Proceedings of the 1988 Winter Simulation Conference
M. Abrams, P. Haigh, and J. Comfort (eds.)

Simulation in a CIM environment: Structure for analysis and real-time control

Catherine M. Harmonosky
Department of Industrial and Management
Systems Engineering
Pennsylvania State University
University Park, PA 16802

Dean C. Barrick
JLG Industries, Inc.
JLG Drive
McConnellsburg, PA 17233

ABSTRACT

Simulation has long been recognized as a valuable tool for analyzing manufacturing systems. It is effective for assessing the impact of changing system parameters (e.g. reducing processing time) upon system performance measures, and it can also aid decisions concerning system configuration. At Penn State, simulation is playing an important role in the development of a Computer Integrated Manufacturing Laboratory. Currently, it is being used as an analysis tool studying system design and computer communication issues. Future plans are to use simulation as a real-time scheduling and control tool. Due to this ultimate long-term goal of the simulation model use, the model structure is different than traditional manufacturing applications of simulation. Rather than having events associated with workpiece movement and processing drive the simulation, computer communication events drive the model. This paper first discusses the general considerations involved when applying simulation as an analysis tool and potential real-time control tool in the CIM environment. The paper then discusses these analysis and real-time control issues in detail using the Penn State CIM Lab application as an illustrative example.

GENERAL ANALYSIS ISSUES

The specific objective of analyzing a CIM system may vary among applications. If a new system is being developed, proper system configuration may be the prime objective (Mills and Talavage 1985). If an ongoing system is being analyzed, the effects of modifications (e.g. different processing time, different product mix) may be a key issue (Ben-Arieh 1985).

In either case, one major consideration when analyzing CIM systems is determining the necessary system components and interactions between system components that must be modeled. For example, major interactions occur when parts are routed between work centers with differing production rates. Blocking or starving of work centers and idle time for expensive equipment are possible operational manifestations of system component interaction. Further, processing or routing time variability accentuates the interaction of parts competing for limited equipment and tooling resources. The system performance response to these interactions is usually not intuitively obvious (Harmonosky 1986). To enable analysis of system response, accurate modeling of actual system components and interactions is important at a level of detail appropriate for analysis objectives.

To represent the dynamic interactions in a CIM system, it is necessary to consider the computer communications. Often, communication time to download programs or machining data is a significant contribution to part flowtime. It is also dead, unproductive time for the piece of equipment waiting for the information (Bedworth and Bailey 1987). Further, when different computers attempt to communicate, communication bottlenecks and computer hang-ups may occur. Naturally, it is desirable to identify this potentially system debilitating condition. Therefore, the communication structure and its associated interactions affecting operations should be included in the simulation for proper and complete analysis. In analysis of traditional manufacturing environments, this would not be necessary.

Another important consideration for analysis is specifying appropriate performance measures to allow a basis for analysis and comparison. Because these systems are usually large capital investments, one common objective used to measure performance is maximizing the system component utilization. A related performance objective may be maximizing throughput, which is related to part flowtime. To accurately assess system performance measured by any objective, the level of detail in the model must match the desired analysis level and the model must incorporate the salient elements of system behavior.

If trying to study short-term CIM system behavior and make short-term operational recommendations, a different type of analysis is required. One common complaint in industry is the inability to rapidly respond to periodic disruptions in the manufacturing process (e.g. breakdown, high priority order, etc.). It would be desirable to have a decision-making tool for this situation to allow more informed decisions that consider the long-term impact upon system operations of the real-time decisions. A simulation model which emulates an existing operation and interfaces with real-time system data may provide this capability (Erickson, Vandenberge, and Miles 1987, Sadowski 1985, Wu 1987). The simulation would access actual system status information to allow running the simulation in real-time, constantly providing a true snapshot of the operating system. Then, the simulation could provide look-ahead system analysis and evaluation capabilities by running the simulation ahead a specified length of time based on the starting conditions of the current system status.

