SUPPLEMENT 2 TO CHAPTER 19

A Discounted Cost Criterion

Throughout Chap. 19 we have measured policies on the basis of their (long-run) expected average cost per unit time. We now turn to an alternative measure of performance, namely, the **expected total discounted cost.**

As first introduced in Sec. 18.2, this measure uses a *discount factor* α, where $0 < \alpha < 1$. The discount factor α can be interpreted as equal to $1/(1 + i)$, where i is the current interest rate per period. Thus, α is the *present value* of one unit of cost one period in the future. Similarly, α^m is the *present value* of one unit of cost m periods in the future.

This *discounted cost criterion* becomes preferable to the *average cost criterion* when the time periods for the Markov chain are sufficiently long that the *time value of money* should be taken into account in adding costs in future periods to the cost in the current period. Another advantage is that the discounted cost criterion can readily be adapted to dealing with a *finite-period* Markov decision process where the Markov chain will terminate after a certain number of periods.

Both the policy improvement technique (see Supplement 1) and the linear programming approach (see Sec. 19.3) still can be applied here with relatively minor adjustments from the average cost case, as we describe next. Then we will present another technique, called the method of successive approximations, for quickly approximating an optimal policy.

A Policy Improvement Algorithm

To derive the expressions needed for the value determination and policy improvement steps of the algorithm, we now adopt the viewpoint of *probabilistic dynamic programming* (as described in Sec. 11.4). In particular, for each state i ($i = 0, 1, \ldots, M$) of a Markov decision process operating under policy R, let $V_i^n(R)$ be the *expected total discounted cost* when the process starts in state i (beginning the first observed time period) and evolves for n time periods. Then $V_i^n(R)$ has two components: C_{ik}, the cost incurred during the first observed time period, and $\alpha \sum_{j=0}^{M} p_{ij}(k)\, V_j^{n-1}(R)$, the expected total discounted cost of the process evolving over the remaining $n - 1$ time periods. For each $i = 0, 1, \ldots, M$, this yields the recursive equation

8

$$V_i^n(R) = C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k) V_j^{n-1}(R),$$

with $V_i^1(R) = C_{ik}$, which closely resembles the recursive relationships of probabilistic dynamic programming found in Sec. 11.4.

As n approaches infinity, this recursive equation converges to

$$V_i(R) = C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k) V_j(R), \quad \text{for } i = 0, 1, \ldots, M,$$

where $V_i(R)$ can now be interpreted as the expected total discounted cost when the process starts in state i and continues indefinitely. There are $M + 1$ equations and $M + 1$ unknowns, so the simultaneous solution of this system of equations yields the $V_i(R)$.

To illustrate, consider again the prototype example of Sec. 19.1. Under the average cost criterion, we found in Secs. 19.2 and 19.3, as well as Supplement 1, that the optimal policy is to do nothing in states 0 and 1, overhaul in state 2, and replace in state 3. Under the discounted cost criterion, with $\alpha = 0.9$, this same policy gives the following system of equations:

$$V_0(R) = + 0.9\left[\frac{7}{8}V_1(R) + \frac{1}{16}V_2(R) + \frac{1}{16}V_3(R)\right]$$

$$V_1(R) = 1{,}000 + 0.9\left[\frac{3}{4}V_1(R) + \frac{1}{8}V_2(R) + \frac{1}{8}V_3(R)\right]$$

$$V_2(R) = 4{,}000 + 0.9[\quad V_1(R)]$$

$$V_3(R) = 6{,}000 + 0.9[V_0(R)].$$

The simultaneous solution is

$$V_0(R) = 14{,}949$$
$$V_1(R) = 16{,}262$$
$$V_2(R) = 18{,}636$$
$$V_3(R) = 19{,}454.$$

Thus, assuming that the system starts in state 0, the expected total discounted cost is \$14,949.

This system of equations provides the expressions needed for a policy improvement algorithm. After summarizing this algorithm in general terms, we shall use it to check whether this particular policy still is optimal under the discounted cost criterion.

Summary of the Policy Improvement Algorithm (Discounted Cost Criterion)

Initialization: Choose an arbitrary initial trial policy R_1. Set $n = 1$.
Iteration n:
Step 1: Value determination: For policy R_n, use $p_{ij}(k)$ and C_{ik} to solve the system of $M + 1$ equations

$$V_i(R_n) = C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k) V_j(R_n), \quad \text{for } i = 0, 1, \ldots, M,$$

for all $M + 1$ unknown values of $V_0(R_n), V_1(R_n), \ldots, V_M(R_n)$.

