

ADVANCED TUTORIAL - SIMULATION-BASED SCHEDULING AND CONTROL

Brett A. Peters
Jeffrey S. Smith
James Curry
Cynthia LaJimodiere

Department of Industrial Engineering
Texas A&M University
College Station, Texas 77843, U.S.A.

Glenn R. Drake

Systems Modeling Corporation
504 Beaver Street
Sewickley, Pennsylvania 15143, U.S.A.

ABSTRACT

This paper presents a simulation control system developed by the Texas A&M Computer Aided Manufacturing Laboratory (TAMCAM). This control system is a research tool that TAMCAM uses to explore the advantages and disadvantages of on-line simulation for process control. The simulation control system is developed directly from information about the shop floor stored in a relational database. Processing times and fallout rates are also estimated directly from external data sources. The system is implemented in Arena, Microsoft Access, and Microsoft Visual C++. The advanced tutorial will demonstrate the software tools described in this paper.

1 INTRODUCTION

Simulation is commonly used to gain insight into manufacturing systems. Simulation is based on a logical model of how a series of processes interact combined with statistical and deterministic information about the individual processes. Historically, simulations are developed off-line using custom software packages/languages with limited direct connections to the actual data generated by the production system (Drake and Smith 1996). Traditional simulation generally examines long-term system performance for planning and design purposes. These models are usually "throw away models" because they are seldom used after the initial plans or designs of a project are finalized

(Thompson 1994). Primary reasons for this inflexibility are that the input data of the simulation are gathered and analyzed outside the simulation environment and that the simulation system cannot communicate automatically with the Manufacturing Execution System (MES) that is responsible for gathering shop floor status information and executing tasks.

On-line simulation integrates the information system with the simulation model. On-line simulation for process scheduling, real-time "intelligent" shop floor control, performance forecasting, process capability estimation, real-time control systems emulation, real-time displays of system status, and short term decision making is an active area of research (See, for example, Erickson et al. 1987, Wu and Wysk 1989, Harmonosky and Robohn 1991, Rogers and Flanagan 1991, Moser and Engell, 1992, Smith et al. 1994, Drake and Smith. 1995, Jones et al. 1995, Tunali 1995, and Grabot et al. 1996). By using the most current system information, accurate predictions about the system and future control alternatives can be developed. The reason for using simulation is that simulation can often capture and describe the complex interactions within a particular FMS where analytical methods fail (Erickson et al. 1987, Wu and Wysk 1989).

An important element to make on-line simulation effective is the ability to do retrospective studies. To model the past, the TAMCAM Simulation Control System (TSCS) writes the transitions between system states to a temporal database for later analysis. Queries

on the database provide estimates of process times, rework, and fallout rates. These estimates for the distribution of the time required to do a task drive the simulation model as inputs in look-ahead mode. This paper describes the structure of the TSCS and the associated advanced tutorial demonstrates the prototype software tools and describes our limited implementation experience.

2 ENVIRONMENT

The TAMCAM system is based on a three level control system containing a manufacturing information system, a real-time simulation model operating as a Manufacturing Execution System (MES), and execution level controllers for automated and manual equipment (Figure 1). The highest level is a manufacturing information system usually a MRP system that contains information about inventory level and production schedules. This system drives a simulation controller that handles the actual shop floor control. The semantics of the manufacturing information system are enforced at the control and simulation levels.

The simulation level acting as a MES uses a discrete event simulation for decision making (Smith et al. 1994). In addition to decision making, on-line simulation, as proposed by Drake and Smith (1995, 1996), uses a real time link between a simulation model and the production system as a direct method of process control. The TSCS simulation is generated directly from the manufacturing information system. This direct generation from a relational database structure, using Simon or Arena, is similar to the methodology proposed by Centeno and Standridge (1991). This direct link insures consistency between the two levels of the process.

The simulation model can be run in a real-time control mode or in a look-ahead simulation mode to test proposed schedules. In real-time mode, the simulation prompts the client for a task using the task initiation queue (TIQ). After the task is complete, the client returns the length of time required for the task and the outcome of the task using the task completion queue (TCQ). In look-ahead mode, the real-time delay blocks are replaced with statistical distributions for the time required in the delay block. The commercial versions of the Arena simulation package now include the capability for directly implementing real-time control.

The lowest level contains operational clients. The clients of the shop floor controller include automated shop floor equipment or Human Machine Interfaces (HMI) for processes that are not completely automated. When messages pass through the router, the content of the messages is written to the system state database with a time stamp. The workstations are based on a message-

based part state graph (MPSG) controller (Smith and Joshi 1995, Smith 1996).

