KATHMANDU UNIVERSITY JOURNAL OF SCIENCE, ENGINEERING AND TECHNOLOGY
VOL. 5, No. I, JANUARY, 2009, pp 104- 120.

A CRITICAL EVALUATION AND COMPARISON OF FOUR MANUFACTURING SIMULATION SOFTWARES

[1]Rajesh Verma*, [2]Ashu Gupta, [3]Kawaljeet Singh

[1]Lovely School of Business, Lovely Professional University
Phagwara, Punjab (INDIA)

[2]Department of Computer Applications, Apeejay Institute of Management
Rama Mandi-Hoshiarpur Road, Jalandhar-144007 (INDIA)

[3]Director, University Computer Centre, Punjabi University
Patiala, Punjab (INDIA)

*Corresponding author: dr.rajeshverma@rediffmail.com

Received 26 September, 2008; Revised 12 December, 2008

ABSTRACT

An increase in the use of simulation as a modeling and analysis tool has resulted in a growing number of simulation software products in the market. Companies are seeking advice about the desirable features of software for manufacturing simulation, depending on the purpose of its use. Because of this, the importance of an adequate approach to simulation software evaluation and comparison is apparent. This paper presents a critical evaluation of several widely used manufacturing simulators: ProModel, AutoMod, HyperMesh and ProcessModel. Following a review of research into simulation software evaluation, an evaluation and comparison of the above simulators is performed. The main purpose of this evaluation and comparison is to discover the suitability of certain types of simulators for particular purposes.

Keywords: Simulation, Simulation software, Evaluation, Comparison, Rating

INTRODUCTION

Growing competition in many industries has resulted in a greater emphasis on developing and using automated manufacturing systems to improve productivity and to reduce costs. Due to the complexity and dynamic behavior of such systems, simulation modeling is becoming one of the most popular methods of facilitating their design and assessing operating strategies.

An increasing need for the use of simulation is reflected by a growth in the number of simulation languages and simulators in the software market. When a simulation language is used, the model is developed by writing a program using the modeling construct of the language. This approach provides flexibility, but it is costly and time consuming. On the other hand, a simulator allows the modeling of a specific class of systems by data or graphical entry, and with little or no programming.

An evaluation of some of the most popular data driven simulators dedicated to the simulation of manufacturing systems is presented in this paper. The evaluation is not performed in order to discover which is 'the best' simulator, because such a term does not exist in the context of simulation software. The main reason for this is a constant updating of existing software and the release of new software products. Hence, the evaluation presented in this paper is primarily performed to determine the suitability of each simulator for different software purposes.

Following a review of previous research in simulation software evaluation, the manufacturing simulators to be evaluated are briefly introduced, as well as an evaluation framework used for their evaluation. On the basis of the evaluation, a method of rating simulators is proposed, and the suitability of the simulators for different purposes is discussed. The conclusions outline the main findings derived in this research.

RESEARCH IN SOFTWARE EVALUATION AND COMPARISON

The starting point for the research was to review previous studies on the evaluation and comparison of simulation software tools. Although there are many studies that describe the use of particular simulation packages or languages, for example, Fan and Sackett (1988), Taraman (1986), Bollino (1988) and so on, relatively few comparative assessments were found like Abed et al. (1985), Law and Kelton (1991).

Some of the evaluations of simulation languages include: a structural and performance comparison between SIMSCRIPT II.5 and GPSS V by Scher (1978); an efficiency assessment of SIMULA and GPSS for simulating sparse traffic by Atkins (1980); and a quantitative comparison between GPSS/H, SLAM and SIMSCRIPT II.5 by Abed *et a/.* (1985).

SLAM, ECSL and HOCUS were used for the comparison of event, entity and process-based approaches to modeling and simulating manufacturing systems by Ekere and Hannam (1989). Several criteria describing programming features, model development characteristics, experimental and reporting features, and commercial and technical features were specified.

Law and Haider (1989) provided a simulation software survey and comparison on the basis of information provided by vendors. Both simulation languages and simulators such as FACTOR, MAST, WITNESS, XCELL + and SIMFACTORY II.5 are included in this study. Instead of commenting on the information presented about the software, the authors concluded that there is no simulation package which is completely convenient and appropriate for all manufacturing applications.

