Proceedings of the 1994 Winter Simulation Conference
ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila

SOFTWARE FOR SIMULATION

Jerry Banks

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0205

ABSTRACT

This tutorial describes computer languages and other software packages that support discrete-event simulation.

1 INTRODUCTION

The next two sections describe general purpose and manufacturing oriented simulation software, respectively. The fourth section describes simulation environments. Then, an animator for simulation is discussed. Finally, some simulation support software is described.

2 GENERAL PURPOSE SOFTWARE

Applications of simulation exist in many arenas as indicated by the papers presented at the Winter Simulation Conference (WSC). The 1993 WSC included applications in the following areas: manufacturing, material handling, health services, military decision support, natural resources, public services, transportation, and communications to mention a few.

These simulation applications are usually accomplished with the use of specially developed simulation software. This tutorial describes the software in two categories. The first of these is software for general purposes. This type of software can solve almost any discrete simulation problem. In this section, five products, GPSS/H™, GPSS/World™, SIMAN V®, SIMSCRIPT II.5®, and SLAMSYSTEM®, will be discussed to provide a feel for this type of software.

GPSS/H is a product of Wolverine Software Corporation, Annandale, VA (Smith and Crain, 1993). It is a flexible, yet powerful tool for simulation. It provides improvements over GPSS V that had been released by IBM many years earlier. These enhancements include built-in file and screen I/O, use of an arithmetic expression as a block operand, interactive debugger, faster execution, expanded control statement availability, and ampervariables that allow the arithmetic combinations of values used in the simulation. The latest release of GPSS/H is version 2.0. It added a floating point clock, built-in math functions, and built-in random variate generators. Options available include Student GPSS/H, Personal GPSS/H within the 640K memory limit, and GPSS/H 386 providing unlimited model size.

GPSS World™, from Minuteman Software, is a complete redesign of GPSS/PC™ (Cox, 1991). It is designed as a high power environment for simulation professionals. It includes both discrete and continuous simulation. Its features include interactivity, visualizability, and configuration flexibility. It utilizes 32-bit computing, virtual memory, preemptive multitasking, symmetric multiprocessing, and distributed simulation.

Highlights include drag-and-drop model building, 512 megabytes of virtual memory for models, point-and-shoot debugging, an embedded programming language, built-in probability distributions, multiple data types, and many other improvements to GPSS/PC™.

The GPSS World™ family is a set of three software products including:

1. GPSS World™ as the center of the family. This self contained modeling environment includes local Simulation Server™ capabilities.
2. Simulation Server™ provides simulation services on a remote networked computer. It does not include a model-building user network.
3. Simulation Studio™ will provide hierarchical modeling and user drawn simulation capabilities. It is scheduled for release in 1995.

There is an enhanced memory version of GPSS/PC that is also available. It allows access of up to 32 mb of memory.

SIMSCRIPT II.5, from CACI Products Company, is a language that allows models to be constructed that are either process oriented or event oriented (Russell, 1993). The microcomputer and workstation versions include the SIMGRAPHICS animation and graphics package. SIMSCRIPT can be used to produce both dynamic and static presentation quality graphics such as histograms, pie charts, bar charts, levels of meters and dials and time plots of variables. Animation of the simulation output is also constructed using SIMGRAPHICS. SIMGRAPHICS can be used also to produce interactive graphical front-ends or forms for entering model input data. An input form may include such graphical elements as menu bars with pull-down menus, text or data boxes, and buttons that are clicked on with a mouse to select an alternative. The graphical model front-end allows for a certain set of modifications to the model to be made without programming, facilitating model use by those that are not programmers.

SIMAN V, from Systems Modeling Corporation, is a general-purpose program for modeling discrete and/or continuous systems (Glavach and Sturrock, 1993; Banks, Burnette, Jones, and Kozloski, 1995). The program distinguishes between the system model and the experiment frame. The system model defines components of the environment such as machines, queues, transporters and their interrelationships. The experiment frame describes the conditions under which the simulation is conducted including machine capacities and speeds, and types of statistics to be collected. "What-if" questions can usually be asked through changing the experiment frame rather than by changing the model definition. Some important aspects of SIMAN V are as follows:

1. Special features that are useful in modeling manufacturing systems including the ability to describe environments as workcenters (stations) and the ability to define a sequence for moving entities through the system.

