CHAPTER 7

SERVICE ENTERPRISE MODELING

Yuehwern Yih, PhD
School of Industrial Engineering, Purdue University, West Lafayette, Indiana, USA

Alok Chaturvedi, PhD
Krannert School of Management, Purdue University, West Lafayette, Indiana, USA

Introduction

Service vs. Manufacturing System Modeling

System modeling is a common approach to assist decision makers to make choices in system design, monitoring, and control to achieve desired level of performance before implementing to the real system. The modeling methods are heavily dependent on the purpose of modeling, intended use of the model, as well as the characteristics of the system. The modeling of manufacturing system focuses mostly on the tangible goods and their flows and associated processes. Some of the service systems are modeled in a similar fashion as a manufacturing system because they engage in the distribution, handling, and transporting of tangible goods and those activities are the primary focus in the model. Good examples include parcel pickup and delivery services, warehouses and fulfillment centers, and airline industry. The modeling approach for service systems start to diverge when the system is providing intangible products and the process is ill-structured or ill-defined.

Purpose of System Modeling

It is important to know the purpose and the intended use of the developed model before the modeling process. This will help to determine the appropriate levels of abstraction and granularity, as well as the scope of the model. System models are to mimic the behaviors or responses of the system to help decision makers understand the system or gain perspectives, and they are not meant to create a duplicate of the entire system. There are always areas that will be neglected or be represented in an aggregated fashion in the model without the details. The system model remains useful if those areas do not create significant effects on the system. Caution should be taken when the system is complex and ill-structured because the relationships and interactions among entities in the system may not be apparent and the aggregation may introduce significant deviation or bias to the model.

The purpose of system modeling is commonly related to system improvement, which in general can be achieved by:

- Understanding the system responses to changing conditions, changing system entities, or system structures
- Predicting system behaviors based on new designs or new policies
- Observing emerging behaviors

Modeling Principles

Levels of Abstraction

Abstraction is to aggregate selected system entities and their associated interactions in the model development. Some form of abstraction is inevitable to make the system modeling possible. The level of abstraction affects the use (benefits) of the developed model and the effort (costs) required for its development. For example, when we model a hospital to study the patient flow pattern, we may model the radiology department by a simple time delay for patients who need x-ray, or model the entities (such as nurses, technicians, radiologists, equipment, rooms, etc.) and their associated activities and interactions in that department. The latter model provides more insights, but it also requires more effort in data collection and model development. It is also common to have mixed levels of abstraction in one system model where the focused areas are detailed and others are aggregated. Judicious abstraction that is congruent with the purpose of the model is a key factor for successful system modeling.

Levels of Granularity

Granularity is often referred to the relative scale of time or space. The level of time granularity is defined by the shortest time interval (unit) the model can advance, sometimes called the "tick" in computer-based modeling. If the model is developed based on a "tick" of an hour, the model will progress in an hourly fashion discretely and it will not provide meaningful information within the hour. If we model the same system in a "tick" of a second, the computation time will increase by 360 times.

Similarly, the level of space granularity is defined by the smallest "unit" length used in the model if a physical space is presented in the model. A finer grid will give more precise location of entities in the space, and of course, require more computational resources (memory and time). The space granularity is also dependent on the moving speed of each entity in the system as well as the level of time granularity.

Forward vs. Inverse Problem

Mathematically, any phenomenon can be modeled as a forward problem or as an inverse problem. Forward modeling begins with known domain theories. Parameters that drive the diverse behaviors of entities are known. Entities are configured and then their behavior is projected forward in time using models that implement the given theories. For example, forward modeling is used to model changes to an individual's attitude toward other entities, such as leaders and governments. Theories for the political and social sciences domain explain how individuals update their attitudes due to information they receive from media they subscribe to, leaders they support and organizations and social groups they belong to.

Inverse modeling begins with observed behavior and phenomena. Theories are hypothesized to explain how the observed behavior emerges due to underlying, entity-level parameters. Data and assumptions are used to parameterize entities in the system. Assumptions and model variables are calibrated until the modeled system exhibits the desired observed behavior. An example of inverse modeling is the formation of a population group's public opinion of national government in the context of economic activities.

Verification and Validation

Model verification and validation are essential steps in system model development to ensure the integrity of the model. Verification is to ensure the model is implemented (or programmed) correctly based on design specifications in terms of processes, logics, algorithms, rules, etc. This process is often performed throughout the model development process to verify that each component or subsystem is implemented accurately based on the designed specifications. This is mostly an internal audit or quality assurance process.

The validation process usually starts after verification completes. Validation is used to determine if the developed model meets its intended requirements or purpose in terms of addressing the right questions and providing accurate information. There are several testing techniques for validation. Mostly the testing process consists of generating test cases, collecting data for those cases, and comparing the results from the model with the collected data. It is a great challenge to validate a model when the data for test cases are not available or cannot be obtained, or when the system doesn't exist yet.

Verification and validation are time- (and cost-) consuming tasks, but they are important to ensure the usefulness of the model. However, practically, very few verified or validated models exist. Most developers stop the verification and validation process when they run out of funds or time. This results in a model that passes all the tests, not necessarily a verified or validated model. It is important to gain insights of the model limitations and capabilities during the validation process because this will help interpret the outcomes of the developed model.

Special care needs to be taken to validate inverse class of problems. Inverse models are validated using likely sufficient conditions for the observed behavior being modeled. For example, it may be believed that an economic downturn has a significant negative impact on the public opinion of a national government. In such a case, we can inject an economic depression into the system and validate that the public opinions of various population groups drop. Validation may be applied at multiple levels in terms of the number and type of entities, depending on the complexity of the system and the availability of data or subject matter expertise, such as localized behavior of a small group of homogeneous entities, behavior of a small group of heterogeneous entities, and all entities of the system.

Uncertainty Quantification

When analyzing computational models, analysts must gain an understanding of how uncertainties in both models and the data feeding them affect results. In models, there are parametric uncertainties associated with errors in the numerical values of model parameters. There are also nonparametric uncertainties associated with errors in the model form (interconnects, linear vs. nonlinear). In data, there are uncertainties associated with the data source (sensor or archives), the timing of the data (real-time or historical), and rendering the data in a form that is conducive to quantitative analysis (semantics of parameters). Computational models must accurately reflect both the measured data and the uncertainties in that data. For low-order models and stationary time series, it is usually sufficient to characterize the mean and variance. However, for models with a very large number of parameters and data sources, multivariable statistics are used in a recurrent manner to track changes as the real world evolves.

Multivariable analysis is done to verify existing sources of data and identify weaknesses that reveal the need for additional sources of data. For example, high variance may be detected in a measurement system (sensor) relative to all others, suggesting that this measurement system is significantly biased. It is also important to consider the effects of acquisition time and rate in constructing and analyzing the data. For example, certain sensors may be more active over certain periods of time leading to a greater number of samples. During other time periods, these same sensors may report less frequently, leading to a potential for aliasing in the data sources.

Output data may also be analyzed to verify modeling assumptions, such as weak coupling between two separate measurement systems. This analysis provides guidance on how to interpret model outputs in ways that honor variations in measured variables in the real world. For example, different classes of models may produce information at different times, scales, and semantics, leading to significant variance. By understanding these sources of variation up front using multivariable statistical analysis, interpretation systems can be designed to anticipate and manage these differences in opinion using forward models.

Modeling Approaches and Case Studies

In this section we present several system modeling approaches and case studies that use those approaches.

Mathematical Modeling

Mathematical modeling is one of the techniques used to describe a system by mathematical formulation. Mathematical models are used in many disciplines to describe systems of various natures, for example, in the medical and engineering fields. Some of the commonly used mathematical models include linear programs, petri nets, and dynamic programs. The following section provides an outline regarding several types of mathematical models.

Linear Programs. Linear programming is a technique for optimizing (maximizing or minimizing) a linear objective function, subjected to a set of linear inequality constraints. It is the most commonly applied form of constrained optimization (Chinneck, 2001). The main elements in a linear program include decision variables that represent characteristics of the system that the modeler can control. It is usually the modeler's goal to find the values of these variables that provide the best value of the objective function. The objective function is a mathematical expression that could represent profit, loss, utility, reward, etc., depending on the goal of the modeler and the nature of the problem. Constraints are mathematical expressions that impose limits on the possible solutions.

Petri Net. Another class of modeling technique is the use of petri nets. Originated by Carl Adam Petri, the petri net is a graphical and mathematical modeling tool commonly used for describing and studying information-processing systems that are characterized as being concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic (Murata, 1989). As a graphical tool, petri nets can be used as a visual-communication aid similar to flowcharts, block diagrams, and networks. As a mathematical tool, it is possible to set up state equations, algebraic equations, and other mathematical models governing the behavior of systems.