A similar approach to software comparison has been taken by Grant and Weiner (1986). They analyzed simulation software products such as BEAM, CINEMA, PCModel, SEE WHY and SIMFACTORY II.5, on the basis of information provided by the vendors. The authors do not comment on the features provided by the software tools.

Law and Kelton (1991) described the main characteristics and building blocks of AutoMod II, SIMFACTORY II.5, WITNESS and XCELL +, with a limited critical comparison based on a few criteria. Similarly, Carrie (1988) presented features of GASP, EXPRESS, GENETIK, WITNESS and MAST, but again without an extensive comparison.

SIMFACTORY II.5, XCELL +, WITNESS were compared by modeling two manufacturing systems by Banks et al. (1991). The main results of the comparison revealed that SIMFACTORY II.5 and XCELL + did not have robust features, while WITNESS had most of them. Such conclusions were obtained on the basis of twenty two criteria.

Mackulak and Savory (1994) carried out a questionnaire survey on the most important simulation software features. The most important features identified include: a consistent and user friendly user interface; database storage capabilities for input data; an interactive debugger for error checking; interaction via mouse; a troubleshooting section in the documentation;

storage capabilities for simulation models and results; a library of reusable modules of simulation code; and a graphical display of input and output.

Hlupic and Paul (1999) presented criteria for the evaluation and comparison of simulation packages in the manufacturing domain together with their levels of importance for the particular purpose of use. However, it is indicated which criteria are more important than others, according to the purpose of software use.

Tewoldeberhan *et al.* (2002) proposed a two-phase evaluation and selection methodology for simulation software selection. Phase one quickly reduces the long-list to a short-list of packages. Phase two matches the requirements of the company with the features of the simulation package in detail. Different methods are used for a detailed evaluation of each package. Simulation software vendors participate in both phases.

Seila *et al.* (2003) presented a framework for choosing simulation software for discrete event simulation. By evaluating about 20 software tools, the proposed framework first tries to identify the project objective, since a common understanding of the objective will help frame discussions with internal company resources a well as vendors and service providers. It is also prudent to define long-term expectations. Other important questions deal with model dissemination across the organization for others to use, model builders and model users, type of process (assembly lines, counter operations, material handling) the models will be focused, range of systems represented by the models etc.

An analysis of the above studies in simulation software evaluation and comparison reveals that several comparative studies are based on information provided by vendors, and lack any criticism. It seems likely that many authors did not have an opportunity to test all the software tools considered and use them for developing complex models of real systems. Although some of the evaluation studies consider WITNESS, SIMFACTORY, XCELL+ and none of these evaluations and comparisons is comprehensive.

For these reasons, this research set out to produce a more extensive and critical evaluation of four manufacturing simulators, based on 12 main groups of features and having more than 200 features.

EVALUATION OF MANUFACTURING SIMULATORS

Four manufacturing simulators are evaluated in this research: ProModel, AutoMod, HyperMesh and ProcessModel. They are all data-driven, visual, interactive, manufacturing oriented simulators. Nevertheless, there are many differences between these software tools. As simulation software packages usually change in every subsequent release it is wise to indicate which versions of simulators under consideration are evaluated. Appendix A provides this information.

Evaluation has been performed using 12 main groups of features containing more than 200 features. These groups are used as the basis for rating the simulators. Such an approach is taken because it is assumed that it will be more convenient and useful to assess the general performance of each software tool regarding a particular group of criteria, rather than to evaluate every single criterion.

SIMULATION SOFTWARE EVALUATION CRITERIA

The criteria derived can be applied to the evaluation of any general or special purpose simulation package. For this study four main groups are defined to develop the framework for the evaluation. Features within each group are further classified into subcategories, according to their character. The main categories are:

1. *Hardware and software considerations*: coding aspects, software compatibility, user support;
2. *Modeling capabilities*: general features, modeling assistance;
3. *Simulation capabilities*: visual aspects, efficiency, testability, experimentation facilities, statistical facilities; and
4. *Input/Output issues*: input and output capabilities, analysis capabilities.

Owing to the comprehensiveness of the evaluation framework, individual criteria within each group are merely listed, and generally described in the context of a particular group. According to the type of each criterion, the classification determines whether, for example, a certain feature exists in the package, determines the quality of features provided, or lists types of alternatives available within a particular feature.