Step 2: Policy improvement: Using the current values of the $V_i(R_n)$, find the alternative policy R_{n+1} such that, for each state *i*, $d_i(R_{n+1}) = k$ is the decision that minimizes

$$C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k)\, V_j(R_n),$$

i.e., for *each* state *i,*

$$\underset{k=1,2,\ldots,K}{\text{Minimize}} \left[C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k)\, V_j(R_n) \right],$$

and then set $d_i(R_{n+1})$ equal to the minimizing value of *k.* This procedure defines a new policy R_{n+1}.

Optimality test: The current policy R_{n+1} is optimal if this policy is identical to policy R_n. If it is, stop. Otherwise, reset $n = n + 1$ and perform another iteration.

Three key properties of this algorithm are

1. $V_i(R_{n+1}) \leq V_i(R_n)$, for $i = 0, 1, \ldots, M$ and $n = 1, 2, \ldots$.
2. The algorithm terminates with an optimal policy in a finite number of iterations.
3. The algorithm is valid without the assumption (used for the average cost case) that the Markov chain associated with every transition matrix is irreducible.

Your IOR Tutorial includes an *interactive* procedure for applying this algorithm.

Solving the Prototype Example by This Policy Improvement Algorithm. We now pick up the prototype example where we left it before summarizing the algorithm.

We already have selected the optimal policy under the average cost criterion to be our initial trial policy R_1. This policy, its transition matrix, and its costs are summarized below:

Policy R_1

State	Decision
0	1
1	1
2	2
3	3

Transition matrix

State	0	1	2	3
0	0	$\frac{7}{8}$	$\frac{1}{16}$	$\frac{1}{16}$
1	0	$\frac{3}{4}$	$\frac{1}{8}$	$\frac{1}{8}$
2	0	1	0	0
3	1	0	0	0

Costs

State	Cik
0	0
1	1,000
2	4,000
3	6,000

We also have already done step 1 (value determination) of iteration 1. This transition matrix and these costs led to the system of equations used to find $V_0(R_1) = 14{,}949$, $V_1(R_1) = 16{,}262$, $V_2(R_1) = 18{,}636$, and $V_3(R_1) = 19{,}454$.

To start step 2 (policy improvement), we only need to construct the expression to be minimized for the two states (1 and 2) with a choice of decisions.

State 1: $C_{1k} + 0.9[p_{10}(k)(14{,}949) + p_{11}(k)(16{,}262) + p_{12}(k)(18{,}636) + p_{13}(k)(19{,}454)]$

State 2: $C_{2k} + 0.9[p_{20}(k)(14{,}949) + p_{21}(k)(16{,}262) + p_{22}(k)(18{,}636) + p_{23}(k)(19{,}454)]$.

For each of these states and their possible decisions, we show below the corresponding C_{ik}, the $P_{ij}(k)$, and the resulting value of the expression.

	State 1						
Decision	C_{1k}	$p_{10}(k)$	$p_{11}(k)$	$p_{12}(k)$	$p_{13}(k)$	**Value of Expression**	
1	1,000	0	$\frac{3}{4}$	$\frac{1}{8}$	$\frac{1}{8}$	16,262	← Minimum
3	6,000	1	0	0	0	19,454	

	State 2						
Decision	C_{2k}	$p_{20}(k)$	$p_{21}(k)$	$p_{22}(k)$	$p_{23}(k)$	**Value of Expression**	
1	3,000	0	0	$\frac{1}{2}$	$\frac{1}{2}$	20,140	
2	4,000	0	1	0	0	18,636	← Minimum
3	6,000	1	0	0	0	19,454	

Since decision 1 minimizes the expression for state 1 and decision 2 minimizes the expression for state 2, our next trial policy (R_2) is as follows:

Policy R_2	
State	**Decision**
0	1
1	1
2	2
3	3

Since this policy is identical to policy R_1, the optimality test indicates that this policy is optimal. Thus, the optimal policy under the average cost criterion also is optimal under the discounted cost criterion in this case. (This often occurs, but not always.)

