Stochastic Optimization

Operations research has been particularly successful in two areas of decision analysis: (i) optimization of problems involving many variables when the outcome of the decisions can be predicted with certainty, and (ii) the analysis of situations involving a few variables when the outcome of the decisions cannot be predicted with certainty. The earlier chapters of this book have identified this dichotomy as deterministic optimization versus stochastic analysis. The optimization of problems in which the outcomes are uncertain, the subject of this chapter, is still in its infancy so there is still much room for improvement in computational methods.

We have already considered optimization and uncertainty in several places. A sequential process in which decisions yield uncertain results admits a stochastic dynamic programming model as in Section 20.6. Linear programming was used in Chapter 24 to find the optimal mixed strategy in a zero-sum game. The approach to optimization for most stochastic processes is explicit enumeration as in the queueing system decision models of Section 17.5.

In this chapter, we consider two decision models that explicitly incorporate the probability distributions of random variables. Both are solved with linear programming (LP). Although we will see that rather simple situations lead to large LP models, the general availability of efficient LP algorithms may make the solution of such problems practical.

27.1 Decisions with Uncertainty

There are many situations in which optimization models are used that could be more realistically modeled with the explicit incorporation of random variables. For example consider a problem in water resources planning. With appropriate assumptions the decisions associated with a river basin can be represented with a large multiperiod network model. The decision variables are the amounts of water to release into the channels and canals of the river system in each of a number of time periods. The periods are interconnected by variables representing storage in reservoirs. The parameters of the model are the rainfall amounts at various times and locations and the demand for water in the cities and industries of the river basin. The estimates of rainfall may come from historical records and demands may be based on economic projections. Solving such a model with deterministic optimization provides planners with information about the capabilities of the system for meeting demands. By examining various alternatives for basin development, the model aids in the design process. The model is an abstraction, however, because the future rainfalls and demands are obviously unknown. The model would be more accurate and the decisions determined through optimization would be more reasonable if the characteristic of uncertainty were explicitly incorporated.

Deterministic Solutions

The optimization algorithm determines the *best* solution given the parameters of the model. The modeling and decision analysis process is illustrated in Fig. 1 where the shape labeled *situation* represents the real problem under consideration. Various assumptions and abstractions are applied, including the assumption of deterministic information, to obtain a mathematical *model*. The model is input to a computer where an *algorithm* determines the optimal *decision*, represented by the vector **x**. We use the vector notation to indicate that the decision generally has many dimensions.

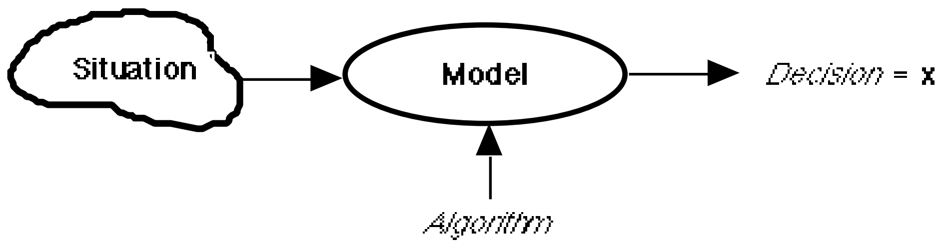

Figure 1. The deterministic approach to decision making

The fault with this approach is that the decision **x** is optimum for the model and not the situation. It is usually readily apparent to the person charged with the task of implementing the decision, that **x** is not at all appropriate for application to the situation. The primary reason for this lies in the assumption of deterministic parameters. When the situation involves uncertainty or risk, the kind of decision taken is quite different than if it does not. Real decision makers hedge against various possible futures. For the water planning example, the person releasing water from a reservoir rarely makes decisions for a known future, but hopes to provide the flexibility required to adapt to a variety of futures. The algorithm operating on a deterministic model is not troubled by doubt, but gives the *best* decision for the information given.

The Scenario Model

A more accurate description of the decision process is shown in Fig. 2, where the situation gives rise to a model of the present with the decisions that must be made immediately represented by the vector **x**. Models are also provided for several alternative futures, or scenarios. The parameters for the futures are each deterministic but they differ from each other. For the water planning example, the present decisions are the amounts to release through the various dams in the system. The models for the scenarios describe various possibilities for water supplies and demands that will occur in the future. Associated with each scenario is a decision vector, $\mathbf{y}_k$ for future k. This model recognizes that future decisions depend on the decisions made today. They must be included explicitly because they affect the evaluation of the present decision, **x**. Perhaps probabilities can be attached to each scenario if sufficient statistical data is available. The figure shows p_k as the probability of future k.

