefore, be appropriate to adopt different scheduling rules at machines with different workloads.

Second, we propose a solution approach which can be considered as an alternative to the use of dispatching rules. This approach treats the dynamic problem as a series of static problems. A static problem is generated whenever there is a change in system status (e.g., a job arrival). It considers only those jobs which are available in the system at that point in time. The static problem is solved using an optimum-seeking solution method which is implemented on a rolling basis. Dispatching rules also can be used to solve such static problems; however, they use only partial information which is usually local to the machine at which the scheduling decision is to be made.

In general, the implementation of optimum solutions to static problems on a rolling basis need not be effective in a dynamic environment. For example, Chand's (1982) study of the dynamic lot sizing problem indicated that the performance of the Wagner-Whitin algorithm is not very effective when it is applied on a rolling basis. However, in Rachamadugu, Talbot and Raman's (1986) investigation of the single machine tardiness problem, this approach yielded impressive results.



The GFMS studied in this paper is the Automated Manufacturing Research Facility at the National Bureau of Standards in Gaithersburg, Maryland. The reader is referred to Jones and McLean (1986) and Simpson, Hocken and Albus (1982) for detailed descriptions of this system.

This paper is organized as follows. The next section deals with the various operation due date assignment procedures and scheduling rules investigated. In the subsequent section, we discuss the decomposition of the dynamic problem into static problems. This section presents an integer programming formulation of the static problem, and outlines a solution procedure based on a resource constrained project scheduling approach. Details of the simulation study and the experimental results follow in the next section. We conclude with a summary of these results. The notation used in this paper is given in Table 1.

## INVESTIGATION OF DISPATCHING RULES AND OPERATION DUE DATE ASSIGNMENT PROCEDURES

### Scheduling Rules

This paper investigates three dispatching rules. The first two, Modified Job Due Date (MDD) rule and Modified Operation Due Date (MOD) rule, were selected because of their reported effectiveness in previous studies. The MDD rule gives the highest priority to the operation with the lowest modified job due date MDD, where MDD of operation  $i$  in job  $j$  is given by

$$MDD_{ji} = \max (t + P_{ji}, d_j)$$

On the other hand, MOD assigns the highest priority to the operation with the lowest modified operation due date MOD, where MOD of operation  $i$  in job  $j$  is given by

$$MOD_{ji} = \max (t + p_{ji}, d_{ji})$$

The robustness of these rules lies in the fact that they combine the Shortest Processing Time (SPT) rule, which is effective when due dates are tight, with the Earliest Due Date (EDD) rule, which performs well for loose due dates. However, neither MDD nor MOD considers differences in machine workloads. The third rule investigated in this paper, HYBRID, does overcome this shortcoming. Under this rule, MOD is used for dispatching jobs at non-bottleneck machines while MDD is employed at the bottleneck machines. For the purpose of the experimental study, any machine with more than average workload was considered to be a bottleneck.

## Operation Due Date Assignment Procedures

While the total work content (TWK) rule has been found effective in previous investigations for setting static ODDs, these studies have addressed the case in which the machine workloads are balanced. TWK does not consider the relative workload on the machine at which a given operation is processed while setting its ODD. Two approaches are adopted in this study to recognize differences in machine workloads and to incorporate them into the assigned ODDs.

The first approach assigns operational flow allowance factors which are proportional to the workload (PWL) of the machine at which the operation is to be performed. Relative to TWK, this approach provides larger operational flow allowance factors for setting ODDs at the bottleneck machines. Correspondingly, the ODDs at the other machines are relatively tighter. Using this ODD assignment procedure in conjunction with the MOD dispatching rule makes another significant difference in the schedules developed for individual machines. When the ODDs for the non-bottleneck machines are tightened, these machines tend to favor jobs with shorter processing times, especially when these operations precede those which are done on bottleneck machines. This implies that bottleneck machines have a richer collection of jobs to choose from.

The second approach is antithetical to PWL in the sense that it generates operational flow allowance factors which are inversely proportional to the workload (IPWL) of the machine at which the operation is to be performed. Relative to TWK, this approach generates tighter operation due dates for those operations which are to be done on bottleneck machines. The argument behind this approach is that, when it is used with MOD, the bottleneck machines tend to process jobs with shorter processing times relatively early, especially when these operations precede those which are processed on non-bottleneck machines. This would tend to reduce queue buildup at bottleneck machines, thereby affecting flow time and tardiness values favorably. Details of PWL and IPWL procedures are given in Raman, Talbot and Rachamadugu (1988).

