CHAPTER

9

Model-Based Decision Making: Optimization and Multi-Criteria Systems

LEARNING OBJECTIVES

- Understand the basic concepts of analytical decision modeling
- Describe how prescriptive models interact with data and the user
- Understand some different, well-known model classes
- Understand how to structure decision making with a few alternatives
- Describe how spreadsheets can be used for analytical modeling and solution
- Explain the basic concepts of optimization and when to use them
- Describe how to structure a linear programming model
- Describe how to handle multiple goals
- Explain what is meant by sensitivity analysis, what-if analysis, and goal seeking
- Describe the key issues of multi-criteria decision making

In this chapter we describe selected techniques employed in prescriptive analytics. We present this material with a note of caution: Modeling can be a very difficult topic and is as much an art as a science. The purpose of this chapter is not necessarily for you to *master the topics* of modeling and analysis. Rather, the material is geared toward *gaining familiarity* with the important concepts as they relate to DSS and their use in decision making. It is important to recognize that the modeling we discuss here is only cursorily related to the concepts of data modeling. You should not confuse the two. We walk through some basic concepts and definitions of modeling before introducing the influence diagrams, which can aid a decision maker in sketching a model of a situation and even solving it. We next introduce the idea of modeling directly in spreadsheets. We then discuss the structure and application of some successful time-proven models and methodologies: optimization, decision analysis, decision trees, and analytic hierarchy process. This chapter includes the following sections:

9.1 OPENING VIGNETTE: Midwest ISO Saves Billions by Better Planning of Power Plant Operations and Capacity Planning

INTRODUCTION

Midwest ISO (MISO) operates in 13 U.S. states as well as the province of Manitoba in Canada. It manages 35 transmission owners and 100 non-transmission owners, ensuring that all members of the organization have equal access to high-voltage power lines. Together, the United States and the province of Manitoba constitute one of the largest energy markets in the world, with yearly energy transactions amounting to about $23 billion. Before Midwest ISO existed, each transmission company operated independently. Now, after a company joins MISO, it still maintains control of its power plants and transmission lines, and shares in the responsibility of supplying and buying energy in a wholesale electricity market to meet demand. MISO, however, has the responsibility of deciding when and how much energy to produce and administer to the market in such a way as to increase benefit to society.

PRESENTATION OF PROBLEM

Individually, the companies had to make extra investments to manage risk. Their mode of operation resulted in inefficient use of transmission lines. Deregulation policies were introduced by Congress and were implemented by the Federal Energy Regulatory Commission (FERC) for the wholesale electricity industry. When MISO was formed, it first started an energy-only market in 2005 that ensured unbiased access to transmission lines. In 2009, it added ancillary services (regulation and contingency reserves) to its operations. Regulation was supposed to ensure that the frequency did not deviate from 60 hertz. Contingency reserves were supposed to help ensure that in the event of unexpected power loss, demand was met within 10 minutes of the power loss. Operations research methods were considered as means to provide the level of performance demanded by the ancillary services.

METHODOLOGY/SOLUTION

Sequentially, two optimization algorithms were used. These were the commitment algorithm and the dispatch algorithm. The commitment algorithm committed power plants to be either on or off. The dispatch algorithm determined the level of a power plant's output and price. With these two algorithms, facilities were given constraints on how much electricity to carry within their physical limits in order to avoid overload and damage to expensive equipment. The commitment problem for the energy-only market made use of the Lagrangian relaxation method. As mentioned earlier, it determined when each plant should turn on or off. The dispatch problem was solved with a linear programming model. It helped decide how much output should be produced by each power plant. It also helped determine the price of energy based on the location of the power plant. Even though these methods were just fine, they were not appropriate for the ancillary service market commitment problem. Rather, a mixed integer programming model was used as a result of its superior modeling capacity.

RESULTS/BENEFITS

Based on the improvements made, reliability of the transmission grid improved. Also, a dynamic transparent pricing structure was created. Value proposition studies show that Midwest ISO achieved about $2.1 billion and $3 billion dollars in net cumulative savings between 2007 and 2010. Future savings are expected to accrue to about $6.1 billion.

QUESTIONS FOR THE OPENING VIGNETTE

1. In what ways were the individual companies in Midwest ISO better off being part of MISO as opposed to operating independently?
2. The dispatch problem was solved with a linear programming method. Explain the need of such method in light of the problem discussed in the case.
3. What were the two main optimization algorithms used? Briefly explain the use of each algorithm.

LESSONS WE CAN LEARN FROM THIS VIGNETTE

Operations research (OR) methods were used by Midwest ISO to provide efficient and cheaper sources of energy for states in the midwestern region of the United States. A combination of linear programming and the Lagrangian relaxation methods was used to determine an optimized approach to generate and supply power. By extension, this methodology could be used by both government agencies and the private sector to optimize the cost and provision of services such as healthcare and education.

Source: Brian Carlson, Yonghong Chen, Mingguo Hong, Roy Jones, Kevin Larson, Xingwang Ma, Peter Nieuwesteeg, et al., "MISO Unlocks Billions in Savings Through the Application of Operations Research for Energy and Ancillary Services Markets," *Interfaces,* Vol. 42, No. 1, 2012, pp. 58–73.

9.2 DECISION SUPPORT SYSTEMS MODELING

Many readily accessible applications describe how the models incorporated in DSS contribute to organizational success. These include Pillowtex (see ProModel, 2013), Fiat (see ProModel, 2006), Procter & Gamble (see Camm et al., 1997), and others. INFORMS publications such as *Interfaces, ORMS Today,* and *Analytics* magazine all include stories that illustrate successful applications of decision models in real settings. This chapter includes many examples of such applications, as does the next chapter.

Simulation models can enhance an organization's decision-making process and enable it to see the impact of its future choices. Fiat (see ProModel, 2006) saves $1 million annually in manufacturing costs through simulation. IBM has predicted the behavior of the 230-mile-long Guadalupe River and its many tributaries. The prediction can be made several days before the imminent flood of the river. This is important as it would allow for enough time for disaster management and preparation. IBM used a combination of weather and sensor data to build a river system simulation application that could simulate thousands of river branches at a time. Besides flood prediction, the application could also be used for irrigation planning in such a way as to avoid the impact of droughts and surplus water. Even companies under financial stress need to invest in such solutions to squeeze more efficiency out of their limited resources—maybe even more so. Pillowtex, a $2 billion company that manufactures pillows, mattress pads, and comforters, had filed for bankruptcy and needed to reorganize its plants to maximize net profits from the company's operations. It employed a simulation model to develop a new lean manufacturing environment that would reduce the costs and increase throughput. The company estimated that the use of this model resulted in over $12 million savings immediately. (See **promodel.com**.) We will study simulation in the next chapter.

Modeling is a key element in most DSS and a necessity in a model-based DSS. There are many classes of models, and there are often many specialized techniques for solving each one. Simulation is a common modeling approach, but there are several others.

Applying models to real-world situations can save millions of dollars or generate millions of dollars in revenue. Christiansen et al. (2009) describe the applications of such models in shipping company operations. They describe applications of TurboRouter, a DSS for ship routing and scheduling. They claim that over the course of just a 3-week period, a company used this model to better utilize its fleet, generating additional profit of $1–2 million in just a short time. We provide another example of a model application in Application Case 9.1.

Application Case 9.1

Optimal Transport for ExxonMobil Downstream Through a DSS

ExxonMobil, a petroleum and natural gas company, operates in several countries worldwide. It provides several ranges of petroleum products including clean fuels, lubricants, and high-value products and feedstock to several customers. This is completed through a complex supply chain between its refineries and customers. One of the main products ExxonMobil transports is vacuum gas oil (VGO). ExxonMobil transports several shiploads of vacuum gas oil from Europe to the United States. In a year, it is estimated that ExxonMobil transports about 60–70 ships of VGO across the Atlantic Ocean. Hitherto, both ExxonMobil-managed vessels and third-party vessels were scheduled to transport VGO across the Atlantic through a cumbersome manual process. The whole process required the collaboration of several individuals across the supply chain organization. Several customized spreadsheets with special constraints, requirements, and economic trade-offs were used to determine the transportation schedule of the vessels. Some of the constraints included:

1. Constantly varying production and demand projections
2. Maximum and minimum inventory constraints
3. A pool of heterogeneous vessels (e.g., ships with varying speed, cargo size)
4. Vessels that load and discharge at multiple ports
5. Both ExxonMobil-managed and third-party supplies and ports
6. Complex transportation cost that includes variable overage and demurrage costs
7. Vessel size and draft limits for different ports

The manual process could not determine the actual routes of vessels, the timing of each vessel, and the quantity of VGO loaded and discharged. Additionally, consideration of the production and consumption data at several locations rendered the manual process burdensome and inefficient.

Methodology/Solution

A decision support tool that supported schedulers in planning an optimal schedule for ships to load, transport, and discharge VGO to and from multiple locations was developed. The problem was formulated as a mixed-integer linear programming problem. The solution had to satisfy requirements for routing, transportation, scheduling, and inventory management vis-à-vis varying production and demand profiles. A mathematical programming language, GAMS, was used for the problem formulation and Microsoft Excel was used as the

(*Continued*)

Application Case 9.1 (Continued)

user interface. When the solver (ILOG CPLEX) is run, an optimal solution is reached at a point when the objective value of the incumbent solution stops improving. This stopping criterion is determined by the user during each program run.

Results/Benefits

It was expected that using the optimization model will lead to reduced shipping cost and less demurrage expenses. These would be achieved because the tool would be able to support higher utilization of ships and help make ship selection (e.g., Panamax versus Aframax) and design more optimal routing schedules. The researchers expected to extend the research by exploring other alternate mathematical methods to solve the scheduling problem. They also intended to give the DSS tool the capability to consider multiple products for a pool of vessels.

Discussion Questions

1. List three ways in which manual scheduling of ships could result in more operational cost as compared to the tool developed.
2. In what other ways can ExxonMobil leverage the decision support tool developed to expand and optimize their other business operations?
3. What are some strategic decisions that could be made by decision makers using the tool developed.

Source: K. C. Furman, J. H. Song, G. R. Kocis, M. K. McDonald, and P. H. Warrick, "Feedstock Routing in the ExxonMobil Downstream Sector," *Interfaces*, Vol. 41, No. 2, 2011, pp. 149–163.

Current Modeling Issues

We next discuss some major modeling issues, such as problem identification and environmental analysis, variable identification, forecasting, the use of multiple models, model categories (or appropriate selection), model management, and knowledge-based modeling.

IDENTIFICATION OF THE PROBLEM AND ENVIRONMENTAL ANALYSIS One very important aspect of it is **environmental scanning and analysis,** which is the monitoring, scanning, and interpretation of collected information. No decision is made in a vacuum. It is important to analyze the scope of the domain and the forces and dynamics of the environment. A decision maker needs to identify the organizational culture and the corporate decision-making processes (e.g., who makes decisions, degree of centralization). It is entirely possible that environmental factors have created the current problem. BI/business analytics (BA) tools can help identify problems by scanning for them. The problem must be understood and everyone involved should share the same frame of understanding, because the problem will ultimately be represented by the model in one form or another. Otherwise, the model will not help the decision maker.

VARIABLE IDENTIFICATION Identification of a model's variables (e.g., decision, result, uncontrollable) is critical, as are the relationships among the variables. Influence diagrams, which are graphical models of mathematical models, can facilitate the identification process. A more general form of an influence diagram, a cognitive map, can help a decision maker develop a better understanding of a problem, especially of variables and their interactions.

FORECASTING (PREDICTIVE ANALYTICS) **Forecasting** is predicting the future. This form of predictive analytics is essential for construction and manipulating models, because when a decision is implemented the results usually occur in the future. Whereas DSS are typically designed to determine what will be, traditional MIS report what is or what was. There is no point in running a what-if (sensitivity) analysis on the past, because decisions made then have no impact on the future. Forecasting is getting easier as software vendors automate many of the complications of developing such models.

E-commerce has created an immense need for forecasting and an abundance of available information for performing it. E-commerce activities occur quickly, yet information about purchases is gathered and should be analyzed to produce forecasts. Part of the analysis involves simply predicting demand; however, forecasting models can use product life-cycle needs and information about the marketplace and consumers to analyze the entire situation, ideally leading to additional sales of products and services.

Many organizations have accurately predicted demand for products and services, using a variety of qualitative and quantitative methods. But until recently, most companies viewed their customers and potential customers by categorizing them into only a few, time-tested groupings. Today, it is critical not only to consider customer characteristics, but also to consider how to get the right product(s) to the right customers at the right price at the right time in the right format/packaging. The more accurately a firm does this, the more profitable the firm is. In addition, a firm needs to recognize when not to sell a particular product or bundle of products to a particular set of customers. Part of this effort involves identifying lifelong customer profitability. These customer relationship management (CRM) system and revenue management system (RMS) approaches rely heavily on forecasting techniques, which are typically described as *predictive analytics*. These systems attempt to predict who their best (i.e., most profitable) customers (and worst ones as well) are and focus on identifying products and services at appropriate prices to appeal to them. We describe an effective example of such forecasting at Harrah's Cherokee Casino and Hotel in Application Case 9.2.

Application Case 9.2

Forecasting/Predictive Analytics Proves to Be a Good Gamble for Harrah's Cherokee Casino and Hotel

Harrah's Cherokee Casino and Hotel uses a revenue management (RM) system to optimize its profits. The system helps Harrah's attain an average 98.6 percent occupancy rate 7 days a week all year, with the exception of December, and a 60 percent gross revenue profit margin. One aspect of the RM system is providing its customers with Total Rewards cards, which track how much money each customer gambles. The system also tracks reservations and overbookings, with the exception of those made through third parties such as travel agencies. The RM system calculates the opportunity cost of saving rooms for possible customers who gamble more than others, because gambling is Harrah's main source of revenue. Unlike the traditional method of company employees only tracking the "big spenders," the RM system also tracks the "mid-tier" spenders. This has helped increase the company's profits. Only customers who gamble over a certain dollar amount are recommended by the RM system to be given rooms at the hotel; those who spend less may be given complimentary rooms at nearby hotels in order to keep the bigger spenders close by. The RM system also tracks which gaming machines are most popular so that management can place them strategically throughout the casino in order to encourage customers to gamble more money. Additionally, the system helps track the success of different marketing projects and incentives.

The casino collects demand data, which are then used by a forecasting algorithm with several components: smoothed values for base demand, demand trends, annual and day-of-the-week seasonality, and special event factors. The forecasts are used by overbooking and optimization models for inventory-control recommendations. The booking recommendation system includes a linear program (to be introduced later in the chapter). The model updates the recommendations for booking a room periodically or when certain events demand it. The bid-price model is updated or optimized after 24 hours have passed since the last optimization, when five rooms have been booked since the last optimization, or when the RM analyst manually starts a new optimization. The model is a good example of the process of forecasting demand and then using this information to employ a model-based DSS for making optimal decisions.

Source: Based on R. Metters, C. Queenan, M. Ferguson, L. Harrison, J. Higbie, S. Ward, B. Barfield, T. Farley, H. A. Kuyumcu, and A. Duggasani, "The 'Killer Application' of Revenue Management: Harrah's Cherokee Casino & Hotel," *Interfaces*, Vol. 38, No. 3, May/June 2008, pp. 161–175.

MODEL CATEGORIES Table 9.1 classifies DSS models into seven groups and lists several representative techniques for each category. Each technique can be applied to either a **static** or a **dynamic model,** which can be constructed under assumed environments of certainty, uncertainty, or risk. To expedite model construction, we can use special decision analysis systems that have modeling languages and capabilities embedded in them. These include spreadsheets, data mining systems, OLAP systems, and modeling languages that help an analyst build a model. We will introduce one of these systems later in the chapter.

MODEL MANAGEMENT Models, like data, must be managed to maintain their integrity, and thus their applicability. Such management is done with the aid of model base management systems (MBMS), which are analogous to database management systems (DBMS).

KNOWLEDGE-BASED MODELING DSS uses mostly quantitative models, whereas expert systems use qualitative, knowledge-based models in their applications. Some knowledge is necessary to construct solvable (and therefore usable) models. Many of the predictive analytics techniques such as classification, clustering, and so on can be used in building knowledge-based models. As described, such models can also be built from analysis of expertise and incorporation of such expertise in models.

CURRENT TRENDS IN MODELING One recent trend in modeling involves the development of model libraries and solution technique libraries. Some of these codes can be run directly on the owner's Web server for free, and others can be downloaded and run on a local computer. The availability of these codes means that powerful optimization and simulation packages are available to decision makers who may have only experienced these tools from the perspective of classroom problems. For example, the Mathematics and Computer Science Division at Argonne National Laboratory (Argonne, Illinois) maintains the NEOS Server for Optimization at **neos.mcs.anl.gov/neos/index.html**. You can find links to other sites by clicking the Resources link at **informs.org**, the Web site of the Institute for Operations Research and the Management Sciences (INFORMS). A wealth

TABLE 9.1 Categories of Models

Category	Process and Objective	Representative Techniques
Optimization of problems with few alternatives	Find the best solution from a small number of alternatives	Decision tables, decision trees, analytic hierarchy process
Optimization via algorithm	Find the best solution from a large number of alternatives, using a step-by-step improvement process	Linear and other mathematical programming models, network models
Optimization via an analytic formula	Find the best solution in one step, using a formula	Some inventory models
Simulation	Find a good enough solution or the best among the alternatives checked, using experimentation	Several types of simulation
Heuristics	Find a good enough solution, using rules	Heuristic programming, expert systems
Predictive models	Predict the future for a given scenario	Forecasting models, Markov analysis
Other models	Solve a what-if case, using a formula	Financial modeling, waiting lines

of modeling and solution information is available from INFORMS. The Web site for one of INFORMS' publications, *OR/MS Today,* at **lionhrtpub.com/ORMS.shtml** includes links to many categories of modeling software. We will learn about some of these shortly.

There is a clear trend toward developing and using Web tools and software to access and even run software to perform modeling, optimization, simulation, and so on. This has, in many ways, simplified the application of many models to real-world problems. However, to use models and solution techniques effectively, it is necessary to truly gain experience through developing and solving simple ones. This aspect is often overlooked. Another trend, unfortunately, involves the lack of understanding of what models and their solutions can do in the real world. Organizations that have key analysts who understand how to apply models indeed apply them very effectively. This is most notably occurring in the revenue management area, which has moved from the province of airlines, hotels, and automobile rental to retail, insurance, entertainment, and many other areas. CRM also uses models, but they are often transparent to the user. With management models, the amount of data and model sizes are quite large, necessitating the use of data warehouses to supply the data and parallel computing hardware to obtain solutions in a reasonable time frame.

There is a continuing trend toward making analytics models completely transparent to the decision maker. For example, **multidimensional analysis (modeling)** involves data analysis in several dimensions. In multidimensional analysis (modeling) and some other cases, data are generally shown in a spreadsheet format, with which most decision makers are familiar. Many decision makers accustomed to slicing and dicing data cubes are now using OLAP systems that access data warehouses. Although these methods may make modeling palatable, they also eliminate many important and applicable model classes from consideration, and they eliminate some important and subtle solution interpretation aspects. Modeling involves much more than just data analysis with trend lines and establishing relationships with statistical methods.

There is also a trend to build a model of a model to help in its analysis. An **influence diagram** is a graphical representation of a model; that is, it is a model of a model. Some influence diagram software packages are capable of generating and solving the resultant model.

SECTION 9.2 REVIEW QUESTIONS

1. List three lessons learned from modeling.
2. List and describe the major issues in modeling.
3. What are the major types of models used in DSS?
4. Why are models not used in industry as frequently as they should or could be?
5. What are the current trends in modeling?

9.3 STRUCTURE OF MATHEMATICAL MODELS FOR DECISION SUPPORT

In the following sections, we present the topics of analytical mathematical models (e.g., mathematical, financial, engineering). These include the components and the structure of models.

The Components of Decision Support Mathematical Models

All **quantitative models** are typically made up of four basic components (see Figure 9.1): result (or outcome) variables, decision variables, uncontrollable variables (and/or parameters), and intermediate result variables. Mathematical relationships link these components together. In non-quantitative models, the relationships are symbolic or qualitative. The results of decisions are determined based on the decision made (i.e., the values of the decision variables), the factors that cannot be controlled by the decision maker (in the

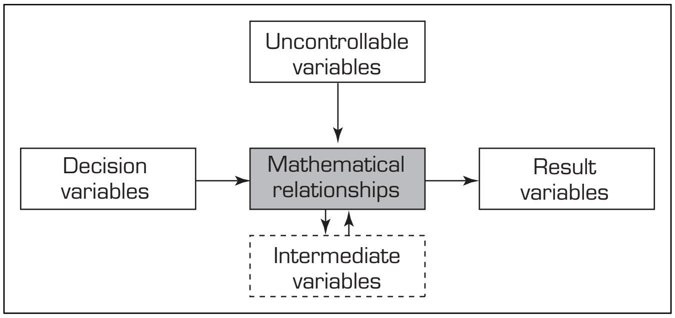


FIGURE 9.1 The General Structure of a Quantitative Model.

environment), and the relationships among the variables. The modeling process involves identifying the variables and relationships among them. Solving a model determines the values of these and the result variable(s).

RESULT (OUTCOME) VARIABLES **Result (outcome) variables** reflect the level of effectiveness of a system; that is, they indicate how well the system performs or attains its goal(s). These variables are outputs. Examples of result variables are shown in Table 9.2. Result variables are considered *dependent variables*. Intermediate result variables are sometimes used in modeling to identify intermediate outcomes. In the case of a dependent variable, another event must occur first before the event described by the variable can occur. Result variables depend on the occurrence of the decision variables and the uncontrollable variables.

DECISION VARIABLES **Decision variables** describe alternative courses of action. The decision maker controls the decision variables. For example, for an investment problem, the amount to invest in bonds is a decision variable. In a scheduling problem, the decision variables are people, times, and schedules. Other examples are listed in Table 9.2.

TABLE 9.2 Examples of the Components of Models

Area	Decision Variables	Result Variables	Uncontrollable Variables and Parameters
Financial investment	Investment alternatives and amounts	Total profit, risk Rate of return on investment (ROI) Earnings per share Liquidity level	Inflation rate Prime rate Competition
Marketing	Advertising budget Where to advertise	Market share Customer satisfaction	Customer's income Competitor's actions
Manufacturing	What and how much to produce Inventory levels Compensation programs	Total cost Quality level Employee satisfaction	Machine capacity Technology Materials prices
Accounting	Use of computers Audit schedule	Data processing cost Error rate	Computer technology Tax rates Legal requirements
Transportation	Shipments schedule Use of smart cards	Total transport cost Payment float time	Delivery distance Regulations
Services	Staffing levels	Customer satisfaction	Demand for services

UNCONTROLLABLE VARIABLES, OR PARAMETERS In any decision-making situation, there are factors that affect the result variables but are not under the control of the decision maker. Either these factors can be fixed, in which case they are called **uncontrollable variables,** or **parameters,** or they can vary, in which case they are called *variables*. Examples of factors are the prime interest rate, a city's building code, tax regulations, and utilities costs. Most of these factors are uncontrollable because they are in and determined by elements of the system environment in which the decision maker works. Some of these variables limit the decision maker and therefore form what are called the *constraints* of the problem.

INTERMEDIATE RESULT VARIABLES **Intermediate result variables** reflect intermediate outcomes in mathematical models. For example, in determining machine scheduling, spoilage is an intermediate result variable, and total profit is the result variable (i.e., spoilage is one determinant of total profit). Another example is employee salaries. This constitutes a decision variable for management: It determines employee satisfaction (i.e., intermediate outcome), which, in turn, determines the productivity level (i.e., final result).

The Structure of Mathematical Models

The components of a quantitative model are linked together by mathematical (algebraic) expressions—equations or inequalities.

A very simple financial model is

$$P = R - C$$

where P = profit, R = revenue, and C = cost. This equation describes the relationship among the variables. Another well-known financial model is the simple present-value cash flow model, where P = present value, F = a future single payment in dollars, i = interest rate (percentage), and n = number of years. With this model, it is possible to determine the present value of a payment of \$100,000 to be made 5 years from today, at a 10 percent (0.1) interest rate, as follows:

$$P = 100{,}000/(1 + 0.1)^5 = 62{,}092$$

We present more interesting and complex mathematical models in the following sections.

SECTION 9.3 REVIEW QUESTIONS

1. What is a decision variable?
2. List and briefly discuss the three major components of linear programming.
3. Explain the role of intermediate result variables.

9.4 CERTAINTY, UNCERTAINTY, AND RISK[1]

Part of Simon's decision-making process described in Chapter 2 involves evaluating and comparing alternatives; during this process, it is necessary to predict the future outcome of each proposed alternative. Decision situations are often classified on the basis of what the decision maker knows (or believes) about the forecasted results. We customarily classify this knowledge into three categories (see Figure 9.2), ranging from complete knowledge to complete ignorance:

- Certainty
- Risk
- Uncertainty

[1]Some parts of the original versions of these sections were adapted from Turban and Meredith (1994).

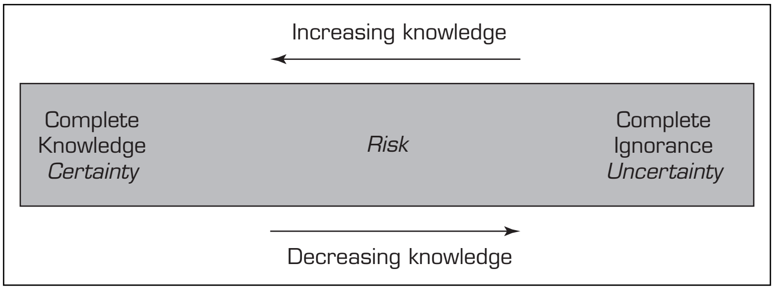


FIGURE 9.2 The Zones of Decision Making.

When we develop models, any of these conditions can occur, and different kinds of models are appropriate for each case. Next, we discuss both the basic definitions of these terms and some important modeling issues for each condition.

Decision Making Under Certainty

In decision making under **certainty,** it is *assumed* that complete knowledge is available so that the decision maker knows exactly what the outcome of *each course of action* will be (as in a deterministic environment). It may not be true that the outcomes are 100 percent known, nor is it necessary to really evaluate *all* the outcomes, but often this assumption simplifies the model and makes it tractable. The decision maker is viewed as a perfect predictor of the future because it is assumed that there is only one outcome for each alternative. For example, the alternative of investing in U.S. Treasury bills is one for which there is complete availability of information about the future return on the investment if it is held to maturity. A situation involving decision making under certainty occurs most often with structured problems with short time horizons (up to 1 year). Certainty models are relatively easy to develop and solve, and they can yield optimal solutions. Many financial models are constructed under assumed certainty, even though the market is anything but 100 percent certain.

Decision Making Under Uncertainty

In decision making under **uncertainty,** the decision maker considers situations in which several outcomes are possible for each course of action. In contrast to the risk situation, in this case, the decision maker does not know, or cannot estimate, the probability of occurrence of the possible outcomes. Decision making under uncertainty is more difficult than decision making under certainty because there is insufficient information. Modeling of such situations involves assessment of the decision maker's (or the organization's) attitude toward risk.

Managers attempt to avoid uncertainty as much as possible, even to the point of assuming it away. Instead of dealing with uncertainty, they attempt to obtain more information so that the problem can be treated under certainty (because it can be "almost" certain) or under calculated (i.e., assumed) risk. If more information is not available, the problem must be treated under a condition of uncertainty, which is less definitive than the other categories.

Decision Making Under Risk (Risk Analysis)

A decision made under **risk**[2] (also known as a *probabilistic* or *stochastic* decision-making situation) is one in which the decision maker must consider several possible outcomes for each alternative, each with a given probability of occurrence.

[2]Our definitions of the terms *risk* and *uncertainty* were formulated by F. H. Knight of the University of Chicago in 1933. Other, comparable definitions also are in use.

The long-run probabilities that the given outcomes will occur are assumed to be known or can be estimated. Under these assumptions, the decision maker can assess the degree of risk associated with each alternative (called *calculated* risk). Most major business decisions are made under assumed risk. **Risk analysis** (i.e., calculated risk) is a decision-making method that analyzes the risk (based on assumed known probabilities) associated with different alternatives. Risk analysis can be performed by calculating the expected value of each alternative and selecting the one with the best expected value. Application Case 9.3 illustrates one application to reduce uncertainty.

Application Case 9.3

American Airlines Uses Should-Cost Modeling to Assess the Uncertainty of Bids for Shipment Routes

Introduction

American Airlines, Inc. (AA) is one of the world's largest airlines. Its core business is passenger transportation but it has other vital ancillary functions that include full-truckload (FTL) freight shipment of maintenance equipment and in-flight shipment of passenger service items that could add up to over $1 billion in inventory at any given time. AA receives numerous bids from suppliers in response to request for quotes (RFQs) for inventories. AA's RFQs could total over 500 in any given year. Bid quotes vary significantly as a result of the large number of bids and resultant complex bidding process. Sometimes, a single contract bid could deviate by about 200 percent. As a result of the complex process, it is common to either overpay or underpay suppliers for their services. To this end, AA wanted a should-cost model that would streamline and assess bid quotes from suppliers in order to choose bid quotes that were fair to both them and their suppliers.

Methodology/Solution

In order to determine fair cost for supplier products and services, three steps were taken:

1. Primary (e.g., interviews) and secondary (e.g., Internet) sources were scouted for base-case and range data that would inform cost variables that affect an FTL bid.
2. Cost variables were chosen so that they were mutually exclusive and collectively exhaustive.
3. The DPL decision analysis software was used to model the uncertainty.

Furthermore, Extended Swanson-Megill (ESM) approximation was used to model the probability distribution of the most sensitive cost variables used. This was done in order to account for the high variability in the bids in the initial model.

Results/Benefits

A pilot test was done on an RFQ that attracted bids from six FTL carriers. Out of the six bids presented, five were within three standard deviations from the mean while one was considered an outlier. Subsequently, AA used the should-cost FTL model on more than 20 RFQs to determine what a fair and accurate cost of goods and services should be. It is expected that this model will help in reducing the risk of either overpaying or underpaying its suppliers.