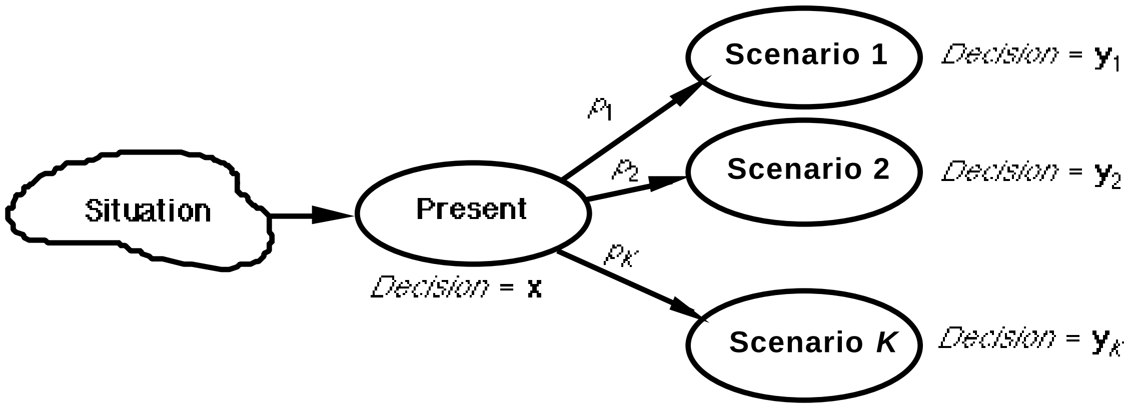

Figure 2. Decision problem with several possible futures

The What-if Approach

When the deterministic assumption is obviously not appropriate, analysts or managers typically turn to a *what if* approach for arriving at a solution. The *what if* approach is illustrated in Fig. 3. The analysis proceeds by considering each scenario in turn and finding the best present and future decisions for each case. For example, given scenario 1, the optimal present decision is $\mathbf{x}_1$ and the optimal future decision given $\mathbf{x}_1$ is $\mathbf{y}_1$. After the *what if* analysis, we have the best decision for each of the possible scenarios, $\mathbf{x}_1$, $\mathbf{x}_2, \ldots, \mathbf{x}_K$. How should the scenario decisions be combined to obtain one decision $\mathbf{x}$ that is *best* for all possible futures? One might suggest averaging the K scenario decisions using the probabilities p_k, but the average solution is often infeasible and rarely optimal.

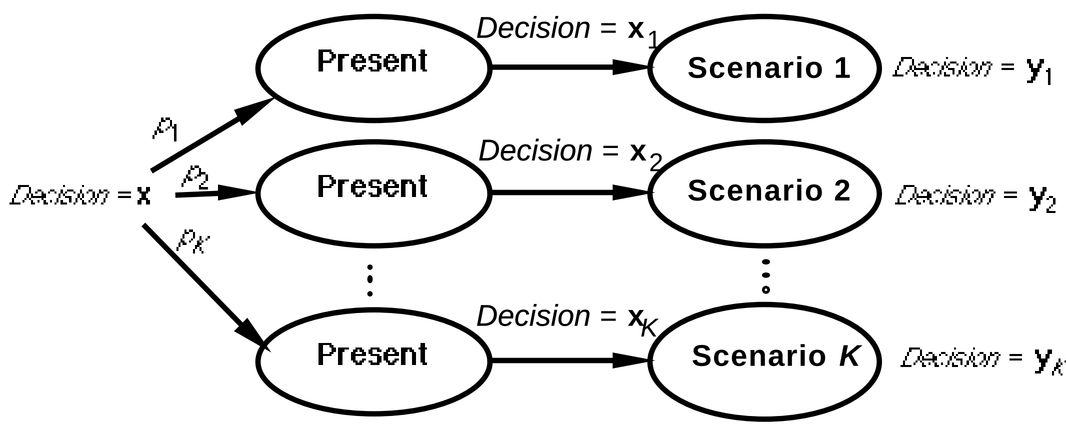

Figure 3. The *what if* analysis of alternative futures

Example 1

Consider a town with a demand for 10 units of water. The town receives its water from a river authority at no cost. The authority has natural water in the amount b, but it can also purchase water from a neighboring state for \$5 per unit. If city does not receive 10 units of water, there is a penalty cost for demand not met. The cost is \$2 per unit for the first five units and \$4 per unit for the next five units. Any amount greater than 10 units of water provided to the town has no value. If the authority has excess water, it is released to down stream users for a benefit of \$1 per unit.

The network model for the situation is shown in Fig. 4. The brackets associated with the nodes indicate the flow that must enter or leave. The [+*b*] on the authority node indicates that *b* will enter at the authority. The [–10] at the town node indicates that 10 units will leave at the town. The upper and lower bound of *x* on the arc from authority to town shows the amount allocated. The arcs from the source model the cost of shortages at the town and the availability of purchased water for the authority. The arcs into the sink carry the amount of excess provided to the town and the releases down stream from the authority. The source node can provide any amount of flow and the sink node can absorb any amount.

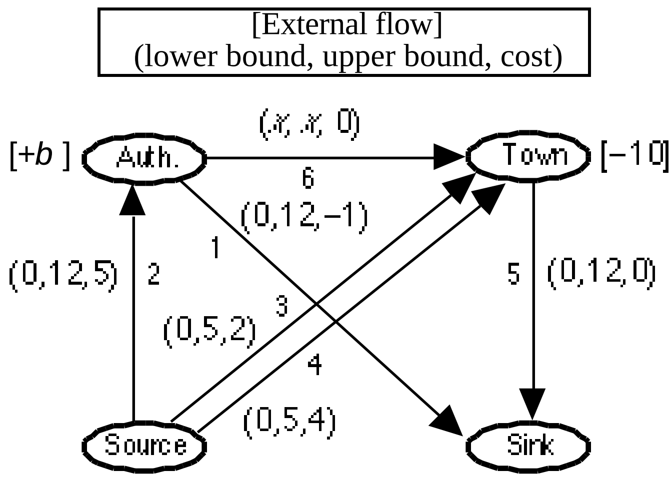

Figure 4. Network model of the example problem

The authority must select an amount to allocate to the town, shown as *x* in Fig. 4, in a manner that minimizes total system cost. This is a trivial problem if *b* is known. The optimal solution is

$$x = b \text{ if } b \leq 10 \text{ and } x = 10 \text{ if } b > 10.$$

Unfortunately, the natural supply *b* is uncertain. It may be either 0, 3, 6, 9, or 12 with equal probability. The probability distribution for water available is in Table 1.