Table 2 illustrates the difference between the operation flow allowances assigned by TWK, PWL and IPWL for a typical part type investigated. For this part type, operation 1 is processed on a non-bottleneck machine, operation 3 requires a bottleneck machine, and operation 2 is performed on a machine with average utilization. Relative to TWK, PWL can be seen to provide smaller flow allowance for operation 1, and larger flow allowance for operation 3. The opposite is true for IPWL.

**TABLE 2**  
**COMPARISON OF ODD ASSIGNMENT PROCEDURES**

Operation Number	Processing Time	Assigned Operation Flow Allowance		
		TWK	PWL	IPWL
1	105	315	150	354
2	216	648	548	645
3	79	237	502	201

## STATIC TARDINESS PROBLEM

### Problem Formulation

Within a dynamic framework, a static problem is generated whenever a new job arrives at the system. At that point in time, we consider all jobs in the system and develop schedules for all machines and the material transporter simultaneously. We assume that the operation sequence for each part type establishes a serial precedence relationship among the operations. In addition, the machines required for each operation, operation processing times and travel times are deterministic and known. Preemption of any operation on any machine or the material transporter is prohibited.

An integer programming formulation of the static tardiness problem is given below. We assume without any loss in generality that the operations of each job are numbered such that the successor operation has a number higher than its predecessor.

$$\text{Minimize } \sum_j T_j \tag{1}$$

subject to

$$\sum_t x_{tjk} = 1; \quad j = 1, \dots, N, \quad k = 1, \dots, N_j \tag{2}$$

$$\sum_t (t - p_{j1}) x_{tj1} \geq \sum_t x_{tjk}; \quad j = 1, \dots, N, \quad k = 1, \dots, N_j, \quad \text{and } (k, 1) \in S_j \tag{3}$$

$$\sum_j \sum_k \sum_{q=t}^{t+p_{jk}-1} R_{jkm} x_{qjk} \leq 1; \quad t = 1, \dots, T, \quad m = 1, \dots, M \tag{4}$$

$$\sum_t x_{tjk} + E_j - T_j = d_j; k = N_j, j= 1, \dots, N \quad (5)$$

$$x_{tjk} \in \{0,1\}; E_j, T_j \geq 0, \text{ integer; for all } j, k, t \quad (6)$$

Equation (1) corresponds to the objective of minimizing total tardiness. Constraints (2) ensure that each operation is completed exactly once. Constraints (3) indicate the precedence relationships among the various operations of a job and ensure that the operation processing times are taken into consideration appropriately. Constraints (4) ensure that each resource (machine and transporter) is assigned to at most one operation at any given time. Constraints (5) measure the tardiness of each job. Finally, constraints (6) specify the integer nature of the variables.

In the above formulation, transportation is treated as a move operation between two machining operations, or the load/unload station and a machining operation. This is a reasonable approximation of the real system for the following reasons. First, in the system modeled, the transporter always returns to the load/unload station after moving parts between machines. Second, there are small, but adequate, input and output buffers at each machine. Third, the time to return the transporter to the load/unload station is small relative to the machining times.

If the transporter did not return to the load/unload station, then the formulation would have to be modified to account for the potentially large number of possible alternative routings. If there were no buffers, or if the buffers were serious bottlenecks, then these conditions would have to be modeled explicitly, otherwise the schedule resulting from (1) through (6) could be infeasible. Also, the above formulation and the proposed solution approach assume that any machining operation does not begin until the transporter returns to the load/unload station. Because of condition three given above, this is a reasonable approximation of reality. As a consequence, travel time can be treated as the sum of the transporter round trip time and the transfer time from one machine to another.

It should be emphasized that (1) through (6) is an approximation of the actual system. It captures the real system characteristics only to the extent that they have an effect on the scheduling process. If necessary, this model can be generalized to include cases of no buffers, multiple transporters, alternative machine assignments and "non-renewable" resources such as tool life (see Talbot (1982)). The above considerations, although not trivial, could be modeled in a straightforward manner by modifying the presented formulation and solution procedure. Similarly, permitting random routing of the material transporter (that is, not requiring it to return to the load/unload



station each time) could be formulated. However, this would have the most dramatic impact on computational complexity, and would require significant modification of the suggested solution procedure.

