A UNIFORM METHODOLOGY FOR DISCRETE-EVENT AND ROBOTIC SIMULATION

Eric S. Grajo
Ali Gunal, Ph.D.
Dev Sathyadev, Ph.D.
Onur M. Ulgen, Ph.D.

Production Modeling Corporation
3 Parklane Boulevard, Suite 910W
Dearborn, MI 48126
(313) 441-4460

Abstract

Simulation is a very powerful and flexible tool in the design and analysis of many different types of systems. Discrete-event and kinematic simulation are the most commonly used types of this tool in manufacturing industry. Although it is a very powerful and flexible means for designing and analyzing many different types of systems, without a systematic approach, it can be inefficient, expensive, and even misleading. Discrete-event and kinematic simulation have different application areas. However, a close examination of the steps one has to take through real-life applications shows that a common methodology can be used to successfully utilize simulation in all application areas. This paper presents such a uniform methodology that is comprised of eight major phases. Each phase consists of several steps that are discussed in detail using some case studies. It is found that depending on the type of simulation (discrete vs. kinematic,) the objectives of the study, and the detail level of the problem considered, each of the steps of the methodology may be given different emphasis. However, adherence to this common methodology will ensure the proper usage of this powerful technique and will enable the users to gather better payoffs from the investment on simulation.

Introduction

Computer simulation is a technique that allows building of and experimenting with a model of a real system on a computer. Although there are several different types of simulation, the focus of this paper is on discrete-event and robotic (or more generally kinematic) simulation as applied to manufacturing systems.

The discrete-event simulation technique is used for systems where the state changes occur when events happen at discrete points in time. For example, release of an order to shop floor, breakdown of a machine, completion of a machine cycle are events that change the states of a manufacturing system. This kind of simulation is more appropriate for systems with multiple entity types and event types. A job shop, an assembly line, a paint shop in a vehicle manufacturing plant are examples of such systems. The types of entities that can be seen in a job shop are orders, machines, and workers. Clearly, some of the entities would be considered as permanent, e.g. a machine, and some others would be temporary, e.g. a part, with respect to their lifetime in the model. Analysis of the throughput capability of the system, determination of the bottleneck operations, evaluation of different material handling systems are typical applications of discrete-event simulation models.

Kinematic simulation is a technique for simulating a system whose state changes continuously based on the motion(s) of one or more kinematic devices. Typical examples of such systems are robotic workcells and machines with several moving components. These structures are typically parts of larger manufacturing systems. Analyses that are made via this kind of simulation models are generally in the form of evaluating the movement capabilities of one or more devices. Verifying that a robot can perform its tasks free of collisions, finding an optimal motion path, and determining an optimal placement for a robot and other fixtures are typical examples of such studies.

Although these two types of simulation have different application areas, a common path can be easily observed in their use. Whether discrete-event or robotic, the overall process to be used in a simulation study starts with a problem definition. Once a problem is defined and simulation is selected as an analysis tool for solving it, data about the real system must be gathered. After collecting sufficient information, a model of the real system is built. Analyses with a simulation model generally involve extensive experimentation using various what-if scenarios. Based on such experimentation, conclusions are drawn and a report is written to document the findings of the study.

Learning to use a simulation tool has been made relatively easy by highly interactive graphical user interfaces. Also, the overall process described above can be seen as straightforward by the virtue of the fact that there seems to be only few steps involved. However, experience shows that as much as it is an exciting and easy to use tool, simulation can be inefficient, expensive, and even misleading when it is not used properly. The purpose of this paper is to introduce a common methodology that can be used for applying either kind of simulation properly. The methodology consists of eight phases. There are also several steps identified in each phase. This methodology have been developed and tested by Production Modeling Corporation over the last fifteen years in discrete-event simulation arena. It also has been successfully adopted and used in robotic simulation over the past three years.

The next section of the paper introduces this common methodology with a discussion of some important points regarding the differences between discrete-event and kinematic simulation in application of some of the steps. Then, the advantages of using the methodology are discussed by using two case studies. The final section of the paper gives the conclusions on the use of this methodology.