Table 1. Probability Distribution for Water Available

Available (*b*)	0	3	6	9	12
Probability *P*(*b*)	0.2	0.2	0.2	0.2	0.2

Because the allocation *x* must be made before *b* is known, a number of new questions arise. If *x* > *b*, the authority must purchase the amount (*x*

– b) to fulfill its pledge. If $x < b$, the town will have unsatisfied demand while water is sent down stream. Certainly, the decision problem becomes more difficult.

To use a deterministic approach, we might solve the problem for the *average* availability. This average is $\mu_b = 6$. The solution is to allocate all six units to the town. The town has 4 units of unsatisfied demand for a cost of 8. This solution, however, does not model the uncertainty present. If b happens to be more than 6, the town experiences unnecessary shortage. If b is less than 6, the authority will have to buy water, when it is less expensive for the town to experience shortage.

We might try the *what-if* approach illustrated in Fig. 3. The present decision is the amount to allocate, x. The scenarios are the five possible future values of b. The future decisions are the arc flows after the value of b is realized. Table 2 shows the results for the five scenarios.

Table 2. Deterministic Scenario Solutions

Scenario	Probability	b_k	x_k
1	0.2	0	0
2	0.2	3	3
3	0.2	6	6
4	0.2	9	9
5	0.2	12	10

The table does not give much guidance on the single decision we must make. Using the probability distribution, the expected value of x is

$$E\{x\} = 0.2(0 + 3 + 6 + 9 + 10) = 5.6$$

Although this solution is not far from the optimum, it is not the solution that minimizes the expected cost so there is really no rational basis for its use. We find the optimal solution in the next section.

27.2 Stochastic Programming

More rational decisions are obtained with stochastic programming. Here a model is constructed that is a direct representation of Fig. 2. The present decisions $\mathbf{x}$, and the future decisions, $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_K$, are all represented explicitly in a linear programming model. When this model is solved, the optimal values of all these decisions are determined. The optimal values for the present decisions, $\mathbf{x}^*$, maximize the *expected* benefit (in a statistical sense) over all scenarios included in the model. The solution has the *hedging* and *flexibility* characteristics typically appearing in real decision situations and should be much more acceptable to decision makers than the results for a deterministic model.

Decision Making with Recourse

With the situation of Fig. 2, we have a two stage problem. An initial decision $\mathbf{x}$ is followed by a second stage decision $\mathbf{y}$. Some random occurrence comes between the first and second stages. This problem is often called *decision making with recourse*. The initial decision is made in the presence of uncertainty, but we recover from the possibly ill effects of the realization of the randomness with the recourse decision.

The Recourse Problem

Consider a problem that has the linear programming format except the right side vector $\mathbf{b}$ is originally not known with certainty. Rather it is a random vector with discrete realizations $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_K$. The probabilities of the realizations are given as $p_1, p_2, \ldots, p_K$. If the decision maker can wait until the value of $\mathbf{b}$ becomes known, the optimal decision is determined by solving the usual linear programming model

$$\begin{aligned} \text{Minimize} \quad & \mathbf{cx} \\ \text{subject to} \quad & \mathbf{Ax} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

The model has n variables and m constraints.

The decision process requires, however, that the original decision, $\mathbf{x}$, be made before the realization of the random variable. After $\mathbf{b}$ is known, a second decision $\mathbf{y}$, called the recourse decision, is required to adjust the decision $\mathbf{x}$ so that the constraints are satisfied and the objective is minimized. The problem of determining $\mathbf{y}$ is called the recourse problem.

$$\begin{aligned} \text{Minimize} \quad & \mathbf{qy} \\ \text{subject to} \quad & \mathbf{My} = \mathbf{b} - \mathbf{Ax} \\ & \mathbf{y} \geq \mathbf{0} \end{aligned}$$

The recourse problem has n_0 variables in the vector $\mathbf{y}$ and m constraints. The vector $\mathbf{q}$ defines the recourse costs. The matrix $\mathbf{M}$ describes the interaction between the original decisions $\mathbf{x}$ and the recourse decisions $\mathbf{y}$.

The problem is to determine the original decision vector $\mathbf{x}$ that will minimize the original objective value plus the expected recourse costs.

$$\begin{array}{ll} \text{Minimize} & \mathbf{cx} + E[\mathbf{qy} \mid \mathbf{My} = \mathbf{b} - \mathbf{Ax},\ \mathbf{y} \geq \mathbf{0}] \\ \text{subject to} & \mathbf{x} \geq \mathbf{0} \end{array}$$

Here, $E[\cdot]$ is the expected value taken over the random vector $\mathbf{b}$.