Questions for Discussion

1. Besides reducing the risk of overpaying or underpaying suppliers, what are some other benefits AA would derive from its "should be" model?
2. Can you think of other domains besides air transportation where such a model could be used?
3. Discuss other possible methods with which AA could have solved its bid overpayment and underpayment problem.

Source: M. J. Bailey, J. Snapp, S. Yetur, J. S. Stonebraker, S. A. Edwards, A. Davis, and R. Cox, "Practice Summaries: American Airlines Uses Should-Cost Modeling to Assess the Uncertainty of Bids for Its Full-Truckload Shipment Routes," *Interfaces*, Vol. 41, No. 2, 2011, pp. 194–196.

SECTION 9.4 REVIEW QUESTIONS

1. Define what it means to perform decision making under assumed certainty, risk, and uncertainty.
2. How can decision-making problems under assumed certainty be handled?
3. How can decision-making problems under assumed uncertainty be handled?
4. How can decision-making problems under assumed risk be handled?

9.5 DECISION MODELING WITH SPREADSHEETS

Models can be developed and implemented in a variety of programming languages and systems. These range from third-, fourth-, and fifth-generation programming languages to computer-aided software engineering (CASE) systems and other systems that automatically generate usable software. We focus primarily on *spreadsheets* (with their add-ins), modeling languages, and transparent data analysis tools. With their strength and flexibility, spreadsheet packages were quickly recognized as easy-to-use implementation software for the development of a wide range of applications in business, engineering, mathematics, and science. Spreadsheets include extensive statistical, forecasting, and other modeling and database management capabilities, functions, and routines. As spreadsheet packages evolved, add-ins were developed for structuring and solving specific model classes. Among the add-in packages, many were developed for DSS development. These DSS-related add-ins include Solver (Frontline Systems Inc., **solver.com**) and What'sBest! (a version of Lindo, from Lindo Systems, Inc., **lindo.com**) for performing linear and nonlinear optimization; Braincel (Jurik Research Software, Inc., **jurikres.com**) and NeuralTools (Palisade Corp., **palisade.com**) for artificial neural networks; Evolver (Palisade Corp.) for genetic algorithms; and @RISK (Palisade Corp.) for performing simulation studies. Comparable add-ins are available for free or at a very low cost. (Conduct a Web search to find them; new ones are added to the marketplace on a regular basis.)

The spreadsheet is clearly the most popular *end-user modeling tool* because it incorporates many powerful financial, statistical, mathematical, and other functions. Spreadsheets can perform model solution tasks such as linear programming and regression analysis. The spreadsheet has evolved into an important tool for analysis, planning, and modeling (see Farasyn et al., 2008; Hurley and Balez, 2008; and Ovchinnikov and Milner, 2008). Application Case 9.4 describes an interesting application of a spreadsheet-based optimization model in a small business.

Application Case 9.4

Showcase Scheduling at Fred Astaire East Side Dance Studio

The Fred Astaire East Side Dance Studio in New York City presents two ballroom showcases a year. The studio wanted a cheap, user-friendly, and quick computer program to create schedules for its showcases that involved heats lasting around 75 seconds and solos lasting around 3 minutes. The program was created using an integer programming optimization model in Visual Basic and Excel. The employees just have to enter the students' names, the types of dances the students want to participate in, the teachers the students want to dance with, how many times the students want to do each type of dance, what times the students are unavailable, and what times the teachers are unavailable. This is entered into an Excel spreadsheet. The program then uses guidelines provided by the business to design the schedule. The guidelines include a dance type not being performed twice in a row if possible, a student participating in each quarter of the showcase in order to keep him/her active throughout, all participants in each heat performing the same type of dance (with a maximum of seven couples per heat), eliminating as many one-couple heats as possible, each student and teacher only being scheduled once per heat, and allowing students and teachers to dance multiple times per dance type if desired. A two-step heuristic method was used to help minimize the number of one-couple heats. In the end, the program cut down the time the employees spent creating the schedule and allowed for changes to be calculated and made quickly as compared to when made manually. For the summer 2007 showcase, the system scheduled 583 heat entries, 19 dance types, 18 solo entries, 28 students, and 8 teachers. This combination of Microsoft Excel and Visual Basic enabled the studio to use a model-based decision support system for a problem that could be time-consuming to solve.

Source: Based on M. A. Lejeune and N. Yakova, "Showcase Scheduling at Fred Astaire East Side Dance Studio," *Interfaces,* Vol. 38, No. 3, May/June 2008, pp. 176–186.

Other important spreadsheet features include what-if analysis, goal seeking, data management, and programmability (i.e., macros). With a spreadsheet, it is easy to change a cell's value and immediately see the result. Goal seeking is performed by indicating a target cell, its desired value, and a changing cell. Extensive database management can be performed with small data sets, or parts of a database can be imported for analysis (which is essentially how OLAP works with multidimensional data cubes; in fact, most OLAP systems have the look and feel of advanced spreadsheet software after the data are loaded). Templates, macros, and other tools enhance the productivity of building DSS.

Most spreadsheet packages provide fairly seamless integration because they read and write common file structures and easily interface with databases and other tools. Microsoft Excel is the most popular spreadsheet package. In Figure 9.3, we show a simple loan calculation model in which the boxes on the spreadsheet describe the contents of the cells, which contain formulas. A change in the interest rate in cell E7 is immediately reflected in the monthly payment in cell E13. The results can be observed and analyzed immediately. If we require a specific monthly payment, we can use goal seeking to determine an appropriate interest rate or loan amount.

Static or dynamic models can be built in a spreadsheet. For example, the monthly loan calculation spreadsheet shown in Figure 9.3 is static. Although the problem affects the borrower over time, the model indicates a single month's performance, which is replicated. A dynamic model, in contrast, represents behavior over time. The loan calculations in the spreadsheet shown in Figure 9.4 indicate the effect of prepayment on the principal over time. Risk analysis can be incorporated into spreadsheets by using built-in random-number generators to develop simulation models (see the next chapter).

Spreadsheet applications for models are reported regularly. We will learn how to use a spreadsheet-based optimization model in the next section.

SECTION 9.5 REVIEW QUESTIONS

1. What is a spreadsheet?
2. What is a spreadsheet add-in? How can add-ins help in DSS creation and use?
3. Explain why a spreadsheet is so conducive to the development of DSS.

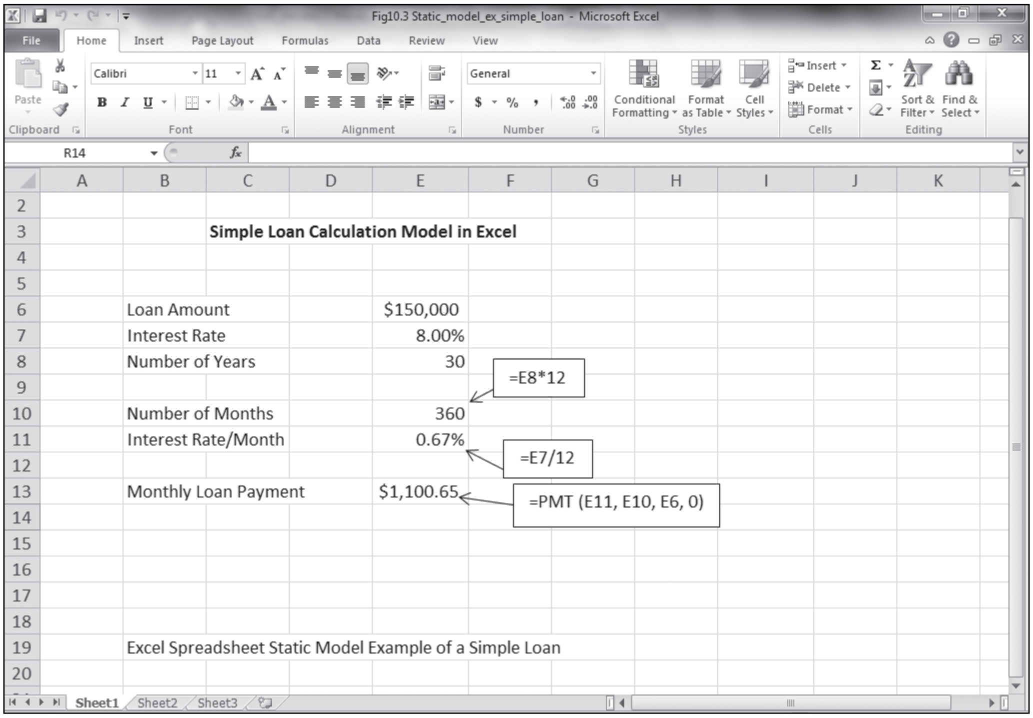


FIGURE 9.3 Excel Spreadsheet Static Model Example of a Simple Loan Calculation of Monthly Payments.

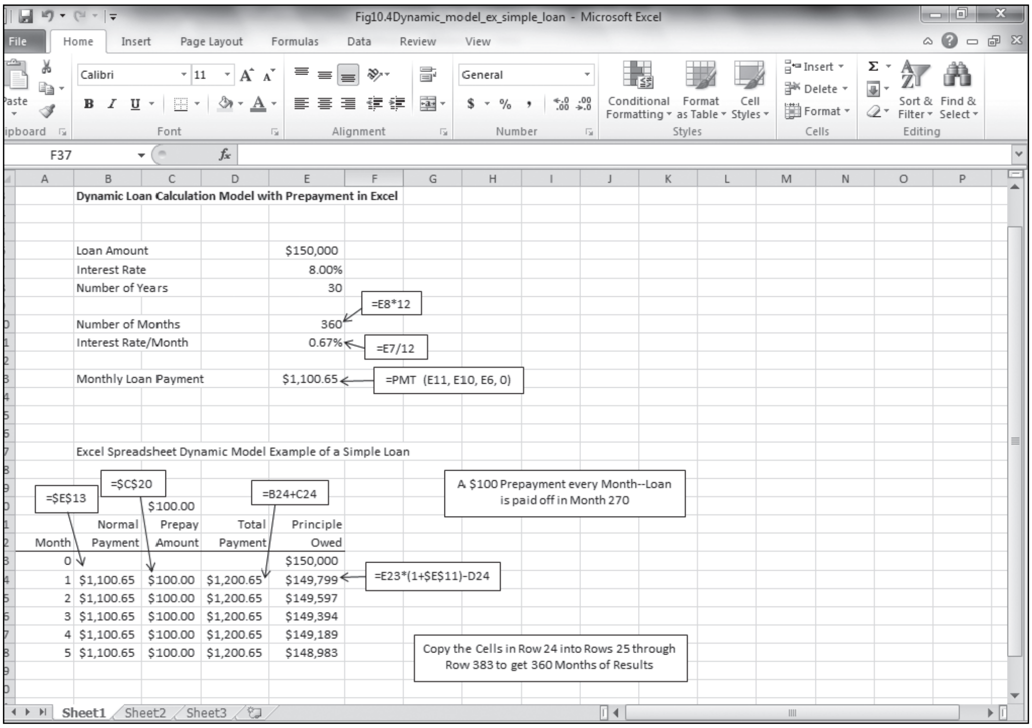


FIGURE 9.4 Excel Spreadsheet Dynamic Model Example of a Simple Loan Calculation of Monthly Payments and the Effects of Prepayment.

9.6 MATHEMATICAL PROGRAMMING OPTIMIZATION

The basic idea of optimization was introduced in Chapter 2. **Linear programming (LP)** is the best-known technique in a family of optimization tools called *mathematical programming*; in LP, all relationships among the variables are linear. It is used extensively in DSS (see Application Case 9.5). LP models have many important applications in practice. These include supply chain management, product mix decisions, routing, and so on. Special forms of the models can be used for specific applications. For example, Application Case 9.5 describes a spreadsheet model that was used to create a schedule for medical interns.

Application Case 9.5

Spreadsheet Model Helps Assign Medical Residents

Fletcher Allen Health Care (FAHC) is a teaching hospital that works with the University of Vermont's College of Medicine. In this particular case, FAHC employs 15 residents with hopes of adding 5 more in the diagnostic radiology program. Each year the chief radiology resident is required to make a yearlong schedule for all of the residents in radiology. This is a time-consuming process to do manually because there are many limitations on when each resident is and is not allowed to work. During the weekday working hours, the residents work with certified radiologists, but nights, weekends, and holidays are all staffed by residents only. The residents are also required to take the "emergency rotations," which involve taking care of the radiology needs of the emergency room, which is often the busiest on weekends. The radiology program is a 4-year program, and there are different rules for the work schedules of the residents for each year they are there. For example, first- and fourth-year residents cannot be on call on holidays, second-year residents cannot be on call or assigned ER shifts during 13-week blocks when they are assigned to work in Boston, and third-year residents must work one ER rotation during only one of the major winter holidays (Thanksgiving or Christmas/New Year's). Also, first-year residents cannot be on call until after January 1, and fourth-year residents cannot be on call after December 31, and so on. The goal that the various chief residents have each year is to give each person the maximum number of days between on-call days as is possible. Manually, only 3 days between on-call days was the most a chief resident had been able to accomplish.

In order to create a more efficient method of creating a schedule, the chief resident worked with an MS class of MBA students to develop a spreadsheet model to create the schedule. To solve this multiple-objective decision-making problem, the class used a constraint method made up of two stages. The first stage was to use the spreadsheet created in Excel as a calculator and to not use it for optimizing. This allowed the creators "to measure the key metrics of the residents' assignments, such as the number of days worked in each category." The second stage was an optimization model, which was layered on the calculator spreadsheet. Assignment constraints and the objective were added. The Solver engine in Excel was then invoked to find a feasible solution. The developers used Premium Solver by Frontline and the Xpress MP Solver engine by Dash Optimization to solve the yearlong model. Finally, using Excel functions, the developers converted the solution for a yearlong schedule from zeros and ones to an easy-to-read format for the residents. In the end, the program could solve the problem of a schedule with 3 to 4 days in between on calls instantly and with 5 days in between on calls (which was never accomplished manually).

Source: Based on A. Ovchinnikov and J. Milner, "Spreadsheet Model Helps to Assign Medical Residents at the University of Vermont's College of Medicine," *Interfaces,* Vol. 38, No. 4, July/August 2008, pp. 311–323.

Mathematical Programming

Mathematical programming is a family of tools designed to help solve managerial problems in which the decision maker must allocate scarce resources among competing activities to optimize a measurable goal. For example, the distribution of machine time (the resource) among various products (the activities) is a typical allocation problem. LP allocation problems usually display the following characteristics:

- A limited quantity of economic resources is available for allocation.
- The resources are used in the production of products or services.
- There are two or more ways in which the resources can be used. Each is called a *solution* or a *program*.
- Each activity (product or service) in which the resources are used yields a return in terms of the stated goal.
- The allocation is usually restricted by several limitations and requirements, called *constraints*.

The LP allocation model is based on the following rational economic assumptions:

- Returns from different allocations can be compared; that is, they can be measured by a common unit (e.g., dollars, utility).
- The return from any allocation is independent of other allocations.
- The total return is the sum of the returns yielded by the different activities.
- All data are known with certainty.
- The resources are to be used in the most economical manner.

Allocation problems typically have a large number of possible solutions. Depending on the underlying assumptions, the number of solutions can be either infinite or finite. Usually, different solutions yield different rewards. Of the available solutions, at least one is the best, in the sense that the degree of goal attainment associated with it is the highest (i.e., the total reward is maximized). This is called an **optimal solution,** and it can be found by using a special algorithm.

Linear Programming

Every LP problem is composed of *decision variables* (whose values are unknown and are searched for), an *objective function* (a linear mathematical function that relates the decision variables to the goal, measures goal attainment, and is to be optimized), *objective function coefficients* (unit profit or cost coefficients indicating the contribution to the objective of one unit of a decision variable), *constraints* (expressed in the form of linear inequalities or equalities that limit resources and/or requirements; these relate the variables through linear relationships), *capacities* (which describe the upper and sometimes lower limits on the constraints and variables), and *input/output (technology) coefficients* (which indicate resource utilization for a decision variable).

Let us look at an example. MBI Corporation, which manufactures special-purpose computers, needs to make a decision: How many computers should it produce next month at the Boston plant? MBI is considering two types of computers: the CC-7, which requires 300 days of labor and $10,000 in materials, and the CC-8, which requires 500 days of labor and $15,000 in materials. The profit contribution of each CC-7 is $8,000, whereas that of each CC-8 is $12,000. The plant has a capacity of 200,000 working days per month, and the material budget is $8 million per month. Marketing requires that at least 100 units of the CC-7 and at least 200 units of the CC-8 be produced each month. The problem is to maximize the company's profits by determining how many units of the CC-7 and how many units of the CC-8 should be produced each month. Note that in a real-world environment, it could possibly take months to obtain the data in the problem statement, and while gathering the data the decision maker would no doubt uncover facts about how to structure the model to be solved. Web-based tools for gathering data can help.

Modeling in LP: An Example

A standard LP model can be developed for the MBI Corporation problem just described. As discussed in Technology Insights 9.1, the LP model has three components: decision variables, result variables, and uncontrollable variables (constraints).

The decision variables are as follows:

$$X_1 = \text{unit of CC-7 to be produced}$$
$$X_2 = \text{unit of CC-8 to be produced}$$

The result variable is as follows:

$$\text{Total profit} = Z$$

The objective is to maximize total profit:

$$Z = 8{,}000X_1 + 12{,}000X_2$$

The uncontrollable variables (constraints) are as follows:

$$\text{Labor constraint: } 300X_1 + 500X_2 \leq 200{,}000 \text{ (in days)}$$
$$\text{Budget constraint: } 10{,}000X_1 + 15{,}000X_2 \leq 8{,}000{,}000 \text{ (in dollars)}$$
$$\text{Marketing requirment for CC-7: } X_1 \geq 100 \text{ (in units)}$$
$$\text{Marketing requirment for CC-8: } X_2 \geq 200 \text{ (in units)}$$

This information is summarized in Figure 9.5.

The model also has a fourth, hidden component. Every LP model has some internal intermediate variables that are not explicitly stated. The labor and budget constraints may each have some slack in them when the left-hand side is strictly less than the right-hand side. This slack is represented internally by slack variables that indicate excess resources available. The marketing requirement constraints may each have some surplus in them when the left-hand side is strictly greater than the right-hand side. This surplus is represented internally by surplus variables indicating that there is some room to adjust the right-hand sides of these constraints. These slack and surplus variables are intermediate. They can be of great value to a decision maker because LP solution methods use them in establishing sensitivity parameters for economic what-if analyses.

TECHNOLOGY INSIGHTS 9.1 Linear Programming

LP is perhaps the best-known optimization model. It deals with the optimal allocation of resources among competing activities. The allocation problem is represented by the model described here.

The problem is to find the values of the decision variables X_1, X_2, and so on, such that the value of the result variable Z is maximized, subject to a set of linear constraints that express the technology, market conditions, and other uncontrollable variables. The mathematical relationships are all linear equations and inequalities. Theoretically, any allocation problem of this type has an infinite number of possible solutions. Using special mathematical procedures, the LP approach applies a unique computerized search procedure that finds a best solution(s) in a matter of seconds. Furthermore, the solution approach provides automatic sensitivity analysis.

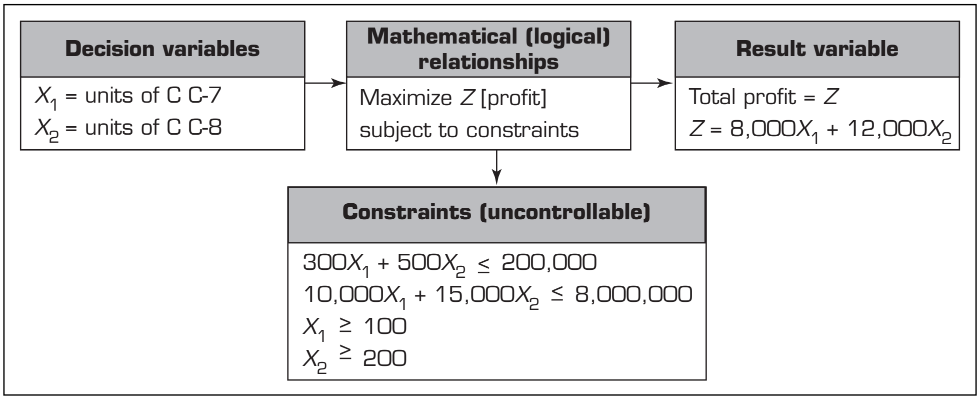


FIGURE 9.5 Mathematical Model of a Product-Mix Example.

The product-mix model has an infinite number of possible solutions. Assuming that a production plan is not restricted to whole numbers—which is a reasonable assumption in a monthly production plan—we want a solution that maximizes total profit: an optimal solution. Fortunately, Excel comes with the add-in Solver, which can readily obtain an optimal (best) solution to this problem. Although the location of Solver Add-in has moved from one version of Excel to another, it is still available as a free Add-in. Look for it under Data tab and on the Analysis ribbon. If it is not there, you should be able to enable it by going to Excel's Options Menu and selecting Add-ins.

We enter these data directly into an Excel spreadsheet, activate Solver, and identify the goal (by setting Target Cell equal to Max), decision variables (by setting By Changing Cells), and constraints (by ensuring that Total Consumed elements is less than or equal to Limit for the first two rows and is greater than or equal to Limit for the third and fourth rows). Cells C7 and D7 constitute the decision variable cells. Results in these cells would be filled after running the Solver Add-in. Target Cell is Cell E7, which is also the result variable, representing a product of decision variable cells and their per unit profit coefficients (in Cells C8 and D8). Note that all the numbers have been divided by 1,000 to make it easier to type (except the decision variables). Rows 9–12 describe the constraints of the problem: the constraints on labor capacity, budget, and the desired minimum production of the two products X_1 and X_2. Columns C and D define the coefficients of these constraints. Column E includes the formulae that multiply the decision variables (Cells C7 and D7) with their respective coefficients in each row. Column F defines the right-hand side value of these constraints. Excel's matrix multiplication capabilities (e.g., SUMPRODUCT function) can be used to develop such row and column multiplications easily.

After the model's calculations have been set up in Excel, it is time to invoke the Solver Add-in. Clicking on the Solver Add-in (again under the Analysis group under Data Tab) opens a dialog box (window) that lets you specify the cells or ranges that define the objective function cell, decision/changing variables (cells), and the constraints. Also, in Options, we select the solution method (usually Simplex LP), and then we solve the problem. Next, we select all three reports—Answer, Sensitivity, and Limits—to obtain an optimal solution of $X_1 = 333.33$, $X_2 = 200$, and Profit = \$5,066,667, as shown in Figure 9.6. Solver produces three useful reports about the solution. Try it. Solver now also includes the ability to solve nonlinear programming problems and integer programming problems by using other solution methods available within it.

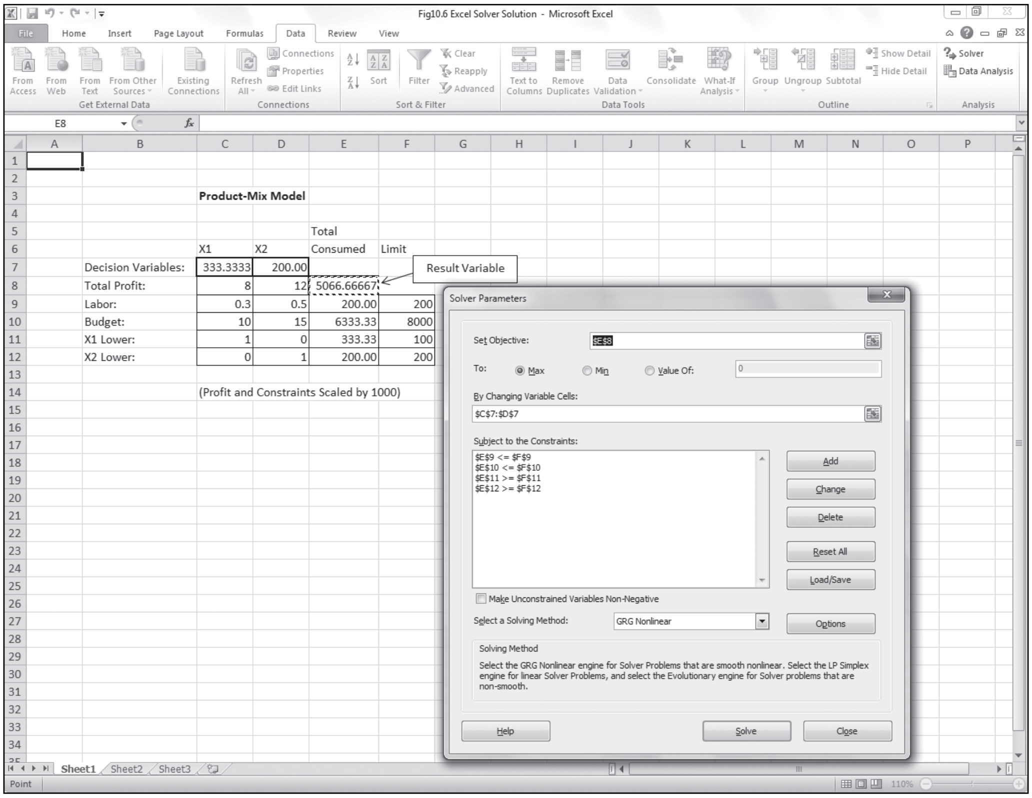


FIGURE 9.6 Excel Solver Solution to the Product-Mix Example.

The following example was created by Prof. Rick Wilson of Oklahoma State University to further illustrate the power of spreadsheet modeling for decision support.

The table in Figure 9.7 describes some estimated data and attributes of nine "swing states" for the 2012 election. Attributes of the nine states include their number of electoral votes, two regional descriptors (note that three states are classified as neither North or South), and an estimated "influence function," which relates to increased candidate support per unit of campaign financial investment in that state.

For instance, influence function F1 shows that for every financial unit invested in that state, there will be a total of a 10-unit increase in voter support (units will stay general here), made up of an increase in young men support by 3 units, old men support by 1 unit, and young and old women each by 3 units.

The campaign has 1,050 financial units to invest in the 9 states. It must invest at least 5 percent in each state of the total overall invested, but no more than 25 percent of the overall total invested can be in any one state. All 1,050 units do not have to be invested (your model must correctly deal with this).

The campaign has some other restrictions as well. From a financial investment standpoint, the West states (in total) must have campaign investment at levels that are at least 60 percent of the total invested in East states. In terms of people influenced, the decision to allocate financial investments to states must lead to at least 9,200 total people influenced. Overall, the total number of females influenced must be greater than or equal to the total number of males influenced. Also, at least 46 percent of all people influenced must be "old."

State	Electoral Votes	W/E	N/S	Influence Function
NV	6	West		F1
CO	9	West		F2
IA	6	West	North	F3
WI	10	West	North	F1
OH	18	East	North	F2
VA	13	East	South	F2
NC	15	East	South	F1
FL	29	East	South	F3
NH	4	East		F3

F1	Young	Old		
Men	3	1	4	
Women	3	3	6	
	6	4	10	Total

F2	Young	Old		
Men	1.5	2.5	4	
Women	2.5	1	3.5	
	4	3.5	7.5	Total

F3	Young	Old		
Men	2.5	2.5	5	
Women	1	2	3	
	3.5	4.5	8	Total

FIGURE 9.7 Data for Election Resource Allocation Example.

Our task is to create an appropriate integer programming model that determines the optimal integer (i.e., whole number) allocation of financial units to states that maximizes the sum of the products of the electoral votes times units invested subject to the other aforementioned restrictions. (Thus, indirectly, this model is giving preference to states with higher numbers of electoral votes). Note that for ease of implementation by the campaign staff, all decisions for allocation in the model should lead to integer values.

The three aspects of the models can be categorized based on the following questions that they answer:

1. ***What do we control?*** The amount invested in advertisements across the nine states, Nevada, Colorado, Iowa, Wisconsin, Ohio, Virginia, North Carolina, Florida, and New Hampshire, which are represented by the nine decision variables, NV, CO, IA, WI, OH, VA, NC, FL, and NH
2. ***What do we want to achieve?*** We want to maximize the total number of electoral votes gains. We know the value of each electoral vote in each state (EV), so this amounts to EV*Investments aggregated over the nine states, i.e.,

$$\text{Max}(6NV + 9CO + 6IA + 10WI + 18OH + 13VA + 15NC + 29FL + 4NH)$$

3. ***What constrains us?*** Following are the constraints as given in the problem description:

a. No more than 1,050 financial units to invest into, i.e., NV + CO + IA + WI + OH + VA + NC + FL + NH <= 1050.

b. Invest at least 5 percent of the total in each state, i.e.,

NV >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

CO >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

IA >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

WI >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

OH >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

VA >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

NC >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

FL >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

NH >= 0.05(NV + CO + IA + WI + OH + VA + NC + FL + NH)

We can implement these nine constraints in a variety of ways using Excel.

c. Invest no more than 25 percent of the total in each state.
As with (b) we need nine individual constraints again since we do not know how much of the 1,050 financial units we will invest. We must write the constraints on "general" terms.

NV <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

CO <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

IA <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

WI <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

OH <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

VA <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

NC <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

FL <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

NH <= 0.25(NV + CO + IA + WI + OH + VA + NC + FL + NH)

d. Western states must have investment levels that are at least 60 percent of the Eastern states.

West States = NV + CO + IA + WI

East States = OH + VA + NC + FL + NH

So, (NV + CO + IA + WI) >= 0.60(OH + VA + NC + FL + NH). Again we can implement this constraint in a variety of ways using Excel.

e. Influence at least 9,200 total people.

(10NV + 7.5CO + 8IA + 10WI + 7.5OH + 7.5VA +10NC + 8 FL + 8 NH) >= 9200

f. Influence at least as many females as males. This requires transition of influence functions.