2. Constructs that enable the modeling of material handling systems including accumulating and non-accumulating conveyors, transporters, and guided vehicles.

3. An interactive run controller that permits breakpoints, watches, and other execution control procedures.

4. The ARENA environment that includes menu-driven point-and-click procedures for constructing the SIMAN V model and experiment, animation of the model using Cinema, the Input Processor that assists in fitting distributions to data, and the Output Processor that can be used to obtain confidence intervals, histograms, correlograms, and so on. (More aspects of the ARENA environment are discussed later in this tutorial.)

5. Portability of the model to all types of computers.

SLAMSYSTEM, from Pritsker Corporation, is an integrated simulation system for PC's based on the Microsoft® Windows™ interface (under MS-DOS) or the OS/2® Presentation Manager™ (O'Reilly, 1993). All features are accessible through pull-down menus and dialog boxes, and are selected from the SLAMSYSTEM Executive Window. A SLAMSYSTEM project consists of one or more scenarios, each of which represents an alternative system configuration. A project maintainer examines the components of the current scenario to determine if any of them have been modified, indicates whether tasks such as model translation should be performed, and allows the user to accomplish these tasks before the next function is requested. SLAMSYSTEM allows multiple tasks to be performed in parallel while the simulation is operating in the background.

Some of the features of SLAMSYSTEM are as follows:

1. Models may be built using a graphical network builder and a forms-oriented control builder, or text editor. Using the first method, a network symbol is selected with the mouse, then a form is completed specifying the parameters for that symbol. The clipboard allows many other operations such as grouping one or more symbols and placing them elsewhere on the network.

2. Output analysis includes a "report browser" that allows alternative text outputs to be compared side-by-side. Output may be viewed in the form of bar charts, histograms, pie charts, and plots. Output from multiple scenarios can be displayed at the same time in bar chart form. Using the Windows environment, multiple output windows can be opened simultaneously.

3. Animations are created under Windows using the Facility Builder to design the static background and the Script Builder to specify which animation actions should occur when a particular simulation event occurs. Animations can be performed either concurrently or in a post-processing mode. Two screens can be updated simultaneously and up to 225 screens can be swapped into memory during an animation.

4. SLAMSYSTEM was designed to be used in an integrated manner. For example, historic data may be read to drive the simulation. CAD drawings may be loaded. Output charts and plots created by SLAMSYSTEM may be exported via the clipboard to other applications.

The newest release of SLAMSYSTEM is Version 4.0. Some of its unique features include the following:

1. Multiple networks in a single scenario. Networks can be constructed in sections and combined at run time. The sections can be reused in future models.

2. New output graphics. These graphics support 3-D, X-Y grids and displaying of point plot data.

3. Direct interface to SimStat (product of MC^2 Analysis Systems). These files may be loaded for advanced statistical analysis.

4. OS/2 metafiles for graphics. The OS/2 metafile format can be read for animation backgrounds or icons.

3 MANUFACTURING ORIENTED SOFTWARE

The software discussed in this section is limited to those associated with manufacturing and further to only six within that category including SIMFACTORY II.5, ProModel for Windows, AutoMod, Taylor II, WITNESS, and AIM. References for these software packages include the following: Goble (1991) for SIMFACTORY II.5, Harrell and Leavy (1993) for ProModel, Norman and Farnsworth (1993) for AutoMod, Thompson (1993) for WITNESS, and Hillen and Werner (1993) for Taylor II. For AIM, the references are Lilegdon (1993) and Lilegdon, Martin, and Pritsker (1994).

