Proceedings of the 1991 Winter Simulation Conference
Barry L. Nelson, W. David Kelton, Gordon M. Clark (eds.)

AVOIDING THE PROBLEMS AND PITFALLS IN SIMULATION

Randall P. Sadowski

Systems Modeling Corporation
504 Beaver Street
Sewickley, PA 15143

ABSTRACT

This tutorial will present an approach to conducting a simulation project that will aid in avoiding many common problems and pitfalls. The presentation will provide recommendations on how to scope the project, develop a functional specification, formulate and construct the model, verify and validate, collect data, document the work and perform the required analysis. The intent is to provide the novice simulation modeler with proven techniques for conducting a successful simulation project. A variety of case studies will be presented during the tutorial to illustrate both the right and wrong ways to conduct a project.

1 INTRODUCTION

Although there have been numerous descriptions of the simulation process put forth, there is general agreement on the basic steps required to complete a simulation project. One must first identify and define the problem to be solved and then must determine whether simulation is an appropriate tool. The process then continues with development of a clear, concise statement of the objectives. Having defined the objectives, the modeler can begin to formulate the model, then proceed to the model construction stage. At some point in this process the modeler must define the data required and initiate the collection of the necessary data. Having completed the model construction, the modeler must then verify and validate the model.

The modeler can then begin to experiment with the model. The experimentation stage provides insight to the possible behavior of the modeled system, but these results must be carefully interpreted in light of the stated objectives. Finally, conclusions must be drawn and implemented in order to complete the entire process.

Given this fairly rigid set of steps for the simulation process, it would seem to be easily implemented and almost always guarantee a success. However, anyone who has undertaken a simulation task has found that only rarely does the entire process proceed in a logical and orderly fashion. The simulation process is as much an art as it is a science. Teaching the simulation process is somewhat like teaching someone how to ride a bicycle. You can describe and sometimes demonstrate the process, but the new rider quickly learns that there is no substitute for experience. One wrong move or turn of the wheel and the bicycle reacts in a fashion totally different than anticipated; but in hindsight, the bicycle's reaction could have been predicted.

The purpose of this tutorial is to provide some instruction and insight into the simulation process. There will be no attempt made to define an ideal procedure for success, since it is not clear that one exists. The concentration instead will be on identifying some of the right and wrong ways to approach each phase of a simulation project.

2 DEFINING THE PROBLEM

The first step in any simulation process is to define the scope of the project. Typically a simulation is undertaken because an actual or potential problem is perceived. Understanding what this problem is can often make the simulation task much easier. One should be careful to define, as best as is possible, "the real problem" rather than a symptom of the problem. For example, the perceived problem may be the lack of throughput from a manufacturing system. If this is just a symptom of poor quality control or a difficulty in getting raw materials, a simulation model may not help to solve the underlying cause of the perceived problem.

An accurate definition of the problem can dictate the level of detail required in the model and may indicate specific areas where special care must be taken. The problem definition often leads directly to a statement of the objectives the resulting simulation model is to achieve. Although this may also seem to be rather straightforward, it can require a significant amount of time and effort and still may not result in a well-defined set of objectives if all parties do not share a clear view of the problem at hand.

Consider the following analogy. We have identified a problem: we need transportation to and from work. Our objective, then, is to purchase a car that will satisfy this need for transportation. The concept sounds simple, but the options range from purchasing an old, and hopefully reliable, used car costing several hundred dollars to investing in a fancy, high-performance sports car costing many times more.

At this point, we should examine the objective and determine if our solution utilizes the proper tool. In the case of our transportation problem, should a pass to the local bus system be purchased rather than a car? If we assume that the car is the right approach, then we must further examine the problem and refine our objectives in order to reduce the number of available options. We should identify what the conditions are for the drive and what impact they have on our choice of vehicle. For example, does the trip require air conditioning, room for other riders, 4-wheel drive for bad weather, etc? Having addressed these issues and further refined our problem and objectives, we might also consider the more cosmetic features. Do we really care about the color, white walls, radio, sun roof, etc?