F1 = 6 women influenced, F2 = 3.5 women
F3 = 3 women influenced
F1 = 4 men influenced, F2 = 4 men
F3 = 5 men influenced
So implementing females >= males, we get:

$$(6NV + 3.5CO + 3\,IA + 6WI + 3.5OH + 3.5VA + 6NC + 3FL + 3NH) >= (4NV + 4CO + 5IA + 4WI + 4OH + 4VA + 4NC + 5FL + 5NH)$$

As before, we can implement this in Excel in a couple of different ways.

g. At least 46 percent of all people influenced must be old.
All people influenced was on the left-hand side of the constraint (e). So, old people influenced would be:

$$(4NV + 3.5CO + 4.5IA + 4WI + 3.5OH + 3.5VA + 4NC + 4.5FL + 4.5NH)$$

This would be set >= 0.46* the left-hand side of constraint (e) (10NV + 7.5CO + 8IA + 10WI + 7.5OH + 7.5VA + 10NC + 8FL + 8NH), which would give a right-hand side of 0.46NV + 3.45CO + 3.68IA + 4.6WI + 3.45OH + 3.45VA + 4.6NC + 3.68FL + 3.68NH
This is the last constraint other than to force all variables to be integers.

All told in algebraic terms, this integer programing model would have 9 decision variables and 24 constraints (one constraint for integer requirements).

Implementation

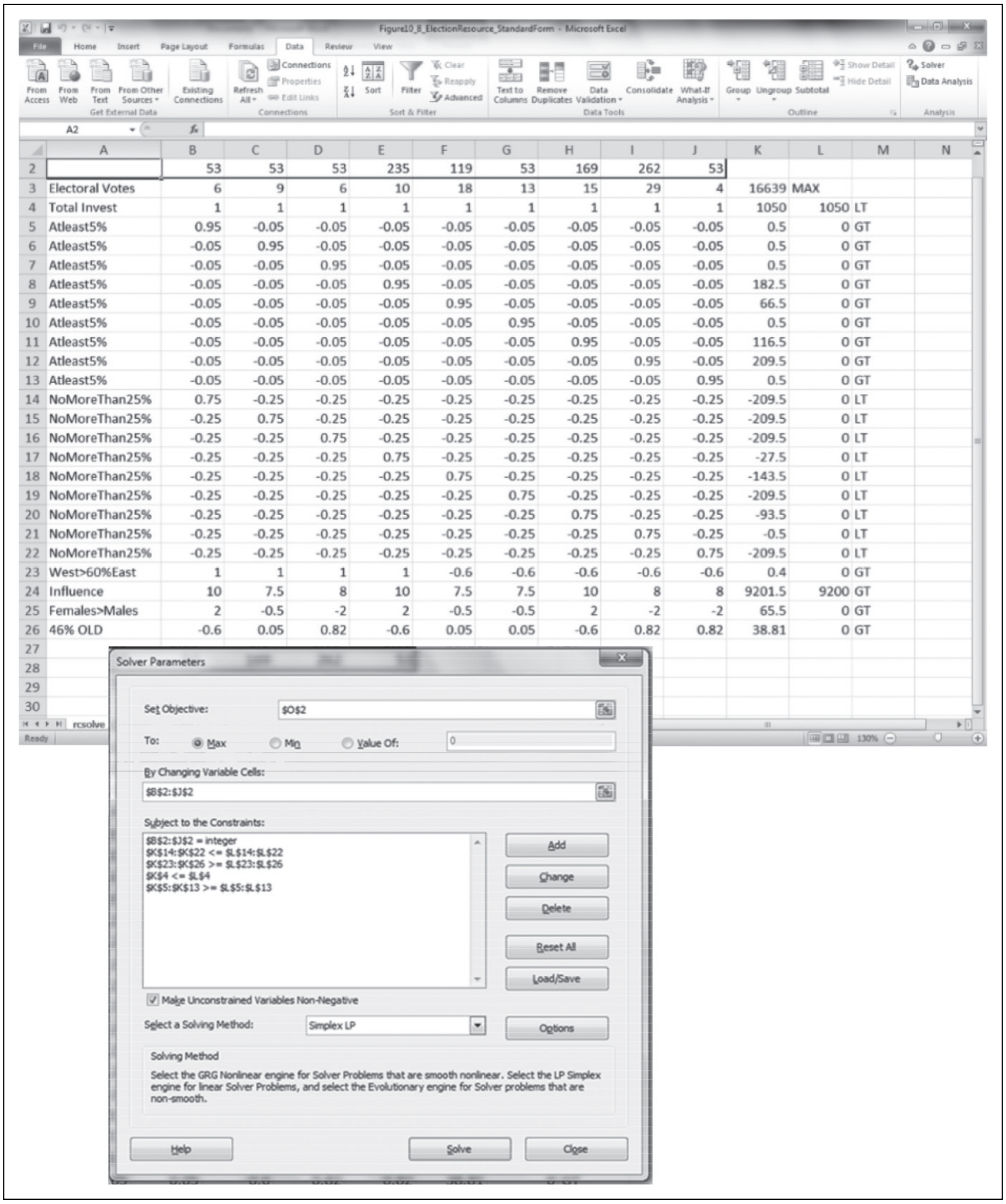
One approach would be to implement the model in strict "standard form," or a row-column form, where all constraints are written with decision variables on the left-hand side, and a number on the right-hand side. Figure 9.8 shows such an implementation and displays the solved model.

Alternatively, we could use the spreadsheet to calculate different parts of the model in a less rigid manner as well as uniquely implementing the repetitive constraints (b) and (c), and have a much more concise (but not as transparent) spreadsheet. This is shown in Figure 9.9.

LP models (and their specializations and generalizations) can be also specified directly in a number of other user-friendly modeling systems. Two of the best known are Lindo and Lingo (Lindo Systems, Inc., **lindo.com**; demos are available). Lindo is an LP and integer programming system. Models are specified in essentially the same way that they are defined algebraically. Based on the success of Lindo, the company developed Lingo, a modeling language that includes the powerful Lindo optimizer and extensions for solving nonlinear problems. Many other modeling languages such as AMPL, AIMMS, MPL, XPRESS, and others are available.

The most common optimization models can be solved by a variety of mathematical programming methods, including the following:

- Assignment (best matching of objects)
- Dynamic programming
- Goal programming
- Investment (maximizing rate of return)
- Linear and integer programming
- Network models for planning and scheduling
- Nonlinear programming
- Replacement (capital budgeting)
- Simple inventory models (e.g., economic order quantity)
- Transportation (minimize cost of shipments)



	A	B	C	D	E	F	G	H	I	J	K	L	M
2		53	53	53	235	119	53	169	262	53			
3	Electoral Votes	6	9	6	10	18	13	15	29	4	16639	MAX	
4	Total Invest	1	1	1	1	1	1	1	1	1	1050	1050	LT
5	Atleast5%	0.95	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.5	0	GT
6	Atleast5%	-0.05	0.95	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.5	0	GT
7	Atleast5%	-0.05	-0.05	0.95	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.5	0	GT
8	Atleast5%	-0.05	-0.05	-0.05	0.95	-0.05	-0.05	-0.05	-0.05	-0.05	182.5	0	GT
9	Atleast5%	-0.05	-0.05	-0.05	-0.05	0.95	-0.05	-0.05	-0.05	-0.05	66.5	0	GT
10	Atleast5%	-0.05	-0.05	-0.05	-0.05	-0.05	0.95	-0.05	-0.05	-0.05	0.5	0	GT
11	Atleast5%	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.95	-0.05	-0.05	116.5	0	GT
12	Atleast5%	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.95	-0.05	209.5	0	GT
13	Atleast5%	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	0.95	0.5	0	GT
14	NoMoreThan25%	0.75	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-209.5	0	LT
15	NoMoreThan25%	-0.25	0.75	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-209.5	0	LT
16	NoMoreThan25%	-0.25	-0.25	0.75	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-209.5	0	LT
17	NoMoreThan25%	-0.25	-0.25	-0.25	0.75	-0.25	-0.25	-0.25	-0.25	-0.25	-27.5	0	LT
18	NoMoreThan25%	-0.25	-0.25	-0.25	-0.25	0.75	-0.25	-0.25	-0.25	-0.25	-143.5	0	LT
19	NoMoreThan25%	-0.25	-0.25	-0.25	-0.25	-0.25	0.75	-0.25	-0.25	-0.25	-209.5	0	LT
20	NoMoreThan25%	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	0.75	-0.25	-0.25	-93.5	0	LT
21	NoMoreThan25%	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	0.75	-0.25	-0.5	0	LT
22	NoMoreThan25%	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	-0.25	0.75	-209.5	0	LT
23	West>60%East	1	1	1	1	-0.6	-0.6	-0.6	-0.6	-0.6	0.4	0	GT
24	Influence	10	7.5	8	10	7.5	7.5	10	8	8	9201.5	9200	GT
25	Females>Males	2	-0.5	-2	2	-0.5	-0.5	2	-2	-2	65.5	0	GT
26	46% OLD	-0.6	0.05	0.82	-0.6	0.05	0.05	-0.6	0.82	0.82	38.81	0	GT

FIGURE 9.8 Model for Election Resource Allocation—Standard Version.

SECTION 9.6 REVIEW QUESTIONS

1. List and explain the assumptions involved in LP.
2. List and explain the characteristics of LP.
3. Describe an allocation problem.
4. Define the product-mix problem.
5. Define the blending problem.
6. List several common optimization models.

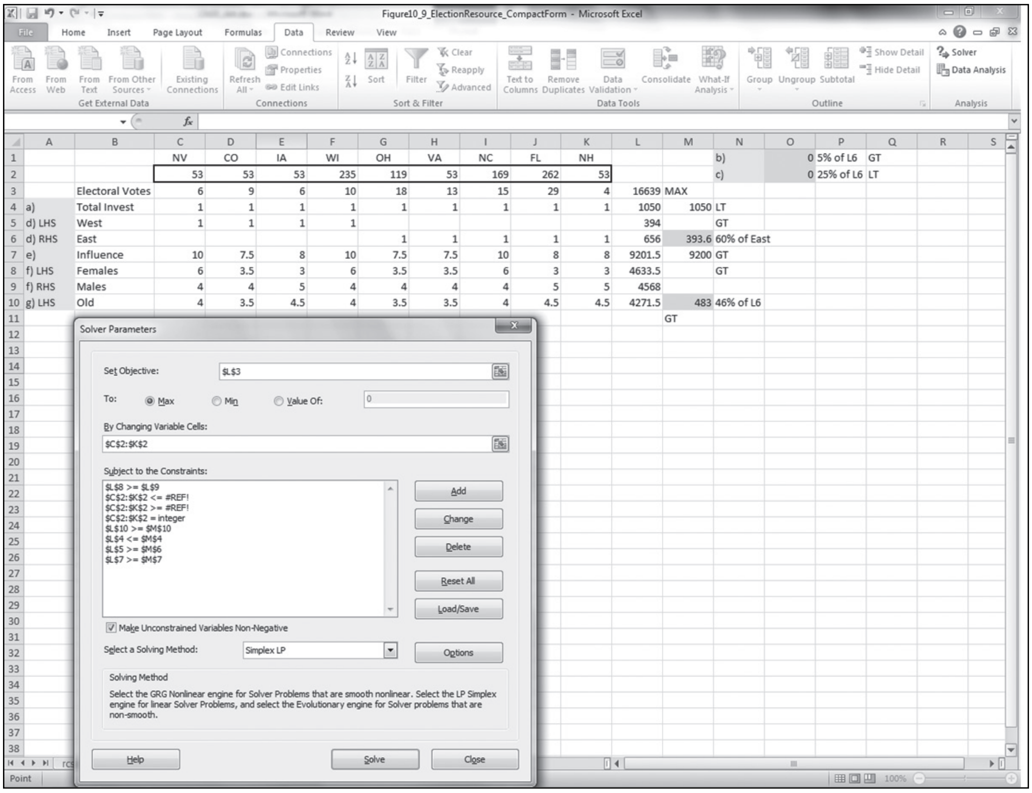


FIGURE 9.9 A Compact Formulation for Election Resource Allocation.

9.7 MULTIPLE GOALS, SENSITIVITY ANALYSIS, WHAT-IF ANALYSIS, AND GOAL SEEKING

The search process described earlier in this chapter is coupled with evaluation. Evaluation is the final step that leads to a recommended solution.

Multiple Goals

The analysis of management decisions aims at evaluating, to the greatest possible extent, how far each alternative advances managers toward their goals. Unfortunately, managerial problems are seldom evaluated with a single simple goal, such as profit maximization. Today's management systems are much more complex, and one with a single goal is rare. Instead, managers want to attain *simultaneous goals*, some of which may conflict. Different stakeholders have different goals. Therefore, it is often necessary to analyze each alternative in light of its determination of each of several goals (see Koksalan and Zionts, 2001).

For example, consider a profit-making firm. In addition to earning money, the company wants to grow, develop its products and employees, provide job security to its workers, and serve the community. Managers want to satisfy the shareholders and at the same time enjoy high salaries and expense accounts, and employees want to increase their take-home pay and benefits. When a decision is to be made—say, about an investment project—some of these goals complement each other, whereas others conflict. Kearns (2004) described how the analytic hierarchy process (AHP), which we will introduce in Section 9.9, combined with integer programming, addressed multiple goals in evaluating IT investments.

Many quantitative models of decision theory are based on comparing a single measure of effectiveness, generally some form of utility to the decision maker. Therefore, it is usually necessary to transform a multiple-goal problem into a single-measure-of-effectiveness problem before comparing the effects of the solutions. This is a common method for handling multiple goals in an LP model.

Certain difficulties may arise when analyzing multiple goals:

- It is usually difficult to obtain an explicit statement of the organization's goals.
- The decision maker may change the importance assigned to specific goals over time or for different decision scenarios.
- Goals and sub-goals are viewed differently at various levels of the organization and within different departments.
- Goals change in response to changes in the organization and its environment.
- The relationship between alternatives and their role in determining goals may be difficult to quantify.
- Complex problems are solved by groups of decision makers, each of whom has a personal agenda.
- Participants assess the importance (priorities) of the various goals differently.

Several methods of handling multiple goals can be used when working with MSS. The most common ones are:

- Utility theory
- Goal programming
- Expression of goals as constraints, using LP
- A points system

Sensitivity Analysis

A model builder makes predictions and assumptions regarding input data, many of which deal with the assessment of uncertain futures. When the model is solved, the results depend on these data. **Sensitivity analysis** attempts to assess the impact of a change in the input data or parameters on the proposed solution (i.e., the result variable).

Sensitivity analysis is extremely important in MSS because it allows flexibility and adaptation to changing conditions and to the requirements of different decision-making situations, provides a better understanding of the model and the decision-making situation it attempts to describe, and permits the manager to input data in order to increase the confidence in the model. Sensitivity analysis tests relationships such as the following:

- The impact of changes in external (uncontrollable) variables and parameters on the outcome variable(s)
- The impact of changes in decision variables on the outcome variable(s)
- The effect of uncertainty in estimating external variables
- The effects of different dependent interactions among variables
- The robustness of decisions under changing conditions

Sensitivity analyses are used for:

- Revising models to eliminate too-large sensitivities
- Adding details about sensitive variables or scenarios
- Obtaining better estimates of sensitive external variables
- Altering a real-world system to reduce actual sensitivities
- Accepting and using the sensitive (and hence vulnerable) real world, leading to the continuous and close monitoring of actual results

The two types of sensitivity analyses are automatic and trial-and-error.

AUTOMATIC SENSITIVITY ANALYSIS Automatic sensitivity analysis is performed in standard quantitative model implementations such as LP. For example, it reports the range within which a certain input variable or parameter value (e.g., unit cost) can vary without having any significant impact on the proposed solution. Automatic sensitivity analysis is usually limited to one change at a time, and only for certain variables. However, it is very powerful because of its ability to establish ranges and limits very fast (and with little or no additional computational effort). For example, automatic sensitivity analysis is part of the LP solution report for the MBI Corporation product-mix problem described earlier. Sensitivity analysis is provided by both Solver and Lindo. Sensitivity analysis could be used to determine that if the right-hand side of the marketing constraint on CC-8 could be decreased by one unit, then the net profit would increase by $1,333.33. This is valid for the right-hand side decreasing to zero. For details, see Hillier and Lieberman (2005) and Taha (2006) or later editions of these textbooks.

TRIAL-AND-ERROR SENSITIVITY ANALYSIS The impact of changes in any variable, or in several variables, can be determined through a simple trial-and-error approach. You change some input data and solve the problem again. When the changes are repeated several times, better and better solutions may be discovered. Such experimentation, which is easy to conduct when using appropriate modeling software, such as Excel, has two approaches: what-if analysis and goal seeking.

What-If Analysis

What-if analysis is structured as *What will happen to the solution if an input variable, an assumption, or a parameter value is changed?* Here are some examples:

- What will happen to the total inventory cost if the cost of carrying inventories increases by 10 percent?
- What will be the market share if the advertising budget increases by 5 percent?

With the appropriate user interface, it is easy for managers to ask a computer model these types of questions and get immediate answers. Furthermore, they can perform multiple cases and thereby change the percentage, or any other data in the question, as desired. The decision maker does all this directly, without a computer programmer.

Figure 9.10 shows a spreadsheet example of a what-if query for a cash flow problem. When the user changes the cells containing the initial sales (from 100 to 120) and the sales growth rate (from 3% to 4% per quarter), the program immediately recomputes the value of the annual net profit cell (from $127 to $182). At first, initial sales were 100, growing at 3 percent per quarter, yielding an annual net profit of $127. Changing the initial sales cell to 120 and the sales growth rate to 4 percent causes the annual net profit to rise to $182. What-if analysis is common in expert systems. Users are given the opportunity to change their answers to some of the system's questions, and a revised recommendation is found.

Goal Seeking

Goal seeking calculates the values of the inputs necessary to achieve a desired level of an output (goal). It represents a backward solution approach. The following are some examples of goal seeking:

- What annual R&D budget is needed for an annual growth rate of 15 percent by 2018?
- How many nurses are needed to reduce the average waiting time of a patient in the emergency room to less than 10 minutes?

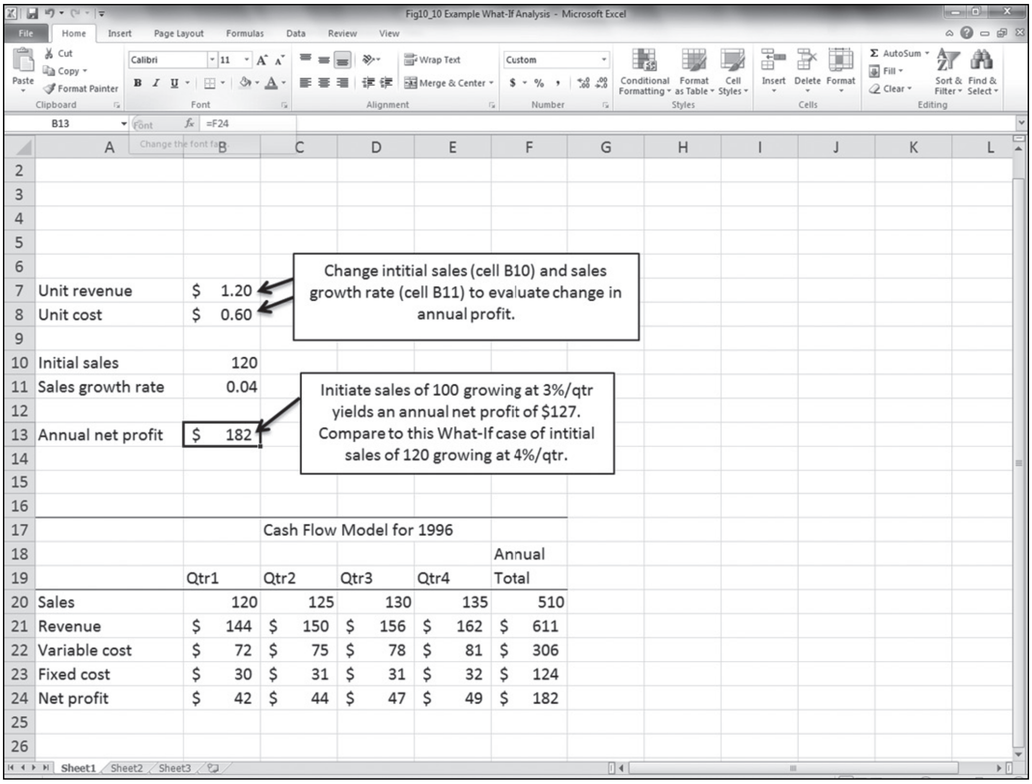


FIGURE 9.10 Example of a What-If Analysis Done in an Excel Worksheet.

An example of goal seeking is shown in Figure 9.11. For example, in a financial planning model in Excel, the internal rate of return is the interest rate that produces a net present value (NPV) of zero. Given a stream of annual returns in Column E, we can compute the net present value of planned investment. By applying goal seeking, we can determine the internal rate of return where the NPV is zero. The goal to be achieved is NPV equal to zero, which determines the internal rate of return (IRR) of this cash flow, including the investment. We set the NPV cell to the value 0 by changing the interest rate cell. The answer is 38.77059 percent.

COMPUTING A BREAK-EVEN POINT BY USING GOAL SEEKING Some modeling software packages can directly compute break-even points, which is an important application of goal seeking. This involves determining the value of the decision variables (e.g., quantity to produce) that generate zero profit.

In many general applications programs, it can be difficult to conduct sensitivity analysis because the prewritten routines usually present only a limited opportunity for asking what-if questions. In a DSS, the what-if and the goal-seeking options must be easy to perform.

SECTION 9.7 REVIEW QUESTIONS

1. List some difficulties that may arise when analyzing multiple goals.
2. List the reasons for performing sensitivity analysis.
3. Explain why a manager might perform what-if analysis.
4. Explain why a manager might use goal seeking.

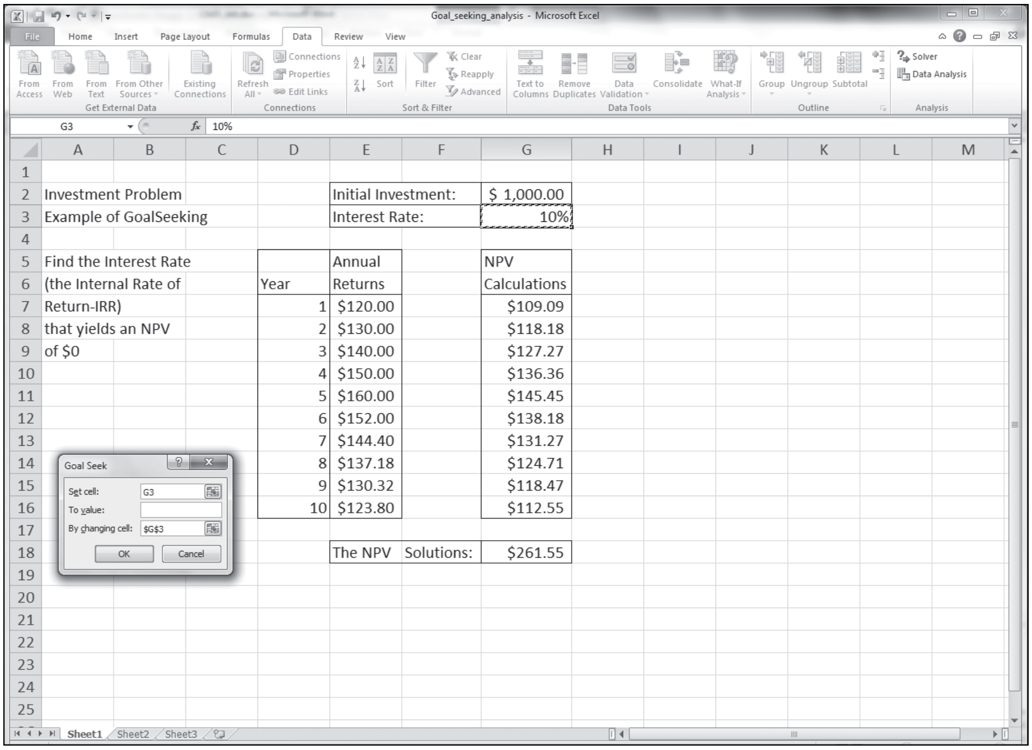


FIGURE 9.11 Goal-Seeking Analysis.

9.8 DECISION ANALYSIS WITH DECISION TABLES AND DECISION TREES

Decision situations that involve a finite and usually not too large number of alternatives are modeled through an approach called **decision analysis** (see Arsham, 2006a, 2006b; and Decision Analysis Society, **decision-analysis.society.informs.org**). Using this approach, the alternatives are listed in a table or a graph, with their forecasted contributions to the goal(s) and the probability of obtaining the contribution. These can be evaluated to select the best alternative.

Single-goal situations can be modeled with *decision tables* or *decision trees*. Multiple goals (criteria) can be modeled with several other techniques, described later in this chapter.

Decision Tables

Decision tables conveniently organize information and knowledge in a systematic, tabular manner to prepare it for analysis. For example, say that an investment company is considering investing in one of three alternatives: bonds, stocks, or certificates of deposit (CDs). The company is interested in one goal: maximizing the yield on the investment after one year. If it were interested in other goals, such as safety or liquidity, the problem would be classified as one of *multi-criteria decision analysis* (see Koksalan and Zionts, 2001).

The yield depends on the state of the economy sometime in the future (often called the *state of nature*), which can be in solid growth, stagnation, or inflation. Experts estimated the following annual yields:

- If there is solid growth in the economy, bonds will yield 12 percent, stocks 15 percent, and time deposits 6.5 percent.

- If stagnation prevails, bonds will yield 6 percent, stocks 3 percent, and time deposits 6.5 percent.
- If inflation prevails, bonds will yield 3 percent, stocks will bring a loss of 2 percent, and time deposits will yield 6.5 percent.

The problem is to select the one best investment alternative. These are assumed to be discrete alternatives. Combinations such as investing 50 percent in bonds and 50 percent in stocks must be treated as new alternatives.

The investment decision-making problem can be viewed as a *two-person game* (see Kelly, 2002). The investor makes a choice (i.e., a move), and then a state of nature occurs (i.e., makes a move). Table 9.3 shows the payoff of a mathematical model. The table includes *decision variables* (the alternatives), *uncontrollable variables* (the states of the economy; e.g., the environment), and *result variables* (the projected yield; e.g., outcomes). All the models in this section are structured in a spreadsheet framework.

If this were a decision-making problem under certainty, we would know what the economy will be and could easily choose the best investment. But that is not the case, so we must consider the two situations of uncertainty and risk. For uncertainty, we do not know the probabilities of each state of nature. For risk, we assume that we know the probabilities with which each state of nature will occur.

TREATING UNCERTAINTY Several methods are available for handling uncertainty. For example, the *optimistic approach* assumes that the best possible outcome of each alternative will occur and then selects the best of the best (i.e., stocks). The *pessimistic approach* assumes that the worst possible outcome for each alternative will occur and selects the best of these (i.e., CDs). Another approach simply assumes that all states of nature are equally possible. (See Clemen and Reilly, 2000; Goodwin and Wright, 2000; and Kontoghiorghes et al., 2002.) Every approach for handling uncertainty has serious problems. Whenever possible, the analyst should attempt to gather enough information so that the problem can be treated under assumed certainty or risk.

TREATING RISK The most common method for solving this risk analysis problem is to select the alternative with the greatest expected value. Assume that experts estimate the chance of solid growth at 50 percent, the chance of stagnation at 30 percent, and the chance of inflation at 20 percent. The decision table is then rewritten with the known probabilities (see Table 9.4). An expected value is computed by multiplying the results (i.e., outcomes) by their respective probabilities and adding them. For example, investing in bonds yields an expected return of $12(0.5) + 6(0.3) + 3(0.2) = 8.4$ percent.

This approach can sometimes be a dangerous strategy because the utility of each potential outcome may be different from the value. Even if there is an infinitesimal chance of a catastrophic loss, the expected value may seem reasonable, but the investor may not be willing to cover the loss. For example, suppose a financial advisor presents you with an "almost sure" investment of $1,000 that can double your money in one day, and then

TABLE 9.3 Investment Problem Decision Table Model

	State of Nature (Uncontrollable Variables)		
Alternative	**Solid Growth (%)**	**Stagnation (%)**	**Inflation (%)**
Bonds	12.0	6.0	3.0
Stocks	15.0	3.0	–2.0
CDs	6.5	6.5	6.5

TABLE 9.4 Multiple Goals

Alternative	Yield (%)	Safety	Liquidity
Bonds	8.4	High	High
Stocks	8.0	Low	High
CDs	6.5	Very high	High

the advisor says, "Well, there is a .9999 probability that you will double your money, but unfortunately there is a .0001 probability that you will be liable for a $500,000 out-of-pocket loss." The expected value of this investment is as follows:

$$0.9999(\$2{,}000 - \$1{,}000) + .0001(-\$500{,}000 - \$1{,}000) = \$999.90 - \$50.10$$
$$= \$949.80$$

The potential loss could be catastrophic for any investor who is not a billionaire. Depending on the investor's ability to cover the loss, an investment has different expected utilities. Remember that the investor makes the decision only *once*.

Decision Trees

An alternative representation of the decision table is a decision tree (for examples, see Mind Tools Ltd., **mindtools.com**). A **decision tree** shows the relationships of the problem graphically and can handle complex situations in a compact form. However, a decision tree can be cumbersome if there are many alternatives or states of nature. TreeAge Pro (TreeAge Software Inc., **treeage.com**) and PrecisionTree (Palisade Corp., **palisade.com**) include powerful, intuitive, and sophisticated decision tree analysis systems. These vendors also provide excellent examples of decision trees used in practice. Note that the phrase *decision tree* has been used to describe two different types of models and algorithms. In the current context, decision trees refer to scenario analysis. On the other hand, some classification algorithms in predictive analysis (see Chapters 5 and 6) also are called decision tree algorithms.