The Methodology

The main difference in applying a common methodology in these two types of simulation techniques is due to difficulties encountered in making abstractions in building the model of a real system. The model building process, regardless of the type of simulation technique used, requires some level of abstraction as a model can represent a only a limited amount of detail. As the model detail increases less abstraction is needed to match the entities of the real system. For a model that represents only the general attributes of a system, higher levels of abstraction are necessary. It can be observed that a robotic simulation model would require less abstraction since by nature those models match closely the physical characteristics of actual systems. The modeler concerns only with the problems of which physical entities are to be included and the level of geometrical detail those entities should have in the model. For a discrete-event model, however, the level of detail can vary significantly from one model to another depending on many factors including the objectives of the simulation model, the software to be used, and the time frame allowed for the study. For example, a job shop can be modeled in great detail by representing each machine, each part, and each worker if the objective of the study is to investigate the impact of different work schedules on the shop throughput. On the other hand, the study might be concerned with the performance of the shop's tool crib. In this case, the job-shop can be represented only as a black box that generates random requests for the crib. Clearly, one can expect that all the components of the crib that affect the overall performance (e.g., the location of different types of tools, servers, number and type of tools and servers, and priority given to different orders) are included in such a model.

The common methodology that is suggested in this paper has eight phases as given below:

Phase 1. Define the Problem
Phase 2. Design the Study
Phase 3. Design the Conceptual Model
Phase 4. Formulate Inputs, Assumptions, and Process Definition
Phase 5. Build, Verify, and Validate the Simulation Model
Phase 6. Experiment with the Model and Look for Opportunities for Design of Experiments
Phase 7. Documentation and Presentation
Phase 8. Define the Model Life Cycle

Each phase is described in terms of detailed steps in Table I with an indication of the importance of the steps as they apply to each type of simulation technique. Items listed for Phase 5 and Phase 7 are interpreted as guidelines rather than steps. In previous papers (Ulgen, Black, Johnsonbaugh, and Klungle 1994a and Ulgen, Black, Johnsonbaugh, and Klungle 1994b), one of the authors described in detail each of these steps. In what follows, we only highlight the differences in applying the steps to the two simulation techniques rather than describing each step in detail.

In phase I, determining the boundaries of the system to be modeled is typically a more critical and harder task while defining the problem in a discrete-event simulation study. One must pay close attention to all the components of the system which may have a direct or an indirect impact on the target performance measure(s). In a robotic simulation study, the boundary of the system is defined only by considering the objectives of the study in most cases.

The advantages and benefits from a kinematic simulation model are easier to estimate before the study begins. However, in a discrete-event simulation study, benefits may be hard to quantify even after the study is completed. A healthy evaluation of the expectations from a discrete-event model must be made prior to building one. Such an evaluation is likely to affect the level of detail that would be put in a discrete-event simulation.

In phase 2, an accurate estimation of the life-cycle of a discrete-event model is crucial in determining the modeling and animation requirements. If, for example, the model is to be used for training purposes after it is used for analysis then, it might be important to include a detailed animation of the system. Similarly, a robotic simulation model might be needed for creating off-line programs after it is used for analysis purposes. Such a model is likely to include much more detail than a model to be used for analysis only.

Table I: The Eight Phases of the Methodology (✓: Step applies, +: More significant, N/A: Not Applicable)