Finally, after thoroughly defining the objectives, we set out to make our purchase only to find that we may have set forth objectives that are beyond our price range while at the same time noticing that there were several options we had simply overlooked. So we again refine our objectives until a compromise between budget and performance is reached.

The thought process for solving the problem of finding transportation to work is quite similar to that which one should go through in defining the objectives of a simulation project. As with the transportation example, the objectives you establish can have a great effect on the final simulation model. For starters, the level of detail incorporated in the simulation model can greatly affect the amount of effort required to create the model.

Let's consider the development of a model for a new manufacturing system. Consider three possible objectives. The first is to construct a model that can be used to evaluate candidate systems to determine if they will work. This type of model normally requires a minimum level of detail because you are only considering the functionality of several systems; it could be developed in a very short time period (e.g., days or weeks).

The second possible objective is that the model will be used to compare two systems and evaluate which is better. More detail is required in this case since you are going to compare the performance of one system to that of another. Here we are concerned that the relative performance of the two systems is accurately measured by the simulation; the actual values predicted are not as important. For example, if the key performance measure is throughput, then you are attempting to establish which system will provide the largest throughput, not necessarily to determine the exact throughput of either system.

The third possible objective is to accurately predict the performance of the selected system. In the previous case, it was sufficient to determine that one system would provide more throughput than the other. For this last objective, though, we must provide a model that will yield an accurate estimate of that throughput. These types of models require the greatest amount of detail as they must be capable of replicating the activity of the real or proposed system very closely in order to provide a reliable prediction of the systems performance.

The construction of a simulation model with too little detail will result in information that may not be accurate enough to achieve the real goal. If the model contains too much detail, it requires more effort to create, longer run times and is more likely to contain errors. If you must err, you should include more detail rather than less. Ideally an accurate definition of the problem and project objectives will prevent one from straying too far in either direction.

The natural tendency of the novice modeler is to include too much detail, whereas the more experienced modeler tends toward greater abstraction. One technique that helps determine what needs to be included is to decompose the perceived model into smaller components such as buffers, operator logic, job priorities, job release strategy, etc. Then examine each component and ask the following question: if this component is not included, will it have a significant effect on the key performance measures? For example, what if the parameter to be evaluated is the number of fork trucks used for material handling in a large system. If there are ample trucks and the materials tend to spend large amounts of time in buffers, then one could probably model this activity as a simple delay, approximating the amount of time typically required for a load to be picked up and moved by a fork truck. On the other hand, if the system has a limited number of trucks with a just-in-time control system, the fork trucks could well be the main bottleneck in the system and should be modeled explicitly.

3 THE FUNCTIONAL SPECIFICATION

One method that will assist in developing an accurate definition of the system problem and simulation objectives is to initiate the project with a complete functional specification. This can be a very time-consuming effort, but it does not necessarily increase the project duration; in most cases, it will shorten the process. Developing a functional specification requires that the modeler completely define all inputs to and outputs from the simulation, describe each of the simulation components, define all control logic for the model, and present at least a short discussion of possible extensions under consideration for the future. It often details the analysis to be performed once the model is completed. If the simulation project is of short duration with very little detail required, this specification may be a single hand-written sheet. For a large, complex model with numerous options and great detail, this document can be quite large.

The first assumption behind the need for a functional specification is that there are always at least two parties involved in a simulation project: the modeler/analyst and the user/customer. The first needs to know what the second expects, and the second needs to know what the first will deliver. This is obviously the case if an outside consultant is hired, but it also applies in most other cases. The simulation may be performed by a corporate group, an in-house group, or even an individual in the same department. Even if the simulation is performed by the individual making the final decision, development of a functional specification is recommended so the scope, objectives and data requirements of the model are clearly identified before the modeler is immersed in the actual modeling process. There might also be teams representing each party. In any case, it is fair to assume that the responsibility of the modeler/analyst is to provide expertise and a final simulation model that will accurately answer the questions put forth. The responsibility of the user/customer is to provide the system description, accurate data and control logic; and support the modeler in achieving the defined goals. Ultimately, since the user/customer will use the results to make decisions and to judge the wisdom of future use of simulation, a well conceived and executed model benefits both.