Linear Programming Formulation

This problem has the linear programming model below (Dantzig and Madansky [1961]).

$$\begin{array}{llllllll} \text{Minimize} & \mathbf{cx} & + p_1\mathbf{qy}_1 & + p_2\mathbf{qy}_2 + & \cdots & + p_K\mathbf{qy}_K & & \\ \text{subject to} & \mathbf{Ax} & + \mathbf{My}_1 & & & & = \mathbf{b}_1 \\ & \mathbf{Ax} & & + \mathbf{My}_2 & & & = \mathbf{b}_2 \\ & \vdots & & & & & \vdots \\ & \mathbf{Ax} & & & & + \mathbf{My}_K & = \mathbf{b}_K \end{array}$$

$$\mathbf{x} \geq \mathbf{0}, \quad \mathbf{y}_1 \geq \mathbf{0}, \quad \mathbf{y}_2 \geq \mathbf{0}, \ldots, \mathbf{y}_K \geq \mathbf{0}$$

This linear programming model has $n + Kn_0$ variables and Km constraints. The constraint matrix has a dual block angular form. The model can be generalized by allowing the recourse matrix $\mathbf{M}$ and the recourse cost $\mathbf{q}$ to vary with the realizations. Let the matrix realizations be $\mathbf{M}_1, \mathbf{M}_2, \ldots, \mathbf{M}_K$, and the objective vectors be $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_K$, correspond to the K right side vector possibilities. We can also require the original variables to satisfy linear constraints that are not affected by uncertainty. These are

$$\mathbf{A}_0\mathbf{x} = \mathbf{b}_0.$$

With these additions, the linear programming formulation becomes

$$\begin{array}{llllllll} \text{Minimize} & \mathbf{cx} & + p_1\mathbf{q}_1\mathbf{y}_1 & + p_2\mathbf{q}_2\mathbf{y}_2 + & \cdots & + p_K\mathbf{q}_K\mathbf{y}_K & & \\ \text{subject to} & \mathbf{A}_0\mathbf{x} & & & & & = \mathbf{b}_0 \\ & \mathbf{Ax} & + \mathbf{M}_1\mathbf{y}_1 & & & & = \mathbf{b}_1 \\ & \mathbf{Ax} & & + \mathbf{M}_2\mathbf{y}_2 & & & = \mathbf{b}_2 \\ & \vdots & & & & & \vdots \\ & \mathbf{Ax} & & & \cdots & + \mathbf{M}_K\mathbf{y}_K & = \mathbf{b}_K \end{array}$$

$$\mathbf{x} \geq \mathbf{0}, \quad \mathbf{y}_1 \geq \mathbf{0}, \quad \mathbf{y}_2 \geq \mathbf{0}, \ldots, \mathbf{y}_K \geq \mathbf{0}$$

In any specific case, the size of this linear program is large because of the large number of possible realizations of the random events that determine the characteristics of the recourse problems. We have, however, a deterministic equivalent of the stochastic problem that can be solved by linear programming.

Example 2

For the example introduced in the last section we write the linear programming model written as if b was a known.

$$
\begin{array}{llllllll}
\text{Minimize} & z = 0x & -1y_1 & +5y_2 & +2y_3 & +4y_4 & +0y_5 & \\
& x & +y_1 & -y_2 & & & & = b \\
& x & & & +y_3 & +y_4 & -y_5 & = 10
\end{array}
$$

$$0 \le x,\ 0 \le y_1 \le 12,\ 0 \le y_2 \le 12,\ 0 \le y_3 \le 5,\ 0 \le y_4 \le 5,\ 0 \le y_5 \le 12$$

The objective is to minimize cost. The y variables are the arc flows indexed according to the arc numbers in Fig. 4. The x variable is the flow for arc 6. We differentiate it because this is the present decision that must be made before the random variable is realized. The constraints are conservation of flow requirements at the authority and town nodes. Conservation is not required at the source and sink nodes.

The stochastic programming model is formed by defining recourse variables for each realization. Thus let

$$y_{jk} = \text{the flow in arc } j \text{ in realization } k$$

Now the deterministic linear program is

$$
\begin{array}{lll}
\text{Minimize} \quad z = & 0x & +0.2(-1y_{11} + 5y_{21} + 2y_{31} + 4y_{41} + 0y_{51}) \\
& & +0.2(-1y_{12} + 5y_{22} + 2y_{32} + 4y_{42} + 0y_{52}) \\
& & +0.2(-1y_{13} + 5y_{23} + 2y_{33} + 4y_{43} + 0y_{53}) \\
& & +0.2(-1y_{14} + 5y_{24} + 2y_{34} + 4y_{44} + 0y_{54}) \\
& & +0.2(-1y_{15} + 5y_{25} + 2y_{35} + 4y_{45} + 0y_{55})
\end{array}
$$

subject to

$$
\begin{array}{lllllllr}
\text{real. 1} & x & +y_{11} & -y_{21} & & & & = 0 \\
& x & & & +y_{31} & +y_{41} & -y_{51} & = 10 \\
\text{real. 2} & x & +y_{12} & -y_{22} & & & & = 3 \\
& x & & & +y_{32} & +y_{42} & -y_{52} & = 10 \\
\text{real. 3} & x & +y_{13} & -y_{23} & & & & = 6 \\
& x & & & +y_{33} & +y_{43} & -y_{53} & = 10 \\
\text{real. 4} & x & +y_{14} & -y_{24} & & & & = 9 \\
& x & & & +y_{34} & +y_{44} & -y_{54} & = 10 \\
\text{real. 5} & x & +y_{15} & -y_{25} & & & & = 12 \\
& x & & & +y_{35} & +y_{45} & -y_{55} & = 10
\end{array}
$$