A simplified investment case of **multiple goals** (a decision situation in which alternatives are evaluated with several, sometimes conflicting, goals) is shown in Table 9.4. The three goals (criteria) are yield, safety, and liquidity. This situation is under assumed certainty; that is, only one possible consequence is projected for each alternative; the more complex cases of risk or uncertainty could be considered. Some of the results are qualitative (e.g., low, high) rather than numeric.

See Clemen and Reilly (2000), Goodwin and Wright (2000), and Decision Analysis Society (**faculty.fuqua.duke.edu/daweb**) for more on decision analysis. Although doing so is quite complex, it is possible to apply mathematical programming directly to decision-making situations under risk. We discuss several other methods of treating risk in the next few chapters. These include simulation and certainty factors.

SECTION 9.8 REVIEW QUESTIONS

1. What is a decision table?
2. What is a decision tree?
3. How can a decision tree be used in decision making?
4. Describe what it means to have multiple goals.

9.9 MULTI-CRITERIA DECISION MAKING WITH PAIRWISE COMPARISONS

Multi-criteria (goal) decision making was introduced in Chapter 2. One of the most effective approaches is to use weights based on decision-making priorities. However, soliciting weights (or priorities) from managers is a complex task, as is calculation of the weighted averages needed to choose the best alternative. The process is complicated further by the presence of qualitative variables. One method of multi-criteria decision making is the analytic hierarchy process developed by Saaty.

The Analytic Hierarchy Process

The **analytic hierarchy process (AHP),** developed by Thomas Saaty (1995, 1996), is an excellent modeling structure for representing *multi-criteria* (multiple goals, multiple objectives) *problems*—with sets of criteria and alternatives (choices)—commonly found in business environments. The decision maker uses AHP to decompose a decision-making problem into relevant criteria and alternatives. The AHP separates the analysis of the criteria from the alternatives, which helps the decision maker to focus on small, manageable portions of the problem. The AHP manipulates quantitative and qualitative decision-making criteria in a fairly structured manner, allowing a decision maker to make trade-offs quickly and "expertly." Application Case 9.6 gives an example of an application of AHP in selection of IT projects.

Application Case 9.6

U.S. HUD Saves the House by Using AHP for Selecting IT Projects

The U.S. Department of Housing and Urban Development's (HUD) mission is to increase homeownership, support community development, and increase access to affordable housing free from discrimination. HUD's total annual budget is $32 billion with roughly $400 million allocated to IT spending each year. HUD was annually besieged by requests for IT projects by its program areas, but had no rational process that allowed management to select and monitor the best projects within its budgetary constraints. Like most federal agencies, HUD was required by congressional act to hire a CIO and develop an IT capital planning process. However, it wasn't until the Office of Management and Budget (OMB) threatened to cut agency budgets in 1999 that an IT planning process was actually developed and implemented at HUD. There had been a great deal of wasted money and manpower in the duplication of efforts by program areas, a lack of a sound project prioritization process, and no standards or guidelines for the program areas to follow.

For example, in 1999 there were requests for over $600 million in HUD IT projects against an IT budget of less than $400 million. There were over 200 approved projects but no process for selecting, monitoring, and evaluating these projects. HUD could not determine whether its selected IT projects were properly aligned with the agency's mission and objectives and were thus the most effective projects.

The agency determined from best practices and industry research that it needed both a rational process and a tool to support this process to meet OMB's requirements. Using the results from this research, HUD recommended that a process and guidelines be developed that would allow senior HUD management to select and prioritize the objectives and selection criteria while allowing the program teams to score specific project requests. HUD now uses the analytic hierarchy process through Expert Choice software with its capital planning process to select, manage, and evaluate its IT portfolio in real time, while the selected IT programs are being implemented.

The results have been staggering: With the new methodology and Expert Choice, HUD has reduced the preparation and meeting time for the

(*Continued*)

Application Case 9.5 (Continued)

annual selection and prioritization of IT projects from months to mere weeks, saving time and management hours. Program area requests of recent IT budgets dropped from the 1999 level of over $600 million to less than $450 million as managers recognized that the selection criteria for IT projects were going to be fairly and stringently applied by senior management, and that the number of projects funded had dropped from 204 to 135. In the first year of implementation, HUD reallocated $55 million of its IT budget to more effective projects that were better aligned with the agency's objectives.

In addition to saving time, the fair and transparent process has increased buy-in at all levels of management. There are few opportunities or incentives, if any, for an "end run" around the process. HUD now requires that each assistant secretary for the program areas sign off on the weighted selection criteria, and managers now know that special requests are likely fruitless if they cannot be supported by the selection criteria.

Source: **http://expertchoice.com/xres/uploads/resource-center-documents/HUD_casestudy.pdf** (accessed February 2013).

Expert Choice (**expertchoice.com**; a demo is available directly on its Web site) is an excellent commercial implementation of AHP. A problem is represented as an inverted tree with a goal node at the top. All the weight of the decision is in the goal (1.000). Directly beneath and attached to the goal node are the criteria nodes. These are the factors that are important to the decision maker. The goal is decomposed into criteria, to which 100 percent of the weight of the decision from the goal is distributed. To distribute the weight, the decision maker conducts pairwise comparisons of the criteria: first criterion to second, first to third,..., first to last; then, second to third,..., second to last;...; and then the next-to-last criterion to the last one. This establishes the importance of each criterion; that is, how much of the goal's weight is distributed to each criterion (how *important* each criterion is). This objective method is performed by internally manipulating matrices mathematically. The manipulations are transparent to the user because the operational details of the method are *not* important to the decision maker. Finally, an inconsistency index indicates how consistent the comparisons were, thus identifying inconsistencies, errors in judgment, or simply errors. The AHP method is consistent with decision theory.

The decision maker can make comparisons verbally (e.g., one criterion is moderately more important than another), graphically (with bar and pie charts), or numerically (with a *matrix*—comparisons are scaled from 1 to 9). Students and business professionals generally prefer graphical and verbal approaches over matrices (based on an informal sample).

Beneath each criterion are the same sets of choices (alternatives) in the simple case described here. Like the goal, the criteria decompose their weight into the choices, which capture 100 percent of the weight of each criterion. The decision maker performs a pairwise comparison of choices in terms of *preferences,* as they relate to the specific criterion under consideration. Each set of choices must be pairwise compared as they relate to each criterion. Again, all three modes of comparison are available, and an inconsistency index is derived for each set and reported.

Finally, the results are synthesized and displayed on a bar graph. The choice with the most weight is the correct choice. However, under some conditions the correct decision may not be the right one. For example, if there are two "identical" choices (e.g., if you are selecting a car for purchase and you have two identical cars), they may split the weight and neither will have the most weight. Also, if the top few choices are very close, there may be a missing criterion that could be used to differentiate among these choices.

Expert Choice also has a sensitivity analysis module. A newer version of the product, called Comparion, also synthesizes the results of a group of decision makers using the same model. This version can work on the Web. Overall, AHP as implemented in Expert Choice attempts to derive a decision maker's preference (utility) structure in terms of the criteria and choices and help him or her to make an expert choice.

In addition to Expert Choice, other software packages allow for weighting of pairwise choices. For example, Web-HIPRE (**hipre.aalto.fi**), an adaptation of AHP and several other weighting schemes, enables a decision maker to create a decision model, enter pairwise preferences, and analyze the optimal choice. These weightings can be computed using AHP as well as other techniques. It is available as a Java applet on the Web so it can be easily located and run online, free for noncommercial use. To run Web-HIPRE, one has to access the site and leave a Java applet window running. The user can enter a problem by providing the general labels for the decision tree at each node level and then entering the problem components. After the model has been specified, the user can enter pairwise preferences at each node level for criteria/subcriteria/alternative. Once that is done, the appropriate analysis algorithm can be used to determine the model's final recommendation. The software can also perform sensitivity analysis to determine which criteria/subcriteria play a dominant role in the decision process. Finally, the Web-HIPRE can also be employed in group mode. In the following paragraphs, we provide a tutorial on using AHP through Web-HIPRE.

Tutorial on Applying Analytic Hierarchy Process Using Web-HIPRE

The following paragraphs give an example of application of the analytic hierarchy process in making a decision to select a movie that suits an individual's interest. Phrasing the decision problem in AHP terminology:

1. The goal is to select the most appropriate movie of interest.
2. Let us identify some criteria for making this decision. To get started, let us agree that the main criteria for movie selection are genre, language, day of release, user/critics rating.
3. The subcriteria for each of main criteria are listed here:
   a. Genre: Action, Comedy, Sci-Fi, Romance
   b. Language: English, Hindi
   c. Day of Release: week day, weekend
   d. User/Critics Rating: High, Average, Low
4. Let us assume that the alternatives are the following current movies: *SkyFall, The Dark Knight Rises, The Dictator, Dabaang, Alien,* and *DDL.*

The following steps enable setting up the AHP using Web-HIPRE. The same can be done using commercial strength software such as Expert Choice/Comparion and many other tools. As mentioned earlier, Web-HIPRE can be accessed online at **hipre.aalto.fi**

Step 1 Web-HIPRE allows the users to create the goal, associated main criteria, subcriteria and the alternatives, and establish appropriate relationships among each of them. Once the application is opened, double-clicking on the diagram space allows users to create all the elements, which are renamed as the goal, criteria, and alternatives. Selecting an element and right-clicking on the desired element will create a relationship between these two elements.

Figure 9.12 shows the entire view of the sample decision problem of selecting a movie: a sequence of goal, main criteria, subcriteria, and the alternatives.

Step 2 All of the main criteria related to the goal are then ranked with their relative importance over each other using a comparative ranking scale ranging from 1 to 9, with ascending order of importance. To begin entering your pairwise

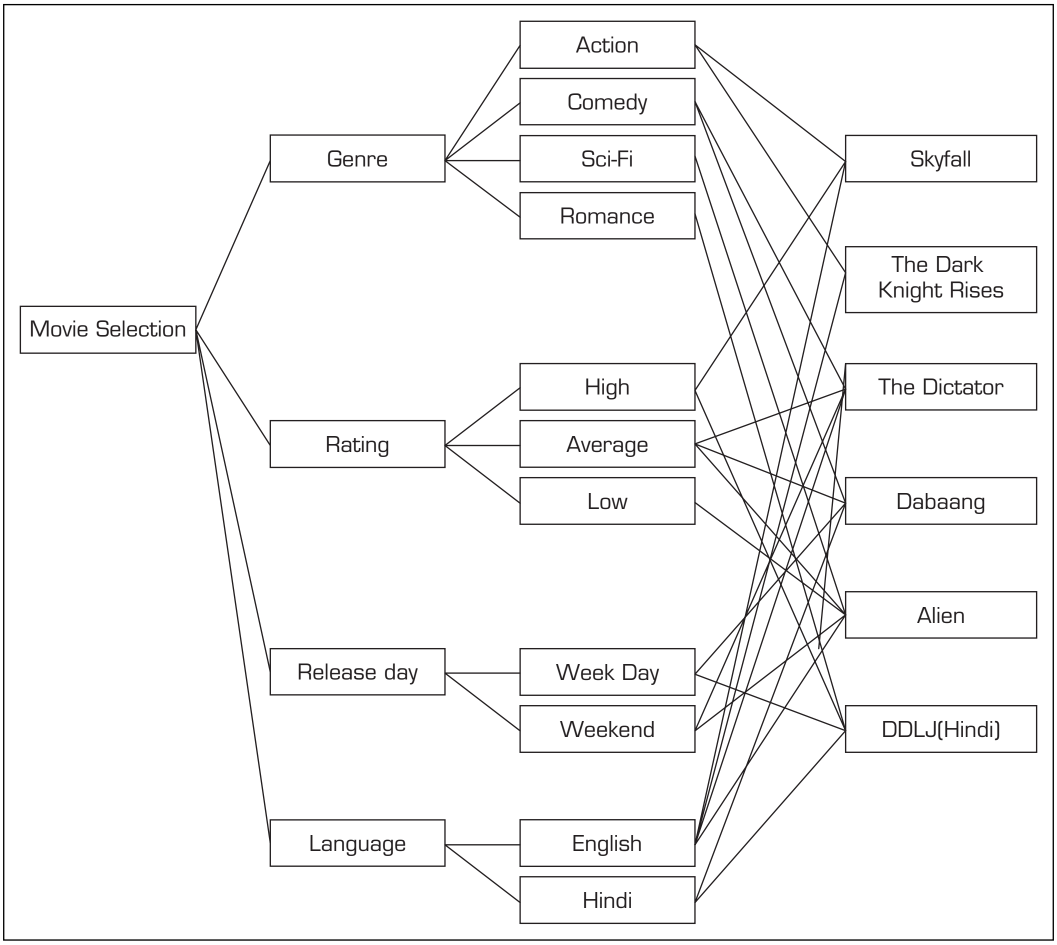


FIGURE 9.12 Main AHP Diagram.

priorities for any element's children nodes, you click on the Priorities Menu, and then select AHP as the method of ranking. Again, note that each comparison is made between just two competing criteria/subcriteria or alternatives with respect to the parent node. For example, in the current problem, the rating of the movie was considered to be the most important criterion, followed by genre, release day, and language. The criteria are ranked or rated in a pairwise mode with respect to the parent node—the goal of selecting a movie. The tool readily normalizes the rankings of each of the main criteria over one another to a scale ranging from 0 to 1 and then calculates the row averages to arrive at an overall importance rating ranging from 0 to 1.

Figure 9.13 shows the main criteria ranked over one another and the final ranking of each of the main criteria.

Step 3 All of the subcriteria related to each of the main criteria are then ranked with their relative importance over one another. In the current example, one of the main criteria, Genre, the subcriterion Comedy is ranked with higher importance followed by Action, Romance, and Sci-Fi. The ranking is normalized and averaged to yield a final score ranging between 0 and 1. Likewise, for each of the main criteria, all subcriteria are relatively ranked over one another.

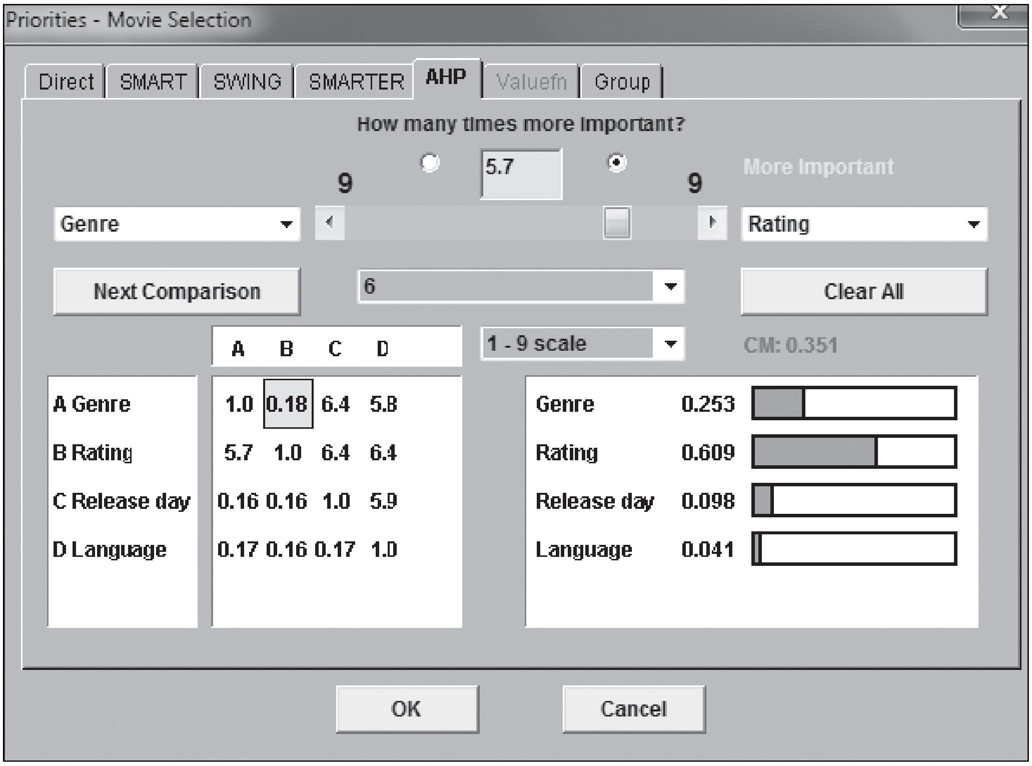


FIGURE 9.13 **Ranking Main Criteria.**

Figure 9.14 shows the subcriteria ranked over one another and the final ranking of each of the subcriteria with respect to the main criterion, Genre.

Step 4 Each alternative is ranked with respect to all of the subcriteria that are linked with the alternatives in a similar fashion using the relative scale of 0–9. Then the overall importance of each alternative is calculated using normalization and row averages of rankings of each of the alternatives.

Figure 9.15 shows the alternatives specific to Comedy–Sub-Genre being ranked over each other.

Step 5 The final result of the relative importance of each of the alternatives, with respect to the weighted scores of subcriteria, as well as the main criteria, is obtained from the composite priority analysis involving all the subcriteria and main criteria associated with each of the alternatives. The alternative with the highest composite score, in this case, the movie *The Dark Knight Rises,* is then selected as the right choice for the main goal.

Figure 9.16 shows the composite priority analysis.

Note that this example follows a top-down approach of choosing alternatives by first setting up priorities among the main criteria and subcriteria, eventually evaluating the relative importance of alternatives. Similarly, a bottom-up approach of first evaluating the alternatives with respect to the subcriteria and then setting up priorities among subcriteria and main criteria can also be followed in choosing a particular alternative.

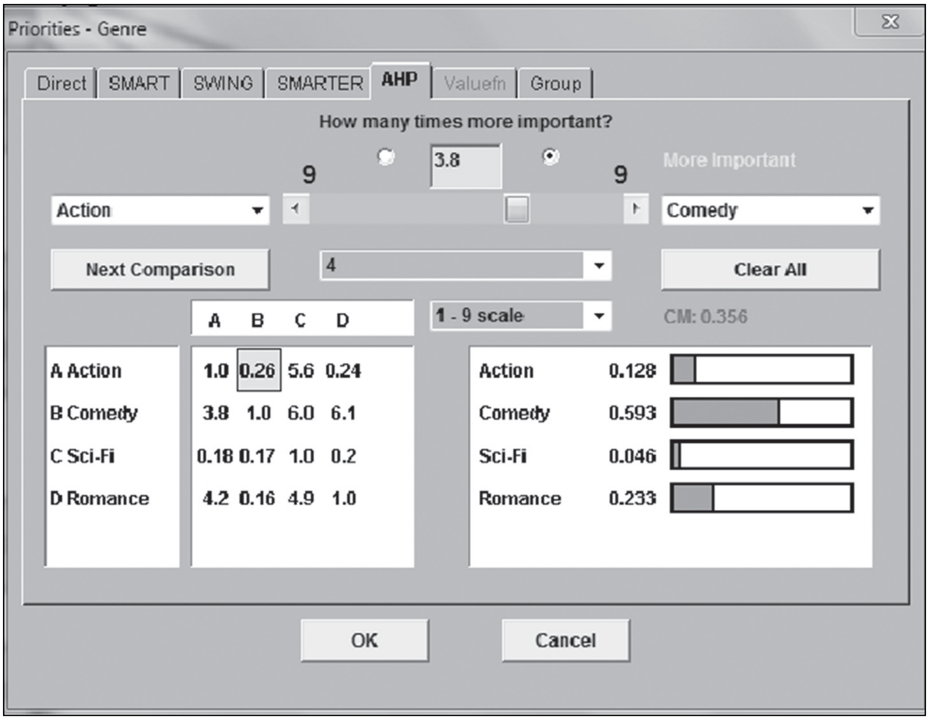


FIGURE 9.14 Ranking Subcriteria.

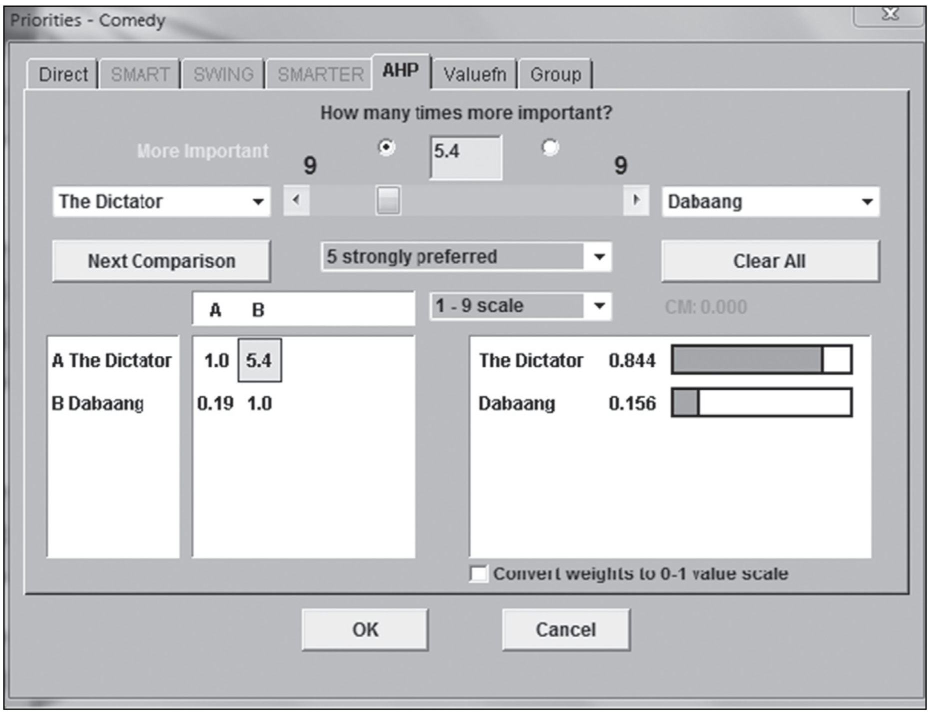


FIGURE 9.15 Ranking Alternatives.

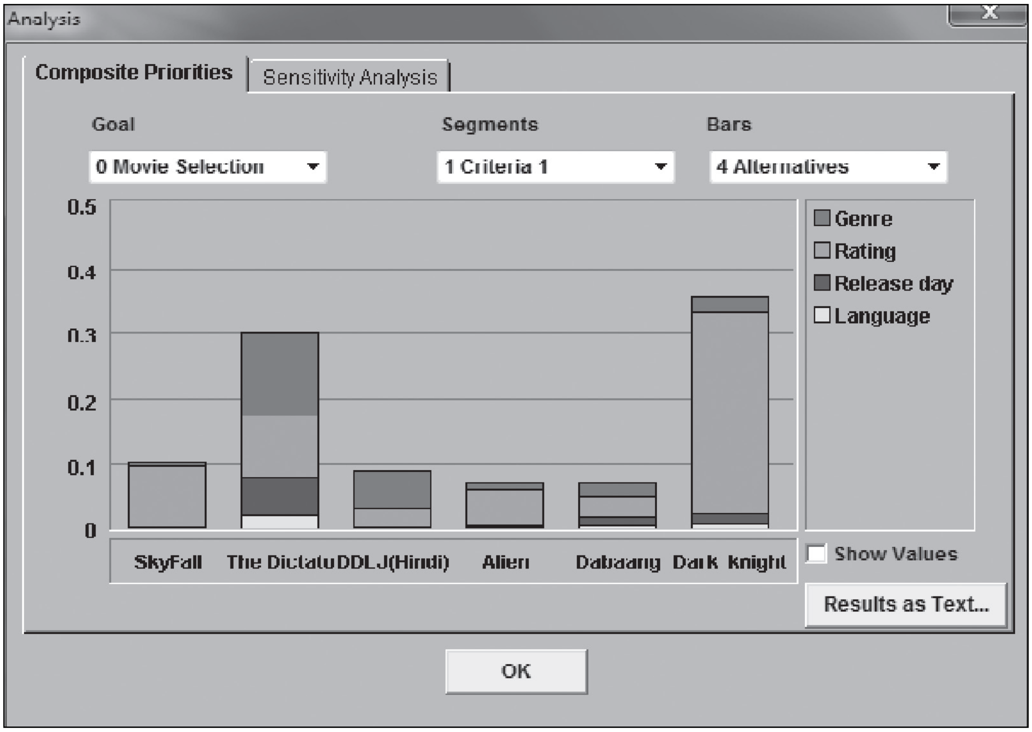


FIGURE 9.16 **Final Composite Scores.**

You can try to build the model in Figure 9.12 yourself and then enter your own pairwise comparisons to make decisions. Do you agree with the choice of the movie?

SECTION 9.9 REVIEW QUESTIONS

1. What is analytic hierarchy process?
2. What steps are needed in applying AHP?
3. What software can be used for AHP?

Chapter Highlights

- Models play a major role in DSS because they are used to describe real decision-making situations. There are several types of models.
- Models can be static (i.e., a single snapshot of a situation) or dynamic (i.e., multiperiod).
- Analysis is conducted under assumed certainty (which is most desirable), risk, or uncertainty (which is least desirable).
- Influence diagrams graphically show the interrelationships of a model. They can be used to enhance the use of spreadsheet technology.
- Spreadsheets have many capabilities, including what-if analysis, goal seeking, programming, database management, optimization, and simulation.
- Decision tables and decision trees can model and solve simple decision-making problems.
- Mathematical programming is an important optimization method.
- LP is the most common mathematical programming method. It attempts to find an optimal allocation of limited resources under organizational constraints.
- The major parts of an LP model are the objective function, the decision variables, and the constraints.
- Multi-criteria decision-making problems are difficult but not impossible to solve.
- The AHP is a leading method for solving multi-criteria decision-making problems.
- What-if and goal seeking are the two most common methods of sensitivity analysis.
- Many DSS development tools include built-in quantitative models (e.g., financial, statistical) or can easily interface with such models.

Key Terms

analytic hierarchy process (AHP)
certainty
decision analysis
decision table
decision tree
decision variable
dynamic models
environmental scanning and analysis
forecasting
goal seeking
influence diagram
intermediate result variable
linear programming (LP)
mathematical (quantitative) model
mathematical programming
multidimensional analysis (modeling)
multiple goals
optimal solution
parameter
result (outcome) variable
risk
risk analysis
sensitivity analysis
static models
uncertainty
uncontrollable variable
what-if analysis

Questions for Discussion

1. What is the relationship between environmental analysis and problem identification?
2. Explain the differences between static and dynamic models. How can one evolve into the other?
3. What is the difference between an optimistic approach and a pessimistic approach to decision making under assumed uncertainty?
4. Explain why solving problems under uncertainty sometimes involves assuming that the problem is to be solved under conditions of risk.
5. Excel is probably the most popular spreadsheet software for PCs. Why? What can we do with this package that makes it so attractive for modeling efforts?
6. Explain how decision trees work. How can a complex problem be solved by using a decision tree?
7. Explain how LP can solve allocation problems.
8. What are the advantages of using a spreadsheet package to create and solve LP models? What are the disadvantages?
9. What are the advantages of using an LP package to create and solve LP models? What are the disadvantages?
10. What is the difference between decision analysis with a single goal and decision analysis with multiple goals (i.e., criteria)? Explain in detail the difficulties that may arise when analyzing multiple goals.
11. Explain how multiple goals can arise in practice.
12. Compare and contrast what-if analysis and goal seeking.
13. Does Simon's four-phase decision-making model fit into most of the modeling methodologies described? Explain.

Exercises

Teradata UNIVERSITY NETWORK (TUN) and Other Hands-on Exercises

1. Explore **teradatauniversitynetwork.com** and determine how models are used in the BI cases and papers.
2. Create the spreadsheet models shown in Figures 9.3 and 9.4.
   a. What is the effect of a change in the interest rate from 8 percent to 10 percent in the spreadsheet model shown in Figure 9.3?
   b. For the original model in Figure 9.3, what interest rate is required to decrease the monthly payments by 20 percent? What change in the loan amount would have the same effect?
   c. In the spreadsheet shown in Figure 9.4, what is the effect of a prepayment of $200 per month? What prepayment would be necessary to pay off the loan in 25 years instead of 30 years?
3. Solve the MBI product-mix problem described in this chapter, using either Excel's Solver or a student version of an LP solver, such as Lindo. Lindo is available from Lindo Systems, Inc., at **lindo.com**; others are also available—search the Web. Examine the solution (output) reports for the answers and sensitivity report. Did you get the same results as reported in this chapter? Try the sensitivity analysis outlined in the chapter; that is, lower the right-hand side of the CC-8 marketing constraint by 1 unit, from 200 to 199. What happens to the solution when you solve this modified problem? Eliminate the CC-8 lower-bound constraint entirely (this can be done easily by either deleting it in Solver or setting the lower limit to zero) and re-solve the problem. What happens? Using the original formulation, try modifying the objective function coefficients and see what happens.
4. Go to **orms-today.com** and access the article "The 'Sound' Science of Scheduling," by L. Gordon and E. Erkut from *OR/MS Today*, Vol. 32, No. 2, April 2005. Describe the overall problem, the DSS developed to solve it, and the benefits.
5. Investigate via a Web search how models and their solutions are used by the U.S. Department of Homeland Security in the "war against terrorism." Also investigate how other governments or government agencies are using models in their missions.