The second assumption underlying development of a functional specification is that it is created jointly after the project has been initiated. If the user/customer develops the specification without input from the modeler, then key elements might be overlooked. After all, the modeler should be the expert on simulation for this project. Remember that an expert is defined as one who knows the most about the subject. Take advantage of this expertise, but don't always assume the modeler is correct.

This specification development process often becomes a series of negotiation sessions between the two parties. The modeler might be presented with several new requests and retire to evaluate the impact on the ultimate simulation model. The user might be informed of implications in either development time or accuracy of results and retire to consider the options. The tendencies are for the modeler to resist any additional work and for the user to want additional capabilities. The overall goal of this process is to specify completely the requirements of the tool that will answer the agreed-upon questions, not to protect one's turf.

Depending on the model complexity, the urgency for completion and the number of participants, this process might be completed in one session or it could require a few months. Although, one to two weeks is typically required. This may appear to be an excessive amount of time, but it serves many purposes. Both parties are required, at an early stage, to consider the final product. It can also serve as an educational process for both the modeler and the user. It provides the modeler with the time to truly understand the system to be modeled. Conversely, the user is given a better understanding of simulation and often gains a more through understanding of how the system really works.

There are other advantages to development of a functional specification prior to model development. The user is forced to define all the logic and data at an early stage. If the data does not exist, development of the functional specification provides an early warning so that a data collection procedure can be developed and initiated. If extensive data collection is necessary, the collection time may exceed the time to develop the simulation model.

Development of the functional specification also prevents the user from periodically changing the scope or objectives of the project based on current events. Often the user has not thoroughly considered the complexity of the problem and assumes that the ultimate model will have many more capabilities than the modeler assumed.

As the specification is being developed, the modeler frequently begins to organize mentally the structure of the simulation model, identifying what system components will be modeled and the specific constructs to be employed. Often this is prompted by requests from the user for the inclusion of additional detail; the modeler must determine how the new issues will be incorporated into the final simulation model. Development of the functional specification also allows time to organize the data structure to be employed in the simulation. The novice modeler often starts the modeling process by developing a simulation model of the physical system, considering the data requirements only as they appear. Generally, though, the complete data structure should be defined before the actual construction of the model begins.

Often the amount and type of data can influence the specific constructs to be used. For example, should individual job characteristics be carried with each entity or placed in a data array and only accessed when needed? Having the data structure defined can also help prevent the "why did I do that?" part of modeling. Most modelers tend to define the data structure in an abstract manner that means it can be changed easily. Once the model construction begins, modelers tend to develop an attitude of ownership that discourages change.

The discussions that take place between the modeler and the user also can help to clarify subtle points and assure that both are interpreting the modeling effort in the same manner. It can also help identify problems at an early stage. For example, a simulation was to be developed of a computer-controlled flexible manufacturing system that had a data collection system built into the central controller. Periodically, the system archived this data, and it was easily available for statistical analysis. It was assumed that the distributions for process, load/unload, travel, inspection, etc., could be extracted from these data files for use by the simulation study. However, during the development of the specification, it was found that almost all the data being collected was unusable. For example, the collected data for loading operations was based on the time from when a request for load was made to when it was reported to have been completed. Thus, it included waiting time, coffee breaks, and other interruptions; however, these were to be represented separately in the model. Since this was uncovered at the beginning of the project rather than well into model development, the user had time to obtain the required data without seriously delaying the project.

Finally, the completed functional specification provides a contractual agreement between the two parties that completely defines the project, although changes often are made as the model development proceeds.

4 MODEL FORMULATION AND CONSTRUCTION

As discussed earlier, most novice modelers simply start coding a model based on the physical system. This can create problems at later stages because data requirements may have been overlooked or control logic may not have been considered. It can sometimes be very difficult to add or modify logic in an almost completed model if it was never contemplated. Thus, it is helpful to separate the actual modeling process into two stages: formulation and construction.