There are many operational considerations for running real-time simulation. The task of obtaining system status data may not be easy due to computer protocol problems. The length of time to run the simulation ahead must be determined such that output information will provide accurate insights, yet run lengths are short enough to keep decisions "real-time". Also, the production system usually remains operating while the simulation is in the decision-making tool mode creating two difficulties. First, some mechanism must keep track of shop-floor activity while the computer is in the simulation decision-making mode (Erickson, Vandenberge, and Miles 1987). Second, when it is returned to the real-time system monitoring mode, the simulation must again be reinitialized to current shop status.

The next section describes an application of simulation in a CIM laboratory setting. The issues and considerations discussed above will be discussed in more detail in the context of the application.

ILLUSTRATIVE EXAMPLE

Defining System Boundary

The first task necessary for a simulation analysis of a CIM system is defining the environment boundary. For this example, the Computer Integrated Manufacturing Lab in Penn State's IE Department provides the test environment. It is a full scale Flexible Manufacturing System (FMS). The system controller is a PDP 11/44 which communicates to a variety of process hardware via a hierarchical communication network. The FMS consists of a Pratt & Whitney Horizon V machining center, a Daewoo Puma 6 turning center, a Fanuc M1L robot, and an IBM 7545 robot all connected by a Cartrac material handling system that conveys raw stock and finished product to/from the system. The Fanuc M1L robot provides machine load and unload capabilities between both the Puma 6 and Horizon V machines and the Cartrac. Some finished part types from the Puma 6 and Horizon V are conveyed by the Cartrac to the IBM 7545 for assembly, after which they are removed from the system by the Cartrac. Figure 1 illustrates the system.

The PDP 11/44 system controller communicates with the Cartrac for constant track statusing, and it communicates with IBM PC's that directly control individual system components. Each piece of equipment has a dedicated PC. The individual PC's store part programs for downloading to the machines. A design PC is linked to the PDP and is used to create orders for release into the system. Figure 2 represents the computer hierarchy.

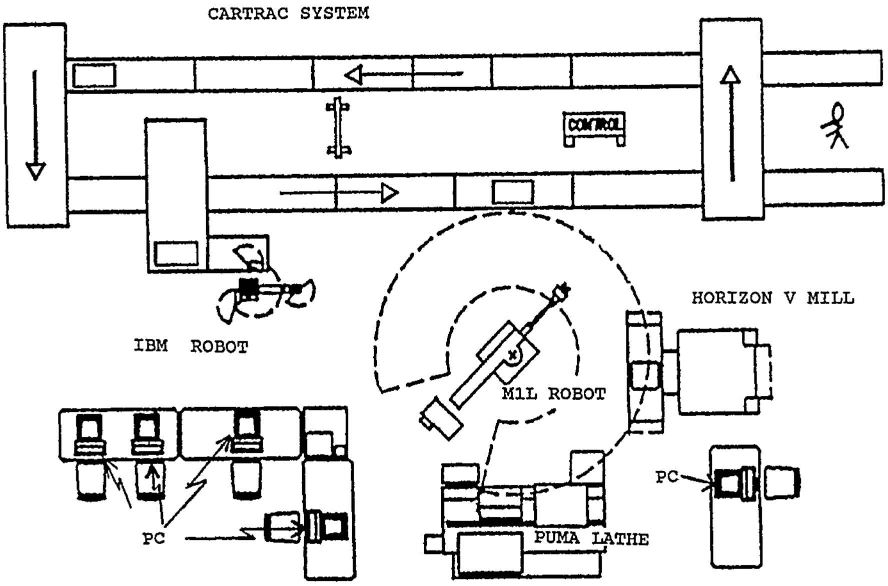

Figure 1. Example CIM Laboratory layout.