Linear Programming Formulation

The linear programming formulation for the discounted cost case is similar to that for the average cost case given in Sec. 19.3. However, we no longer need the first constraint given in Sec. 19.3; but the other functional constraints do need to include the discount factor α. The other difference is that the model now contains constants β_j for $j = 0, 1, \ldots, M$. These constants must satisfy the conditions

$$\sum_{j=0}^{M} \beta_j = 1, \quad \beta_j > 0 \quad \text{for } j = 0, 1, \ldots, M,$$

but otherwise they can be chosen arbitrarily without affecting the optimal policy obtained from the model.

The resulting model is to choose the values of the *continuous* decision variables y_{ik} so as to

$$\text{Minimize} \quad Z = \sum_{i=0}^{M} \sum_{k=1}^{K} C_{ik} y_{ik},$$

subject to the constraints

$$(1) \quad \sum_{k=1}^{K} y_{jk} - \alpha \sum_{i=0}^{M} \sum_{k=1}^{K} y_{ik} p_{ij}(k) = \beta_j, \quad \text{for } j = 0, 1, \ldots, M,$$

$$(2) \quad y_{ik} \geq 0, \quad \text{for } i = 0, 1, \ldots, M; k = 1, 2, \ldots, K.$$

Once the simplex method is used to obtain an optimal solution for this model, the corresponding optimal policy then is defined by

$$D_{ik} = P\{\text{decision} = k \mid \text{state} = i\} = \frac{y_{ik}}{\sum_{k=1}^{K} y_{ik}}.$$

The y_{ik} now can be interpreted as the *discounted* expected time of being in state i and making decision k, when the probability distribution of the *initial state* (when observations begin) is $P\{X_0 = j\} = \beta_j$ for $j = 0, 1, \ldots, M$. In other words, if

$$z_{ik}^n = P\{\text{at time } n, \text{ state} = i \text{ and decision} = k\},$$

then

$$y_{ik} = z_{ik}^0 + \alpha z_{ik}^1 + \alpha^2 z_{ik}^2 + \alpha^3 z_{ik}^3 + \cdots.$$

With the interpretation of the β_j as *initial state probabilities* (with each probability greater than zero), Z can be interpreted as the corresponding expected total discounted cost. Thus, the choice of fij affects the optimal value of Z (but not the resulting optimal policy).

It again can be shown that the optimal policy obtained from solving the linear programming model is deterministic; that is, $D_{ik} = 0$ or 1. Furthermore, this technique is valid without the assumption (used for the average cost case) that the Markov chain associated with every transition matrix is irreducible.

Solving the Prototype Example by Linear Programming. The linear programming model for the prototype example (with $\alpha = 0.9$) is

$$\text{Minimize} \quad Z = 1{,}000y_{11} + 6{,}000y_{13} + 3{,}000y_{21} + 4{,}000y_{22} + 6{,}000y_{23} + 6{,}000y_{33},$$

subject to

$$y_{01} - 0.9(y_{13} + y_{23} + y_{33}) = \frac{1}{4}$$

$$y_{11} + y_{13} - 0.9\left(\frac{7}{8}y_{01} + \frac{3}{4}y_{11} + y_{22}\right) = \frac{1}{4}$$

$$y_{21} + y_{22} + y_{23} - 0.9\left(\frac{1}{16}y_{01} + \frac{1}{8}y_{11} + \frac{1}{2}y_{21}\right) = \frac{1}{4}$$

$$y_{33} - 0.9\left(\frac{1}{16}y_{01} + \frac{1}{8}y_{11} + \frac{1}{2}y_{21}\right) = \frac{1}{4}$$

and

$$\text{all } y_{ik} \geq 0,$$

where β_0, β_1, β_2, and β_3 are arbitrarily chosen to be $\frac{1}{4}$. By the simplex method, the optimal solution is

$$y_{01} = 1.210, \quad (y_{11}, y_{13}) = (6.656, 0), \quad (y_{21}, y_{22}, y_{23}) = (0, 1.067, 0), \quad y_{33} = 1.067,$$

so

$$D_{01} = 1, \quad (D_{11}, D_{13}) = (1, 0), \quad (D_{21}, D_{22}, D_{23}) = (0, 1, 0), \quad D_{33} = 1.$$

This optimal policy is the same as that obtained earlier in this supplement by the policy improvement algorithm.