SIMFACTORY II.5 is a factory simulator written in SIMSCRIPT II.5 for engineers who are not full time simulation analysts. It operates on the PC under Windows 3.x and OS/2 2.0, or on many workstations. A system amenable to SIMFACTORY II.5 can be modeled rapidly. A model is best constructed in stages by first defining the layout consisting of processing stations, buffers, receiving areas and transportation paths, defining the products, then the resources, next the transporters, and finally the interruptions. The animation automatically follows from the definition of the model. These model elements are pulled from a pallet rather than a menu bar. The resulting model may be changed with a text editor. Flexible flow modeling is supported. For example, OR logic may be used (as in Request Part A OR Part B).

The layout is created by positioning icons, selected from a library, on the screen. As each icon is positioned, characteristics describing it are entered. The products are defined by process plans that define the operations performed on each part and the duration of that operation.

Resources are added to the model in two steps. First, the resource is defined and its quantity is set. Second, the stations requiring specified resources are identified. While resources are moving, simulation time can elapse. Resource requirements are flexible, viz., one unit of Resource A and two units of Resource B can be requested.

Transporters may be batch movers such as fork lifts or they may be conveyors. Characteristics of a transporter are specified (pickup speed, delivery speed, load time, unload time, and capacity of a fork lift, as an example). The transporter path is identified on the screen. Transporters can avoid each other by collision detection and they can carry resources.

Any interruption, planned or unplanned, can be applied to any model element or group of elements (e.g., conveyors, queues, resources, and transporters). Interruptions can require any combination of resources.

Reports are available concerning equipment utilization, throughput, product makespan, and buffer utilization. Multiple business graphics (pie charts, histograms and plots) can be compared at the same time. Data can be compared across multiple runs. Text reports can be customized, and specific statistics can be collected on the elements of interest. A summary report of all replications provides means, standard deviations, and confidence intervals on the model output.

ProModel for Windows, from PROMODEL Corporation, has programming features within the environment, and the capability to add C or Pascal type subroutines to a program. Some of the features of ProModel for Windows are as follows:

1. Models are created using a point and click approach. Intuitive interfaces, interactive dialog and online help are provided. An auto-build feature guides the user through the model building process. An online trainer is available.

2. The software operates in the Windows and OS/2 environments, as a 32-bit application, taking advantage of memory management techniques, synchronized windowing and data exchange. Windows fonts, printer drivers, cooperative multitasking, and the Dynamic Link Library are available.

3. Virtually unlimited model size is offered.

4. The simulator offers a 2-D graphics editor with scaling, rotating, and so on. Icons can be defined using either vector based or pixel graphics. These icons are saved as bitmaps at runtime for fast animation during the simulation.

5. CAD drawings as clipart can be imported as well as process information and schedules. Customized output reports and spreadsheet files can be produced. If the data is another Windows application, cutting and pasting can be accomplished.

6. The static and dynamic elements of the animation are developed while defining the model. That is, the simulation model and animation are integrated.

7. Business output graphics are automatically provided and may be printed in color.

8. Only standard hardware is required (IBM or compatible with VGA graphics). No special graphics cards, monitors, or math coprocessor chip is needed.

9. Preprogrammed constructs are provided. This allows for fast modeling of multi-unit and multi-capacity locations, shared and mobile resources, downtime, shifts, and so on.

10. Automatic statistics are available.

11. Submodels allow the creation of a library of templates of work steps, activities, or sub-processes that can be reused. This allows for model construction to be accomplished by a team with later merger of submodels into one model.

12. A free runtime, multiple scenario, capability is provided.

A model is constructed by defining a route for a part or parts, defining the capacities of each of the locations along the route, defining additional resources such as operators or fixtures, defining the transporters, scheduling the part arrivals, and specifying the simulation parameters. The software then prompts the user to define the layout and the dynamic elements in the simulation.

AutoMod, from AutoSimulations, Incorporated, has general programming features including the specification of processes, resources, loads, queues, and variables. Processes are specified in terms of traffic limits, input and output connections, and itineraries. Resources are specified in terms of their capacity, processing time, MTBF, and MTTR. Loads are defined by their shape and size, their attributes, generation rates, generation limits, and start times, as well as their priority.

