Proceedings of the 1987 Winter Simulation Conference
A. Thesen, H. Grant, W. David Kelton (eds.)

USING MATHEMATICAL PROGRAMMING AND SIMULATION TO STUDY FMS MACHINE UTILIZATIONS

Thomas J. Schriber
Kathryn E. Stecke
Graduate School of Business Administration
The University of Michigan
Ann Arbor MI 48109-1234 USA

ABSTRACT

Mathematical programming can be used to determine, from a set of part-type orders, an input stream composition which maximizes machine utilizations in an FMS (flexible manufacturing system) composed of specified machining resources. Simulation can then be used to estimate the degradation in these utilizations due to their dependency on the following factors ignored in the mathematical programming solution: (1) *secondary FMS resources* (e.g. pallets and fixtures; loading and unloading stations; buffers; type and capacity of equipment for transferring work-in-process); (2) *geometric considerations* (e.g., location of loading and unloading stations, machines, and buffers; routes for transfer of work-in-process); (3) *secondary time requirements* (e.g., transfer times; palletizing and depalletizing times; fixturing, defixturing, and refixturing times); (4) *operating procedures* (e.g., quantity of work-in-process; dispatching rules; part input sequence); (5) *operating discontinuities* (e.g., machine breakdowns; scheduled machine maintenance; machine substitution; breakdowns and/or maintenance of equipment for transferring work-in-process); and (6) *secondary job characteristics* (e.g., the sequence in which parts use machines; fixturing and refixturing requirements; due dates; lateness penalties). This paper presents an example illustrating the sequential use of mathematical programming and simulation to: (1) determine the level of selected secondary FMS resources required to maximize machine utilizations when transfer times are realistic; (2) estimate the degradation in machine utilizations when there are inadequate secondary resources; and (3) determine the sensitivity of machine utilizations to selected operating procedures.

1. INTRODUCTION

Section 2 comments on the use of mathematical programming to determine input stream compositions which maximize machine utilizations in an FMS, and presents a specific FMS problem and its mathematical programming solution. Section 3 contrasts the assumptions of the aggregate mathematical programming solution with conditions in a realistic FMS. Section 4 describes the assumptions made in this paper to build a moderately detailed simulation model for a system synthesized from the problem described in Section 2. Section 5 comments on model verification. Section 6 summarizes the experimental settings, and Section 7 displays and discusses selected experimental results. Sections 8, 9, and 10 provide conclusions, suggestions for further work, and references.

2. THE USE OF MATHEMATICAL PROGRAMMING TO MAXIMIZE FMS MACHINE UTILIZATIONS

Assume the machining resources in an FMS consist of 1 mill, 2 drills, and 2 vertical turret lathes (VTL's). Suppose orders exist for 10 types of parts to be produced by this FMS, with the various part-type machining and production requirements shown in Table 1. In run 1 of the production process, the FMS is to be used to build a subset of the 10 part types. (Run 1 ends when the production requirement for one of the part types in this subset has been met.) The input stream composition (that is, the subset of part types to be built, and the proportions in which these part types are to be built) is to be determined so that the run 1 utilization of the FMS machining resources is maximized.

Part Type	Machining Times, Minutes			Total Parts Ordered
	Mill	Drill	VTL	
1	10	60	50	60
2	15	20	40	50
3	40	10	30	30
4	30	20	20	30
5	10	50	20	35
6	10	30	20	45
7	20	10	10	15
8	15	20	30	25
9	25	10	20	30
10	5	40	40	50

Table 1: Specifications for a specific FMS problem

Stecke (1985) has shown how mathematical programming can be used to solve problems of the type just described. And Stecke and Kim (1986) have presented a solution to the specific problem stated above. (Refer to these articles for details, and to Stecke and Kim (1986)) for treatment of the problem of how to operate the FMS *after* run 1 has been completed.) The input stream maximizing run 1 FMS machine utilizations is composed of part types 2, 5, 6, 8, and 10; and the respective proportions of these part types in the input stream are 2, 1, 2, 1, and 1. (That is, of every 7 parts built, 2 are to be of type 2; 1 is to be of type 5; 2 are to be of type 6; etc.) The *overall* run 1 machine utilization resulting from this input stream composition is 95.2%. ("Overall" machine utilization is the average of the individual machine utilizations. In this paper, the phrases *overall machine utilization*, *overall utilization*, and *machine utilizations* are used interchangeably.)