CRITERIA FOR HARDWARE AND SOFTWARE CONSIDERATIONS

Table 1: Items for Coding Aspects

		Very High	High	Medium	Low	Very Low
1.1.1	Quality of the support for programming	O	O	O	O	O
1.1.2	Efficiency of Compilation	O	O	O	O	O
		Very Good	Good	Average	Poor	Very Poor
1.1.3	Built-in logic builder	O	O	O	O	O
1.1.4	Program Generator	O	O	O	O	O
1.1.5	Snippet code help	O	O	O	O	O
		Very Large	Large	Medium	Small	Very Small
1.1.6	Built-in functions	O	O	O	O	O
		Very Easy	Easy	Moderate	Tough	Very Tough
1.1.7	Ease of entering text/code	O	O	O	O	O
		Possible				Not Possible
1.1.8	User defined functions	O				O
1.1.9	Writing comments/notes in model building activity	O				O
1.1.10	Creation of macros and arrays	O				O
		Provided				Not Provided
1.1.11	Global variables	O				O
1.1.12	Interface to user written programs	O				O

Coding Aspects

The possibility of additional coding might be a very important feature of a package. This feature determines the flexibility and robustness of the software, which is especially valuable when complex systems are to be modeled. Criteria included in this group determine compilation efficiency, the programming concepts supported, logic builder availability etc.

Table 2: Items for Software Compatibility

1.2.1	Integration with spreadsheet packages	O Excel	O Lotus	O Other __________
1.2.2	Integration with statistical packages (curve-fitting tools)	O SPSS	O Stat Fit	O Other __________
1.2.3	Integration with computer-aided software	O AutoCAD		O Other __________
1.2.4	Integration with database management systems	OSAP	O Oracle	O Other __________
1.2.5	Integration with manufacturing requirements planning software	O Possible		O Not Possible
1.2.6	Is it possible to do broad level scheduling with Simulation S/W	O Yes		O No

Software Compatibility

These criteria evaluate whether the package can be interfaced to other software systems, in order to exchange data with these systems. This feature can considerably enhance the capabilities of the package, especially when complex real systems are modeled.

User Support These criteria evaluate the type and quality of user support provided by the software supplier, which can facilitate learning and using the package. These criteria not only include technical support in the form of documentation, and demo disks, but also include a variety of services provided by the software supplier which ease the use of the package and keep the user informed about plans for future software improvements.

Table 3: Items for User Support

		Very Good	Good	Average	Poor	Very Poor
1.3.1	Quality of manuals	O	O	O	O	O
1.3.2	Tutorial	O	O	O	O	O
1.3.3	Run-time help	O	O	O	O	O
1.3.4	Software maintenance facility	O	O	O	O	O
1.3.5	Training course	O	O	O	O	O
1.3.6	Web based support	O	O	O	O	O
1.3.7	Troubleshooting facility	O	O	O	O	O
1.3.8	Quality of documentation	O	O	O	O	O
1.3.9	Demo models	O	O	O	O	O
		Very Frequent	Frequent	Average	Rare	Very Rare
1.3.10	User group meetings	O	O	O	O	O
1.3.11	Frequency of training courses	O	O	O	O	O
		Provided				Not Provided
1.3.12	Discussion groups on the internet	O				O
1.3.13	User community web page	O				O

CRITERIA FOR MODELING CAPABILITIES

General features

Criteria included in this group describe general features of the package. Most of these criteria relate to modeling aspects such as the type of formal logic needed for modeling (if any), the method of changing the state of the model (process based, activity based, event based, three phase, or a combination of these methods), type of simulation (discrete event, continuous or combined), the level of modeling transparency, etc. There are also some criteria that evaluate the level of experience and formal education in simulation required by the user, and examine how easy it is to learn and use the package.

Table 4: Items for General Features

2.1.1	Type of simulation	O Discrete event		O Continuous		O Both
2.1.2	Purpose	O General purpose		O Manufacturing Oriented		O Other
		Very High	High	Medium	Low	Very Low
2.1.3	Representativeness of models	O	O	O	O	O
2.1.4	User friendliness	O	O	O	O	O
2.1.5	Experience required for software use	O	O	O	O	O
2.1.6	Formal education in simulation required for software use	O	O	O	O	O
		Very Easy	Easy	Moderate	Tough	Very Tough
2.1.7	Ease of learning	O	O	O	O	O
2.1.8	Ease of using	O	O	O	O	O
		Very Good	Good	Average	Poor	Very Poor
2.1.9	Run-time interface capability for scenario creation	O	O	O	O	O
2.1.10	Conceptual model generator	O	O	O	O	O
2.1.11	Multiple branch decision making	O	O	O	O	O
2.1.12	Probabilistic branch decision making	O	O	O	O	O
		Possible				Not Possible
2.1.13	Distributed simulation on network environment	O				O
2.1.14	Cut, copy, paste of objects	O				O
2.1.15	Possibility to built near Real-time simulation models	O				O
		Provided				Not Provided
2.1.16	Easy to use templates	O				O
2.1.17	Customizable window environment	O				O
2.1.18	Splines, Polygon and orthogonal curve types	O				O