$$
\begin{array}{lllll}
0 \le y_{11} \le 12, & 0 \le y_{21} \le 12, & 0 \le y_{31} \le 5, & 0 \le y_{41} \le 5, & 0 \le y_{51} \le 12 \\
0 \le y_{12} \le 12, & 0 \le y_{22} \le 12, & 0 \le y_{32} \le 5, & 0 \le y_{42} \le 5, & 0 \le y_{52} \le 12 \\
0 \le y_{13} \le 12, & 0 \le y_{23} \le 12, & 0 \le y_{33} \le 5, & 0 \le y_{43} \le 5, & 0 \le y_{53} \le 12 \\
0 \le y_{14} \le 12, & 0 \le y_{24} \le 12, & 0 \le y_{34} \le 5, & 0 \le y_{44} \le 5, & 0 \le y_{54} \le 12 \\
0 \le y_{15} \le 12, & 0 \le y_{25} \le 12, & 0 \le y_{35} \le 5, & 0 \le y_{45} \le 5, & 0 \le y_{55} \le 12
\end{array}
$$

Solving this model we obtain,

$$z^* = 14.6 \text{ and } x^* = 5.$$

The solution for the recourse variables is shown in Table 3.

Table 3. Solution for the Recourse Variables

Realization, k	b_k	y_{1k}	y_{2k}	y_{3k}	y_{4k}	y_{5k}
1	0	0	5	5	0	0
2	3	0	2	5	0	0
3	6	1	0	5	0	0
4	9	4	0	5	0	0
5	12	7	0	5	0	0

The solution that minimizes the expected cost over all realizations is to allocate 5 units to the town. The town experiences a shortage of 5 units with a cost of $10. In the event that the natural supply is greater than 5, the excess is released down stream. When the natural supply is less than 5, the shortage is made up by purchasing water. We have found this solution with a deterministic linear programming model, but it is a rather large model for such a small problem, 26 variables and 10 constraints.

Although this problem is a network flow model for any fixed value of x, the stochastic programming problem no longer has the network structure. The model must be solved as a general linear program.

Multistage Models

The approach can be adapted to multistage problems, as illustrated in Fig. 5. Here, alternative futures are broken down into a series of periodic decisions. An illustration is the water planning problem suggested at the beginning of the chapter. This is not a two stage problem, but decisions must be made in each of a number of future periods. The periods may represent months and the decisions are the amounts to release in each month. Uncertain events are revealed in a periodic manner. Thus the present decision $\mathbf{x}$ determines releases in the first month. These releases must be made when all future supplies and demands are uncertain. After the first month passes, part of the uncertainty is removed; specifically, the supplies and demands in the first month are known. Now the decisions of the second month must be made. The sequential decisions and realizations of uncertain events are represented explicitly in the model, so the optimal decisions at the present time can be determined.

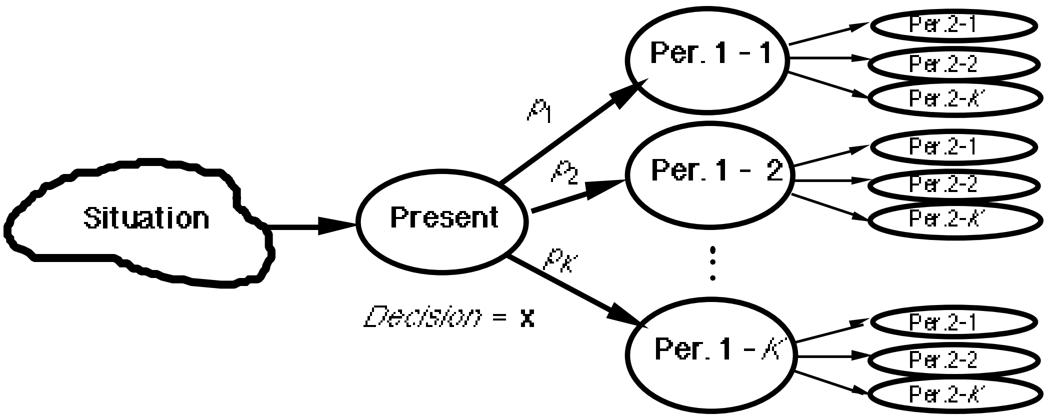

Figure 5. Multistage stochastic model

The difficulty with the stochastic programming model is its size and the corresponding computational difficulty required to obtain a solution. A model representing K scenarios of the type shown in Fig. 2 is K times larger than the similar deterministic problem of Fig. 1. Models for the multiperiod problem of Fig. 5 are $p \times K$ times the size of the deterministic problem when there are p periods with K scenarios at each period. A second difficulty is the structure of the mathematical programming model. When the *what if* problem has the structure of a network flow programming model, the stochastic programming model will be a general linear program, a much harder model to solve.

27.3 Markovian Decision Processes

Chapters 12 - 16 described a stochastic process called the Markov chain. The procedures described in those chapters allow one to analyze the system by computing such measures as the steady-state probabilities. This section adds the dimension of decision making to the Markov chain. Now, at every step of the process, before the transition to the next state is determined, a decision must be made that affects the transition probabilities. This section explains how the problem is stated and solved as a linear program. There are number of interesting applications of the model and the associated solution procedures.

