Proceedings of the 1997 Winter Simulation Conference
ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson

SELECTIVE REROUTING USING SIMULATED STEADY STATE SYSTEM DATA

Catherine M. Harmonosky

Department of Industrial and Manufacturing Engineering
Penn State University
University Park, PA 16802 U.S.A

Robert H. Farr

RPS, Inc.
P.O. Box 108
Pittsburgh, PA 15230 U.S.A.

Ming-Chuan Ni

434, Sec. 4
Ting-Tsao Road
Changhua, 50509 TAIWAN

ABSTRACT

Effective operational control of a manufacturing system that has routing flexibility is dependent upon being able to make informed real-time decisions in the event of a system disruption, such as a machine breakdown or parts shortage. This paper presents a methodology for making a real-time selective rerouting decision using steady-state system performance estimates from simulation models run a priori to any system disturbance. The approach is to create a relatively simple tool based on simulated long-term system performance data. Evaluation of the effectiveness of the approach is based upon system performance measures of average flowtime and average throughput.

1 MOTIVATION

Routing flexibility in a manufacturing system is possible when different machines can perform the same operation, which allows jobs to follow several possible machine sequences (Browne et al. 1984). From an operational control viewpoint, this flexibility may be used in two ways. First, it might be used for continuous real-time scheduling, where every decision regarding where to route a job next is made as needed as time moves forward (a.k.a. "on the fly"). Second, it might be used for exception real-time scheduling, where a real-time scheduling decision is only made when a disruption occurs, such as a machine breakdown or parts shortage, which makes it difficult to adhere to the originally planned schedule. Then, jobs can be sent to alternative machines, rather than sitting idle in a queue. In either case, routing flexibility results in more operational control complexity due to the increased number of routing options that must be considered.

Although optimal algorithms and heuristics have been developed to schedule jobs in a flexible manufacturing system with respect to some particular performance measure, there is a need to develop effective methods for rescheduling when a system disruption occurs such that the global system performance measure is directly considered. It is sometimes assumed that all jobs at a machine experiencing a disruption should be rerouted, hoping that job waiting time might be reduced and consequently throughput might be increased. However, if the disruption will be short, it may be better for the global performance measure to not reroute all jobs. Although a performance measure such as flowtime might improve for an individual job, the effect upon other jobs in the system due to rerouting one job to an alternate machine may cause the average flowtime of the system to increase. This type of exception real-time scheduling will be the focus of this paper.

This paper presents a methodology for making a real-time selective rerouting decision using steady-state system performance estimates from simulation models run a priori to any system disturbance. By implementing such a heuristic, relatively quick decisions can be made which will reduce overall waiting time of jobs minimizing losses due to system disruptions.

2 RELATED PAST WORK

In the area of exception real-time scheduling, both simulation-based and non-simulation-based approaches have appeared in the literature. Regarding non-simulation-based approaches, some techniques use various system information to choose between different dispatching rules (Ishii and Talavage 1991 and 1994, Kim 1990, Slomp et al. 1988, Yammamoto and Nof 1985). Dynamic programming was used by Maimon and Gershwin (1988) to consider instantaneous capacity of the system when rerouting. A graph theoretic approach was described by Leon et al. (1994) to react to disturbances and to make an initial schedule more robust to system disturbances. Dutta (1990) used a knowledge-based methodology to automatically take corrective action when exceptions occur trying to maintain the original system performance. Bean et al. (1991) present a method that reconstructs a portion of the schedule to eventually match up with the original schedule at some future point using integer programming and dynamic priority rule assignments. For a review of manufacturing systems scheduling see Basnet and Mize (1994) and Rachamadugu and Stecke (1994).

Regarding simulation-based approaches, Merchawi and ElMaraghy (1996) describe an approach to hierarchical simulation modeling of flexible manufacturing systems which controls the level of detail of simulations that may be used for on-line real-time decision support. Simulation combined with a knowledge base has been presented by several authors (Katz and Manivannan 1993, Manivannan and Banks 1991 and 1992). Using simulation to evaluate several alternatives when a disruption occurs has been discussed for real-time decision making (Harmonosky and Robohn 1995, Kim and Kim 1994, Harmonosky 1990). Also, some commercial simulation products, such as ARENA by Systems Modeling Corporation and FACTOR Production Manager, provide some real-time shop floor linking capability. For a review of simulation-based real-time scheduling, see Harmonosky (1995).

