1

CONCEPTS OF SIMULATION MODELING

1.1 OVERVIEW

Driven by growing competition and globalization and to remain competitive, companies across the world strive to maintain high product and service quality, low production costs, short lead times, an efficient supply chain, and high customer satisfaction. To this end, companies often relay on traditional process improvement and cost reduction measures and adopt emerging initiatives of quality management, lean manufacturing, and Six Sigma. These initiatives are widely used for system-level design, improvement, and problem-solving with the aim of integrating continuous improvement into the company's policy and strategic planning.

Successful deployment of such initiatives, therefore, requires an accurate system-level representation of underlying production and business processes. Examples of process representation range from transfer functions to process mapping, flow-charting, modeling, and value stream mapping. Real-world production and business systems are, however, characterized by complexity and dynamic and stochastic behavior. This makes mathematical approximation, static representation, and deterministic models less effective in representing the actual system behavior. Alternatively, simulation facilitates better representation of real-world systems and its application for system-level modeling is increasingly used as a common platform in emerging methods of system design, problem-solving, and improvement.

Simulation modeling, as an industrial engineering (IE) tool, has undergone a tremendous development in the last decade. This development can be pictured

Process Simulation Using WITNESS®, First Edition. Raid Al-Aomar, Edward J. Williams, and Onur M. Ülgen.

Companion Website: www.wiley.com/go/processsimulationusingwitness

through the growing capabilities of simulation software tools and the application of simulation solutions to a variety of real-world problems. With the aid of simulation, companies nowadays can design efficient production and business systems, troubleshoot potential problems, and validate/tradeoff proposed solution alternatives to improve performance metrics, and, consequently, cut cost, meet targets, and boost sales and profits.

WITNESS® simulation software is a modern modeling tool that has been increasingly utilized in a wide range of production and business applications. WITNESS® is mainly characterized by ease-of-use, well-designed simulation modules, and integrated tools for system analysis and optimization. WITNESS® is also linked to emerging initiatives of Six Sigma and Lean Techniques through modules that facilitate Sigma calculation and process optimization.

This book discusses the theoretical and practical aspects of simulation modeling in the context of WITNESS® simulation environment. This chapter provides an introduction to the basic concepts of simulation and clarifies the simulation role and rationale. This includes an introduction to the concept, terminology, and types of models, a justification for utilizing simulation in real-world applications, and a brief discussion on the simulation process. Such background is essential to establish a basic understanding of what simulation is all about and to understand the key simulation role in process engineering and emerging technologies.

1.2 SYSTEM MODELING

System modeling as a term includes two important commonly used concepts; *system* and *modeling*. It is imperative to clarify such concepts before attempting to focus on their relevance to the "Simulation" topic. This section will introduce these two concepts and provide a generic classification to the different types of systeml models.

1.2.1 System Concept

System thinking is a fundamental skill in simulation modeling. The word *system* is commonly used in its broad meaning in a variety of engineering and nonengineering fields. In simple words, a system is often referred to as a set of elements or operations that are logically related and effectively configured toward the attainment of a certain goal or objective. To attain the intended goal or to serve the desired function, it is necessary for the system to receive a set of inputs, process them correctly, and produce the required outcomes. To sustain such flow, a certain control is required to govern the system behavior. Given such definition, we can analyze any system (*S*) based on the architecture shown in Figure 1.1.

As shown in Figure 1.1, each system (*S*) can be mainly defined in terms of a set of Inputs (*I*) received by a Process (*P*) and transformed into a specific set of Outputs (*O*). The process consists of a set of system Elements or Entities (*EN*) that are configured based on a set of logical Relationships (*RL*). An overall Goal (*G*) is often defined to

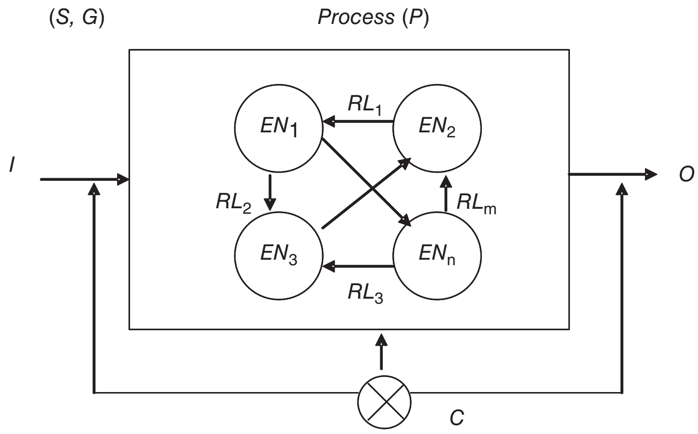

Figure 1.1 Definition of system concept. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

represent the purpose and objective of the system. To sustain a flawless flow and correct functionality of *I–P–O*, some kind of controls (*C*) is essentially applied to system Inputs, Process, and Outputs. Thus, building a system or a system model primarily requires the following:

1. Defining the goal (*G*) or the overall system objective and relating system structure to the goal attainment.
2. Specifying the set of outcomes (*O*) that should be produced and their specifications that result in attaining the specified Goal (*G*).
3. Specifying the set of system Inputs (*I*) that are required in order to produce the specified system Outcomes (*O*) along with the specifications of these Inputs (*I*).
4. Listing system entities $S = (EN_1, EN_2, EN_3, \ldots, EN_n)$ and defining the characteristics and the individual role of each entity (resources, storage, etc.).
5. Setting the logical relationships $(RL_1, RL_2, RL_3, \ldots, RL_m)$ among the defined set of system elements to perform the specified process activities.
6. Specifying the system Controls (*C*) and their role in monitoring the specifications of system Inputs (*I*) and Outputs (*O*) and adjusting the operation of the Process (*P*) to meet the specified Goal (*G*).

This understanding requires for any arrangement of objects to be called a system to be structured logically and to have an interaction that leads to a useful outcome. Transforming system inputs into desired outputs is often performed through system resources. Correct processing is often supported by controls and inventory systems to assure quality and maintain steady performance. This understanding of system concept is our gateway to the broader subject of system engineering. Examples of common real-life systems include classrooms, computer systems, factories, hospitals, and so on. In the classroom example, students are subject to various elements of the educational process (P) in the classroom, which involves attendance, participation in class activities, submitting assignments, passing examinations, and so on in order to complete the class with certain qualifications and skills. Applying the definition of system to the classroom example leads to the following:

1. The overall system goal (G) is set to educate students on a certain subject and provide quality education to students attending classes.
2. System Inputs (I) are students of certain age, academic level, major, and so on.
3. System Outputs (O) are also students upon fulfilling class requirements.
4. The set of system entities is defined as follows:

$$S = \{\text{Tables, Chairs, Students, Instructor, Books, Whiteboard}\}$$

5. The defined entities in S are logically related through a set of relationships (RL). For example, chairs are located around tables that face the instructor, the instructor stands in front of students and writes on whiteboard, and so on.
6. Finally, class regulations and policies for admission, attendance, grading, and graduation represent process Controls (C).

It is worth mentioning that the term "system" covers both products and processes. A product system can be an automobile, a cellular telephone, a computer, a calculator, and so on. Any of these products involves the defined components of the system in terms inputs, outputs, elements, relationships, controls, and goal. Try to analyze all the mentioned examples from this perspective. On the other hand, a process system can be a manufacturing process, an assembly line, a power plant, and a business process. Similarly, any of these processes involves the defined components of the system. Try to analyze all the mentioned examples from this perspective.

1.2.2 Modeling Concept

The word *modeling* refers to the process of representing a system with a selected model that is easier to understand and less expensive to build compared to the actual system. The system model includes a representation of system elements, relationships, inputs, controls, and outputs. Modeling a system, therefore, has two prerequisites:

1. Understanding the structure of the actual (real-world) system and the functionality of its components. It is imperative for the analyst to be familiar with the system and understand its purpose and functionality. For example, in an automobile assembly plant, the modeler needs to be familiar with the production system of building vehicles before attempting to model the vehicle assembly operations. Similarly, the modeler needs to be familiar with different types of bank transactions to develop a useful model of a bank.

2. Being familiar with different modeling and system representation techniques and methods. This skill is essential to choose the appropriate modeling technique for representing the underlying real-world system under budgetary and time constraints. The selection of the most feasible modeling method is a decision of economy, attainability, and usefulness.

Modeling a system of interest is a combination of both art and science. It involves abstracting a real-world system into a clear, comprehensive, accurate, reliable, and useful representation. Such model can be used to better understand the system and to facilitate system analysis and improvement. As shown in Figure 1.2, the objects of the real-world system are replaced by objects of representation and symbols of indication. This includes the set of system entities (*EN*), relationships among entities (*RL*), system inputs (*I*), Controls (*C*), and system outputs (*O*). Actual system *EN–RL–I–C–O* is mimicked to a degree in the system model, leading to a representation that captures the characteristics of the real-world process for the purpose at hand. However, in a reliable model, this approximation should be as realistic as needed and should not overlook the key system characteristics.

1.2.3 Types of Models

Several modeling methods can be used to develop a system model. The analyst's choice of modeling method is based on several criteria including modeling objective, system nature and complexity, and the time and cost of modeling. As shown in Figure 1.3, we can classify the different types of models into four major categories: physical models, graphical models, mathematical models, and computer (logical) models. The following is a summary of those types of models.

1. *Physical models*: Physical models are tangible prototypes of the actual products or processes in a one-to-one scale or in any other feasible scale of choice. Such models provide a close-to-reality direct representation of the actual system to demonstrate its structure and functionality in a physical manner. They are common in large-scale engineering projects such as new car and airplane concepts, bridges, buildings, ships, and other architectural designs. They help designers

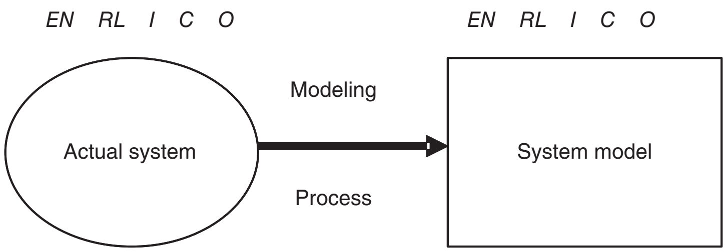

Figure 1.2 The process of system modeling. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

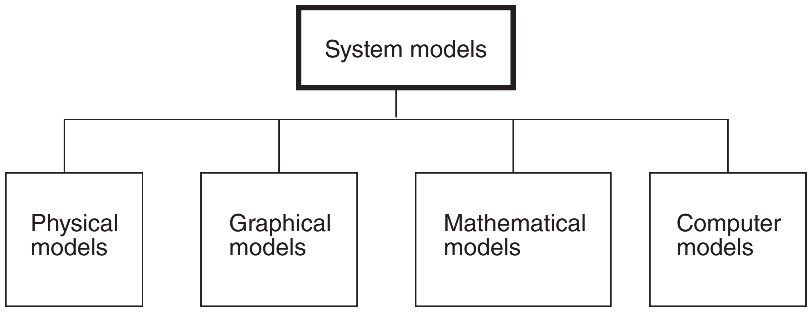

Figure 1.3 Types of system models. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