The formulation stage should be a mental activity that attempts to develop an overall strategy for the simulation and to evaluate alternative modeling approaches. This is the time when the modeler should lean back, prop his feet up on the desk, and endeavor to determine the most effective way to construct the model. It is during this stage that fellow modelers should be consulted. Someone not directly involved in the project can often help to identify unique ways to replicate the real system effectively. Ideally, several different approaches should be considered and evaluated before the model construction stage begins.

Experience has shown that if a functional specification is developed, much of the model formulation is completed by the time the specification is accepted. The model construction phase then becomes an integration of the data structure and the modeling framework into a working simulation model.

5 VERIFICATION AND VALIDATION

Almost every paper dealing with the topic of simulation at least touches on the subjects of verification and validation. However, there are no step-by-step procedures available to guide the modeler in performing these important tasks. Verification -- ensuring that the model behaves in the way it was intended -- is the easier of the two tasks. Even so, it can be difficult to conduct a complete verification of a complex model.

If the model is constructed in components, then each component should be verified separately as it is completed. As groups of these components are put together, it is wise also to verify the model at this point. The final verification has to be performed with the completed model. The difficulty is in contriving all of the possible situations that could occur when the completed model is subjected to experimentation. Modelers who have animation capabilities find the verification task much easier. It allows the modeler to display all the key elements on the screen to observe the important interactions within the model. The alternative is to watch a trace of the system activities and create a paper scoreboard in order to capture the interactions.

During the verification stage, one should be sure to check the function of the model under extreme conditions. What happens when the system becomes overloaded or when multiple failures occur? Consider using deterministic times that will allow you to predict more easily the outcome for simple simulation runs. Often the modeler tests the developing model at various stages under the same set of conditions. In the interest of time, these are normally very short runs designed as quick checks of newly incorporated code. Occasionally, one should make extended runs to assure that the randomness in the model does not create circumstances that were not previously considered.

Validation -- ensuring that the model behaves the same as the real system -- can be difficult and, in some cases, impossible to perform. If the system currently exists, then some kind of comparison can be made to ensure that the model represents the real world. If the system does not exist, but similar ones do, then the simulation results can be compared to the similar system and at least a partial validation can be performed. If there is no real system to compare with the simulation, then validation cannot be performed. If this is the case, then it is recommended that the extra effort be devoted to the verification and that those who are familiar with the system design be closely involved with the simulation effort.

6 DEVELOPING THE MODEL CONTROL LOGIC

As discussed earlier, the control logic that dictates how decisions are to be made during a simulation run should be completely defined during the development of the functional specification. This is fairly easy if the real-world system is completely automated and all decisions are made by a controlling computer. However, most systems are not automated to the extent that there is no human intervention that impacts the performance of the system. Trying to capture the logic employed by human operators or controllers can be extremely challenging. Even if documented procedures exist, they are often outdated or not rigorously followed. Such documentation often yields a set of rules that do not really capture the true operating conditions. There are always numerous undocumented exceptions that tend to become the rule.

This problem is further complicated by the fact that the modeler is frequently separated from the real shop floor decision maker by at least one layer of management. Only rarely is the user/customer the individual who controls these day-to-day decisions. Experience would tend to imply that activity on the shop floor is quite different than that perceived by management. Obtaining the actual logic which controls the system requires that someone document the normal control policies. In most simulation studies, there are many people involved in this task. It is necessary to get them together to define and agree upon what really happens "out there."

At the specification stage there is no easy way to assure that the acquired control logic represents reality. The modeler should assume that it is reasonably close, but will probably have to be changed. The best way to uncover the actual operating procedures is during the verification process. Involving the individuals who make these day-to-day decisions in this process normally yields at least a set of rules that will allow the simulation model to approximate the real system control logic. This verification is best performed with the aid of an animation. Using a pictorial model of the system for this purpose is far better than flow charts or tables of data. It is not uncommon to see the shop floor personnel become extremely interested and cooperative in providing the needed information when viewing an animation of "their" system.