The formulation given by equations (1) through (6) results in a large number of variables and constraints for problems of practical size, thereby precluding the use of general-purpose integer programming codes as solution methods. This problem can, however, be viewed as a resource constrained project scheduling problem (or its subset, the resource constrained job shop scheduling problem) for which reasonably efficient optimum-seeking codes exist for some objective functions. For example, the procedure developed by Talbot (1982) can be used directly to solve the objective of minimizing makespan, and it has been modified by us to solve the tardiness problem. Details of the solution procedure are given in Raman, Talbot and Rachamadugu (1988).

### **Solution Implementation in a Dynamic Environment**

In a dynamic environment, a static problem is generated whenever a new job enters the system. At such points in time, Equations (1) through (6) are developed afresh taking into account the amount of processing remaining in each job. Similarly, job due dates used in the formulation reflect the time remaining until the job is due. Note that at the time of a job arrival, one or more machines or the material transporter can be busy. Since preemption is not permitted, such resources are blocked out for the period of commitment. The optimal (or the best) solution determined by the solution procedure provides a schedule of operations for each machine and the transporter. This schedule is implemented until the next job arrives. At that time, the process of generating and solving the static problem is repeated. The static solution is, therefore, implemented on a rolling basis.

## **EXPERIMENTAL STUDY**

### **The Simulation Model**

A brief description of the simulation model is given below. For further details, the reader is referred to Raman et al. (1988). Job arrival followed a Poisson process. An incoming job was equally likely to belong to any of the twenty part types. Upon its arrival, a job was assigned a due date based on the Total Work Content (TWK) rule. According to this rule, the due date  $d_j$  of job  $j$  is given by,

$$d_j = a_j + FP_{j1}$$

where  $a_j$  is the arrival time of job  $j$ , and  $F$  is the flow allowance factor. As seen from the above equation, due date tightness can be controlled by varying the flow allowance factor.

Two different experiments were conducted. The first experiment addressed the impact of workload-dependent ODD assignment procedures and scheduling rules, while the second experiment investigated the effectiveness of implementing the solution to the static problem on a rolling basis.

The first experiment was designed to yield an overall system utilization of 80%. Three levels of due date tightness were achieved by using job flow allowance factors of 3, 4, and 5. In addition, another set of simulation runs was conducted in which the job flow allowance factor was allowed to vary uniformly between 1 and 7 (to yield a mean of 4). This set was used primarily to assess the robustness of ODD assignment rules with respect to variability in the flow allowance factor.

The first stage of this experiment dealt with the impact of the three ODD assignment procedures discussed earlier. For all three procedures, MOD was used as the dispatching rule. The second stage addressed the relative effectiveness of MOD, MDD and HYBRID under both balanced and unbalanced workloads. Correspondingly, two sets of simulation runs were conducted. In the first set, the machine workloads were nearly equal (the realized utilizations varied between 78% and 82%). The second set yielded workload imbalance with the actual utilizations ranging from 66% to 93%.

In the second experiment, we compared MOD, MDD and HYBRID with the scheduling approach which implemented the solution obtained by the optimum-seeking method to the static problem on a rolling basis (hereafter called the BB rule). The system utilization was designed to be 20%. The utilization was kept deliberately small in order to reduce the size of the static problem generated and thereby enhance the possibility of obtaining optimal solutions within the time trap of 0.5 CPU seconds. (This decision was motivated solely by the desire to keep computational costs within reason.) In the experiment conducted, the size of the static problem, expressed in terms of the number of operations, varied between 9 and 107. The size of the largest solved to known optimality was 36. The flow allowance factors used in the second experiment yielded due date tightness levels, measured in terms of the proportion of tardy jobs, similar to those obtained in the first experiment. As in the first experiment, another set of runs was conducted in which the flow allowance factor was sampled from a uniform distribution.

The scheduling rules were coded in FORTRAN and were interfaced with the simulation model written in SIMAN. The experiments were run on the AMDAHL V8/480 computer at the University of Michigan.

## Experimental Results

The experimental results are given in Figures 1 through 4, and Tables 3 through 7. Table 3 shows the mean tardiness values for the case of unbalanced workloads under the alternative ODD assignment procedures. It can be seen that while TWK dominates PWL at all flow allowances, the difference in the performance of these three procedures is only marginal.