3. MATHEMATICAL PROGRAMMING VS. REALISTIC FLEXIBLE MANUFACTURING SYSTEMS

The mathematical programming solution for the Section 2 problem is *aggregate* in the sense that, in addition to the part-type information in Table 1, it only takes into account the *machining* resources of the FMS. The solution ignores *secondary* FMS resources such as pallets and fixtures, loading and unloading stations, buffers for work-in-process (WIP), and resources for transferring WIP from point to point in the system.

Furthermore, the mathematical programming solution ignores such *geometric aspects* of the FMS as the location of the various system resources, and the routes for transfer of WIP (such as pathways for automated guided vehicles, and/or the placement of conveyors). The geometry of a system can play an important role in system operation.

Nor does the mathematical programming solution take into account such *secondary time requirements* as the time required for WIP transfer, or the time required for palletizing, depalletizing, fixturing, defixturing, and refixturing.

FMS *operating procedures* are ignored in the mathematical programming solution, too. Among such operating procedures are the quantity of WIP permitted in the system; the rule used to dispatch *differing* types of WIP to machines (e.g., first-come, first-served; shortest processing time); and the rule used to dispatch *identical* types of WIP to machines (e.g., give WIP coming from another *machine* higher priority than identical WIP coming from a *buffer).*

Potential system *operating discontinuities* resulting from tool failures, other types of machine breakdowns, and the periodic withdrawal of machines from service for such things as scheduled maintenance, are also ignored in the mathematical programming solution of the machine utilization problem. The possibility of machine substitution is not considered, either. (Machine substitution is the use of one type of machine to accomplish a step normally done by another type of machine.)

Finally, the mathematical programming solution does not take into account such *secondary job characteristics* as due dates or lateness penalties. (Note that no due dates or lateness penalties are included in Table 1).

The mathematical programming solution to problems of the Section 2 type clearly ignores many potentially limiting aspects of an FMS. How closely does the *achieved* overall machine utilization in a ealistic FMS environment come to the *theoretical* overall machine utilization computed under the assumptions of the mathematical programming solution? Answers to questions of this type can be investigated with simulation modeling. Here are some fundamental questions whose answers can be explored via simulation:

1. What level (or what alternative levels) of secondary FMS resources is needed to maximize overall machine utilization?
2. For a given level of secondary FMS resources, what machine utilizations can be achieved?
3. How do achieved machine utilizations vary as a function of varying levels of secondary FMS resources?
4. What influences do *operating procedures* have on the answers to the three preceding questions?
5. What types and what levels of detail is it important to model when researching various aspects of the design and operation of an FMS? For example, under what circumstances is it satisfactory to work with average transfer time in a model, as contrasted with using specific transfer times which depend on the starting and ending locations of the WIP being transferred?

4. THE SIMULATION MODEL

The simulation model built for this work relaxes a number of the assumptions made in the mathematical programming solution to the machine utilization problem. Characteristics of the model are summarized here, using the same categories as listed in the abstract:

(1) *Secondary FMS Resources*

Automated Guided Vehicles (AGVs) transport work-in-process. The number of AGVs is a model parameter.
Buffers are provided for work-in-process. Only system-wide buffers are supplied. (That is, no machines have one or more of their own input and/or output buffers.) Any buffer can be used by any WIP unit at any stage in the manufacturing process. The number of buffers is a model parameter.

Pallets and fixtures, and loading and unloading stations, are modeled explicitly, and are model parameters. These resources are not specific to the type of work-in-process.

(2) *Geometric Considerations*

The geometry of the system is not modeled. That is, neither relative nor absolute locations of loading stations, machines, buffers, or unloading stations is taken into account; and neither the positioning of AGV guidepaths nor the point-to-point movement of AGVs is taken into account.

(3) *Secondary Time Requirements*

Transfer times, palletizing and depalletizing times, and fixturing and defixturing times, are model parameters. It is assumed that no refixturing of work-in-process is required, and so refixturing time is not an issue here.

(4) *Operating Procedures*

Either of two alternative rules can be used to dispatch *nonidentical* types of WIP to machines: first-come, first-served (FCFS); or shortest processing time (SPT). As for dispatching *identical* WIP types to machines, WIP coming from another machine has priority over WIP coming from a buffer.

Both the quantity of work-in-process permitted in the system and the part input sequence are model parameters.