The components of a petri net include places (e.g., condition of a system), transitions, and arcs that connect places to transitions. Some recent extensions to the general petri net classification include colored petri nets, timed petri nets, and modular petri nets (www.cse.fau.edu/~maria/COURSES/CEN4010-SE/C10/10-7.html). Petri net and its related development are used in a wide range of applications, including communication-based systems, concurrent object-oriented programming (Agha et al., 2001), intelligent task planning (Cao and Sanderson, 1996), and supply chain modeling and analysis (Nikolic et al., 2004).

Dynamic Programming. Another class of mathematical approach to problem-solving is dynamic programming. Dynamic programming is an approach to finding the optimal decisions one after another in a system that exhibits properties of overlapping subproblems, where each subproblem has a feasible optimized solution. Dynamic programming is also applicable to problems where time intervals are not consistent. An example of a methodology that uses dynamic programming is the *Markov decision process* (MDP). The MDP is a discrete time stochastic control process where the process modeled by the MDP exhibits the Markov property. This property states that the transition from one system state to the next state is dependent only on the current state, not the previous states or actions. The MDP has a wide range of applications, including in the areas of finance, communication networks, water reservoir (Feinberg, 2002), and space science (Chiam et al., 2008, 2009a, 2009b). The next section outlines a study of the NASA system where the MDP is applied.

A Case Study of the Advanced Life Support/NASA Center of Research and Training. As described in the previous section, the Markov decision process has a wide range of applications. This section describes how it is used in a research study performed within the ALS/NSCORT (Advanced Life Support/NASA Center of Research and Training).

Due to the long distance between the Earth and location of future space missions (e.g., Mars), supplying the people on these missions with food, air, and water gives rise to the need for regenerative technology development, as transportation of such resources is limited and expensive. Hence, the major focus areas covered by the ALS/NSCORT include developing efficient treatment and resource-recovery options for solid, liquid, and gaseous human, crop, and food-process wastes, effective food-processing and food-safety-testing procedures, low-energy crop-production technologies, and global systems-analysis procedures.

FIGURE 7.1 PROCESS MAP OF THE ALS/NSCORT SUBSYSTEMS, COMPONENTS, AND RELATIONSHIPS AMONG COMPONENTS.

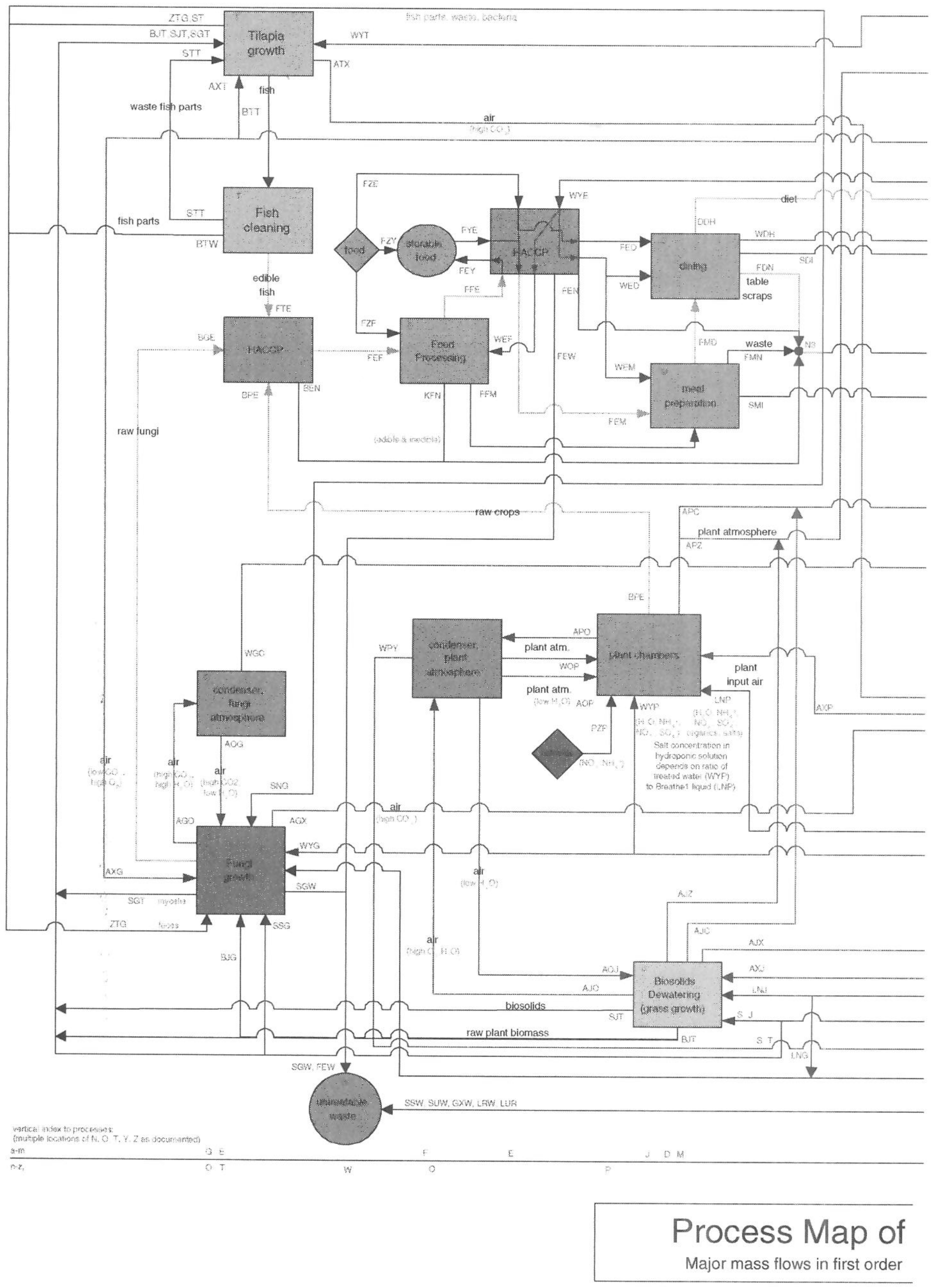
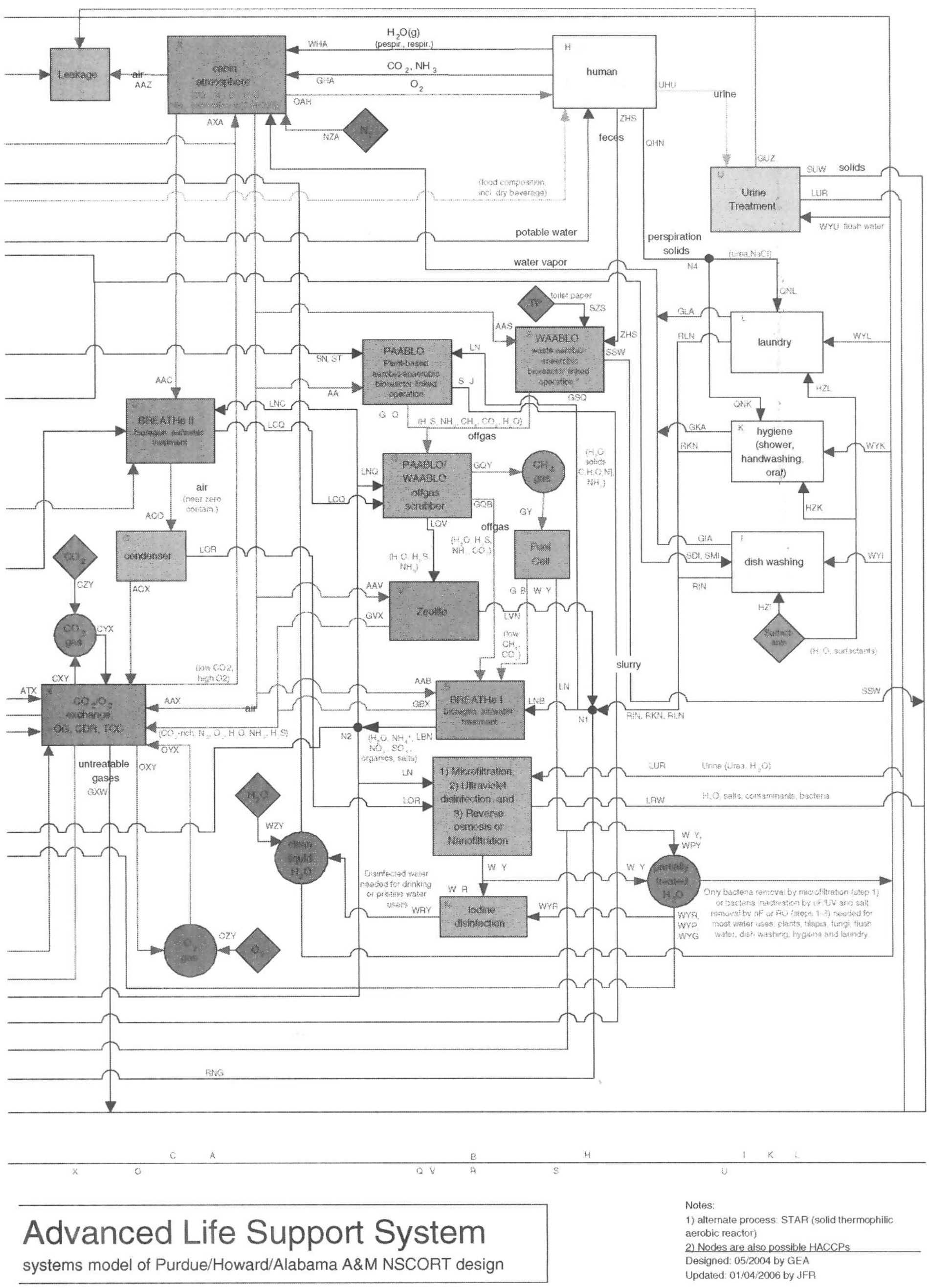


