state 0, 1, or 2, and replacing it when it is in state 3. The minimum expected total discounted cost of the system starting in state i, $i = 0, 1, 2, 3$, and evolving for one period is given by 0, 1,000, 3,000, and 6,000, respectively. The optimal solution to the two-period machine-maintenance model is

Period 1 Leave machine alone when it is in state 0 or 1.
Overhaul machine when it is in state 2.
Replace machine when it is in state 3.

Period 2 Leave machine alone when it is in state 0, 1, or 2.
Replace machine when it is in state 3.

The minimum expected total discounted cost of the system starting in state i, $i = 0, 1, 2, 3$, and evolving for two periods is given by 1,294, 2,688, 4,900, and 6,000, respectively. Finally, the optimal solution to the three-period model is

Period 1 and *Period 2* { Leave machine alone when it is in state 0 or 1.
Overhaul machine when it is in state 2.
Replace machine when it is in state 3.

Period 3 Leave machine alone when it is in state 0, 1, or 2.
Replace machine when it is in state 3.

The minimum expected total discounted costs over three periods, if the system starts in state i, $i = 0, 1, 2, 3$, are given by 2,730, 4,041, 6,419, and 7,165, respectively.

20.6 A Water-Resource Model

A multipurpose dam is used for generating electric power as well as for flood control. The capacity of the dam is 3 units. The probability distribution of the quantity of water, W_t, that flows into the dam during month t (for $t = 0, 1, \ldots$) is given by $P_W(m)$, where

$$P_W(0) = P\{W = 0\} = \tfrac{1}{6}$$
$$P_W(1) = P\{W = 1\} = \tfrac{1}{3}$$
$$P_W(2) = P\{W = 2\} = \tfrac{1}{3}$$
$$P_W(3) = P\{W = 3\} = \tfrac{1}{6}.$$

For the purpose of generating electric power, 1 unit of water is required. At the beginning of each month, water is released from the dam. The first unit is used to generate electric power and then used for irrigation purposes, the latter function being worth \$100,000. If additional units are released, they can also be used for irrigation purposes, and each unit is worth \$100,000. If the dam contains less than 1 unit at the beginning of a month, additional power must be purchased at a cost of \$300,000. If at any time the water in the dam exceeds the capacity of 3 units, the excess water is released through the spillways at no cost or gain.

A release policy is sought. Policies are to be compared on the basis of expected discounted cost, with discount factor $\alpha = 0.99$. The *policy improvement algorithm* will be used.

Let X_t denote the amount of water in the dam at time t. Then $X_t = 0, 1, 2, 3$. The natural laws of motion for this system (no water released) are given by the transition matrix:

State	0	1	2	3
0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$
1	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$
2	0	0	$\frac{1}{6}$	$\frac{5}{6}$
3	0	0	0	1

For example, the element in the second row and fourth column, p_{13}, is obtained as follows: If the dam contains 1 unit of water now, then for it to contain 3 units of water a month later, 2 or 3 units of water must flow into the dam during the month (recall that dam capacity is 3 units, so that a flow of 3 units will result in 1 unit being released through the spillways). This occurs with probability $\frac{1}{3} + \frac{1}{6} = \frac{1}{2}$.

There are three possible decisions that can be made at the beginning of each month:

Decision	Action
1	Release 1 unit
2	Release 2 units
3	Release 3 units

It is clear that releasing no units is not a sensible action, because 1 unit is needed for electric power generation anyway. Thus a policy calls for determining how many units to release as a function of the quantity of water found in the dam. A typical policy R_1 might call for releasing all the water in the dam if it contains 0, 1, or 2 units, and releasing 2 units if it contains 3 units. The resultant transition matrix is given by

State	0	1	2	3
0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$
1	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$
2	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$
3	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$

Of course, a policy that calls for releasing 3 units when there is only 1 unit in the dam is to be interpreted as calling for releasing all the available water. Necessary

cost information can be obtained from the following data:

State	Decision	Cost (In Hundred Thousands)
0	1	3
	2	3
	3	3
1	1	−1
	2	−1
	3	−1
2	1	−1
	2	−2
	3	−2
3	1	−1
	2	−2
	3	−3

The policy R_1 will be used in the value-determination step (step 1) of the policy improvement algorithm. Using the cost information just given, the values of C_{ik_1} are

$$C_{0k_1} = 3$$

$$C_{1k_1} = -1$$

$$C_{2k_1} = -2$$

$$C_{3k_1} = -2.$$

The following four equations must be solved:

$$V_0(R_1) = 3 + 0.99[\tfrac{1}{6}V_0(R_1) + \tfrac{1}{3}V_1(R_1) + \tfrac{1}{3}V_2(R_1) + \tfrac{1}{6}V_3(R_1)]$$
$$V_1(R_1) = -1 + 0.99[\tfrac{1}{6}V_0(R_1) + \tfrac{1}{3}V_1(R_1) + \tfrac{1}{3}V_2(R_1) + \tfrac{1}{6}V_3(R_1)]$$
$$V_2(R_1) = -2 + 0.99[\tfrac{1}{6}V_0(R_1) + \tfrac{1}{3}V_1(R_1) + \tfrac{1}{3}V_2(R_1) + \tfrac{1}{6}V_3(R_1)]$$
$$V_3(R_1) = -2 + 0.99[\tfrac{1}{6}V_1(R_1) + \tfrac{1}{3}V_2(R_1) + \tfrac{1}{2}V_3(R_1)].$$