METHODOLOGY	Type of Simulation Technique	
	Discrete-Event	**Robotics**
Phase 1: DEFINE THE PROBLEM		
Step 1. Define the objectives of the study.	✓	✓
Step 2. List the specific issues to be addressed.	✓	✓
Step 3. Determine the boundary or domain of the study.	✓+	✓
Step 4. Determine the level of detail or proper abstraction level.	✓+	✓
Step 5. Determine if a simulation model is actually needed; will an analytical method work?	✓	N/A
Step 6. Estimate the required resources needed to do the study.	✓	✓
Step 7. Perform a cost-benefit analysis.	✓	✓
Step 8. Create a planning chart of the proposed project.	✓	✓
Step 9. Write a formal proposal.	✓	✓
Phase 2: DESIGN THE STUDY		
Step 1. Estimate the life cycle of the model.	✓+	✓
Step 2. List broad assumptions.	✓	✓
Step 3. Estimate the number of models required.	✓	✓
Step 4. Determine the animation requirements.	✓	✓
Step 5. Select the tool.	✓	✓
Step 6. Determine the level of data available and what data is needed.	✓	✓+
Step 7. Determine the human requirements and skill levels.	✓	✓
Step 8. Determine the audience (usually more than one level of management).	✓+	✓
Step 9. Identify the deliverables.	✓	✓
Step 10.Determine the priority of this study in relationship to other studies.	✓	✓
Step 11.Set milestone dates.	✓	✓
Step 12.Write the Project Functional Specifications.	✓+	✓
Phase 3: DESIGN THE CONCEPTUAL MODEL		
Step 1. Decide on continuous, discrete, or combined modeling.	✓+	✓
Step 2. Determine the elements that drive the system.	✓+	✓
Step 3. Determine the entities that should represent the system elements.	✓+	✓
Step 4. Determine the level of detail needed to describe the system components.	✓	✓
Step 5. Determine the graphics requirements of the model.	✓+	✓
Step 6. Identify the areas that utilize special control logic.	✓	✓+
Step 7. Determine how to collect statistics in the model and communicate results to the customer.	✓	✓
Phase 4: FORMULATE INPUTS, ASSUMPTIONS, AND PROCESS DEFINITION		
Step 1. Specify the operating philosophy of the system.	✓	✓
Step 2. Describe the physical constraints of the system.	✓	✓
Step 3. Describe the creation and termination of dynamic elements.	✓+	✓
Step 4. Describe the process in detail.	✓	✓
Step 5. Obtain the operation specifications.	✓	✓
Step 6. Obtain the material handling specifications.	✓	✓
Step 7. List all the assumptions.	✓	✓
Step 8. Analyze the input data.	✓	N/A
Step 9. Specify the runtime parameters.	✓	✓
Step 10.Write the detailed Project Functional Specifications.	✓	✓
Step 11.Validate the conceptual model.	✓	✓

Table I: The Eight Phases of the Methodology (Continued)

	Discrete-Event	Robotics
Phase 5: BUILD, VERIFY, AND VALIDATE THE SIMULATION MODEL		
1. Beware of tool limitations.	✔	✔
2. Construct flow diagrams as needed.	✔	✔
3. Use modular techniques of model building, verifications, and validation.	✔	✔
4. Reuse existing code as much as possible.	✔	✔
5. Make verification runs using deterministic data and trace as needed.	✔	✔
6. User proper naming conventions.	✔	✔
7. Use macros as much as possible.	✔	✔
8. Use structured programming techniques.	✔	✔
9. Document the model code as model is built.	✔	✔
10.Walk through the logic or code with the client.	✔	✔
11.Set up official model validation meetings.	✔	✔
12.Perform input-output validation.	✔	✔
13.Calibrate the model, if necessary.	✔	✔
Phase 6: EXPERIMENT WITH THE MODEL AND LOOK FOR OPPORTUNITIES FOR DESIGN OF EXPERIMENTS		
Step 1. Make a pilot run to determine warm-up and steady-state periods.	✔	N/A
Step 2. Identify the major variables by changing one variable at a time for several scenarios.	✔	✔
Step 3. Perform design of experiments if needed.	✔	N/A
Step 4. Build confidence intervals for output data.	✔	N/A
Step 5. Apply variance reduction techniques whenever possible.	✔	N/A
Step 6. Build confidence intervals when comparing alternatives.	✔	N/A
Step 7. Analyze the results and identify cause-effect relations among input and output variables.	✔	✔
Phase 7: DOCUMENTATION AND PRESENTATION		
1. Project Book	✔	✔
2. Documentation of model input, code, and output.	✔	✔
3. Project Functional Specifications.	✔	✔
4. User Manual.	✔	✔
5. Maintenance Manual.	✔	✔
6. Discussion and explanation of model results.	✔	✔
7. Recommendations for further areas of study.	✔	✔
8. Final Project Report and presentation.	✔	✔
Phase 8: DEFINE THE MODEL LIFE CYCLE		
Step 1. Construct user-friendly model input and output interfaces.	✔✚	✔
Step 2. Determine model and training responsibility.	✔	✔
Step 3. Establish data integrity and collection procedures.	✔	✔✚
Step 4. Perform field data validation tests.	✔	✔

A robotic simulation model requires highly precise geometry and layout data as well as accuracy in robot motion parameters. Approximations are often not acceptable and building a robotic model in absence of data is impossible in most cases. However, a discrete-event model can be built without waiting for data to become available as long as the logic and the components of the system are defined. Clearly, meaningful results can be obtained only if reliable data are made available to a discrete-event model. Also, in most cases several types of data such as downtime frequencies, repair times, and part interarrival time distributions are not readily available. In some other cases, there may not be data available since a completely new system is being simulated. Even under those circumstances, building a useful discrete-event model might be possible through several layers of abstractions and well established, frequently used assumptions (e.g., repair times are distributed exponentially.)