The simulator is very powerful in its description of material handling systems. AGVs, conveyors, bridge cranes, AS/RS's, and power and free devices can be defined. The range of definition is extensive. For example, an AGV can be defined in terms of the following: multiple vehicle types, multiple capacity vehicles, path options (unidirectional or bidirectional), variable speed paths, control points, flexible control and scheduling rules, arbitrary blocking geometries, automatic shortest-distance routing, and vehicle procedures.

Numerous control statements are available. For example, process control statements include If-Then-Else, While-Do, Do-Until, Wait-Until, and Wait For. Load control, resource control and other statements are also available. C functions may be defined by the user. Attributes and variables may be specified.

The animation capabilities include true 3-D graphics, rotation, and tilting, to mention a few. A CAD-like drawing utility is used to construct the model. Business graphics can be generated.

The latest release, Version 7.0, contains a simulator within AutoMod. Features of the simulator include its spreadsheet interface. This eliminates the need for programming in building models. The spreadsheet interface also allows the definition of models outside of AutoMod.

Another option as a separate utility is AutoStat. It provides simulation warmup capability, scenario management, confidence interval generation, and design of experiments capability.

Taylor II is a Dutch product developed by F&H Logistics and Automation B.V. Working with Taylor II starts with building a model. All model building is menu driven. A model in Taylor II consists of four fundamental entities: elements, jobs, routings, and products. The element types are inout, machine, buffer, conveyor, transport, path, aid, warehouse, and reservoir. One or more operations can take place at an element. The three basic operations are processing, transport, and storage.

Defining a layout is the first step when building a model. Layouts consist of element types. By selecting the elements in sequence, the product path or routing is defined. Routing descriptions may be provided from external files.

The next step is detailing the model. In this step the parameters are provided. In addition to a number of default values, Taylor II uses a macro language called TLI for Taylor Language Interface. TLI is a programming language that permits modifications of model behavior in combination with simulation-specific predefined and user-definable variables. TLI can also be used interactively during a simulation run to make queries and updates. An interface to C, Basic, and Pascal is also available. Local and global attributes are available.

During simulation, zoom, pan, rotate, and pause are possible. Modifications can be made on-the-fly. The time representation is fully user-definable (hours, days, seconds, and so on can be mixed).

Output analysis possibilities include predefined graphics, user-defined graphics, predefined tabular reports, and user defined reports. Examples of predefined graphics are queue histograms and utilization pies. User-defined outputs include bar graphs, stacked bars, and other business graphics. Predefined tables include job, element and cost reports.

Animation capabilities include both 2-D and 3-D. The 3-D animation can be shaded. Standard indicators can be shown for elements. Icon libraries for both 2-D and 3-D animation are provided. Each of these libraries contains more than 50 icons.

Additional features include 500 pages of online, context sensitive help with index and page-search capability. Educational support materials are available.

WITNESS, from AT&T Istel, contains many elements for discrete-part manufacturing. For example, machines can be single, batch, production, assembly, multi-station, or multi-cycle. Conveyors can be accumulating or non-accumulating. Options exist for labor, vehicles, tracks, and shifts. WITNESS also contains elements for continuous processing including processors, tanks, and pipes.

Variables and attributes may be specified. Parts arrivals may be scheduled using a file. Distributions and functions can be used for specifying operation times and for other purposes. Machine downtime can be scheduled on the basis of operations, busy time, or available time. Labor is a resource that can be pre-empted, use a priority system, and be scheduled based on current model conditions.

Track and vehicle logic allow requests for certain types of jobs, vehicle acceleration and deceleration, park when idle, and change destinations dynamically. Many types of routing logic are possible in addition to the standard push and pull. For example, If-Then-Else conditions may be specified.

Simulation actions, performed at the beginning and end of simulation events, may employ programming constructs such as For-Next, While-End, and GoTo-Label. The user can look at an element at any time and determine the status of a part.

