PEARSON NEW INTERNATIONAL EDITION

Civil and Environmental Systems Engineering

Revelle Whitlatch Wright

Second Edition

Pearson New International Edition

Civil and Environmental
Systems Engineering
Revelle Whitlatch Wright
Second Edition

PEARSON®

Pearson Education Limited
Edinburgh Gate
Harlow
Essex CM20 2JE
England and Associated Companies throughout the world

Visit us on the World Wide Web at: www.pearsoned.co.uk

PEARSON®

ISBN 10: 1-292-02708-8
ISBN 13: 978-1-292-02708-1

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Printed in the United States of America

Table of Contents

Explaining Systems Analysis

A INTRODUCTION: BUILDING MODELS

This chapter is about building mathematical models that assist in the design or management of natural or constructed systems. These models may also assist in the development of policy relative to these systems. **Mathematical models** may not be a familiar term, but virtually all people have experience with other types of models.

As children, many will have built and flown paper airplanes. Some will have built model airplanes of balsa or plastic. Many will have built model villages in school exercises. In the adult world, architects build scale models to see how proposed buildings fit within their intended environments. Engineers build model cars and test them in wind tunnels for drag resistance in order to develop fuel-efficient cars. Such physical models that mimic their larger and real cousins are termed **iconic models**.

In addition to physical models, most of us, especially when we were younger, *participated* in models of human systems. These models or simulations gave us the opportunity to experiment with other roles. You may well have played cops and robbers, cowboys and Indians, space invaders, house, doctor/hospital, circus, or any of a number of other games that allowed the simulation of other environments. Although Monopoly™ is entertainment, it tested financial acumen in an environment with much randomness. Checkers and chess helped to develop our spatial intuition and our ability to foresee the consequences of our actions. Military forces regularly conduct "war games" to test the readiness of their troops in realistic situations. Fire drills are used to simulate the conditions and situations that may occur during fire episodes. Pilots may test their flying skills in a "flight simulator." Iconic and role-playing (participatory) models pervade society and enrich it with the insights they provide.

Scientists have been using models as well; they have been building mathematical models at least since the time of Newton in the late seventeenth century. By the late 1800s and early 1900s, physicists, engineers, chemists, biologists, mathematicians, and statisticians were all actively building such quantitative models—to assist them in their understanding of atomic, mechanical, chemical, and biological systems. In the last half of the twentieth century, economists joined in the model building movement—in an attempt to predict future economic conditions.

For the most part, these models from the various disciplines were built using differential and difference equations. Such models are still regularly being built throughout the sciences, social sciences, and engineering as a means to explain, to comprehend, and to predict natural phenomena. These mathematical representations are collectively called **descriptive models** because they offer, for a given set of inputs and initial conditions, a description of the outputs through time of the phenomena under study. For instance, a common model in biology predicts the population of a species, given initial population and growth parameters.

Since World War II, however, two important developments have transformed the world of mathematical modeling. The first development was the creation of a *new* mathematics, a mathematics that focused on the science of decision making and policy development. The second development was the invention and continual improvement of the digital computer—a development made possible by the silicon chip. The invention and subsequent development of the computer have dramatically extended the power of descriptive models. Such models are now far more capable of mimicking natural phenomena successfully—even on a global scale.

The invention of a mathematics of decision making, on the other hand, has opened avenues of research not previously thought possible. Although this new mathematics was originally applied only to small problems, much as descriptive models initially were applied, the rapid evolution of the computer has now made possible the study and consideration of enormously large problems. These problems have gone far beyond the limits imagined by those who originated the mathematics of decision making. Thus, the computer greatly facilitated the application and development of both descriptive and decision-making mathematics.

We called the first type of mathematical representation a **descriptive model** because it describes. In contrast, the representation that uses the mathematics of decision making is called a **prescriptive model** because it prescribes a course of action, a design, or a policy. The *descriptive model* is said to answer the question, "*If I follow this course of action, what will happen?*" In contrast, the *prescriptive model* may be said to answer the question, "*What is the best course of action that I might follow?*" For a particular strategy that was specified in advance, the descriptive model predicts the quantitative outcomes, possibly through time. The prescriptive model, on the other hand, finds and suggests the best strategy to choose out of the universe of all possible strategies. Implied in the question of what strategy to select is likely to be some notion of cost or of effectiveness. That is, the course of action derived by the decision model is chosen to be the least costly or the most effective or the most cost-effective.