Modeling Assistance

Criteria systematized in this group evaluate the type and level of assistance provided by the package during modeling. For example, these criteria examine the comprehensiveness of prompting, on-line help if it is provided, whether the package enables modular model development and writing the documentation notes (this feature enables the writing of documentation concurrently with the model development), and whether the model and data can be separated.

Table 5: Items for Modeling Assistance

		Very Good	Good	Average	Poor	Very Poor
2.2.1	Libraries and templates of simulation objects	O	O	O	O	O
2.2.2	Warning messages	O	O	O	O	O
2.2.3	Intelligent Prompting	O	O	O	O	O
2.2.4	Facility for designing reusable user defined elements	O	O	O	O	O
2.2.5	3D models library	O	O	O	O	O
2.2.6	Bubble help	O	O	O	O	O
2.2.7	Context sensitive prompt to facilitate model development	O	O	O	O	O
		Provided				Not Provided
2.2.8	Undo/redo commands	O				O
2.2.9	Facility to insert comments	O				O

CRITERIA FOR SIMULATION CAPABILITIES

Visual Aspects

Graphical presentations of simulation models and animation of simulation are very important characteristics of simulation software.

Criteria included in this group relate to the type and quality of graphical facilities provided by the package. These criteria evaluate, for example, whether it is possible to perform an animation of the simulation experiments, the types of animation provided by the package, and whether it is possible to manipulate icons.

Table 6: Items for Visual Aspects

		Very Good	Good	Average	Poor	Very Poor
3.1.1	Shape libraries	○	○	○	○	○
3.1.2	3D-animator	○	○	○	○	○
3.1.3	Logical animation	○	○	○	○	○
3.1.4	Network animation	○	○	○	○	○
3.1.5	Scenario viewer	○	○	○	○	○
3.1.6	Antialias display	○	○	○	○	○
3.1.7	Dashboard facility	○	○	○	○	○
3.1.8	Customizable entity appearance	○	○	○	○	○
3.1.9	Customizable path appearance	○	○	○	○	○
3.1.10	Library for real-time simulations	○	○	○	○	○
3.1.11	Virtual reality animation	○	○	○	○	○
		Provided				Not provided
3.1.12	HotSpot Evaluator	○				○
3.1.13	Flowcharting Module	○				○
3.1.14	Animation of image changes	○				○
3.1.15	Facility for customizing the view of the model	○				○
3.1.16	Playback mode	○				○
		Provided				Not provided
3.1.17	Animation with visual clock	○				○
3.1.18	Zoom function	○				○
3.1.19	Panning	○				○
3.1.20	Print screen facility	○				○
		Possible				Not Possible
3.1.21	Import of AutoCAD drawings	○				○
3.1.22	Multiple screen layout	○				○
3.1.23	Merging icon files	○				○
3.1.24	Resizing of icons	○				○
3.1.25	Changing the color of the element status display	○				○
3.1.26	Change of icons during simulation	○				○

Efficiency

Criteria classified in this group determine the effectiveness and the power of simulation software. Efficiency is expressed both by the capability of the software to model a variety of complex systems and by the characteristics which can save time needed for modeling, and improve the quality of modeling, such as model reusability, reliability, compilation and execution time and multitasking.

