

SIMULATION OF MANUFACTURING SYSTEMS

Averill M. Law
Michael G. McComas

Averill M. Law & Associates, Inc.
P.O. Box 40996
Tucson, Arizona 85717, U.S.A.

ABSTRACT

This paper discusses how simulation is used to design new manufacturing systems and to improve the performance of existing ones. Topics to be discussed include: manufacturing issues addressed by simulation, simulation software for manufacturing applications, techniques for building valid and credible models, and statistical considerations. A comprehensive example will be given in the conference presentation.

1 INTRODUCTION

One of the largest application areas for simulation modeling is that of manufacturing systems, with the first uses dating back to at least the early 1960's. In this paper we present an overview of the use of simulation in the design and analysis of manufacturing systems. Detailed discussions of simulation, in general, may be found in Banks, Carson, and Nelson (1996) and Law and Kelton (1991). A practical discussion of the steps in a sound simulation study is given in Law and McComas (1990). This paper is a synopsis of a three-day short course with the same title as this paper, which the first author has given more than fifty times since 1985.

2 MANUFACTURING ISSUES ADDRESSED BY SIMULATION

The following are some of the specific issues that simulation is used to address in manufacturing:

The need for and the quantity of equipment and personnel

- Number and type of machines for a particular objective

- Number, type, and physical arrangement of transporters, conveyors, and other support equipment (e.g., pallets and fixtures)
- Location and size of inventory buffers
- Evaluation of a change in product volume or mix
- Evaluation of the effect of a new piece of equipment on an existing manufacturing system
- Evaluation of capital investments
- Labor-requirements planning

Performance evaluation

- Throughput analysis
- Time-in-system analysis
- Bottleneck analysis

Evaluation of operational procedures

- Production scheduling
- Inventory policies
- Control strategies [e.g., for an automated guided vehicle system (AGVS)]
- Reliability analysis (e.g., effect of preventive maintenance)
- Quality-control policies

Following are some of the performance measures commonly estimated by simulation:

- Throughput
- Time in system for parts
- Times parts spend in queues
- Queue sizes
- Timeliness of deliveries
- Utilization of equipment or personnel

3 SIMULATION SOFTWARE FOR MANUFACTURING APPLICATIONS

Most organizations that simulate manufacturing or material-handling systems use a commercial simulation software product, rather than a general-purpose programming language (e.g., C). Furthermore, the two most common criteria for selecting simulation software are *modeling flexibility* (ability to model any system regardless of its complexity or uniqueness) and *ease of use*.

We now define the major types of simulation software for manufacturing. A *simulation language* is a software package that is *general in nature* (in terms of the applications it can address) and where model development is done by “programming.” Traditionally, programming meant the development of a simulation model by writing code, but in recent years there has been a strong movement toward simulation languages that employ a graphical model-building approach. Example of simulation languages are Arena, AweSim!, Extend, GPSS/H, Micro Saint, MODSIM III, SES/*workbench*, SIMPLE++, SIMSCRIPT II.5, SIMUL8, and SLX. The major advantage of a *good* simulation language is modeling flexibility, whereas the major disadvantage is that programming expertise is required.

A *manufacturing-oriented simulation language* is one where the modeling constructs are specifically oriented toward manufacturing or material handling. Examples of such software are AutoMod and Quest. One advantage of this type of software is that programming time may be reduced (compared to a simulation language) due to powerful constructs for such things as conveyors and AGVS.

In the last five to ten years, there has been considerable interest in having simulation software that is easier to use, which largely means reducing the amount of programming required to build a model. This has given rise to what we call a *manufacturing-oriented simulator*, which is a simulation package designed to model a manufacturing system in a specific class of

systems. This type of software has two main characteristics:

- Orientation is toward manufacturing
- Little or no programming is required to build a model (relative to simulation languages)

Examples of simulators are FACTOR/AIM, ProModel, Taylor II, and WITNESS. A simulation model is developed using a simulator by using graphics (e.g., dragging and dropping icons), by selecting items from menus with a mouse, and by filling in dialog boxes. The major advantage of a simulator is that if it is applicable to your problem, then the amount of time required to develop (“program”) the model may be reduced considerably. The major disadvantage of simulators is that they are not as flexible as simulation languages, since they do not allow full-blown programming as in simulation languages (see below for further discussion).

