Proceedings of the 1995 Winter Simulation Conference
ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman

DEVELOPING INDUSTRIAL STRENGTH SIMULATION MODELS

Marvin S. Seppanen

Productive Systems
Route Five, Box 46
Winona, Minnesota 55987, U.S.A.

ABSTRACT

This paper is based on over 25 years of experience in the field of discrete event simulation. Over this time span there has been dramatic increases in computer power, the number and the capabilities of simulation languages, and the total number of people using computers in their daily work. However, these changes have yet to make discrete event simulation a standard manufacturing decision support tool. This paper explores some reasons for this short-fall and offers some near-term and long-term solutions.

1 PROBLEM

Over the past decade, with computing power moving from the computer center to the desk top, simulation tools have become easier to use for the beginning modeler. This ease of use is manifested in point and click model building, integrated animation, and graphic results (unfortunately too often in the form of pie charts). These advances have been tailored to enhance software sales to a broadening market of potential users.

While these software advancements have made it easier to develop an initial simulation model, they have done little to address the real decision issues faced by most manufacturing firms. Simulation models are still islands within the computerized database structure of the firm. Inventory, new order, or manufacturing technology changes must be manually incorporated into simulation models, thereby becoming dated and rapidly useless as a decision support tool.

The primary shortcoming of today's discrete event simulation tools is their limited capability to effectively handle data from a variety of sources. In addition these tools do not report their input data in a format familiar to those without extensive simulation modeling experience.

Using current simulation tools to model large scale manufacturing systems can result in the following problems:

- Most real systems are larger than the one to four screens which can be conveniently modeled with graphic interface tools. True, most software packages now offer an infinite drawing window, however the model developer can rapidly becomes lost in the details while panning about the model.
- Most manufacturing firms now have extensive databases of information which typically are required in a comprehensive simulation model. However, it is difficult, if not impossible, to import routing, shift, and downtime information directly from those databases into the simulation model.
- Most simulation tools support the use of variables, attributes, and/or parameters. Some simulation tools still limit the number and naming conventions of these important data elements. Thus, the simulation analyst who is asked to rerun the model with a 10% demand increase is forced to make multiple model changes using the graphical interface. While this may be a routine clerical task, it is often difficult to assure that every required model change was accurately completed.
- Most simulation tools do not provide a concise report of the initial state of the modeled system. It is difficult to assure that all parameters and options have been properly set for the desired experimental run.

Recent simulation language improvements have focused on model development and graphic results while for the most part ignoring the data input problem. The typical simulation tools being sold today are very good for modeling systems involving resources and parts numbering in the 10's but lack the constructs to expand to the level of 100's or 1000's required for many manufacturing applications. Several characteristics distinguish industrial strength simulation models from the common text book examples and vendor demonstration models:

- Industrial strength simulation models are too large to view with a single animation window. The ability to selectively view several parts of the same model simultaneously would be helpful. These multiple windows should have the ability to graphically display dynamic statistical data.

- Many manufacturing systems never reach a so called steady state. The bottleneck work centers are likely to shift depending on product mix and staffing. Thus, full run summary statistics do not capture the operating characteristics of the manufacturing system. The ability to study problematic time slices in detail are required.
- Manufacturing systems typically involve complex logic that is difficult to model. The pace of work is likely to change during the last few minutes prior to a break or the end of the shift. Overtime is used to extend the shift to complete the current task. Multipurpose workcenters can often be used as an alternative when the preferred workcenter is backlogged. Process yield can also vary depending on the time of shift or mix of crew controlling the operation.

The level of detail required in the manufacturing simulation model is often difficult to determine and can have a great impact on execution speed and model validity. Many simulation tools imply a specific level of detail at the individual part and machine level. While this may be appropriate for designing a workcell, it may be too detailed when the entire facility is modeled.

It is sometimes possible to decompose a large manufacturing system into small components for detailed analysis. This decomposition may not be possible when the components are tightly linked or when the system does not reach steady state conditions. Likewise, it is sometimes necessary to model very short cycle times when those operations are critical to the final decision.

Manufacturing managers are not likely to understand the data formats or reports provided by most simulation tools. To gain confidence in the simulation model, they want to know what data and logic assumptions are used. This may be reported as the number of jobs started per unit of time. The simulation tool should support the need for the entire user community to gain confidence in the model and its results. Only after this confidence has been gained can the model's results be used to improve the manufacturing system.

2 TODAY'S SOLUTION

Over the past decade this author has developed a number of industrial strength simulation models using a variety of data sources (Seppanen 1990 and 1991). Mazziotti and Armstrong (1994) have also developed and presented similar work. Figure 1 illustrates one such typical application. In this case, SIMAN V, is used as the underlying simulation engine. The manufacturing system data are maintained in Excel 5.0 workbooks.