FIGURE 7.1 (CONTINUED)



The Advanced Life Support system proposed consists of 6 main subsystems: water, air, biomass, food, thermal, and waste. Figure 7.1 illustrates the components of these subsystems as well as the relationships among the components. The subsystem that the research study focuses on is the water subsystem, as water has a direct impact on the crew members' health. Various water treatment technologies are being developed for this purpose, such as bio-regenerative technologies that make use of bacteria to break down contaminants, physico-chemical (PC) technologies that make use of physical and chemical processes for purifying dirty water, as well as hybrids of the two types of technology. However, there is a lack of study and development of dynamic control strategies. One of the main goals of this study is to develop such strategies to identify system conditions, and dynamically select the best policy to be applied that will allow the system to behave at a level that ensures a "healthy" system condition. Meanwhile, these strategies should be developed in such a way that they are function-driven instead of technology-driven; hence they are applicable to the overall system, regardless of the choice of technology used. To achieve this goal, a theoretical water-treatment system based on the NASA Baseline Values and Assumptions Document (BVAD) (Hanford, (2004)) has been developed.

In this system, water is consumed by crew members for potable and hygiene purposes. Potable water includes drinking water and water in food. Hygiene water includes water used for laundry, urinal flush, dish washing, oral hygiene, and showers.

To have an understanding of the system conditions, parameters that represent various aspects of the system are defined. These parameters measure conditions such as clean water available for usage, dirty water in the system, water deficiency, water overflow, and demand for water in both subsystems. The demand for water in both subsystems follows the water consumption requirements developed by Ang (2006) and Ang and Yih (2008). These parameters are chosen because of their importance and direct impact to the behavior of the system. The values of these parameters are captured at the beginning of every hour. The combination of each of the parameters forms the state vector, which indicates the overall condition of the system. Based on this state vector, an appropriate policy is then applied to the system. An action space, which consists of all the available policies, is defined. These policies are the treatment efficiencies of both the potable and hygiene water subsystem.

Because of the complexity of the system, the parameters are defined in such granularity as to capture as much detailed information as possible about the system. As a result, a large state space is formed. To capture as many states as possible, the random walk (also known as drunkard's walk) algorithm is used. This algorithm randomly chooses a policy to apply to the system regardless of system state to study the change in system behavior when various policies are applied.

A cost-and-reward structure is also defined. Every state has an associated reward. States where water deficiency is minimal receive a higher reward value (defined as reward of the state) compared to those where water deficiency exceeds a predetermined threshold value. Every policy is also associated with a cost. It is assumed that the higher the treatment efficiency, the higher the cost. This is because higher treatment efficiency requires more power, crew time, and resources. The net reward at a state is then defined as the difference between the reward of the state and the cost of policy used.

The Markov decision process is chosen as the appropriate modeling approach to this system for various reasons. First, the system under study consists of random processes while crew members consume clean water and produce dirty water. Random processes are also present within the treatment technologies while dirty water is being treated. Second, the transition of system state follows the memoryless property, which states that the transition is dependent on the current system state, but not the previous states visited by the system. Third, the cost and reward structure is needed and is defined to penalize encourage respective system behavior. Lastly, policies are defined to be applied to the system as appropriate.

Based on the system state transitions, transition probabilities are captured. These probabilities describe the chance of the system transiting from one state to another given the current state, while applying a particular policy. The transition probabilities, together with the net reward, serve as an input to the *policy iteration algorithm* (Howard, 1960). This algorithm, which consists of policy evaluation and policy improvement, is used to obtain the "best" policies to be applied to the system at a given system state. Various other solution algorithms do exist, such as the value iteration algorithm (Bellman, 1957).

A computer simulation is built to simulate the crew members' consumption of clean water and production of dirty water in this system. Two scenarios are studied. In the first scenario, a baseline policy is defined as a treatment efficiency of 75 percent for both the potable and hygiene water subsystems. In the second scenario, the baseline policy is modified into different efficiency levels. The state transition of the second scenario is also modeled as a Markov process, and the decision of choosing the best policies to apply is modeled as a Markov decision process, with the properties described above. A comparison of these two scenarios shows that the second scenario produces a higher overall system reward than the first scenario. This illustrates an application where the MDP is more superior to a system that does not follow such an approach.

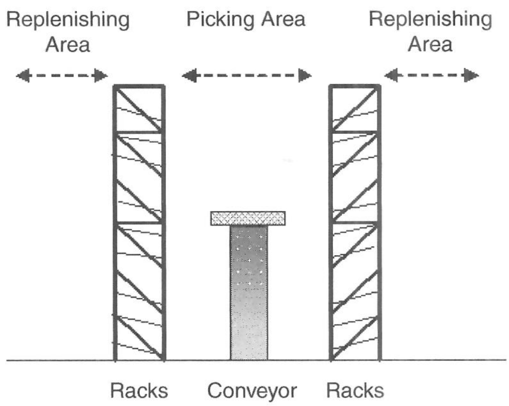
Discrete Event Simulation (DES)

In *discrete event simulation* (DES), the operation of a system is represented as a chronological sequence of events. Each event occurs at an instant in time and marks a change of state in the system (Robinson, 2004). Simulation can be viewed as an input and output experiment within a controlled environment. The inputs are termed as parameters and are usually multivariate. The outputs are the performance measures that are stochastic in nature and the objective of the model. (Marc, 1987)

Discrete event simulation, having roots in the manufacturing industry, has been widespread across the service industry. In the freight industry, a simulation model was developed to analyze the daily taxi and takeoff operation of United Parcel Service (UPS) aircraft at the Louisville International Airport. Inputs to the model include aircraft schedules, runway information, and flight patterns. Outputs include aircraft departure statistics for each flight and runway utilization. The model assists planners in developing aircraft departure schedules that minimize taxi and ramp delay times (Ottman et al., 1999).

Applications can also be easily found in the health-care area because DES provides an opportunity for decision makers to test their design without interrupting the daily operation. A comprehensive survey on simulation application in 21 health-care areas was completed by England and Roberts (1978). Ninety two simulation models were cited out of twelve hundred models reviewed. Among the cited papers, simulation models were used to model laboratory studies, emergency services, and the national health-care system. The advantages and

FIGURE 7.2 SIDE VIEW OF ONE MODULE.



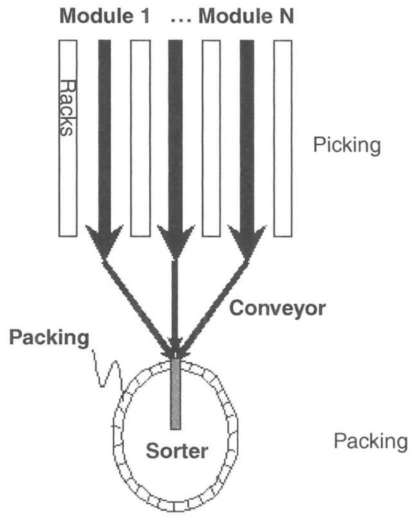
disadvantages of DES are well documented in Banks et al. (2000), Law and Kelton (2000), Schriber (1991), and Pooch and Wall (1992).