The value of the objective function for the optimal solution is $Z = 17{,}325$. This value is closely related to the values of the $V_i(R)$ for this optimal policy that were obtained by the policy improvement algorithm. Recall that $V_i(R)$ is interpreted as the expected total discounted cost given that the system starts in state *i,* and we are interpreting β_i as the probability of starting in state *i.* Because each β_i was chosen to equal $\frac{1}{4}$,

$$17{,}325 = \frac{1}{4}[V_0(R) + V_1(R) + V_2(R) + V_3(R)]$$

$$= \frac{1}{4}(14{,}949 + 16{,}262 + 18{,}636 + 19{,}454).$$

Finite-Period Markov Decision Processes and the Method of Successive Approximations

We now turn our attention to an approach, called the *method of successive approximations,* for *quickly* finding at least an *approximation* to an optimal policy.

We have assumed so far that the Markov decision process will be operating indefinitely, and we have sought an optimal policy for such a process. The basic idea of the method of successive approximations is to instead find an optimal policy for the decisions to make in the first period when the process has only n time periods to go before termination, starting with $n = 1$, then $n = 2$, then $n = 3$, and so on. As n grows large, the corresponding optimal policies will converge to an optimal policy for the infinite-period problem of interest. Thus, the policies obtained for $n = 1, 2, 3, \ldots$ provide *successive approximations* that lead to the desired optimal policy.

The reason that this approach is attractive is that we already have a quick method of finding an optimal policy when the process has only n periods to go, namely, probabilistic dynamic programming as described in Sec. 11.4.

In particular, for $i = 0, 1, \ldots, M,$ let

V_i^n = expected total discounted cost of following an optimal policy, given that process starts in state i and has only n periods to go.[1]

By the *principle of optimality* for dynamic programming (see Sec. 11.2), the V_i^n are obtained from the recursive relationship

$$V_i^n = \min_k \left\{ C_{ik} + \alpha \sum_{j=0}^{M} p_{ij}(k)\, V_j^{n-1} \right\}, \quad \text{for } i = 0, 1, \ldots, M.$$

The minimizing value of k provides the optimal decision to make in the first period when the process starts in state *i.*

To get started, with $n = 1$, all the $V_i^0 = 0$ so that

$$V_i^1 = \min_k \{C_{ik}\}, \quad \text{for } i = 0, 1, \ldots, M.$$

Although the method of successive approximations may not lead to an optimal policy for the infinite-period problem after only a few iterations, it has one distinct advantage over the policy improvement and linear programming techniques. It never requires solving a system of simultaneous equations, so each iteration can be performed simply and quickly.

Furthermore, if the Markov decision process actually does have just n periods to go, n iterations of this method definitely will lead to an optimal policy. (For an n-period problem, it is permissible to set $\alpha = 1$, that is, no discounting, in which case the objective is to minimize the expected total cost over n periods.)

Your IOR Tutorial includes an interactive procedure to help guide you to use this method efficiently.

[1]Since we want to allow n to grow indefinitely, we are letting n be the *number of periods to go,* instead of the *number of periods from the beginning* (as in Chap. 11).

Solving the Prototype Example by the Method of Successive Approximations

We again use $\alpha = 0.9$. Refer to the rightmost column of Table 19.1 at the end of Sec. 19.1 for the values of C_{ik}. Also note in the first two columns of this table that the only feasible decisions k for each state i are $k = 1$ for $i = 0$, $k = 1$ or 3 for $i = 1$, $k = 1, 2$, or 3 for $i = 2$, and $k = 3$ for $i = 3$.

For the first iteration ($n = 1$), the value obtained for each V_i^1 is shown below, along with the minimizing value of k (given in parentheses).

$$V_0^1 = \min_{k=1} \{C_{0k}\} = 0 \qquad (k = 1)$$
$$V_1^1 = \min_{k=1,3} \{C_{1k}\} = 1{,}000 \qquad (k = 1)$$
$$V_2^1 = \min_{k=1,2,3} \{C_{2k}\} = 3{,}000 \qquad (k = 1)$$
$$V_3^1 = \min_{k=3} \{C_{3k}\} = 6{,}000 \qquad (k = 3)$$

Thus, the first approximation calls for making decision 1 (do nothing) when the system is in state 0, 1, or 2. When the system is in state 3, decision 3 (replace the machine) is made.