Because a simulator that does not allow programming in any shape or form just cannot be as flexible as a simulation language, the vendors of the major manufacturing-oriented simulators have introduced programming into their software in one or both of the following ways:

- The ability to use “programming-like” constructs (e.g., setting values for attributes or global variables, if-then-else logic, etc.) at certain *selected* points in the model-building process
- The ability to call external routines written in a general-purpose programming language at certain *selected* points in the model-building process

Simulators with either or both of the above programming options are still not, in general, as flexible as a *good* simulation language where anything can be programmed from scratch. For example, manufacturing simulators have such *fundamental* modeling constructs as machines, parts, and conveyors. Since in the real world conveyors can come in a myriad of forms, there is a good chance that none of the built-in conveyor options is completely correct. Furthermore, because of the fundamental nature of the conveyor modeling construct, it may not be possible to change their logic in a substantive manner.

The distinction between simulation languages and simulators has become less clear in recent years. Languages have gone to graphical user interfaces to increase ease of use and simulators have added some programming capabilities to increase modeling

flexibility. However, we can still say that a simulation language is general in nature and uses programming (syntactical or graphical) to develop a model. Simulators, on the other hand, are application specific (for the most part) and, perhaps, at most twenty percent of the model is developed using some form of programming. A much more detailed discussion of the topics in this section is given in Law (1997).

4 DEVELOPING VALID AND CREDIBLE SIMULATION MODELS

A simulation model is a surrogate for actually experimenting with a manufacturing system, which is often infeasible or not cost-effective. Thus, it is important for a simulation analyst to determine whether the simulation model is an accurate representation of the system being studied, i.e., whether the model is *valid*. It is also important for the model to be *credible*; otherwise, the results may never be used in the decision-making process, even if the model is “valid.”

The following are some important ideas/techniques for deciding the appropriate level of model detail (one of the most difficult issues when modeling a complex system), for validating a simulation model, and for developing a model with high credibility:

- State *definitively* the issues to be addressed and the performance measures for evaluating a system design at the beginning of the study.
- Collect information on the system layout and operating procedures based on conversations with the “expert” for each part of the system.
- Delineate all information and data summaries in an “assumptions document,” which becomes the major documentation for the model.
- Interact with the manager on a regular basis to make sure that the correct problem is being solved and to increase model credibility.
- Perform a structured walk-through (before any programming is performed) of the conceptual simulation model as embodied in the assumptions document before an audience of all key project personnel.
- Use sensitivity analyses [see Law and Kelton (1991)] to determine important model factors, which have to be modeled carefully.
- Simulate the existing manufacturing system (if there is one) and compare model performance measures

(e.g., throughput and average time in system) to the comparable measures from the actual system.

5 STATISTICAL ISSUES IN SIMULATING MANUFACTURING SYSTEMS

Since random samples from input probability distributions “drive” a simulation model of a manufacturing system through time, basic simulation output data (e.g., times in system of parts) or an estimated performance measure computed from them (e.g., average time in system from the entire simulation run) are also random. Therefore, it is important to model system randomness correctly and also to design and analyze simulation experiments in a proper manner. These topics are briefly discussed in this section.

5.1 Modeling System Randomness

The following are some sources of randomness in simulated manufacturing systems:

- Arrivals of orders, parts, or raw materials
- Processing, assembly, or inspection times
- Machine times to failure
- Machine repair times
- Loading/unloading times
- Setup times

In general, each source of system randomness needs to be modeled by an *appropriate* probability distribution, not what is perceived to be the mean value. Note that sources of randomness encountered in practice are rarely normally distributed. A detailed discussion of simulation input modeling is given in Chapter 6 of Law and Kelton (1991).