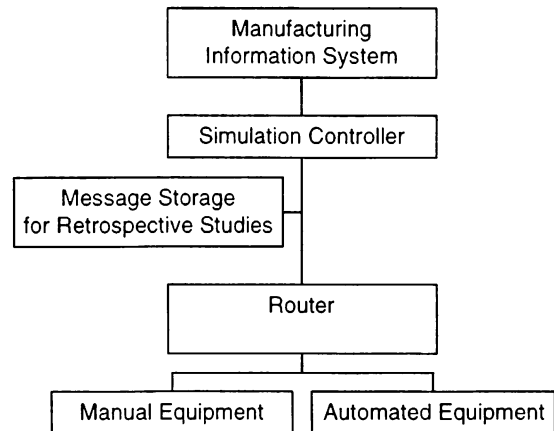


Figure 1: Diagram of the System

3 DEFINITION OF STATES

The key to on-line simulation is determining at what level to model the system. The TSCS model is based on constructing the system state information such that operations only occur on one part where all resources employed in the operation work on the part for the duration of the operation. This restriction avoids estimating process times and fallout rates from processes that are mixed together and facilitates direct estimation of process times from the messages being sent in the system. An example of a mixed process to be avoided is a worker loading a machine and then the worker being released after the completion of the loading task before the completion of the machining process. This process is modeled as two tasks since the number of resources required by the part changes during the operation. If the worker is seized for the entire period of time then the operation would be modeled as one task in the simulation.

All operations that act on multiple parts are modeled in the logic of the simulation. For assembly, the parts are combined in the logic of the simulation before the actual operation occurs. For disassembly, parts in the simulation model are created for the parts disassembled after the operation has occurred. Batching is modeled as an assembly operation followed by the actual operation with a disassembly operation to return the parts to their original status.

The task initiation message contains the operation and the part ID. All tasks are sent only to free resources that have the parts required to complete a task in their part queue or buffers. This restriction avoids including

queuing times of parts waiting for resources or resources waiting for parts in the processing time. Queuing times are excluded from the model because queuing time arises primarily from the scheduling choices developed by the decision maker.

The task completion message contains the outcome of the task and the active processing time required for the task. The unique identifier for the task completion message is the part number and workstation. The basic active processing time includes set up time, down time, and transportation time within the workstation. For particular resources where a more detailed description of the system is required, the active processing time is split into subgroups including down time, reaction time, setup time and activity time for particular sub-tasks (Table 1). These additional attributes are only required if needed by the simulation.

Table 1: Messages Sent within the System

	Initiation Sequence	Completion Messages
Required Information	Part Number, Work Station	Part Number, Workstation, Active Processing Time.
Optional Information	Due Date, Special Instructions, Priority	Setup Time, Reaction Time, Sub-Operation, Processing Times, Success of the Operation, Additional Routing Based on the Operation.

4 INFORMATION SYSTEM

The information system is split into a description of the manufacturing system and a historic record of the on-line simulation.

The description of the manufacturing system contains a list of parts (i.e., part types), a list of specific parts to be produced with due dates, a list of resources for the production process with feasible messages for each resource, routings for the parts, alternative routings for the parts, and any additional information required to develop the simulation. Many of these inputs including the current production schedule, due dates, part routings, and resource relationships are deterministic and can be directly accessed from a relational database using SQL as suggested by Drake (1996). However, several key inputs to the proposed simulation model, including distribution of the process times, fallout rates, and shift schedules, are stochastic and vary for different part types and equipment stations and cannot be directly accessed from a static database.

This information system can access the simulation to run the simulation in look-ahead mode. The look-ahead simulation tool can predict whether a given production schedule is feasible by using the simulation as a query tool. The feasibility can be expressed as either the probability of completing a set of tasks within a given period of time or the entire estimated cumulative density function for completing the task as a function of time. Using this ability, multiple simulations can be used to improve the scheduling of resources by changing the operating rules (Tautuo and Pierreval 1995). The instruction for on-line simulation tools for manufacturing systems should resemble commands commonly used in database systems. A method for achieving this goal is to have the commands as an extension of SQL (Balasubramanian and Tuzhilin 1996). Even though our model does not follow the command set proposed by Balasubramanian and Tuzhilin, the TSCS model tries where possible to use a command structure that is easily understood to database and industrial/manufacturing engineering professionals.

The results of the real-time simulation execution are stored in a database. The real-time results are the data on how the system actually operates. The combination of a MES system with a database system is extremely common with most vendors using Open Database Connectivity (ODBC) software or a similar methodologies to connect controllers to database systems. These systems warehouse this information for many purposes including statistical quality control.