(5) *Operating Discontinuities*

Neither machine breakdowns nor the periodic removal of machines from service (such as for routine machine maintenance) is modeled. Breakdowns and maintenance of equipment used for WIP transfer are not modeled. Machine substitution is not allowed. (A milling operation can only be performed on a mill; and similarly for drills and turret lathes.)

(6) *Secondary Job Characteristics*

All part types are assumed to use machining resources in a mill/drill/vertical-turret-lathe sequence. (That is, the system is operated as a flexible flow system (FFS).) Each WIP unit goes to each machine type only one time. As mentioned under (3), there is no refixturing of work-in-process.

Neither order due dates nor lateness penalties are considered.

5. MODEL VERIFICATION

The simulation model, which was built in GPSS/H (Henriksen and Crain, 1983), was *verified* (that is, the correctness of the computer code was established) by techniques reported in Schriber and Stecke (1986). These techniques included simulating with cases for which model outputs were checked against correct results determined in independent fashion; and interactively monitoring the movement of randomly chosen work-in-process as it passed through the system.

6. EXPERIMENTAL SETTINGS

All experimentation reported here is based on the assumptions that there is no limit to the number of pallets and fixtures; and that palletizing, fixturing, defixturing, and depalletizing are done external to the system (or, equivalently, are done in zero time).

All transfer times are assumed to be 2 minutes.

An important experimental variable is the *quantity of work-in-process*. In this specific problem, the *minimum* quantity of work-in-process *of interest* is 5 (because there are 5 machines in the system). The logical *maximum* for this quantity is the *minimum* of: (1) the number of pallets; (2) the number of fixtures; (3) the *sum* of the number of machines and the number of buffers. With *no limit* assumed for pallets and fixtures, the maximum quantity of work-in-process in this research equals the sum of the number of machines and the number of buffers. (Each WIP unit must be either at a machine or in a buffer, except during the 2-minute intervals when it is being transferred between consecutive locations.)

With WIP at its maximum level in this work, there are no *slack buffers* in the system. (A slack buffer is a buffer not strictly needed to support the quantity of work-in-process.) With the quantity of WIP set at 1 or more units below this maximum level, the system correspondingly has 1 or more slack buffers. Slack buffers can play a key role by reducing the occurrence of *output blocking* at machines. (Output blocking occurs when WIP cannot be removed from a machine and remains there, temporarily preventing use of the machine by the next unit of WIP.) Experiments were performed for the cases of 0, 1, 2, and 3 slack buffers.

In some experiments, *nonidentical* types of WIP were dispatched FCFS to machines; in other experiments, SPT was used.

In all experiments, *identical* types of WIP were dispatched to machines by giving WIP coming from another machine priority over WIP coming from a buffer. (This dispatching rule was chosen to minimize the number of WIP transfers from machines to buffers.)

Recall that for the first production run, the respective proportions of part types 2, 5, 6, 8, and 10 in the input stream are 2, 1, 2, 1, and 1 (see Section 2). Corresponding to these proportions, the part input sequence used in this work, expressed in terms of part-type numbers, is 2, 6, 5, 2, 8, 6, and 10. That is, the first part admitted to the system is of type 2; the next part admitted is of type 6; the next admitted after that is of type 5; then another part of type 2 is admitted; and so on. This input sequence is cycled through repeatedly.

For each experimental setting, appropriate statistics were recorded under conditions of operating equilibrium for 250 8-hour shifts. (For example, overall machine utilization for shift 1, shift 2, shift 3, ..., shift 250, was recorded.) Operating equilibrium was brought about and tested for by techniques described in (4). ("Operating equilibrium" is not the same as statistically stationary operating conditions, which cannot be achieved in these experiments because of the cyclical way in which part types are admitted to the system.)

7. EXPERIMENTAL RESULTS

Overall machine utilizations, which are the only experimental results reported here, are expressed for each experimental setting as the average of the 250 values observed for that setting. The sample standard deviations of these utilizations were only about 1% of the sample mean, and are not reported. (Even though deterministic times were used in the model, overall machine utilization can vary from 8-hour interval to 8-hour interval because the input stream is heterogeneous and cyclic in character, as described above.)