Table 7: Items for Efficiency

		Very High	High	Medium	Low	Very Low
3.2.1	Robustness	O	O	O	O	O
3.2.2	Level of detail	O	O	O	O	O
3.2.3	Adaptability to model changes	O	O	O	O	O
3.2.4	Reliability	O	O	O	O	O
		Very Large	Large	Medium	Small	Very Small
3.2.5	Number of elements in the model	O	O	O	O	O
3.2.6	Number of queuing policies	O	O	O	O	O
3.2.7	Time scale for model building	O	O	O	O	O
3.2.8	Model execution time	O	O	O	O	O
		Very Good	Good	Average	Poor	Very Poor
3.2.9	Model Protection	O	O	O	O	O
		Possible				Not Possible
3.2.10	Model status saving	O				O
3.2.11	Multitasking	O				O
3.2.12	Model chaining (i.e. linking outputs from different models)	O				O
3.2.13	Editing partially developed models	O				O
3.2.14	Interactive handling of parameters during experimentation	O				O
3.2.15	Model reusability	O				O
		Provided				Not provided
3.2.16	Variable watches	O				O
3.2.17	Activity based costing	O				O

Testability

This group comprises criteria that examine which facilities for model verification are provided by the package. These facilities include error messages, displays of the values of logical elements such as functions and variables, the possibility of obtaining special files for verification such as list, trace and echo files, provision of step function, etc.

Table 8: Items for Testability

3.3.1	Moment of error diagnosis	O Model entry	O Compilation			O Execution
		Possible				Not Possible
3.3.2	Display of attributes	O				O
3.3.3	Display of variables	O				O
3.3.4	Display of element's state	O				O
3.3.5	Replication of Run-length	O				O
3.3.6	Change in simulation speed	O				O
3.3.7	Execution trace	O				O
3.3.8	Logic checks	O				O
3.3.9	Runtime error viewer	O				O
		Possible				Not Possible
3.3.10	Explode function (showing a state of an element)	O				O
3.3.11	List of used elements	O				O
3.3.12	Backward clock	O				O
3.3.13	Step function (event to event jumping)	O				O
3.3.14	Display of parts flow tracking record collected during simulation run	O				O
3.3.15	Audible alarms	O				O
3.3.16	Rejection of illegal inputs	O				O
3.3.17	Syntax checker	O				O
3.3.18	Search & replace capability	O				O
3.3.19	Antithetic numbers	O				O
3.3.20	Multiple windows during simulation run	O				O
		Provided				Not provided
3.3.21	User Pause facility	O				O
3.3.22	OLE compatibility	O				O
		Very Good	Good	Average	Poor	Very Poor
3.3.23	Display of events on the screen	O	O	O	O	O
3.3.24	Display of the workflow path	O	O	O	O	O
3.3.25	Flow analysis	O	O	O	O	O
3.3.26	Interactive debugger	O	O	O	O	O
3.3.27	Line by line debugging	O	O	O	O	O
3.3.28	Interaction with model while running	O	O	O	O	O

Experimentation facilities

Criteria classified in this group evaluate the variety and characteristics of experimentation facilities. These facilities are required for improving the quality of simulation results and for speeding up the process of designing experiments and of the experimentation itself.

Table 9: Items for Experimentation facilities

		Very Good	Good	Average	Poor	Very Poor
3.4.1	Quality of experimental design facility	O	O	O	O	O
		Very High	High	Medium	Low	Very Low
3.4.2	Warm-up period	O	O	O	O	O
		Possible				Not Possible
3.4.3	Automatic batch run	O				O
3.4.4	Restart from non-empty state	O				O
3.4.5	Stepwise simulation run	O				O
3.4.6	Resource variability	O				O
		Provided				Not provided
3.4.7	Independent replications of experiments for multiple runs	O				O
3.4.8	Breakpoints	O				O
3.4.9	Accuracy check	O				O
3.4.10	Automatic determination of run length	O				O
3.4.11	Shift editor	O				O
3.4.12	Scheduled execution of scripts	O				O
3.4.13	Sensitivity analysis	O				O

Statistical facilities

Owing to the randomness that is present in the majority of simulation models, good statistical facilities are very important. Criteria included in this group examine the range and quality of statistical facilities provided by the simulation package.

Table 10: Items for Statistical facilities

		Very High	High	Medium	Low	Very Low
3.5.1	Quality of data analysis facility	O	O	O	O	O
		Very Large	Large	Medium	Small	Very Small
3.5.2	Number of theoretical statistical distributions	O	O	O	O	O
3.5.3	Number of different random number streams	O	O	O	O	O
		Provided				Not provided
3.5.4	Time dependent distributions	O				O
3.5.5	Ability to specify the random number seed	O				O
3.5.6	Random number generation by probability distributions	O				O
3.5.7	Distribution fitting	O				O
3.5.8	Goodness-of-fit tests	O				O
3.5.9	Output data analysis	O				O

CRITERIA FOR INPUT/OUTPUT ISSUES

Input and Output Capabilities

Criteria included in this group investigate how the user can present the data to the package and the type and quality of output reports provided by the package. These criteria evaluate, for example, whether the package has a menu-driven interface, whether static and dynamic output reports are provided, and how understandable these reports are.