Case Study—A Fulfillment Center. A fulfillment center serves online customer orders, which usually contain few items. The operations in this center include picking, packing, and shipping all received customer orders. Products or inventories are stored on racks that are organized in a modular manner, as shown in Figure 7.2. Each module has its own dedicated pickers. Pickers are responsible of picking products assigned to them while circumnavigating their modules. With the aid of a hand-held scanner, wireless technology, and the warehouse management system, a picker instantly knows the exact location and the required quantity of each product to be picked. Once products are picked they are place in a plastic tote for transporting to the packing area by a conveyor system, as shown in Figure 7.3. Each tote contains multiple products and is bar coded to identify its contents.

All pickers work as a team and work concurrently to pick orders that are released to them. In this approach it is more economical to group multiple customer orders into batches and consolidate the products picked from different modules in the packing area. In the packing area, all products are removed from the tote, scanned, and automatically dispensed into a packing slot in the sorter. Once all the products belonging to one customer order arrive at the packing slot, the packer will be signaled to put the items in a shipping box, apply shipping labels, and move the box to the loading area for shipping. The time difference between the last arrival item and the first arrival item to the same packing slot is defined as the order consolidation time.

The objective for this system model is to determine the effect of the batch size on the average order consolidation time and the average number of orders packed in 12 hours of operations. The batch is formed by selecting a set of orders from the database that contains all customer orders, as shown in Figure 7.4. Once a batch is formed, each picker in each module will receive a list of items that need to be picked, called a pick list. The pick list contains a sorted list of items based on product locations and the required quantity.

FIGURE 7.3 AN ILLUSTRATION OF THE FULFILLMENT CENTER.



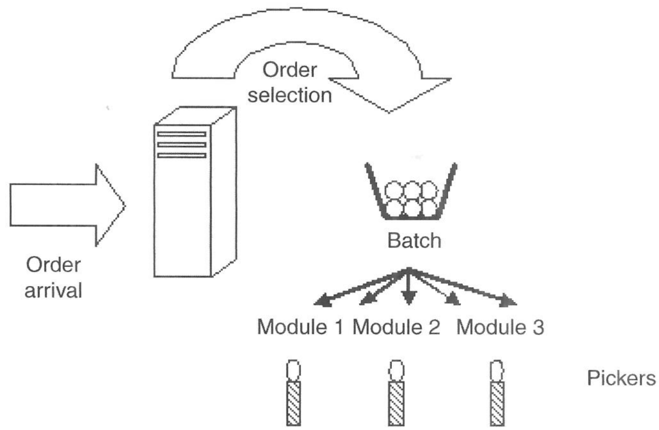
Examples of entities include the following.

- *Customer order lines (COLs)*: A customer order line represents a unique product type ordered by a customer order. The associated attributes include module location, rack location, and quantity ordered.
- *Pickers*: The hand-held scanner shows the picker the location of next item on the pick list and the picker scans the bar code on the racks after the items are picked, which will trigger the display of the next item on the pick list. The picker position is constantly tracked by the last scan location. The attributes associated with each picker include module number, current location, and pick list.
- *Plastic totes*: Each picker starts the picking process with an empty tote. When the tote is full, a bar-coded label will be attached on the lid and scanned to register all the items in this tote. The attribute for each tote is the list of the items placed in the tote.
- *Batcher*: The batcher is an entity that represents the warehouse management system. Its function is to select customer orders to form a batch when the picker finishes the previous pick list.

The following will be modeled as resources:

- *Induction station*: Each tote that arrives at the packing area will be emptied at the induction station and all items will be scanned individually for sorting. Thus, the entities competing for these resources are the totes that arrived to the packing area.
- *Packing slots*: Each slot is used for one unique customer order. Thus the entities competing for these resources are the customer orders.

FIGURE 7.4 ILLUSTRATION OF BATCH FORMING.



This operation is modeled using discrete event simulation because all the processes for each entity is well defined and follows a routine. The processing time for each activity is stochastic in nature, for example, picking time and walking time, and can be captured by a distribution of collected data. Replications are done by applying identical sets of input parameters and model structure with a different set of random numbers, which then can be analyzed. The results of 30 replications for each batch size are illustrated in Figures 7.5 and 7.6. The number of replications will affect the power of the statistical analysis. Thus, the decision on the number of replications is a trade-off between the result quality and the experiment run time. Pooch and Wall (1992) address the issue of sampling size.

The experimental results indicate that the batch size has strong correlations with the average consolidation time and the average number of orders packed. The increase of the average number of orders packed is attributed to the fact that the picker spends more time picking than walking when the batch size is large. The reduction of picker's walking time (similar to setup time in a manufacturing process) increases the productivity of the picker. This proposition suggests that it is always more economical to have a larger batch.

On the other hand, the batch sizes also have a positive relationship with the average order consolidation time, as shown in Figure 7.6, which is not desirable when there is a limited number of packing slots. When more customer orders are batched together, the time to complete each pick list will be longer. This introduces higher variations in COL arrival times to the packing slots. A high average consolidation time implies that the customer order spends more time waiting in the packing slot. If all the slots are occupied and none of the orders are ready to be packed, the incoming tote will be blocked and this may escalate to a deadlock in the system, which is not desirable.

FIGURE 7.5 THE AVERAGE NUMBER OF PACKED ORDERS FOR FIVE DIFFERENT BATCH SIZES. EACH DATA POINT REPRESENTS THE AVERAGE OF 30 REPLICATIONS. THERE ARE SIGNIFICANT DIFFERENCES AT THE 95% CONFIDENCE LEVEL BETWEEN BATCH SIZES.

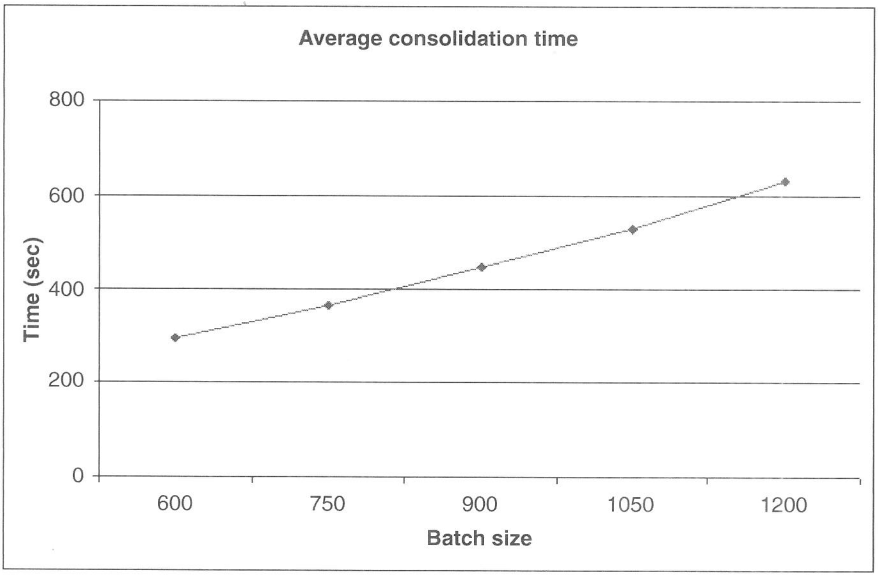
3.3 Agent-Based Simulation

Agent-based simulation (ABS) is a computationally intensive approach to simulate autonomous elements in a complex environment and how they act, react, and interact. The idea was proposed by mathematician John von Neuman in the 1940s and improved upon by other mathematicians, including Stanislaw Ulam and Craig Reynolds. Adequate computing facilities were not widely available for implementation of ABS until the 1990s. It was then that Sugarscape (Epstein and Axtell, 1996), the first large-scale ABS, was developed. Sugarscape modeled interaction under simple to complex behavior rules, demonstrating the applicability of ABS to biological and social systems operations.

ABS is considered a bottom-up approach because each element, called an 'agent', executes programmed behavior individually. Agent behavior occurs at the micro-level, but through actions and interactions with other agents, macroscopic system behaviors are exhibited (Wilensky, 2008). This emergent behavior represents a phenomenon that is a main advantage of the ABS approach, because it is often counterintuitive (Bonabeau, 2002).