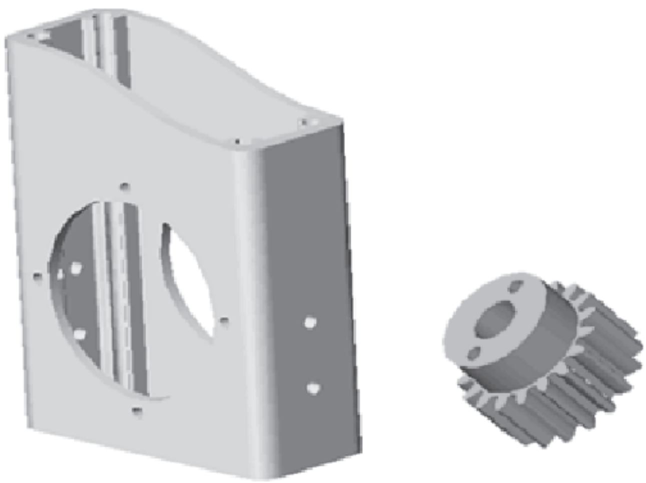

Figure 1.4 Examples of product prototypes. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

better understand the system of interest and allow them to try out different configurations of design elements before the actual build up. Physical models may be built from clay or wood such as car prototypes or developed using 3D printing machines using different materials. Various techniques of rapid prototyping and reverse engineering are also used to develop product/process prototypes. Figure 1.4 shows examples of product prototypes. Physical models can also be operational models such as flight simulators and real time simulators of chemical operations. Another form of physical models can be Lego-type machines, conveyor structures, and plant or reactor models. The benefit of physical models is the direct and easy to understand tangible representation of the underlying system. However, there are several limitations to physical models. The cost of physical modeling could be enormous in some cases. Some systems are too complex to be prototyped. Other physical models might be time consuming and require superior crafting skills to be built. For example, think of building a physical model for an internal combustion engine (ICE) or an assembly line of personal computers (PCs). What kind of cost, time, and skill would be involved in developing such prototypes?

2. *Graphical models*: Graphical models are abstracts of products or processes developed using graphical tools. These tools range from paper and pencil sketches to engineering drawings. Common graphical representations include process maps, flow and block diagrams, networks, and operations charts. Figure 1.5 presents an example of a graphical model (operations chart of can opener assembly). The majority of graphical representations are static models that oversimplify the reality of the system and do not provide technical and functionality details of the process, which makes it difficult to try out what-if scenarios and to explain how the system responds to various changes in process parameters and operating conditions. Thus, graphical models as commonly used to develop physical and computer models.

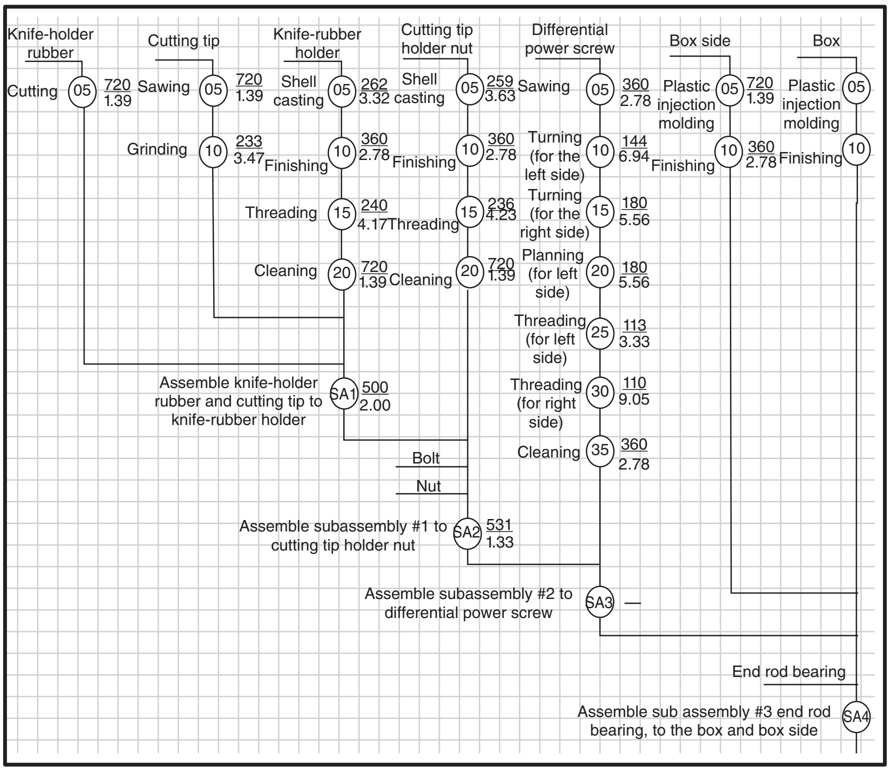

Figure 1.5 Example of a process graphical model. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

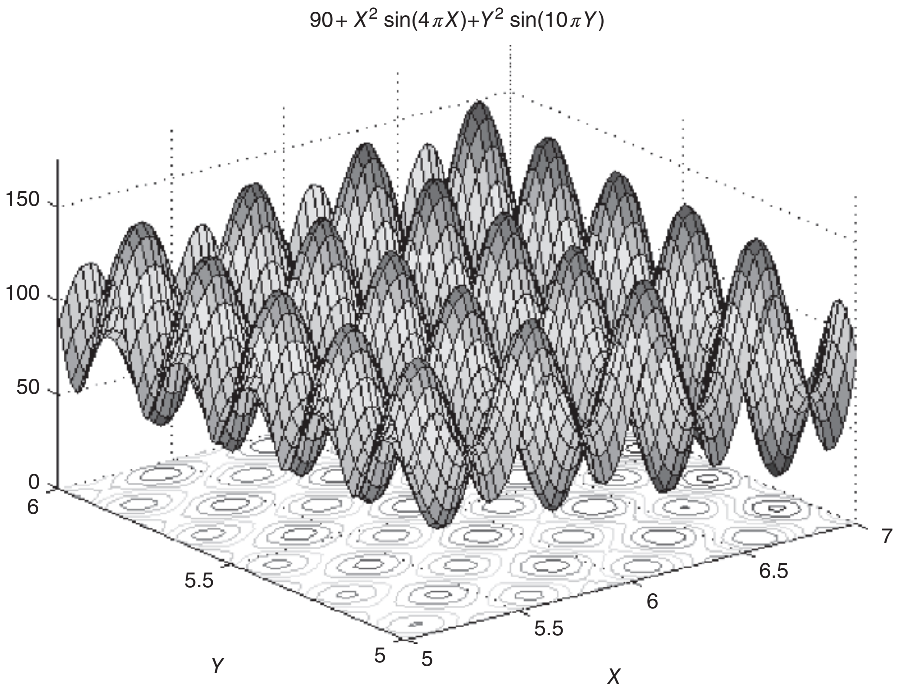

Figure 1.6 An example of a MATLAB mathematical model. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

3. *Mathematical Models*: Mathematical modeling is the process of representing the system behavior with formulas, mathematical equations, and calculus-based methods. They are symbolic representations of systems functionality, decision (control) variables, response, and constraints. Design formulas for stress-strain analyses, probabilistic and statistical models, and mathematical programming models are examples of mathematical models. Typical example of mathematical models includes using linear programming (LP) in capital budgeting, production planning, resources allocation, and facility location. Other examples include queuing models, Markov chains, and economic order quantity (EOQ) model. Some mathematical models can be also empirical models derived from regression analysis and transfer functions. Typically, a mathematical formula is a closed-form relationship between a dependent variable (Y) and one or more independent variables (X) with the form of $Y = f(X)$. Such a formula can be linear or nonlinear. Figure 1.6 shows an example of a mathematical model built using MATLAB software. The dependent variable is often selected to measure a key characteristic of the system such as the speed of a vehicle or the yield of a process. Independent variables of the formula represent the key or critical parameters on which

system response depends such as time, distance, or force. Unfortunately, not all system responses can be modeled using mathematical formulas. Complexity of most real-world systems challenges the application of such models. Hence, a set of simplification assumptions often accompanies the application of mathematical models in order for the derived mathematical formulas to hold. For example, applying the EOQ inventory model assumes constant demand and lead-time. Such assumptions often lead to impractical results that have a limited chance of being implemented. For example, think of developing a formula that computes a production system throughput given parameters such as machine cycle times, speeds of conveyance systems, number of assembly operators, sizes of system buffers, and plant operating pattern. What kind of mathematical model would you use to approximate such response? How representative will the mathematical model be? Can you use the throughput numbers obtained from such a mathematical model to plan schedule deliveries to customers?

4. *Computer Models*: Computer models are numerical, graphical, mathematical, and logical representation of a system (a product or a process) that utilizes the capability of computers in fast computations, large capacity, consistency, animation, and accuracy. Computer simulation models, in particular, are virtual representations of real-world products and processes on the computer. Simulations of products and processes are developed using different application programs and software tools. For example, a computer program can be used to develop a finite element analysis (FEA) model to analyze stress and strains for a certain product design, as shown in Figure 1.7. Similarly, several mathematical models that represent complex mathematical operations, control systems, fluid mechanics, and others can be built, animated, and analyzed with computer tools. Software tools are also available to develop static and dynamic animations of many industrial processes.