6. Have a group meeting and discuss how you chose a place to live when you relocated to start your college program (or relocated to where you are now). What factors were important for each individual then, and how long ago was it? Have the criteria changed? As a group, identify the five to seven most important criteria used in making the decision. Using the current group members' living arrangements as choices, develop an AHP model that describes this decision-making problem. Do not put your judgments in yet. You should each solve the AHP model independently. Be careful to keep the inconsistency ratio less than 0.1. How many of the group members selected their current home using the software? For those who did, was it a close decision, or was there a clear winner? If some group members did not choose their current homes, what criteria made the result different? (In this decision-making exercise, you should not consider spouses or parents, even those who cook really well, as part of the home.) Did the availability of better choices that meet their needs become known? How consistent were your judgments? Do you think you would really prefer to live in the winning location? Why or why not? Finally, average the results for all group members (by adding the synthesized weights for each choice and dividing by the number of group members). This is one way AHP works. Is there a clear winner? Whose home is it, and why did it win? Were there any close second choices? Turn in your results in a summary report (up to two typed pages), with copies of the individual AHP software runs.
7. Consider a problem with the following goal, main criteria, and subcriteria. Assume relative importance that is most appropriate to your personal experience when ranking the main criteria and subcriteria across various alternatives.
   a. Goal: To select the right university for pursuing your current academic program
   b. Main criteria: Location, weather, cost of the program, reputation, student life, duration
   c. Subcriteria:

   Location → City Proximity, Part-Time Jobs, Full-Time Industry Proximity
   Weather → Hot, Cold, Snow
   Cost of the program → Dollar Amount, Scholarship, Living Expenses
   Reputation → Rank, Public Image
   Student life → Cultural Events, Athletics
   Duration → Length of Course, Flexibility in Changing Duration

   d. Choose various alternatives that you have considered.

   Solve the problem using the AHP methodology and any AHP software. Write a report on how best AHP matched your decision of choice.
8. Consider a problem with the following goal, main criteria, and subcriteria. Assume relative importance that is most appropriate to your personal experience when ranking the main criteria and subcriteria across various alternatives.
   a. Goal: To promote the most deserved candidate to a higher position either at your current work location or your previous work location
   b. Main criteria: Performance, Managerial Skill, Team Orientation
   c. Subcriteria:

   Performance → Subject Knowledge, Quality of work, Responsibility, Accountability
   Managerial skill → Leadership Abilities, Interpersonal Skills, Communication
   Team orientation → Outgoing Behavior, Work Load Distribution, Issue Resolution Ability

   d. Choose various alternatives as the persons that you have considered.

   Solve the problem using the AHP methodology and any AHP software. Write a report on how best AHP matched your decision of choice.
9. This problem was contributed by Dr. Rick Wilson of Oklahoma State University.

   The recent drought has hit farmers hard. Cows are eating candy corn! (**healthyliving.msn.com/blogs/daily-apple-blog-post?post=bdb849dd-ad6c-4868-b3c6-22c6e1817a08#scptmd**)

   You are interested in creating a feed plan for the next week for your cattle using the following 7 nontraditional feeding products: Chocolate Lucky Charms cereal, Butterfinger bars, Milk Duds, Vanilla Ice Cream, Cap'n Crunch cereal, Candy Corn (since the real corn is all dead), and Chips Ahoy cookies.

   Their per pound cost is shown, as is the protein units per pound they contribute, their total digestible nutrients (TDN) they contribute per pound, and the calcium units per pound.

   You estimate that total amount of nontraditional feeding products contribute the following amount of nutrients: at least 20,000 units of protein, at least 4,025 units of TDN, at least 1,000 but no more than 1,200 units of calcium.

	Choc Lucky Charms	Butterfinger	Milk Duds	Vanilla Ice Cream	Cap'n Crunch	Candy Corn	Chips Ahoy
$$/lb	2.15	7	4.25	6.35	5.25	4	6.75
choc	YES	YES	YES	NO	NO	NO	YES
Protein	75	80	45	65	72	26	62
TDN	12	20	18	6	11	8	12
Calcium	3	4	4.5	12	2	1	5

There are some other miscellaneous requirements as well.

- The chocolate in your overall feed plan (in pounds) cannot exceed the amount of non-chocolate poundage. Whether a product is considered chocolate or not is shown in the table (YES = chocolate, NO = not chocolate).
- No one feeding product can make up more than 25 percent of the total pounds needed to create an acceptable feed mix.
- There are two cereals (Choco Lucky Charms and Cap'n Crunch). Combined, they can be no more than 40 percent (in pounds) of the total mix required to meet the mix requirements.

Determine the optimal levels of the 7 products to create your weekly feed plan that minimizes cost. Note that all amounts of products must NOT have fractional values (whole numbered pounds only).

10. This exercise was contributed by Dr. Rick Wilson of Oklahoma State University to illustrate the modeling capabilities of Excel Solver.

You are working with a large set of temporary workers (collection of interns, retirees, etc.) to create a draft plan to staff a nighttime call center (for the near future). You also have a handful of full-time workers who are your "anchors"—but you have already placed them in the schedule and this has led to your staffing requirements. They (full-time workers) are of no concern to you in the model.

These staffing requirements are by day: You need 15, 20, 19, 22, 7, 32, and 35 staff for M, T, W, Th, F, Sat, Sun (respectively).

You have between 8 and 10 of the pool who cannot work on the weekend (Saturday or Sunday).

For these "Weekday Only" folks, there are 3 shifts possible: They will work 4 of the 5 weekdays, one shift will have Tuesday off, one shift will have Wednesday off, and one shift will have Thursday off.

You must have at least eight people total assigned to these "Weekday Only" shifts.

For all other shifts (and you are not constrained by size of employee pool), a person works 4 of the 7 days each week. Workers will work 2 weekdays and both weekend days (a "2/2" shift). All possible "2" day combinations of days are relevant shifts—except any combinations where workers have three consecutive days off; those are not allowed and should not be in the model.

We are going with a very simple model—no costs. The objective of our model is to find the fewest number of workers that meet stated minimum call center daily requirements *and* not have more than 4 extra workers (above min requirements) assigned during any one day.

Also, all shifts ("Weekday Only" or the 2/2 shifts) can have no more than 6 people "allocated" to them.

Create a core model that satisfies these constraints and minimizes the total number of people needed to meet the minimum requirements. If it's an issue, yes, assume that number of people are integers (whole).

11. This exercise was also contributed by Dr. Rick Wilson of Oklahoma State University. The following simple scenario mimics the "Black Book" described in a *Business Week* article (**http://www.businessweek.com/articles/2013-01-31/coke-engineers-its-orange-juice-with-an-algorithm,** accessed February 2013) about Coca-Cola's production of orange juice. Create an appropriate LP model for this scenario.

For the next production period, there are five different batches of raw orange juice that can be blended together to make orange juice products SunnyQ, GlowMorn, and OrenthalJames. In creating the optimal blend of the three products from the five different batches, an LP model should seek to maximize the net of the sales price per gallon of the products less the assessed per gallon cost of the raw juice.

The 5 raw batches of orange juice are described here. Brix is a measure of sweetness, pulp, available stock, and cost—all self-explanatory:

Batch 1—Pineapple Orange A, brix = 16, pulp = 1.2, 250 gallons, $2.01/gallon

Batch 2—Pineapple Orange B, brix = 17, pulp = 0.9, 200 gallons, $2.32/gallon

Batch 3—Mid Sweet, brix = 20, pulp = 0.8, 175 gallons, $3.14/gallon

Batch 4—Valencia, brix = 18, pulp = 2.1, 300 gallons, $2.41/gallon

Batch 5—Temple Orange, brix = 14, pulp = 1.6, 265 gallons, $2.55/gallon

Note that in order to make sure that the raw juice doesn't get too "old" over time, one production requirement is that at least 50 percent of each batch's available stock must be used in blending the three orange juice products (obviously, more than what is available cannot be used).

From a product perspective, there must be at least 100 gallons of SunnyQ blended, and at least 125 gallons each of GlowMorn and OrenthalJames. Likewise, the projected future demand for the products indicates that in this period, there should be a maximum of 400 gallons of SunnyQ, a maximum of 375 gallons of GlowMorn, and a maximum of 300 gallons of OrenthalJames produced. Also, when blending the products from the five batches, an individual batch can provide no more than 40 percent of the total amount of a given product. This is to be enforced individually on each product.

Attributes of the three products include sales price, the maximum average brix of the final mixed product, the minimum average brix of the final mixed product, and the maximum average pulp content. In the three "average" requirements, this implies the weighted average of all juice mixed together for that product must meet that specification.

SunnyQ—Sales = $3.92/gallon, Max Brix = 19, Min Brix = 18.5, Max Pulp = 1.6

GlowMorn—Sales = $4.13/gallon, Max Brix = 17, Min Brix = 16.75, Max Pulp = 1.8

OrenthalJames—Sales = $3.77/gallon, Max Brix = 17.75, Min Brix = 17.55, Max Pulp = 1.1

End-of-Chapter Application Case

Pre-Positioning of Emergency Items for CARE International

Problem

CARE International is a humanitarian organization that provides relief aid to areas that are affected by natural disasters such as earthquakes and hurricanes. The organization has relief programs in over 65 countries worldwide. Just like other humanitarian organizations, CARE International faces challenges in offering the needed help to affected areas in the event of natural disasters. In the event of a disaster, CARE International identifies suppliers that could provide the needed relief items. Arrangements are then made regarding the acquisition of warehouses to transport the items. With respect to the transportation of the items, a third-party company transports the items by air to the affected country from where they are further transported by road to CARE International's warehouse and distribution center. This mode of response to disasters could be slow, not to mention the unreliability of the transportation network used. Hitherto, CARE has preferred purchasing relief items from local suppliers since they are closer to the disaster areas and, also, it helps reinvigorate the local economy after a disaster. However, in the wake of a disaster, there are always issues with availability, price, and quality of needed items.

Specifically, CARE International's challenges are twofold as identified by the authors of the research. First, the organization wanted the ability to gather supplies and relief items from both local and international suppliers in an agile manner so they could better serve people affected by disasters. Second, once the supplies are mobilized, they wanted to be able to effectively distribute them in the most timely and cost-efficient manner to affected regions.

Methodology/Solution

In collaboration with Georgia Institute of Technology, CARE developed a model in which relief items were placed in a pre-positioned network to serve as a complement to the existing mode of supplying relief items to disaster areas. Using a mixed-integer programming (MIP) inventory-location model, a pre-positioning network was designed based on two main factors. The first factor was up-front investment related to initial stocking of inventory and warehouse setup. The second factor was related to the average response time it takes to get relief items to affected regions. Basically, the main concern was to determine a configuration that would allow for the least response time given an up-front investment value. Demand data for the model was based on historical records of previous operations. Supply data was estimated hypothetically since historical data was not present. It was assumed that any supplier would be able to ship relief items within 2 weeks. The model for warehouse establishment was built based on 12 locations CARE considered as low or no-cost, as well as seven relief items necessary for most disaster relief operations. The object function was to reduce the total response time in moving items to affected areas. The capacity constraints employed were the number of warehouses to maintain and the amount of items to keep in them. The MIP model consisted of 470,000 variables and 56,000 constraints. It took the ILOG OPL Studio with CPLEX solver application about 4 hours to produce an optimal solution.

Results/Benefits

The main purpose of the model was to increase the capacity and swiftness to respond to sudden natural disasters like earthquakes, as opposed to other slow-occurring ones like famine. Based on up-front cost, the model is able to provide the best optimized configuration of where to locate a warehouse and how much inventory should be kept. It is able to provide an optimization result based on estimates of frequency, location, and level of potential demand that is generated by the model. Based on this model, CARE has established three warehouses in the warehouse pre-positioning system in Dubai, Panama, and Cambodia. In fact, during the Haiti earthquake crises in 2010, water purification kits were supplied to the victims from the Panama warehouse. In the future, the pre-positioning network is expected to be expanded.

Questions for the End-of-Chapter Application Case

1. What were the main challenges encountered by CARE International before they created their warehouse pre-positioning model?
2. How does the objective function relate to the organization's need to improve relief services to affected areas?
3. Conduct online research and suggest at least three other applications or types of software that could handle the magnitude of variable and constraints CARE International used in their MIP model.
4. Elaborate on some benefits CARE International stands to gain from implementing their pre-positioning model on a large scale in future.

Source: S. Duran, M. A. Gutierrez, and P. Keskinocak, "Pre-Positioning of Emergency Items for CARE International," *Interfaces*, Vol. 41, No. 3, 2011, pp. 223–237.

References

Arsham, H. (2006a). "Modeling and Simulation Resources." **home.ubalt.edu/ntsbarsh/Business-stat/RefSim.htm** (accessed February 2013).

Arsham, H. (2006b). "Decision Science Resources." **home.ubalt.edu/ntsbarsh/Business-stat/Refop.htm** (accessed February 2013).

Bailey, M. J., J. Snapp, S. Yetur, J. S. Stonebraker, S. A. Edwards, A. Davis, and R. Cox. (2011). "Practice Summaries: American Airlines Uses Should-Cost Modeling to Assess the Uncertainty of Bids for Its Full-Truckload Shipment Routes." *Interfaces,* Vol. 41, No. 2, pp. 194–196.

Businessweek.com. "Coke Engineers Its Orange Juice—With an Algorithm." **www.businessweek.com/articles/2013-01-31/coke-engineers-its-orange-juice-with-an-algorithm** (accessed February 2013).

Camm, J. D., T. E. Chorman, F. A. Dill, J. R. Evans, D. J. Sweeney, and G. W. Wegryn. (1997, January/February). "Blending OR/MS, Judgment, and GIS: Restructuring P&G's Supply Chain." *Interfaces*, Vol. 27, No. 1, pp. 128–142.

Carlson, Brian, Yonghong Chen, Mingguo Hong, Roy Jones, Kevin Larson, Xingwang Ma, Peter Nieuwesteeg, et al. (2012). "MISO Unlocks Billions in Savings Through the Application of Operations Research for Energy and Ancillary Services Markets." *Interfaces,* Vol. 42, No. 1 pp. 58–73.

Christiansen, M., K. Fagerholt, G. Hasle, A. Minsaas, and B. Nygreen. (2009, April). "Maritime Transport Optimization: An Ocean of Opportunities." *OR/MS Today,* Vol. 36, No. 2, pp. 26–31.

Clemen, R. T., and T. Reilly. (2000). *Making Hard Decisions with Decision Tools Suite*. Belmont, MA: Duxbury Press.

Duran, S., M. A. Gutierrez, and P. Keskinocak. (2011). "Pre-Positioning of Emergency Items for CARE International." *Interfaces,* Vol. 41, No. 3, pp. 223–237.

Expertchoice.com. "U.S. Department of Housing and Urban Development (HUD) Case Study." **http://expertchoice.com/xres/uploads/resource-center-documents/HUD_casestudy.pdf** (accessed February 2013).

Farasyn, I., K. Perkoz, and W. Van de Velde. (2008, July/August). "Spreadsheet Models for Inventory Target Setting at Procter and Gamble." *Interfaces,* Vol. 38, No. 4, pp. 241–250.

Furman, K. C., J. H. Song, G. R. Kocis, M. K. McDonald, and P. H. Warrick. (2011). "Feedstock Routing in the ExxonMobil Downstream Sector." *Interfaces,* Vol. 41, No. 2, pp. 49–163.

Goodwin, P., and G. Wright. (2000). *Decision Analysis for Management Judgment*, 2nd ed. New York: Wiley.

Healthyliving.msn.com. "Cows Eating 'Candy' Corn." **http://healthyliving.msn.com/blogs/daily-apple-blog-post?post=bdb849dd-ad6c-4868-b3c6-22c6e1817a08#scptmd** (accessed February 2013).

Hillier, F. S., and G. J. Lieberman. (2005). *Introduction to Operations Research*, 8th ed. New York: McGraw-Hill.

Hurley, W. J., and M. Balez. (2008, July/August). "A Spreadsheet Implementation of an Ammunition Requirements Planning Model for the Canadian Army." *Interfaces,* Vol. 38, No. 4, pp. 271–280.

Kearns, G. S. (2004, January–March). "A Multi-Objective, Multi-Criteria Approach for Evaluating IT Investments: Results from Two Case Studies." *Information Resources Management Journal*, Vol. 17, No. 1, pp. 37–62.

Kelly, A. (2002). *Decision Making Using Game Theory: An Introduction for Managers*. Cambridge, UK: Cambridge University Press.

Koksalan, M., and S. Zionts (eds.). (2001). *Multiple Criteria Decision Making in the New Millennium*. Berlin: Springer-Verlag.

Kontoghiorghes, E. J., B. Rustem, and S. Siokos. (2002). *Computational Methods in Decision Making, Economics, and Finance*. Boston: Kluwer.

Lejeune, M. A., and N. Yakova. (2008, May/June). "Showcase Scheduling at Fred Astaire East Side Dance Studio." *Interfaces,* Vol. 38, No. 3, pp. 176–186.

Metters, M., C. Queenan, M. Ferguson, L. Harrison, J. Higbie, S. Ward, B. Barfield, T. Farley, H. A. Kuyumcu, and A. Duggasani. (2008, May/June). "The 'Killer Application' of Revenue Management: Harrah's Cherokee Casino & Hotel." *Interfaces*, Vol. 38, No. 3, pp. 161–175.

Ovchinnikov, A., and J. Milner. (2008, July/August). "Spreadsheet Model Helps to Assign Medical Residents at the University of Vermont's College of Medicine." *Interfaces,* Vol. 38, No. 4, pp. 311–323.

ProModel. (2006, March). "Fiat Case." **promodel.com** (accessed February 2013).

ProModel. (2009). "Throughput, Cycle Time, and Bottleneck Analysis with ProModel Simulation Solutions for Manufacturing." **promodel.com** (accessed February 2013).

ProModel. (2013). "Pillowtex Case." **promodel.com** (accessed February 2013).

Saaty, T. L. (1995). *Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World,* Rev. ed. Pittsburgh, PA: RWS Publishers.

Saaty, T. L. (1996). *Decision Making for Leaders,* Vol. II. Pittsburgh, PA: RWS Publishers.

Saaty, T. L. (1999). *The Brain: Unraveling the Mystery of How It Works (The Neural Network Process)*. Pittsburgh, PA: RWS Publishers.

Taha, H. (2006). *Operations Research: An Introduction*, 8th ed. Upper Saddle River, NJ: Prentice Hall.

TreeAge Software, Inc. (2009). "Dr. Victor Grann Uses Decision Analysis to Weigh Treatment Options for Patients at High Risk of Developing Cancer." (accessed February 2013).

CHAPTER

10

Modeling and Analysis: Heuristic Search Methods and Simulation

LEARNING OBJECTIVES

- Explain the basic concepts of simulation and heuristics, and when to use them
- Understand how search methods are used to solve some decision support models
- Know the concepts behind and applications of genetic algorithms
- Explain the differences among algorithms, blind search, and heuristics
- Understand the concepts and applications of different types of simulation
- Explain what is meant by system dynamics, agent-based modeling, Monte Carlo, and discrete event simulation
- Describe the key issues of model management

In this chapter, we continue to explore some additional concepts related to the model base, one of the major components of decision support systems (DSS). As pointed out in the last chapter, we present this material with a note of caution: The purpose of this chapter is not necessarily for you to *master the topics* of modeling and analysis. Rather, the material is geared toward *gaining familiarity* with the important concepts as they relate to DSS and their use in decision making. We discuss the structure and application of some successful time-proven models and methodologies: search methods, heuristic programming, and simulation. Genetic algorithms mimic the natural process of evolution to help find solutions to complex problems. The concepts and motivating applications of these advanced techniques are described in this chapter, which is organized into the following sections:

10.1 OPENING VIGNETTE: System Dynamics Allows Fluor Corporation to Better Plan for Project and Change Management

INTRODUCTION

Fluor is an engineering and construction company with over 36,000 employers spread over several countries worldwide. The company's net income in 2009 amounted to about $680 million based on total revenue of $22 billion. As part of its operations, Fluor manages varying sizes of projects that are subject to scope changes, design changes, and schedule changes.

PRESENTATION OF PROBLEM

Fluor estimated that changes accounted for about 20 to 30 percent of revenue. Most changes were due to secondary impacts like ripple effects, disruptions, and productivity loss. Previously, the changes were collated and reported at a later period and the burden of cost allocated to the stakeholder responsible. In certain instances when late surprises about cost and project schedule are attributed to clients, it causes friction between clients and Fluor, which eventually affect future business dealings. Sometimes, cost impacts occur in such a time and fashion when it is difficult to take preventive measures. The company determined that to improve on its efficiency, reduce legal ramification with clients, and keep them happy it had to review its method of handling changes to projects. One challenge the company faced was the fact that changes stayed extremely remote from the situation, which warranted the change. In such a case, it is difficult to determine the cause of a change, and it affects subsequent measures to handle related change issues.

METHODOLOGY/SOLUTION

For sure, Fluor knew that one way of combating the issue was to foresee and avoid the events that might lead to changes. However, that alone would not be enough to solve the problem. The company needed to understand the dynamics of the different situations that could warrant changes to project plans. Systems dynamics was used as a base method in a three-part analytical solution for understanding the dynamics between different factors that could cause changes to be made. System dynamics is a methodology and simulation-modeling technique for analyzing complex systems using principles of cause and effect, feedback loops, and time-delayed and nonlinear effects. Building tools for rapidly tailoring a solution to different situations form the next part of the three-part analytical solution. In this part, industry standards and company references are embedded. The project plan is also embedded as an input. The model is then converged to simulate the correct amounts and timing of other factors like staffing, project progress, productivity, and effects on productivity. The last part of the analytical solution was to deploy the project models to nonmodelers. Basically, the system takes inputs that are specific to a particular project being worked and its environment, such as the labor market. Some other input parameters, transformed into numerical data, are related to progress curves, expenses, and labor laws and constraints.

The resultant system provides reports on project impacts as well as helps perform cause–effect diagnostics.

RESULTS/BENEFITS

With this system, customers are able to perform "what-if" analysis even before a project is started so the project performance can be gauged. Through diagnostics, the system also helps explain why certain effects are realized based on impact to the project plan. Since its development, Fluor has recorded over 100 extensive uses of their system dynamics model and project simulation system. As an example, the model was used to analyze and save $10 million in the future impact of changes to a mining project. Also, based on the what-if capability of Fluor's model, a company saved $10 million when the project team used the model to redesign the process of reviewing changes so that the speed of the company's definition and approval procedures was increased.

QUESTIONS FOR THE OPENING VIGNETTE

1. Explain the use of system dynamics as a simulation tool for solving complex problems.
2. In what ways was it applied in Fluor Corporation to solve complex problems?
3. How does a what-if analysis help a decision maker to save on cost?
4. In your own words, explain the factors that might have triggered the use of system dynamics to solve change management problems in Fluor Corporation.
5. Pick a geographic region and business domain and list some corresponding relevant factors that would be used as inputs in building such a system.

WHAT WE CAN LEARN FROM THIS VIGNETTE

Changes to project plans and timelines are a major contributing factor to upward increase in cost from initial amount budgeted for projects. In this case, Fluor relied on system dynamics to understand what, why, when, and how changes occurred to project plans. The models that the system dynamics model produced helped them correctly quantify the cost of projects even before they started. The vignette demonstrates that system dynamics is still a credible and robust methodology in understanding business processes and creating "what-if" analyses of the impact of both expected and unexpected changes in project plans.

Source: E. Godlewski, G. Lee, and K. Cooper, "System Dynamics Transforms Fluor Project and Change Management," *Interfaces*, Vol. 42, No. 1, 2012, pp. 17–32.

10.2 PROBLEM-SOLVING SEARCH METHODS

We next turn to several well-known search methods used in the choice phase of problem solving. These include analytical techniques, algorithms, blind searching, and heuristic searching.

The choice phase of problem solving involves a search for an appropriate course of action (among those identified during the design phase) that can solve the problem. Several major search approaches are possible, depending on the criteria (or criterion) of choice and the type of modeling approach used. These search approaches are shown in Figure 10.1. For normative models, such as mathematical programming-based ones, either an analytical approach is used or a complete, exhaustive enumeration (comparing the outcomes of all the alternatives) is applied. For descriptive models, a comparison of a limited number of alternatives is used, either blindly or by employing heuristics. Usually the results guide the decision maker's search.

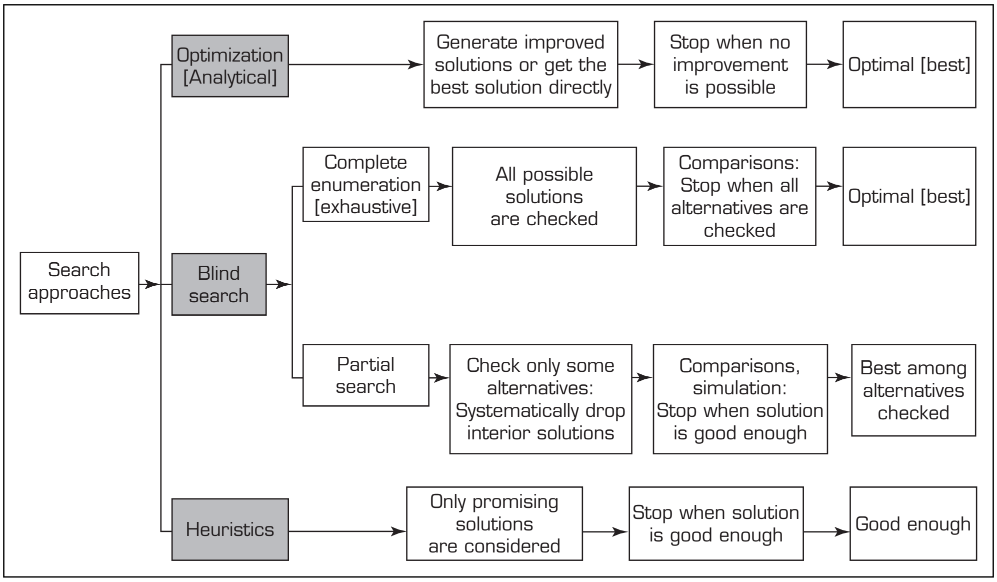


FIGURE 10.1 Formal Search Approaches.

Analytical Techniques

Analytical techniques use mathematical formulas to derive an optimal solution directly or to predict a certain result. Analytical techniques are used mainly for solving structured problems, usually of a tactical or operational nature, in areas such as resource allocation or inventory management. Blind or heuristic search approaches generally are employed to solve more complex problems.

Algorithms

Analytical techniques may use algorithms to increase the efficiency of the search. An algorithm is a step-by-step search process for obtaining an optimal solution (see Figure 10.2). (Note: There may be more than one optimum, so we say *an* optimal solution rather than

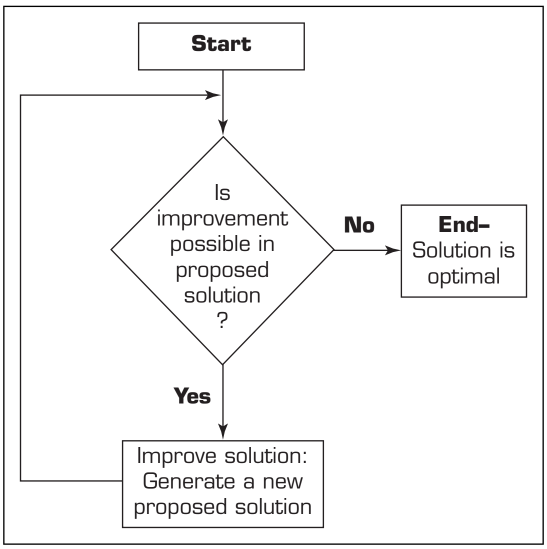


FIGURE 10.2 The Process of Using an Algorithm.

the optimal solution.) Solutions are generated and tested for possible improvements. An improvement is made whenever possible, and the new solution is subjected to an improvement test, based on the principle of choice (i.e., objective value found). The process continues until no further improvement is possible. Most mathematical programming problems are solved by using efficient algorithms. Web search engines use various algorithms to speed up searches and produce accurate results.

Blind Searching

In conducting a search, a description of a desired solution may be given. This is called a *goal*. A set of possible steps leading from initial conditions to the goal is called the *search steps*. Problem solving is done by searching through the possible solutions. The first of these search methods is blind searching. The second is heuristic searching.

Blind search techniques are arbitrary search approaches that are not guided. There are two types of blind searches: a *complete enumeration*, for which all the alternatives are considered and therefore an optimal solution is discovered; and an *incomplete*, or partial, search, which continues until a good-enough solution is found. The latter is a form of suboptimization.

There are practical limits on the amount of time and computer storage available for blind searches. In principle, blind search methods can eventually find an optimal solution in most search situations, and, in some situations, the scope of the search can be limited; however, this method is not practical for solving very large problems because too many solutions must be examined before an optimal solution is found.

Heuristic Searching

For many applications, it is possible to find rules to guide the search process and reduce the number of necessary computations through heuristics. **Heuristics** are the informal, judgmental knowledge of an application area that constitute the rules of good judgment in the field. Through domain knowledge, they guide the problem-solving process. **Heuristic programming** is the process of using heuristics in problem solving. This is done via heuristic search methods, which often operate as algorithms but limit the solutions examined either by limiting the search space or stopping the method early. Usually, rules that have either demonstrated their success in practice or are theoretically solid are applied in heuristic searching. In Application Case 10.1, we provide an example of a DSS in which the models are solved using heuristic searching.

Application Case 10.1

Chilean Government Uses Heuristics to Make Decisions on School Lunch Providers

The Junta Nacional de Auxilio Escolar y Becas (JUNAEB), an agency of the Chilean government, promotes integration and retention of socially vulnerable children in the country's school system. JUNAEB's school meal program provides meals for approximately 10,000 schools. Decisions on meal providers are made through an annual tender using a combinatorial auction, where food industry firms bid on supply contracts, based on a series of disjoint, compact geographical areas called territorial units (TUs). These territorial units consist of districts spanning the country.

When the Chilean economy suffered a downturn, many competing meal service providers ceased their operations. Thus, the number of suppliers participating in the combinatorial auction was reduced.

(*Continued*)

Application Case 10.1 (Continued)

The entire school meal policy was called into question. The central problem was in defining TUs. JUNAEB divided Chile's 13 official regions, consisting of several districts, into 136 TUs based on geographical criteria, which attempted to equalize the number of meals to be served in each TU. This process led to severe disparities as the districts in regions requiring large numbers of meals were assigned to a single TU; the remaining districts were combined into TUs requiring similar quantities of numbers of meals but for a possibly larger geographical area and number of schools in each district. Sometimes, a firm that ended up bagging an attractive TU was paired with another unattractive TU and, hence, was unable to fulfill its contract.