The simultaneous solution of these equations results in the values

$$V_0(R_1) = -103.881,\ V_1(R_1) = -107.881,$$

and

$$V_2(R_1) = -108.881,\ V_3(R_1) = -110.358.$$

Step 2 can now be applied. We want to find an improved policy R_2 that has the property that $d_0(R_2) = k_2^0$, $d_1(R_2) = k_2^1$, $d_2(R_2) = k_2^2$, and $d_3(R_2) = k_2^3$ minimizes the following expressions:

(0) $$C_{0k_2^0} + 0.99[-103.881p_{00}(k_2^0) - 107.881p_{01}(k_2^0) - 108.881p_{02}(k_2^0) - 110.358p_{03}(k_2^0)]$$

(1) $$C_{1k_2^1} + 0.99[-103.881p_{10}(k_2^1) - 107.881p_{11}(k_2^1) - 108.881p_{12}(k_2^1) - 110.358p_{13}(k_2^1)]$$

(2) $C_{2k_2^2} + 0.99[-103.881p_{20}(k_2^2) - 107.881p_{21}(k_2^2) - 108.881p_{22}(k_2^2) - 110.358p_{23}(k_2^2)]$

(3) $C_{3k_2^3} + 0.99[-103.881p_{30}(k_2^3) - 107.881p_{31}(k_2^3) - 108.881p_{32}(k_2^3) - 110.358p_{33}(k_2^3)]$.

To find k_2^0, the best decision when the system is in state 0, it is necessary to evaluate the first expression for all possible decisions. It is clear that when the system is in state 0 (dam empty), there is no choice among the decisions because they all are equivalent. The data for the necessary calculations for evaluating expression (0) follow:

State 0

Decision	$p_{00}(k_2)$	$p_{01}(k_2)$	$p_{02}(k_2)$	$p_{03}(k_2)$	C_{0k_2}	Total Value of Expression 0
1, 2, 3	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	3	−103.881

Similarly, for state 1 there is no choice among the decisions because they are all equivalent. The data for the necessary calculations for evaluating follow:

State 1

Decision	$p_{10}(k_2)$	$p_{11}(k_2)$	$p_{12}(k_2)$	$p_{13}(k_2)$	C_{1k_2}	Total Value of Expression 1
1, 2, 3	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	−1	−107.881

For the remaining two states, the appropriate transition probabilities and costs generally depend upon the decisions made. The data for the necessary calculations for finding the best decisions, given the dam is in state 2 or 3, follow:

State 2

Decision	$p_{20}(k_2)$	$p_{21}(k_2)$	$p_{22}(k_2)$	$p_{23}(k)_2$	C_{2k_2}	Total Value of Expression 2
1	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$	−1	−109.358
2, 3	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	−2	−108.881

State 3

Decision	$p_{30}(k_2)$	$p_{31}(k_2)$	$p_{32}(k_2)$	$p_{33}(k_2)$	C_{3k_2}	Total Value of Expression 3
1	0	0	$\frac{1}{6}$	$\frac{5}{6}$	−1	−110.011
2	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$	−2	−110.358
3	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	−3	−109.881

Thus $d_0(R_2) = k_2^0 = d_1(R_2) = k_2^1 = 1, 2,$ or 3; $d_2(R_2) = k_2^2 = 1$; and $d_3(R_2) = k_2^3 = 2$. Hence policy R_2 calls for releasing all the water when there is 1 unit in the dam, 1 unit of water when there are 2 units available in the dam, and 2 units when there are 3 units available in the dam. This policy differs from R_1, so that another iteration is required. For the value-determination step, the equations that must now be solved are

$$\begin{aligned}
V_0(R_2) &= 3 + 0.99[\tfrac{1}{6}V_0(R_2) + \tfrac{1}{3}V_1(R_2) + \tfrac{1}{3}V_2(R_2) + \tfrac{1}{6}V_3(R_2)]\\
V_1(R_2) &= -1 + 0.99[\tfrac{1}{6}V_0(R_2) + \tfrac{1}{3}V_1(R_2) + \tfrac{1}{3}V_2(R_2) + \tfrac{1}{6}V_3(R_2)]\\
V_2(R_2) &= -1 + 0.99[\tfrac{1}{6}V_1(R_2) + \tfrac{1}{3}V_2(R_2) + \tfrac{1}{2}V_3(R_2)]\\
V_3(R_2) &= -2 + 0.99[\tfrac{1}{6}V_1(R_2) + \tfrac{1}{3}V_2(R_2) + \tfrac{1}{2}V_3(R_2)].
\end{aligned}$$

The simultaneous solution of these equations results in the values $V_0(R_2) = -119.642$, $V_1(R_2) = -123.642$, $V_2(R_2) = -125.119$, and $V_3(R_2) = -126.119$.