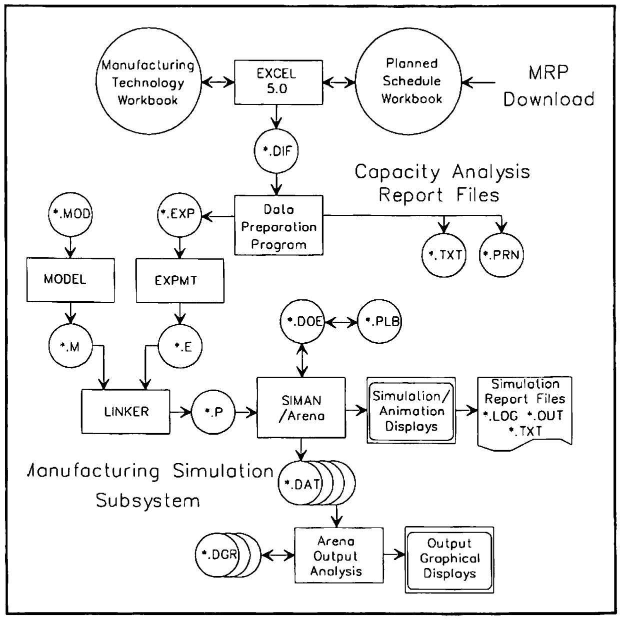

Figure 1: Typical Industrial Strength Manufacturing Simulation Model Data Flow Diagram

At present, the best alternative for data exchange appears to be spreadsheet programs such as Lotus 1-2-3 or Microsoft Excel. These programs support the reading, editing, and writing of data in many common formats such as DIF, CSV, WK?, and TXT. If necessary, the input data can be parsed to separate strings of data into individual data fields. Once in the standard spreadsheet format, the data can be readily edited, sorted, and printed. Data sorting is often an effective means to isolate obvious errors. Summing columns or finding extreme values can also aid in assuring data quality. Spell checking can isolate data problems in text fields. Excel provides the capability to incorporate Visual Basic coding into the spreadsheet. These Macros can be used to assist in data entry, data validation, report generation, and data saving. Saving the edited data in the DIF format has proven to be an effective means to provide simulation processing access.

In general, this spreadsheet approach does not provide a straight forward means to return the edited data to their original sources. Thus, bill of materials errors found during data preparation for the simulation model must usually be manually re-entered into the host system.

Using "real data" to drive the simulation model often presents its own set of problems. Some data fields include special characters such as & and " which may serve as delimiters for the simulation language or the translation programs. The real data may also contain too much information. One model done by the author involving airline scheduling found five different codes for

747 aircraft, whereas the simulation model considered all such aircraft to have the same operating characteristics. Thus, most real data must undergo some editing or error checking prior to being imported into the simulation model.

SIMAN has been used as the underlying simulation engine because of its two ASCII file input data structures. The model data file, *.MOD, contains the system processing logic. This logic defines how individual operations are performed. The model data file does not include any numerical data, like the number of work centers, crew size, reliability, or shift schedule, which are likely to change as alternative operating scenarios are tested. The experiment frame data file, *.EXP, contains all system data and the simulation control parameters. The experiment frame data file is computer generated based on the data the user has provided in the Excel 5.0 workbooks.

In this example case, the manufacturing system involves 80 work centers supported by ten crews working up to three shifts per day and can include scheduled overtime. The system manufactures one general family of products. These products vary in physical size by a factor of 30. The manufacturing facility separates producing into two areas based on product size. In total, the product family includes over 400 options based on size, material, and configuration. The production rate is relatively low volume with 20 to 40 units being started each week. The unit throughput time ranges from one to ten weeks depending on the total processing time and the plant configuration. A simulation model was developed to test alternative configurations to increase system capacity.

Two major SIMAN constructs, Stations and Sequences, were used to develop the example industrial strength manufacturing simulation model illustrated in Figure 1.

- SIMAN Station blocks are collections of code statements which represent physical areas within the real system. A SIMAN Station was developed for each work center. A few additional Stations were coded to support special activities such as collecting components prior to their final assembly. Since the processing logic for most work centers is the same, SIMAN permits one set of code to be used for all common stations. In spite of this ability to effectively reuse model code, the SIMAN model file for the simulation model code still exceeds 680 blocks or 30,000 characters.
- The SIMAN Sequence element is used to specify the routing and associated processing data for each unique production unit. The Sequence element for a typical production unit involves about 80 processing steps and about 9,000 characters of data. The SIMAN experiment frame file for the simulation model has 23 elements and from 50,000 to 2,000,000 characters of data depending on the complexity and the length of the schedule being simulated. Manual maintenance for such an extensive data file would require a well trained user and could be extremely time consuming. In addition, the size of the experiment file makes manual verification a difficult task to assure that all input data are correct.