**TABLE 3**  
**IMPACT OF VARIATION IN ODD ASSIGNMENT ON MEAN TARDINESS**

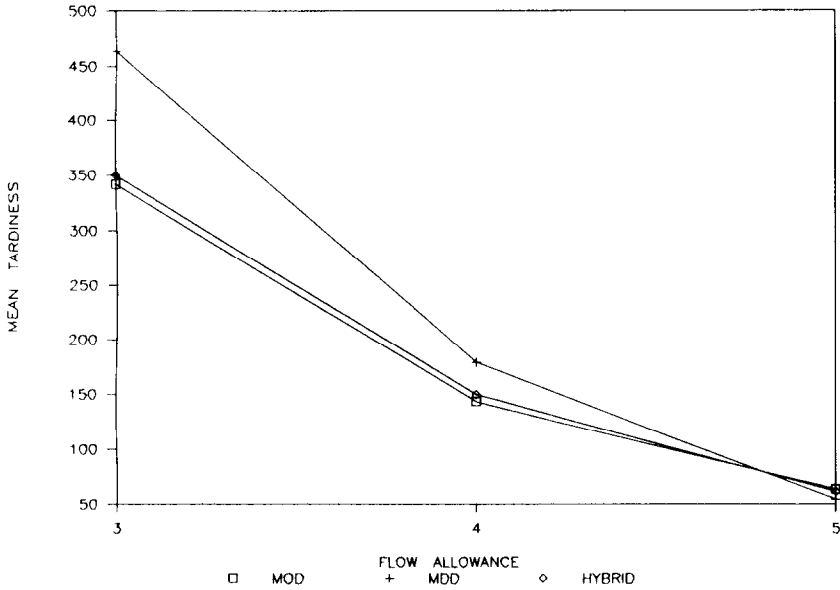
ODD Assignment Rule	Flow Allowance			UN(1,7)
	3	4	5	
TWK	1213.1	878.1	661.0	987.7
PWL	1222.5	885.8	671.3	992.7
IPWL	1223.5	880.2	632.3	980.4

The relative performance of the dispatching rules with respect to mean tardiness under balanced workloads is depicted in Figures 1 and 2 for the cases of deterministic and random flow allowances respectively. Consistent with Baker's (1984) findings, MOD is, in general, superior to the other rules although it is not significantly different from HYBRID. When due dates are set randomly, these three rules yield similar values. As shown in Figure 2, randomness in due date generation leads to a significant increase in tardiness for all three scheduling rules.

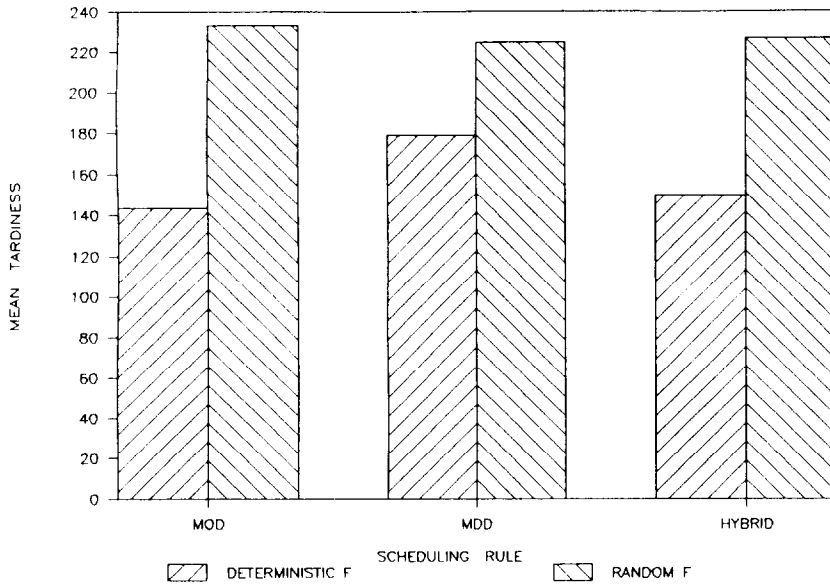
However, as seen from Figure 3, HYBRID emerges as the best rule across all flow allowances for unbalanced workloads. The difference between HYBRID and MOD is increasingly significant with an increase in the flow allowance. Figure 4 shows that while random due dates lead to a deterioration in system performance, HYBRID remains the most effective rule in this case as well.