Accurate and well-built computer models compensate for the limitations of the other types of models. They are typically easier, faster, and cheaper than building physical models. In addition, the flexibility and computation capability of computer models allow for making quick model changes, easy testing of what-ifs, and accurate evaluation of system performance for experimental design and optimization studies. Computer models also provide the benefits of graphical models through modern animation and graphical modeling tools. Compared to complex mathematical models, computer models are generally more realistic and efficient. They utilizel computer capabilities for more accurate approximations, they run tremendous computations in little time, and they can measure system performance without the need for a closed-form definition of the system objective function. Such capabilities made computer models the most common modeling techniques. Limitations of computer models stem from the limitation of computer hardware, the limitations of application software, the simulated system complexity and data availability, and the limited skills

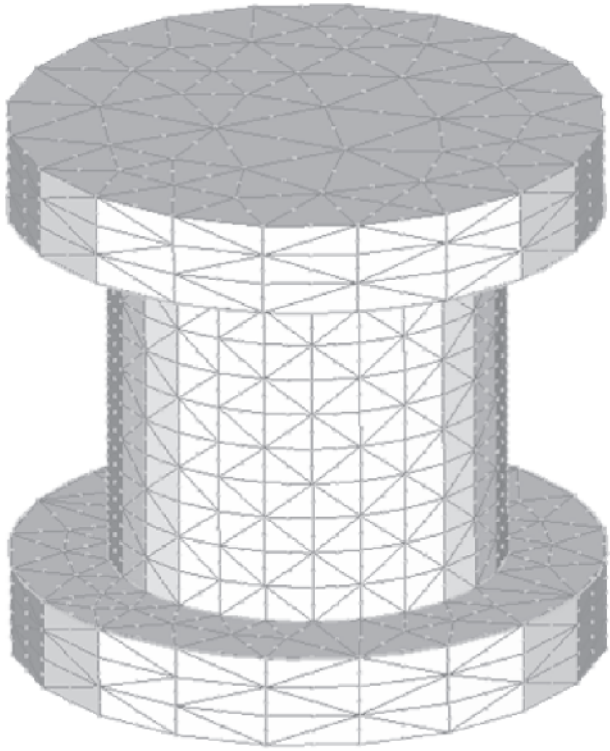

Figure 1.7 Example of a FEA computer model. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

of the simulation analyst in benefiting from the software features and in conducting the simulation analyses.

discrete event simulation (DES) is the type of computer simulation that mimics the operation of real-world processes as they evolve over time. The mechanism of DES computer modeling, discussed in Chapter 2, assists in capturing the dynamics and logics of system processes and estimating the system's long-term performance under stochastic conditions. Moreover, DES models allow the user to test various "what-if" system scenarios, make model changes to mimic potential changes in the physical conditions, and run the system many times for long periods to "simulate" the impacts of such changes. The model results are then analyzed to gain insight into the behavior of the system. For example, a DES plant model can be used to estimate the assembly line throughput by running the model dynamically and tracking its throughput hour-by-hour or shift-by-shift. The model can also be used to assess multiple production scenarios based on long-term average throughput. As shown in Figure 1.8, simulation software tools provide a flexible environment of modeling and analysis that makes DES more preferable compared to graphical, mathematical, and physical models.

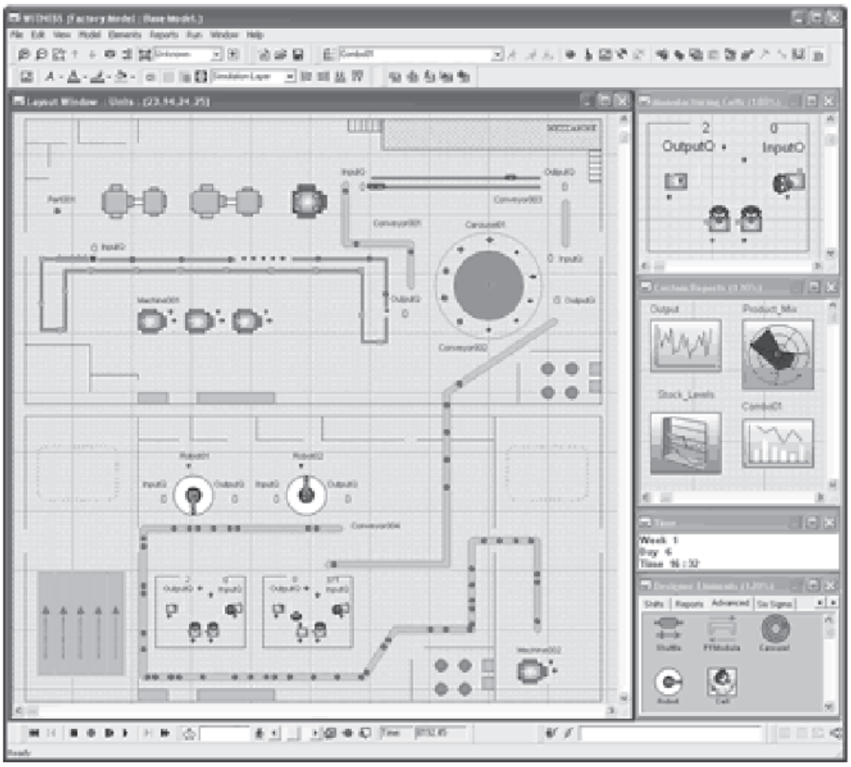

Figure 1.8 Example of a DES model built-in WITNESS® software.

1.3 SIMULATION MODELING

Modeling, as shown earlier, is the art and science of capturing the functionality and the relevant characteristics of real-world systems. Modeling involves presenting such systems in a form that provides sufficient knowledge and facilitates system analyses and improvement. Physical, graphical, mathematical, and computer models are the major types of models developed for different purposes and applications. This section focuses on defining the *simulation* concept, developing a taxonomy of different types of simulation models, and explaining the role of simulation in planning, designing, and improving the performance of business and production systems.

1.3.1 Simulation Defined

Simulation is a widely used term in reference to computer models that represent physical systems (products or processes). It provides a simplified representation that captures important operational features of a real system. For example, FEA represents the mathematical basis for a camshaft product simulation. Similarly, production flow, scheduling rules, and operating pattern represent the logical basis for developing a plant process model.

System simulation model is the computer mimicking of the complex, stochastic, and dynamic operation of a real-world system (including inputs, elements, logic, controls, and outputs). Examples of system simulation models include mimicking the day-to-day operation of a bank, the production flow in an assembly line, or the departure/arrival schedule in an airport. As an alternative to impractical mathematical models or costly physical prototypes, computer simulation has made it possible to model and analyze real-world systems.

As shown in Figure 1.9, the primary requirements for simulation are: a system to be simulated, a simulation analyst, a computer system, and simulation software. The analyst has a pivotal role in the simulation process. He or she is responsible for understanding the real-world system (inputs, elements, logic, and outputs), developing a conceptual model, and collecting pertinent data. The analyst then operates the computer system and uses the simulation software to build, validate, and verify the system simulation model. Finally, the analyst analyzes simulation results and determines best process setting.

Computer system provides the hardware and software tools required to operate and run the simulation model. The simulation software or language provides the platform and environment that facilitates model building, testing, debugging, and running. The simulation analyst utilizes the simulation software on a capable computer system to develop a system simulation model that can be used as a practical (close-to-reality) representation of the actual system.

1.3.2 Simulation Taxonomy

Based on the selected internal representation scheme, simulation models can be *discrete*, *continuous*, or *combined*. DES models, which are the focus of this book, are

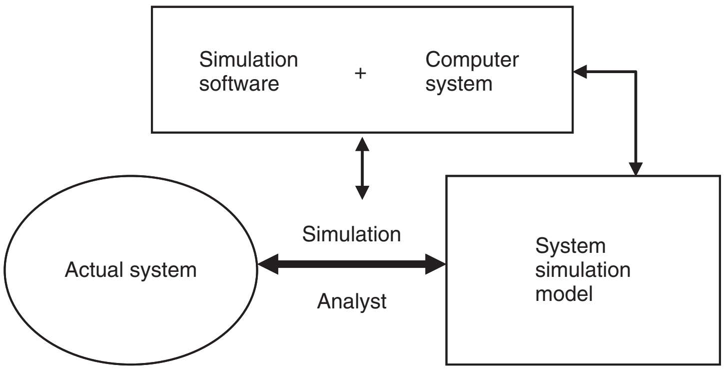

Figure 1.9 The simulation process. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

the most common among simulation types. DES models are based on a discrete internal representation of model variables (variables that change their state at discrete points in time). DES mechanics will be discussed in detail in Chapter 2. In general, discrete simulation models focus on modeling discrete variables that receive values from random or probabilistic distributions, where the state of the system changes in discrete points in time. A discrete variable can be the number of customers in a bank, products and components in an assembly process, or cars in a drive-through restaurant.

Continuous simulation models, on the other hand, focus on continuous variables, receiving values from random or probabilistic distributions, where the state of the system changes continuously. Examples of continuous variables include waiting time, level of water behind a dam, and fluids flow in chemical processes and distribution pipes. Continuous simulation is less popular than discrete simulation since the majority of production and business systems are modeled using discrete random variables (customers, units, entities, orders, etc.).

Combined simulation models include both discrete and continuous elements in the model. For example, separate (discrete) fluid containers arrive to a chemical process where fluids are poured into a reservoir to be processed in a continuous manner. This kind of simulation requires the capability to define and track both discrete and continuous variables.

Furthermore, models are either *deterministic* or *stochastic*. A stochastic process is modeled using probabilistic models. Examples of stochastic models include customers arriving to a bank, servicing customers, and equipment failure. In these examples, the random variable can be the inter-arrival time, the service or processing time, and equipment time to failure (TTF), respectively.

Deterministic models, on the other hand, involve no random or probabilistic variables in their processes. Examples include modeling fixed cycle time operations in an automated system or modeling the scheduled arrivals to a clinic. The majority of real-world operations are probabilistic. Hence most simulation studies involve random generation and sampling from theoretical or empirical probability distributions to model random system variables. Variability in model inputs leads to variability in model outputs. As shown in Figure 1.10, a deterministic model $Y = f(X)$ will generate a stochastic response (Y) when model inputs (X_1, X_2, and X_3) are stochastic. If the response represents the productivity of a production system, model inputs such as parts arrival rates, demand forecast, and model mix generate a variable production rate.

Finally, and based on the nature of model evolvement with time, models can be *static* or *dynamic*. As shown in Figure 1.11, a simulation model can involve both static and dynamic responses. In static models, system state (defined in state variables) does not change over time. For example, a static variable (X_1) can be a fixed number of workers in an assembly line, which does not change with time. Alternatively, a dynamic variable (X_2) can be the number of units in a buffer, which changes dynamically over time. Monte Carlo simulation models are time independent (static) models that deal with a system of fixed state. In such spreadsheet-like models, certain variable values change based on random distributions and performance measure

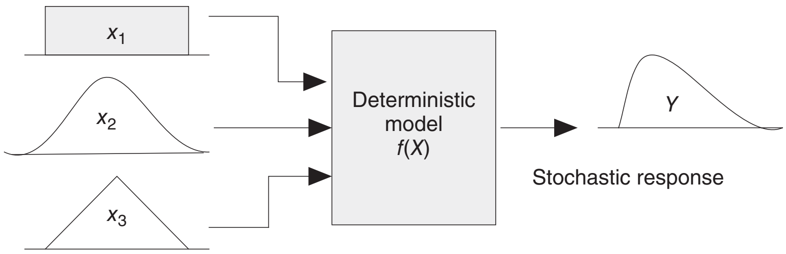

Figure 1.10 A deterministic model with stochastic inputs and response. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

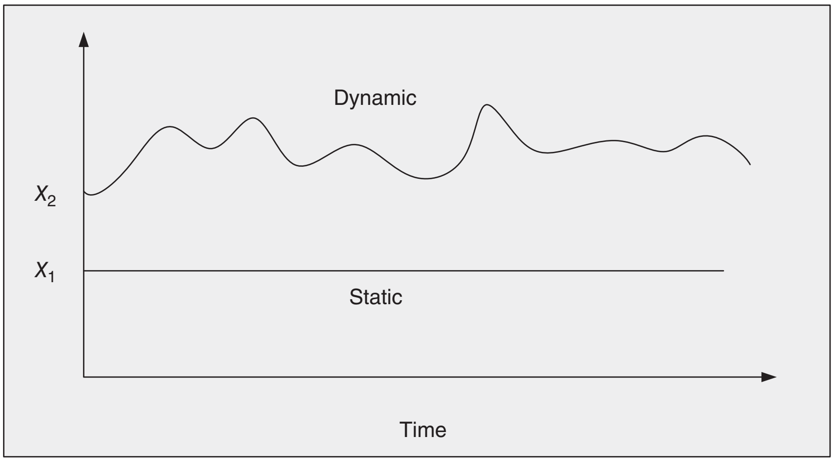

Figure 1.11 Static and dynamic model variables. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

are evaluated per such changes without considering the timing and the dynamics of such changes. Most operational models are, however, dynamic. System state variables often change with time and the interactions that result from such dynamic changes do impact the system behavior.