ABS is a methodological tool that has the capability to integrate and synthesize contentious theoretical frameworks for a comprehensive understanding of political, economic, and social systems and processes. In recent years research has been using ABS methodology in such applications as the development of synthetic economies (Chaturvedi and Mehta, 1999) and societies (Bhavani and Backer, 2000), artificial labor market (Chaturvedi et al., 2005), market and supply chain co-design (Chaturvedi et al., 2007), web search (Boudriga, 2004),

FIGURE 7.6 THE AVERAGE CONSOLIDATION TIME IN SECONDS FOR FIVE DIFFERENT BATCH SIZES. EACH DATA POINT REPRESENTS THE AVERAGE OF 30 REPLICATIONS. THERE ARE SIGNIFICANT DIFFERENCES AT THE 95% CONFIDENCE LEVEL BETWEEN BATCH SIZES.

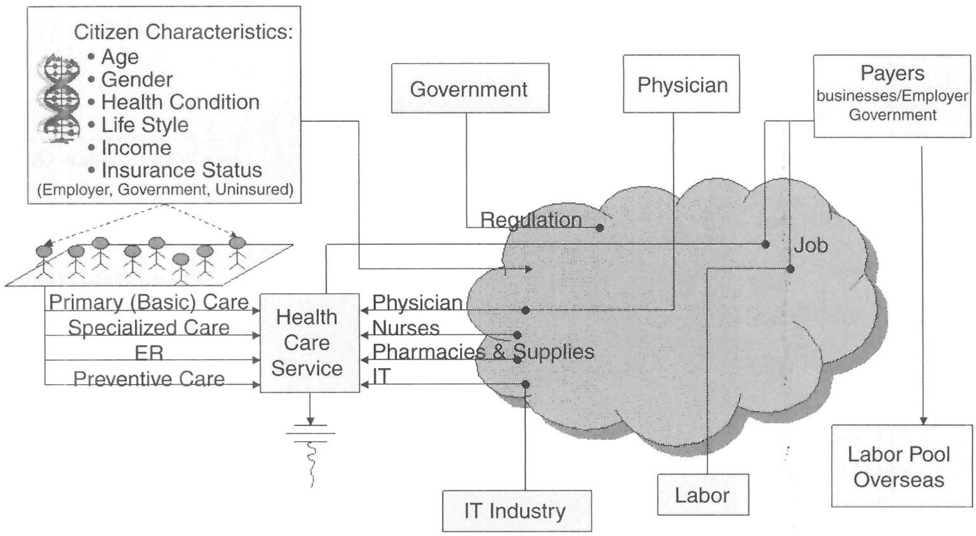


investigating conditions for alliance among nations (Axelrod, 1997), identity development and diffusion (Lustick, 2000); secessionism in multiethnic states (Lustick et al., 2004); and emergence of ethnocentrism (Axelrod and Hammond, 2003).

Case Study—A Synthetic Environment for Analysis and Simulation for Health-Care Reform. The Synthetic Environment for Analysis and Simulation for Healthcare Reform (SEAS-HCR) is an agent-based simulation that mimics the United States' current health-care system. It models citizens' life styles, significant diseases, medical services, and their associated costs into a synthetic health-care system, as shown in Figure 7.7. It articulates the micro-level behaviors of individual entities of the real-life environment from which the macro-level behaviors of the synthetic health-care system emerges. Agents, representing the population, including the patients, have the following characteristics: age, gender, health condition, life style, income, and insurance status. These characteristics allow for a more accurate identification of the medical services these citizens will require. The type of services needed by different citizen classes falls into the following four categories: primary (basic) care, specialized care, preventative care, and emergency care. SEAS-HCR integrates models of various aspects of entities in the health-care system together in a computational experimentation environment.

SEAS-HCR is parameterized with data from multiple sources, such as the Centers for Disease Control and Prevention and the Kaiser Family Foundation. Multiple runs are made

FIGURE 7.7 CONCEPTUAL MODEL OF THE SYNTHETIC ENVIRONMENT FOR ANALYSIS AND SIMULATION FOR HEALTHCARE REFORM (SEAS-HCR).

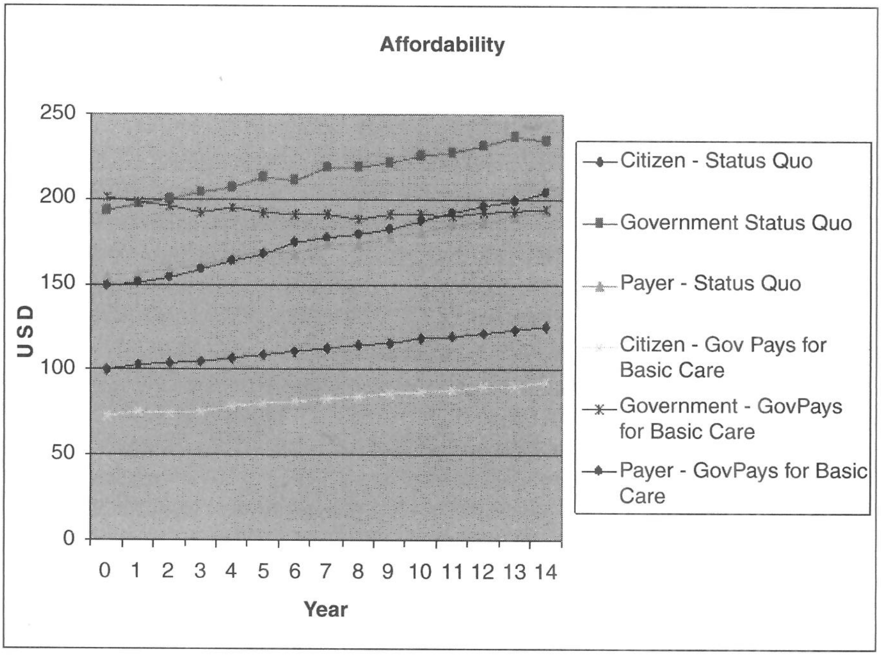


and model parameters are tweaked such that SEAS-HR outputs match the real-world data. The calibrated system can then be used for experimentation for policy analysis.

SEAS-HCR allows for *n*-sided analysis. Different sides modeled in SEAS-HCR include (1) government as a regulator and a payer; (2) businesses, employers, and individuals who are responsible for paying in the health-care system; (3) the physicians, nurses, and technology that make possible the services provided; and (4) the labor force of the United States, both domestic and foreign, which makes programs like Medicare and Medicaid realizable. Each side inputs policy choices based on its objectives, impacting the outcome of the simulation. For example, the Government would regulate to lower cost of Medicaid and Medicare, industry would like to transfer their health-care cost to the consumer or the Government, and the consumer would like to have health-care coverage.

Computational Experimentation. SEAS-HCR shows the relationship between the policy inputs, the citizens' lifestyles and behaviors, and the government regulations. By making changes to one group, the health-care model projects the effects that these changes have on other groups. The goal of the health-care system is the continuous increase of accessibility and affordability to the wider population. The simulated health-care system, just as in the real world, is an interlinked system of systems where seemingly small changes in one aspect of a system may alter the dynamics of the entire system of systems. We use the computational experimentation to analyze such changes within the health-care system as discussed in the two sample experiments below.

FIGURE 7.8 AFFORDABILITY GOES UP FOR THE CITIZENS WHEN THE GOVERNMENT PAYS FOR BASIC HEALTH CARE.

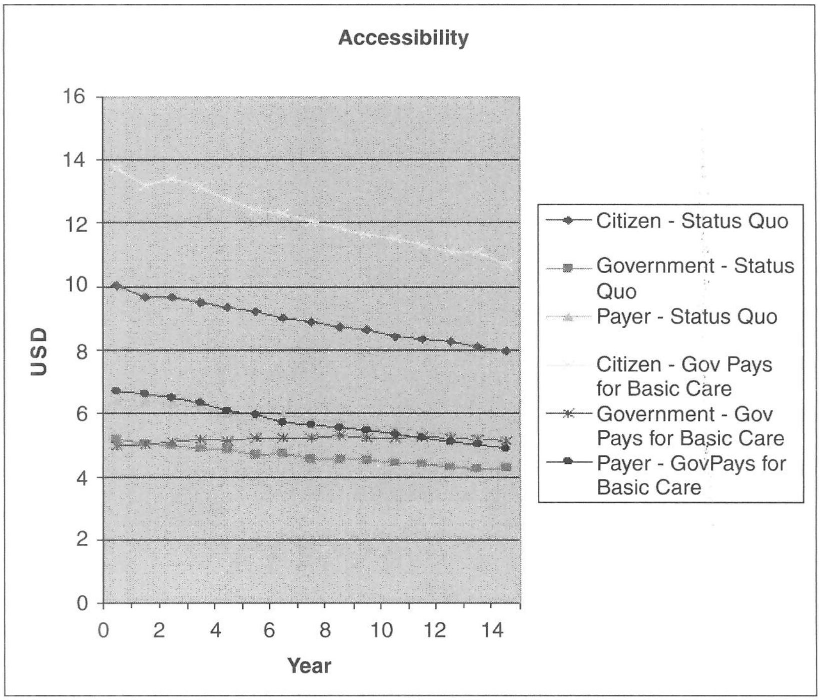


Experiment 1: The first simulation investigates the policies to increase accessibility and affordability of the health-care services. To achieve this, the model selected basic services, including regular medical examinations, vaccinations, and basic care provided at the emergency department and assigned their cost to the government. The results of assigning government funding for basic care indicate that health care would become both more affordable and more accessible to the public, as shown in Figure 7.8.