Table 4 depicts the standard deviation of tardiness values. Once again, the overall performance of HYBRID is the best among the three rules for unbalanced workloads. The significance level of the difference between HYBRID and the other rules increases with an increase in flow allowance; at flow allowance of 5 this difference is significant at 85% confidence level. However, MDD is marginally superior when due dates are set randomly. HYBRID's effectiveness is carried forward to the case of balanced workloads as well when due dates are set tightly.

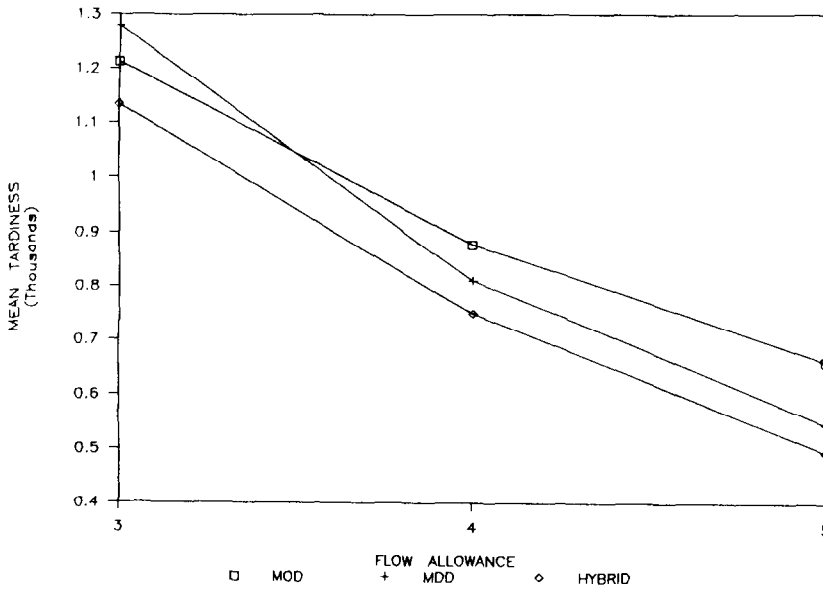
**FIGURE 1  
BALANCED WORKLOADS  
DETERMINISTIC FLOW ALLOWANCE**



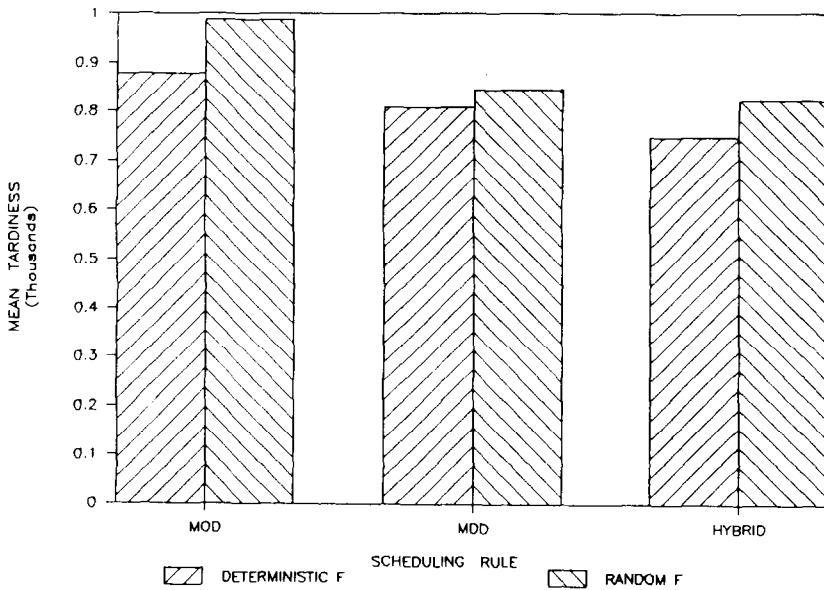
**FIGURE 2  
BALANCED WORKLOADS  
DETERMINISTIC VS. RANDOM FLOW ALLOWANCES**



**FIGURE 3**  
**UNBALANCED WORKLOADS**  
**DETERMINISTIC FLOW ALLOWANCE**



**FIGURE 4**  
**UNBALANCED WORKLOADS**  
**DETERMINISTIC VS. RANDOM FLOW ALLOWANCES**



**TABLE 4**  
**STANDARD DEVIATION OF TARDINESS**

Scheduling Rule	Degree of Workload Imbalance	Flow Allowance			
		3	4	5	UN(1,7)
MOD	Balanced	699.8	450.9	285.8	468.9
	Unbalanced	1787.6	1563.0	1340.6	1554.3
MDD	Balanced	803.6	479.6	230.3	421.9
	Unbalanced	1768.6	1342.5	1013.2	1283.2
Hybrid	Balanced	671.7	429.6	246.9	420.9
	Unbalanced	1702.4	1260.3	943.8	1337.5

As seen from Table 5, the relative performance of these three rules follows a similar pattern for the measure of mean flow time. HYBRID remains the best rule at all flow allowances for unbalanced workloads. When workloads are balanced, it is superior when due dates are tight.