Dynamic simulation models are further divided into terminating and nonterminating models based on run time. Terminating models are stopped by a certain *natural* event such as the number of items processed or reaching a certain condition. For example, a bank model stops at the end of the day and a workshop model stops when finishing all tasks in a certain order. These models are impacted by initial conditions (system status at the start). Nonterminating models, on the other hand, can run continuously making the impact of initialization negligible. For example, a plant runs in continuous mode where production starts every shift without emptying the system. The run time for such models is often determined statistically to obtain a steady-state response. Figure 1.12 presents a simulation taxonomy with highlighted attributes

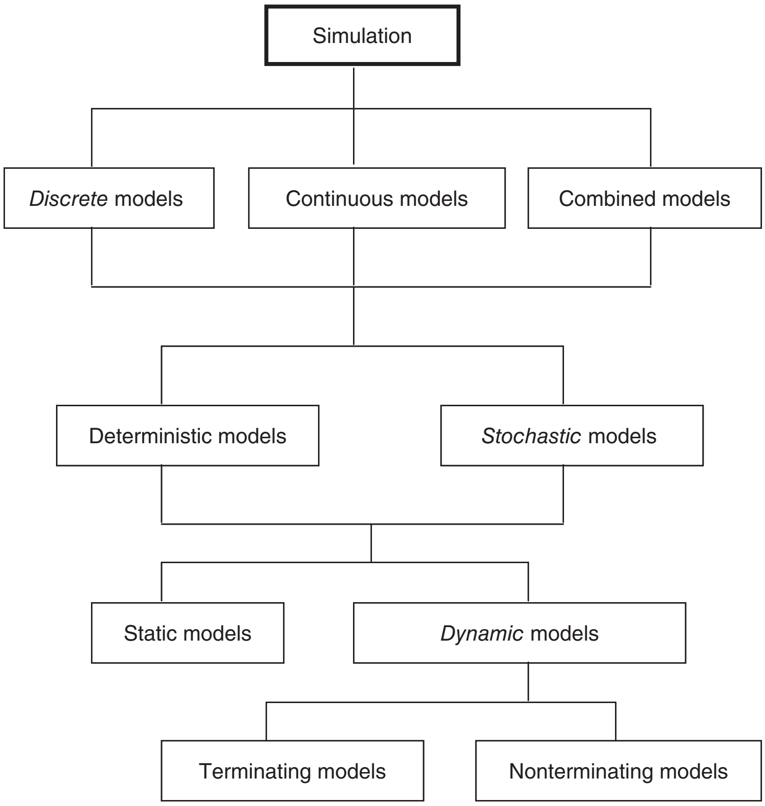

Figure 1.12 Simulation taxonomy. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

of DES (discrete, stochastic, and dynamic models of terminating or nonterminating response).

1.4 THE ROLE OF SIMULATION

After understanding the different concepts and aspects of the term "simulation modeling," it is necessary to clarify the role that the simulation plays in developing production and business systems. We first justify the use of simulation *technically* and *economically* and then present the spectrum of simulation applications in manufacturing and service sectors.

1.4.1 Simulation Justified

"Why and when to simulate?" and "How can we justify a simulation project?" are key questions that often cross the mind of simulation practitioners, engineers, and decision-makers. We simply turn to simulation because of simulation capabilities that are unique and powerful in system representation and performance estimation under real-world conditions. Most real-world processes in production and business systems are complex, stochastic, and highly nonlinear and dynamic. Other modeling types such as graphical, mathematical, and physical models fall short in providing a cost-effective and usable system representation under such conditions.

"Decision support" is another common justification of simulation studies. Obviously, engineers and managers want to make the best decisions possible, especially when encountering critical stages of design, expansion, or improvement projects. Simulation studies may reveal insurmountable problems and save cost, effort, and time. They reduce the cost of wrong capital commitments, reduce investments risk, increase design efficiency, and improve the overall system performance.

Although simulation studies might be costly and time-consuming in some cases, the benefits and savings obtained from such studies often recover the simulation cost and avoid much larger costs. Simulation costs are typically the initial simulation software and computer cost, yearly maintenance and upgrade cost, training cost, engineering time cost, and other costs for traveling, preparing presentations with multimedia tools, and so on. Such costs are often recovered through the long-term savings from increasing productivity and efficiency.

1.4.2 Simulation Applications

A better answer to the question "why simulate?" can be reached by exploring the wide spectrum of simulation applications to various aspects of business, science, and technology. This spectrum starts by designing queuing systems and extends to designing communication networks, production systems, and business operations. The focus in this book is on the wide range of simulation applications in both manufacturing and business operations. Simulation models of manufacturing systems can be used for many objectives including:

- Determining throughput capability of a manufacturing cell, an assembly line, or a production system.
- Configuring labor resources in an intensive assembly process.
- Determining the needed number of automated guided vehicles (AGVs) in a complex material handling system (MHS).
- Determining the size and resources in a complex automated storage and retrieval system (AS/RS).
- Determining best ordering policies for an inventory control system.
- Validating the outcomes of material requirement planning (MRP).
- Determining buffer sizes for work-in-progress (WIP) in an assembly line.

For business operations, simulation models can be also used for a wide range of applications including:

- Determining the number of bank tellers that results in reducing customers waiting time by a certain percentage.
- Designing distribution and transportation networks to improve the performance of logistic and supply chains.
- Analyzing the financial portfolio of a company over time.
- Designing the operating policies in a fast food restaurant to reduce customer Time-In-System and increase customer satisfaction.
- Evaluating hardware and software requirements for a computer network.
- Scheduling the working pattern of the medical staff in an emergency room (ER) to reduce patients' waiting time.
- Testing the feasibility of different product development processes and evaluating their impact on the company's budget and strategy.
- Designing communication systems and data transfer protocols.
- Designing traffic control systems.

Table 1.1 shows a summary of ten examples of simulation applications in both manufacturing and service sectors.

To reach the goals of the simulation study, certain elements of each simulated system often become the focus of the simulation model. Modeling and tracking such elements provide attributes and statistics necessary to design, improve, and optimize the underlying system performance. Table 1.2 shows a summary of ten examples of simulated systems with examples of principal model elements.

1.4.3 Simulation Precautions

Like any other engineering tool, simulation has limitations. Such limitations should be realized by practitioners and should not discourage analysts and decision-makers from using simulation. Knowing the limitations of simulation should emphasize using

TABLE 1.1 Examples of Simulation Applications

Manufacturing	Services
Automotive	Banking industry
Aerospace	Health systems
Plastics industry	Hotel operations
Paper mills	Communication services
House appliances	Computer networks
Furniture manufacturing	Transportation systems
Chemical industry	Logistics and supply chain
Clothing and textile	Restaurants and fast food
Packaging	Postal services
Storage/retrieval	Airport operations

TABLE 1.2 Examples of Simulated Systems

Simulated System	Examples of Model Elements
Manufacturing system	Parts, machines, operators, conveyors, storage
Emergency room (ER)	Patients, beds, doctors, nurses, waiting room
Bank	Customers, bank tellers, ATMs, loan officers
Retail store	Shoppers, checkout cash registers, customer service
Computer network	Server, client PCs, administrator, data protocol
Freeway system	Cars, traffic lights, road segments, interchanges
Fast food restaurant	Servers, customers, cars, drive-through windows
Border crossing point	Cars, customs agents, booths, immigration officers
Classes registration office	Students, courses, registration stations, helpers
Supply chain/logistics	Suppliers and vendors, transportation system, clients

it wisely and should motivate the user to develop creative methods and establish the correct assumptions in order to benefit from the powerful simulation capabilities. Still, however, certain precautions should be considered to avoid the potential pitfalls of simulation studies. We should pay attention to the following issues when considering simulation:

- The simulation analyst as well as the decision-maker should be able to answer the question "when not to simulate?" Simulation studies may not be used for solving problems of relative simplicity. Such problems can be solved using engineering analysis, common sense, or mathematical models.
- The cost and time of simulation should be considered and planned well. Many simulation studies are underestimated in terms of time and cost. Some decision-makers think of simulation as model building although it consumes less time and cost when compared to data collection and output analysis.
- The skill and knowledge of the simulation analyst need to be addressed. Essential skills for simulation practitioners include systems thinking, fluency in programming and simulation software, knowledge in statistics, strong communication and analytical skills, project management (PM) skills, ability to work in teams, and creativity in design and problem-solving.
- Expectations from the simulation study should be realistic and not exaggerated. A lot of professionals think of simulation as a "crystal ball" through which they can predict and optimize system behavior. It should be clear that simulation models by themselves are not system optimizers. They are flexible experimental platforms that facilitate planning, what-if analysis, statistical analyses, experimental design, and optimization.
- The time frame of the simulation project needs to be realistic and properly set. Insufficient time and resources at various project stages, improper work breakdown structure, and lack of project control are issues that result in project delays and low-quality deliverables. Typical PM skills are essential to execute the simulation project in an efficient manner.

- The results obtained from simulation models are as good as the model data inputs, assumptions, and logical design. The commonly used phrase of "garbage-in-garbage-out (GIGO)" is very much applicable to simulation studies. Hence, special attention should be paid to data inputs selection, filtering, and simulation assumptions.
- The analyst should pay attention to the level of detail incorporated into the model. Some study objectives can be reached with macro-level modeling while some others require micro-level modeling. The analyst should decide on the proper level of model detail and avoid details that are irrelevant to simulation objectives.
- Model verification and validation is not a trivial task. As will be discussed later, model verification aims at making sure that the model behaves according to intended model logic. Model validation, on the other hand, focuses on making sure that the model behaves as the actual system. Both practices determine the degree of model reliability and usefulness.
- The results of simulation can be easily misinterpreted. Hence, the analyst should concentrate the effort on collecting reliable results from the model through proper settings of run controls and by using the proper statistical analyses. Typical mistakes in interpreting simulation results include relying on short run time, including biases caused by model initial conditions in the results, using the results of only one simulation replication, and relying on the mean of the response while ignoring variability inherent in response.
- The analyst should pay attention to communicating simulation inputs and outputs clearly and correctly to all parties of the simulation study. Also, the results of the simulation model should be communicated to get feedback from parties on relevancy and accuracy of the results.
- The analyst should avoid using wrong measures of performance when building and analyzing the model results. Such measures should represent the kind of information required for the analyst and the decision-maker to draw conclusions and inferences on model behavior.
- The analyst should also avoid the misuse of model animation. In fact, animation is an important simulation capability that provides engineers and decision-makers with a valuable tool of system visualization. Such capability is also useful for model debugging, verification, and validation. However, some may misuse model animation by relying solely on observing the model for short-term, which may not necessarily reflect its long-term behavior.
- Finally, the analyst should select the appropriate simulation software tool that is capable of modeling the underlying system and providing the required simulation results. Criteria for selecting the proper simulation software tool typically include price, modeling capabilities, learning curve, animation, produced reports, input modeling, output analysis, and add-in modules. Simulation packages vary in their capabilities and inclusiveness of different modeling systems and techniques such MHS, human modeling, statistical tools, animation.