The second iteration leads to

$$V_0^2 = 0 + 0.9\left[\frac{7}{8}(1{,}000) + \frac{1}{16}(3{,}000) + \frac{1}{16}(6{,}000)\right] = 1{,}294 \ (k = 1)$$

$$V_1^2 = \min\left\{1{,}000 + 0.9\left[\frac{3}{4}(1{,}000) + \frac{1}{8}(3{,}000) + \frac{1}{8}(6{,}000)\right], \right.$$
$$\left. 6{,}000 + 0.9[1(0)]\right\} = 2{,}688 \ (k = 1)$$

$$V_2^2 = \min\left\{3{,}000 + 0.9\left[\frac{1}{2}(3{,}000) + \frac{1}{2}(6{,}000)\right], \right.$$
$$\left. 4{,}000 + 0.9[1(1{,}000)], 6{,}000 + 0.9(1(0))\right\} = 4{,}900 \ (k = 2)$$

$$V_3^2 = 6{,}000 + 0.9[1(0)] = 6{,}000 \ (k = 3).$$

where the *min* operator has been deleted from the first and fourth expressions because only one alternative for the decision is available. Thus, the second approximation calls for leaving the machine as is when it is in state 0 or 1, overhauling when it is in state 2, and replacing the machine when it is in state 3. Note that this policy is the optimal one for the infinite-period problem, as found earlier in this supplement by both the policy improvement algorithm and linear programming. However, the V_i^2 (the expected total discounted cost when starting in state i for the two-period problem) are not yet close to the V_i (the corresponding cost for the infinite-period problem).

The third iteration leads to

$$V_0^3 = 0 + 0.9\left[\frac{7}{8}(2{,}688) + \frac{1}{16}(4{,}900) + \frac{1}{16}(6{,}000)\right] = 2{,}730 \ (k = 1)$$

$$V_1^3 = \min\left\{1{,}000 + 0.9\left[\frac{3}{4}(2{,}688) + \frac{1}{8}(4{,}900) + \frac{1}{8}(6{,}000)\right],\right.$$
$$\left. 6{,}000 + 0.9[1(1{,}294)]\right\} = 4{,}041 \ (k = 1)$$

$$V_3^2 = \min\left\{3{,}000 + 0.9\left[\frac{1}{2}(4{,}900) + \frac{1}{2}(6{,}000)\right],\right.$$
$$\left. 4{,}000 + 0.9[1(2{,}688)], 6{,}000 + 0.9[1(1{,}294)]\right\} = 6{,}419 \ (k = 2)$$

$$V_3^3 = 6{,}000 + 0.9[1(1{,}294)] = 7{,}165 \ (k = 3).$$

Again the optimal policy for the infinite-period problem is obtained, and the costs are getting closer to those for that problem. This procedure can be continued, and V_0^n, V_1^n, V_2^n, and V_3^n will converge to 14,949, 16,262, 18,636, and 19,454, respectively.

Note that termination of the method of successive approximations after the second iteration would have resulted in an optimal policy for the infinite-period problem, although there is no way to know this fact without solving the problem by other methods.

As indicated earlier, the method of successive approximations definitely obtains an optimal policy for an n-period problem after n iterations. For this example, the first, second, and third iterations have identified the optimal immediate decision for each state if the remaining number of periods is one, two, and three, respectively.

LEARNING AIDS FOR THIS SUPPLEMENT ON THIS WEBSITE

Interactive Procedures in IOR Tutorial:

Enter Markov Decision Model
Interactive Policy Improvement Algorithm—Discounted Cost
Interactive Method of Successive Approximations

"Ch. 19—Markov Decision Proc" Files for Solving the Linear Programming Formulations:

Excel Files
LINGO/LINDO File

Glossary for Chapter 19

See Appendix 1 for documentation of the software.

PROBLEMS

The symbols to the left of some of the problems (or their parts) have the following meaning:

I: We suggest that you use the corresponding interactive procedure listed above (the printout records your work).

C: Use the computer with any of the software options available to you (or as instructed by your instructor) to solve your linear programming formulation.