**TABLE 5**  
**MEAN FLOW TIME**

Scheduling Rule	Degree of Workload Imbalance	Flow Allowance			
		3	4	5	UN(1,7)
MOD	Balanced	2607.1	2701.3	2771.2	2707.5
	Unbalanced	3619.2	3733.5	3851.7	3765.5
MDD	Balanced	2751.6	2713.3	2667.2	2694.9
	Unbalanced	3686.7	3620.6	3629.8	3596.4
Hybrid	Balanced	2589.3	2645.9	2684.2	2672.5
	Unbalanced	3520.9	3537.0	3543.2	3562.1

Table 6 depicts the performance of these rules with respect to the proportion of tardy jobs. For balanced workloads, HYBRID gives the best results, though MOD is nearly as effective. For unbalanced workloads, however, MOD is easily seen to be the best rule across all values of due date tightness.

**TABLE 6  
PROPORTION OF TARDY JOBS**

Scheduling Rule	Degree of Workload Imbalance	Flow Allowance			
		3	4	5	UN(1,7)
MOD	Balanced	0.276	0.115	0.045	0.261
	Unbalanced	0.383	0.209	0.132	0.344
MDD	Balanced	0.334	0.137	0.049	0.264
	Unbalanced	0.456	0.266	0.171	0.376
Hybrid	Balanced	0.271	0.106	0.042	0.258
	Unbalanced	0.427	0.253	0.155	0.371

Table 7 shows the results of the second experiment for mean tardiness under unbalanced workloads. All four scheduling rules yield similar values; however, BB yields the best overall results. Its superiority is retained for the case of randomly assigned due dates as well. The results for other performance measures have been omitted as they indicate very little difference among the various scheduling rules.

**TABLE 7  
MEAN TARDINESS  
(UNBALANCED WORKLOADS)**

Scheduling Rule	Flow Allowance			
	1.12	1.22	1.30	UN(1.0,1.44)
MOD	56.9	30.8	18.4	39.6
MDD	57.5	31.5	19.2	40.2
HYBRID	56.7	30.9	18.8	39.7
BB	56.8	30.3	18.4	38.4

## SUMMARY

This paper examines three questions of practical interest: (i) When machine workloads are unbalanced, does applying different scheduling rules at bottleneck and non-bottleneck machines lead to improved system performance? (ii) How should job due dates be decomposed into operations due dates (ODDs) when workloads are unbalanced? (iii) Is it effective to implement optimum-seeking solution procedures to the static problem on a rolling basis in a dynamic environment?

This study gives some answers to these questions. For unbalanced workloads, the selection of a different (and appropriate) scheduling rule at the bottleneck and non-bottleneck machines improved the system performance substantially. In particular, the HYBRID rule produced results which were both effective and robust across varying due date tightness levels.

On the other hand, the three ODD assignment procedures investigated in this study yielded similar tardiness values in spite of the fact that they generated substantially different operation due dates. This result indicates that, when workloads are unbalanced, mean tardiness may not be overly sensitive to the job flow allowance factor. The experimental results suggest that a consistent job due date assignment policy, through the use of a constant flow allowance factor, is likely to improve system performance.

The results given in this paper reveal that the efficacy of implementing near-optimal solutions to the static problems on a rolling basis reported in Rachamadugu et al. (1986) is extended to a multiple machine system. This extension is partial because the relative performance of various scheduling rules could only be tested at a low utilization level for computational cost reasons. However, the reported results do indicate that by expending extra computational effort in improving the solution quality for the static problem, significant improvement over the performance of dispatching rules can be achieved. Note that for the dynamic problem BB used a time trap of 0.5 CPU second. In a real system, the CPU time trap would not be needed. Computations could continue until there was a system change (e.g., a job arrival) that triggered the need for a new schedule. Generally, this time would be in minutes or hours (not 0.5 second). Hence, larger problems associated with higher utilizations, say 80%, could be solved optimally or near optimally, even on shop floor microcomputers.