Figure 7.9 shows the change to the model that enabled a citizen's health-care dollar to go further as a result of additional government aid for basic care. This increase in the dollar value for citizens makes the health-care system more accessible. Consequently, the model validates that the first solution of government subsidization for basic care reaches the goal of increased affordability and accessibility.

Experiment 2: Another goal of the simulation was to increase consumer choice and responsibility to make better-educated health-care decisions. For example, the overweight population was increased to mimic the current upward trend of the health issue in the United States. As expected, this increase raises the total health-care cost as well as the incidence and costs of specific diseases, such as diabetes and cardiovascular-related

FIGURE 7.9 ACCESSIBILITY IN TERMS OF NUMBER OF SERVICES RENDERED PER ADDITIONAL DOLLAR SPENT GOES UP.

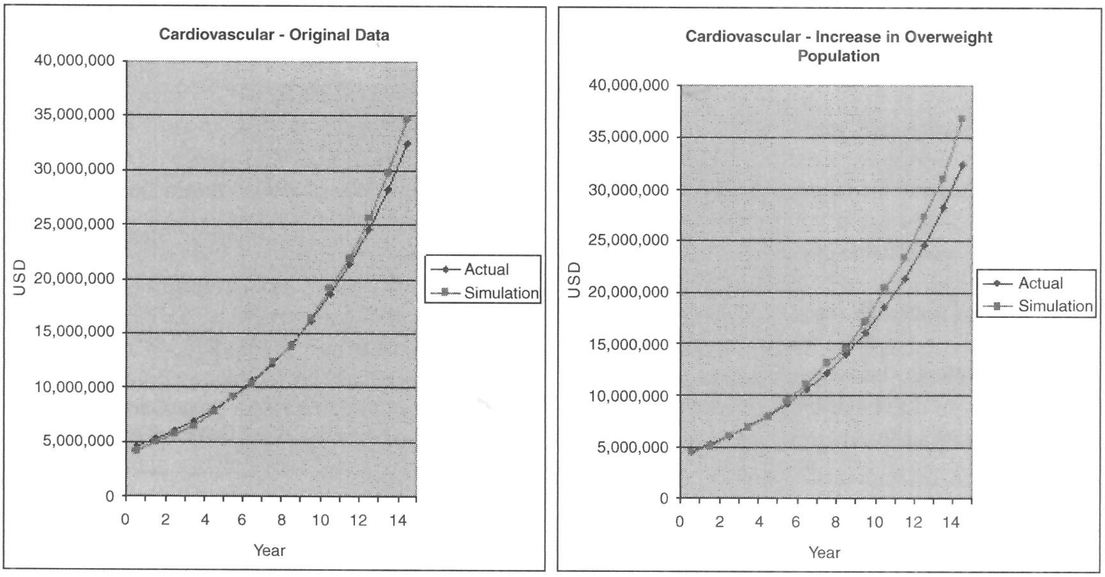


diseases seen in Figure 7.10. The model shows that personal responsibility aspects of health care play a key role in reducing health-care costs.

With the baby boomer population aging, it is important to look ahead and plan for the future. To analyze the policies that need to be in place to prepare for this change in demographics, the proportions of children, adults, and elderly Americans were adjusted in the computational experimentation environment according to predictions of the Census Bureau, estimating the percentage of Americans aged 65 and over to be over 16 percent of the population. As shown in Figure 7.11, this modification affects the disease incidence and, especially, the total cost of health care. The effects of the growth of the aging population on the health-care system evidence a need for personalized medical care, as the reduced number of working aged citizens will eventually be unable to cover the increasing medical costs.

This case study demonstrates how an agent-based simulation can be used for policy analysis of various systemic as well as peripheral changes to the health-care industry within the United States. It captures the essence of the dynamic nature of the population demographics, while mimicking the collaborative and even conflicting behaviors of various participants of the health-care delivery system. *N*-sided "what-if" analyses may be conducted to observe short-

FIGURE 7.10 THE SIMULATED HEALTH-CARE COSTS OF CARDIOVASCULAR DISEASES RISES WHEN MODELING THE PROJECTED RISE IN THE OVERWEIGHT POPULATION IN THE UNITED STATES.



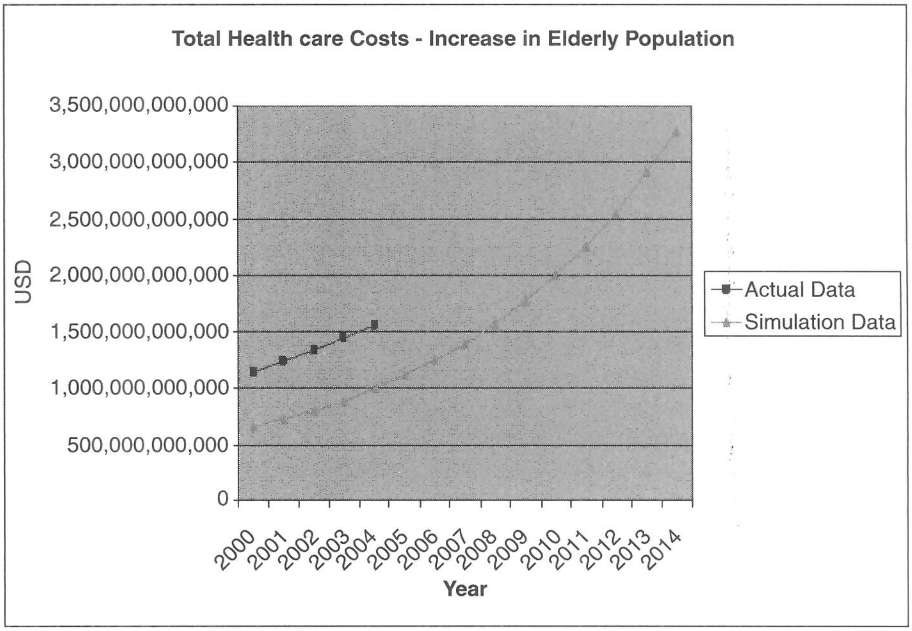
term and long-term trends in the health-care system. SEAS-HCR may be used for policy research and analysis for reform of the health-care industry and to simulate and test strategies that may lead to a better quality of life.

Hybrid System

A hybrid modeling approach may be beneficial when the system has multiple facets where each facet naturally fits in a different modeling method. Hybrid systems can be developed to analyze complex scenarios involving both social and physical systems. The interface between the subsystem models is challenging because the models may not have consistent time and space scales.