I **19S2-1.** Joe wants to sell his car. He receives one offer each month and must decide immediately whether to accept the offer. Once rejected, the offer is lost. The possible offers are \$600, \$800, and \$1,000, made with probabilities $\frac{5}{8}$, $\frac{1}{4}$, and $\frac{1}{8}$, respectively (where successive offers are independent of each other). There is a maintenance cost of \$60 per month for the car. Joe is anxious to sell the car and so has chosen a discount factor of $\alpha = 0.95$.

Using the policy improvement algorithm, find a policy that minimizes the expected total discounted cost. *(Hint:* There are two actions: Accept or reject the offer. Let the state for month t be the offer in that month. Also include a state ∞, where the process goes to state ∞ whenever an offer is accepted and it remains there at a monthly cost of 0.)

19S2-2. Reconsider Prob. 19S2-1.

(a) Formulate a linear programming model for finding an optimal policy.

C **(b)** Use the simplex method to solve this model. Use the resulting optimal solution to identify an optimal policy.

I **19S2-3.** For Prob. 19S2-1, use three iterations of the method of successive approximations to approximate an optimal policy.

I **19S2-4.** The price of a certain stock is fluctuating between \$10, \$20, and \$30 from month to month. Market analysts have predicted that if the stock is at \$10 during any month, it will be at \$10 or \$20 the next month, with probabilities $\frac{4}{5}$ and $\frac{1}{5}$, respectively; if the stock is at \$20, it will be at \$10, \$20, or \$30 the next month, with probabilities $\frac{1}{4}$, $\frac{1}{4}$, and $\frac{1}{2}$, respectively; and if the stock is at \$30, it will be at \$20 or \$30 the next month, with probabilities $\frac{3}{4}$ and $\frac{1}{4}$, respectively. Given a discount factor of 0.9, use the policy improvement algorithm to determine when to sell and when to hold the stock to maximize the expected total discounted profit. *(Hint:* Include a state that is reached with probability 1 when the stock is sold and with probability 0 when the stock is held.)

19S2-5. Reconsider Prob. 19S2-4.

(a) Formulate a linear programming model for finding an optimal policy.

C **(b)** Use the simplex method to solve this model. Use the resulting optimal solution to identify an optimal policy.

I **19S2-6.** For Prob. 19S2-4, use three iterations of the method of successive approximations to approximate an optimal policy.

19S2-7. A chemical company produces two chemicals, denoted by C1 and C2, and only one can be produced at a time. Each month a decision is made as to which chemical to produce that month. Because the demand for each chemical is predictable, it is known that if C2 is produced this month, there is a 60 percent chance that it will also be produced again next month. Similarly, if C1 is produced this month, there is only a 30 percent chance that it will be produced again next month.

To combat the emissions of pollutants, the chemical company has two processes, process *A,* which is efficient in combating the pollution from the production of C2 but not from C1, and process *B,* which is efficient in combating the pollution from the production of C1 but not from C2. Only one process can be used at a time. The amount of pollution from the production of each chemical under each process is

	C1	C2
A	15	2
B	3	8

Unfortunately, there is a time delay in setting up the pollution control processes, so that a decision as to which process to use must be made in the month prior to the production decision. Management wants to determine a policy for when to use each pollution control process that will minimize the expected total discounted amount of all future pollution with a discount factor of $\alpha = 0.5$.

(a) Formulate this problem as a Markov decision process by identifying the states, the decisions, and the C_{ik}. Identify all the (stationary deterministic) policies.

I **(b)** Use the policy improvement algorithm to find an optimal policy.

19S2-8. Reconsider Prob. 19S2-7.

(a) Formulate a linear programming model for finding an optimal policy.

C **(b)** Use the simplex method to solve this model. Use the resulting optimal solution to identify an optimal policy.

I **19S2-9.** For Prob. 19S2-7, use two iterations of the method of successive approximations to approximate an optimal policy.

I **19S2-10.** Reconsider Prob. 19S2-7. Suppose now that the company will be producing either of these chemicals for only 4 more months, so a decision on which pollution control process to use 1 month hence only needs to be made three more times. Find an optimal policy for this three-period problem.

I **19S2-11** Reconsider the prototype example of Sec. 19.1. Suppose now that the production process using the machine under consideration will be used for only 4 more weeks. Using the discounted cost criterion with a discount factor of $\alpha = 0.9$, find the optimal policy for this four-period problem.