With both simulation-based and non-simulation-based approaches, issues of complexity of the approach and how quickly a decision can really be made must be considered before implementation can occur. Further, there is often the underlying assumption that rerouting is always the best approach and the long-term effects upon system performance of short-term decisions are not specifically considered.

3 METHODOLOGY

This section will present an approach to selectively reroute jobs in the event of a machine failure, attempting to improve the average system flowtime and system throughput. It is based upon steady state estimates of mean queue length and mean waiting times for each machine obtained from simulations of the manufacturing system run with no breakdowns. The simulations are done off-line long before any actual system breakdowns occur. Although other methods for estimating steady state mean queue time and mean queue length exist (e.g. mean value analysis), using simulation provides very accurate values since the model can be very detailed with no underlying assumptions about processing time distributions, material handing delays, etc. Also, assuming there are multiple part types in the system, a steady state estimate for the mean processing time for each machine (over all part types) may be obtained from the simulation.

At the time of a breakdown, these a priori estimates are used as input to the selective routing procedure to immediately consider jobs in the queue at the machine that is down and later consider newly arriving jobs. The only input data required from the physical system is an estimate of the breakdown duration, typically an estimate of repair time. Consequently, a decision regarding whether or not to reroute a job can be made in a matter of seconds, making this a truly real-time scheduling tool.

3.1 Environment

Some assumptions regarding the environment in which this selective rerouting approach is to be used must be discussed. The selective rerouting approach is to be used in day-to-day operations in a manufacturing system with routing flexibility in reaction to a machine failure. It is assumed that the operation sequence for each job type is fixed and selecting an alternate machine for rerouting depends only upon the type of operation needed not the job type. Accommodation for a time penalty, stated as a percentage of the processing time, to account for some additional time to move the job or minor set-up at the alternate machine is included.

3.2 Selective Rerouting Approach

When trying to selectively reroute jobs, there is the potential for having a massive rerouting at the time of machine failure if the influence upon the whole system when the job is rerouted is not appropriately considered. In other words, if the decision criteria is too myopic, the result may be no different than a simple 'reroute all' policy. The approach used here is borrowed from economics where a company will raise their level of production only so long as the profit is increasing to ensure that the benefit of rerouting one job is positive to the global performance measure.

The procedure begins in the event of a machine failure and uses the previously obtained steady-state values from simulation, mean queue length, mean waiting time, and mean processing time, as input along with one input from the physical system, breakdown duration, D. First, the heuristic will handle the jobs at the failed machine queue one by one. Next, the heuristic considers newly arriving jobs at the failed machine until the machine is repaired. Basically, the approach compares anticipated waiting time at the failed machine with the anticipated waiting time at the alternate machine plus the influence that rerouting has upon global system performance measured by increase in queue time for other jobs.

Step 1. Consider job 1 in the queue of the failed machine, machine k. Find $S(a)$, minimum steady state expected waiting time, $W(m)$, over set J_k containing all alternative machines for machine k as follows:

$$S(a) = \min_{m \in J_k} W(m) \tag{1}$$

where a is the machine associated with the minimum alternative processing time.

Step 2. The waiting time for a job i at the failed machine needs to reflect the job's position in queue and the duration of the breakdown. For each job, the steady-state waiting time in queue is weighted by the ratio of rank in queue, n_i, to steady-state mean queue length for the failed machine, $Q(k)$, to yield $W_i(k)$, according to the following equation:

$$W_i(k) = n_i / Q(k) * W(k) \tag{2}$$

Then the total expected waiting time at the failed machine k for the job i, $W_i(k)'$, is the estimated duration of the breakdown plus this weighted expected waiting time:

$$W_i(k)' = W_i(k) + D \tag{3}$$

Step 3. To calculate the amount of waiting time for this job when it is rerouted to the alternate machine queue, the extra time for the job to reroute to the alternate machine is the mean processing time of the machine, $T_a(m)$, multiplied by a penalty percentage, C. Then the amount of waiting time for this job if it is sent to the alternate machine queue is $S(a)$ plus the extra time:

$$W_i(a) = [C * T_a(m) + S(a)] \tag{4}$$

Step 4. Rerouting will influence the time spent in queue for each job coming into the queue behind the rerouted job by the mean processing time multiplied by (1+C). Thus, the influence to the whole system when the job is rerouted is $Q(a)$ multiplied by the mean processing time multiplied by (1+C):

$$W_i(a)' = [Q(a) * T_a(m) * (1 + C)] \tag{5}$$

Step 5. If $[C * T_a(m) + S(a)] + [Q(a) * T_a(m) * (1 + C)] <$, $W_i(k)'$, then re-route job 1 to machine a. If this inequality is true, then it means rerouting the job will yield positive benefits for global system performance and the job should be rerouted to its alternate machine. Otherwise, the job should remain in queue at failed machine k.