Tables 2 and 3 (shown on the next two pages) display overall machine utilizations achieved when FCFS and SPT were respectively used as the alternative rules to dispatch nonidentical types of WIP to machines. The rows in Tables 2 and 3 correspond to WIP levels ranging from 5 to 10 in steps of 1. The columns correspond to numbers of AGVs ranging from 1 to 5 in steps of 1. For each WIP/AGV combination, overall machine utilizations are shown for the alternatives of 0, 1, 2, and 3 slack buffers.

With 2-minute transfer times, the maximum overall machine utilization achievable for these simulations is 0.892 (computed by hand). (If transfer times were zero, the overall machine utilization would be 0.952 for this problem, as computed by hand. Realistic transfer times consequently degrade the maximum feasible overall machine utilization by 6.3%, from 0.952 to 0.892.) With FCFS, this theoretical maximum was not achieved with 0 or 1 slack buffers, but was consistently achieved with 2 and 3 slack buffers when the WIP level was 7 or more and there were at least 4 AGVs. (These cases are highlighted in Table 2.) *With enough "resources," then, the overall machine utilization predicted by aggregate mathematical programming can be achieved.* The least complicated operating conditions under which this comes about in Table 2 correspond to a WIP level of 7, 4 AGVs, and 2 slack buffers.

The simplest operating conditions in Table 2 correspond to a WIP level of 5, 1 AGV, and 0 slack buffers. The overall machine utilization achieved under these conditions is 0.735, which is 82% of the theoretical maximum of 0.892. This means the number of AGVs must be increased from 1 to 4, the WIP level from 5 to 7, and the number of slack buffers from 0 to 2, to increase overall machine utilization by 21% (from 0.735 to 0.892).

As Table 2 demonstrates, slack buffers can improve system productivity by providing a temporary destination for WIP which can't be transferred immediately from its current machine to its next machine. Beyond a certain point, additional buffers are not useful. For example, there are no instances in Table 2 in which overall machine utilization is improved by going from 2 to 3 slack buffers.

As in Table 2, highlighting is used in Table 3 to indicate operating conditions which succeed in achieving the maximum overall machine utilization of 0.892. Table 3 indicates that an *advantage* of SPT is to achieve maximum overall machine utilization with *fewer resources* than required in the best case with FCFS. This happens for the case of 3 AGVs (vs. 4 for FCFS) and 1 slack buffer (vs. 2 for FCFS) at a WIP level of 7. Unlike FCFS, however, the

NUMBER OF AGVs

WIP LEVEL	1		2		3		4		5	
5	.735	.740	.757	.787	.781	.793	.781	.793	.781	.793
	.724	.724	.800	.800	.800	.800	.800	.800	.800	.800
6	.719	.754	.769	.852	.781	.877	.781	.877	.781	.877
	.760	.760	.862	.862	.869	.869	.877	.877	.877	.877
7	.730	.776	.769	.864	.784	.847	.784	.897	.787	.877
	.793	.793	.871	.871	.886	.886	.892	.892	.892	.892
8	.730	.779	.783	.829	.784	.847	.784	.879	.787	.877
	.793	.793	.888	.888	.884	.884	.892	.892	.892	.892
9	.730	.776	.783	.829	.784	.847	.784	.879	.787	.877
	.772	.772	.888	.888	.884	.884	.892	.892	.892	.892
10	.729	.776	.783	.864	.784	.877	.784	.879	.787	.877
	.799	.799	.888	.888	.884	.884	.892	.892	.892	.892

LEGEND: SLACK BUFFERS

0	1
2	3

Table 2: Overall Machine Utilizations with FIFO as the Dispatching Rule

SPT rule fails to produce maximum overall machine utilization at WIP/AGV combinations of 7/4 and 7/5. Furthermore, a *disadvantage* of SPT is that it results in WIP *system residence times* whose means and standard deviations are substantially greater than those resulting from FCFS. (Residence-time results cannot be shown in this limited space. See Schriber and Stecke (1986) for details.) This is to be expected in a system of the flow-shop type modeled here (French 1982).