This research motivates further work along two directions. First, there is a need to study dynamic ODD assignment procedures in greater detail. Second, more effective methods for solving the static tardiness problem are required. This is known to be a hard problem to solve optimally. It may, however, be possible to generate efficient



schedules by focusing on the bottleneck machine(s). Ow (1985) shows the efficacy of this approach for a proportionate flow shop. The results shown in this paper indicate that it may be appropriate for a GFMS (and a job shop) as well.

## ACKNOWLEDGEMENT

The authors extend their appreciation to the referees and the editor for their constructive comments. This research was partially supported by the National Institute of Standards and Technology (formerly National Bureau of Standards) and the University of Michigan School of Business.

## REFERENCES

1. Baker, K.R. "Sequencing Rules and Due Date Assignments in a Job Shop." *Management Science*, vol. 30, no. 9, September 1984.
2. Baker K.R., and J.M.W. Bertrand. "A Dynamic Priority Rule for Sequencing Against Due Dates." *Journal of Operations Management*, vol. 3, no. 1, November 1982, 37–42.
3. Baker, K.R., and J.J. Kanet. "Job Shop Scheduling with Modified Due Dates." *Journal of Operations Management*, vol. 4, no. 1, November 1983, 11–22.
4. Carroll, D.C. "Heuristic Sequencing of Single and Multiple Component Jobs." Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge MA, 1965.
5. Chand, S. "A Note on Dynamic Lot Sizing in a Rolling Horizon Environment." *Decision Sciences*, vol. 13, no. 1, January 1982, 113–119.
6. Groover, M.P. *Automation, Production Systems and Computer-Aided Manufacturing*. Englewood Cliffs: Prentice-Hall, 1980.
7. Jones, A., and C.J. McLean. "A Proposed Hierarchical Control Model for Automated Manufacturing Systems." *Journal of Manufacturing Systems*, vol. 5, no. 1, 1986, 15–25.
8. Kanet, J.J., and J.C. Hayya. "Priority Dispatching with Operation Due Dates in a Job Shop." *Journal of Operations Management*, vol. 2, no. 3, May 1982, 155–163.
9. Merchant, M.E. "World Trends in Flexible Manufacturing Systems." *The FMS Magazine*, vol. 1, 1982, 4–5.
10. Ow, P.S. "Focused Scheduling in Proportionate Flowshops." *Management Science*, vol. 31, no. 7, July 1985, 852–869.
11. Panwalker, S.S., R.A. Dudek, and M.L. Smith. "Sequencing Research and the Industrial Problem." in *Symposium in the Theory of Scheduling and Its Applications*, S.E. Elmaghraby (ed.) New York: Springer, 1973, 29–38.
12. Rachamadugu, R.V., and K.E. Stecke. "Classification and Review of FMS Scheduling Procedures." Working Paper No. 481C, Graduate School of Business Administration, University of Michigan, Ann Arbor MI, 1988.

13. Rachamadugu, R.V., N. Raman, and F. B. Talbot. "Real Time Scheduling of an Automated Manufacturing Center." *Proceedings of the Conference on Real Time Optimization in Automated Manufacturing Facilities*. Gaithersburg MD: National Bureau of Standards, January 1986, 293—316.
14. Raman, N., F.B. Talbot, and R.V. Rachamadugu. "Due Date Based Scheduling in a General Flexible Manufacturing System." Working Paper, University of Illinois at Urbana—Champaign, Champaign IL, 1988.
15. Simpson, J.A., R.J. Hocken, and J.S. Albus. "The Automated Manufacturing Research Facility of the National Bureau of Standards." *Journal of Manufacturing Systems*, vol. 1, no. 1, 1982, 17—32.
16. Smith, M.L., R. Ramesh, R.A. Dudek, and E.L. Blair, "Characteristics of U.S. Flexible Manufacturing Systems—A Survey." *Proceedings of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems*. New York: Elsevier, 1986, 477—486.
17. Talbot, F.B. "Resource Constrained Project Scheduling with Time Resource Tradeoffs: The Nonpreemptive Case." *Management Science*, vol. 28, no. 10, October 1982, 1197—1210.
18. Vepsalainen, A.P.J., and T.E. Morton. "Priority Rules for Job Shops with Weighted Tardiness Costs." *Management Science*, vol. 33, no. 8, August 1987, 1035—1047.