Reporting capabilities include dynamic on-screen information about machines and elements. Reports may be exported to spreadsheet software.

Tools within the language include access to the model database through C language, arithmetic and logical operators, save current status of model, built-in status functions and many others. In addition, all of the above mentioned features can be enhanced through the use of C language.

Debugging or brainstorming can be accomplished by stopping the model, changing desired parameters, and continuing with the model from the same point in simulation time.

An animation is constructed along with the model definition. This animation and statistical feedback can be turned on or off during any run. Many changes to the model may be made at any time.

Built-in experimentation capabilities are available from the menu bar. The results of the experiments are output to a CSV file by default, or other file types by user choice. The CSV file is in a format that allows the internal statistics package to create confidence intervals.

The latest release of the software is Version 6. Its capabilities include the following:

1. The ability to store up to 15,000 variables or attributes.

2. Up to 1000 distinct random number streams.

3. Bitmap import/export with icon sizes increased to 256x256 pixels.

4. 'Module' element for hierarchical modeling. One icon represents the detail existing in another portion of the model.

5. 'Selector Bar' for filling fields with rules, distributions, built-in functions, and so on, for defining elements 'on the fly.'

AIM (Analyzer for Improving Manufacturing), from Pritsker Corporation, is one of three applications of **FACTOR**. Other packages are Factor Production Manager and Leitstand. Factor Production Manager performs detailed planning/scheduling of operations, order promising, order release, and supply-chain management. All of these applications use the same data base.

AIM models are OS/2 based and built graphically with icons that represent machines, operators, conveyors, and so on, placed directly on the screen. The animations are created in a virtual window. The current release of AIM is version 5.3. During a simulation, the model can be stopped to check its status or add other components, then continue the simulation. Performance data is dynamically updated and displayed while the simulation is running. A dynamic Gantt chart is provided for tracking machine and operator status. Inventory levels and material handling utilization can also be graphed dynamically. Outputs include bar charts, pie charts, and plots of inventory levels. Alternately, information can be transferred to other software for development of presentation graphics.

Features of AIM include the following:

1. Manufacturing representation. Manufacturing specific modeling components can represent a variety of discrete manufacturing processes. Standard rules provide choices that are interpreted with processes. Custom rules may be written to extend the logic.

2. Integrated with scheduling applications. Models written with AIM can be used with other FACTOR applications providing support for capacity, logistics, production scheduling, supply-chain analysis, and schedule management.

3. Manufacturing data. AIM is built around a relational database that stores the manufacturing operation and simulation output. Part descriptions, process plans, order release schedules, machine locations and schedules, shift schedules, and so on can be transferred from other data sources to the AIM database.

4. Animation support. AIM models are built graphically and are animated automatically during model construction.

5. Interactive model building and simulation. Components are located on a scaled facility background. Intelligent defaults are provided for all components. Components are customized by completing forms. During execution, the modeler can change the status of a component and observe the simulated impact on the manufacturing system.

6. Comparison of alternatives. The AIM project framework organizes all aspects of a manufacturing simulation project. Alternative models of the manufacturing process are stored. Comparison reports show model performance data to identify differences between alternatives.

7. AIM Gantt charts. The latest release of AIM supports the creation of Gantt charts for the improved verification and validation of models. Model

performance can be reviewed in the Gantt chart to follow a single load or the decisions of a resource.

8. Cost modeling methodology. The next release of AIM, scheduled for the first quarter of 1995, will include a detailed cost modeling capability. AIM models will represent alternative costing philosophies such as standard cost, ABC, and so on.

4 SIMULATION ENVIRONMENTS

A simulation environment contains many utilities to conduct a simulation study. These capabilities include input data analysis, model entry support, scenario management, animation, and output data analysis.