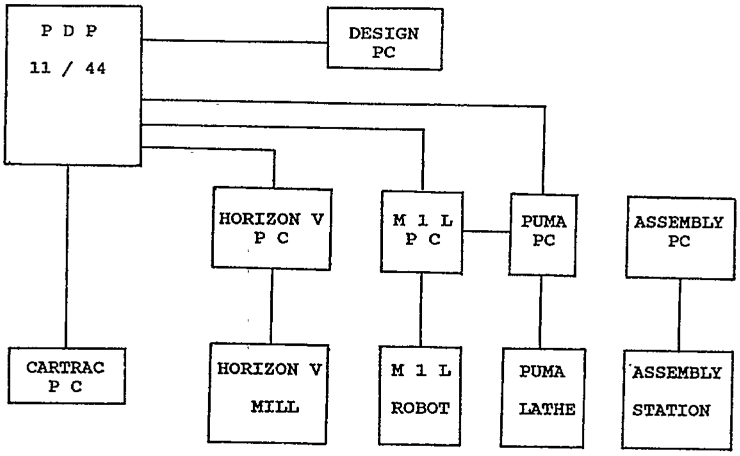

Figure 2: CIM computer communication hierarchy

Simulation Model Structure

When developing the simulation model for the CIM Lab, both near-term and long-term objectives were considered. The near-term goal was to study the proposed system computer communication hierarchy and planned production flow operation. In the long-term, the simulation is to be linked to real-time system status information and act as a real-time decision tool for sequencing and scheduling. Therefore, the simulation was developed at a level of detail that would support these two goals.

Consequently, the model of the laboratory took a slightly different approach than typical manufacturing applications. In typical simulation models of manufacturing environments, the model represents the problem from a traditional viewpoint by considering individual workpieces that move between operations. This CIM Lab model assumes a different system perspective. Because the key to integration in the CIM environment is the computer communication hierarchy, this model focuses upon signaling logic among the controlling computers. The logic used in the actual system communication signal paths which trigger system activities (e.g. cutting on the lathe, retrieving material from the cart, etc.) is the same logic used to control the simulation. The simulation emulates communication signals which trigger activities, making it useful for critiquing the communication logic employed in the system. This type of structure also meshes well with the long-term goal of eventually linking into real-time status information, which will be discussed later. The performance measures used as an indicator of system operations are part flow times and equipment utilizations. The model was developed in the SIMAN simulation language.

Considering the basic principle of this signal-triggering perspective, this structure was accomplished using a series of detached queues. Entities representing both physical parts and requests for service are placed in various queues. Placement of entities in the queues typically sends a signal to trigger some event. The entities are subsequently released from the queues by signals generated by other system activities. The delay times experienced in the queues represents system delays due to computer communication transmissions, processing times, and waiting for system equipment availability.

For example, consider a part being placed on the lathe for processing, being delayed by the processing time, then being removed by the material handling robot. In a typical simulation, the part would be placed in a queue to wait until the lathe resource is available. When it is available, it seizes the resource and holds onto it for a delay equal to the length of the processing time. After the processing time, it releases the lathe resource and is placed in another queue to wait for availability of the material handling robot resource. When the robot is available, it will seize the robot resource and release the lathe resource.

Now, consider describing this same sequence of events in the developed model from the computer communications viewpoint. This model places requests for service in a queue and sends a signal to the model section emulating the lathe controlling computer. Thus the part entity does not seize the lathe resource. Rather it signals the lathe controlling computer that it is requesting to be admitted to the lathe, and the part entity waits in a queue, Queue A, until it is signaled by the lathe controlling computer that it may begin processing. At that point, the part entity is again placed in a queue, Queue B, to wait for a signal that its processing time has been completed, effectively delaying the entity by processing time. When that signal arrives, the entity is released from Queue B, it sends a signal to the robot controlling computer requesting the robot to remove it from the lathe, and the entity is placed in another queue, Queue C. The entity is released from this Queue C when a signal is received indicating the robot is available for servicing. The entity is then placed in a queue, Queue D, where it is delayed until a signal is received that the robot has completed handling the part, (i.e. delayed by the handling time). In other words, the part is delayed in different queues for communication delays and for processing or handling delays, without actually seizing resources, and the signals trigger movement. Therefore, it emulates the real communication triggering scheme very well.