For the manufacturing simulation model illustrated in Figure 1, two types of Excel workbooks are maintained. The Manufacturing Technology Workbook is comprised of ten worksheets. These worksheets define the physical plant, the available crew, the workstations, and general structure of the units to be produced. The Manufacturing Technology Workbook includes a Visual Basic Macro to save the workbook data in both XLS and DIF formats. The Planned Schedule Workbook contains the planned production release schedule either directly downloaded from the MRP system or manually generated by the user to test a potential product mix.

The Data Preparation Program was written by the author to convert the workbook data into SIMAN experiment frames. While this programming includes much reusable code, each application represents a substantial rewrite effort. As illustrated in Figure 1, the Data Preparation Program generates many data files and/or reports in addition to the SIMAN experiment frame. These report files allow the user to fully document exactly how the manufacturing system was modeled for the simulation run. These reports indicate potential bottlenecks due to limited capacity prior to the simulation execution. These report data can also be loaded in Excel for additional analysis and reporting. The Data Preparation Program illustrated in Figure 1 generates the expected standard processing time for each scheduled unit based on its option characteristics and the data contained in the Manufacturing Technology Workbook. The results of these calculations are reported in detail for verification prior to executing the simulation model.

3 FUTURE SOLUTIONS

The advance of simulation into routine manufacturing applications is restricted in part due to a lack of data exchange capabilities. Software with the ability to be customized is needed to convert bills of material and standard data files into the formats required by simulation tools. The simulation market is too small to economically support the development of such special purpose tools. Instead, general purpose tools are required.

Three current software tools are suggested as prototypes for further research and development consideration:

- HiJaak by Inset Systems is a MS DOS/Windows-based graphical software tool which suggests a pattern for such development. HiJaak can be used to convert graphic images or drawings between a wide variety of formats. This conversion can be manually controlled by the user or, in some cases, automated. Because Inset Systems supports a wide customer base, the product is frequently updated with new capabilities to support additional graphical formats. HiJaak is reasonably priced ($100 range), due to its broad market.
- PREACTOR 200 by The CIMulation Centre Ltd. and Systems Modeling Corp. suggests another form of such a general purpose tool. PREACTOR 200 is a Windows-based finite capacity scheduling tool for supporting manufacturing operations. The unique feature of PREACTOR 200 is its ability to be customized by altering a set of ASCII configuration files. These configuration files control the user interface, menu options, and database contents. Thus, the tool can be radically customized without changing the underlying computer code. In its current form, PREACTOR 200 does not directly offer the capability of translating manufacturing databases into simulation input data files; it does suggest a potential future direction.
- The Visual Basic Macros supported by Microsoft Excel allow for considerable customization within the basic workbook format. This customization might be as simple as user buttons to facilitate moving around the workbook or to completely generate the simulation input data files in their required formats.

One method by which the developers of commercial simulation software could increase the functionality of their products while at the same time possibly decrease total cost is to incorporate standard software wherever possible. For example, few if any commercial simulation software packages have anywhere near the data analysis and graphical capabilities of products such as Lotus 123 or Microsoft Excel. Likewise most simulation software lacks drawing functionality of Corel Draw or AutoCAD LT. Incorporating such tools into the simulation software package could increase the functionality of the package while having a minimal effect on the overall cost. In fact, when the duplicate development time is removed from the simulation package, the combined product should actually be lower in overall cost. Windows based application using OLE tools should allow simulation software developers to concentrate on their unique features and allow data exchange with widely available low-cost standard software.

REFERENCES

Mazziotti, B.W., and F.B. Armstrong. 1994. Creating A Focused Application Simulator With Flexible Decision Making Capabilities. In *Proceedings of the 1994 Winter Simulation Conference*, eds. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, 1296-1303. Institute of Electrical and Electronics Engineers, Lake Buena Vista, Florida.

Seppanen, M.S. 1991. Selecting the Proper Simulation Tool. In *Proceedings of the 1991 European Simulation MultiConference*, ed. E. Mosekilde, 146-151. The Society for Computer Simulation, Copenhagen, Denmark.

Seppanen, M.S. 1990. Special Purpose Simulator Development. In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R. P. Sadowski, and R. E. Nance, 67-71. Institute of Electrical and Electronics Engineers, New Orleans, Louisiana.

AUTHOR BIOGRAPHY

MARVIN S. SEPPANEN is the Principal Consultant for Productive Systems of Winona, Minnesota, an independent Industrial Engineering consulting firm specializing in simulation modeling and analysis of manufacturing systems. He holds his BME, MSIE, and Ph.D. (Operations Research) degrees from the University of Minnesota. Before starting Productive Systems in 1983 he was an Associate Professor of Industrial Engineering at General Motors Institute and The University of Alabama. His consulting and research activities have been primarily in the area of manufacturing systems simulation. He is a Registered Professional Engineer; Senior Member, Institute of Industrial Engineers; Member, The Society for Computer Simulation; Member, Society of Manufacturing Engineers; and is certified at the Fellow Level by the American Production and Inventory Control Society. He currently teaches simulation to Manufacturing Systems Engineering students at the University of St. Thomas in St. Paul, MN.