1.5 SIMULATION METHODOLOGY

A systematic approach should be followed when conducting simulation studies. The simulation methodology consists of five main stages, as shown in Figure 1.13.

1.5.1 Identify Problem/Opportunity

Any simulation study should start by defining the problem to be solved and improvement opportunity to be explored. System design challenges, operational problems, and improvement opportunities are the main categories of simulation problems. The simulation problem is defined in terms of study scope, study objectives, and model assumptions:

1. The study scope includes a clear description of the system problem or opportunity, the overall goal of simulation, the challenges, limitations, and issues with the current state. It starts by describing the system structure, logic, and functionality. The problem can be structured using a schematic diagram or process map to make it easy for the analyst to understand different aspects of the problem.

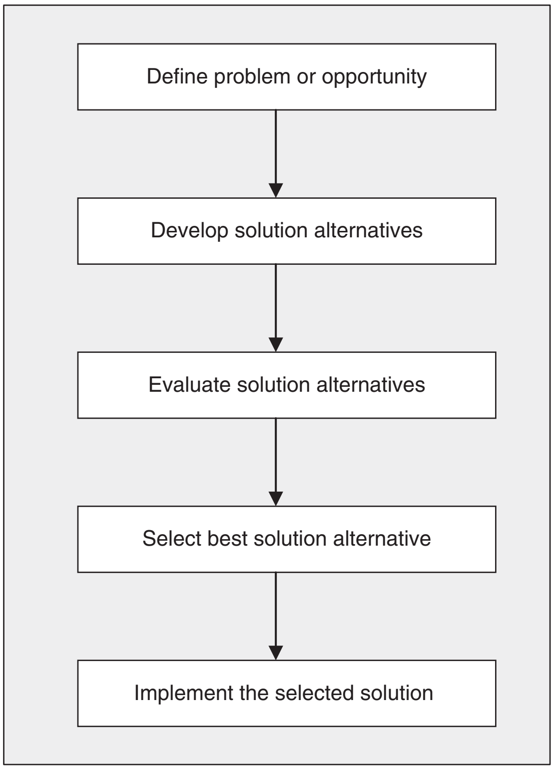

Figure 1.13 Systematic simulation approach. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

2. Based on the problem scope, the simulation study objectives are set and expressed in the most concise way possible. Defining quantifiable metrics that measure the defined objectives provides the criteria and the mechanism for solving the problem. For example, the overall goal might be to increase customer satisfaction and the specific objectives might be to reduce the customer waiting time and increase delivery speed. It is always better to define quantifiable metrics that measure the defined objectives.
3. Model assumptions are defined to narrow the scope of the problem and focus the study objectives. Most model assumptions can be related to the underlying system conditions and constraints, data, external factors, and study time and cost.

1.5.2 Develop Solution/Improvement Alternatives

Once a problem or an opportunity has been identified, solution alternatives can be proposed and developed. Defining the set of solution alternatives is approached by exploring the solution domain and listing the potential solution methods. Such alternatives are aimed to define the parameter setup, structural changes, and logic necessary to meet the problem objectives without violating defined constraints. The analyst should be able to pinpoint the differences among solution alternatives. Tables, graphs, and summary sheets can be used to present and compare the developed solution alternatives. The following needs to be considered when developing solution alternatives:

1. Simulation flexibility facilitates idea generation, brainstorming, and creative problem-solving. Through the close-to-reality representation of the system, the model can be used to observe the system behavior and predict the performance at each solution alternative.
2. The generated ideas are transformed into a set of solution alternatives. The feasibility of each alternative is checked using the model combined with cost-benefit analysis. Each feasible solution is structured in terms of a concise and specific plan that is aimed at making the change required to eliminate the problem or improve system performance.

1.5.3 Evaluate Solution Alternatives

At this stage, the selected set of solution alternatives is evaluated based on the defined objectives and ranked based on decision criteria. This includes evaluating the set of defined performance measures at each solution alternative. Simulation plays a primary role in performance evaluation under complex, dynamic, and stochastic behavior.

Defining a set of performance criteria includes providing the proper quantitative metrics that measure various aspects of the system performance. This includes monetary criteria such as cost, profit, and rate of return. This also includes technical and operational criteria such as throughput, effectiveness, and delivery speed. For example, in a manufacturing system model, system throughput is a function of buffer sizes, machines cycle times and reliability, conveyor speeds, and so on. In addition to the simulation capability of flexible programming, model counters, tallies, and statistics provide the analyst with a variety of techniques to track performance measures. Some of these criteria can be taken at a system-level such as throughput, lead-time, unit cost, and inventory level. Others can be assessed at the process or operation level such as utilization, effectiveness, and reliability. Once the set of performance measures is defined, solution alternatives are then compared and the alternative with best performance is selected.

1.5.4 Select the Best Alternative

In this stage, the best solution or improvement alternative is selected based on the simulation evaluation (values of performance measures and the overall performance). This may involve comparing the performance of solution alternatives based on multiple objectives, establishing a tradeoff among criteria, and forming an overall value function.

If the comparison is made based on a single objective, the alternative with best performance can be directly selected. In case multiple performance measures are used, multi-criteria decision-making (MCDM) techniques such as goal programming (GP) and analytical hierarchy process (AHP) are used to support the multi-criteria decision. Such methods are based on both subjective and objective judgments of a decision-maker or a group of experts in weighting decision criteria and ranking solution alternatives. An overall utility function (often referred to as a multi-attribute utility function, MAUF) is developed by combining criteria weights and performance evaluation. MAUF is used to provide an overarching utility score to rank solution alternatives. Statistical comparative analysis and hypothesis testing are also used to compare solution alternatives. The selected solution is then recommended for implementation in the real-world system.

1.5.5 Implement the Selected Alternative

Finally, the selected solution alternative is considered for deployment. Depending on the nature of the problem, implementation preparations are often taken prior to the actual implementation. As with any other project, implementing the solution recommended by the simulation study is performed in phases. PM techniques are often used to structure the time frame for the execution plan and allocate the required resources. The model can be used as a tool for guiding the implementation process at its different stages.

1.6 STEPS IN A SIMULATION STUDY

This section presents a procedure for conducting simulation studies in terms of a step-by-step approach. This procedure is a detailed translation of the systematic simulation approach presented in Figure 1.13. The details of these steps may vary from one analyst to another based on the nature of the problem and the simulation software used. However, the building blocks of the simulation procedure are typically common among simulation studies. Figure 1.14 shows a flowchart of the step-by-step simulation procedure.

1.6.1 Problem Formulation

The simulation study should start by a concise definition and statement of the underlying problem. The problem statement includes a description of the situation or the system of the study and the problem that needs to be solved. Formulating the problem in terms of an overall goal and a set of constraints provides a better representation of the problem statement. A thorough understanding of the elements and structure of the system under study often helps in developing the problem statement.

Formulating a design problem includes stating the overall design objective and the constraints on the design process. For example, the goal might be to design a MHS that is capable of transferring a certain item from point A to point B. The constraints on the design process may include certain throughput requirements, budget limitations, unit load capacity, path inclination and declination, and so on. Thus the design problem is formulated such that the defined goal is met without violating any of the defined constraints.

Similarly, formulating a problem in an existing system includes stating the overall problem-solving objective and the constraints on the proposed solution. For example, the problem might be identified as a drop in system throughput by a certain percentage. Hence, the simulation goal is set to boost the system throughput to reach a certain target. The constraints on the proposed solution may include limited capacity of workstations, conveyor speeds, number of operators, product mix, budget limitations, and so on. Thus the problem is formulated such that the defined goal is met without violating any of the defined constraints.

Finally, formulating an improvement problem may include stating the overall improvement objective in terms of multiple and often competing goals while meeting process constraints. For example, the first goal might be to reduce the manufacturing lead-time (MLT) by a certain percentage in order to apply lean manufacturing principles. The second goal may include meeting certain throughput requirements. Both goals are often subject to a similar set of process constraints such as budget limitations, variations in manufacturing operations, inventory capacity, flow requirements, and so on. Thus the improvement problem is formulated such that the two defined goals are met without violating any of the defined constraints.

1.6.2 Setting Study Objectives

Based on the problem formulation, a set of objectives can be assigned to the simulation study. Such objectives represent the criteria through which the overall goal of the study is achieved. Study objectives simply indicate questions that should be answered by the simulation study. Examples include testing design alternatives for plant operations, studying the effects of using trucks rather than rail system in a supply chain,

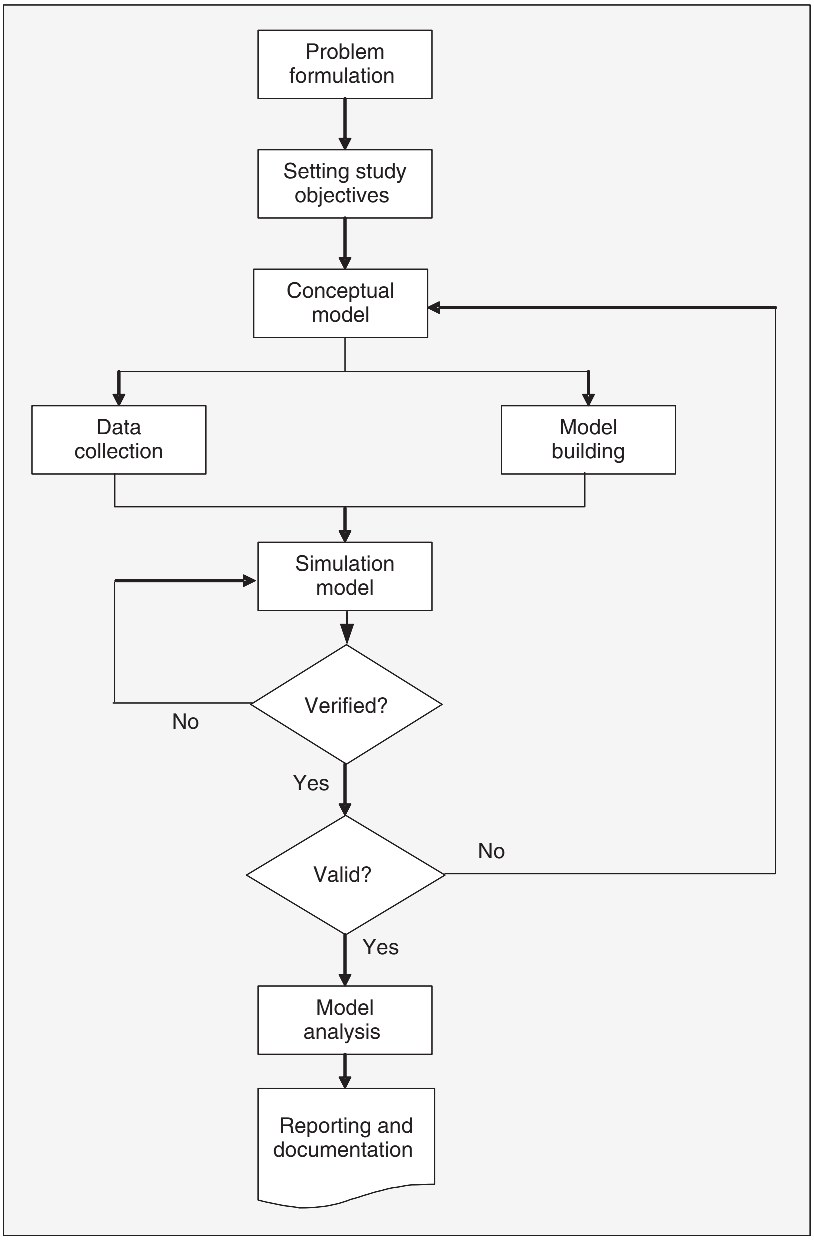

Figure 1.14 The simulation procedure. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