In Table 3, overall machine utilization decreases in going from 3 AGVs to 4 or 5 AGVs at a WIP level of 7 with 1 slack buffer. What harm could additional AGVs do (given that AGV contention for pathways is not modeled)? By being available more often, the AGVs occasionally move WIP into buffers when the WIP would otherwise have to wait at machines. This increases the number of time-consuming WIP transfers to buffers, and while in transit to buffers, WIP is not available to machines (in the model used here). *This apparently increases the average extent to which machines are feed starved in these cases, resulting in decreased machine utilizations.*

Overall machine utilization also decreases in Table 3 in going from a WIP level of 7 to 8 (or 9 or 10) with 3 AGVs and 1, 2, or 3 slack buffers. Why might additional WIP have this effect? The dispatching rule for nonidentical WIP in Table 3 is SPT, and increasing the WIP level increases the degree of heterogeneity of the work-in-process. (That is, with a greater number of parts in the system, the range of part types in the system increases.) *The result is to occasionally alter the order in which WIP is put onto machines (relative to lower WIP levels), and this is evidently counterproductive in these cases.*

It is interesting to speculate why maximum overall machine utilization is achieved by FCFS but not by SPT at a WIP level of 7 with 4 or 5 AGVs and 2 or 3 slack buffers. By giving higher priority to part types with shorter processing times, the tendency of SPT is to get such parts through the system faster (on average) than FCFS does. *This changes the average work-in-process mix for SPT vs. FCFS, and this is apparently counterproductive for machine utilization.*

WIP LEVEL	SLACK BUFFERS	NUMBER OF AGVs: 1	2	3	4	5
5	0	.735	.763	.784	.781	.781
5	1	.753	.787	.799	.799	.799
5	2	.751	.800	.800	.800	.800
5	3	.751	.800	.800	.800	.800
6	0	.753	.785	.847	.847	.822
6	1	.764	.858	.877	.877	.877
6	2	.787	.865	.877	.877	.877
6	3	.787	.865	.877	.877	.877
7	0	.753	.785	.847	.847	.822
7	1	.764	.858	.892	.847	.876
7	2	.806	.869	.884	.885	.885
7	3	.781	.869	.884	.885	.885
8	0	.753	.785	.847	.847	.882
8	1	.764	.858	.883	.869	.877
8	2	.781	.856	.881	.892	.892
8	3	.781	.856	.881	.892	.892
9	0	.753	.785	.847	.784	.882
9	1	.764	.858	.892	.879	.877
9	2	.781	.855	.886	.892	.892
9	3	.781	.855	.886	.892	.892
10	0	.753	.785	.847	.802	.822
10	1	.764	.858	.867	.869	.877
10	2	.781	.867	.884	.892	.892
10	3	.781	.867	.884	.892	.892

LEGEND: 0 | 1 / 2 | 3 SLACK BUFFERS

Table 3: Overall Machine Utilizations with SPT as the Dispatching Rule

Although there are evident trends in Tables 2 and 3, these are difficult to quantify in general, or even to rank in general. For example, when overall machine utilization is short of maximum, does increasing the quantity of WIP by 1 or the number of AGVs by 1 have the more beneficial effect, other things being equal? The answer depends on the operating conditions assumed. In Table 3, for example, assuming a WIP level of 5, 2 AGVs, and 0 slack buffers, adding 1 to the quantity of WIP increases overall utilization from 0.757 to 0.769 (an improvement of only 0.012), whereas adding an AGV increases overall utilization from 0.757 to 0.781 (an improvement of 0.024). *In this case, adding an AGV is more beneficial.*

(The differences in overall machine utilizations reported here, although small, are statistically significant because the standard deviation of overall machine utilization is small, as reported above.)

On the other hand, when the WIP level is 5 and there are 2 AGVs and 1 slack buffer, adding 1 to the WIP level increases overall utilization from 0.787 to 0.852 (an improvement of 0.065), whereas adding an AGV increases overall utilization from 0.787 to 0.793 (an improvement of only 0.006). *In this case, adding 1 to the WIP level is more beneficial.*

Furthermore, the effect of increasing the WIP level, or of adding an AGV or a slack buffer, *might be detrimental* rather than beneficial. This phenomenon has been discussed above for several situations in Table 3, and is also evident in Table 2. For example, consider the Table 2 case of 2 AGVs, a WIP level of 7, and 1 slack buffer. Adding 1 to the WIP level causes overall machine utilization to drop from 0.864 to 0.829. Adding an AGV causes utilization to drop, too, from 0.864 to 0.847. Machine utilization clearly is a complicated function of the variables on which it depends.

9. CONCLUSIONS

Conditioned on the fact that our findings are specific to one problem and to one model synthesized from that problem, we draw these tentative conclusions:

1. Overall machine utilization depends on many of the factors ignored in the aggregate mathematical programming solution for the machine utilization problem. In particular, machine utilizations depends significantly on the quantity of work-in-process, the number of slack buffers, the level of resources used to transfer work-in-process, and the rule used to dispatch WIP to machines.