Sometimes the modeler discovers that the complexity of the operational rules prevents their implementation in an exact form. If the actual rules require simplification, the assumptions should be verified with the shop floor personnel before the simulation is used for decision making. Often, they can assist in arriving at control procedures that accurately represent the majority of the decisions made in the real-world system.

7 DATA REQUIREMENTS

Acquiring accurate data in the right form often is the modeler's most formidable task. Typically there are two extremes when it comes to data requirements. The most frequent case is the lack of data. If possible, the modeler should define and initiate a data collection procedure early in the simulation project so the required information is available when needed. Occasionally, one finds that it is not feasible to collect good data due to the amount required or cost limitations. If this is the case, the people at the source of the data should be interviewed for approximate values. However, the modeler should be cautious of the quality of this information as the memory of these individuals tends to be very short-termed. The most recent past will be weighted heavily in the estimates. One should attempt to address questions to the long-term system performance rather than to recent events.

Given that you can only obtain rough estimates of the data, you must then fit these estimates to distributions for inclusion in the simulation model. There is no magic set of rules to guide you in this process. In most systems, the time between failures follows an exponential distribution. This is the only statement that can normally be made without fear of being wrong. Even here one must be careful: is it the "time" between failure or the "number" of cycles between failure? It seems logical that a machine used twenty percent of the time will fail less often than the same machine used ninety percent of the time. Failures are often keyed to the work load as they are dependent on tool wear or similar factors that will vary greatly if the utilization of the machine changes drastically.

Process times, load/unload times, repair times and other similar data can vary greatly in the form of distribution which they follow. Thus, the exact parameters required are dependent on the chosen distribution. You are often forced to adjust your choice of distributions based on how the information on the data is provided. You should recognize that the way this information is given is very dependent on how the initial questions are posed. If you request only the minimum and maximum values, you will probably assume the distribution is uniform or maybe is a symmetrical triangular distribution. Be cautious to avoid the inclusion of catastrophic values in the estimates. One should stress that the values you want are those for normal operating conditions.

If the one-of-a-kind catastrophic data point is incorporated into the data to be used by the simulation, it can distort the simulation results. For example, consider the case where delivery times are typically between three and five days, but six years ago a delivery took nine weeks. First of all, we should not be considering situations where a combination of a snowstorm, truck and rail strike, lost order, shortage of raw materials and quality problems caused a nine-week delivery. Secondly, specifying a distribution to capture accurately this phenomenon along with the normal delivery times would be rather difficult. Finally, the simulation runs would have to be extremely long in order to be sure that a representative sample of the catastrophic events had occurred.

One typically finds that it is easier to obtain information on the average time than it is about the variation of the distribution. Unfortunately, it is the variation that often has the greatest affect on the performance of the system. In attempting to define this variation, one should be careful to identify the cause. Is the variation in a processing time due to uncontrollable events in process or is it caused by the need for an operator who is monitoring several processes. If the latter is true and the operator activity is being modeled explicitly, then the processing time may well be a constant. In fact, the cycle times for most automated equipment are constants. The cause of most variation in processing times is due to the variability caused by an operator, not the process itself.

Repair times are frequently operator-dependent and multi-modal. This is caused by the different types of failures that typically occur. For example, in an automated circuit board facility, there might be three distinct types of repairs. The shortest could be caused by a jammed board that is repaired by an operator moving to the machine and "banging" the board holder. The second repair type might require several minutes while an operator disassembles the feeder mechanism to clear the board. The last type may require a maintenance operator and may take several hours.

In dealing with failure and repair times, one should recognize that there is a direct relationship between the mean time between failure, the mean time to repair, and the fraction of up or available time. Given any two, the third can be calculated. The most critical of the three is the distribution of the repair time, as a repair activity implies that the machine is in a non-productive mode and potentially creating starving and/or blocking of other equipment in the system. The greater the variation of the repair time, the more likely the impact on the system performance. For example, if a machine is unavailable for production for a duration of approximately 6 minutes every hour, the long-term impact is far less than if it were unavailable for one year out of every ten. In both cases, the average up time is ninety percent.