5.2 Design and Analysis of Simulation Experiments

Because of the random nature of simulation input, a simulation run produces a *statistical estimate* of the (true) performance measure not the measure itself. In order for an estimate to be statistically precise (have a small variance) and free of bias, the analyst must specify for each system design of interest appropriate choices for the following:

- Length of each simulation run

- Number of independent simulation runs
- Length of the warmup period, if one is appropriate

We recommend always making at least three to five independent runs for each system design, and using the average of the estimated performance measures from the individual runs as the overall estimate of the performance measure. (Independent runs means using different random numbers for each run, starting each run in the same initial state, and resetting the model's statistical counters back to "zero" at the beginning of each run.) This overall estimate should be more statistically precise than the estimated performance measure from one run. Note that independent runs (as compared to one very long run) are required to obtain legitimate and *simple* variance estimates and confidence intervals.

For most simulation studies of manufacturing systems, we are interested in the long-run (or steady-state) behavior of the system, i.e., its behavior when operating in a "normal" manner. On the other hand, simulations of these kinds of systems generally begin with the system in an empty and idle state. This results in the output data from the beginning of the simulation run not being representative of the desired "normal" behavior of the system. Therefore, simulations are often run for a certain amount of time, the *warmup period*, before the output data are actually used to estimate the desired performance measure. Use of the warmup-period data would bias the estimated performance measure.

A comprehensive treatment of simulation output-data analysis can be found in Chapter 9 of Law and Kelton (1991).

6 SIMULATION ANALYSIS OF A MANUFACTURING SYSTEM

In the actual conference presentation, we will give a detailed analysis of a manufacturing system. We will address the following issues:

- Evaluating different machine and forklift-truck resource levels
- Sizing of work-in-process buffers
- Determining the impact of random machine downtimes
- Determining the effect of different logic for the forklift trucks

REFERENCES

- Banks, J., J. S. Carson, and B. L. Nelson. 1996. *Discrete-event system simulation*. 2d ed. Upper Saddle River, New Jersey: Prentice-Hall.
- Law, A. M. 1997. How to select simulation software. Tucson, Arizona: Averill M. Law & Associates.
- Law, A. M., and W. D. Kelton. 1991. *Simulation modeling and analysis*. 2d ed. New York: McGraw-Hill.
- Law, A. M., and M. G. McComas. Secrets of successful simulation studies. *Industrial Engineering* 22: 47-48, 51-53, 72.

AUTHOR BIOGRAPHIES

AVERILL M. LAW is President of Averill M. Law & Associates, Inc. (Tucson, Arizona), a company specializing in simulation consulting, training, and software. He has been a simulation consultant to more than 100 organizations, including General Motors, IBM, AT&T, General Electric, Nabisco, Xerox, NASA, the Air Force, the Army, and the Navy. He has presented more than 250 simulation short courses in 16 countries, and delivered more than 100 talks on simulation modeling at technical conferences.

He is the author (or coauthor) of three books and more than 35 papers on simulation, manufacturing, communications, operations research, and statistics, including the textbook *Simulation Modeling and Analysis* that is used by more than 50,000 people worldwide. His series of papers on the simulation of manufacturing systems won the 1988 Institute of Industrial Engineers' best publication award. He is the codeveloper of the ExpertFit software package for selecting simulation input probability distributions, and he has developed several simulation videotapes. Dr. Law wrote a regular column on simulation for *Industrial Engineering* magazine from 1990 through 1991.

He has been a tenured faculty member and has taught simulation at the University of Wisconsin and the University of Arizona. Dr. Law has a Ph.D. in industrial engineering and operations research from the University of California at Berkeley.

MICHAEL G. MCCOMAS is Vice President for Consulting Services of Averill M. Law & Associates, Inc. He has considerable simulation modeling experience in such manufacturing industries as food processing, paper products, microcomputers, aerospace materials, medical components, electronic components, pet-care products, and basic metal processing. He is also the coauthor of seven published papers on simulation. His educational background includes an M.S. in systems and industrial engineering from the University of Arizona.