Average Cost Markov Decision Processes: Policy-Iteration and Value-Iteration

This material is an addendum to Section 10.4 in the book *Operations Research: Introduction to Models and Methods* by Boucherie et. al. Section 10.4 in the book considers only the linear programming approach for the long-run average reward criterion in Markovian control. This addendum discusses the policy-iteration method and the value-iteration method for the finite-state Markov decision model with the long-run average cost per unit time as optimality criterion. The basic concept of relative values is first introduced. Next it is shown how the policy-improvement step uses the relative values in order to improve a given policy. This leads to the policy-iteration algorithm. The idea of the policy-improvement step is flexible and powerful. It enables us to give bounds on the average costs of the policies generated in the value-iteration algorithm.

1.1. Notation

The notation is slightly different from the notation used in the book.[1] A dynamic system is reviewed at equidistant points of time $t = 0, 1, \ldots$. At each review the system is classified into one of a possible number of states and subsequently a decision has to be made. The set of possible states is denoted by I. For each state $i \in I$, a set $A(i)$ of decisions or actions is given. The state space I and the action sets $A(i)$ are assumed to be *finite*. The economic consequences of the decisions taken at the review times (decision epochs) are reflected in costs. This controlled dynamic system is called a *discrete-time Markov model* when the following Markovian property is satisfied. If at a decision epoch the action a is chosen in state i, then regardless of the past history of the system, the following happens:

(**a**) an immediate cost $c_i(a)$ is incurred,
(**b**) at the next decision epoch the system will be in state j with probability $p_{ij}(a)$, where $\sum_{j\in I} p_{ij}(a) = 1$ for $i \in I$.

The one-step costs $c_i(a)$ and the one-step transition probabilities $p_{ij}(a)$ are assumed to be time homogeneous. In specific problems the ‘immediate’ costs $c_i(a)$ will often represent the expected cost incurred until the next decision epoch when action a is chosen in state i. The choice of the state space and of the action sets often depends on the cost structure of the specific problem considered.

[1] The notation $p_{ij}(a)$ is used rather than $p(j \mid i, a)$ for the transition probabilities, and costs $c_i(a)$ rather than rewards $r(i, a)$. The stationary policies are denoted by R rather than δ.

A rule or policy for controlling the system is a prescription for taking actions at each decision epoch. In principle a control rule may be quite complicated in the sense that the prescribed actions may depend on the whole history of the system. An important class of policies is the subclass of stationary policies. A *stationary policy* R is a policy that assigns to each state i a fixed action $a = R_i$ and always uses this action whenever the system is in state i. In view of the Markov assumption made and the fact that the planning horizon is infinitely long, it will be intuitively clear that it is sufficient to consider only the class of stationary policies. However, other policies are conceivable: policies whose actions depend on the past states or policies whose actions are determined by a random mechanism. This issue touches a fundamental question in Markov decision theory: does there exist an optimal policy among the class of all conceivable policies and, if an optimal policy exists, is such a policy a stationary policy? The answer to these questions is in the affirmative for the finite-state and finite-action Markov decision model with the average cost criterion. However, a mathematical proof requires rather deep arguments.

Let X_n be the state of the system at time n just prior to a decision. Under a stationary policy R, the process $\{X_n\}$ is a discrete-time Markov chain with one-step transition probabilities

$$P\{X_{n+1} = j \mid X_n = i\} = p_{ij}(R_i).$$

This Markov chain incurs a cost $c_i(R_i)$ each time the system visits state i. Thus we can invoke results from Markov chain theory to specify the long-run average cost per time unit under a given stationary policy. Unless stated otherwise, the following assumption is made throughout this chapter.

Unichain assumption. *For each stationary policy R, the associated Markov chain $\{X_n\}$ has a single recurrent class of states.*

In most practical applications the Markov chain $\{X_n\}$ is unichain for each stationary policy. The unichain assumption allows for transient states. Under this assumption, the Markov chain $\{X_n\}$ has a unique equilibrium distribution for each stationary policy. The equilibrium probabilities under stationary policy R are denoted by $\pi_j(R)$ for $j \in I$ and are the unique solution to the equilibrium equations

$$\pi_j(R) = \sum_{k \in I} \pi_k(R) p_{kj}(R_k) \quad \text{for } j \in I$$

together with the normalization equation $\sum_{j \in I} \pi_j(R) = 1$. By a well-known ergodic theorem from Markov chain theory, the long-run average cost per unit time under a stationary policy R is given by

$$g(R) = \sum_{j \in I} c_j(R_j)\pi_j(R),$$

independently of the starting state of the Markov chain. A stationary policy R^* is said to be *average cost optimal* if $g(R^*) \leq g(R)$ for each stationary policy R. As pointed out before, policy R^* is not only optimal among the class of stationary policies but it is also optimal among the class of all conceivable policies.