and so on. Differently in a call center, the objective could be to observe the duration of routing calls, and so on.

Specifying study objectives serves various purposes. First, we can decide if simulation is the right tool to solve the underlying problem. Do we have enough data to determine metrics for the defined objectives? Can we use analytical methods to answer the questions raised? Is the software tool capable of presenting and analyzing study requirements so that the study objectives are achieved? Such questions can be better answered by unambiguously stating the objectives of the simulation study.

In terms of modeling, model elements and model logic are selected and designed to provide appropriate measures of performance that quantify the study objectives. For example, to meet study objectives, an accumulating conveyor is used as a MHS. Also, statistics that represent throughput, lead-time, delays, number of carriers used, are tracked in the model over time to accumulate the required data at the end of the simulation run. Specifying study objectives provides a clear vision for designing and analyzing model outputs so that the raised questions are answered by the simulation.

1.6.3 Conceptual Modeling

Developing a conceptual model is the process through which the modeler abstracts the structure, functionality, and essential features of the real-world system into a structural and logical representation that is transferable into a simulation model. The model concept can be a simple or complex graphical representation such as a block diagram, a flowchart, or a process map that depicts key characteristics of the simulated system such as inputs, elements, parameters, logic, flow, and outputs. The conceptual model should be eventually programmable and transferable into a simulation model using available simulation software tools. Thus, a successful model concept is one that takes into consideration the method of transferring each abstracted characteristics, building each model element, and programming the conceptual logic using the software tool.

The model concept is developed taking into consideration the formulated problem and the objectives of the simulation study. Figure 1.15 shows the requirements of conceptual modeling. Study objective is the main factor that facilitates the structure of the conceptual model and the level of detail to be used in the model.

The art of conceptual modeling also requires system knowledge and model building skills. The modeler starts by establishing a thorough understanding of the simulated system whether it is a new system or an existing one. The modeler studies system inputs, elements, structure, logic, flow, and outputs and abstracts the overall structure and the interrelationships of structure elements into a conceptual model.

The model concept is presented considering the components and capabilities of the simulation environment. For example, developing a concept that includes a power and free conveyor system should consider whether there is a capability for modeling such system in the used simulation software tool. Key parameters of system elements are also specified as a part of model concept. For example, a concept of using a conveyor system to transfer entities from point A to point B should include parameters of conveyor type, speed, reliability, and capacity. Such parameters guide data collection and element selection.

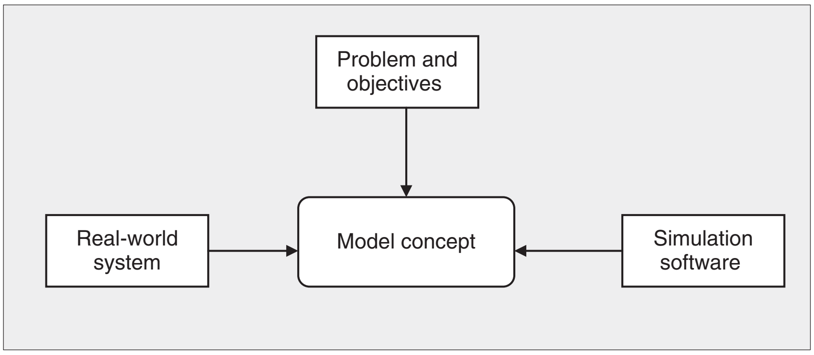

Figure 1.15 Developing a model concept. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

1.6.4 Data Collection

Simulation models are data-driven computer programs that receive input data, execute a designed logic, and produce certain outputs. Hence, data collection step is a key component of any simulation study. Simulation data can be, however, collected in parallel to building the model using the simulation software. This is recommended since data collection may be time-consuming in some cases and building the model structure and designing model logic can be independent of model data. Default parameters and generic data can be used initially until the system data is collected.

The quality of data used in the model drives the overall model quality and validity. It also impacts the accuracy of the model results and collected statistics. Hence, the term "garbage-in-garbage-out" or GIGO is often common among engineers. For a model to be representative, it has to be driven by reliable data in terms of data integrity, accuracy, and comprehensiveness.

Data elements required for constructing a simulation model are often determined based on the model concept, model structure, and the nature and type of model outcomes and statistics to be collected. For example, typical input data collected for modeling a bank operation includes customer inter-arrival time and the service time distributions.

Depending on the nature of the simulation study, model data is collected by reviewing historical data, observing and monitoring system operations, or using benchmark data assumptions. The three types of data collected in model development are shown in Figure 1.16.

Historical data is often used when modeling existing systems that have been in operation for a certain time. Examples of historical data include actual production data, maintenance data, inventory records, customer feedbacks, weekly and monthly reports on operations performance, and so on. Statistical methods of input modeling (discussed in Chapter 8) such as descriptive statistics, removing outliers, fitting data distributions, and so on play an important role in analyzing historical data.

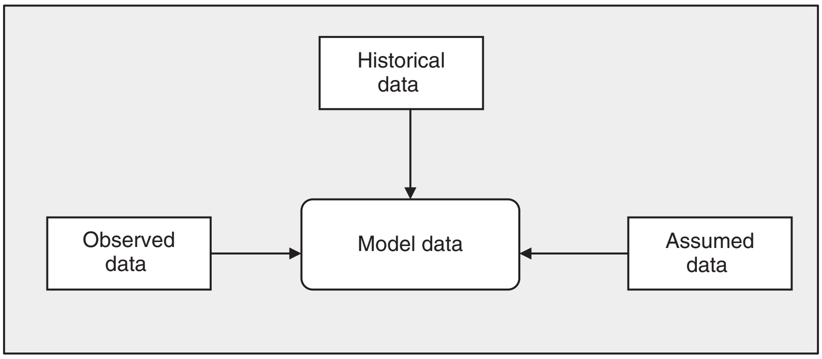

Figure 1.16 Collecting model data. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

In case historical data is not available for a sufficient period of time or is not representative of the current system, the actual performance of system can be observed to collect pertinent data. Time studies using a stopwatch or predetermined time standards (MTM, MOST, MODAPST, etc.) are common data collection methods in this regard. A stopwatch time study uses a standard form to collect data and starts by monitoring the system behavior. System monitoring includes watching the operation of a certain system element, understanding its functionality, and deciding on the parameters to be collected and the times of collecting data. The collected data should be statistically representative and should be distributed to cover the whole spectrum of system operation. Selecting the appropriate sample size and times of taking observations are essentials to obtain representative data.

Finally, when no historical data is available and it is not permissible to collect data using time studies, simulation data may be benchmarked or assumed. Educated guesses, benchmark data, and theoretical statistical models can be used to develop the data required for simulation. This often occurs when modeling new systems and testing proposed design alternatives. Experience in both modeling and system analysis often equips the simulation analyst with the knowledge and skill that is essential to provide educated guesses of different model data. Data of a similar business or manufacturing process can also be used in the model at least as a starting point. Finally, theoretical statistical distributions with estimated parameters can be used for model data. All types of assumed data are subject to modification and alteration as the model building progresses and more insight is gained into model behavior.

1.6.5 Model Building

Data collection and model building often consume the majority of the time required for completing a simulation study. To reduce this time, the modeler should start building the simulation model while data is being collected. The conceptual model can be used to construct the computer model using assumed data until the collected data

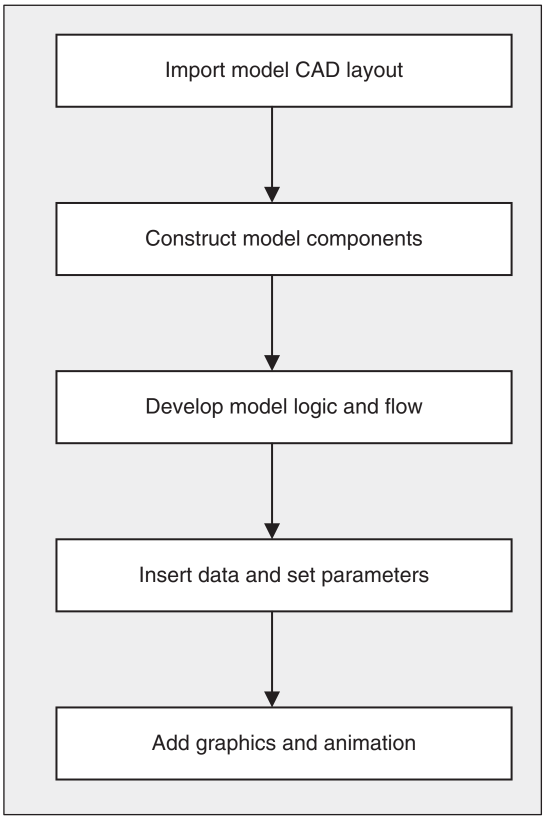

Figure 1.17 Model building procedure. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

become available. The overlap between model building and data collection does not impact the logical sequence of the simulation procedure. Constructing model components, flow of entities, and logic depends mostly on the model concept and in most cases is independent of model data. Once the model is ready, model input data and parameter settings can then be inserted into the model. Also, since a large portion of a simulation study is often spent in collecting model data, building the model simultaneously reduces significantly the overall duration of the simulation study and provides more time for model analysis and experimentation.

There is no standard procedure for building a simulation model. The procedure is often based on the modeler's approach and on the simulation software tool used. However, a generic procedure for building a simulation model effectively is expected to include key basic steps. This procedure is shown in Figure 1.17.