Another term used for a prescriptive model is to call it an **optimizing** or **optimization model**, in the sense that the policy or design that is selected achieves the *best* value of some objective. This chapter will focus on prescriptive models, but it is often true that descriptive models may be contained within prescriptive models. These descriptive models are sometimes so simple that a reader may not even realize that a descriptive model is being used. Often, the descriptive model may consist of the simple assumption that the amount of a particular resource consumed is directly proportional to the number of items manufactured of a given type.

The descriptive/prescriptive classifications are just one of several ways we can divide types of mathematical models. Another realistic way we can divide model types is by the kind of data that they utilize. Some models utilize data that are considered to be known with relative certainty. An example might be the number of table tops of a given size that can be cut from a 4 ft. × 8 ft. sheet of plywood. Except for very occasional cutting errors or flaws in the plywood, that number is fixed and known. Another example is the operation through time of a materials stockpile, say heating oil or even beer or grain. In such a situation, the size of a given month's demand for the product is fairly predictable year to year. Models of this type, in which data elements are not thought of as variable but are relatively fixed and predictable quantities, are referred to as **deterministic models**.

In deterministic models, parameter values are determined and known at the outset. Given the initial contents of the stockpile and a specified release of materials and a stated purchase or manufacture of new materials during a unit of time, a deterministic model suggests that there is just one possibility for the final, end-of-period condition of the stockpile. That is, only a single outcome can occur from a month's events given the choice of action (see Figure 1a). The stockpile's contents is precisely the sum of the initial storage plus new purchases less the stated release.

In contrast to deterministic models, other models might utilize data elements that are not precisely known but can be characterized by a mean and some random variation about the mean. The September inflow to a reservoir might fit in this category. September is part of hurricane season in the eastern United States. During some Septembers, hurricanes may cross the Northeastern states and will produce very large rainfalls and runoff. In other Septembers, when no hurricane tracks across the Northeast, little rain may occur and inflows to reservoirs may be quite low. No one really knows in advance whether hurricanes will cross a particular geographic area in September, so reservoir inflows cannot really be precisely predicted.

Models in which the data elements are random or variable—capable of taking on any value from a range of values—are called **stochastic models**. Given an initial value of the storage in the reservoir, and a known amount of release for water supply,

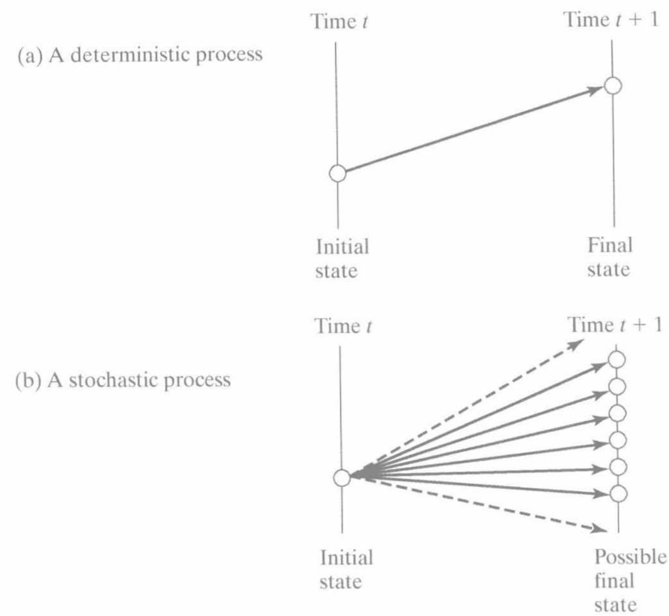

Figure 1 Deterministic and Stochastic Processes

the stochastic model suggests that the end-of-month contents of the reservoir can be stated, but with some uncertainty. This is because of the random inflow to the reservoir—an inflow that cannot be predicted but will fall within some range of possible inflows. Many end-of-period values of storage are possible and final storages will be in some range of outcomes. (See Figure 1b.) These two additional model classifications, deterministic and stochastic, allow us to classify models into four basic types; these are summarized in Table 1, discussed more fully shortly.