Case Study—A Fire and Agent Hybrid Simulation. Fire and Agent Hybrid Simulation is a Dynamic Data Driven Application System (DDDAS) created to study interaction between fire and agent models during an evacuation from a two different scenarios (Chaturvedi, 2007). NIST fire code FDS was used for the simulation of fire. The agent software was designed to simulate agent behaviors during evacuation by tracking the behavior of each individual in the building, taking into account the effects of temperature, carbon monoxide, carbon dioxide, and smoke on the behavior and health of each agent. The created shared environment was designed to provide the bridge between multiple simulations for data transfer and model

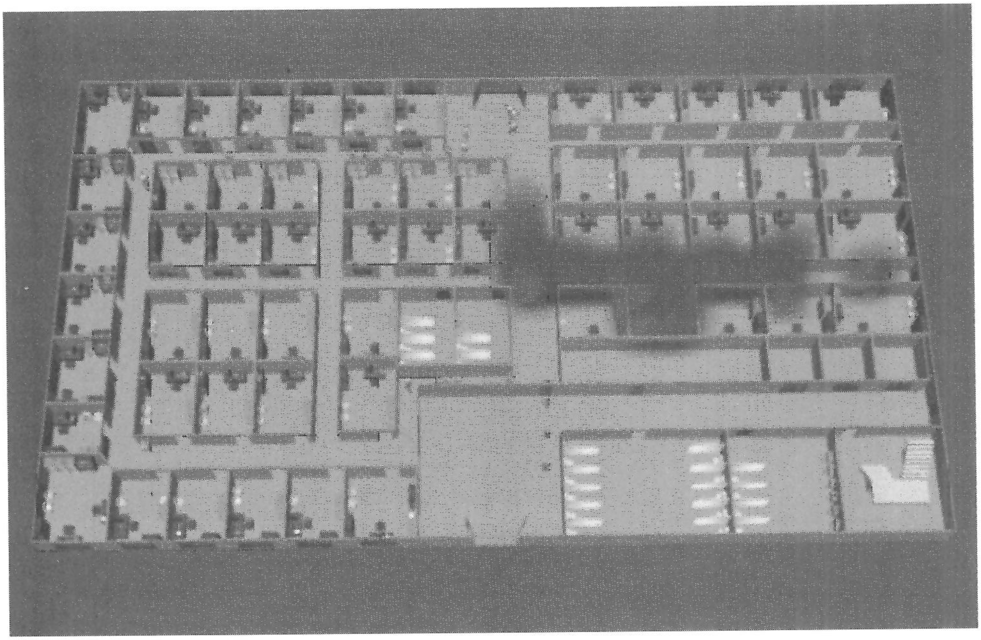
FIGURE 7.11 THE EXPONENTIAL RISE IN HEALTH-CARE COSTS PROJECTED DUE TO THE RISE IN THE ELDERLY POPULATION.



interaction. It was shown that the number of available exits significantly affects agent health and death toll. The results can be used for better fire safety building design and regulations and for training first responders. The goal of the project is to create a state-of-the-art integrated DDDAS to study the interaction between fire and agents in a crisis situation. Realistic and complex models of fire and agents were used to simulate accurate behavior. These simulations were created to run independently using individual inputs and outputs. The shared reality engine was used to exchange information from one simulation to another. Many fire and evacuation simulations exist to date. To provide in-depth analysis the system should allow for both changing the scenario and conditions and studying the effects of different model applications. In other words, DDDAS should allow for a relatively simple way of replacing a fire and/or evacuation model as well as changes of geometry, initial conditions, etc.

Human-in-the-Loop Experimentation. Measured Response was an exercise held annually from 2002 to 2005 at Purdue University. During this exercise, we employed the Synthetic Environment for Analysis and Simulation (SEAS) to *simulate* large-scale terrorist attacks on our nation's critical infrastructure, to *test* our preparedness to respond to these high-consequence events, and to *educate* and *prepare* participants through a series of coordinated exercises and workshops. Measured Response experiments were designed to help develop and test disaster preparedness and response strategies and to conduct research that would refine our decision-

FIGURE 7.12 FIRE AND SMOKE SPREADING THROUGH A VIRTUAL BUILDING FLOOR.



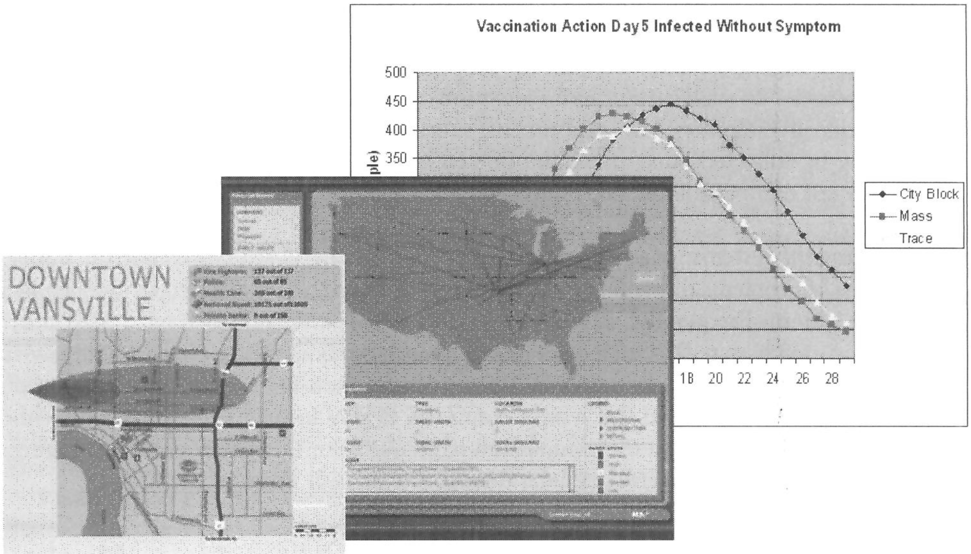
making models and better prepare our nation for catastrophic attacks. Invited participants from local, state, and federal government agencies, industry, nongovernmental organizations, and the media coordinated a comprehensive plan to respond to crises, minimize impact of such events and assist our nation in recovery. In addition to an accurate virtual population, SEAS creates synthetic replicas of societies and economies.

We have also modeled various chemical (phosgene gas released in the air, dioxin contaminating the food supply chain), biological (epidemiology of Ebola, SARS, smallpox, plague, staphylococcus, anthrax, and pandemic influenza), and radiological (cesium-137) crisis events in our simulation, providing the ability for participants to respond to a multievent crisis. Enhanced data visualizations present the participants with high-level overviews and detailed accounts of the data generated in the virtual environments. With input from human participants responding to simulated threats and incidents, Measured Response modeled realistic, but synthetic, critical incidents on a national and global scale (Drnevich et. al., 2009).

Discussion

Each modeling approach offers different advantages and capabilities. One key factor in selecting an appropriate modeling approach is the purpose of the model. The mathematical

FIGURE 7.13 RADIOLOGICAL PLUME IN A VIRTUAL CITY BLOCK; NATIONWIDE FOOD SUPPLY CHAIN NETWORK; IMPACT ON PUBLIC HEALTH DUE TO DIFFERENT VACCINATION STRATEGIES.



modeling approach uses mathematical formulas or constraints to describe system behaviors. It is mostly used for determining the best policy to optimize a defined system performance.

The discrete event simulation approach focuses on the modeling of events (such as processes, activities, movements). It is often adopted to replace real-world experiments to save time and cost. This approach can be used to study the system dynamics, the impact of each factor, what-if scenarios, alternative designs, and alternative control policies.

Agent-based simulation presents a different view of the system than discrete event simulation. One of the most distinct features of the agent-based simulation approach is its ability to model inverse classes of problems to study the emergent behavior in the system. In those cases, we mostly understand the individual agent's behavior pattern and its interactions with other agents. The purpose of the model is identifying emergent behavior when those agents are "living" in a system.

In service enterprises, each system may have different characteristics. Some are primarily physical systems, some are social systems, and some are hybrid. Most service systems consist of humans and the system is greatly affected by human behaviors, such as decision making, movement, emotional state, and processing and exchanging information. Consequently, service system modeling, in general, is more challenging, both in the development phase and in the verification/validation phase, than manufacturing system modeling. Some of those challenges are as follows.

Good Models Need Knowledge and Quality Data. Quality of knowledge and data are important in developing models that closely represent systems. Knowledge regarding processes, procedures, or policy protocols needs to be verified and defined. These definitions may come from validated theories or keen observations. In addition, data that dictate known system behaviors are required. Missing information or errors in these requirements may ultimately undermine the model's validity. In a complex service system, some of the human interactions and changing behaviors may not be fully understood and therefore not completely captured in the model. In such cases, careful attention must be given in defining the scope of the behaviors under investigation, and experiments must be appropriately designed. In addition, the data may be collected once, but cannot be collected again because the environmental conditions are not controllable or not repeatable.

Transient Systems. Many complex service systems are ill-structured and fall under the inverse class of problems displaying emergent behavior and/or are affected by the people modeled in the system. Frequently, such systems demonstrate emergent behaviors that may tend to be "non-steady state." In such cases, the models that rely on "long-term" or "steady-state" results may be less reliable. In addition, those systems often behave differently based on the initial starting condition and the system continues to stay in the "transient" state without convergence. This will make the comparison of results from the model very challenging. As such, a carefully crafted uncertainty quantification approach must be applied.

Hidden Assumptions. Assumptions are often applied when

- The data for a certain component in the system is not available
- The knowledge of certain portion of system structure is unknown
- The assumption has been verified by prior work in a similar system
- It is believed that the assumption closely represents the system behavior

Sensitivity analysis is commonly used to test some of the assumptions. The ones that are difficult to detect are those embedded in the system model due to the bias in the data or the unintended modeling deviation from the system structure.