ARENA, a product of Systems Modeling Corporation, is intended to provide the power of SIMAN to those for whom learning the language is burdensome and enhance the use of tools used by SIMAN modelers (Collins and Watson, 1993). Assume that a person, other than a simulation analyst, wants to use SIMAN. Currently, he or she must understand the blocks used in the model and the elements used in the experiment frame to proceed. Under ARENA, the user could extract a module, place it in its appropriate location, and parameterize it without learning the SIMAN language. For SIMAN language modelers, ARENA is intended to increase their functionality, eliminating the need for writing similar code in different models.

SIMAN is the engine for ARENA and Cinema is the animation system that is used. Other products included in ARENA are an input processor, and an output processor. A shop-floor analysis capability is also being developed and will be supported by ARENA. This latter product will be oriented toward scheduling and real-time shop-floor applications of simulation.

The term "modules" is used to represent the building blocks available for creating models. The most fundamental feature of ARENA is that a simulation analyst can construct a module definition for use by others. These module definitions may be combined to create other modules. SIMAN Base Modules form the lowest possible level of modules. These correspond to basic SIMAN modeling constructs (blocks and elements). All other modules, called Derived Modules, are built from Base Modules or other Derived Modules. This increases the speed at which models can be built, and aids in understanding by those not familiar with SIMAN blocks and elements. Templates provide modelers with a domain-specific Module Definition set. For example, a Manufacturing Template could be sold by Systems Modeling or by third parties.

A much revised Cinema is contained within ARENA. This animation capability is integrated with ARENA modules. For example, when adding a module to represent a manufacturing process, a modeler might get both the modeling logic to represent the process, as well as the Cinema components representing work-in-process, and the status of the resource (busy, idle, in repair, etc.).

5 ANIMATORS

Most simulation animators are integrated with the software. However, this is not always the case, and the introduction of general purpose animation packages allows the use of custom made environments.

Proof Animation is a product of Wolverine Software Corporation (Earle and Henriksen, 1993). Any software that can write ASCII data to a file can drive Proof Animation™. Thus, BASIC, C, FORTRAN, GPSS/H, SIMAN V, and SIMSCRIPT II.5, among others, can serve as drivers. Animation is accomplished by using a static background, the layout file, and a trace file that contains dynamic events. Some of the features of the software are as follows:

1. Graphics are vector based, similar to CAD programs.
2. Zoom in and zoom out are supported. Maximum resolution is guaranteed at any scale as the drawing is recalculated.
3. Drawing takes place on a coordinate grid using mouse-driven primitives.
4. Moving objects are defined internally by their geometry.
5. Statistics can be displayed dynamically.
6. Animation occurs in a postprocessing mode.
7. Motion is smooth on VGA PC's.
8. There is a steady ratio of animated (simulation) time to viewing (wall clock) time. This ratio may be varied while the animation is running.
9. Top view can be changed to isometric and back to top view instantly.
10. An option allows the construction of a demo disk.
11. CAD layouts can be imported and exported through an optional utility.

As an example of the integration of PROOF with other simulation software, consider the interface with MOGUL from High Performance Software, Inc. The user can graphically build a model using PROOF, complete the model using MOGUL, execute the model using GPSS/H, and view an animation using PROOF.

6 SIMULATION SUPPORT SOFTWARE

Two products, among many that are available, are discussed in this section. The first is UNIFIT® II from

Averill M. Law and Associates, for input data modeling (Vincent and Law, 1993). The second is SIMSTAT from MC^2 Analysis Systems for input and output data analysis (Blaisdell and Haddock, 1993).

UNIFIT II is used to model input data distributions. The product can be used in conjunction with the major simulation software in producing the necessary code to enter distributions. The software actually augments the built-in distribution capability of most of the software described previously. For example, GPSS/H has four built-in distributions, but UNIFIT increases that number to 21. The software can be operated in three different modes. Guided selection mode automatically determines the best fitting distribution. Manual selection mode is designed for experienced simulationists allowing them to select the appropriate statistical tools and the order of application in determining an appropriate distribution. Finally, the no-data model selection mode assists in choosing a source of randomness when no corresponding data exist.