Utilization of individual system components may be determined using a 0,1 (non-working,working) flagging scheme to accurately reflect productive time. For example, the variable associated with lathe utilization will be at 1 only during a cutting time delay. This avoids the problem sometimes encountered when seizing resources in a simulation if the lathe resource is seized to assure being able to load the part, then a delay occurs for the loading time. In this case, the resource utilization statistic will be artificially high because it indicates the lathe being "busy" during the loading time plus processing time plus any delay after cutting stops and prior to actual unloading. Therefore, some time included in the "lathe busy" statistic is actually idle time for the lathe. In terms of evaluating statistics, nothing is lost by collecting statistics using the flagging idea and it can actually give a much more accurate picture of the system activity regarding utilizations. Also, by structuring the model in the queue-until-signaled approach, information concerning waiting times for separate activities (e.g. communication time, unavailable equipment, etc.) is readily available by studying different queue time statistics, which are easily obtained in a simulation.

Another advantage from an analysis perspective, for this signaling model structure is the ability to critique the communication logic employed in the system. Consideration of the actual analysis will be considered in more detail later. This perspective of driving the model was accomplished by using the WAIT-SIGNAL combination in SIMAN.

This structure is particularly useful for this lab application because of the long-term objective, accessing real-time system status information and feeding it into the simulation. This will enable the simulation to be used as a real-time decision tool for intelligent sequencing and scheduling.

Constructing the simulation model as described above, with the communications logic specifically modeled, creates the framework for this interface. The simulation logic will then be controlled by the actual system communication signals dictating start and stop of robot movement, equipment processing, and cart movement. Therefore, the simulation will always reflect the current system status. When a system production control decision is needed, several simulation runs may be executed for a set period of time under different control decision options. The future impact upon the system due to different decisions may be evaluated by analyzing simulation results. In this mode, the simulation model is used as a production control tool with look-ahead system assessment capabilities. Figure 3 illustrates this scenario.

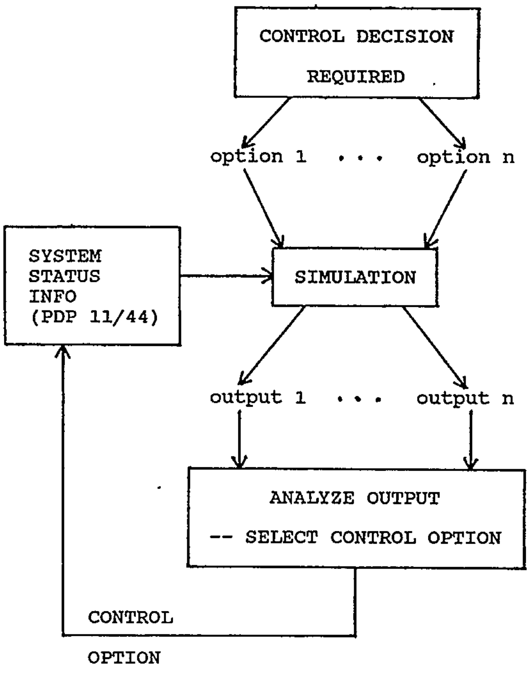

Figure 3. Real-time decision-making mode.

Before discussing the analysis aspects of the application, mention should be made of the use of another helpful analysis tool associated with simulation--that is animation. Simulation model animation has developed rapidly over the past few years and has become a helpful analysis tool in its own right. By graphically representing the activities of entities, resources, transportation mechanisms, etc., that occur within the mathematical simulation model, the user can effectively "view" a system operating according to simulation model specifications and controls. The movement of parts through the system, queue length variation, and blocking or starving of work stations are displayed according to what is happening during the running of the simulation. As a result, this provides insight into dynamic interactions among system components which are responsible for long-term statistical information obtained at the end of a simulation run. It complements the statistical analysis to provide some detailed understanding of model operation.

One primary use of animation is in model verification. The animation provides a visual means to verify that a model correctly represents the real system. By graphically monitoring the movement of entities through a model, subtle logic errors may be quickly identified which might be difficult to detect with standard output analysis. This can be particularly useful with CIM systems which tend to have quite complex system interactions and routing logic. Animation was found to be very helpful for the application discussed in this paper for identifying a subtle simulation model problem caused be a very infrequent (nonetheless present) system anomaly due to a particular combination of system component conditions. The cause of this problem would have been very difficult to pinpoint using traditional simulation output analysis. The animation allowed rapid identification of when and why the error occurred so proper model modification was expedited.