Step 2 can now be applied. We want to find an improved policy R_3 that has the property that $d_0(R_3) = k_3^0$, $d_1(R_3) = k_3^1$, $d_2(R_3) = k_3^2$, and $d_3(R_3) = k_3^3$ minimizes the following expressions:

$$\begin{aligned}
(0)\quad & C_{0k_3^0} + 0.99[-119.642p_{00}(k_3^0) - 123.642p_{01}(k_3^0) \\
& \qquad - 125.119p_{02}(k_3^0) - 126.119p_{03}(k_3^0)]\\
(1)\quad & C_{1k_3^1} + 0.99[-119.642p_{10}(k_3^1) - 123.642p_{11}(k_3^1) \\
& \qquad - 125.119p_{12}(k_3^1) - 126.119p_{13}(k_3^1)]\\
(2)\quad & C_{2k_3^2} + 0.99[-119.642p_{20}(k_3^2) - 123.642p_{21}(k_3^2) \\
& \qquad - 125.119p_{22}(k_3^2) - 126.119p_{23}(k_3^2)]\\
(3)\quad & C_{3k_3^3} + 0.99[-119.642p_{30}(k_3^3) - 123.642p_{31}(k_3^3) \\
& \qquad - 125.119p_{32}(k_3^3) - 126.119p_{33}(k_3^3)].
\end{aligned}$$

The data on the transition matrices and the costs from the previous iteration can again be used; the resulting values of the expression are

Decision	Value of Expression 0	Value of Expression 1	Value of Expression 2	Value of Expression 3
1	−119.642	−123.642	−125.119	−125.693
2	−119.642	−123.642	−124.642	−126.119
3	−119.642	−123.642	−124.642	−125.642

Thus $d_0(R_3) = k_3^0 = d_1(R_3) = k_3^1 = 1, 2,$ or 3; $d_2(R_3) = k_3^2 = 1$; and $d_3(R_3) = k_3^3 = 2$. Hence policy R_3 and policy R_2 are identical, and the optimal release policy calls for releasing all the water when there is 1 unit in the dam, 1 unit of water when there are 2 units available in the dam, and 2 units when there are 3 units available in the dam.

Of course, direct enumeration would have been just as simple a technique to use in this situation, but the policy improvement algorithm was used for illustrative purposes.

20.7 Inven

In Chap. 15 the
particular mod
the demand for th
It is assumed th
having a Poisson
of cameras on h
week one, X_2 the
On Saturday nigh
of the store on M
cameras on hand
store orders up to
in stock, no orde
the inventory on
curring a penalty
If $z > 0$ camera
are ordered, no c
15.7, this policy
time as the criter
this section is to
must be of the (s
assume that thre
stock. The *polic*
programming for
. Because X_t
hand at the end
are four possible

Decision	Ac
0	Do not
1	Order 1
2	Order 2
3	Order 3

The possible tra

State	0
0	1
1	$P\{D \geq 1\}$
2	$P\{D \geq 2\}$
3	$P\{D \geq 3\}$

Note that in this e

20.7 Inventory Model

In Chap. 15 the following inventory problem was considered. A camera store stocks a particular model camera that can be ordered weekly. Let $D_1, D_2, \ldots,$ represent the demand for this camera during the first week, the second week, . . . , respectively. It is assumed that the D_t are independent, identically distributed random variables having a Poisson distribution with parameter λ equal to 1. Let X_0 represent the number of cameras on hand at the outset, X_1 the number of cameras on hand at the end of week one, X_2 the number of cameras on hand at the end of week two, and so forth. On Saturday night the store places an order that is delivered in time for the opening of the store on Monday. The store uses an (s, S) ordering policy. If the number of cameras on hand at the end of the week is less than $s = 1$ (no cameras in stock), the store orders up to $S = 3$. Otherwise, the store does not order (if there are any cameras in stock, no order is placed). It is assumed that sales are lost when demand exceeds the inventory on hand (no backlogging). The cost structure considered calls for incurring a penalty cost of \$50 per unit for each unit of unsatisfied demand (lost sales). If $z > 0$ cameras are ordered, the cost incurred is $10 + 25z$ dollars. If no cameras are ordered, no ordering cost is incurred. Holding costs are to be neglected. In Sec. 15.7, this policy was evaluated by using the (long-run) expected average cost per unit time as the criterion. It is not evident that this policy is optimal, and the purpose of this section is to find the optimal policy. Even though we know that the optimal policy must be of the (s, S) form, we shall consider all possible policies, although we shall assume that three cameras is the maximum number of cameras that the store will stock. The *policy improvement algorithm* will be used first, followed by the *linear programming formulation*.