2. The concept of relative values

There is an alternative approach to compute the average cost $g(R)$ of a given stationary policy R. This approach yields a so-called relative-value function which is the basis for an improvement of policy R. The so-called *value-determination equations* for policy R are defined by

$$v_i = c_i(R_i) - g + \sum_{j \in I} p_{ij}(R_i)v_j \quad \textit{for } i \in I.$$

Theorem 1. *For any policy R, it holds under the unichain assumption*:
(**a**) *The value-determination equations are solvable.*
(**b**) *Each solution to the value-determination equations satisfies $g = g(R)$. The relative-value function $v_i, i \in I$ is uniquely determined up to an additive constant.*

Proof. (**a**) A Markov chain is a regenerative stochastic process and any recurrent state is a regeneration state. Take any recurrent state r. Define $T_i(R)$ as the expected time until the first transition into state r after epoch 0 when the initial state is i and policy R is used, and define $K_i(R)$ as the expected costs incurred during this time including the cost incurred in the initial state i at epoch 0 but excluding the cost incurred at the epoch of the first transition into state r. Also, define the function $w_i(R)$ by

$$w_i(R) = K_i(R) - g(R)T_i(R) \quad \text{for } i \in I,$$

Letting a cycle be the time elapsed between two consecutive visits to the regeneration state r, it follows from the theory of renewal-reward processes that the average cost per time unit equals the expected costs incurred in one cycle divided by the expected length of one cycle and so $g(R) = \frac{K_r(R)}{T_r(R)}$, implying that $w_r(R) = 0$. By a conditioning argument,

$$\begin{aligned} T_i(R) &= 1 + \sum_{j \in I, j \neq r} T_j(R)p_{ij}(R_i) \quad \text{for } i \in I, \\ K_i(R) &= c_i(R_i) + \sum_{j \in I, j \neq r} K_j(R)p_{ij}(R_i) \quad \text{for } i \in I. \end{aligned}$$

Multiplying both sides of the first equation with $g(R)$ and subtracting the resulting two equations, we get $w_i(R) = c_i(R_i) - g(R) + \sum_{j\in I, j\neq r} w_j(R)p_{ij}(R_i)$ for $i \in I$. Noting that $w_r(R) = 0$, we have verified that $g(R)$ and $w_i(R)$, $i \in I$ satisfy the value-determination equations.
(**b**) Let g and v_i, $i \in I$ by any solution to the value-determination equations. Multiplying both sides of the equations by $\pi_i(R)$, summing over i and using the equilibrium equations for the $\pi_i(R)$, it follows after an interchange of the order of summation that $g = \sum_{i\in I} c_i(R_i)\pi_i(R)$, showing that $g = g(R)$. To prove that the relative-value function is uniquely determined up to an additive constant, let $\{g, v_i\}$ and $\{g, w_i\}$ be any two solutions to the value-determination equations. It is convenient to use matrix notation with $\mathbf{v} = (v_i)$, $\mathbf{w} = (w_i)$ and $\mathbf{P} = (p_{ij}(R_i))$. Then $\mathbf{v} - \mathbf{w} = \mathbf{P}(\mathbf{v} - \mathbf{w})$. Iterating this equation, we get $\mathbf{v} - \mathbf{w} = \mathbf{P}^n(\mathbf{v} - \mathbf{w})$ for all $n \geq 1$. This gives $\mathbf{v} - \mathbf{w} = \mathbf{Q}^n(\mathbf{v} - \mathbf{w})$ for all $n \geq 1$, where $\mathbf{Q}^n = (1/n)\sum_{k=1}^n \mathbf{P}^k$. It is well-known from Markov chain theory that the (i, j)th element of $\mathbf{Q}^n$ converges to $\pi_j(R)$ as $n \to \infty$ for all $i, j \in I$. This shows that $v_i - w_i = \sum_{j\in I}(v_j - w_j)\pi_j(R)$ for all $i \in I$, proving that the relative-value function is uniquely determined up to an additive constant.

Why the name relative-value function? To answer this question, we first state the following result.

Theorem 2. *For a fixed stationary policy R, define $V_n(i, R)$ as the total expected cost incurred over the first n decision epochs when the starting state is i. Then, under the assumption that the Markov chain $\{X_n\}$ associated with policy R is aperiodic, there exists a finite function $v_i(R)$ such that*

$$\lim_{n\to\infty} V_n(i, R) - ng(R) = v_i(R) \quad \textit{for } i \in I.$$

Moreover, $g(R)$ and the $v_i(R)$ satisfy the value-determination equations.