Positioned in concept somewhere between the deterministic and stochastic model is a **statistical model**. In a statistical model, system inputs have been observed or recorded, and system outputs have been measured. The relationship, however,

TABLE 1 TYPES OF MODELS BASED ON TWO-WAY CLASSIFICATION

	Deterministic	Stochastic
Prescriptive	Linear programming Integer programming Multiobjective programming Dynamic programming	Stochastic programming
Descriptive	Difference equations Differential equations	Stochastic differential equations Queueing theory Monte Carlo simulation

between systems inputs and outputs does not seem to be consistent. As an example, nitrogen-containing fertilizer may be applied to a field at the beginning of the growing season for a corn crop. The additional yield of corn from an increment of fertilizer may prove to be different in different years, perhaps because of rainfall, and perhaps because of different soil characteristics of fields. Data may be gathered from many experiments, and the yield of corn per hectare may be plotted against kilograms of nitrogen applied per hectare. The data do not appear to fall on a straight line or even a smooth curve, but are scattered above and below the line that one might draw to approximately fit the curve.

A statistical model is a hypothesis of the relationship between output (corn yield) and input (fertilizer applied). The model may suggest that the relationship is linear or nonlinear. A statistical model may be thought of as neither deterministic nor stochastic, but a model that provides the most likely or expected outcome of conditions (yield) given the input (fertilizer) and uncertain events (rainfall).

The intersections of these two sets of categories—deterministic/stochastic and prescriptive/descriptive—give rise to a four-component classification table that contains (except for statistical models) nearly all major model types that are utilized today. Referring to Table 1, models that are deterministic and descriptive are typically differential equation models or models that use difference equations. These are models that a scientist or engineer probably encountered in a calculus or applied mathematics course. The models typically incorporate empirically derived parameters, rate constants, and known (or at least assumed) initial conditions. These models may be linear or nonlinear, depending on the nature of the system or on the degree of realism needed for the model structure in the particular application.

The intersection of descriptive and stochastic models contains several types of models. One is the differential equation/difference equation model coupled with parameters that are random variables. Such equations are called stochastic differential equations—a form of mathematics that becomes exceedingly complex when the equation(s) contain more than one random parameter. As soon as two random parameters are introduced, the structure of the correlation between the parameters also needs to be known in order to create the range of model outcomes. As an example, July streamflows may be a function of both July rainfall and temperature—but the rainfall and temperature are themselves interrelated, lower rainfalls being associated with higher temperatures.

A second model form at the intersection of descriptive and stochastic models is embodied in the mathematics of queueing theory or, more generally, the field known as stochastic processes. The mathematics of stochastic processes presume known parameters that describe arrivals and departures in a random environment. A third model type at this intersection is known as **simulation**, a computer-intensive form of modeling that generates realistic events and system responses through time. Here, the statistics of the events and the responses are designed to correspond to the actual statistics of parameters in the system being studied. As an example, the backup of cars at a toll plaza is modeled using the rate of arrival of vehicles at a toll booth and rate of collecting tolls. The toll collection process might be modeled to observe the impact or influence of different numbers of toll collectors and of automated toll booths on the number of cars that get backed up at the plaza. All three model types that describe systems responding to random variation—differential equation models, stochastic process models, and simulation models—allow the modeler to observe the range of possible outputs that might evolve through time from different sets of initial conditions and control actions.

Another intersection in the two-way table is that of prescriptive models with deterministic models. These are the **deterministic optimization** models and are known by various names. The names of the models depend on whether the mathematical descriptions are linear or nonlinear; they may depend on whether the models are static or evolve through time; and they depend on whether the variables are restricted to integers or are continuous. One of the model names is **linear programming** (LP), the form of optimization that is arguably the most popular form of optimization. Linear programming models have a linear or linearizable objective and linear constraints. Exact algorithms, solution procedures that are iterative in nature and which find global optimal solutions, exist for linear programming problems. Furthermore, computer software that implements the algorithms is widely available to solve even very large linear programming problems.

Other model names and forms of optimization include quadratic programming, gradient methods, optimal control theory, dynamic programming, multi-objective programming, and integer programming. These names reflect either the mathematical forms being used or are descriptive of the algorithm or the setting.

Quadratic programming deals with problems having a quadratic objective function. **Gradient methods** direct computations to follow slopes of objective functions to locally optimal or optimal solutions. **Optimal control theory** finds an optimal control or decision function in time or an optimal trajectory to achieve some goal. **Dynamic programming** typically considers problems with a number of time stages. **Multiobjective programming** operates on problems with more than one objective and derives tradeoffs between those objectives. **Integer programming** considers only integer-valued decisions as practical or desirable.