Another prime use of animation is, naturally, explaining the operation of the model to people unfamiliar with simulation. This is particularly helpful in the academic environment when trying to convey simulation concepts to students or visitors from industry. Animations should be developed to convey the appropriate information for the particular application. The animation of the Lab showed the communications logic and showed when different computers were communicating in addition to the machines and material handling. This vividly indicated why delays were occurring (i.e. machines idle due to communication) and specifically the types of communication that take the longest time.

Analysis Results for Illustrative Example

Computer communication is an important issue in a CIM system since the system depends on computer control. This, of course, is not true of more traditional systems. The affect of communication logic and communication times is not always intuitively obvious. Therefore, insight of the interaction of computer communication with the rest of the system is needed before effective changes to communications can be made.

To gain insight into communication activity, the effect of reducing communication times on the CIM Laboratory was analyzed. Three possible changes to the system were compared along with the original system. The three alternatives were defined after conferring with the computer communication designer to outline reasonable system modifications. All the alternatives concentrate on reducing communication delay times. The alternatives were defined as follows:

Alternative 1 - original system
Alternative 2 - all communication times reduced to 1/2 of the original times
Alternative 3 - all communication times reduced to 1/10 of the original times
Alternative 4 - only material handling robot instruction times reduced to 1/10 of the original times

All communication time reductions could be accomplished by upgrading hardware or using different communication protocols. Material handling instruction times could be reduced by knowing the next material handling task before the actual request is made. Time would be saved by having the M1L move to its next destination during the idle time before the request is actually made. For example, in the original system, an NC part program is downloaded to the Puma. The M1L will be requested to retrieve raw material only after downloading is complete. In Alternative 4, the M1L would be requested to retrieve raw material while the NC part program is being downloaded.

The three alternatives and original system were compared by a variety of measures. These measures include part flowtimes, part throughput, machine utilizations, and the time a part or request was delayed to wait for a material handling cart or to wait for the M1L material handling robot. Data for analysis was gathered by running the simulation model for 1440 minutes (the equivalent of three eight hour shifts) and making 5 replications. The simulation output is summarized in Tables 1 through 4.

From the output, it is clear that Alternative 3 (reducing all times to 1/10 of original) resulted in the lowest part flowtimes, and thus produced the most finished parts (191). Computing the amount of productive time (machining time) as a percentage of total busy time for the Horizon V, Alternative 3 was the highest at 66% (.22/.33) compared to 51%, 55%, 51% for Alternatives 1, 2, and 4, respectively.

Alternatives 2 and 4 resulted in very similar performance measures with one exception. Alternative 4 (reducing material handling instruction times to 1/10 of original) resulted in the lowest waiting time for the M1L to become available (0.7 minutes) while Alternative 2 (reducing all times to 1/2 of original) resulted in the highest waiting time for the M1L (1.21 minutes). The fact that Alternatives 2 and 4 resulted in such similar performance measures was not intuitively obvious and clearly exhibits the need to include computer communications in the simulation model.

TABLE 1: Part type average flowtimes (minutes)

ALTERNATIVE #	1	2	3	4	RANKING
FLOWTIME PART 1	10.20	7.29	7.08	7.20	3, 4, 2, 1
FLOWTIME PART 2	14.71	12.27	10.88	12.20	3, 4, 2, 1
FLOWTIME PART 3	14.88	13.83	11.99	13.74	3, 4, 2, 1
FLOWTIME ASSEMBLY	28.00	22.90	21.50	22.48	3, 4, 2, 1

TABLE 2: Number of parts through system in 1440 minutes

ALTERNATIVE #	1	2	3	4	RANKING
TOTAL # PARTS	143	178	191	184	3, 4, 2, 1

TABLE 3: Average machine utiliations (%)