Some discrete-event models are used for important financial decisions (e.g., buying another AGV, opening up additional AS/RS lanes, including new conveyors etc.) In those situations, perception of the underlying problems and assumptions might be different at different levels of management. Therefore, it is critical that these different layers of management are kept involved (or, at least, informed) during the course of a discrete-event simulation study. Doing so is particularly important in large-scale simulation studies that may involve more than one model.

Mostly because of the fact that several layers of abstractions are possible and defining systems' boundaries and its critical components is a critical task, Phase 3, namely, the development of a conceptual model is a very crucial phase for a discrete-event simulation study. Correct identification of the relevant components, temporary and permanent entities, and detail level in representing those items is essential for successful application of the discrete-event simulation technique. For a robotic simulation study, defining special control logic (synchronization of various events and motions) is the more critical step in the conceptual model development process. Also, one must be aware of physical phenomena that can not be modeled accurately (e.g., weight of objects, torque, and momentum).

In Phase 5, an important difference in the model development processes of the two types of simulation techniques is in the actual model building phase. In general, the modeling of work locations and motion paths for kinematic devices is an iterative process. At each iteration, a new set of work locations and paths are developed by aiming at improving and tuning the robot motions for better cycle time and smoother motions. In other words, modeling and analysis go together for most of the development cycle. Also, verification of a kinematic simulation model would be clearly different from that of a discrete-event model. Particularly, models of kinematic devices must be tested to verify that the motions observed in simulations reflect the behavior of actual devices (or the expected behavior in the case of newly designed devices.)

There are also differences in the types of analyses performed on a model in the last phase, that is Phase 6, of a simulation project. A discrete-event simulation model typically involves randomly developing events. Therefore, measures of performance obtained from those models are random variables and require proper treatment in the form of statistical analyses.

Case Studies

A DISCRETE-EVENT MODEL OF A VEHICLE PAINT SHOP

The paint shop involved in this study was part of a vehicle assembly plant. The new paint facility is an upgrade of an existing one which had to be redesigned to increase the throughput of the system. The new paint shop consists of approximately 30 conveyor chains that move jobs from one process to another. The output of the shop goes into an AS/RS from where parts are sent to final assembly line. There are several types of vehicles with different paint requirements.

As the entire plant was undergoing an upgrade, the design team included members from several levels of management and engineering. As the simulation team found earlier in the study, there were significant differences in management's views of what was important. There were even conflicting views of the objectives of the study. As part of the application of the methodology, the simulation team focused on defining a common set of objectives at a very early stage of the study. Consequently, potential problems regarding the proper usage of simulation were successfully avoided. Furthermore, the model was built and modified with the minimum amount of detail necessary to obtain valid measures of performance. The level of detail and abstraction were adjusted to the changing needs and objectives of the design process.

As any design process, the expansion project consisted of several iterations of making and evaluating alternative layout designs. As the design evolved, the simulation model went through several modifications. Developing and continuously updating project functional specifications helped to establish a common information source for everyone participated in the project. As several groups of people involved in the process, up-to-date documentation of the model, its assumptions, input data, and current results proved to be extremely beneficial in eliminating communication problems. Furthermore, updating periodically a designated simulation coordinator at the client site on the status of the project helped to establish and maintain an excellent communication channel. The periodical status reports made clear what the status of the project was and what the next steps were and helped to minimize frustrations of parties involved in the project.

The results obtained from the simulation study had significant impact on various equipment, layout and scheduling issues. Determination of the size of banks of buffer conveyors, adjustment of the speed of several production conveyors, and establishment of job sequencing rules were the most important results of the study. The simulation model was also updated for the final version of the conveyor controller logic and turned over to client with a user-friendly front-end to be used by the engineers as an on-going decision and training tool.

DISCRETE-EVENT AND KINEMATIC MODELS FOR A VEHICLE FRAME PLANT

The frame plant involved in this study had undertaken an improvement project to automate an arc welding operation through a series of robotic welding cells and a conveyor system to move materials and finished products in and out of those cells. There were concerns about the performance of both the conveyor system and the robot cells. As part of the problem definition phase, the simulation team interviewed with the managers and engineers to capture the objectives of the study. As a result, the nature of the questions formulated for the cell and for the conveyor system required that separate simulations be made for each system. Clearly, a Discrete-event model was appropriate for modeling the conveyor system whereas a kinematic simulation was required for modeling the robotic workcell.