Summary

The service industry has become an important sector in the United States, accounting for 55 percent of economic activities based on the survey by the U.S. Bureau of the Census. System modeling tools, however, are mostly inherited from the manufacturing sector. The structure and activities in service system sometimes are not defined as clearly as those in manufacturing and this creates many challenges in system modeling and analysis. In addition, the lack of proper verification and validation methods for service system models impairs the full utilization of the developed model. There is great need for new developments and methods in those areas to make service system modeling more effective. Advances in agent-based simulation and hybrid approaches are quite promising and have demonstrated value in tackling large-scale complex service enterprise problems.

References and Further Reading

Agha, G. A., Cindio, F. D., and Rozenberg, G. Concurrent object-oriented programming and Petri nets, advances in Petri nets. *Lecture Notes in Computer Science*. Springer, Berlin, 2001.

Ang, C. H. *A Simulation Approach to Minimize Water Supply, Water Storage Capacity and Water Treatment Capacity Requirements in an Advanced Life Support System for Mars Missions*. Thesis, Purdue University, West Lafayette, IN, 2006.

Ang, C. H., and Yih, Y. A simulation approach to minimize water supply, water storage capacity and water treatment capacity requirements in an advanced life support system for Mars missions. *Habitation—An International Journal for Human Support Research*, *11*, 173–183, 2008.

Axelrod, R. Advancing the art of simulation in the social sciences. In: R. Conte, R. Hegelsmann, and P. Terna,eds. *Simulating Social Phenomena*. Springer-Verlag, Berlin, 1997, pp. 21–40.

Axelrod, R., and Hammond, R. A. The evolution of ethnocentric behavior. Paper presented at Midwest Political Science Convention, Chicago, IL, April 2003.

Axtell, R. L. Why Agents? On the varied motivations for agent computing in the social sciences. Center on Social and Economic Dynamics. Working paper No. 17, November 2000. www.amitkoth.com/weblog/dissertation/documents/axtell.pdf, accessed Nov. 25, 2008.

Banks, J., Carson, J. S. II, Nelson, B. L., and Nicol, D. M. *Discrete-event System Simulation*, 3rd ed. Prentice-Hall, Upper Saddle River, NJ, 2000.

Bellman, R. E. A Markovian decision process. *Journal of Mathematics and Mechanics* *6*, 679–684, 1957.

Bhavani, R., and Backer, D. Localized ethnic conflict and genocide: Accounting for differences in Rawanda and Burundi. *Journal of Conflict Resolution* *44*, 283–306, 2000.

Bonabeau, E. Agent-based modeling: Methods and techniques for simulating human systems. Proceedings of the National Academy of Sciences of the United States of America *99*(3), May 2002, pp. 7280–7287. www.pnas.org/content/99/suppl.3/7280.full.pdf+html, accessed Nov. 25, 2008.

Boudriga, N., and Obaidat, M. Intelligent agents on the web: A review. *Computing in Science & Engineering* *6*(4), 35–42, 2004.

Cao, T. H., and Sanderson, A. C. *Intelligent Task Planning Using Fuzzy Petri Nets* (Intelligent Control and Intelligent Automation Series vol. 3). World Scientific, River Edge, NJ, 1996.

Centers for Disease Control and Prevention (CDC). www.cdc.gov/

Chaturvedi, A. R., and Mehta, S. Simulations in economics and management: Using the SEAS simulation environment. *Communications of the ACM* *42*(3), 60–61, 1999.

Chaturvedi, A. R., Mehta, S., Dolk, D., and Ayer, R. Artificial labor market. *European Journal of Operational Research* *166*, 694–716, 2005.

Chaturvedi, A., Mellema, A., Hsieh, C., Bhatt, T., Cheng, M., Dietz, E., and Stearns, S. Decision support for security: A mixed reality approach to bio-terror preparedness and response. In: F. Burstein, and C. Holsapple,eds. *Handbook on Decision Support Systems*. Springer-Verlag, Berlin, 2007, pp. 463–481.

Chiam, T. C., Yih, Y., and Mitchell, C. A. Control policies for a water-treatment system for human habitation of space using the Markov Decision Process. Paper presented at 37th COSPAR Scientific Assembly, Montreal, Canada, 2008.

Chiam, T. C., Yih, Y., and Mitchell, C. A. Control policies for a water-treatment system using the Markov Decision Process. Part 1: Mathematical formulation. *Habitation—An International Journal for Human Support Research*, in press, 2009a.

Chiam, T. C., Yih, Y., and Mitchell, C. A. Control policies for a water-treatment system using the Markov Decision Process. Part 2: Simulation and analysis. *Habitation—An International Journal for Human Support Research*, in press, 2009b.

Chinneck, J. W. Practical optimization: A gentle introduction. 2001. www.sce.carleton.ca/faculty/chinneck/po.html, accessed Nov. 25, 2008.

Desel, J., and Esparza. J. *Free Choice Petri Nets*. Cambridge University Press, Cambridge, UK, 1995.

Drnevich, P., Ramanujam, R. Mehta, S., and Chaturvedi, A., Affiliation or situation: What drives strategic decision making in crisis response? *Journal of Managerial Issues*, June 2009.

England, W., and Roberts, S. Applications of computer simulation in health care. Proceedings of the 10th Conference on Winter Simulation. Vol. 2, Miami Beach, FL, Dec. 4-6, 1978, pp. 665–676.

Epstein, J., and Axtell, R. Growing Artificial Societies: *Social Science from the Bottom Up*. Brookings Institution Press, Washington DC, 1996.

Feinberg, E. A., and Shwartz, A. *Handbook of Markov Decision Process: Methods and Applications*. Kluwer, Boston, MA, 2002.

Hanford, A. J.Advanced life support baseline values and assumptions document, JSC 47804 (CR-2004-208941), NASA Lyndon B. Johnson Space Center, Houston, TX, August 2004.

Howard, R. A. *Dynamic Programming and Markov Processes*. M.I.T Press, Cambridge, MA, 1960.

Kaiser Family Foundation (KFF). www.kff.org/

Law, A. M., and Kelton, W. D. *Simulation Modeling and Analysis*, 3rd ed. McGraw-Hill, New York, 2000.

Lustick, I., Agent-based modeling of collective identity: Testing constructivist theory. *Journal of Artificial Societies and Social Simulation* *3*(1), 2000. jasss.soc.surrey.ac.uk/3/1/1.html, accessed Nov. 25, 2008.

Lustick, I., Miodownik, D., and Eidelson, R. Secessionism in multicultural states: Does sharing power prevent or encourage it? *American Political Science Review* *98*(2), 209–229, 2004.

Marc, S. M. Optimization in simulation: A survey of recent results. Proceedings of the 19th Conference on Winter Simulation. Atlanta, GA, 58–67, 1987.

Mcleish, K. Petri nets. www.cse.fau.edu/~maria/COURSES/CEN4010-SE/C10/10-7.html

Murata, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, *77*(4), 541–580, 1989.

Nikolic, D. M, Panic, B., and Vujosevic, M. Bullwhip effect and supply chain modeling and analysis using CPN tools. In: K. Jensen, ed. Proceedings of the 5th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, October 2004. Department of Computer Science, University of Aarhus, Aarhus, DK, PB-570, 219–234, 2004.

Ottman, W. S., Ford, A. C., and Reinhardt, G. R. An aircraft taxi simulation model for the United Parcel Service Louisville air pack, Proceedings of the 1999 Winter Simulation Conference, Phoenix, AZ, 1999, pp. 1221–1225.

Pooch, U. W. and Wall, J. *Discrete Event Simulation: A Practical Approach*. CRC Press, Boca Raton, FL1994.

Reynolds, C. Homepage: http://www.red3d.com/cwr, accessed Nov. 25, 2008.

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

Schriber, T. J. *An Introduction to Simulation Using GPSS/H*. Wiley, New York, 1991.

Schriber, T. J., and Brunner, D. B. Inside discrete event simulation software: How it works and why it matters. Proceedings of the 39th Winter Simulation Conference, Washington, DC, 2005, pp. 113–123.

Schach, S. R. *Classical and Object Oriented Software Engineering (with UML and C++), 4th edition*. McGraw Hill, New York, NY, 1998.

U.S. Census Bureau. www.census.gov/

Wilensky, U. *NetLogo. ccl.northwestern.edu/netlogo/*. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 2008.