ALTERNATIVE #	1	2	3	4	RANKING
PUMA BUSY	100	100	100	100	ALL TIE
PUMA MACHINING	24	30	32	31	3, 4, 2, 1
HORIZ5 BUSY	33	38	33	41	4, 2, (1,3)
HORIZ5 MACHINING	17	21	22	21	3, (2,4), 1
ASSEMBLY BUSY	2	3	3	3	(2, 3, 4), 1
M1L BUSY	61	53	46	53	1, (2,4), 3

TABLE 4: Average material handling wait times (minutes)

ALTERNATIVE #	1	2	3	4	RANKING
WAIT FOR CART	0.50	0.27	0.40	0.36	2, 4, 3, 1
WAIT FOR IDLE M1L	1.00	1.21	0.91	0.70	4, 3, 1, 2

NOTES:
Parts 1 & 2 are assembled.
(#,#) means tied in ranking.
Puma utilization is 100% because there is an infinite queue of parts waiting for the Puma.

An economic analysis of Alternatives 2, 3, and 4 would be needed to choose one as the most economical. Even though Alternative 3 resulted in the best performance, the cost of attaining a reduction in all communication times to 1/10 of the original may be prohibitive. To enable such a reduction, major network modifications would be required, in addition to significant personnel time and effort for installation. When compared to other alternatives requiring less hardware changes and less implementation time, the increased cost may be difficult to justify for marginal performance gain.

Also worth noting is the difference in utilization measures between "busy" and "machining." The "busy" utilization index measures the time that a machine is machining and communicating or waiting for servicing. The "machining" utilization index measures only the time it is machining. A traditional simulation structure modeling the machine as a resource would yield a utilization index equivalent to the "busy" index. From the output it is easy to see the need to differentiate between the two utilization indexes. For instance, the "machining" utilization of the Horizon V remained constant under Alternatives 2, 3, and 4 with values of 21%, 22%, and 21%, respectively. The "busy" utilization of the Horizon V varied under Alternatives 2, 3, and 4 with values of 38%, 33%, and 41%, respectively. In all cases the "busy" utilization was greater than the "machining" utilization. In fact, the Puma was "busy" 100% of the time while it was actually "machining" only 32% of the time. The difference in utilization measures indicates the usefulness and correctness of this unique simulation model structure. Using a traditional structure (seizing a resource), could lead to misleading utilization measures.

Another important issue to examine is that of material handling. In the CIM lab at Penn State, material is conveyed by a Cartrac system. Of interest is the number of carts to be in the system. If too many carts are used then the throughput of the system will decrease because of unneeded carts causing bottlenecks along the Cartrac. If too few carts are used then the throughput of the system will decrease because of starving the machines of raw material and prolonged waiting time to unload finished parts.

To examine the effect of differing the number of carts used, the simulation was run under original conditions except for the number of carts. The number of carts was varied using 2, 3, 4, and 5 carts. The simulation output is summarized in Tables 5 through 8. It is easy to see that the throughput of the system was not significantly affected by using 3, 4, or 5 carts. When 2 carts were used the

TABLE 5: Part type average flowtimes (minutes)

# OF CARTS	2	3	4	5
FLOWTIME PART 1	10.11	10.17	10.20	10.20
FLOWTIME PART 2	15.72	14.67	14.63	14.71
FLOWTIME PART 3	15.39	14.74	14.91	14.88
FLOWTIME ASSEMBLY	28.21	27.25	27.84	28.00

TABLE 6: Number of parts through system in 1440 minutes

# OF CARTS	2	3	4	5
TOTAL # PARTS	139	144	144	143

TABLE 7: Average machine utilizations (%)

# OF CARTS	2	3	4	5
PUMA BUSY	100	100	100	100
PUMA MACHINING	23	24	24	24
HORIZ5 BUSY	33	33	32	33
HORIZ5 MACHINING	16	17	17	17
ASSEMBLY BUSY	2	2	3	2
M1L BUSY	62	61	60	61

TABLE 8: Average material handling wait times (minutes)

# OF CARTS	2	3	4	5
WAIT FOR CART	0.69	0.45	0.49	0.50
WAIT FOR IDLE M1L	1.10	1.02	1.01	1.00

NOTES:
Parts 1 & 2 are assembled.
Puma utilization is 100% because there is an infinite queue of parts waiting for the Puma.

throughput of the system decreased slightly from 144 to 139 total parts. Also, the waiting times for the carts increased from 0.5 minutes to 0.69 minutes per request. In this application, the rate of cart movement was sufficiently greater than processing times to require only 2 carts to effectively serve the system. Using either 3, 4, or 5 carts did not significantly improve system performance in this system. This illustrates a practice of designing unnecessary material handling capacity in an automated system, which is often the case when a system is hastily designed or material handling is considered as an afterthought.