Because X_t represents the state of the system, i.e., the number of cameras on hand at the end of week t (before ordering), then $X_t = 0, 1, 2, 3$. Similarly, there are four possible decisions:

Decision	Action
0	Do not order
1	Order 1 camera
2	Order 2 cameras
3	Order 3 cameras

The possible transitions are given by[1]

	Decision 0			
State	0	1	2	3
0	1	0	0	0
1	$P\{D \geq 1\}$	$P\{D = 0\}$	0	0
2	$P\{D \geq 2\}$	$P\{D = 1\}$	$P\{D = 0\}$	0
3	$P\{D \geq 3\}$	$P\{D = 2\}$	$P\{D = 1\}$	$P\{D = 0\}$

[1] Note that in this example the set of possible decisions varies with the states.

Decision 1

State	0	1	2	3
0	$P\{D \geq 1\}$	$P\{D = 0\}$	0	0
1	$P\{D \geq 2\}$	$P\{D = 1\}$	$P\{D = 0\}$	0
2	$P\{D \geq 3\}$	$P\{D = 2\}$	$P\{D = 1\}$	$P\{D = 0\}$
3	Decision 1 not permitted			

Decision 2

State	0	1	2	3
0	$P\{D \geq 2\}$	$P\{D = 1\}$	$P\{D = 0\}$	0
1	$P\{D \geq 3\}$	$P\{D = 2\}$	$P\{D = 1\}$	$P\{D = 0\}$
2, 3	Decision 2 not permitted			

Decision 3

State	0	1	2	3
0	$P\{D \geq 3\}$	$P\{D = 2\}$	$P\{D = 1\}$	$P\{D = 0\}$
1, 2, 3	Decision 3 not permitted			

Recalling that the demand D is a Poisson random variable with parameter $\lambda = 1$, and using appendix Table A.5.4, these transitions can now be expressed as

Decision 0

State	0	1	2	3
0	1	0	0	0
1	0.632	0.368	0	0
2	0.264	0.368	0.368	0
3	0.080	0.184	0.368	0.368

Decision 1

State	0	1	2	3
0	0.632	0.368	0	0
1	0.264	0.368	0.368	0
2	0.080	0.184	0.368	0.368
3	Decision 1 not permitted			

Decision 2

State	0	1	2	3
0	0.264	0.368	0.368	0
1	0.080	0.184	0.368	0.368
2, 3	Decision 2 not permitted			

Decision 3

State	0	1	2	3
0	0.080	0.184	0.368	0.368
1, 2, 3	Decision 3 not permitted			

The cost information required is similar to that given in Sec. 15.7, and you are urged to review this material. A summary is given by

State	Decision
0	0
	1
	2
	3
1	0
	1
	2
	3
2	0
	1
	2, 3
3	0
	1, 2, 3

Choose th
the value-determ
R_1, calls for ord
hand); otherwise
must be solved s
is arbitrarily tak

or, alternatively,

$g(R_1) = 8$

$= 1$

$=$

$=$

State	Decision	Actual Cost Per Week	Expected Cost Per Week, C_{ik}
0	0	$50D$	$50E(D) = 50$
	1	$35 + 50 \max\{(D - 1), 0\}$	$35 + 50[1P\{D = 2\} + 2P\{D = 3\} + \cdots] = 53.4$
	2	$60 + 50 \max\{(D - 2), 0\}$	$60 + 50[1P\{D = 3\} + 2P\{D = 4\} + \cdots] = 65.2$
	3	$85 + 50 \max\{(D - 3), 0\}$	$85 + 50[1P\{D = 4\} + 2P\{D = 5\} + \cdots] = 86.2$
1	0	$50 \max\{(D - 1), 0\}$	$50[1P\{D = 2\} + 2P\{D = 3\} + \cdots] = 18.4$
	1	$35 + 50 \max\{(D - 2), 0\}$	$35 + 50[1P\{D = 3\} + 2P\{D = 4\} + \cdots] = 40.2$
	2	$60 + 50 \max\{(D - 3), 0\}$	$60 + 50[1P\{D = 4\} + 2P\{D = 5\} + \cdots] = 61.2$
	3	Decision 3 not permitted	
2	0	$50 \max\{(D - 2), 0\}$	$50[1P\{D = 3\} + 2P\{D = 4\} + \cdots] = 5.2$
	1	$35 + 50 \max\{(D - 3), 0\}$	$35 + 50[1P\{D = 4\} + 2P\{D = 5\} + \cdots] = 36.2$
	2, 3	Decisions 2, 3 not permitted	
3	0	$50 \max\{(D - 3), 0\}$	$50[1P\{D = 4\} + 2P\{D = 5\} + \cdots] = 1.2$
	1, 2, 3	Decisions 1, 2, 3 not permitted	