If only estimates of data are available, then a two-step approach is recommended. The first step is to obtain the data and incorporate it into the simulation. Once the simulation has been verified, it should be used to test the sensitivity of the performance of the system to the data estimates. Sensitivity to both the mean and variance should be checked. If large variations in the estimated data have little or no impact on the results of the simulation, then there is little incentive to spend additional time and money to obtain better values. If slight variations in the estimated data have a significant effect on the results, then it may be wise to initiate a data collection procedure or at least consider this sensitivity in the analysis stage.

If the system to be simulated is a new design for which no data exists, then the estimates are likely to come from the equipment vendors. One should expect that these values would tend to be optimistic. If a sensitivity analysis indicates that certain pieces of equipment can easily become bottlenecks, the simulation could be used to determine what the operation times or failure rates must be in order to meet the stated goals of the system. These can then be used in developing the equipment specifications. If monetary incentives are tied to these specifications and included in the purchase contracts, the likelihood of the system performing as designed is greatly increased.

If adequate data exists, either from historical records or from shop floor collection systems, then the use of a statistical package to fit distributions is highly recommended, at least for the critical data. Prior to devoting a large amount of time to this task, one should make sure that the data is truly representative of what is required by the simulation. There is really very little difference between a statistically fitted distribution of bad data and unreliable estimates where no data exists. Even worse, the analyst may confidently draw the wrong conclusion with bad data that appears to be good.

8 ANALYSIS

The process of analyzing results from a simulation model can, and often should, bring a look of fear to the eyes of the analyst. He is about to use the model that he created to draw conclusions that could lead to major changes and expenditures. There is an enormous amount of literature available to the analyst, but much of it is not relevant and most of it is hard to understand. Remember that the basic issues to be faced in any analysis are of a statistical nature. The first recommendation is to acquire a background in statistics, or even better, acquire a statistician.

There are generally two questions that must be answered prior to the analysis stage: how long a simulation run is required to obtain significant information, and is there a difference among sets of results to be compared? The first question is often the most difficult to answer as it requires the analyst to define steady-state conditions and minimum sample size.

Most simulation models of complex systems start with the system empty, which requires that the statistics be cleared after a reasonable time in order to eliminate the initial bias due to the empty system. Once this problem has been overcome, a decision must be made as to how long the remaining run must be in order to avoid misleading results. There are statistical procedures that can be employed to aid in answering these questions;but if time or expertise does not permit a statistical analysis, the use of plots on key performance measures can provide much insight. If time is an issue, there should be a balance between the length of run and the number of runs. When in doubt, make longer runs.

The nature of the analysis should have been considered in the definition of the problem and statement of the simulation objectives. The type of analysis required should be consistent with the types of decisions to be made. If the model was designed to predict the performance of a system, then the statistical issues can be very important and the analysis will generally require much longer runs than if a comparative analysis is being performed.

The appearance of animation has provided an additional analysis tool, but it can easily be misused, as with any tool. Animation is a excellent tool for bottleneck analysis and for identifying the cause of strange system behavior. However, it should never be used as the sole analysis tool. Using only animation for making important system decisions is somewhat equivalent to walking out and observing the shop floor for an hour and then making major equipment decisions based on the observations. Most systems are very dynamic and change dramatically over time. What appears to be a problem today may well have been caused by an equipment failure several days ago. A careful statistical analysis should never be excused away by observation of an animation.

9 THE NEED FOR DOCUMENTATION

Model documentation is a task that almost no one enjoys. It can consume a considerable amount of time and often seems unimportant at the time it is developed. However, documentation can be a critical task and there are very few cases where it could be considered unnecessary. Rarely does the modeling task result in a small throwaway model that requires no documentation.