Separate validation meetings with several members of the design team helped calibrating each model. For the discrete-event model, several iterations of validation meetings were held as the objectives and the system changed during the course of the study. Since the engineers from the client site were kept involved in the course of the project, there was a high level of interaction between the design and the model building processes that benefited both sides. The modeling effort also benefited from the modular approach, the correct identification of the components of the system, and the proper definition of the special control logic. The kinematic simulation model was extensively used to investigate the impact of alternative welding patterns on the cell cycle time. As the simulations showed, the cycle time of the cell would be longer than those expected because of delays to avoid collisions. Adjustments to welding patterns helped overcome the cycle time problem. Furthermore, the kinematic simulation model provided the cycle time information that the discrete-event model needed for accurate representation of the welding operation.

The application of the methodology enabled the simulation team to contribute to the success of the expansion project by improving the design both at the cell level and at the line level. Program managers at the client site found the simulation methodology very valuable even though involvement of simulation in the project began later in the study.

Conclusions

Discrete-event and kinematic simulation techniques show differences from each other in the way models are built, the software used, and the types of analyses made. However, as the paper suggests, the nature of simulation studies, regardless of the type of simulation technique, lends itself to the application of a common methodology. The eight-phase methodology presented in this paper allows the users of simulation, whether analysts or clients, to gather maximum benefits from the use of this powerful tool.

An examination of the phases and steps of the methodology indicates that client involvement is perhaps the most important overall recommendation. Being aware of the needs and the objectives of the clients, getting their feedback and approval on the model, updating them on the status of the study, and providing them clearly stated results are crucially important for the success of any simulation study. Adherence to the methodology by paying close attention to each step ensures that those general guidelines are met. Finally, it can be recommended that checklists representing the steps of all phases of the methodology be developed and used throughout any simulation project.

References

Ulgen, O. M., J. J. Black, B. Johnsonbaugh, R. Klungle (1994a) “Simulation Methodology in Practice - Part I: Planning For The Study” International Journal of Industrial Engineering, Volume 1, Number 2, pp. 119-128.

Ulgen, O. M., J. J. Black, B. Johnsonbaugh, R. Klungle (1994b) “Simulation Methodology in Practice - Part I: Selling The Results” International Journal of Industrial Engineering, Volume 1, Number 2, pp. 129-137.

BIOGRAPHY

Eric Grajo holds a master’s degree in Industrial and Systems Engineering from The University of Michigan; a Ph.D. in Industrial and Operations Engineering is in progress. Mr. Grajo is the Technical Manager and Systems Consultant at Production Modeling Corp., an industrial engineering and simulation consulting firm. Mr. Grajo specializes in computer simulation and animation for various manufacturing and material handling applications in automotive, electronics, food, pharmaceutical, and service industries.

Ali Gunal is a systems consultant at Production Modeling Corporation. He received his Ph.D. degree in Industrial Engineering from Texas Tech University in 1991. He is currently involved in application of kinematic and discrete-event simulation in manufacturing industry using IGRIP, QUEST, RobCad, SIMAN/CINEMA, and AutoMod II. His current consulting and research interests include kinematic simulation, off-line programming and robot calibration, simulation of manufacturing and material handling systems, and object-oriented programming.

Dev Sathyadev is a Systems Consultant at Production Modeling Corporation. He received his Ph.D. in Mechanical Engineering from the University of Cincinnati in 1990. Prior to his current position he was faculty in the department of Mechanical and Industrial Engineering for four years. He is currently involved in the development and application of kinematic and dynamic simulation of robotic manipulators for the manufacturing industry using IGRIP and RobCad. His interests include calibration of robotic manipulators, off-line programming of robots, and kinematic and dynamic simulation of mechanical systems.

Onur M. Ulgen is a senior consultant at Production Modeling Corporation and also a Professor of Industrial and Manufacturing Systems Engineering at the University of Michigan-Dearborn. He received his Ph.D. degree in Industrial Engineering from Texas Tech University in 1979. His present consulting and research interests include the applications of discrete-event and robotic (kinematic) simulation to manufacturing problems, object-oriented simulation program generators, scheduling, and project management.