SIMSTAT 2.0 is an interactive graphical software tool that performs statistical analysis on simulation input and output data. It is designed to work seamlessly with many simulation packages. The software uses pull down menus and is integrated into the Windows environment. Data is maintained in a spreadsheet format for editing, examination and analysis. SIMSTAT takes advantage of the Windows Clipboard. Some of the many graphical capabilities of SIMSTAT include the fitting of input distributions to data, the determination of initialization bias, and autocorrelation plots.

7 SUMMARY

This tutorial describes software for simulation organized in four categories. The first of these is general purpose software. The second is manufacturing oriented software. Next, a simulation environment is discussed. Then, an animator is described. Finally, some simulation support software is discusssed.

REFERENCES

Banks, J., B. Burnette, J.D. Rose, and H. Kozloski. 1995. *SIMAN V and CINEMA V.* New York: John Wiley.

Blaisdell, W.E., and J. Haddock. 1993. Simulation analysis using SIMSTAT 2.0. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 213-217. Association for Computing Machinery, New York, NY.

Collins, N. and C.M. Watson. 1993. Introduction to ARENA. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 205-212. Association for Computing Machinery, New York, NY.

Cox, S.W. 1991. GPSS World: A brief preview. Simulation Conference, ed. B.L. Nelson, G.M. Clark, 59-61. Association for Computing Machinery, New York, NY.

Cox, S.W. 1992. Simulation Studio™. In *Proceedings of the 1992 Winter Simulation Conference*, ed. J.J. Swain, D. Goldsman, R.C. Crain, J.R. Wilson, 347-351.Association for Computing Machinery, New York, NY.

Earle, N.J. and J.O. Henriksen. 1993. PROOF Animation: Better animation for your simulation. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 172-178 Association for Computing Machinery, New York, NY.

Glavach, M.A. and D.T. Sturrock. 1993. Introduction to SIMAN/Cinema. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 190-192. Association for Computing Machinery, New York, NY.

Goble, J. 1991. Introduction to SIMFACTORY II.5 In *Proceedings of the 1991 Winter Simulation Conference*, ed. B.L. Nelson, W.D. Kelton, G.M. Clark, 77-80. Association for Computing Machinery, New York, NY.

Harrell, C.R. and J.J. Leavy. 1993. ProModel tutorial. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 184-189. Association for Computing Machinery, New York, NY.

Hillen, D.W. and D. Werner. 1993. Taylor II manufacturing simulation software. In *Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 276-280. Association for Computing Machinery, New York, NY.

Lilegdon, W.R., D.L. Martin, and A.A.B. Pritsker. 1994. FACTOR/AIM: A manufacturing simulation system. *Simulation* 62:367-372.

Norman, V.B. and K.D. Farnsworth. 1993. AutoMod. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 249-254. Association for Computing Machinery, New York, NY.

O'Reilly, J.J. 1993. Introduction to SLAM II and SLAMSYSTEM. In *1993 Winter Simulation*

Conference Proceedings, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 179-183. Association for Computing Machinery, New York, NY.

Russell, E.C. 1993. SIMSCRIPT II.5 and SIMGRAPHICS tutorial. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 223-227. Association for Computing Machinery, New York, NY.

Smith, D.S. and R.C. Crain. 1993. Industrial strength simulation using GPSS/H. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 223-227. Association for Computing Machinery, New York, NY.

Thompson, W.B. 1993. A tutorial for modeling with the WITNESS visual interactive simulator. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 159-164. Association for Computing Machinery, New York, NY.

Vincent, S.G., and A.M. Law. 1993. Unifit II: Total support for simulation input modeling. In *1993 Winter Simulation Conference Proceedings*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles, 199-204. Association for Computing Machinery, New York, NY.

AUTHOR AUTOBIOGRAPHY

JERRY BANKS is a Professor in the School of Industrial and Systems Engineering at the Georgia Institute of Technology, Atlanta, Georgia, USA. He is the author, co-author or editor of numerous texts, several chapters in texts, and many technical papers. He served for eight years as IIE's representative to the Board of WSC, including two years as Board Chair.