The Decision Model

The Markov chain is defined by states numbered 0 through $m - 1$. For each pair of states i and j, there is a transition probability p_{ij}. The transition probabilities are arranged into the transition matrix $\mathbf{P}$. Steady-state probabilities are

$$\pi = (\pi_0, \pi_1, \ldots, \pi_{m-1})$$

They can be found by solving the set of linear equations

$$\pi(\mathbf{P} - \mathbf{I}) = \mathbf{0} \ \text{ and } \sum_{j=0}^{m-1} \pi_j = 1.$$

We add a decision dimension to this problem by defining for each state i a decision d_i, where d_i is an index ranging from 1 through K representing possible actions one might take while in state i. A policy R specifies a decision for each state.

$$\{d_0(R), d_1(R), \ldots, d_{m-1}(R)\}$$

Decisions affect the transition probabilities, so we must define the transition probability as a function if the decision. Thus $p_{ij}(k)$ is the probability of a transition from state i to state j if one makes the decision k. The values of $p_{ij}(k)$ are data that must be provided for every pair of states for every decision.

We also define the cost incurred if decision k is made while in state i as C_{ik}. The steady state probabilities and the cost of operating the Markov chain depends on the policy R. We define $\pi(R)$ as the steady-state probabilities with policy R, and $C(R)$ as the expected cost of operation with policy R.

$$C(R) = \sum_{i=0}^{m-1} \left[C_{ik}\pi_i(R) \mid k = d_i(R) \right]$$

The goal is to find the policy that minimizes the expected cost.

The *decision matrix*, $\mathbf{D}$, is an $m \times K$ matrix with component D_{ik} defined as the probability that the decision k will be taken, given that the system is in state i.

$$D_{ik} = P\{\text{decision} = k \mid \text{state} = i\}, \ i = 0, \ldots, m-1 \text{ and } k = 1, \ldots, K.$$

The decision matrix completely defines a decision policy. Although the model allows a mixed policy[1], the optimal solution is always a pure policy. The components of the optimal decision matrix are either 0 or 1.

Linear Programming Model

Given the data $p_{ij}(k)$and C_{ik} for all values of i, j and k, this problem is modeled and solved as a linear program. The general formulation follows with an example provided in the next section.

The decision variables of the linear program are

$$y_{ik} = P\{\text{state is } i \text{ and the decision is } k\},\ \ i = 0,\ldots,m–1 \text{ and } k = 1,\ldots,K$$

The objective is to minimize the expected cost of operation.

$$\text{Minimize } z = \sum_{i=0}^{m-1}\sum_{k=1}^{K} C_{ik}y_{ik}$$

The first set of constraints requires that the steady-state probabilities sum to 1.

$$\sum_{i=0}^{m-1}\sum_{k=1}^{K} y_{ik} = 1$$

The second set of constraints are equations define the steady-state probabilities.

$$\sum_{k=1}^{K} y_{jk} - \sum_{i=0}^{m-1}\sum_{k=1}^{K} y_{ik}p_{ij}(k) = 0,\ \ j = 0,1,\ldots,m–1$$

Finally, we require nonnegativity for the probabilities.

$$y_{ik} \geq 0,\ \ i = 0, 1,\ldots, m–1 \text{ and } k = 1, 2,\ldots,K$$

After the linear program is solved, the results in more usable form are obtained by computing the decision probabilities. The steady state probability π_i is the sum over all possible decisions in state i.

$$\pi_i = \sum_{k=1}^{K} y_{ik},\ \ i = 0,1,\ldots,m–1$$

The decision probabilities are

$$D_{ik} = y_{ik}/\pi_i \ \ \text{ for } i = 0,1,\ldots,m–1 \text{ and } k = 1, 2,\ldots,K$$

Example 3

Consider again the computer repair problem that was described in Section 12.1. An office has two computers that are used for word processing. It has been observed that when both are working in the morning, there is a 30% chance that one will be failed by evening and a 10% chance that both

[1] Mixed policies are defined in the context of game theory in Chapter 24.

will be failed. If it happens that only one computer is working in the morning, there is a 20% chance that it will be failed by evening.

With one computer failed, some of the office work must be sent to a typing service at a cost of \$20 per day. With both machines failed, the cost of the service increases to \$100 per day.

The technician charges \$40 to fix a computer and it takes 1 day for the repair. When he is called in the morning he picks up the broken machines immediately and guarantees their return by the next morning. He charges a fixed fee of \$50 for a pick up. This fee is independent of how many need repair.

The office manager must decide when to call the technician. Should she call after the first machine fails or wait until both have failed? Which policy yields the smallest average cost?

Solution

There are really only two possible decisions at any state.

1. Do not call the technician
2. Call the technician

Thus we have $K = 2$ with the alternatives numbered as above.

The state of the system is the number of failed computers observed after the technician has returned repaired machines. There are three states for this problem:

0. No computers have failed
1. One computer has failed
2. Two computers have failed

The obvious decision for state 0 is not to call the technician. Thus $d_0 = 1$. When the system is in state 2, the optimal plan is to call the technician (the machines must eventually be repaired so there is no reason to delay), so $d_2 = 2$. State 1 involves the only question with both alternatives as reasonable possibilities.

We use a linear programming model to solve this problem. Since there are two decisions and three states, there are potentially six decision variables.