Table 11: Items for Input/Output Capabilities

		Very Good	Good	Average	Poor	Very Poor
4.1.1	Static graphical output	O	O	O	O	O
4.1.2	Dynamic graphical output	O	O	O	O	O
4.1.3	Snapshot reports	O	O	O	O	O
4.1.4	Database maintenance for input/output	O	O	O	O	O
4.1.5	Dialogue boxes	O	O	O	O	O
4.1.6	Data Charting	O	O	O	O	O
4.1.7	Custom report generation	O	O	O	O	O
		Very High	High	Medium	Low	Very Low
4.1.8	Quality of output reports	O	O	O	O	O
4.1.9	Understandability of output reports	O	O	O	O	O
		Possible				Not Possible
4.1.10	Multiple inputs	O				O
4.1.11	Multiple outputs	O				O
4.1.12	Output export to excel	O				O
4.1.13	Printed report after each simulation run	O				O
4.1.14	Exchange data via internet	O				O
4.1.15	Task timeline report	O				O
4.1.16	Task execution report	O				O
4.1.17	Queue data collection report	O				O
		Provided				Not provided
4.1.18	Automatic rescaling of histograms and time series	O				O
4.1.19	Periodic output of simulation results	O				O
4.1.20	Writing reports to files	O				O
4.1.21	Summary reports for multiple run	O				O
4.1.22	Formattable result summary	O				O

Analysis Capabilities

Table 12: Items for Analysis Capabilities

4.2.1	Capability to do What-if Analysis	O Yes	O No
4.2.2	Conclusion-making support	O Provided	O Not provided
4.2.3	Optimization	O Provided	O Not provided

COMPARISON AND RATING OF THE EVALUATED SIMULATION SOFTWARES

This section provides a comparison of the evaluated simulation softwares. Information presented here is collected from various simulation software developer companies.

In order to compare the evaluated simulation softwares, a rating of these has been established. This rating is based on an analysis of the simulation softwares being evaluated. As such it should be considered as a relative measure of quality of these softwares from the perspective of groups of criteria rather than as an absolute value.

Methodology to calculate Rating for various groups of features

There are total 12 groups of features i.e. coding aspects, software compatibility, user support, general features, modeling assistance, visual aspects, efficiency, testability, experimentation facilities, statistical facilities, input and output capabilities, analysis capabilities. The value (out of 10) of these groups of features is calculated for the four simulation softwares under consideration.

$$\text{Evaluated Value} = \frac{\text{Calculated Value} \times 10}{\text{Maximum Value}}$$

where Maximum Value = Sum of highest possible values that can be selected in a particular group of features

and Calculated Value = Sum of actual values selected in a particular group of features

Table 13: Scaling Values

0	Not Provided, Not Possible, No	1	Provided, Possible, Yes
2	Very Low, Very Poor, Very Small, Very Rare	3	Low, Poor, Small, Rare
4	Average, Medium, Moderate	5	Easy, Large, Good, High
6	Very Easy, Very Large, Very Good, Very High		

For Ex.: If we take the case of Coding Aspects,

Maximum Value = 6+6+6+6 = 6+6+6+1+1+1+1+1 = 42

Table 14 shows a proposed rating for the simulation softwares being evaluated, in terms of the general quality of features within particular groups of criteria. The rating interval used in this assessment is similar to the one proposed by Ekere and Hannam (1989). The general quality of softwares with respect to particular groups of criteria is rated from 1 to 10, where 1 represents very poor quality or absence of the features within particular groups of criteria, whilst grade 10 represents excellent quality. Accordingly, we propose that 5 is taken to be a 'nominal acceptance level', or NAL for short. The grades for a certain group of criteria that are above the NAL indicate that a package is performing adequately, whereas those below signify the opposite.