The reader will be interested in the meaning of the word programming in this context. The body or collection of all optimization methods is known as **mathematical programming**, and its subspecialties are known as linear programming, dynamic programming, integer programming, and so on. It is common for the student without familiarity with these methods to presume that the term *programming*, which is used to describe optimization models and methods, is associated with and means computer programming. It does not, however, refer to the use of the computer. The "programming" in mathematical programming is a term that means *scheduling*, *the setting of an agenda*, or *the creation of a plan of activities*. Some confusion seems inevitable, however, because virtually all optimization, except that done as learning exercises for homework, classes, or labs, requires extensive use of digital computation.

The remaining intersection of model categories in the four-component table is that of prescriptive and stochastic models. There is a form of optimization that deals with models whose parameters are random variables; that is, the parameters can take on any of a number of values or any value in a range of values simply by chance. The form of optimization is known as stochastic programming, and it requires of the student a relatively strong background in probability theory. The importance of stochastic programming to applied studies is growing, but it is also probably one of the most challenging forms of mathematical programming and is the province of specialists.

Of the various forms of programming or optimization, the most widely used in practice is linear programming. The wide application of linear programming is not simply a matter of the existence of an efficient and exact method of solution—although an efficient and exact solution method is available. No matter how efficient a solution procedure may be, a methodology for a form of optimization that does not match the needs of real problem settings would not be expected to be utilized widely. It follows that the wide use of linear programming is not driven by the availability of a solution method so much as it is driven by the form of the linear program. A linear programming problem statement has a widely applicable and universally appealing structure. Its form places an *objective* or goal alongside *constraints*. The objective is the element to be optimized—perhaps it is cost that is to be minimized or profit that is to be maximized. Constraints are conditions that any and every solution must satisfy—for instance, resources cannot be exceeded.

Many objectives are possible for a linear programming problem. These include, but are not limited to, minimum cost, maximum production, maximum equity, maximum access, minimum waiting time, minimum waste, and maximum profit. Constraints, on the other hand, are natural limits on achievement or imposed limits on resources use. The most commonly constrained quantities are resources such as personnel, vehicles, level of investment, time, and materials. Other constraints that might be used in integer programming problems enforce the logic of system development. For instance, a particular nearby link of a road network must be built before some other more remote link can be built. Finally, some constraints provide only definitions.

Remarkably, the language of objectives and constraints of the linear programming problem turns out to be the language of real problem statements. Practical problems are often stated in precisely this format of objective and constraints. These problem statements are frequently offered by people who have absolutely no training in modeling or in systems engineering or in optimization. It is an absolutely striking and unforgettable phenomenon to find people without any systems training describing their problems in the language of the linear programming problem. It is the naturalness of the linear programming problem statement that accounts for the widespread appeal of this form of optimization. Some other forms of optimization also have this structure, but no methodology that uses this structure is more versatile and more available than is linear programming.

B HISTORY OF SYSTEMS AND OPTIMIZATION

Two great events punctuate the history of applied mathematics. The first step, the invention of calculus, occurred in the seventeenth century. Sir Isaac Newton was one of two inventors of the calculus. Although Newton in Britain was the first to create the calculus in 1665–1666, Baron von Gottfried Wilhelm Liebniz independently invented the calculus in 1675. In that era, publication of ideas was often long delayed. Liebniz published his calculus in 1684, nine years after he conceived the idea. Newton, in order to retain claim to his invention, rushed into print by 1687. The world of mathematics was never the same.

Interestingly, the stimulus for Newton's invention of the calculus was not a consideration of abstract mathematical issues. Instead, Newton was interested in explaining the effects of the planets on one another, and in the process of this quest, he needed to create the calculus. That is, the calculus was invented to solve a general problem that Newton was considering. In the same way, the invention of linear programming was propelled by the necessity of solving real problems.

Almost three centuries later, another event shook and reoriented not only the world of mathematics, but also the fields of economics and engineering. The invention of linear programming was to influence not only economics but would form the core of an entirely new discipline, operations research or systems engineering. In the same way that the calculus can be traced to two central figures, the development of linear programming is attributed to several towering people.

At about the same time, Koopmans in the United Kingdom and Kantorovich in the former U.S.S.R. independently attacked the problem of least-cost distribution of items. Kantorovich's work (1939) was suppressed for more than twenty years by Soviet authorities. Koopmans came to the United States where he encountered a young George Dantzig who had just created an algorithm, a set of semi-automated mathematical steps, to solve linear programming problems. Dantzig's algorithm provided a practical method of solution to the problem Koopmans had been studying. Dantzig invented the simplex procedure for solving linear programming problems in 1947 as part of a U.S. Air Force research project. His procedure, with modifications and enhancements to take advantage of modern computers, is in wide use today.