Step 6. If the job is rerouted, increment the expected waiting time at the alternate machine $W(a)$ by the mean processing time of the machine plus the extra time associated with the rerouted job, and increment the mean queue length of the alternate machine $Q(a)$ by one to reflect the rerouting using the following equations:

$$W(a) = S(a) + [(1 + C) * T_a(m)] \tag{6}$$

$$Q(a) = Q(a) + 1 \tag{7}$$

This reflects the actual situation of the system and prevents a massive rerouting to the alternate machine which could cause a bottleneck at that machine.

Step 7. Repeat steps 1-6 for all other jobs (i = 2,3,...r) in queue at machine k. Note that all rerouting decisions for the r jobs in queue of machine k are assumed to be made as soon as a breakdown occurs.

Step 8. For new jobs arriving at machine k before machine k has been repaired, waiting time at the failed machine is expressed as the sum of the original steady-state expected waiting time for a job in queue and the estimated time remaining until the failed machine will be repaired according to the following equation:

$$W_{new}(k) = W(k) + (t_{BD} + D + t_{NOW}) \tag{8}$$

where $(t_{BD} + D + t_{NOW})$ represents the estimated time remaining until machine k is repaired.

The updated mean waiting times $W(a)$ and mean queue lengths $Q(a)$ resulting from any rerouting will be used.

Step 9. If $W_i(a) + [Q(a) * T_a(m) * (1 + C)] < W_{new}(k)$, then reroute the newly arriving job to the alternate machine. Update $W(a)$ and $Q(a)$ to reflect the rerouting using equations (6) and (7). Otherwise, the newly arriving job remains in queue at machine k.

Step 10. When one rerouted job is completed during the duration of the machine failure, $Q(a)$ should subtract 1 to reflect the current situation. The mean waiting times and mean queue lengths of all machines will be reset to their original values once the machine is repaired.

4 EXPERIMENTATION AND EVALUATION

Two different systems were simulated to test the effectiveness of the selective rerouting procedure. The first is based on Example 4C from Pegden [1987, pp. 116] which was modified by Wisser [1990]. The second system comes from an example provided by Systems Modeling Corporation [1989]. The two systems differ only in the number of machines and their associated processing time. Both systems are closed networks with 36 dedicated pallets. There are three job types with 12 pallets assigned to job type 1, 9 pallets assigned to job type 2, and 15 pallets assigned to job type 3. Each machine has one alternate machine which can perform the same operation with the same processing time plus a C penalty time to account for any additional setup.

Experimentation was done to evaluate the effectiveness of the procedure under different system conditions and to determine what effect certain system parameters had upon this effectiveness. This experiment was designed with four factors that could affect effectiveness, test system, time between machine breakdowns, duration of machine breakdown, and penalty percentage. Each factor was set at two levels as shown in Table 1. In addition to running a full factorial experimental design with these factors, runs were made at all design points for the case with all jobs rerouted and at appropriate design points for the case where no rerouting occurred. At each design point there were 10 replications of the simulation model. Each simulation was run for 20000 time units to allow for a sufficient amount of time for breakdowns and rerouting to occur. The initial data in the first 480 time units was also discarded to allow the system to warm up. For each run of the simulation model average flowtime and average throughput were recorded. Normal probability plots of the effect estimates were used to determine which factors significantly affect the system performance measure when using the selective rerouting procedure.

The performance measures of interest are average flowtime and average throughput. But, since there will be different levels of disruption in different systems based on the combination of the time between machine failures and the duration of the failure, a direct comparison of the performance measures is not always feasible, so a relative comparison is more desirable. The relative comparison used here is based on the assumption that the theoretical lower bound for average flowtime and the upper bound for average throughput values for each system will be the values determined at steady state with no machine failures. Also, a good point of comparison is the case where no rerouting would occur. Consequently, average flowtime and average throughput performance measures will be presented as a percentage recovery (or improvement) of how much closer to the theoretical lower bound the rerouting heuristic obtained compared to doing nothing.