With realization of the need to determine new configurations of territorial units, homogenization of characteristics across territorial units was achieved based on a score that considered each constituent district's four characteristics: number of meals, number of schools, geographic area, and accessibility. A series of operating research methodologies was applied toward reaching the goal of homogenization of TUs.

The analytic hierarchy process was first applied to determine the relative weight of each of four characteristics for each TU in each region, and then total scores for each TU were calculated. Then a local search heuristic was employed to find a set of homogenously attractive TUs within each region. The TU's attractiveness was calculated using the values derived from the AHP process for each characteristic, and the TU's criterion weights were calculated for the local search heuristic's assessment in each region. The degree of homogeneity was measured as the standard deviation, which measures the dispersion of a TU's attractiveness level by quantifying the divergence of each TU in a region from the regional average. The heuristic attempts to minimize this measure by exchanging the combination of districts in each TU with the districts in other TUs existing in the same region. The initial set of TUs in the region are defined based on expert opinions. Then heuristics proceeds by searching the local minima and approaching the best solution by transferring districts from one TU to another until a local minima is reached where the combination of districts across all TUs separate the TUs with lowest standard deviation.

The new configuration limited the minimum and maximum number of meals for each TU between 15,000 and 40,000, and each of the 13 geographical regions was assigned TUs accordingly. The districts belonging to the TUs served as the basic units in homogenizing the TUs. Each district in the TU that served more than 10,000 meals was again divided into an equal number of subdistricts.

An integer linear programming (ILP) model was applied to the results generated by a cluster enumeration algorithm, which formed TUs as clusters created by grouping contiguous districts and subdistricts into a TU. For each region, the ILP model selected a set of clusters constituting a partition of region that minimizes the difference between the most and least attractive clusters based on the TU scores that were calculated using the weights of criteria used in a cluster.

Finally, a combination of ILP and heuristics was applied in which the results obtained from ILP were used as the initial solution on which local search heuristics were applied. This further aimed to reduce the standard deviation of attractiveness scores of TUs.

Existing data about the TUs from 2007 was used as the baseline, and the results from each of three methodologies showed a significant level of homogeneity that did not exist in the 2007 data.

Questions for Discussion

1. What were the main challenges faced by JUNAEB?
2. What operation research methodologies were employed in achieving homogeneity across territorial units?
3. What other approaches could you use in this case study?

What We Can Learn from This Application Case

Heuristic methods can work best in providing solutions for problems that involve exhaustive, repetitive processes to arrive at a solution. The application case also shows that combinations of operations research methodologies can play a vital role in solving a particular problem.

Source: D. M. G. Alfredo, E. N. R. David, M. Cristian, and Z. V. G. Andres, "Quantitative Methods for a New Configuration of Territorial Units in a Chilean Government Agency Tender Process," *Interfaces,* 2011.

SECTION 10.2 REVIEW QUESTIONS

1. What is a search approach?
2. List the different problem-solving search methods.
3. What are the practical limits to blind searching?
4. How are algorithms and heuristic search methods similar? How are they different?

10.3 GENETIC ALGORITHMS AND DEVELOPING GA APPLICATIONS

Genetic algorithms (GA) are a part of global search techniques used to find approximate solutions to optimization-type problems that are too complex to be solved with traditional optimization methods (which are guaranteed to produce the best solution to a specific problem). Genetic algorithms have been successfully applied to a wide range of highly complex real-world problems, including vehicle routing (Baker and Syechew, 2003), bankruptcy prediction (Shin and Lee, 2002), and Web searching (Nick and Themis, 2001).

Genetic algorithms are a part of the machine-learning family of methods under artificial intelligence. Because they cannot guarantee the truly optimal solution, genetic algorithms are considered to be heuristic methods. Genetic algorithms are sets of computational procedures that conceptually follow the steps of the biological process of evolution. That is, better and better solutions evolve from the previous generation of solutions until an optimal or near-optimal solution is obtained.

Genetic algorithms (also known as **evolutionary algorithms**) demonstrate self-organization and adaptation in much the same way that biological organisms do by following the chief rule of evolution, *survival of the fittest*. The method improves the solutions by producing offspring (i.e., a new collection of feasible solutions) using the best solutions of the current generation as "parents." The generation of offspring is achieved by a process modeled after biological reproduction whereby mutation and crossover operators are used to manipulate genes in constructing newer and "better" chromosomes. Notice that a simple analogy between genes and decision variables and between chromosomes and potential solutions underlies the genetic algorithm terminology.

Example: The Vector Game

To illustrate how genetic algorithms work, we describe the classical Vector game (see Walbridge, 1989). This game is similar to MasterMind. As your opponent gives you clues about how good your guess is (i.e., the outcome of the fitness function), you create a new solution, using the knowledge gained from the recently proposed solutions and their quality.

Description of The Vector Game Vector is played against an opponent who secretly writes down a string of six digits (in a genetic algorithm, this string consists of a *chromosome*). Each digit is a decision variable that can take the value of either 0 or 1. For example, say that the secret number that you are to figure out is 001010. You must try to guess this number as quickly as possible (with the least number of trials). You present a sequence of digits (a guess) to your opponent, and he or she tells you how many of the digits (but not which ones) you guessed are correct (i.e., the fitness function or quality of your guess). For example, the guess 110101 has no correct digits (i.e., the score = 0). The guess 111101 has only one correct digit (the third one, and hence the score = 1).

Default Strategy: Random Trial and Error There are 64 possible six-digit strings of binary numbers. If you pick numbers at random, you will need, on average, 32 guesses to obtain the right answer. Can you do it faster? Yes, if you can interpret the feedback provided to you by your opponent (a measure of the goodness or fitness of your guess). This is how a genetic algorithm works.

Improved Strategy: Use of Genetic Algorithms The following are the steps in solving the Vector game with genetic algorithms:

1. Present to your opponent four strings, selected at random. (Select four arbitrarily. Through experimentation, you may find that five or six would be better.) Assume that you have selected these four:

   (A) 110100; score = 1 (i.e., one digit guessed correctly)
   (B) 111101; score = 1
   (C) 011011; score = 4
   (D) 101100; score = 3

2. Because none of the strings is entirely correct, continue.
3. Delete (A) and (B) because of their low scores. Call (C) and (D) parents.
4. "Mate" the parents by splitting each number as shown here between the second and third digits (the position of the split is randomly selected):

   (C) 01:1011
   (D) 10:1100

   Now combine the first two digits of (C) with the last four of (D) (this is called crossover). The result is (E), the first offspring:

   (E) 011100; score = 3
   Similarly, combine the first two digits of (D) with the last four of (C). The result is (F), the second offspring:

   (F) 101011; score = 4
   It looks as though the offspring are not doing much better than the parents.
5. Now copy the original (C) and (D).
6. Mate and crossover the new parents, but use a different split. Now you have two new offspring, (G) and (H):

   (C) 0110:11
   (D) 1011:00
   (G) 0110:00; score = 4
   (H) 1011:11; score = 3

   Next, repeat step 2: Select the best "couple" from all the previous solutions to reproduce. You have several options, such as (G) and (C). Select (G) and (F). Now duplicate and crossover. Here are the results:

   (F) 1:01011
   (G) 0:11000
   (I) 111000; score = 3
   (J) 001011; score = 5

   You can also generate more offspring:

   (F) 101:011
   (G) 011:000
   (K) 101000; score = 4
   (L) 011011; score = 4

   Now repeat the processes with (J) and (K) as parents, and duplicate the crossover:

   (J) 00101:1
   (K) 10100:0
   (M) 001010; score = 6

   That's it! You have reached the solution after 13 guesses. Not bad compared to the expected average of 32 for a random-guess strategy.

Terminology of Genetic Algorithms

A genetic algorithm is an iterative procedure that represents its candidate solutions as strings of genes called **chromosomes** and measures their viability with a fitness function. The fitness function is a measure of the objective to be obtained (i.e., maximum or minimum). As in biological systems, candidate solutions combine to produce offspring in each algorithmic iteration, called a *generation*. The offspring themselves can become candidate solutions. From the generation of parents and children, a set of the fittest survive to become parents that produce offspring in the next generation. Offspring are produced using a specific genetic reproduction process that involves the application of crossover and mutation operators. Along with the offspring, some of the best solutions are also migrated to the next generation (a concept called **elitism**) in order to preserve the best solution achieved up until the current iteration. Following are brief definitions of these key terms:

- ***Reproduction.*** Through **reproduction**, genetic algorithms produce new generations of potentially improved solutions by selecting parents with higher fitness ratings or by giving such parents a greater probability of being selected to contribute to the reproduction process.
- ***Crossover.*** Many genetic algorithms use a string of binary symbols (each corresponding to a decision variable) to represent chromosomes (potential solutions), as was the case in the Vector game described earlier. **Crossover** means choosing a random position in the string (e.g., after the first two digits) and exchanging the segments either to the right or the left of that point with those of another string's segments (generated using the same splitting schema) to produce two new offspring.
- ***Mutation.*** This genetic operator was not shown in the Vector game example. **Mutation** is an arbitrary (and minimal) change in the representation of a chromosome. It is often used to prevent the algorithm from getting stuck in a local optimum. The procedure randomly selects a chromosome (giving more probability to the ones with better fitness value) and randomly identifies a gene in the chromosome and inverses its value (from 0 to 1 or from 1 to 0), thus generating one new chromosome for the next generation. The occurrence of mutation is usually set to a very low probability (0.1 percent).
- ***Elitism.*** An important aspect in genetic algorithms is to preserve a few of the best solutions to evolve through the generations. That way, you are guaranteed to end up with the best possible solution for the current application of the algorithm. In practice, a few of the best solutions are migrated to the next generation.

How Do Genetic Algorithms Work?

Figure 10.3 is a flow diagram of a typical genetic algorithm process. The problem to be solved must be described and represented in a manner amenable to a genetic algorithm. Typically, this means that a string of 1s and 0s (or other more recently proposed complex representations) are used to represent the decision variables, the collection of which represents a potential solution to the problem. Next, the decision variables are mathematically and/or symbolically pooled into a *fitness function* (or *objective function*). The fitness function can be one of two types: maximization (something that is more is better, such as profit) or minimization (something that is less is better, such as cost). Along with the fitness function, all of the constraints on decision variables that collectively

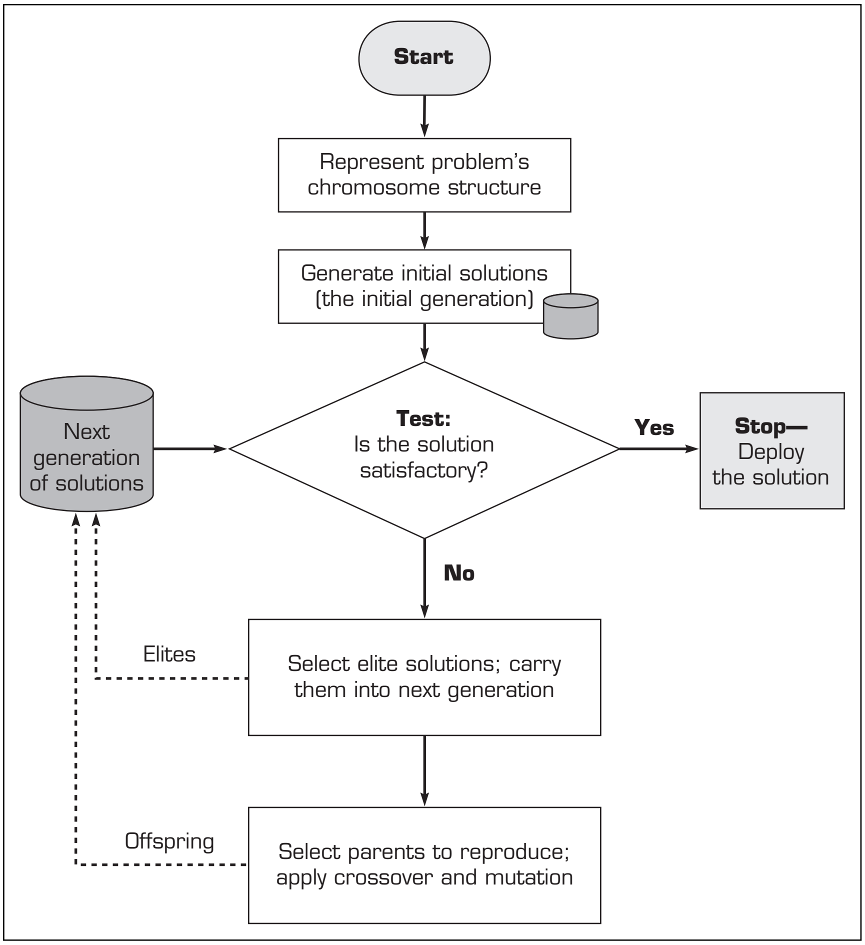


FIGURE 10.3 A Flow Diagram of a Typical Genetic Algorithm Process.

dictate whether a solution is a feasible one should be demonstrated. Remember that only feasible solutions can be a part of the solution population. Infeasible ones are filtered out before finalizing a generation of solutions in the iterations process. Once the representation is complete, an initial set of solutions is generated (i.e., the initial population). All infeasible solutions are eliminated, and fitness functions are computed for the feasible ones. The solutions are rank-ordered based on their fitness values; those with better fitness values are given more probability (proportional to their relative fitness value) in the random selection process.

A few of the best solutions are migrated to the next generation. Using a random process, several sets of parents are identified to take part in the generation of offspring. Using the randomly selected parents and the genetic operators (i.e., crossover and mutation), offspring are generated. The number of potential solutions to generate is determined by the population size, which is an arbitrary parameter set prior to the evolution of solutions. Once the next generation is constructed, the solutions go through the evaluation and generation of new populations for a number of iterations. This iterative process continues until a good-enough solution is obtained (an optimum is not guaranteed), no improvement occurs over several generations, or the time/iteration limit is reached.

As mentioned, a few parameters must be set prior to the execution of the genetic algorithm. Their values are dependent on the problem being solved and are usually determined through trial and error:

- Number of initial solutions to generate (i.e., the initial population)
- Number of offspring to generate (i.e., the population size)
- Number of parents to keep for the next generation (i.e., elitism)
- Mutation probability (usually a very low number, such as 0.1 percent)
- Probability distribution of crossover point occurrence (generally equally weighted)
- Stopping criteria (time/iteration based or improvement based)
- The maximum number of iterations (if the stopping criteria are time/iteration based)

Sometimes these parameters are set and frozen beforehand, or they can be varied systematically while the algorithm is running for better performance.

Limitations of Genetic Algorithms

According to Grupe and Jooste (2004), the following are among the most important limitations of genetic algorithms:

- Not all problems can be framed in the mathematical manner that genetic algorithms demand.
- Development of a genetic algorithm and interpretation of the results require an expert who has both the programming and statistical/mathematical skills demanded by the genetic algorithm technology in use.
- It is known that in a few situations the "genes" from a few comparatively highly fit (but not optimal) individuals may come to dominate the population, causing it to converge on a local maximum. When the population has converged, the ability of the genetic algorithm to continue to search for better solutions is effectively eliminated.
- Most genetic algorithms rely on random-number generators that produce different results each time the model runs. Although there is likely to be a high degree of consistency among the runs, they may vary.
- Locating good variables that work for a particular problem is difficult. Obtaining the data to populate the variables is equally demanding.
- Selecting methods by which to evolve the system requires thought and evaluation. If the range of possible solutions is small, a genetic algorithm will converge too quickly on a solution. When evolution proceeds too quickly, thereby altering good solutions too quickly, the results may miss the optimum solution.

Genetic Algorithm Applications

Genetic algorithms are a type of machine learning for representing and solving complex problems. They provide a set of efficient, domain-independent search heuristics for a broad spectrum of applications, including the following:

- Dynamic process control
- Induction of optimization of rules
- Discovery of new connectivity topologies (e.g., neural computing connections, neural network design)
- Simulation of biological models of behavior and evolution
- Complex design of engineering structures
- Pattern recognition
- Scheduling
- Transportation and routing

- Layout and circuit design
- Telecommunication
- Graph-based problems

A genetic algorithm interprets information that enables it to reject inferior solutions and accumulate good ones, and thus it learns about its universe. Genetic algorithms are also suitable for parallel processing.

Because the kernels of genetic algorithms are pretty simple, it is not difficult to write computer codes to implement them. For better performance, software packages are available.

Several genetic algorithm codes are available for fee or for free (try searching the Web for research and commercial sites). In addition, a number of commercial packages offer online demos. Representative commercial packages include Microsoft Solver and XpertRule GenAsys, an ES shell with an embedded genetic algorithm (see **xpertrule.com**). Evolver (from Palisade Corp., **palisade.com**) is an optimization add-in for Excel. It uses a genetic algorithm to solve complex optimization problems in finance, scheduling, manufacturing, and so on.

SECTION 10.3 REVIEW QUESTIONS

1. Define *genetic algorithm.*
2. Describe the evolution process in genetic algorithms. How is it similar to biological evolution?
3. Describe the major genetic algorithm operators.
4. List major areas of genetic algorithm application.
5. Describe in detail three genetic algorithm applications.
6. Describe the capability of Evolver as an optimization tool.

We now turn our attention to simulation, a class of modeling method that has enjoyed significant actual use in decision making.

10.4 SIMULATION

Simulation is the appearance of reality. In MSS, simulation is a technique for conducting experiments (e.g., what-if analyses) with a computer on a model of a management system.

Typically, real decision-making situations involve some randomness. Because DSS deals with semistructured or unstructured situations, reality is complex, which may not be easily represented by optimization or other models but can often be handled by simulation. Simulation is one of the most commonly used DSS methods. See Application Cases 10.2 and 10.3 for examples. Application Case 10.3 illustrates the value of simulation in a setting where sufficient time is not available to perform clinical trials.

Application Case 10.2

Improving Maintenance Decision Making in the Finnish Air Force Through Simulation

The Finnish Air Force wanted to gain efficiency in its maintenance system in order to keep as many aircraft as possible safely available at all times for training, missions, and other tasks, as needed. A discrete event simulation program similar to those used in manufacturing was developed to accommodate workforce issues, task times, material handling delays, and the likelihood of equipment failure.

The developers had to consider aircraft availability, resource requirements for international operations, and the periodic maintenance program. The information for normal conditions and conflict conditions was input into the simulation program because the maintenance schedule could be altered from one situation to another.

The developers had to estimate some information due to confidentiality, especially with regards to conflict scenarios (no data of battle-damage probabilities was available). They used several methods to acquire and secure data, such as asking experts in aircraft maintenance fields at different levels for their opinions and designing a model that allowed the confidential data to be input into the system. Also, the simulations were compared to actual performance data to make sure the simulated results were accurate.

The maintenance program was broken into three levels:

1. The organizational level, in which the fighter squadron takes care of preflight checks, turnaround checks (which occur when an aircraft returns), and other minor repairs at the main command airbase in normal conditions
2. The intermediate level, in which more complicated periodic maintenance and failure repairs are taken care of at the air command repair shop at the main airbase in normal conditions
3. The depot level, in which all major periodic maintenance is taken care of and is located away from the main airbase

During conflict conditions, the system is decentralized from the main airbase. The maintenance levels just described may continue to do the exact same repairs, or periodic maintenance may be eliminated. Additionally, depending on need, supplies, and capabilities, any of these levels may take care of any maintenance and repairs needed at any time during conflict conditions.

The simulation model was implemented using Arena software based on the SIMAN language and involved using a graphical user interface (GUI) that was executed using Visual Basic for Applications (VBA). The input data included simulation parameters and the initial system state: characteristics of the air commands, maintenance needs, and flight operations; accumulated flight hours; and the location of each aircraft. Excel spreadsheets were used for data input and output. Additionally, parameters of some of the input data were estimated from statistical data or based on information from subject matter experts. These included probabilities for time between failures, damage sustained during a single-flight mission, the duration of each type of periodic maintenance, failure repair, damage repair, the times between flight missions, and the duration of a mission. This simulation model was so successful that the Finnish Army, in collaboration with the Finnish Air Force, has now devised a simulation model for the maintenance for some of its new transport helicopters.

Source: Based on V. Mattila, K. Virtanen, and T. Raivio, "Improving Maintenance Decision Making in the Finnish Air Force Through Simulation," *Interfaces*, Vol. 38, No. 3, May/June 2008, pp. 187–201.

Application Case 10.3

Simulating Effects of Hepatitis B Interventions

Although the United States has made significant investments in healthcare, some problems seem to defy solution. For example, a sizable proportion of the Asian population in the United States is more prone than others to the Hepatitis B viral disease. In addition to the social problems associated with the disease (like isolation), one out of every four chronically infected individuals stands the risk of suffering from liver cancer or cirrhosis if the disease is not treated effectively. Managing this disease could be very costly. There are a number of control measures, including screening, vaccination, and treatment procedures. The government is reluctant to spend money on any method of control if it is not cost-effective and there is no proof of increased health for people afflicted with the disease. Even though not all the control measures are optimal for all situations, the best method or combination of methods for combatting the disease are not yet known.

(*Continued*)

Application Case 10.3 (Continued)

Methodology/Solution

A multidisciplinary team consisting of those with medical, management science, and engineering backgrounds developed a mathematical model using operations research (OR) methods that determined the right combination of control measures to be used to combat Hepatitis B in the Asian and Pacific Island populations. Normally, clinical trials are used in the medical field to determine the best course of action in disease treatment and prevention. Complicating this situation is the unusually long period of time it takes Hepatitis B to progress. Because of the high cost that would accompany clinical trials in this situation, operations research models and methods were used. A combination of Markov and decision models offered a more cost-effective way for determining what combination of control measures to use at any point in time. The decision model helps measure the economic and health benefits of various possibilities of screening, treatment, and vaccination. The Markov model was used to model the progression of Hepatitis B. The new model was created based on past literature and expertise from one of the researchers and draws from actual current infection and treatment data. Policymakers built the new model using Microsoft Excel because it is user friendly.

Results/Benefits

The resultant model was analyzed vis-à-vis existing control programs in both the United States and China. In the United States four strategies were developed and compared to the existing strategy. The four strategies are:

a. All individuals are vaccinated.
b. Individuals are first screened to determine whether they have a chronic infection. If yes, then they are treated.
c. Individuals are first screened to determine whether they have a chronic infection. If they have the infection, they are treated. In addition, close associates of those infected are also screened and vaccinated, if necessary.
d. Individuals are first screened to determine whether they have a chronic infection or need vaccination. If they are infected, they are treated. If they need vaccination, they are vaccinated.

Results of the simulations indicated that performing blood tests to determine chronic infection and vaccinating associates of infected people are cost-effective.

In China, the model helped design a catch-up vaccination policy for children and adolescents. This catch-up policy was compared with current coverage levels of Hepatitis B vaccination. It was concluded that when individuals under the age of 19 years are vaccinated, the health outcomes are improved in the long run. In fact, this policy was more financially cost-effective than the current disease control policy in place at the time of the evaluation.

Questions for Discussion

1. Explain the advantage of operations research methods such as simulation over clinical trial methods in determining the best control measure for Hepatitis B.
2. In what ways do the decision and Markov models provide cost-effective ways of combating the disease?
3. Discuss how multidisciplinary background is an asset in finding a solution for the problem described in the case.
4. Besides healthcare, in what other domain could such a modeling approach help reduce cost?

Source: D. W. Hutton, M. L. Brandeau, and S. K. So, "Doing Good with Good OR: Supporting Cost-Effective Hepatitis B Interventions," *Interfaces,* Vol. 41, No. 3, 2011, pp. 289–300.

Major Characteristics of Simulation

Simulation is not strictly a type of model; models generally *represent* reality, whereas simulation typically *imitates* it. In a practical sense, there are fewer simplifications of reality in simulation models than in other models. In addition, simulation is a technique for *conducting experiments*. Therefore, it involves testing specific values of the decision or uncontrollable variables in the model and observing the impact on the output variables. At DuPont, decision makers had initially chosen to purchase more railcars; however, an alternative involving better scheduling of the existing railcars was developed, tested, and found to have excess capacity, and it ended up saving money.

Simulation is a *descriptive* rather than a *normative* method. There is no automatic search for an optimal solution. Instead, a simulation model describes or predicts the characteristics of a given system under different conditions. When the values of the characteristics are computed, the best of several alternatives can be selected. The simulation process usually repeats an experiment many times to obtain an estimate (and a variance) of the overall effect of certain actions. For most situations, a computer simulation is appropriate, but there are some well-known manual simulations (e.g., a city police department simulated its patrol car scheduling with a carnival game wheel).

Finally, simulation is normally used only when a problem is too complex to be treated using numerical optimization techniques. Complexity in this situation means either that the problem cannot be formulated for optimization (e.g., because the assumptions do not hold), that the formulation is too large, that there are too many interactions among the variables, or that the problem is stochastic in nature (i.e., exhibits risk or uncertainty).

Advantages of Simulation

Simulation is used in decision support modeling for the following reasons:

- The theory is fairly straightforward.
- A great amount of *time compression* can be attained, quickly giving a manager some feel as to the long-term (1- to 10-year) effects of many policies.
- Simulation is descriptive rather than normative. This allows the manager to pose what-if questions. Managers can use a trial-and-error approach to problem solving and can do so faster, at less expense, more accurately, and with less risk.
- A manager can experiment to determine which decision variables and which parts of the environment are really important, and with different alternatives.
- An accurate simulation model requires an intimate knowledge of the problem, thus forcing the MSS builder to constantly interact with the manager. This is desirable for DSS development because the developer and manager both gain a better understanding of the problem and the potential decisions available.
- The model is built from the manager's perspective.
- The simulation model is built for one particular problem and typically cannot solve any other problem. Thus, no generalized understanding is required of the manager; every component in the model corresponds to part of the real system.
- Simulation can handle an extremely wide variety of problem types, such as inventory and staffing, as well as higher-level managerial functions, such as long-range planning.
- Simulation generally can include the real complexities of problems; simplifications are not necessary. For example, simulation can use real probability distributions rather than approximate theoretical distributions.
- Simulation automatically produces many important performance measures.
- Simulation is often the only DSS modeling method that can readily handle relatively unstructured problems.
- Some relatively easy-to-use simulation packages (e.g., Monte Carlo simulation) are available. These include add-in spreadsheet packages (e.g., @RISK), influence diagram software, Java-based (and other Web development) packages, and the visual interactive simulation systems to be discussed shortly.

Disadvantages of Simulation

The primary disadvantages of simulation are as follows:

- An optimal solution cannot be guaranteed, but relatively good ones generally are found.
- Simulation model construction can be a slow and costly process, although newer modeling systems are easier to use than ever.
- Solutions and inferences from a simulation study are usually not transferable to other problems because the model incorporates unique problem factors.
- Simulation is sometimes so easy to explain to managers that analytic methods are often overlooked.
- Simulation software sometimes requires special skills because of the complexity of the formal solution method.

The Methodology of Simulation

Simulation involves setting up a model of a real system and conducting repetitive experiments on it. The methodology consists of the following steps, as shown in Figure 10.4:

1. ***Define the problem.*** We examine and classify the real-world problem, specifying why a simulation approach is appropriate. The system's boundaries, environment, and other such aspects of problem clarification are handled here.
2. ***Construct the simulation model.*** This step involves determination of the variables and their relationships, as well as data gathering. Often the process is described by using a flowchart, and then a computer program is written.
3. ***Test and validate the model.*** The simulation model must properly represent the system being studied. Testing and validation ensure this.
4. ***Design the experiment.*** When the model has been proven valid, an experiment is designed. Determining how long to run the simulation is part of this step. There are two important and conflicting objectives: accuracy and cost. It is also prudent to identify typical (e.g., mean and median cases for random variables), best-case (e.g., low-cost, high-revenue), and worst-case (e.g., high-cost, low-revenue) scenarios. These help establish the ranges of the decision variables and environment in which to work and also assist in debugging the simulation model.
5. ***Conduct the experiment.*** Conducting the experiment involves issues ranging from random-number generation to result presentation.

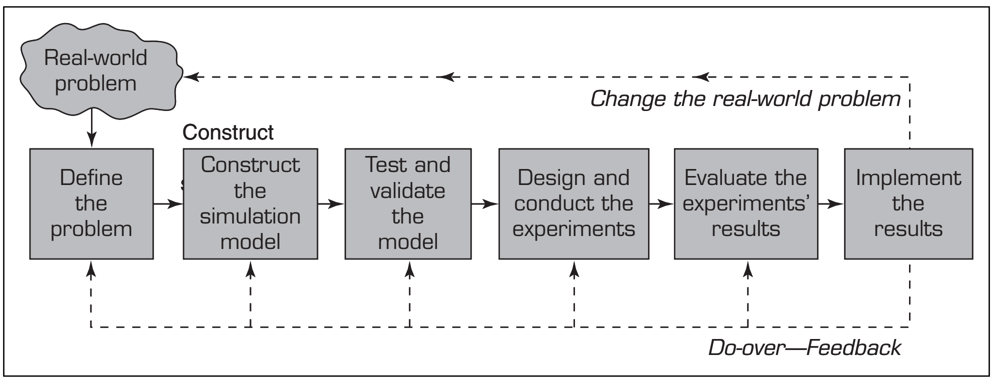


FIGURE 10.4 The Process of Simulation.

6. ***Evaluate the results.*** The results must be interpreted. In addition to standard statistical tools, sensitivity analyses also can be used.
7. ***Implement the results.*** The implementation of simulation results involves the same issues as any other implementation. However, the chances of success are better because the manager is usually more involved with the simulation process than with other models. Higher levels of managerial involvement generally lead to higher levels of implementation success.

Banks and Gibson (2009) presented some useful advice about simulation practices. For example, they list the following seven issues as the common mistakes committed by simulation modelers. The list, though not exhaustive, provides general directions for professionals working on simulation projects.