The model building procedure may start by importing a computer aided design (CAD) file of the system layout especially of manufacturing systems. The CAD file is a 2D or 3D layout file developed in any commercial drafting and design software tool such as AutoCAD™. Although the layout has no impact on model performance, it is a useful graphical representation of the system that provides realistic system animation, especially the 3D layout. Because of the large size of 3D files, most simulation models are built on a 2D layout. The 2D layout provides locations and spaces for different system elements such as machines, pieces of equipment, conveyors, storage, and load/unloading decks. It also provides a physical reference to the entities' flow by showing physical system elements such as aisles, walkways, walls, empty spaces, and offices. Thus, when running the model, the entities' flow between different model elements will match the flow of parts or customers in the real-world system.

Different model elements and components are built using the set of modules or building blocks defined in the simulation software selected for the study. Examples include resources or machines, queues or buffers, conveyors, loads or parts, and so on. Simulation tools provide built-in, ready-to-use, and drag-and-drop modules of such components with some graphical representation. Constructing a model component includes selecting the simulation element that best represents the component and locating the element on its actual location in the system CAD layout without connecting them.

Elements' interrelationships, decision points, and entities' flow are defined by developing the model logic. Depending on the simulation tool used, the model logic can be developed by defining input and output routing rules at each simulation element or the model logic can be written and debugged in a certain editing environment, or a combination of both. Although most new simulation software tools strive to reduce the programming effort, writing code is still necessary to implement the logic in most models.

Once the model logic is developed, one can start running the model. However, the model performance and results may not reflect the behavior of the system of interest without inserting representative data into the model and setting the parameters of different system components. Such data should be collected while the first three steps in the model building procedure are executed. As the model data become available, we start inserting such data in the model following the instructions of the simulation tool used. This is usually an easy and quick step in the model building procedure.

The last step in building a simulation model is to add the animation and graphical representations of model complements and surrounding environment. Examples may include developing an animation that reflects a plant environment, a banking system, a restaurant, and so on. Although animation does not really add to the statistical quality simulation results, it greatly helps in model verification and validation. It is also a great selling and presentation tool, especially if developed with 3D graphics. Some simulation environments allow only for 2D or $2\frac{1}{2}$D graphical representation. In addition to providing a graphics editing environment, simulation software tools also include libraries of ready-to-use graphics that suit different simulated systems for both manufacturing and services applications.

1.6.6 Model Verification

Model verification is the quality control check that is applied to the built simulation model. Like any other computer program, the simulation model should perform based on the intended logical design used in building the model. Although model logic can be defined using different methods and can be implemented using different programming techniques, the execution of the logic when running the model should reflect the way the programmer or the modeler has initially designed. Different methods are, therefore, used for debugging logical (programming) errors as well as errors in inputting data and setting model parameters. Corrected potential code and data discrepancies should always be verified by carefully observing the changes in model behavior.

To verify a model, we simply check if the model is doing what it is supposed to do. For example, does the model read the input data properly? Does the model send the right part to the right place? Does the model implement the defined production schedule? Do customers in the model follow the defined queuing discipline? Does the model provide the right output? And so on. Other verification techniques include applying rules of common sense, watching the model animation periodically during run time, examining model outputs, and asking another modeler to review the model and check its behavior. The observations made by other analysts are valuable since the model builder will be more focused on the programming details and less focused on the implication of different programming elements. In cases where model logic is complex, more than one simulation analyst may have to work on building the model.

1.6.7 Model Validation

Model validation is the process of checking the accuracy of the built model representation to the simulated real-world system. It is simply about answering the following question: does the model behave similarly to the simulated system? Since the model will be used to replace the actual system in experimental design and performance analysis, can we rely in its representation of the actual system?

Knowing that the model is only an approximation of the real-world system, key characteristics of the actual system behavior should be captured in the model, especially those related to comparing alternatives, drawing inferences, and making decisions. Hence, necessary changes and calibrations that are made to the model in order to better represent the actual system returns the modeler to the model concept. The model concept represents the modeler's abstraction of the real-world system structure and logic. Thus, if the model was not fully valid, such abstraction need to be enhanced in the model concept and then translated into the simulation model.

Several techniques are usually followed by modeler to check the validity of the built model before using it for such purposes. Examples include checking the data used in the model and comparing it to that of the actual system data, validating the model logic in terms of flow, sequence, routing, decisions, scheduling, and so on to that of the real-world system, and matching the results of the model statistics to those of actual system performance measures, and so on.

Cross validation using actual system results and running certain what-if scenarios can be also used to check model validity. For example, the last year's throughput data can be used to validate the throughput number produced by the model for the same duration and under similar defined conditions. We can also double the cycle time of a certain operation and see if the produced system throughput is impacted accordingly or if the collected MLT reflects this increase in cycle time.

1.6.8 Model Analysis[1]

Having a verified and validated simulation model is a great opportunity to analysts since it provides a flexible platform to run experiments and apply various types of engineering analyses effectively. With the latest advances in CPU speed and capacity, even large-scale simulation models of intensive graphics can be run for several replications in a relatively short time. Hence, it takes only a few minutes to run multiple simulation replications and for long periods of time in most simulation environments.

As shown in Figure 1.18, model analysis often includes statistical analysis, experimental design, and optimization search. The objective of such methods is to analyze the performance of the simulation model, compare the performance of proposed design alternatives, and provide best model structure and parameter settings that lead to best level of performance. Statistical analyses include representing performance with descriptive statistics such as mean, variance, and confidence intervals. It also includes performing different statistical tests for hypothesis testing and comparative analysis. Experimental design with simulation includes conducting a partial or full factorial design of experiments (DOEs) to provide best settings to model control variables. Finally, search algorithms include the application of optimization search methods such as exhaustive search, genetic algorithm, simulation annealing, and tabu search to optimize a selected objective function.

The speed and flexibility of the simulation model also facilitate conducting experimental design and running optimization algorithms. Also, the availability of different methods and analyses in commercial simulation software tools facilitates the application of output analyses. Most full versions of simulation packages are equipped with modules for statistical analyses and add-ins of experimental design and optimization methods.

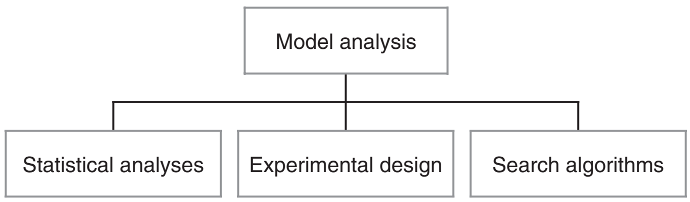

Figure 1.18 Continuous improvement with simulation. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

[1]Chapter 9 provides further details on analyzing simulation outputs

1.6.9 Study Documentation

The final step in a simulation study is to document the study and report its results. Proper documentation is crucial to the success of a simulation study. The simulation process often includes communicating with many people, writing complex logic, encountering enormous amount of data, conducting extensive experimentation, and going through several progress reviews and milestones. Thus, without proper documentation, the analyst loses track and control of the study and can neither deliver the required information nor meet the study expectations. This often results in an inaccurate simulation model with poor results, inability to justify model behavior and explain model results, and losing confidence in study findings and recommendations.

Documenting the simulation study is the development of a study file that includes the details of each simulation step. Comprehensive documentation of a simulation study comprises three main elements: detailed documentation of the simulation model, the development of an engineering simulation report, and the presentation of the simulation results to customers and partners of the simulation project. Figure 1.19 presents the three elements of documenting a simulation study.

The first element in documenting the simulation study includes the documentation of the simulation model. This includes documenting both the concept model and the simulation program. Documenting model concept includes documenting the system process map, flowcharts, block diagrams and sketches that explain the model concept, model elements, modeling approach, and model logical design. Simulation model documentation includes a description of model structure, program details and flowchart, code and routines explanation using within-code comments and notes, and an explanation of the applied scheduling and sequencing rules, routing rules, and the decision-making process followed at decision points within the model. Model reporting also includes a statement of model assumptions and a description of model inputs and outputs. By reviewing such documents, we should be able to understand how the model reads inputs, processes the designed logic, uses available simulation elements, and provides the required outputs. Such documentation facilitates making model changes, explaining data used in the model, debugging code and logic, understanding model behavior, and interpreting model results.

The major deliverable of the simulation study is the development of a simulation report. This includes a formal engineering report and supplemental materials of the

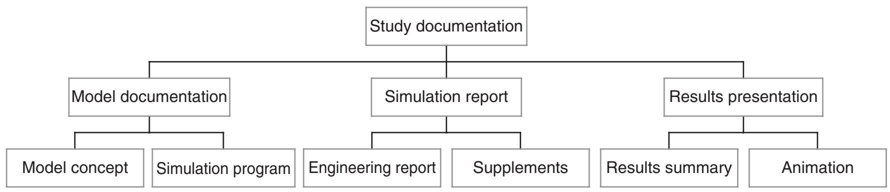

Figure 1.19 Documenting a simulation study. El-Haik, B., Al-Aomar, R. (2006). Reproduced with permission of John Wiley & Sons, Inc.

- The simulated system
  - A.Background
  - B.System description
  - C.System structure
- The simulation problem
  - A.Problem formulation
  - B.Problem assumptions
  - C.Study objectives
- The simulation model
  - A.Model structure
  - B.Model logic
  - C.Model inputs
  - D.Model assumptions
- Simulation results
  - A.Experimental design
  - B.Results analysis
  - C.Results summary
- Study conclusion
  - A.Study findings
  - B.Study recommendations
- Study supplements
  - A.Drawings and graphs
  - B.Input data
  - C.Output data
  - D.Experimental design
  - E.Others

Figure 1.20 Elements of a simulation report.

simulation process. Such a report is the communication media between the simulation analyst and the other parties involved in the simulation study. Some of those parties may not even see the model or talk to the simulation analyst; hence the report should be comprehensive, unambiguous, and understandable by people of various skills and backgrounds.

The elements of a formal simulation report include description of the underlying system, formulation of the simulation problem, a statement of study objectives and assumptions, sources of model data, a description of model structure, and summary of simulation results, findings, conclusions, and recommendations. Supplement materials include the data inputs used, system and model graphs, sketches out and drawings, details of experimental design and output analysis, and a printout of simulation results. It is also common among simulation engineers to provide a one-page summary of key information provided by the simulation study. This kind of executive review provides concise and focused information for decision-makers, managers, and firm executives who may not be interested in the details of model inputs, structure, and results analysis. Figure 1.20 shows elements of a simulation report.

The simulation study is often concluded by presenting the simulation results to customers, managers, and different parties of the simulation project. Such presentation often includes a summary of the simulation results (summary of study steps, results and findings, conclusions, and recommendations). The presentation should also include running animations, movies, or snapshots of the simulation model at different situations. Such animations help explain the study results and aid the analyst in selling proposed design, comparing design alternatives, and convincing management of the viability of proposed solutions and plans of actions. Communication and presentation skills of the simulation analyst play a major role in developing a successful presentation and gaining managerial support to implement the recommendation of the simulation study. Chapter 11 provides more in-depth explanation of the simulation project steps as they relate to the success of a simulation project.