The TSCS database system stores the results of operations including process times and fallout rate. The user can exclude data points from the estimation of processing times for statistical quality control or look-ahead simulation. The primary reasons for excluding data are that the observations are influenced by a problem where the underlying cause is resolved and will not reoccur or that the data is in error.

From the database system, a data analysis tool estimates the distribution for process times and fallout rates at each workstation for each part type. These estimates are stored and used in the simulation model. The default distribution for process times is the gamma distribution. If the sample is not sufficiently large, then the estimation of the process times is based on a grouping of parts with similar characteristics. The estimation procedure identifies extreme values and allows the user to delete these values from the estimation.

The results of the simulated system in look-ahead mode can also be compared to the actual flow time for the system in real time. A difference between the simulated system in real time and the estimated performance in look-ahead mode could reveal problems

in the simulation or the actual system that occur in a series of stations.

5 IMPLEMENTATION ISSUES

The demonstration of the TSCS consists of a simulation-based controller developed in Arena, a message router and client controls developed in Microsoft Visual C++, and an external database system developed in Microsoft Access. The simulation-based controller is built in Arena using the Arena Real-Time template. The Arena model also uses a user-coded dynamic link library (DLL) written in Microsoft Visual C++ to provide the implementation specific communications functions required by the router. All connections within the real-time system are implemented using the TCP/IP protocols. The connections within the forecast system are implemented with Database Access Objects (DAO). The connections between the different software applications are shown in Figure 2 where the solid lines are TCP/IP connections and the dashed lines are DAO connections.

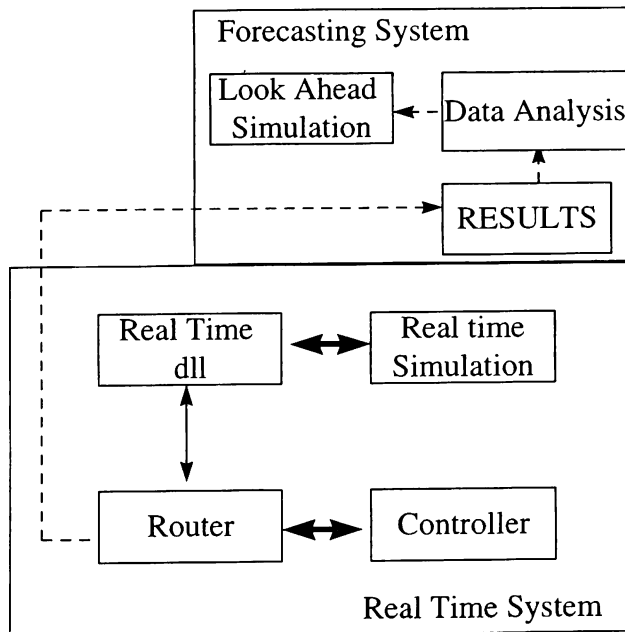


Figure 2: Layout of Software System

The ability to estimate statistical distribution directly from the information system is critical to this model. For implementing real-time simulation in an actual manufacturing environment, the key information challenge is insuring semantic correctness in a distributed database system. For information systems that are

loosely federated together, the accessing of data may include changes to the data structure, attribute naming, and data schema. An example of these differences is extracting processing times from a database system that stores the processing time as an attribute in a table and a system that stores processing times as the difference between start time and completion time in a series of records. The major obstacle to this integration is not the ability to communicate between different data sources that is standardized with ODBC, DAO and embedded SQL; the challenge is identifying the ways information is stored and having a clear understanding of what the information implies about the manufacturing process. For data quality in on-line simulation, the information must be accurate and represent what the model thinks it represents.

For the TSCS control system, these information system design problems were solved by making the information system meet the needs of the simulation model. For a commercial implementation of these concepts, substantial commitment for standardization of system states would be required. Based on the amount of resources spent on the simulation of manufacturing processes compared to relational database systems and MRP systems, the majority of the flexibility must come from the simulation package. This flexibility includes the ability to analyze data that arise from a mixture of processes, the ability to access different databases and a clear set of rules for defining the states of the system.

6 CONCLUSIONS

Combining simulation, information systems, and real-time control is a very promising framework for the study and optimization of the dynamic characteristics of a flexible manufacturing system. Look-ahead simulation is also a promising tool for predicting future lead times and calculating due dates. As flexible and agile manufacturing become prevalent, the ability to describe the short term future performance of a manufacturing system becomes critical. This tutorial session will describe the framework developed for simulation-based control. In addition, it will demonstrate the implementation and use of a simulation control system using Systems Modeling's Arena, Microsoft Visual C++, and Microsoft Access as the core demonstration platform.