At a minimum, the model documentation should describe the data structure, the key elements of the model, the general flow logic, and all variables, queues, etc. Sufficient documentation might consist simply of comments throughout the model. The depth of documentation depends on the user, the expected life of the model, and the complexity of the model. Never underestimate the time required to document adequately a simulation model. It normally requires far more time than one would think.

There are several reasons why documentation is important. If the simulation model is to be utilized over a long period of time, then changes will undoubtedly have to be made to logic and/or data. The documentation provides the means to understand the approach at a later date, often by someone different than the original modeler. If the model is extremely large and complex, the documentation may be very useful in the later stages of development. It can provide a quick reference back to sections completed weeks or months ago and prevent duplicate use of variables or names. If more than one individual is working on the model development, then the documentation can provide the link between them.

As is often the case, it seems that if you do detailed documentation, then it is never needed. However, if there is no documentation, then the model always comes back to haunt you. It seems that the poorly documented models are always the ones which are resurrected and require modification.

Many simulations are developed for users who desire to have the ability to exercise the models after completion. In this case, some form of documentation is critical. Often the greatest need for documentation is in what values the user can change. One method that can simplify the documentation task is to remove the specification of these variables from the simulation model and place them in some form of data configuration file. Upon execution, the simulation reads this data file and assigns the values indicated at the start of each run. This prevents the novice user from changing values that should not be altered and provides a readable file of the values. It also prevents the novice user from changing a value that causes the simulation to terminate due to an error, or worse yet, yield incorrect results. The use of a data configuration file can also be used to allow changes in the model logic, provided these have been included. It can also greatly simplify the development of a users' manual and decrease the time to train the user.

10 CONCLUDING REMARKS

The art of simulation is best learned through experience. With the completion of each new project, the modeler becomes more adept at avoiding the problems and pitfalls, many of which may be specific to their environment. Although mistakes will be made, hopefully the simulation tool will become a valuable addition to the analyst's set of capabilities.

ACKNOWLEDGEMENTS

The major portions of this paper have been reprinted from a paper presented at the 1989 Winter Simulation Conference, Sadowski (1989).

REFERENCES

Balci, O. 1990. Guidelines for Successful Simulation Studies, In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R.P. Sadowski, and R.E. Nance, New Orleans, LA, 25-32.

Johnson, M.E. and J.P. Poorte. 1988. A Hierarchical Approach to Computer Animation in Simulation Modeling. *Simulation* 50(1): 30-36.

Sadowski, R.P. 1989. The Simulation Process: Avoiding the Problems and Pitfalls. In *Proceedings of the 1989 Winter Simulation Conference*, eds. E.A. MacNair, K.J. Musselman, and P. Heidelberger, Washington, DC, 72-79.

Sargent, R.G. 1987. A Tutorial on Validation and Verification of Simulation Models. In *Proceedings of the 1988 Winter Simulation Conference*, eds. M.A. Abrams, P.L. Haigh, and J.C. Comfort, San Diego, California, 33-39.

Seila, A.F. 1990. Output Analysis for Simulation, In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R.P. Sadowski, and R.E. Nance, New Orleans, LA, 49-53.

Thesen, A. and L.E. Travis. 1990. Introduction to Simulation, In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R.P. Sadowski, and R.E. Nance, New Orleans, LA, 14-21.

AUTHOR'S BIOGRAPHY

Dr. Sadowski is currently vice president of Systems Modeling Corporation in charge of consulting and user-education services. He was previously on the faculty at Purdue University and at the University of Massachusetts. He received his bachelors and masters degrees from Ohio University and his Ph.D. from Purdue. Dr. Sadowski's research interests are in manufacturing and production systems with emphasis on modeling, control, and applied scheduling. He has authored numerous technical articles and papers, served as chairman of the Third International Conference of Production Research, and was the General Chair of the 1990 Winter Simulation Conference. He is a senior member of the Institute of Industrial Engineers, served as editor of a two-year series on Computer Integrated Manufacturing Systems for IE Magazine which received the 1987 IIE outstanding publication award and was a co-editor of Computer Integrated Manufacturing Systems: Selected Readings, published by IIE.