Table 14: Comparison of Evaluated Softwares in terms of groups of criteria

Feature Groups	ProModel	AutoMod	HyperMesh	ProcessModel
Coding Aspects	9	7	7	10
Compatibility	5	4	7	7
User-Support	6	7	10	10
General Features	7	9	7	7
Modeling Assistance	10	5	7	8
Visual Aspects	9	9	6	6
Efficiency	8	6	10	7
Testability	7	9	7	7
Experimentation	6	6	6	5
Statistical	7	6	6	7
Input/Output	7	6	6	7
Analysis	10	10	7	10

Whilst the NAL is clearly subjective, it does provide a level against which the relative performance of a package can be measured and reflected on. Since evaluation cannot be entirely objective, this qualitative measure of performance, the NAL, does provide a relative measure. However, clearly any particular grade is merely a 'qualitative' number, and the rules of arithmetic can only be applied with caution and with caveats.

With respect to ***coding aspects,*** ProcessModel rated the highest and ProModel has been graded higher than AutoMod and HyperMesh. However, none of them have achieved a rating below NAL.

The quality of features with regard to ***software compatibility*** is above the NAL for HyperMesh and ProcessModel, but not very high. ProModel acquired rating equal to NAL and AutoMod is even below that**.** Whilst all simulators under consideration enable integration with word processors and spreadsheet packages, HyperMesh and ProcessModel are slightly better ranked, because they can be linked with data bases and statistical packages also.

With regard to ***user support****,* HyperMesh and ProcessModel *ar*e rated the highest. The suppliers of both simulators provide a high level of support in the form of user group meetings, help-lines, fairly good documentation, training course, consultancy etc. The next in sequence are AutoMod and ProModel, both with above the NAL levels but no so extensive as is the case for the other two simulation softwares.

All softwares are rated quite high regarding ***general features.*** They are all data driven and manufacturing oriented. AutoMod is ranked highest and rests all are equally ranked.

Modeling assistance is better ranked for ProModel and ProcessModel than for the other two. These simulator has a reasonably good prompting, it enables model and data separation and an automatic editing of data is provided. Similarly, HyperMesh enables model and data separation, the automatic editing of data, but there is no proper prompting facility. AutoMod is raked at NAL for this group of criteria.

Visual aspects are rated highest for ProModel and Automod, which satisfies the majority of criteria within this group. For example, ProModel has full animation, with user-friendly icons and screen editors. The animation layout can be created before or after model development. An icon library is provided and it is possible to change the icons during the simulation according to the specified logic. HyperMesh and ProcessModel are just above NAL. Animation layout development is less flexible for these simulators. There is no screen editor to enhance graphical representations of simulation models nor is it possible to change the icons for a particular entity during simulation.

The ***efficiency*** related rating of the simulators also shows good quality but significant variation. HyperMesh is rated the highest mainly because of its high robustness and interactivity. In addition, this simulator has good adaptability, and models are easy to edit. ProModel also has good features related to adaptability and interactivity, whilst ProcessModel is better regarding robustness, and has an automatic model building feature. AutoMod is lacking robustness, it is not flexible and adaptable, but it has a short time scale for model building.

With regard to ***testability****,* AutoMod outperforms all the other simulators. It is rated quite high because it has many features that facilitate model verification, such as error messages, animation, step function, display of variables and function values, and list and trace files. Rest all the simulation softwares are equally rated for testability.

ProModel, AutoMod and HyperMesh are equally rated slightly above the NAL regarding ***experimentation*** facilities, providing features such as facilities for multiple runs, accuracy checks and the automatic determination of run length or a facility for the automatic testing of 'what if’ scenarios. ProcessModel ranked exactly at NAL value.

It is judged that ProcessModel and ProModel has the same ***statistical*** *facilities* in comparison to other two softwares. These not only provides features such as a number of theoretical statistical distributions and antithetic sampling, they also enables distribution fitting and goodness-of-fit tests. AutoMod and HyperMesh are lower ranked than ProcessModel and ProModel but ranked slightly higher than NAL because of their small number of theoretical statistical distributions, and lack of antithetic sampling and distribution fitting.

The ratings for ***Input/Output*** features are same as that for statistical facilities for all the four simulators and follow the same order.

The ***analysis*** facility is rated equally for ProModel, AutoMod and ProcessModel. HyperMesh has lower rank as compared to rest of the three softwares.