1.7 SIMULATION SOFTWARE

Simulation software tools are common nowadays among analysts and engineers in various types of applications and for different purposes. The major benefit simulation packages provide to practitioners is the capability of dynamic system modeling under close-to-real-world conditions. Processes of transactional nature in both manufacturing and services often behave in a complex, dynamic, and stochastic manner. Such capability, therefore, allows analysts to better express new concepts and designs for transactional processes, measure process performance with time-based metrics, conduct statistical analyses to capture variability, and improve/optimize new and current systems.

Indeed, many simulation software tools were initially built by practitioners to model some particular real-world systems such as manufacturing systems, health-care systems, supply chain systems. They were built from entities and processes that mimic the objects and activities in the real system. To further meet the needs of practitioners and engineers, simulation vendors developed and integrated modules into their software products to simplify model building, facilitate model customization, allow for the creation of impressive animation, and enable analysts to conduct statistical analyses and run optimization searches. Some of these modules include Six Sigma definitions and calculations, lean manufacturing measures, and provide links to spreadsheet and statistical software such as MINITAB to directly run experimental designs and output analyses.

The spectrum of simulation software currently available ranges from procedural languages and simulation languages to special “simulators.” The vast amount of simulation software available can be overwhelming for the new users. Common software packages for process simulation include Arena, AutoMod, WITNESS®, Simul8, Enterprise Dynamics, and ProModel. The WITNESS® simulation package in particular will be the focus of this book because of its simple-to-understand and effective features as well as its popularity. In addition, WITNESS® includes modules for Six Sigma calculation and optimization as well as a direct link to the MINITAB statistical package.

1.7.1 WITNESS® Simulation Software

WITNESS® is one of the leading software products in the field of visual interactive simulation. More than 6500 licenses have been sold worldwide across the manufacturing, service, and noncommercial sectors. The system is also widely used within academia. A full range of maintenance, consultancy, and training services are provided for support of the software.

WITNESS® gives the power and flexibility to model any working environment, simulates the implications of different business decisions, and is also designed with simplicity in mind. Some of the key features of WITNESS® are:

- Simple and powerful building block design
- Modular and hierarchical structure
- Discrete event and Continuous simulation modeling capabilities
- Easy to use standard Windows PC implementation
- Interactive model-development user interface
- Powerful range of logic and control options
- Elements for discrete manufacture, process industries, business process re-engineering (BPR), e-commerce, call centers, health, finance and government.
- Comprehensive statistical input and reports
- Quality graphical displays
- Ease of linkage to databases (ORACLE, SQL Server, Access, etc), direct spreadsheet links in/out, XML save formats, HTML reports, links from partner BPM and CAD applications, and more

The WITNESS® family of simulation products includes the following:

- Manufacturing, Service, and Process editions of the simulation modeling software
- Optional fully integrated 3D/VR views or Post Processed VR including headsets, powerwalls, and more
- Optional intelligent optimization of models—unique algorithms to find the best answer fast
- Optional Developers' edition (SIMBA) to develop simulation applications with custom interfaces (includes a full object model, ActiveX displays, and special viewer software)
- A range of Microsoft VISIO linked solutions
- A range of direct and graphical CAD linkage solutions
- HLA compliance options for military and other applications

WITNESS® 2008 is Lanner Group's simulation software package. It is the culmination of more than two decades' experience with computer-based simulation.

A WITNESS® model consists of a collection of WITNESS® elements, together with the control logic (rules and actions) that ensures that the model performs in the same way as a real-life system. WITNESS® elements can be combined into modules which in turn can be used in other models.

WITNESS® provides elements that represent tangible objects:

- Elements that move through the model, being processed (parts and fluids).
- Elements that transport parts and fluids (conveyors, track, vehicles, carriers, paths, and pipes).
- Elements that store parts and fluids (buffers and tanks).
- Elements that process parts and fluids (machines and processors).
- Elements that are predominantly associated with parts (machines, buffers, conveyors, tracks, vehicles, paths, and parts themselves) are grouped together as discrete elements. Labor is also treated as a discrete element.
- Elements that are predominantly associated with fluids (tanks, processors, pipes and fluids themselves) are grouped together as continuous processing elements.
- Parts can contain fluids, and it is possible for machines to fill parts with fluids and empty them again, so WITNESS® is capable of modeling on a volume basis as well as on a quantity basis.

WITNESS® also provides elements that represent intangible objects:

- Elements that represent shift patterns (shifts).
- Elements that provide statistical variation (distributions).
- Elements that provide a source of data outside the model (files).
- Elements that are referenced in the model's control logic (variables, attributes, and functions).

Finally, WITNESS® provides elements that enhance the presentation of the model and supply information about the state of the model while it is running:

- Charts
- Histograms
- Time series

1.8 SUMMARY

Basic concepts of simulation modeling include the system concept, model concept, and simulation concept. Systems include inputs, entities, relationships, controls, and outputs. System elements should be tuned toward attaining an overall system goal. A system model is a representation or an approximation of the real-world system.

Models can be physical, graphical, mathematical, and computer models. The goal of modeling is to provide a cheaper, faster, and easier-to-understand tool for system analysis, design, and improvement. Simulation is the art and science of mimicking the operation of a real-world system on the computer. It is aimed at capturing complex, dynamic, and stochastic characteristics of the real-world process, where other types of models fall short. Based on the type of state variables, computer simulation models can be discrete, continuous, or combined. They can also be deterministic or stochastic based on randomness in the system. Finally, they can be static or dynamic based on the changes of system state. The focus of this book is the DES models that represent systems' complex and stochastic behavior dynamically in terms of discrete and continuous state variables. DES is justified by its capability, software availability, and wide spectrum of real-world applications. Simulation is an engineering methodology for system design, problem-solving, and continuous improvement. The WITNESS® simulation software is the particular focus of this book due to its simple yet comprehensive modeling capability and wide range of applications.

QUESTIONS AND EXERCISES

Several questions addressing the basic concepts of simulation modeling include the system concept, model concept, and simulation, and how system thinking is applied to: a factory, a clinic, and a restaurant

1. What is a system? What is a system model?
2. Mention three examples of a production system?
3. Mention three examples of a service system?
4. Identify the *I–P–O* to the system examples in questions 2 and 3.
5. Explain why the *I–P–O* flow is not sufficient to develop a sustainable system. Address the *I–P–O* limitations of the system examples in questions 2 and 3.
6. Identify the *G*, *RL*, and *C* to the system examples in questions 2 and 3. Are these systems now complete and can perform in a sustainable manner?
7. What is modeling? How does it differ from simulation?
8. Give three examples of physical models. Explain the limitations of these models in real-world applications.
9. Give three examples of graphical models. Explain the limitations of these models in real-world applications.
10. Give three examples of mathematical models. Explain the limitations of these models in real-world applications.
11. What is the difference between product and process models?

12. What are the advantages of using computer models? What are the limitations of computer models?
13. What is simulation? What is computer simulation?
14. What are the elements of the simulation process? Which one is most critical?
15. What are the types of computer simulation models? Which type in particular is more relevant to modeling production and business systems?
16. Distinguish between deterministic and stochastic systems using examples.
17. Distinguish between discrete and continuous systems using examples.
18. Distinguish between static and dynamic systems using examples.
19. Apply the discrete, stochastic, and dynamic characteristics to the example systems in questions 2 and 3.
20. Justify the use of simulation for modeling the example systems in questions 2 and 3.
21. Apply the simulation precautions to the example systems in questions 2 and 3.
22. Apply the simulation methodology to one of the example systems in questions 2 and 3.
23. Apply the step-by-step simulation procedure in Figure 1.14 to one of the example systems in questions 2 and 3.
24. Distinguish between model verification and validation.
25. Explain the importance of documenting simulation studies.
26. Review three commercial simulation software and compare them based on:
    - **a.** Ease of use
    - **b.** Features
    - **c.** Animation
    - **d.** Applications
27. Visit WITNESS® software website and compare its features to simulation tools in question 26.

BIBLIOGRAPHY

Altiok T, Melamed B. *Simulation Modeling and Analysis with ARENA*. Academic Press; 2007.

Balci O. Validation, verification, and testing techniques throughout the life cycle of a simulation study. Annals of Operations Research 1994;53:121–173.

Balci O, Ormsby WF. Conceptual modeling for designing large-scale simulations. Journal of Simulation 2007;1:175–186.

Banks J, editor. *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*. Wiley-Interscience; 1998.

Banks J, Carson J, Nelson BL, Nicol D. *Discrete-Event System Simulation*. Prentice-Hall International Series in Industrial and Systems. 4th ed. Prentice-Hall; 2004.

Browne J, Rathmill K. *Simulation in Manufacturing*. IFS Publications/Springer-Verlag; 1998.

Checkland PB. *Systems Thinking, Systems Practice*. Chichester, UK: Wiley; 1981.

Chung CA. *Simulation Modeling Handbook: A Practical Approach*. CRC Industrial and Manufacturing Engineering Series. CRC; 2003.

El-Haik B, Al-Aomar R. *Simulation-Based Lean Six-Sigma and Design for Six-Sigma*. John Wiley; 2006.

Fishman G. *Discrete-Event Simulation*. Springer; 2001.

Harrell C, Ghosh BK, Bowden R. *Simulation Using Promodel*. McGraw-Hill Series in Industrial Engineering and Management Science. 3rd ed. McGraw-Hill; 2000.

Kelton W. *Simulation with Arena*. McGraw-Hill Series in Industrial Engineering and Management. 4th ed. McGraw-Hill; 2006.

Law A. *Simulation Modeling and Analysis*. 4th ed. McGraw-Hill; 2006.

Law A, McComas MG. How to build valid and credible simulation models. In Proceedings of the 2001 Winter Simulation Conference, 22–28; 2001.

Leemis LM. *Discrete-Event Simulation, a First Course*. Prentice Hall; 2006.

Obaidat MS, Papadimitriou GI. *Applied System Simulation: Methodologies and Applications*. Springer; 2003.

Robinson S. *Simulation: The Practice of Model Development and Use*. Chichester, UK: Wiley; 2004.

Robinson S. Conceptual modeling for simulation: issues and research requirements. In Proceedings of the 2006 Winter Simulation Conference, 792–799; 2006.

Robinson S. Simulation projects: building the right conceptual model. Industrial Engineering 1994;26(9):34–36.

Sargent RG. Validation and verification of simulation models. In Proceedings of the 2004 Winter Simulation Conference, 17–28; 2004.

Skyttner L. *General Systems Theory: Problems, Perspective, Practice*. World Scientific Publishing Company; 2006.

Sterman JD. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. McGraw Hill Higher Education; 2000.

Zeigler B, Kim TG, Praehofer H. *Theory of Modeling and Simulation*. 2nd ed. Academic Press; 2000.