- Focusing more on the model than on the problem
- Providing point estimates
- Not knowing when to stop
- Reporting what the client wants to hear rather than what the model results say
- Lack of understanding of statistics
- Confusing cause and effect
- Failure to replicate reality

In a follow-up article they provide additional guidelines. The reader should consult this article: **analytics-magazine.org/spring-2009/205-software-solutions-the-abcs-of-simulation-practice.html**

Simulation Types

As we have seen, simulation and modeling are used when pilot studies and experimenting with real systems are expensive or sometimes impossible. Simulation models allow us to investigate various interesting scenarios before making any investment. In fact, in simulations, the real-world operations are mapped into the simulation model. The model consists of relationships and, consequently, equations that all together present the real-world operations. The results of a simulation model, then, depend on the set of parameters given to the model as inputs.

There are various simulation paradigms such as Monte Carlo simulation, discrete event, agent based, or system dynamics. One of the factors that determine the type of simulation technique is the level of abstraction in the problem. Discrete events and agent-based models are usually used for middle or low levels of abstraction. They usually consider individual elements such as people, parts, and products in the simulation models, whereas systems dynamics is more appropriate for aggregate analysis.

In the following sections, we introduce several major types of simulation: probabilistic simulation, time-dependent and time-independent simulation, visual simulation, system dynamics modeling, and agent-based modeling.

PROBABILISTIC SIMULATION In probabilistic simulation, one or more of the independent variables (e.g., the demand in an inventory problem) are probabilistic. They follow certain probability distributions, which can be either discrete distributions or continuous distributions:

- *Discrete distributions* involve a situation with a limited number of events (or variables) that can take on only a finite number of values.
- *Continuous distributions* are situations with unlimited numbers of possible events that follow density functions, such as the normal distribution.

The two types of distributions are shown in Table 10.1.

TABLE 10.1 Discrete Versus Continuous Probability Distributions

Daily Demand	Discrete Probability	Continuous Probability
5	.10	Daily demand is normally distributed with a mean of 7 and a standard deviation of 1.2.
6	.15	
7	.30	
8	.25	
9	.20	

TIME-DEPENDENT VERSUS TIME-INDEPENDENT SIMULATION *Time-independent* refers to a situation in which it is not important to know exactly when the event occurred. For example, we may know that the demand for a certain product is three units per day, but we do not care *when* during the day the item is demanded. In some situations, time may not be a factor in the simulation at all, such as in steady-state plant control design. However, in waiting-line problems applicable to e-commerce, it is important to know the precise time of arrival (to know whether the customer will have to wait). This is a *time-dependent* situation.

Monte Carlo Simulation

In most business decision problems, we usually employ one of the following two types of probabilistic simulations. The most common simulation method for business decision problems is **Monte Carlo simulation**. This method usually begins with building a model of the decision problem without having to consider the uncertainty of any variables. Then we recognize that certain parameters or variables are uncertain or follow an assumed or estimated probability distribution. This estimation is based upon analysis of past data. Then we begin running sampling experiments. Running sampling experiments consists of generating random values of uncertain parameters and then computing values of the variables that are impacted by such parameters or variables. These sampling experiments essentially amount to solving the same model hundreds or thousands of times. Then we can analyze the behavior of these dependent or performance variables by examining their statistical distributions. This method has been used in simulations of physical as well as business systems. A good public tutorial on the Monte Carlo simulation method is available on **Palisade.com** (**palisade.com/risk/monte_carlo_simulation.asp**). Palisade markets @RISK, a popular spreadsheet-based Monte Carlo simulation software. Another popular software in this category has been Crystal Ball, now marketed by Oracle as Oracle Crystal Ball. Of course, it is also possible to build and run Monte Carlo experiments within an Excel spreadsheet without using any add-on software such as the two just mentioned. But these tools make it more convenient to run such experiments in Excel-based models. Monte Carlo simulation models have been used in many commercial applications. Examples include Procter & Gamble using these models to determine hedging foreign-exchange risks; Lilly using the model for deciding optimal plant capacity; Abu Dhabi Water and Electricity Company using @Risk for forecasting water demand in Abu Dhabi; and literally thousands of other actual case studies. Each of the simulation software companies' Web sites includes many such success stories.

One DSS modeling language Planners Lab that was mentioned in Chapter 2 (and is available online for free for academic use) also includes significant Monte Carlo simulation capabilities. The reader is urged to review the online tutorial for Planners Lab to appreciate how easy it can be to build and run Monte Carlo simulation models for analyzing the uncertainty in a problem.

Discrete Event Simulation

Discrete event simulation refers to building a model of a system where the interaction between different entities is studied. The simplest example of this is a shop consisting of a server and customers. By modeling the customers arriving at various rates and the server serving at various rates, we can estimate the average performance of the system, waiting time, the number of waiting customers, etc. Such systems are viewed as collections of customers, queues, and servers. There are thousands of documented applications of discrete event simulation models in engineering, business, etc. Tools for building discrete event simulation models have been around for a long time, but these have evolved to take advantage of developments in graphical capabilities for building and understanding the results of such simulation models. We will discuss this modeling method further in the next section.

VISUAL SIMULATION The graphical display of computerized results, which may include animation, is one of the most successful developments in computer–human interaction and problem solving. We describe this in the next section.

SECTION 10.4 REVIEW QUESTIONS

1. List the characteristics of simulation.
2. List the advantages and disadvantages of simulation.
3. List and describe the steps in the methodology of simulation.
4. List and describe the types of simulation.

10.5 VISUAL INTERACTIVE SIMULATION

We next examine methods that show a decision maker a representation of the decision-making situation in action as it runs through scenarios of the various alternatives. These powerful methods overcome some of the inadequacies of conventional methods and help build trust in the solution attained because they can be visualized directly.

Conventional Simulation Inadequacies

Simulation is a well-established, useful, descriptive, mathematics-based method for gaining insight into complex decision-making situations. However, simulation does not usually allow decision makers to see how a solution to a complex problem evolves over (compressed) time, nor can decision makers interact with the simulation (which would be useful for training purposes and teaching). Simulation generally reports statistical results at the end of a set of experiments. Decision makers are thus not an integral part of simulation development and experimentation, and their experience and judgment cannot be used directly. If the simulation results do not match the intuition or judgment of the decision maker, a *confidence gap* in the results can occur.

Visual Interactive Simulation

Visual interactive simulation (VIS), also known as **visual interactive modeling (VIM)** and *visual interactive problem solving*, is a simulation method that lets decision makers see what the model is doing and how it interacts with the decisions made, as they are made. The technique has been used with great success in operations management DSS. The user can employ his or her knowledge to determine and try different decision strategies while interacting with the model. Enhanced learning, about both the problem and the impact of the alternatives tested, can and does occur. Decision makers also contribute to model validation. Decision makers who use VIS generally support and trust their results.

VIS uses animated computer graphic displays to present the impact of different managerial decisions. It differs from regular graphics in that the user can adjust the decision-making process and see the results of the intervention. A visual model is a graphic used as an integral part of decision making or problem solving, not just as a communication device. Some people respond better than others to graphical displays, and this type of interaction can help managers learn about the decision-making situation.

VIS can represent static or dynamic systems. Static models display a visual image of the result of one decision alternative at a time. Dynamic models display systems that evolve over time, and the evolution is represented by animation. The latest visual simulation technology has been coupled with the concept of virtual reality, where an artificial world is created for a number of purposes, from training to entertainment to viewing data in an artificial landscape. For example, the U.S. military uses VIS systems so that ground troops can gain familiarity with terrain or a city in order to very quickly orient themselves. Pilots also use VIS to gain familiarity with targets by simulating attack runs. The VIS software can also include GIS coordinates.

Visual Interactive Models and DSS

VIM in DSS has been used in several operations management decisions. The method consists of priming (like priming a water pump) a visual interactive model of a plant (or company) with its current status. The model then runs rapidly on a computer, allowing managers to observe how a plant is likely to operate in the future.

Waiting-line management (queuing) is a good example of VIM. Such a DSS usually computes several measures of performance for the various decision alternatives (e.g., waiting time in the system). Complex waiting-line problems require simulation. VIM can display the size of the waiting line as it changes during the simulation runs and can also graphically present the answers to what-if questions regarding changes in input variables. Application Case 10.4 gives an example of a visual simulation that was used to explore the applications of RFID technology in developing new scheduling rules in a manufacturing setting.

Application Case 10.4

Improving Job-Shop Scheduling Decisions Through RFID: A Simulation-Based Assessment

A manufacturing services provider of complex optical and electro-mechanical components seeks to gain efficiency in its job-shop scheduling decision because the current shop-floor operations suffer from a few issues:

- There is no system to record when the work-in-process (WIP) items actually arrive at or leave operating workstations and how long those WIPs actually stay at each workstation.
- The current system cannot monitor or keep track of the movement of each WIP in the production line in real time.

As a result, the company is facing two main issues at this production line: high backlogs and high costs of overtime to meet the demand. Additionally, the upstream cannot respond to unexpected incidents such as changes in demand or material shortages quickly enough and revise schedules in a cost-effective manner. The company is considering implementing RFID on a production line. A discrete event simulation program is then developed to examine how track and traceability through RFID can facilitate job-shop production scheduling activities.

The visibility-based scheduling (VBS) rule that utilizes the real-time traceability systems to track those WIPs, parts and components, and raw materials in shop-floor operations is proposed. A simulation approach is applied to examine the benefit of the VBS rule against the classical scheduling rules: the first-in-first-out (FIFO) and earliest due date (EDD) dispatching rules. The simulation model is developed using Simio^{TM}. Simio is a 3D simulation modeling software package that employs an object-oriented approach to modeling and has recently been used in many areas such as factories, supply chains, healthcare, airports, and service systems.

Figure 10.5 presents a screenshot of the SIMIO interface panel of this production line. The parameter estimates used for the initial state in the simulation model include weekly demand and forecast, process flow, number of workstations, number of shop-floor operators, and operating time at each workstation. Additionally, parameters of some of the input data such as RFID tagging time, information retrieving time, or system updating time are estimated from a pilot study and from the subject matter experts. Figure 10.6 presents the process view of the simulation model where specific simulation commands are implemented and coded. Figures 10.7 and 10.8 present the standard report view and pivot grid report of the simulation model. The standard report and pivot grid format provide a very quick method to find specific statistical results such as average, percent, total, maximum, or minimum values of variables assigned and captured as an output of the simulation model.

The results of the simulation suggest that an RFID-based scheduling rule generates better performance compared to traditional scheduling rules with regard to processing time, production time, resource utilization, backlogs, and productivity.

Source: Based on J. Chongwatpol and R. Sharda, "RFID-Enabled Track and Traceability in Job-Shop Scheduling Environment," *European Journal of Operational Research*, Vol. 227, No. 3 , pp. 453–463, **2013 http://dx.doi.org/10.1016/j.ejor.2013.01.009**.

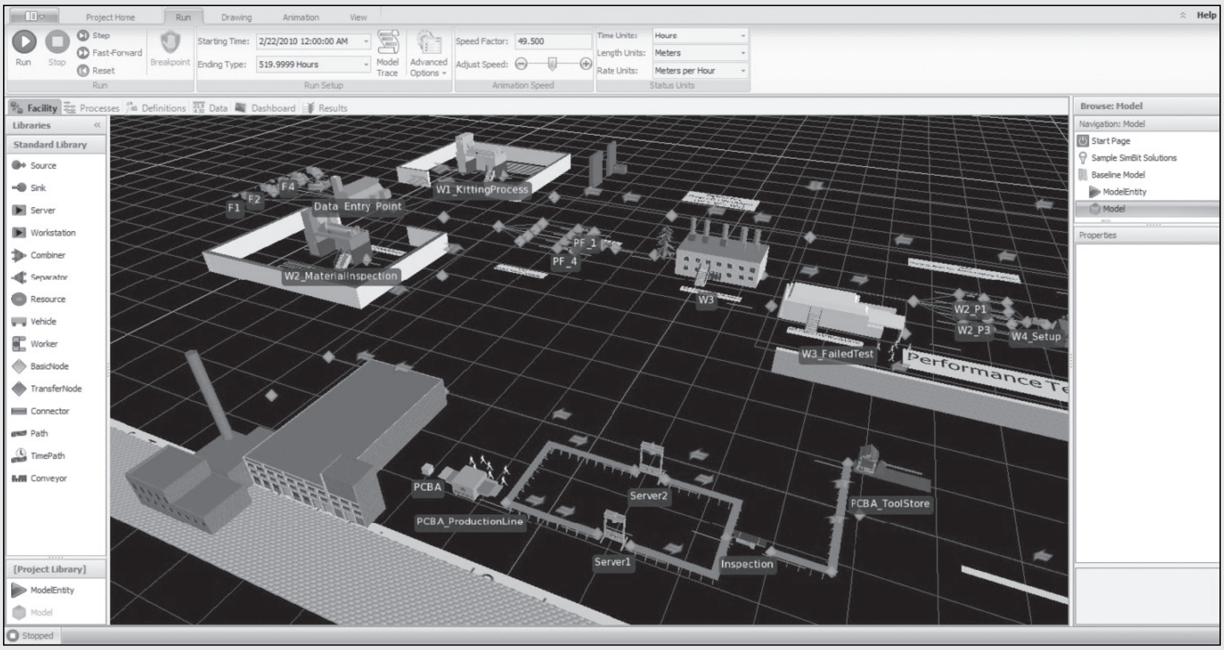


FIGURE 10.5 **SIMIO Interface View of the Simulation System.**

(Continued)

Application Case 10.4 (Continued)

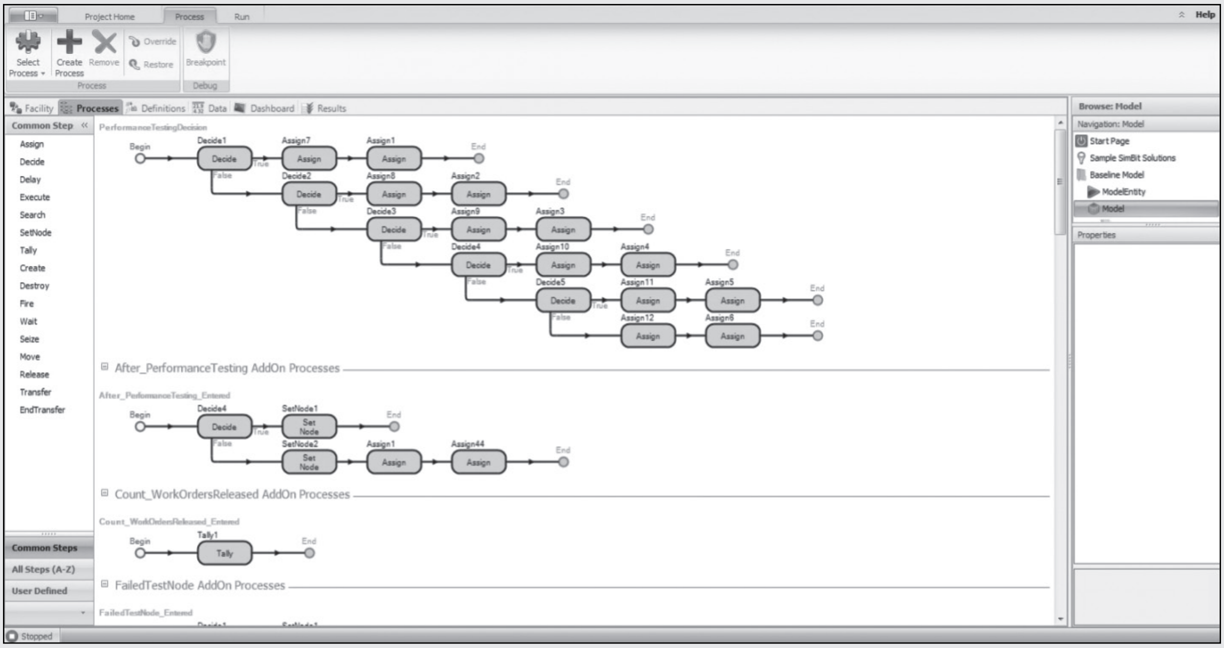


FIGURE 10.6 Process View of the Simulation Model.

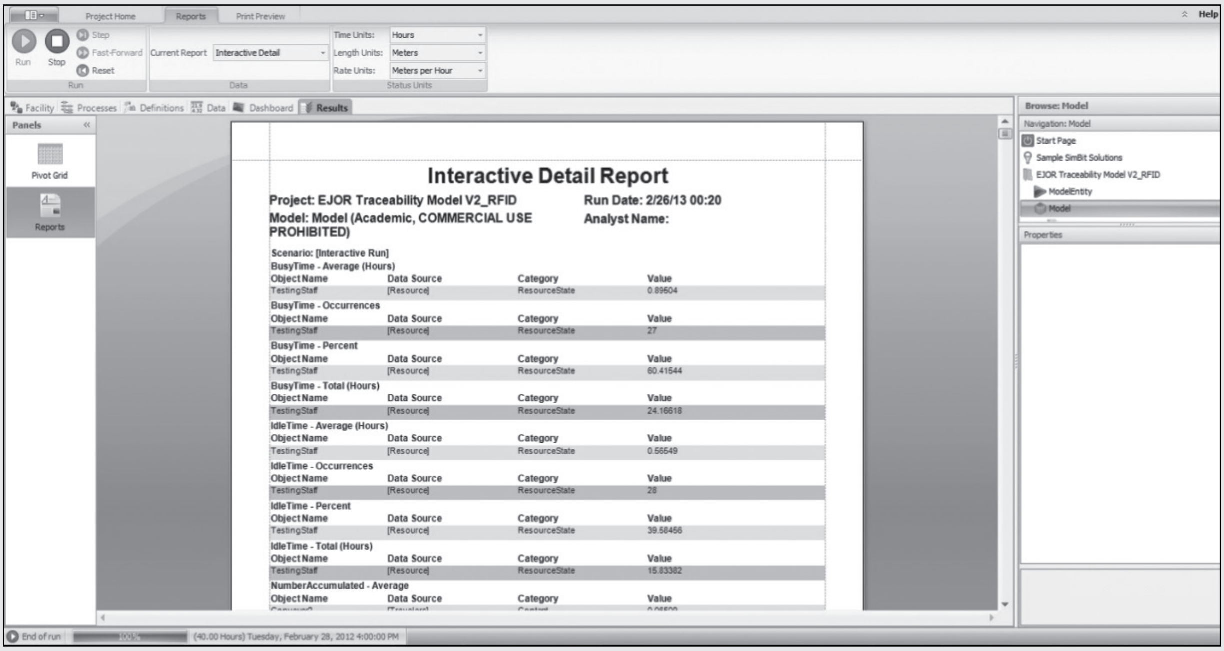


FIGURE 10.7 Standard Report View.

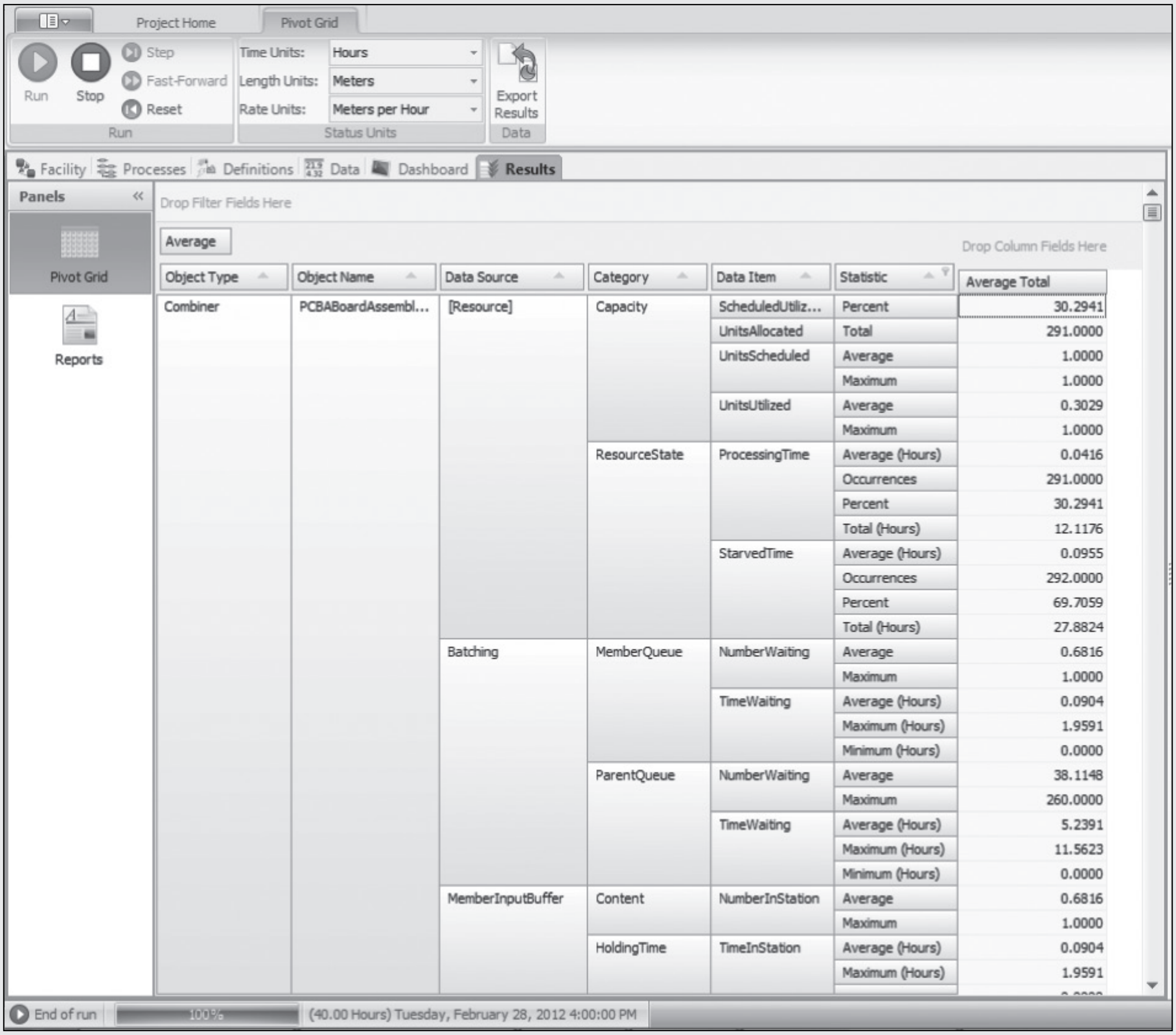


FIGURE 10.8 Pivot Grid Report from a SIMIO Run.

The VIM approach can also be used in conjunction with artificial intelligence. Integration of the two techniques adds several capabilities that range from the ability to build systems graphically to learning about the dynamics of the system. These systems, especially those developed for the military and the video-game industry, have "thinking" characters who can behave with a relatively high level of intelligence in their interactions with users.

Simulation Software

Hundreds of simulation packages are available for a variety of decision-making situations. Many run as Web-based systems. *ORMS Today* publishes a periodic review of simulation software. One recent review is located at **orms-today.org/surveys/Simulation/Simulation.html** (accessed February 2013). PC software packages include Analytica

(Lumina Decision Systems, **lumina.com**) and the Excel add-ins Crystal Ball (now sold by Oracle as Oracle Crystal Ball, **oracle.com**) and @RISK (Palisade Corp., **palisade.com**). A major commercial software for discrete event simulation has been Arena (sold by Rockwell Intl., **arenasimulation.com**). Original developers of Arena have now developed Simio (**simio.com**), which was used in the screens shown above. Another popular discrete event VIS software is ExtendSim (**extendsim.com**). SAS has a graphical analytics software package called JMP that also includes a simulation component in it.

For information about simulation software, see the Society for Modeling and Simulation International (**scs.org**) and the annual software surveys at *ORMS Today* (**orms-today.com**).

SECTION 10.5 REVIEW QUESTIONS

1. Define *visual simulation* and compare it to conventional simulation.
2. Describe the features of VIS (i.e., VIM) that make it attractive for decision makers.
3. How can VIS be used in operations management?
4. How is an animated film like a VIS application?

10.6 SYSTEM DYNAMICS MODELING

System dynamics was introduced in the opening vignette as a powerful method of analysis. System dynamics models are macro-level simulation models in which aggregate values and trends are considered. The objective is to study the overall behavior of a system over time, rather than the behavior of each individual participant or player in the system. The other major key dimension is the evolution of the various components of the system over time and as a result of interplay between the components over time. System dynamics (SD) was first introduced by Forrester (1958) to address problems in industrial systems. He later expanded his work and used system dynamics to model and simulate a classic supply chain (1961). Since then, system dynamics has contributed to theory building, problem solving, and research methodology. SD has been used with operations research and management science approaches (Angerhofer & Angelides, 2000) where SD and operations research are considered complementary techniques in which SD can provide a more qualitative analysis for understanding a system, while operations research techniques build analytical models of the problem. System dynamics has been used extensively in the area of information technology, which usually changes an organization's business processes and behavior. Using system dynamics, possible changes in organizations are projected and analyzed through conceptual models and simulations. The SD technique also has been used in evaluating IT investments: Marquez and Blanchar (2006) developed a system dynamics model to analyze a variety of investment strategies in a high-tech company. Their simulation allows them to analyze strategies and trade-offs that are hard to investigate in real cases. A system dynamics model can capture IT benefits that are sometimes nonlinear and achieved over years.

To create an SD model, we need to draw causal loop diagrams for all processes that lead to some benefits. This is a qualitative step in which the processes, variables, and relationships within the conceptual model are identified. These causal loop diagrams are then transformed into mathematical equations that represent the relations among variables. The equations and stock and flow diagrams are then used to simulate different practical and theoretical scenarios.

Causal loop diagrams show the relationships between variables in a system. A link between two elements shows that changes in one element lead to changes in the other one. The direction of the link shows the direction of influence between two elements. The sign of each arrow shows the direction of change between each pair of elements.

A positive sign means both elements change in the same direction while a negative sign means the elements change in opposite directions. Feedback processes in the **causal loops** are the key components by which a variable re-affects itself over time through a chain of causal relationships.

We illustrate a basic application of system dynamics modeling through a partial model of the impact of electronic health record (EHR) systems. This is based on Kasiri, Sharda, and Asamoah (2010). Implementing electronic health record (EHR) systems is on the agenda for many healthcare organizations in the next few years. Before investing in an EHR system, however, decision makers need to identify and measure the benefits of such systems. Using a system dynamics approach, it is possible to map complex relationships among healthcare processes into a model by which one can dynamically measure the effect of any changes in the parameters over time. Simulation of EHR implementations using a system dynamics model produces useful data on the benefits of EHRs that are hard to obtain through empirical data collection methods. The results of an SD model can then be transformed into economic values to estimate financial performance.

Let us consider some of the factors that impact healthcare delivery in the hospital as a result of the implementation of an electronic health records system. The causal loop diagram in Figure 10.9 shows how different processes and variables interrelate in an electronic health records system to offer significant benefits to healthcare delivery. The sign on each arrow indicates the direction of change between each pair of elements. A positive relationship means both elements change in the same direction while a negative relationship means the elements change in opposite directions.

Electronic notes (e-notes) and electronic prescribing (e-Rx) are shown as two common processes in EHRs that contribute to an increase in the amount of staff time saved

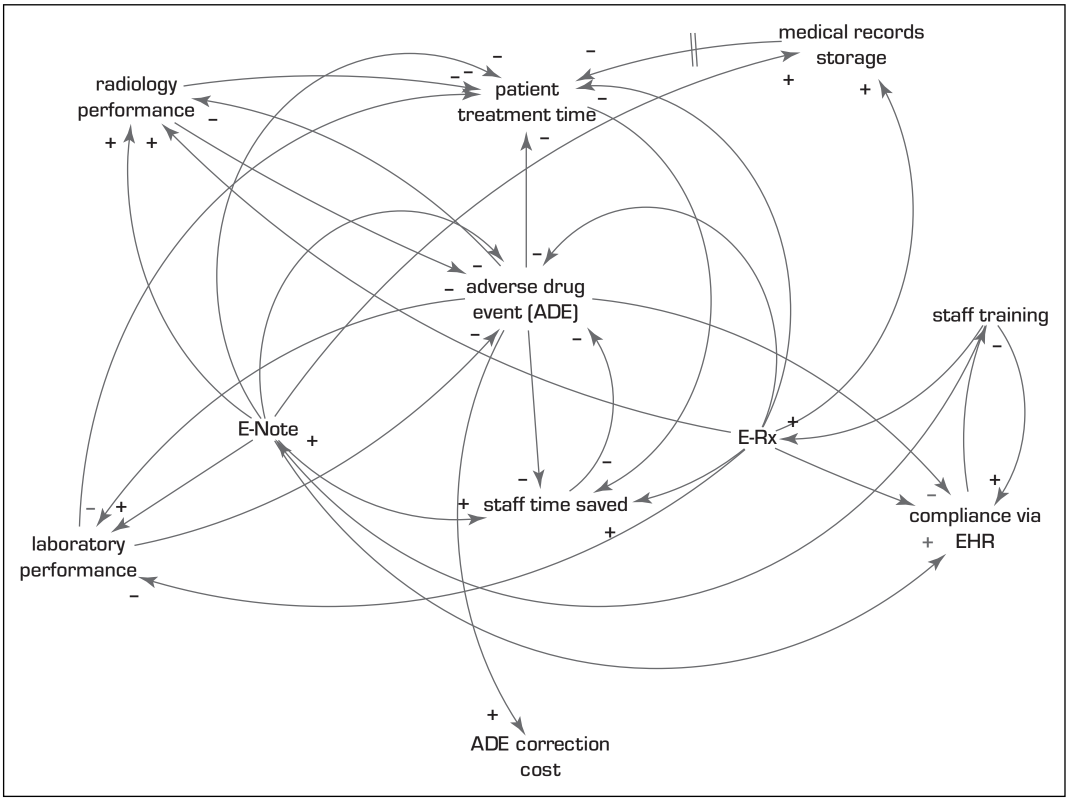


FIGURE 10.9 Causal Loop Diagram for Effects of EHR. *Source:* From Kasiri et al., 2010.