$y_{01} = P\{$no computers have failed and technician is not called$\}$

$y_{02} = P\{$no computers have failed and technician is called$\}$

$y_{11} = P\{$one computer has failed and technician is not called$\}$

$y_{12} = P\{$one computer has failed and technician is called$\}$

$y_{21} = P\{$two computers have failed and technician is not called$\}$

$y_{22} = P\{$two computers have failed and technician is called$\}$

We delete y_{02} and y_{21} as impractical alternatives.

We define $\mathbf{P}_i(k)$ as row i of the transition matrix if decision k is selected. For the four remaining decisions, we have

$$\mathbf{P}_0(1) = [\ 0.6 \quad 0.3 \quad 0.1\ ]$$

$$\mathbf{P}_1(1) = [\ 0 \quad 0.8 \quad 0.2\ ]$$

$$\mathbf{P}_1(2) = [\ 0.8 \quad 0.2 \quad 0\ ]$$

$$\mathbf{P}_2(2) = [\ 1 \quad 0 \quad 0\ ]$$

The cost of operation given the state and decision can also be computed.

$C_{01} =$ 0, there is no cost if no typewriters are failed

$C_{11} =$ 20, this is the cost of the typing service with one failed

$C_{12} =$ 110, this is the cost of the typing service plus the cost of repairing the machine

$C_{22} =$ 230, this is the cost of the typing service plus the cost of repairing two machines

Using general notation, we have the following linear programming model.

$$\text{Minimize } z = C_{01}y_{01} + C_{11}y_{11} + C_{12}y_{12} + C_{22}y_{22}$$

$$\text{subject to } \quad y_{01} + y_{11} + y_{12} + y_{22} = 1$$

$$y_{01} - [p_{00}(1)y_{01} + p_{10}(1)y_{11} + p_{10}(2)y_{12} + p_{20}(2)y_{22}] = 0$$

$$y_{11} + y_{12} - [p_{01}(1)y_{01} + p_{11}(1)y_{11} + p_{11}(2)y_{12} + p_{21}(2)y_{22}] = 0$$

$$y_{22} - [p_{02}(1)y_{01} + p_{12}(1)y_{11} + p_{12}(2)y_{12} + p_{22}(2)y_{22}] = 0$$

$$y_{01} \geq 0,\ y_{11} \geq 0,\ y_{12} \geq 0,\ y_{22} \geq 0$$

Substituting the probabilities and costs, gives

$$\text{Minimize } z = 0y_{01} + 20y_{11} + 110y_{12} + 230y_{22}$$

$$\text{subject to } \quad y_{01} + y_{11} + y_{12} + y_{22} = 1$$

$$0.4y_{01} - 0.8y_{12} - y_{22} = 0$$

$$-0.3y_{01} + 0.2y_{11} + 0.8y_{12} = 0$$

$$-0.1y_{01} - 0.2y_{11} + y_{22} = 0$$

$$y_{01} \geq 0, \;\; y_{11} \geq 0, \;\; y_{12} \geq 0, \;\; y_{22} \geq 0.$$

with optimal solution

$$z^* = 43.56, \; y^*_{01} = 0.678, \;\; y^*_{11} = 0, \; y^*_{12} = 0.254, \; y^*_{22} = 0.068.$$

From this result, we compute the steady -state probabilities

$$\pi_0 = 0.678, \;\; \pi_1 = 0.254, \;\; \pi_2 = 0.068,$$

and the optimal decisions

$$D_{01} = 1, \;\; D_{11} = 0, \;\; D_{12} = 1, \;\; D_{22} = 1.$$

The optimal policy is to call the technician if there are either one or two failed machines. The expected cost per day is $43.56.

27.4 Exercises

Section 27.2

1. A manufacturer has supplies of some commodity at three warehouses which he routinely ships to three customers. The unit shipping costs, the supplies at the warehouses, the revenue at each customer, and three possible demand levels are shown in the table.

	Shipping costs			Supply
Warehouse 1	10	20	18	20
Warehouse 2	8	15	12	15
Warehouse 3	13	17	19	15
Revenue	20	25	27	
Low demand	5	13	10	
Medium demand	10	20	16	
High demand	12	23	20	

The manufacturer must ship products to the customers, but their demands are not known with certainty. Based on past experience each customer has been given three demand levels: low, medium and high, as shown in the table. For this example, all customers will have the same level of demand but the level is determined with the following probabilities.

$$P\{\text{low}\} = 0.2,\ \ P\{\text{medium}\} = 0.5,\ \ P\{\text{high}\} = 0.3$$

If the amount shipped to a customer is less than the amount demanded, sales are lost. The cost of a lost sale is the lost revenue. If the amount shipped to a customer is more than the amount demanded, the excess product must be sold at a discount. The revenue for such a sale is $5. Not all supplies need be shipped. Determine how much to ship to each customer to maximize expected profit.

2. Solve Exercise 1 using the following data.

$$P\{\text{low}\} = 0.5,\ \ P\{\text{medium}\} = 0.3,\ \ P\{\text{high}\} = 0.2.$$

3. Consider again Exercise 1, but allow any shortages to be made up with emergency shipments from the warehouses using any supplies that remain after the original shipments. The cost of an emergency shipment is $5 more than the original shipping cost shown in the table. Shortages can also result in lost sales as originally proposed. Solve this problem.