Choose the (s, S) policy already introduced as the initial policy for carrying out the value-determination step (step 1) of the *policy improvement algorithm*. This policy, R_1, calls for ordering up to 3 units whenever the system is in state 0 (no cameras on hand); otherwise, no order is placed. With this policy, the following four equations must be solved simultaneously for $g(R_1)$, $v_0(R_1)$, $v_1(R_1)$, and $v_2(R_1)$ [recall that $v_3(R_1)$ is arbitrarily taken to be zero]:

$$g(R_1) = C_{0k_1} + \sum_{j=0}^{3} p_{0j}(k_1)v_j(R_1) - v_0(R_1)$$

$$= C_{1k_1} + \sum_{j=0}^{3} p_{1j}(k_1)v_j(R_1) - v_1(R_1)$$

$$= C_{2k_1} + \sum_{j=0}^{3} p_{2j}(k_1)v_j(R_1) - v_2(R_1)$$

$$= C_{3k_1} + \sum_{j=0}^{3} p_{3j}(k_1)v_j(R_1) - v_3(R_1),$$

$R_1 = [3000]$
$(s, S) = (1, 3)$

or, alternatively,

$$g(R_1) = 86.2 + 0.080v_0(R_1) + 0.184v_1(R_1) + 0.368v_2(R_1) - v_0(R_1)$$

$$= 18.4 + 0.632v_0(R_1) + 0.368v_1(R_1) - v_1(R_1)$$

$$= 5.2 + 0.264v_0(R_1) + 0.368v_1(R_1) + 0.368v_2(R_1) - v_2(R_1)$$

$$= 1.2 + 0.080v_0(R_1) + 0.184v_1(R_1) + 0.368v_2(R_1).$$

The simultaneous solution of this system of equations yields

$$g_1(R_1) = 31.43$$

$$v_0(R_1) = 85.00$$

$$v_1(R_1) = 64.38$$

$$v_2(R_1) = 31.49.$$

Step 2 can now be applied. It is necessary to find the improved policy R_2, which has the property that $d_0(R_2) = k_2^0$, $d_1(R_2) = k_2^1$, $d_2(R_2) = k_2^2$, and $d_3(R_2) = k_2^3$ minimizes the following expressions:

(0) $C_{0k_2^0} + p_{00}(k_2^0)85 + p_{01}(k_2^0)64.38 + p_{02}(k_2^0)31.49 - 85$

(1) $C_{1k_2^1} + p_{10}(k_2^1)85 + p_{11}(k_2^1)64.38 + p_{12}(k_2^1)31.49 - 64.38$

(2) $C_{2k_2^2} + p_{20}(k_2^2)85 + p_{21}(k_2^2)64.38 + p_{22}(k_2^2)31.49 - 31.49$

(3) $C_{3k_2^3} + p_{30}(k_2^3)85 + p_{31}(k_2^3)64.38 + p_{32}(k_2^3)31.49.$

To find the optimal decisions, the following data are required:

State 0

Decision	$p_{00}(k_2)$	$p_{01}(k_2)$	$p_{02}(k_2)$	C_{0k_2}	Total Value of Expression 0
0	1	0	0	50	50
1	0.632	0.368	0	53.4	45.81
2	0.264	0.368	0.368	65.2	37.92
3	0.080	0.184	0.368	86.2	31.43

State 1

Decision	$p_{10}(k_2)$	$p_{11}(k_2)$	$p_{12}(k_2)$	C_{1k_2}	Total Value of Expression 1
0	0.632	0.368	0	18.4	31.43
1	0.264	0.368	0.368	40.2	33.54
2	0.080	0.184	0.368	61.2	27.05

State 2

Decision	$p_{20}(k_2)$	$p_{21}(k_2)$	$p_{22}(k_2)$	C_{2k_2}	Total Value of Expression 2
0	0.264	0.368	0.368	5.2	31.43
1	0.080	0.184	0.368	36.2	34.94

State 3

Decision	$p_{30}(k_2)$	$p_{31}(k_2)$	$p_{32}(k_2)$	C_{3k_2}	Total Value of Expression 3
0	0.080	0.184	0.368	1.2	31.43

Thus $d_0(R_2) = k_2^0 = 3$, $d_1(R_2) = k_2^1 = 2$, $d_2(R_2) = k_2^2 = d_3(R_2) = k_2^3 = 0$. Hence policy R_2 calls for ordering up to three cameras whenever there is 0 or 1 camera in stock; otherwise, no ordering is done; i.e., if the number of cameras on hand at the end of the week is less than $s = 2$ cameras, the store orders up to $S = 3$ cameras. Because policy R_2 differs from policy R_1, another iteration is required. The following four equations must be solved simultaneously for $g(R_2)$, $v_0(R_2)$, $v_1(R_2)$, and $v_2(R_2)$:

$$\begin{aligned} g(R_2) &= 86.2 + 0.080v_0(R_2) + 0.184v_1(R_2) + 0.368v_2(R_2) - v_0(R_2) \\ &= 61.2 + 0.080v_0(R_2) + 0.184v_1(R_2) + 0.368v_2(R_2) - v_1(R_2) \\ &= 5.2 + 0.264v_0(R_2) + 0.368v_1(R_2) + 0.368v_2(R_2) - v_2(R_2) \\ &= 1.2 + 0.080v_0(R_2) + 0.184v_1(R_2) + 0.368v_2(R_2). \end{aligned}$$