2. After transfer time has been taken into account, theoretical machine utilizations can be achieved under relatively realistic FMS operating conditions.

3. Machine utilizations are a complicated function of WIP level, slack buffers, and transportation resources. Machine utilizations may increase with increases in these resources up to a certain point, and may then decrease as the level of these resources is further increased.

4. When machine utilizations are below the theoretical maximum, it cannot in general be stated whether changing the quantity of WIP, or the number of slack buffers, or the level of transportation resources, will bring about the greatest change in machine utilizations. (That is, the ranking of the gradients associated with these factors depends on the conditions of FMS operation.)

10. FUTURE RESEARCH

Here are several examples of aspects of this work which require further study:

1. The value of letting machines have one or more of their own input and/or output buffers needs to be determined.

2. The influence of alternative part input sequences on machine utilizations needs to be studied. If an important influence is found, guidelines need to be developed for determining good part input sequences.

3. Mathematical programming often produces multiple solutions to the machine utilization problem. Work must be done to determine whether one (or more) of these solutions is better than the others in the sense that by using it (or them), overall machine utilization can be maximized with simpler sets of secondary FMS resources than would otherwise be required.

4. For a given FMS, solutions to a variety of problems of the Section 2 type need to be experimented with to determine whether the level of secondary resources needed to maximum overall machine utilization depends only (or mostly) on the FMS itself, or depends strongly on the particular problem as well (or instead). (Generalizations can be more easily made eventually if the secondary resources needed to maximize machine utilizations depend more on the FMS itself than on the particular problem imposed on the FMS.)

5. The work called for in (1) through (4) needs to be extended to a variety of other types of FMSs.

REFERENCES

French, Simon (1982). *Sequencing and Scheduling*. Ellis Harwood Ltd., Chichester, England.

Henriksen, J. O. and Crain, R. C. (1983). *GPSS/H User's Manual*. Wolverine Software Corporation, Annandale Virginia.

Schriber, T. J. and Stecke, K. E. (1986). Machine utilizations achieved using balanced FMS production ratios in a simulated setting. Working Paper No. 486, Graduate School of Business Administration, The University of Michigan, Ann Arbor, Michigan.

Stecke, K. E. (1985). Procedures to determine both appropriate production ratios and minimum inventory requirements to maintain these ratios in flexible manufacturing systems. Working Paper No. 448, Graduate School of Business Administration, The University of Michigan, Ann Arbor, Michigan.

Stecke, K. E. and Kim, I (1986). A flexible approach to implementing the short-term FMS planning function. In: *Flexible Manufacturing Systems: Operations Research Models and Applications* (K.E. Stecke and R. Suri, eds.). Elsevier Science Publishers B.V., Amsterdam, 283-295.

AUTHORS' BIOGRAPHIES

THOMAS J. SCHRIBER is a professor of Computer and Information Systems in the Graduate School of Business at The University of Michigan. He holds a B.S. (University of Notre Dame) and M.S.E., M.A. (Mathematics) and Ph.D. degrees (all from the University of Michigan). Professor Schriber teaches, does research, and consults in the area of discrete-event simulation. He has published several dozen articles, and has authored or edited nine books. He is a member of ACM, DSI, SCS, SME/CASA, and TIMS.

Thomas J. Schriber
Graduate School of Business
The University of Michigan
Ann Arbor MI 48109-1234 U.S.A.
313-764-1398

KATHRYN E. STECKE is an associate professor of Operations Management in the Graduate School of Business at The University of Michigan. Professor Stecke holds a BS (Boston State College) and two MS's and a Ph.D. (Purdue University). Professor Stecke teaches and does research in operations management, with particular emphasis on flexible manufacturing systems. She has authored numerous papers in a wide variety of journals on various aspects of FMS planning and scheduling She was Co-Chairperson (with Rajan Suri) of the First (and Second) ORSA/TIMS Conferences on Flexible Manufactuimg Systems, held at The University of Michigan in 1984 (and 1986). She is the Editor of the *International Journal of Flexible Manufacturing Systems*. Her professional affiliations include DSI, IIE, ORSA, SME/CASA, and TIMS.

Kathryn E. Stecke
Graduate School of Business
The University of Michigan
Ann Arbor MI 48109-1234 U.S.A.
313-763-0485