The percentage flowtime recovery is calculated as follows:

$$\begin{array}{c}\textit{Percentage}\\ \textit{flowtime}\\ \textit{recovery}\end{array} = 1 - \frac{\begin{array}{c}\textit{Flowtime}\\ \textit{with}\\ \textit{rerouting}\end{array} - \begin{array}{c}\textit{Steady}\\ \textit{state}\\ \textit{flowtime}\end{array}}{\begin{array}{c}\textit{Flowtime}\\ \textit{without}\\ \textit{rerouting}\end{array} - \begin{array}{c}\textit{Steady}\\ \textit{state}\\ \textit{flowtime}\end{array}}$$

The percentage throughput recovery is calculated as follows:

$$\begin{array}{c}\textit{Percentage}\\ \textit{throughput}\\ \textit{recovery}\end{array} = 1 - \frac{\begin{array}{c}\textit{Steady}\\ \textit{state}\\ \textit{throughput}\end{array} - \begin{array}{c}\textit{Throughput}\\ \textit{with}\\ \textit{rerouting}\end{array}}{\begin{array}{c}\textit{Steady}\\ \textit{state}\\ \textit{throughput}\end{array} - \begin{array}{c}\textit{Throughput}\\ \textit{without}\\ \textit{rerouting}\end{array}}$$

A summary of experimental results is in Table 2. Over all design points, the percentage of flowtime recovery for the selective rerouting procedure is 36.8% and percentage of throughput recovery is 26.1%. Interestingly, the results when each design point was run with rerouting all jobs every time there was a breakdown showed the average percentage of flowtime recovery for rerouting all jobs is approximately -55% and percentage of throughput recovery is approximately -50%. This means that rerouting all jobs made flowtime and throughput values substantially worse than doing no rerouting at all. The factors that were statistically significant were duration of breakdown, time between breakdown, and test system. The procedure performed best in the larger system with long duration of breakdown and more frequent breakdowns. Under these conditions the recoveries were 43.2% for flowtime and 31.2% for throughput.

Table 1: Two Level Factor Settings

Factor	Low	High
Test System	# of machines = 6	# of machines = 4
Time between breakdowns	Unif(120,360)	Unif(360,600)
Duration of breakdown	Unif(15,60)	Unif(240,480)
Penalty Percentage	5%	15%

Table 2: Experimental Results for System Performance

CASE	% FLOWTIME RECOVERY	%THROUGHPUT RECOVERY
Selective Rerouting Procedure - Average over all design points	36.8%	26.1%
Selective Rerouting Procedure - best case	43.2%	31.2%
Rerouting All Jobs	-55%	-50%

5 CONCLUSION

This paper presented an approach to real-time scheduling in case of a machine failure in a manufacturing system with alternate routing flexibility. Using simulation off-line well before any breakdowns occur provides a priori accurate estimates of long-term steady-state system performance measures that are input to a selective rerouting procedure used at the time of breakdown. This provides a link to global system performance in the real-time decision mode. Experimentation using simulated systems with the selective rerouting procedure, which only reroutes a job if there is a benefit to the global system performance and not just benefit to that job, showed that decisions could be made in a matter of seconds, truly real-time, with overall percentage of flowtime recovery of 36.8% and percentage of throughput recovery of 26.1%. For design points with the most favorable conditions for this procedure (larger system with long duration of breakdown and more frequent breakdowns) the recoveries were 43.2% for flowtime and 31.2% for throughput. Also, experimentation with rerouting all jobs in the case of a failure, which may seem to be a reasonable approach, showed that global system performance measures for flowtime and throughput were worse than doing nothing at all. Further testing of the procedure, particularly in an actual system would be desirable.

REFERENCES

Basnet, C., and J. H. Mize. 1994. Scheduling and control of flexible manufacturing systems: a critical review. *International Journal of Computer Integrated Manufacturing* 7:340-355.

Bean, J. C., J. R. Birge, J. Mittenthal, and C. E. Noon. 1991. Matchup scheduling with multiple resources, release dates, and disruptions. *Operation Research* 39: 470-483.

Browne, J., D. Dubois, K. Rathmill, S. P. Sethi, and K. E. Stecke. 1984. Classification of flexible manufacturing systems. *The FMS magazine* 22:114-117.

Dutta, A. 1990. Reacting to scheduling exceptions in FMS environments. *IIE Transactions* 22:300-314.