In the period 1948–1952, Charnes and his coworkers pioneered industrial applications of linear programming and created the **simplex tableau**—the special tabular data storage methodology used in the repeated calculations of the simplex procedure. Charnes and coworkers went on to adapt linear programming to deal with convex rather than linear functions, to invent goal programming, and to create new forms of optimization to deal with problems that operated with random parameters. Charnes did not stop at industrial applications of linear programming. With students and coworkers he pushed on to the first applications of linear programming to civil and environmental engineering. Dantzig went on to make major contributions to the solution of network and logistic problems. Koopmans and Kantorovich received the 1975 Nobel Prize in Economics for their work in linear programming. The magnificent achievements of Dantzig and Charnes have not been honored in such a way.

There is more to the history of systems, though, than the development of the mathematics and its application in new settings. Dantzig tells us that the calculations and data manipulations that were needed for the first application of the simplex procedure were so extensive and voluminous that the application was carried out on a large tablecloth. (Of course, the computers of the time were too primitive for such calculations.) This calculation procedure was carried out for a linear problem of a size that the reader can do by hand. In fact, the problem was minute by today's standard. Problems solved today may have dimensions that are more than four orders of magnitude (10,000 times) larger than those first linear programming problems. The difference, of course, that makes the solution of such large problems possible is the appearance and explosive evolution of the computer. From the 1950s onward, computers have advanced in speed and power, making possible the solution of larger and larger linear programming problems.

Whereas a careful person may solve—by hand calculations alone—a problem with perhaps ten variables and five constraints, modern codes on up-to-date personal computers can easily handle problems with more than 20,000 variables and 5,000 constraints. With continual advances in computer technology, even these numbers will soon be far surpassed. We can say with complete assurance that linear programming would today only be a fascinating but small branch of applied mathematics and economics if it were not for the co-development in time of the electronic computer. With the development and evolution of the computer, linear programming has become the foremost mathematical tool of operations research, of management science, industrial engineering, and engineering management.

Up to now, we have been using the terms *systems* and *systems analysis* without any sort of definition, although we have attempted to operate with an intuitive feel for the terms. Before going on to examples of systems analysis applications, it is appropriate to offer some definition of the term systems analysis, so that the examples can be viewed in the context of our definition. Of course, the easiest definition can be offered by re-interpreting the words **systems analysis** so that we get the phrase "an analysis of systems." By this we mean that we are investigating the behavior of a system, typically by choosing various options for the control or management of the system.

For example, we might be investigating a river system, and we would try various levels of pollution control at cities along the river to see what levels of water quality result. From this study, we can enumerate the various measures of control that achieve some particular desired level of water quality. Further, from this list of many choices for control, all of which achieve the desired level of water quality, we can select that single set of choices for pollution control that yields the desired water quality at the least system-wide cost. So *systems analysis* implies the organized study of alternatives and options for the management or design of a system.

The alternatives in a systems analysis can be generated by almost any of the model types we have discussed above—from simulation with many different sets of control options in place, from solution of differential equations with an exploration of many possible management strategies, etc. With this definition in mind, we now need to explain how optimization methods such as linear programming fit within the idea of systems analysis, although the answer may be nearly evident by now. Optimization via linear programming arrives at the "best" mathematical alternative by iteration. The iterations automatically consider *only* the control options that achieve the required outcome, in terms of, let us say, constraints on water quality. However, the iterations are internal to the process, invisible in a sense because the options examined along the way, in the iteration process, are usually never seen by the analyst. The optimization process provides only the final and best strategy to the analyst for further study and consideration. Hence, we may think of optimization as an *automated form of systems analysis* in which many many examined strategies are implicitly (internally) generated and investigated. Typically, only one of those many examined strategies will be presented by the optimization code to the analyst for further study. Optimization may be thought of as systems analysis automated; it may be considered a "power" systems analysis, in contrast to the older style of explicit exploration of numerous feasible alternatives. The preceding description, while accurate, may not yet be fully clear; clarity should come when the simplex solution procedure of linear programming is ultimately understood.

We next provide a description of typical and important members of the broad array of applications of our automated form of systems analysis—of optimization applications.

C APPLICATIONS OF LINEAR PROGRAMMING

To provide you with an idea of how widely utilized linear programming and its derivative types of optimization are, we describe a number of settings in the public sector, in industry, and in business where linear programming and allied methods have been put to use.