Through simulation analysis it is clear that reducing communication times will improve the performance of the CIM lab at Penn State. The analysis of Alternatives 2 and 4 clearly shows the need to include computer communications in the simulation model because the performance of differing systems can not be predicted intuitively. The "busy" and "machining" utilization performance measures show the appropriateness of using the signal-wait model structure to model a CIM system as opposed to a more traditional approach of seizing a resource.

SUMMARY

Using simulation as an analysis tool in a CIM environment, requires specific consideration of the computer communication structure. When computer communications controls and delays are accurately modeled, the insight gained may significantly guide future communication network modifications or upgrades. The example presented in this paper illustrated the advantage of the simulation insight. Analysis indicated that a simple modification to the system yielded system performance improvements equal to the improvements of a much more difficult modification.

The use of simulation as a real-time decision-making tool for system control is appealing. However, as previously discussed, many operational issues remain to be solved, including obtaining system status data, simulation run times, and continued system monitoring during the decision-making mode. The simulation construction described in the illustrative example, utilizing the signal-triggering concept, shows promise of being able to easily interface with the actual system status when that data eventually becomes available.

REFERENCES

Bedworth, D.D. and Bailey, J.E. (1987). Integrated Produciton Control Systems, John Wiley & Sons, Inc., New York, NY.

Ben-Arieh, D. (1985). "A Knowledge Based System for Simulation and Control of a CIM," Proceedings of the First International Conference on Simulation in Manufacturing, Stratford-upon-Avon, UD, March.

Erickson, C., Vandenberge, A., Miles, T. (1987). "Simulation, Animation, and Shop-Floor Control", Proceedings of the 1987 Winter Simulation Conference, Atlanta, GA, December.

Harmonosky, C.M. (1986). "Integrating Automated Systems Control and Design," CIM Review, Vol. 2, No. 3, Spring.

Mills, R. I. and Talavage, J.J. (1985). "Simulation Programs for FMS Design," Proceedings of the First International Conference on Simulation in Manufacturing, Stratford-upon-Avon, UD, March.

Sadowski, R.P. (1985). "Improving Automated Systems Scheduling," CIM Review, Vol. 2, No. 1, Fall.

Wu, S.D. (1987). "An Expert System Approach for the Control and Scheduling of Flexible Manufacturing Cells," Unpublished PhD Thesis, Pennsylvania State University, University Park, PA.

AUTHOR'S BIOGRAPHIES

CATHERINE M. HARMONOSKY is an Assistant Professor in the Department of Industrial and Management Systems Engineering at The Pennsylvania State University. She 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. Dr. Harmonosky's research interests are manufacturing and production systems analysis, scheduling, and computer integrated manufacturing. Her special interests include studying the impact of system design decisions upon production control strategies and innovative computer simulation applications in systems analysis and scheduling. She is an active member of IIE, SME, Tau Beta Pi, NSPE, and SWE.

Catherine M. Harmonosky
The Pennsylvania State University
Department of Industrial and
Management Systems Engineering
207 Hammond Building
University Park, PA 16802
(814) 865-2107

DEAN C. BARRICK has a B.S.I.E. and M.S.I.E. degree from the Department of Industrial and Management Systems Engineering at The Pennsylvania State University. His interests are in the areas of manufacturing and production control, quality control, and simulation. Mr. Barrick is working as a manufacturing engineer with JLG Industries in McConnellsburg, PA. He is a member of IIE, ASQC, and SME.

Dean C. Barrick
Manufacturing Engineer
JLG Industries, Inc.
JLG Drive
McConnellsburg, PA 17233
(717) 485-5161