SUMMARY AND CONCLUSIONS

This paper provides an evaluation of four manufacturing simulation softwares. During the evaluation the aim was to generally perceive basic features of each simulator. Specific features are probably going to change and be added to with new releases of the simulators under consideration.

A comparison of the evaluated simulators is provided. The general quality of each group of criteria was ranked for each simulator. This revealed that although all simulators belong to the same type of simulation software, there is a variety of differences between them. In addition, none of the simulators satisfies all criteria to the highest level, and none is equally good for all purposes. Although some simulators are more comprehensive and flexible than others, a simulator that can suit any manufacturing problem does not exist.

The fact that the selection of a piece of simulation software is a matter of compromise between many factors is substantiated by this research. One of the most important factors that determined which software is more suitable than others is its intended purposes. Other factors to consider are financial constraints and subjective factors such as individual preferences and previous experience in using simulation software.

Appendix A: Versions of the Simulation Softwares being evaluated

S. No.	Simulation Software	Version
1	ProModel	7.5
2	AutoMod	12.2
3	HyperMesh	9.0
4	ProcessModel	5.3

ACKNOWLEDGEMENTS

Much of the information in this paper was derived from extended discussions with software vendor personnel. The authors gratefully acknowledge the time investments in this project that were generously provided by Mr. Jai Shankar from ProModel Corporation, Mr. Vikram Ramnath and Mr. Jerry Banks from Larvik Engineering Consultants Pvt. Ltd., Mr. Rajat Tikku from Altair Corporation and Mr. Travis Healy from ProcessModel, Inc.

REFERENCES

1. Abed, S.Y., Barta, T.A. and McRoberts, K.L. 1985a. A Qualitative Comparison of Three Simulation Languages: GPSS/H, SLAM, SIMSCRIPT. Computers and Industrial Engineering, 9: 136-144.
2. Atkins, M.S. 1980. A Comparison of SIMULA and GPSS for Simulating Sparse Traffic. Simulation, 34: 93-98.
3. Banks, J., Aviles, E., McLaughlin, J.R. and Yuan, R.C. 1991. The Simulator: New Member of the Simulation Family. Interfaces, 21 (1): 76-86.

4. Bollino, A. 1988. Study and Realisation of Manufacturing Scheduler using FACTOR. Proceedings of the 4th International Conference on Simulation in Manufacturing, London. 9-20.

5. Carrie, A. 1988. Simulation of Manufacturing Systems. UK: Wiley.

6. Ekere, N. N. and Hannam, R.G. 1989. An Evaluation of Approaches to Modeling and Simulating Manufacturing Systems. International Journal of Production Research, 27 (4): 599-611.

7. Fan, I.S. and Sackett, P.J. 1988. A PROLOG Simulator for Interactive Manufacturing Systems Control. Simulation, 50(6): 239-247.

8. Grant, J.W. and Weiner, S.A. 1986. Factors to Consider in Choosing a Graphical Animated Simulation System. Industrial Engineer, 40: 65-68.

9. Hlupic, V. and Paul, R.J. 1999. Guidelines for Selection of Manufacturing Simulation Software. IIE Transactions, 31: 21-29.

10. Law, A.M. and Haider, S.W. 1989. Selecting Simulation Software for Manufacturing Applications: Practical Guidelines and Software Survey. Industrial Engineer, 34: 33-46.

11. Law, A.M. and Kelton, W.D. 1991. Simulation Modelling and Analysis, 2nd edn, Singapore; McGraw-Hill.

12. Mackulak, G.T. and Savory, P.A. 1994. Ascertaining important features for industrial simulation environments, Simulation, 211–221.

13. Scher, J.M. 1978. Structural and Performance Comparison between SIMSCRIPT 11.5 and GPSS V. Proceedings of the 9th Annual Conference on Modeling and Simulation, Pittsburgh, 1267-1272.

14. Seila, A.F., Ceric, V. and Tadikamalla, P. 2003. Applied Simulation Modeling, Thomson Learning, Australia: Thomson Learning.

15. Taramans, S.R. 1986. An Interactive Graphic Simulation Model for Scheduling the factory of the Future. Proceedings of the AUTOFACT '86 Conference, Detroit. 4-31.

16. Tewoldeberhan, T.W., Verbraeck, A., Valentin, E. and Bardonnet, G. 2002. An Evaluation and Selection Methodology for Discrete-Event Simulation Software. Proceedings of the 2002 Winter Simulation Conference, 67- 75.