The simultaneous solution of this system of equations yields

$$g_1(R_2) = 30.33$$

$$v_0(R_2) = 85.00$$

$$v_1(R_2) = 60.00$$

$$v_2(R_2) = 30.68.$$

Step 2 can now be applied. It is necessary to find the improved policy R_3, which has the property that $d_0(R_3) = k_3^0$, $d_1(R_3) = k_3^1$, $d_2(R_3) = k_3^2$, and $d_3(R_3) = k_3^3$ minimizes the following expressions:

(0) $C_{0k_3^0} + p_{00}(k_3^0)85 + p_{01}(k_3^0)60 + p_{02}(k_3^0)30.68 - 85$

(1) $C_{1k_3^1} + p_{10}(k_3^1)85 + p_{11}(k_3^1)60 + p_{12}(k_3^1)30.68 - 60$

(2) $C_{2k_3^2} + p_{20}(k_3^2)85 + p_{21}(k_3^2)60 + p_{22}(k_3^2)30.68 - 30.68$

(3) $C_{3k_3^3} + p_{30}(k_3^3)85 + p_{31}(k_3^3)60 + p_{32}(k_3^3)30.68.$

Using the data from the previous iteration, the relevant calculations are

Decision	Total Value of Expression 0	Total Value of Expression 1	Total Value of Expression 2	Total Value of Expression 3
0	50	34.20	30.33	30.33
1	44.20	36.01	34.65	—
2	36.01	30.33	—	—
3	30.33	—	—	—

Thus $d_0(R_3) = k_3^0 = 3$, $d_1(R_3) = k_3^1 = 2$, $d_2(R_3) = k_3^2 = d_3(R_3) = k_3^3 = 0$. Hence policy R_3 and policy R_2 are identical, so that the optimal policy calls for ordering up to three cameras when there is 0 or 1 camera in stock; otherwise, no ordering is done.

The *linear programming formulation* calls for finding the y_{ik} that

$$\text{Minimize} \quad 50y_{00} + 53.4y_{01} + 65.2y_{02} + 86.2y_{03} + 18.4y_{10} + 40.2y_{11} + 61.2y_{12} + 5.2y_{20} + 36.2y_{21} + 1.2y_{30},$$

subject to

$$y_{00} + y_{01} + y_{02} + y_{03} + y_{10} + y_{11} + y_{12} + y_{20} + y_{21} + y_{30} = 1,$$

$$y_{00} + y_{01} + y_{02} + y_{03} - [y_{00} + y_{01}(0.632) + y_{02}(0.264) + y_{03}(0.080) + y_{10}(0.632) + y_{11}(0.264) + y_{12}(0.080) + y_{20}(0.264) + y_{21}(0.080) + y_{30}(0.080)] = 0,$$

$$y_{10} + y_{11} + y_{12} - [y_{01}(0.368) + y_{02}(0.368) + y_{03}(0.184) + y_{10}(0.368) + y_{11}(0.368) + y_{12}(0.184) + y_{20}(0.368) + y_{21}(0.184) + y_{30}(0.184)] = 0,$$

$$y_{20} + y_{21} - [y_{02}(0.368) + y_{03}(0.368) + y_{11}(0.368) + y_{12}(0.368) + y_{20}(0.368) + y_{21}(0.368) + y_{30}(0.368)] = 0,$$

$$y_{30} - [y_{03}(0.368) + y_{12}(0.368) + y_{21}(0.368) + y_{30}(0.368)] = 0,$$

and

$$y_{00}, y_{01}, y_{02}, y_{03}, y_{10}, y_{11}, y_{12}, y_{20}, y_{21}, y_{30} \geq 0.$$

This linear program can be solved by using the simplex method. The results yield all y_{ik} equal to zero, except for

$$y_{03} = 0.148, \quad y_{12} = 0.252, \quad y_{20} = 0.368, \quad y_{30} = 0.233.$$

The corresponding D_{ik} are given by

$$D_{03} = D_{12} = D_{20} = D_{30} = 1,$$

and all the remaining $D_{ik} = 0$.

20.8 Conclusions

The material presented in this chapter represents a powerful tool for formulating models and finding the optimal policies for controlling a large *class* of systems—those that are *Markovian decision processes*. These techniques are applicable to the solution of problems in such areas as queueing theory, inventory, maintenance, and probabilistic dynamic programming, in general.