Section 27.3

4. Consider the example in Section 27.3, but assume that the company buys another computer for a total of three machines. When all three machines are working in the morning there is a 32% probability that one will have failed by the evening, a 10% probability that two will have failed, and a 3% probability that all three will have failed. As before, when two are working in the morning, there is a 30% chance that one will have failed by the evening and a 10% chance that both will have failed. If it

happens that only one computer is working in the morning, there is a 20% chance that it will have failed by evening.

With one computer out of commission, some of the office work must be sent to a typing service at a cost of $10 per day. With two machines out of commission the cost will be $25, and with all three machines failed the cost of the service increases to $100 per day.

The technician charges $40 to fix a computer and it takes 1 day for the repair. When he is called in the morning he picks up the failed machines immediately and will guarantee their return by the next morning. He charges a fixed fee of $50 to pick up the machines. This fee is independent of how many need repair. With a minimum cost policy, how many machines must be out of commission before the office manager calls the technician?

5. This is an abstract problem in which there are three states. In each state, one of three decisions must be made. The matrices $\mathbf{P}(k)$ show the transition probabilities, assuming the same decision k is made in every state. This is not a limitation, however, since the policy may use a different decision for each state. Also shown is the decision cost matrix $\mathbf{C}$. Set up and solve a linear program to determine the optimal decision in each state. Also find the steady-state probabilities. The goal is to minimize steady state average cost.

$$\mathbf{P}(1) = \begin{bmatrix} 0.0 & 0.3 & 0.7 \\ 0.9 & 0.1 & 0.0 \\ 0.2 & 0.0 & 0.8 \end{bmatrix}$$

$$\mathbf{P}(2) = \begin{bmatrix} 0.1 & 0.6 & 0.3 \\ 0.7 & 0.2 & 0.1 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}$$

$$\mathbf{P}(3) = \begin{bmatrix} 0.3 & 0.5 & 0.2 \\ 0.6 & 0.1 & 0.3 \\ 0.2 & 0.4 & 0.4 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 10 & 15 & 20 \\ 15 & 20 & 25 \\ 20 & 25 & 30 \end{bmatrix}$$

6. A dealer selling high-priced cars faces the following weekly demand distribution.

Demand	0	1	2
Probability	0.1	0.5	0.4

The dealer looks at the inventory at the end of each week and must determine if an order should be placed and, if so, how many cars to order. The cars arrive 1 week after the order is placed. There may be a total of no more than five cars either in inventory or on order. The cost of placing an order is $600, independent of how many cars are ordered. If there is a demand for a car and there is no inventory, the sale is lost with a lost profit of $2000. The cost associated with an unsold car in inventory at the end of a week is $200. Set up this problem as a Markov decision process and find the optimal policy.

7. A queueing system has three service channels. Using a 5-minute study period, the following probabilities have been determined for the number of arrivals during the period.

Arrivals	0	1	2	3
Probability	0.1	0.3	0.4	0.2

For simplicity assume that the arrivals occur at the beginning of the period. These arrivals will enter the system until the number in the system is 10. When this occurs the arrivals will balk. The cost of a balking customer is $100.

The number of customers who leave the system during a period depends on the number of channels open and the number in the system at the beginning of the period. The probability distribution for the number of service completions given that there are sufficient customers in the system is given below for the various numbers of channels.

Completions	0	1	2	3	4
One channel open	0.3	0.5	0.2	0	0
Two channels open	0.1	0.2	0.4	0.3	0
Three channels open	0	0.1	0.2	0.5	0.2

If there aren't enough customers in the system to utilize the full capacity of a particular channel configuration, the probabilities for the excess capacities are combined with the probability for the specific number of customers in the system. For instance, if there are only two customers in the system at some period, the service probability distributions appears as below.

Completions	0	1	2	3	4
One channel open	0.3	0.5	0.2	0	0
Two channels open	0.1	0.2	0.7	0	0
Three channels open	0	0.1	0.9	0	0

Let the charge for an open channel be $7 per period. Let the waiting cost for a customer be $2 for each whole period spent in the system. You must determine a policy for how many channels to open as a function of the number of customers in the system. Describe this problem as a Markov decision problem and solve it with a linear programming code.

Bibliography

András, P., *Stochastic Programming*, Kluwer, Norwell, MA, 1995.

Bertsekas, D.P., Dynamic Programming: Deterministic and Stochastic Models, Prentice-Hall, Englewood Cliffs, NJ, 1987.

Birge, J.R. and F. Louveaux, *Introduction to Stochastic Programming*, Springer-Verlag, New York, 1997.

Dantzig, G.B. and A. Madansky, "On the Solution to Two Stage Linear Programs under Uncertainty", In Neyman, I.J. (editor), *Proceedings of the 4th Berkeley Symposium on Statistics and Probability*," pp. 165-176, Berkeley, CA, 1961.

Kall, P. and S.W. Wallace, *Stochastic Programming*, John Wiley & Sons, New York, 1994.

Mulvey, J.M. and H. Vladimirou, "Stochastic Network Optimization Models for Investment Planning," *Annals of Operations Research*, Vol. 20, pp. 187-217, 1989.

Ross, S.M., *Probability Models*, Seventh Edition, Academic Press, San Diego, 2000.

Sen, S. and J.L. Higle, "An Introductory Tutorial on Stochastic Linear Programming Models," *Interfaces*, Vol. 29, No. 2, pp. 33-61, 1999.