7 REFERENCES

- Balasubramanian, P. and Alexander Tuzhilin. 1996. Using query driven simulation for querying outcomes of business processes, *Decision Support Systems*, Vol. 16, 275-295.

- Centeno, M. and C. Standridge. 1991. Modeling manufacturing systems: an information base approach, *Simulation Symposium*.
- Drake, Glenn R. and Jeffrey Smith. 1996. Simulation systems for real time planning scheduling and control, *In the Proceedings of the 1996 Winter Simulation Conference*.
- Drake, Glenn R. and Jeffrey Smith. 1995. Simulation as a planning and scheduling tool for flexible manufacturing systems, *In the Proceedings of the 1995 Winter Simulation Conference*, December, Washington DC.
- Erickson, C., Vandenberg, A. and T. Mills. 1987. Simulation, animation, and shop floor control, *In the Proceedings of the 1987 Winter Simulation Conference*, 649-652.
- Grabot, B., Blanc, J., and C. Binda. 1996. A decision support system for production activity control, *Decision Support System*, 87-110.
- Godwin, A. N. and D. Sandars. 1995. Forecast standards and existing databases, *Factory 2000*, IEE Conf. Publ. No. 398.
- Harmononosky, C. M. and S. F. Robohn. 1991. Real-time scheduling in computer integrated manufacturing: A review of recent literature. *International Journal of Computer Integrated Manufacturing*, 331-340.
- Jones, A., Rabelo, L., and Y. Yuehwern. 1995. A hybrid approach for real-time sequencing and scheduling. *International Journal of Computer Integrated Manufacturing*, 145-154.
- Moser, M. and S. Engell. 1992. Avoiding scheduling errors by partial simulation of the suture, *In the Proceedings of the 31st Conference on Decision and Control*, Tucson, Az., December.
- Prabhu, V. and Neil Duffie. 1995. A distributed simulation approach for enabling cooperation between entities in heterarchical manufacturing systems, *SPIE*, vol. 2596, 234-242.
- Rogers, P., and M. Flanagan. 1991. On-line simulation for real-time scheduling of manufacturing systems, *Industrial Engineering*, Vol. 23, 37-40.
- Smith, J. S., Wysk, R. A., Sturrock, D., Ramaswamy, S., Smith, G., and S. Joshi. 1994. Discrete event simulation for shop floor control, *In the Proceedings of the 1994 Winter Simulation Conference*, December, Lake Buena Vista, FL, 962-969.
- Smith, J. and Joshi, S. 1995. A shop floor controller class for computer-integrated manufacturing, *International Journal of Computer Integrated Manufacturing*, Vol. 8, No. 5, 327-339.
- Smith, J. S. 1996. Design and implementation of FMS control software, *In the Proceedings of the 1996 FAIM Conference*, Atlanta, GA, May.
- Smith, J. S. and Joshi, S. B. 1993. Message-based part state graphs (MPSG) for shop floor control, *In the Proceedings of the 2nd IIE Research Conference*, Los Angeles, CA, May 26-28.
- Tautuo, L. and H. Pierreval. 1995. Using evolutionary algorithms and simulation for the optimization of manufacturing systems, *Emerging Technologies and Factory Automation INRIA/ IEEE Symposium*.
- Thompson, M. B. 1994. Expanding simulation beyond planning and design, *Industrial Engineering*, Vol. 26, No. 10, 65-67.
- Tunali, S. 1995. Simulation for evaluating machine and AGV scheduling rules in an FMS environment, *Engineering Management Conference*, 433-438.
- Wu, D. S. and R. A. Wysk. 1989. An application of discrete event simulation to on-line control and scheduling in flexible manufacturing, *International Journal of Production Research*, Vol. 27, 1603-1624.

AUTHOR BIOGRAPHIES

JEFFREY S. SMITH is an assistant professor in the Industrial Engineering Department at Texas A&M University. His research interests are shop floor control, manufacturing systems design, analysis, control, and simulation.

BRETT A. PETERS is an assistant professor in the Industrial Engineering Department at Texas A&M University. His research interests are manufacturing systems design, analysis and control, and simulation. He is an active member of IIE, SME, and INFORMS.

JAMES CURRY is a research assistant in the Industrial Engineering Department at Texas A&M University.

GLENN R. DRAKE is an associate applications engineer for Systems Modeling Corporation. He received his B.S.I.E. in 1994 and his M.S.I.E. in 1996 from Texas A&M University.

CYNTHIA LAJIMODIERE is a research assistant in the Industrial Engineering Department at Texas A&M University.