Two algorithms were presented, the *policy improvement algorithm* and the *linear programming formulation*, for finding optimal policies. It is evident from the examples that data-collection requirements are high. Even if the solution converges rapidly in the policy improvement algorithm, completing step 2 requires considerable calculation for systems with a large number of states. Using the linear programming formulation with, say, 50 states and 25 decisions leads to 1,250 variables and 51 constraints (excluding the nonnegativity constraints), which represents a large linear program. Nevertheless, these two solution methods are useful for solving real-world problems

Chapter 18

OR by
Hamdy A. Taha

ɔn Problem:
ming Model
)iscounting
ɔunting
f the
hains

gramming to the solution of a
a finite number of states. The
ɔed by a Markov chain. The
matrix whose individual ele-
n moving from one state to
pend on the decision alterna-
f the problem is to determine
ue of the process over a finite

18.1 SCOPE OF THE MARKOVIAN DECISION PROBLEM: THE GARDENER EXAMPLE†

In this section we introduce a simple example that will be used as a vehicle of explanation throughout the chapter. In spite of its simplicity, the example paraphrases a number of important applications in the areas of inventory, replacement, cash flow management, and regulation of water reservoir capacity.

Every year, at the beginning of the gardening season, a gardener applies chemical tests to check the soil's condition. Depending on the outcomes of the tests, the garden's productivity for the new season is classified as good, fair, or poor.

Over the years, the gardener observed that current year's productivity can be assumed to depend only on last year's soil condition. The transition probabilities over a 1-year period from one productivity state to another can thus be represented in terms of the following Markov chain:

$$\begin{array}{rc} & \overbrace{\begin{array}{ccc} 1 & 2 & 3 \end{array}}^{\text{State of the system next year}} \\ \text{State of the system this year} \begin{cases} 1 \\ 2 \\ 3 \end{cases} & \begin{bmatrix} .2 & .5 & .3 \\ 0 & .5 & .5 \\ 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{P}^1$$

The representation assumes the following correspondence between productivity and the states of the chain:

Productivity (Soil Condition)	State of the System
Good	1
Fair	2
Poor	3

The transition probabilities in $\mathbf{P}^1$ indicate that the productivity for a current year can be no better than last year's. For example, if the soil condition for this year is fair (state 2), next year's productivity may remain fair with probability .5 or become poor (state 3), also with probability .5.

The gardener can alter the transition probabilities $\mathbf{P}^1$ by invoking other courses of action. Typically, fertilizer is applied to boost the soil condition, which yields the following transition matrix $\mathbf{P}^2$:

$$\mathbf{P}^2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} .3 & .6 & .1 \\ .1 & .6 & .3 \\ .05 & .4 & .55 \end{bmatrix} \end{array}$$

With the application of the fertilizer, it is possible to improve the condition of soil over last year's.

† A review of Markov chains is given in Section 18.6.

To put the decision problem in perspective, the gardener associates a return function (or a reward structure) with the transition from one state to another. The return function expresses the gain or loss during a 1-year period, depending on the states between which the transition is made. Since the gardener has the options of using or not using fertilizer, gain and losses are expected to vary depending on the decision made. The matrices $\mathbf{R}^1$ and $\mathbf{R}^2$ summarize the return functions in hundreds of dollars associated with the matrices $\mathbf{P}^1$ and $\mathbf{P}^2$, respectively. Thus $\mathbf{R}^1$ applies when no fertilizer is used; otherwise, $\mathbf{R}^2$ is utilized in the representation of the return function.

$$\mathbf{R}^1 = \|r_{ij}^1\| = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 7 & 6 & 3 \\ 0 & 5 & 1 \\ 0 & 0 & -1 \end{bmatrix} \end{array}$$

$$\mathbf{R}^2 = \|r_{ij}^2\| = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 6 & 5 & -1 \\ 7 & 4 & 0 \\ 6 & 3 & -2 \end{bmatrix} \end{array}$$

Notice that the elements r_{ij}^2 of $\mathbf{R}^2$ take into account the cost of applying the fertilizer. For example, if the system is in state 1 and remains in state 1 during next year, its gain will be $r_{11}^2 = 6$compared to $r_{11}^1 = 7$when no fertilizer is used.

What kind of a decision problem does the gardener have? First, we must know whether the gardening activity will continue for a limited number of years or, for all practical purposes, indefinitely. These situations are referred to as **finite-stage** and **infinite-stage** decision problems. In both cases, the gardener would need to determine the *best* course of action to be followed (fertilize or do not fertilize) given the outcome of the chemical tests (state of the system). The optimization process will be based on maximization of expected revenue.

The gardener may also be interested in evaluating the expected revenue resulting from following a prespecified course of action whenever a given state of the system occurs. For example, fertilizer may be applied whenever the soil condition is poor (state 3). The decision-making process in this case is said to be represented by a **stationary policy**.