Harmonosky, C. M. 1990. Implementation issues using simulation for real-time scheduling, control, and monitoring. In *Proceedings of the 1990 Winter Simulation Conference*, ed. O. Balci, R. P. Sadowski, and R. E. Nance, 595-598. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Harmonosky, C. M. 1995. Simulation-based real-time scheduling: review of recent developments. In *Proceedings of the 1995 Winter Simulation Conference*, ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman, 185-191. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Harmonosky, C. M., and S. F. Robohn. 1995. Investigating the application potential of simulation to real-time control decisions. *International Journal of Computer Integrated Manufacturing* 8:126-132.

Ishii, N., and J. J. Talavage. 1991. A transient-based real-time scheduling algorithm in FMS. *International Journal of Production Research* 29: 2501-2520.

Ishii, N., and J. J. Talavage. 1994. A mixed dispatching rule approach in FMS scheduling. *The International Journal of Flexible Manufacturing Systems* 6: 69-87.

Katz, D., and S. Manivannan. 1993. Exception management on a shop floor using online simulation. In *Proceedings of the 1993 Winter Simulation Conference*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 888-896. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Kim, M. H., and Y. Kim. 1994. Simulation-based real-time scheduling in a flexible manufacturing system. *Journal of Manufacturing Systems* 13:85-93.

Kim, Y. 1990. A comparison of dispatching rules for job shops with multiple identical jobs and alternative routings. *International Journal of Production Research* 28:953-962.

Leon, V. J., S. D. Wu, and R. H. Storer. 1994. A game-theoretic control approach for job shops in the presence of disruptions. *International Journal of Production Research* 32:1451-1476.

Maimon, O. Z., and S. B. Gershwin. 1988. Dynamic scheduling and routing for flexible manufacturing systems that have unreliable machines. *Operation Research* 36:279-292.

Manivannan, S., and J. Banks. 1991. Real-time control of a manufacturing cell using knowledge-based simulation. In *Proceedings of the 1991 Winter Simulation Conference*, ed. B.L. Nelson, W.D. Kelton, and G.M. Clark, 251-260. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Manivannan, S., and J. Banks. 1992. Design of a knowledge-based on-line simulation system to control a manufacturing shop floor. *IIE Transactions* 24:72-83.

Merchawi, N. S. and H. A. ElMaraghy. 1996. Hierarchical modeling and simulation for real-time decision support in flexible manufacturing systems, *Manufacturing Systems* 25:2: 1-7.

Pegden, C. D. 1987. *Introduction to SIMAN*. Systems Modeling Corporation.

Rachamadugu, R., and K. E. Stecke. 1994. Classification and review of FMS scheduling procedures. *Production Planning & Control* 5:2-20.

Slomp, J., G. J. C. Gaalman, and W. M. Nawijn. 1988. Quasi on-line scheduling procedures for flexible manufacturing systems. *International Journal of Production Research* 26:585-598.

Systems Modeling Corporation. 1989. Four machine job shop with station visitation sequences. *Introduction to Simulation Using Siman Basic Course*.

Wisser, A. L. 1990. Analyzing system controllability with respect to system flexibility. Master's thesis, Department of Industrial and Manufacturing Engineering, Pennsylvania State University, University Park, Pennsylvania.

Yammamoto, M. and S. Y. Nof. 1985. Scheduling/ rescheduling in the manufacturing operation system environment. *International Journal of Production Research* 23:705-722.

AUTHOR BIOGRAPHIES

CATHERINE M. HARMONOSKY is an Associate Professor in the Department of Industrial and Manufacturing Engineering at The Pennsylvania State University. Dr. Harmonosky's research interests are manufacturing and production systems analysis, scheduling, and computer integrated manufacturing. Her special interests include integrated scheduling and control methodology development, studying the impact of system design decisions upon production control strategies and system costs, and innovative application of existing analytical techniques, particularly simulation. Dr. Harmonosky received her B.S.I.E. from Penn State in 1981 and her M.S.I.E. and Ph.D. from Purdue University in 1982 and 1987, respectively. She is an active member of IIE, SME, SWE, and NSPE.

ROBERT H. FARR is an Operations Engineering Analyst with RPS, Inc., a subsidiary of Caliber System Inc. Caliber System is a leading provider of value-added transportation, logistics and related information services. His current work assignments include operating cost analysis and forecasting, and productivity improvement projects. He received his B.S. in Industrial Engineering from Penn State University in 1992, and his M.S. in Industrial Engineering and Operations Research in 1994. He is a member of IIE and INFORMS.

MING-CHUAN NI has a Master's degree in Industrial Engineering from the Pennsylvania State University. His thesis research was in the area of applications of simulation to real-time scheduling problems.