C.1 Distribution, Warehousing, and Industrial Siting

From the beginning, distribution has played a key role in the development of linear programming. It was the problem of least-cost distribution of goods from multiple sources to multiple destinations that motivated both Koopmans and Kantorovich to structure their linear programs; these problems are known today as **transportation problems**, and very large problems are solved routinely, even on desktop computers. Transportation problems assume that direct and separate shipments are made over known routes. Another class of problems, known as **delivery** or **routing problems**, also move goods from multiple origins to multiple destinations. However, the challenge of delivery and routing problems is to find the tours or routes for vehicles that will drop off the needed amounts at multiple destinations as the vehicles traverse the prescribed route. These problems can be structured as linear integer programs.

Warehouse problems, another class of problem that has been solved by linear programming, examine the optimal stocking of goods and suggest the release quantities of those goods through time to the distribution system. The siting of warehouses between factories and markets has also been studied with linear integer programming, as has the siting of manufacturing plants, which may supply either warehouses or customers.

C.2 Solid Waste Management

Within the environmental area, linear programming and allied methods have been used to site landfills and the stations at which small trucks transfer their loads to larger trucks. As well, these methods have been utilized to outline solid waste collection districts. These techniques have also been directed at the routing of solid waste collection vehicles through street networks and, within the framework of hazardous waste management, the routing of spent nuclear fuel from power plants to storage sites.

C.3 Manufacturing, Refining, and Processing

Some of the earliest applications of linear programming took place in these areas. The problem Koopmans called **activity analysis** consists of choosing which items of many to manufacture in order to achieve either least cost or maximum profit. In these problems, constraints limit the total amount of each of various resources that would be consumed in the manufacturing process. Another manufacturing area to which linear programming and allied methods have been applied is the design of factory floors. Known as the **facility layout problem**, this model sites the various activities on the factory floor to minimize interaction costs.

The operation of a refinery, especially the blending of aviation fuel, was studied early in the history of linear programming—in this case by Charnes and coworkers. Chemical process design remains a fertile area to this day for the application of linear programming.

C.4 Educational Systems

Educational systems are a rich setting for the application of systems methodology. Linear programming and allied procedures have been applied to class scheduling and room scheduling. These methods have also been used in school bus routing and to draw school district boundaries for efficient transportation and efficient utilization of school capacity. LP has also been utilized to allocate pupils to schools to achieve mandated desegregation plans. LP models have also been used for enrollment planning at colleges.

C.5 Personnel Scheduling and Assignment

Systems techniques have found application in the scheduling of personnel through shift rotations. They have also been used to assign people to jobs or tasks in large organizations. The scheduling and assignment of airline crews to flight legs is an ongoing and important use of linear integer programming. Manpower planning models have also been developed using linear programming to project needs and policies relative to such areas as physician and nurse availability. The efficient assignment of crews to snow plows for winter highway maintenance has also been structured as a linear programming problem.

C.6 Emergency Systems

Since the late 1960s, linear programming and allied methods have been applied to the siting of fire engines, fire trucks, fire stations, and ambulances. Early problem statements focused on minimizing average travel time, given budget constraints. Later formulations required or sought "coverage," the stationing of at least one vehicle within a travel time or distance standard of every point of demand. Cost was either a constraint or an objective in these models. Most recently, congestion in emergency systems has been investigated with the emphasis on ensuring the actual availability of a server within the time standard at the moment of a call. Dozens of linear programming models have been built in the area of emergency facility siting. Other areas of siting have also been investigated, and these are referred to in Section C.1, "Distribution, Warehousing, and Industrial Siting."

C.7 The Transportation Sector

Linear programming or variants have been used extensively in the design of transportation networks including highway networks, rail networks, and airline networks. Efficiency and cost objectives have been utilized in such formulations with constraints on connectivity (or continuity) of the network or on population proximity to the network. LP and integer programming have also been used to design bus routes, assign drivers to buses, schedule buses, and choose bus stop locations. Traffic light timing at intersections and at freeway entrance ramps have also been approached as linear programming problems.

Goods movement, as mentioned in Section C.1, "Distribution, Warehousing, and Industrial Siting," is a classic application of linear programming. Empty railcar movement has also been structured as a linear programming problem, as has the selection of freight terminals to open or close and the specification of hubs in an airline network. The development of pipeline networks for oil and natural gas can also be structured as a linear programming problem. Military applications of linear programming are often of a goods movement/logistics nature. The vertical alignment or grade design of highways, as well as the determination of optimal cut-and-fill strategies, can be cast as linear programming problems.