We must note that each stationary policy will be associated with a different transition and return matrices, which, in general, can be constructed from the matrices $\mathbf{P}^1$, $\mathbf{P}^2$, $\mathbf{R}^1$, and $\mathbf{R}^2$. For example, for the stationary policy calling for applying fertilizer only when the soil condition is poor (state 3), the resulting transition and return matrices, $\mathbf{P}$ and $\mathbf{R}$, respectively, are given as

$$\mathbf{P} = \begin{bmatrix} .2 & .5 & .3 \\ 0 & .5 & .5 \\ .05 & .4 & .55 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 7 & 6 & 3 \\ 0 & 5 & 1 \\ 6 & 3 & -2 \end{bmatrix}$$

These matrices differ from $\mathbf{P}^1$ and $\mathbf{R}^1$ in the third rows only, which are taken directly from $\mathbf{P}^2$ and $\mathbf{R}^2$. The reason is that $\mathbf{P}^2$ and $\mathbf{R}^2$ are the matrices that result when fertilizer is applied in *every* state.

Exercise 18.1-1

(a) Identify the matrices $\mathbf{P}$ and $\mathbf{R}$ associated with the stationary policy calling for using fertilizer whenever the soil condition is fair or poor.

Problem 9.1 (page 252) - Ref: Applied Probability & Stochastic Processes by Feldman & Valdez-Flores

HW

Let $X = \{X_0, X_1, \cdots\}$ be a stochastic process with a four-state state space $E = \{a, b, c, d\}$. This process will represent a machine that can be in one of four operating conditions, denoted by the states a through d, indicating increasing levels of deterioration. As the machine deteriorates, not only is it more expensive to operate, but also production is lost. Standard maintenance activities are always carried out in states b through d so that the machine may improve due to maintenance; however, improvement is not guaranteed. In addition to the state space, there is an *action space* that gives the decisions possible at each step. (We sometimes use the words "decisions" and "actions" interchangeably when referring to the elements of the action space.) In this example we shall assume the action space is $A = \{1, 2\}$; that is, at each step there are two possible actions: use an inexperienced operator (action 1) or use an experienced operator (action 2). To complete the description of a Markov decision problem, we need a cost vector and a transition matrix for each possible action in the action space. For our example, define the two cost vectors[2] and two Markov matrices as

$$\boldsymbol{f}_1 = (100, 125, 150, 500)^T,$$
$$\boldsymbol{f}_2 = (300, 325, 350, 600)^T,$$

$$\boldsymbol{P}_1 = \begin{bmatrix} 0.1 & 0.3 & 0.6 & 0.0 \\ 0.0 & 0.2 & 0.5 & 0.3 \\ 0.0 & 0.1 & 0.2 & 0.7 \\ 0.8 & 0.1 & 0.0 & 0.1 \end{bmatrix},$$

$$\boldsymbol{P}_2 = \begin{bmatrix} 0.6 & 0.3 & 0.1 & 0.0 \\ 0.75 & 0.1 & 0.1 & 0.05 \\ 0.8 & 0.2 & 0.0 & 0.0 \\ 0.9 & 0.1 & 0.0 & 0.0 \end{bmatrix}.$$

The dynamics of the process ~~are illustrated in Figure 9.1 and are~~ are as follows: If, at time n, the process is in state i and the decision k is made, then a cost of $f_k(i)$ is incurred and the probability that the next state will be j is given by $P_k(i, j)$. To illustrate, if $X_n = a$ and decision 1 is made, then a cost of \$100 is incurred (representing the operator cost, lost production cost, and machine operation cost) and $\Pr\{X_{n+1} = a\} = 0.1$; or, if $X_n = d$ and decision 2 is made, then a cost of \$600 is incurred (representing the operator cost, machine operation cost, major maintenance cost, and lost-production cost) and $\Pr\{X_{n+1} = a\} = 0.9$. ■

Returning to the maintenance problem given ~~in Example 9.1~~ above, we wish to maximize profits. Profits are determined by revenue minus costs, where revenue is a function of the state but not of the decision and is given by the function $\boldsymbol{r} = (900, 400, 450, 750)^T$. The costs and transition probabilities are as presented in the example.

(a) Compute the profit function $\boldsymbol{g}_1$ associated with the stationary policy that uses action 1 in every state.

(b) Using a discount factor $\alpha = 0.95$, verify that the vector $\boldsymbol{v}^\alpha = (8651.88, 8199.73, 8233.37, 8402.65)^T$ is the optimal value that maximizes the total discounted profit. ~~(You need to use Property 9.6 with a minor modification because this is a maximizing problem.)~~

(c) Using a discount factor of $\alpha = 0.7$, use the value iteration algorithm to find the optimal value function that maximizes the total discounted profit.

(d) Using the policy improvement algorithm, find the policy that maximizes the total discounted profit if the discount factor is such that \$1 today is worth \$1.12 after one time period.