(McGowan et al., 2008). They also contribute to a decrease in patient treatment time, which is the time it takes for a patient to receive medical assistance starting from initial contact with the receptionist to the time he or she leaves the hospital after receiving medical care from the physician and other hospital staff. The average increase in patient treatment time as a result of adverse drug events (ADEs) is 1.74 per occurrence (Classen et al., 1997). According to Anderson (2002), entering records directly entered into computer-based medical information systems contributes to increased quality of care and reduces costs related to ADEs. Hence, instead of paper notes and paper prescriptions, doctors can reduce costs when notes on patients and prescriptions are entered directly into the EHR system. Quality of care is directly affected by the amount of time a patient spends at the hospital. Based on the diagram in Figure 10.9, there is a positive link between e-note and staff time saved as well as e-Rx and staff time saved. This indicates that the more physicians use the EHR system, the less time nurses and other staff need to manually retrieve records and files on patients in order to offer medical support to them; in fact, there is no need to transfer files and paper documents from one department to another physically. Staff can therefore transfer the time saved on dealing with documentation to having direct contact with the patients and, hence, improve the quality of healthcare given to patients and decrease ADEs.

E-note and e-Rx also impact the occurrence of adverse drug events (Garrido, 2005; McGowan et al., 2008). The more the system is utilized to record notes on patients and to write prescriptions, the fewer the mistakes in the administration of drugs that stem directly from inefficiencies in manual drug administration processes. Hence, patients spend less time at the hospital as a result of not having to deal with delays related to complications that could occur with paper notes and paper prescriptions. Also, "staff time saved" is increased because the time needed to correct the mistakes related to ADEs is eliminated. The occurrence of ADEs in hospitals is estimated to be an average of 6.5 events per 100 hospitals (Bates et al., 1995; Leape et al., 1995). Subsequently, when the ADE rate decreases through the use of e-note and e-Rx, ADE correction costs also decrease.

The electronic records storage (e-storage) variable refers to the capability to store records in the hospital that otherwise would have been stored in paper format. E-storage is important because it helps in easy retrieval of medical records of patients even after many years. For instance, EHR enables the use of e-note and e-Rx, electronic forms of paper notes and paper prescriptions, which are easier to store and retrieve than are data in hard-copy formats. Hence, EHR helps facilitate the storage and retrieval of health records. Access to the patient's electronic health records helps physicians easily make decisions and diagnoses based on past records. The delay link from e-storage to patient treatment time indicates that patients can be taken care of much faster if electronic data that offer quicker retrieval are available. Uncertainty in clinical decision making on the part of physicians is greatly reduced as a result of e-storage capability (Garrido, 2005). Of course, electronic storage of this data is enhanced by greater use of e-Rx and e-note.

Hospitals are required to comply with certain standards regarding the administration of medication and other related healthcare administration processes (Sidorov, 2006). Certain drugs may be restricted, and the amount given to a particular patient must be closely watched at any period in time by staff. With EHR, physicians can easily track patients' records to know how much has been given and what amount is yet to be given. If an attempt is made to prescribe an amount that is more than the requisite amount for that particular patient, a "red flag message" can be generated to warn the physician of the imminent breach in compliance. In this way, it is easier to comply with regulations regarding the dispensing of a particular medicine and ensure that the maximum amount that is supposed to be given to the patient is not exceeded. Also, rules can be set in the EHR system to prevent physicians from prescribing certain combinations of drugs because of negative reactions such combinations may cause. If a particular rule is violated during e-prescribing, a warning message can be immediately generated to warn the physician of the imminent danger. ADEs that may occur as a result of incorrect amounts and combinations of drugs given can hence be minimized.

The likelihood that any information system in an organization will be used is closely related to how well the users are trained in using the system. Hence, when staff, including nurses, physicians, and lab assistants, are given adequate periodic training, the use and acceptance of e-Rx, e-note, and the EHR system in general increases. Training also leads to greater compliance with standards.

In addition, when EHR is integrated with other healthcare delivery departments such as the radiology and laboratory departments, their performance level is increased. Greater efficiency in the radiology and laboratory departments leads to fewer ADEs and shorter patient treatment times. Also, using EHR reduces the rate of duplication in radiology work and provides quicker access to radiology records and, hence, directly increases the savings in staff time. With the EHR system, a functional department like the radiology department can directly access the patient's x-ray order through the e-note functionality. Hence, mistakes related to incorrect interpretation of physicians' handwritten orders can be avoided, leading to a decrease in patient treatment time at the hospital.

The causal loop diagram shows various benefits of EHRs such as lower rate of ADEs, higher amounts of staff time saved, and lower patient treatment times. In the next section, we develop a stock and flow diagram with loops that reflect some of the most important factors that impact the flows. These relationships and effects can be translated into mathematical equations for simulation purposes. Based on estimated parameters and initial values, we simulate the model and discuss the results.

Because the goal of this section is only to introduce some concepts of system dynamics simulation, we will not go into all the details of the technique. Once the causal loop diagrams are built, one can build the stock and flow diagrams, which lead to developing the mathematical equations for simulating the behavior of the underlying system under study. Results can provide considerable insight into the growing behavior of the system under consideration. In another project, Kasiri and Sharda (2012), for example, studied the effects of introducing radio-frequency identification (RFID) tags in retail stores on each item. They built system dynamics models to identify impacts of such technology in a retail store—increased visibility of information about what is on the shelves leading to a decrease in inventory inaccuracy, better pricing management, etc. Industry participants were able to provide inputs on such effects to be able to build models for investment decisions.

Many software tools are now available for building system dynamics models. Such listings are usually updated on Wikipedia and other sites. Some of the popular tools that include academic and commercial pricings include VenSim, Vissim, and many others. One free software, Insightmaker, appears to offer both system dynamics and agent-based modeling capabilities in its Web version.

SECTION 10.6 REVIEW QUESTIONS

1. What is the key difference between system dynamics simulation and other simulation types?
2. What is the purpose of a causal loop diagram?
3. How are relationships between two variables represented in a causal loop diagram?

10.7 AGENT-BASED MODELING

The term *agent* is derived from the concept of agency, referring to employing someone to act on one's behalf. A human agent represents a person and interacts with others to accomplish a predefined task. The concept of agents goes surprisingly far back.

More than 60 years ago, Vannevar Bush envisioned a machine called a *memex*. He imagined the memex assisting humans to manage and process huge amounts of data and information.

Agent-based modeling (ABM) is a simulation modeling technique to support complex decision systems where a system or network is modeled as a set of autonomous decision-making units called *agents* that individually evaluate their situation and make decisions on the basis of a set of predefined behavior and interaction rules. This technique is a bottom-up approach to modeling complex systems particularly suitable for understanding evolving and dynamic systems. An ABM approach focuses on modeling an "adaptive learning" property rather than "optimizing" nature. Characteristics such as heterogeneity, rule of thumb, or optimization strategies and adaptive learning leading to new capabilities in the system can be defined as a set of rules and behaviors. Also, ABM is able to capture emergent phenomena that exhibit as a result of interacting components of a system with each other, and influencing each other through these interactions. These kinds of characteristics make a system difficult to understand and predict and inherently more unstable. Flocks of birds, social dynamics of science, and the birth and decline of disciplines (Sun, Kaur, et al., 2013), traffic jams and crowds simulation, ant colony, financial contagion, movements of ancient societies (2005), housing segregation and other urban issues (Crooks, 2010), disease propagation (Carley, Altman, et al., 2004), and operations management problems (Caridi and Cavalieri, 2004; Allwood and Lee, 2005, 2008) are some past applications of ABMs. For business problems in which many interrelated factors, irregular data, and high uncertainty and emergent behaviors exist, interactions between agents are complex, discrete, or nonlinear, the population is heterogeneous, agents exhibit learning and adaptive behaviors and also spatial issues, or social networks are of interest, agent-based modeling can be used.

According to the framework developed by Macal and North (2005), to build an agent-based model, the following steps should be taken. First of all, it should be questioned what specific problem should be solved by the model, and particularly what values agent-based modeling brings to the problem that the other problem-solving approaches cannot bring. The second step includes identifying the agents and getting a theory of agent behavior. What agents should be included in the model, who are the decision makers in the system, which agents have behaviors? What kinds of data on agents are available? Is it simply descriptive (static attributes)? Or does it have to be calculated endogenously by the model and informed to the agents (dynamic attributes)? Third, the agent relationships should be identified and a theory of agent interaction should be taken into account. That is, the agents' environment should be studied to determine how the agents interact with the environment, what agent behaviors are of interest, what behavior and interaction rules the agent creates and follows, what decisions the agents make, and what behaviors or actions are being acted upon by the agents. Next, the required agent-related data should be collected. Finally, the performance of the agent-based system should be validated against reality either at the individual agent level or the model as a whole; particularly, the agent behaviors should be examined.

Agent-based modeling can be implemented either using general programing languages or through some specially designed applications that address the requirements of agent modeling. Among agent-based platforms, SWARM (**www.swarms.org**), Netlogo (**http://ccl.northwestern.edu/netlogo**), RePast/Sugarscape (**www.repast.sourceforge.net**), and Escape (**www.metascapeabm.com**) provide an appropriate graphical user interface and comprehensive documentation (Railsback, Lytinen, et al., 2006). Application Case 10.5 describes a really useful application of agent-based modeling to simulate effects of disease mitigation strategies.

Application Case 10.5

Agent-Based Simulation Helps Analyze Spread of a Pandemic Outbreak

Knowledge about the spread of a disease plays an important role in both preparing for and responding to a pandemic outbreak. Previous models for such analyses are mostly homogenous and make use of simplistic assumptions about transmission and the infection rates. These models assume that each individual in the population is identical and typically has the same number of potential contacts with an infected individual in the same time period. Also each infected individual is assumed to have the same probability to transmit the disease. Using these models, implementing any mitigation strategies to vaccinate the susceptible individuals and treating the infected individuals become extremely difficult under limited resources.

In order to effectively choose and implement a mitigation strategy, modeling of the disease spread has to be done across the specific set of individuals, which enables researchers to prioritize the selection of individuals to be treated first and also gauge the effectiveness of mitigation strategy.

Although nonhomogenous models for spread of a disease can be built based on individual characteristics using the interactions in a contact network, such individual levels of infectivity and vulnerability require complex mathematics to obtain the information needed for such models.

Simulation techniques can be used to generate hypothetical outcomes of disease spread by simulating events on the basis of hourly, daily, or other periods and tallying the outcomes throughout the simulation. A nonhomogenous agent-based simulation approach allows each member of the population to be simulated individually, considering the unique individual characteristics that affect the transmission and infection probabilities. Furthermore, individual behaviors that affect the type and length of contact between individuals, and the possibility of infected individuals recovering and becoming immune, can also be simulated via **agent-based models**.

One such simulation model, built for the Ontario Agency for Health Protection and Promotion (OAHPP) following the global outbreak of severe acute respiratory syndrome (SARS) in 2002–2003, simulated the spread of disease by applying various mitigation strategies. The simulation models each state of an individual in each time unit, based on the individual probabilities to transition from susceptible state to infected stage and then to recovered state and back to susceptible state. The simulation model also uses an individual's duration of contact with infected individuals. The model also accounts for the rate of disease transmission per time unit based on the type of contact between individuals and for behavioral changes of individuals in a disease progression (being quarantined or treated or recovered). It is flexible enough to consider several factors affecting the mitigation strategy, such as an individual's age, residence, level of general interaction with other members of population, number of individuals in each household, distribution of households, and behavioral aspects involving daily commutes, attendance at schools, and asymptotic time period of disease.

The simulation model was tested to measure the effectiveness of a mitigation strategy involving an advertising campaign that urged individuals who have symptoms of disease to stay at home rather than commute to work or school. The model was based on a pandemic influenza outbreak in the greater Toronto area. Each individual agent, generated from the population, was sequentially assigned to households. Individuals were also assigned to different ages based on census age distribution; all other pertinent demographic and behavioral attributes were assigned to the individuals.

The model considered two types of contact: close contact, which involved members of the same household or commuters on the public transport; and causal contact, which involved random individuals among the same census tract. Influenza pandemic records provided past disease transmission data, including transmission rates and contact time for both close and causal contacts. The effect of public transportation was simplified with an assumption that every individual of working age used the nearest subway line to travel. An initial outbreak of infection was fed into the model. A total of 1,000 such simulations was conducted.

The results from the simulation indicated that there was a significant decrease in the levels of infected

(*Continued*)

Application Case 10.5 (Continued)

and deceased persons as an increasing number of infected individuals followed the mitigation strategy of staying at home. The results were also analyzed by answering questions that sought to verify issues such as the impact of 20 percent of infected individuals staying at home versus 10 percent staying at home. The results from each of the simulation outputs were fed into geographic information system software, ESRI ArcGIS, and detailed shaded maps of the greater Toronto area, showing the spread of disease based on the average number of cumulative infected individuals. This helped to determine the effectiveness of a particular mitigation strategy. This agent-based simulation model provides a what-if analysis tool that can be used to compare relative outcomes of different disease scenarios and mitigation strategies and help in choosing the effective mitigation strategy.

Questions for Discussion

1. What are the characteristics of an agent-based simulation model?
2. List the various factors that were fed into the agent-based simulation model described in the case.
3. Elaborate on the benefits of using agent-based simulation models.
4. Besides disease prevention, in which other situations could agent-based simulation be employed?

What We Can Learn from This Application Case

Advancements in computing technology allow for building advanced simulation models that are nonhomogeneous in nature and factor for many socio-demographic and behavioral factors. These simulation models further enhance the support for policy decision making by hypothetically simulating many real-time complex problem situations.

Source: D. M. Aleman, T. G. Wibisono, and B. Schwartz, "A Nonhomogeneous Agent-Based Simulation Approach to Modeling the Spread of Disease in a Pandemic Outbreak," *Interfaces*, Vol. 41, No. 3, 2011, pp. 301–315.

Chapter Highlights

- Heuristic programming involves problem solving using general rules or intelligent search.
- Genetic algorithms are search techniques that emulate the natural process of biological evolution. They utilize three basic operations: reproduction, crossover, and mutation.
- Reproduction is a process that creates the next-generation population based on the performance of different cases in the current population.
- Crossover is a process that allows elements in different cases to be exchanged to search for a better solution.
- Mutation is a process that changes an element in a case to search for a better solution.
- Simulation is a widely used DSS approach that involves experimentation with a model that represents the real decision-making situation.
- Simulation can deal with more complex situations than optimization, but it does not guarantee an optimal solution.
- There are many different simulation methods. Some that are important for DSS include Monte Carlo simulation, discrete event simulation, systems dynamics modeling, and agent-based simulations.
- VIS/VIM allows a decision maker to interact directly with a model and shows results in an easily understood manner.

Key Terms

agent-based models
causal loops
discrete event simulation
chromosome
crossover
elitism
evolutionary algorithm
genetic algorithm
heuristic programming
heuristics
Monte Carlo simulation
mutation
reproduction
simulation
system dynamics
visual interactive modeling (VIM)
visual interactive simulation (VIS)

Questions for Discussion

1. Compare the effectiveness of genetic algorithms against standard methods for problem solving, as described in the literature. How effective are genetic algorithms?
2. Describe the general process of simulation.
3. List some of the major advantages of simulation over optimization and vice versa.
4. What are the advantages of using a spreadsheet package to perform simulation studies? What are the disadvantages?
5. Compare the methodology of simulation to Simon's four-phase model of decision making. Does the methodology of simulation map directly into Simon's model? Explain.
6. Many computer games can be considered visual simulation. Explain why.
7. Explain why VIS is particularly helpful in implementing recommendations derived by computers.

Exercises

Teradata University Network (TUN) and Other Hands-on Exercises

1. Each group in the class should access a different online Java-based Web simulation system (especially those systems from visual interactive simulation vendors) and run it. Write up your experience and present it to the class.
2. Solve the knapsack problem from Section 10.3 manually, and then solve it using Evolver. Try another code (find one on the Web). Finally, develop your own genetic algorithm code in Visual Basic, C++, or Java.
3. Search online to find vendors of genetic algorithms and investigate the business applications of their products. What kinds of applications are most prevalent?
4. Go to **palisade.com** and examine the capabilities of Evolver. Write a summary about your findings.
5. Each group should review, examine, and demonstrate in class a different state-of-the-art DSS software product. The specific packages depend on your instructor and the group interests. You may need to download a demo from a vendor's Web site, depending on your instructor's directions. Be sure to get a running demo version, not a slideshow. Do a half-hour in-class presentation, which should include an explanation of why the software is appropriate for assisting in decision making, a hands-on demonstration of selected important capabilities of the software, and your critical evaluation of the software. Try to make your presentation interesting and instructive to the whole class. The main purpose of the class presentation is for class members to see as much state-of-the-art software as possible, both in breadth (through the presentations by other groups) and in depth (through the experience you have in exploring the ins and outs of one particular software product). Write a 5- to 10-page report on your findings and comments regarding this software. Include screenshots in your report. Would you recommend this software to anyone? Why or why not?

End-of-Chapter Application Case

HP Applies Management Science Modeling to Optimize Its Supply Chain and Wins a Major Award

HP's groundbreaking use of operations research not only enabled the high-tech giant to successfully transform its product portfolio program and return $500 million to the bottom line over a 3-year period, but it also earned HP the coveted 2009 Edelman Award from INFORMS for outstanding achievement in operations research. "This is not the success of just one person or one team," said Kathy Chou, vice president of Worldwide Commercial Sales at HP, in accepting the award on behalf of the winning team. "It's the success of many people across HP who made this a reality, beginning several years ago with mathematics and imagination and what it might do for HP."

To put HP's product portfolio problem into perspective, consider these numbers: HP generates more than $135 billion annually from customers in 170 countries by offering tens of thousands of products supported by the largest supply chain in the industry. You want variety? How about 2,000 laser printers and more than 20,000 enterprise servers and storage products? Want more? HP offers more than 8 million configure-to-order combinations in its notebook and desktop product line alone.

The something-for-everyone approach drives sales, but at what cost? At what point does the price of designing, manufacturing, and introducing yet another new product, feature, or option exceed the additional revenue it is likely to generate? Just as important, what are the costs associated with too much or too little inventory for such a product, not to mention additional supply chain complexity, and how does all of that impact customer satisfaction? According to Chou, HP didn't have good answers to any of those questions before the Edelman award–winning work.

"While revenue grew year over year, our profits were eroded due to unplanned operational costs," Chou said in

HP's formal Edelman presentation. "As product variety grew, our forecasting accuracy suffered, and we ended up with excesses of some products and shortages of others. Our suppliers suffered due to our inventory issues and product design changes. I can personally testify to the pain our customers experienced because of these availability challenges." Chou would know. In her role as VP of Worldwide Commercial Sales, she's "responsible and on the hook" for driving sales, margins, and operational efficiency.

Constantly growing product variety to meet increasing customer needs was the HP way—after all, the company is nothing if not innovative—but the rising costs and inefficiency associated with managing millions of products and configurations "took their toll," Chou said, "and we had no idea how to solve it."

Compounding the problem, Chou added, was HP's "organizational divide." Marketing and sales always wanted more—more SKUs, more features, more configurations—and for good reason. Providing every possible product choice was considered an obvious way to satisfy more customers and generate more sales.

Supply chain managers, however, always wanted less. Less to forecast, less inventory, and less complexity to manage. "The drivers (on the supply chain side) were cost control," Chou said. "Supply chain wanted fast and predictable order cycle times. With no fact-based, data-driven tools, decision making between different parts of the organization was time-consuming and complex due to these differing goals and objectives."

By 2004, HP's average order cycle times in North America were nearly twice that of its competition, making it tough for the company to be competitive despite its large variety of products. Extensive variety, once considered a plus, had become a liability.

It was then that the Edelman prize–winning team—drawn from various quarters both within the organization (HP Business Groups, HP Labs, and HP Strategic Planning and Modeling) and out (individuals from a handful of consultancies and universities) and armed with operations research thinking and methodology—went to work on the problem. Over the next few years, the team: (1) produced an analytically driven process for evaluating new products for introduction, (2) created a tool for prioritizing existing products in a portfolio, and (3) developed an algorithm that solves the problem many times faster than previous technologies, thereby advancing the theory and practice of network optimization.

The team tackled the product variety problem from two angles: prelaunch and postlaunch. "Before we bring a new product, feature, or option to market, we want to evaluate return on investment in order to drive the right investment decisions and maximize profits," Chou said. To do that, HP's Strategic Planning and Modeling Team (SPaM) developed "complexity return on investment screening calculators" that took into account downstream impacts across the HP product line and supply chain that were never properly accounted for before.

Once a product is launched, variety product management shifts from screening to managing a product portfolio as sales data become available. To do that, the Edelman award–winning team developed a tool called revenue coverage optimization (RCO) to analyze more systematically the importance of each new feature or option in the context of the overall portfolio.

The RCO algorithm and the complexity ROI calculators helped HP improve its operational focus on key products, while simultaneously reducing the complexity of its product offerings for customers. For example, HP implemented the RCO algorithm to rank its Personal Systems Group offerings based on the interrelationship between products and orders. It then identified the "core offering," which is composed of the most critical products in each region. This core offering represented about 30 percent of the ranked product portfolio. All other products were classified as HP's "extended offering."

Based on these findings, HP adjusted its service level for each class of products. Core offering products are now stocked in higher inventory levels and are made available with shorter lead times, and extended offering products are offered with longer lead times and are either stocked at lower levels or not at all. The net result: lower costs, higher margins, and improved customer service.

The RCO software algorithm was developed as part of HP Labs' "analytics" theme, which applies mathematics and scientific methodologies to help decision making and create better-run businesses. Analytics is one of eight major research themes of HP Labs, which last year refocused its efforts to address the most complex challenges facing technology customers in the next decade.

"Smart application of analytics is becoming increasingly important to businesses, especially in the areas of operational efficiency, risk management, and resource planning," says Jaap Suermondt, director, Business Optimization Lab, HP Labs. "The RCO algorithm is a fantastic example of an innovation that helps drive efficiency with our businesses and our customers."

In accepting the Edelman Award, Chou emphasized not only the company-wide effort in developing elegant technical solutions to incredibly complex problems, but also the buy-in and cooperation of managers and C-level executives and the wisdom and insight of the award-winning team to engage and share their vision with those managers and executives. "For some of you who have not been a part of a very large organization like HP, this might sound strange, but it required tenacity and skill to bring about major changes in the processes of a company of HP's size," Chou said. "In many of our business [units], project managers took the tools and turned them into new processes and programs that fundamentally changed the way HP manages its product portfolios and bridged the organizational divide."

Questions for the End-of-Chapter Application Case

1. Describe the problem that a large company such as HP might face in offering many product lines and options.

2. Why is there a possible conflict between marketing and operations?

3. Summarize your understanding of the models and the algorithms.

4. Perform an online search to find more details of the algorithms.

5. Why would there be a need for such a system in an organization?

6. What benefits did HP derive from implementation of the models?

Source: Adapted with permission, P. Horner, "Less Is More for HP," *ORMS Today,* Vol. 36, No. 3, June 2009, pp. 40–44.

References

Aleman, D. M., T. G. Wibisono, and B. Schwartz. (2011). "A Nonhomogeneous Agent-Based Simulation Approach to Modeling the Spread of Disease in a Pandemic Outbreak." *Interfaces*, Vol. 41, No. 3, pp. 301–315.

Alfredo, D. M. G., E. N. R. David, M. Cristian, and Z. V. G. Andres. (2011, May/June). "Quantitative Methods for a New Configuration of Territorial Units in a Chilean Government Agency Tender Process." *Interfaces,* Vol. 41, No. 3, pp. 263–277.

Allwood, J. M., and J. H. Lee. (2005). "The Design of an Agent for Modelling Supply Chain Network Dynamics." *International Journal of Production Research*, Vol. 43, No. 22, pp. 4875–4898.

Angerhofer, B. J., and M. C. Angelides. (2000, Winter). "System Dynamics Modeling in Supply Chain Management: Research Review." IEEE, *Simulation Conference, 2000,* Vol. 1, pp. 342–351.

Baker, B. M., and M. A. Syechew. (2003). "A Genetic Algorithm for the Vehicle Routing Problem." *Computers and Operations Research*, Vol. 30, No. 5, pp. 787–800.

Banks, J., and R. R. Gibson. (2009). "Seven Sins of Simulation Practice." *INFORMS Analytics*, pp. 24–27. **www.analytics-magazine.org/summer-2009/193-strategic-problems-modeling-the-market-space** (accessed February 2013).

Bates, D. W., Cullen, D. J., Laird, N., Petersen, L. A., Small, S. D., Servi, D., . . . & Edmondson, A. (1995). Incidence of adverse drug events and potential adverse drug events. JAMA: the journal of the American Medical Association, 274(1), 29–34.

Caridi, M., and S. Cavalieri. (2004). "Multi-Agent Systems in Production Planning and Control: An Overview." *Production Planning & Control*, Vol. 15, No. 2, pp. 106–118.

Carley, K. M., et al. (2004). "BioWar: A City-Scale Multi-Agent Network Model of Weaponized Biological Attacks." **http://handle.dtic.mil/100.2/ADA459122.**

Chongwatpol, J., and R. Sharda. (2013). "RFID-Enabled Track and Traceability in Job-Shop Scheduling Environment." *European Journal of Operational Research*.

Classen, D., M. Pestotnik, R. Evans, J. Lloyd, and J. Burke. (1997). "Adverse Drug Events in Hospitalized Patients: Excessive Length of Stay, Extra Cost, and Attributable Mortality." *The Journal of the American Medical Association*, Vol. 277, No. 4, pp. 301–311.

Crooks, A. T. (2010). "Constructing and Implementing an Agent-Based Model of Residential Segregation Through Vector GIS." *International Journal of Geographical Information Science,* Vol. 24, No. 5, pp. 661–675.

Dardan, S., et al. (2006). "An Application of the Learning Curve and the Nonconstant-Growth Dividend Model: IT Investment Valuations at Intel Corporation." *Decision Support Systems*, Vol. 41, No. 4, pp. 688–697.

Garrido, Anderson J. (2002). "Evaluation in Health Informatics: Computer Simulation." *Computers in Biology and Medicine*, Vol. 32, No. 3, pp. 151–164.

Garrido, T., L. Jamieson, Y. Zhou, A. Wiesenthal, and L. Liang. (2005). "Effect of Electronic Health Records in Ambulatory Care: Retrospective, Serial, Cross-Sectional study." *Information in Practice, BJM*, Vol. 330, No. 7491, pp. 1–5.

Godlewski, E., G. Lee, and K. Cooper. (2012). "System Dynamics Transforms Fluor Project and Change Management." *Interfaces*, Vol. 42, No. 1, pp. 17–32.

Grupe, F. H., and S. Jooste. (2004, March). "Genetic Algorithms: A Business Perspective." *Information Management and Computer Security*, Vol. 12, No. 3, pp. 288–297.

Horner, P. (2009, June). "Less Is More for HP." *ORMS Today*, Vol. 36, No. 3, pp. 40–44.

Hutton, D. W., M. L. Brandeau, and S. K. So. (2011). "Doing Good with Good OR: Supporting Cost-Effective Hepatitis B Interventions." *Interfaces*, Vol. 41, No. 3, pp. 289–300.

Kasiri, N., R. Sharda, and D. Asamoah. (2012, June). "Evaluating Electronic Health Record Systems: A System Dynamics Simulation." *SIMULATION,* Vol. 88, No. 6, pp. 639–648.

Leape, L., D. Bates, D. Cullen, J. Cooper, H. Demonaco, T. Gallivan, R. Hallisey, J. Ives, N. Laird, G. Laffel, R. Nemeskal, L. Petersen, K. Porter, D. Servi, B. Shea, S. Small, B. Sweitzer, C. M. Macal, and M. J. North. (2005). "Tutorial on Agent-Based Modeling and Simulation." *Proceedings of the 37th Conference on Winter Simulation*. Orlando, Florida, Winter Simulation Conference, pp. 2–15.

Marquez, A. C., and C. Blanchar. (2006). "A Decision Support System for Evaluating Operations Investments in High-Technology Business." *Decision Support Systems*, Vol. 41, No. 2, pp. 472–487.

Mattila, V., K. Virtanen, and T. Raivio. (2008, May/June). "Improving Maintenance Decision Making in the Finnish Air Force Through Simulation." *Interfaces*, Vol. 38, No. 3, pp. 187–201.

McGowan, J., C. Cusack, and E. Poon. (2008). "Formative Evaluation: A Critical Component in EHR Implementation." *Journal of the American Medical Informatics Association*, Vol. 15, No. 3, pp. 297–301.

Nick, Z., and P. Themis. (2001). "Web Search Using a Genetic Algorithm." *IEEE Internet Computing*, Vol. 5, No. 2.

Railsback, S., et al. (2006). "Agent-Based Simulation Platforms: Review and Development Recommendations." *Simulation,* Vol. 82, No. 9, pp. 609–623.

Shin, K., and Y. Lee. (2002). "A Genetic Algorithm Application in Bankruptcy Prediction Modeling." *Expert Systems with Applications*, Vol. 23, No. 3.

Sidorov, J. (2006). "It Ain't Necessarily So: The Electronic Health Record and the Unlikely Prospect of Reducing Health Care Costs." *Health Affairs*, Vol. 25, No. 4, pp. 1079–1085.

Thompson, B., and M. Vliet. (1995). "Systems Analysis of Adverse Drug Events." *The Journal of the American Medical Association,* Vol. 274, No. 1, pp. 35–43.

Thompson, D. I., J. Osheroff, D. Classen, and D. F. Sittig. (2007). "A Review of Methods to Estimate the Benefits of Electronic Medical Records in Hospitals and the Need for a National Benefits Database." *Healthcare Information and Management Systems Society (HIMSS)*, Vol. 21, No. 1, pp. 62–68.

Walbridge, C. T. (1989, June). "Genetic Algorithms: What Computers Can Learn from Darwin." *Technology Review (USA),* Vol. 92, No. 1.

Xiaoling, S., J. Kaur, S. Milojević, A. Flammini, and F. Menczer. (2013). "Social Dynamics of Science." *Scientific Reports,* Vol. 3, p. 1069.