Proof. A sketch of the proof is as follows. Denoting by $p_{ij}^{(k)}(R)$ the k-step transition probabilities of the Markov chain $\{X_n\}$ associated with policy R, we have $V_n(i, R) = \sum_{k=0}^{n-1}\sum_{j\in I} p_{ij}^{(k)}(R)c_j(R_j)$, where $p_{ij}^{(0)}$ is 1 for $j = i$ and is 0 otherwise. Together with the representation $g(R) = \sum_{j\in I} c_j(R_j)\pi_j(R)$, this leads after an interchange of the order of summation to

$$V_n(i, R) - ng(R) = \sum_{j\in I} c_j(R_j) \sum_{k=0}^{n-1} \left[p_{ij}^{(k)}(R) - \pi_j(R)\right].$$

We now invoke the assumption that the Markov chain $\{X_n\}$ associated with policy R is aperiodic. Then, the k-step transition probability $p_{ij}^{(k)}(R)$ converges exponentially fast to $\pi_j(R)$ as $k \to \infty$ for all $i, j \in I$, see e.g. Theorem

3.5.12 in Tijms (2003). That is, there are constants $\alpha > 0$ and $0 < \beta < 1$ such that $|p_{ij}^{(k)}(R) - \pi_j(R)| \leq \alpha\beta^k$ for all $k \geq 1$ and $i, j \in I$. This gives that the series $\sum_{k=0}^{\infty} \left[p_{ij}^{(k)}(R) - \pi_j(R)\right]$ is absolutely convergent, implying that $\lim_{n\to\infty} \sum_{k=0}^{n-1} \left[p_{ij}^{(k)}(R) - \pi_j(R)\right]$ exists and is finite for all $i \in I$, proving that $V_n(i, R) - ng(R)$ has a finite limit $v_i(R)$ as $n \to \infty$. To verify that $v_i(R)$, $i \in I$ is a relative-value function, use the recursive equation

$$V_n(i, R) = c_i(R_i) + \sum_{j\in I} p_{ij}(R_i)V_{n-1}(j, R),$$

subtract $ng(R)$ from its both sides and let $n \to \infty$ to obtain that the $v_i(R)$ satisfy the value-determination equations for policy R.

We can now explain the term relative-value function. Theorem 2 implies that

$$\lim_{n\to\infty} V_n(i, R) - V_n(j, R) = v_i(R) - v_j(R) \quad \text{for all } i, j \in I.$$

In other words, using the fact that the relative-value function is uniquely determined up to an additive constant, we have for any relative-value function v_i, $i \in I$ for policy R that $v_i - v_j$ represents the difference in the total expected costs over the infinite planning period $t = 1, 2, \ldots$ when the starting state is i rather than j provided that the Markov chain $\{X_n\}$ associated with policy R is aperiodic.

3. The policy-improvement step

In this section we come to the most important solution tool in Markovian control. This tool is the policy-improvement step and provides a flexible method to improve a given stationary policy R in order to obtain a stationary policy with a lower average cost per unit time. We first give a heuristic motivation of the policy-improvement step and next prove that it indeed leads to a better policy. The heuristic idea to improve a given stationary policy R is as follows. Suppose that you ask yourselves the question "how does change the total expected cost over the first n decision epochs when deviating from policy R by taking some other decision a in state i at the first decision epoch and next using policy R over the remaining decision epochs?" The change in the total expected costs over the first n decision epochs is given by

$$c_i(a) + \sum_{j\in I} p_{ij}(a)V_{n-1}(j, R) - V_n(i, R).$$

For the moment, assume that the Markov chain $\{X_n\}$ associated with policy R is aperiodic. Then, by Theorem 2 in the previous section, we have that $V_n(i, R) \approx ng(R) + v_i(R)$ for n large enough. Inserting this into the above expression, we get that the change in the expected costs is approximately given by $c_i(a) + \sum_{j\in I} p_{ij}(a)v_j(R) - g(R) - v_i(R)$. This suggests to look for an action a in state i so that the so-called *policy-improvement inequality*

$$c_i(a) - g(R) + \sum_{j\in I} p_{ij}(a)v_j(R) \leq v_i(R)$$

is satisfied. This heuristic discussion motivates our main theorem.

Theorem 3 (improvement theorem). *Let g and v_i, $i \in I$, be given numbers. Suppose that the stationary policy $\overline{R}$ has the property*

$$c_i(\overline{R}_i) - g + \sum_{j\in I} p_{ij}(\overline{R}_i)v_j \leq v_i \quad \textit{for all } i \in I.$$

Then the long-run average cost of policy $\overline{R}$ satisfies

$$g(\overline{R}) \leq g$$

with strict inequality if the strict inequality sign holds in the policy-improvement inequality for some state i that is recurrent under policy $\overline{R}$.

Proof. Since the Markov chain $\{X_n\}$ associated with policy $\overline{R}$ is unichain, this Markov chain has a unique equilibrium distribution $\{\pi_i(\overline{R}), i \in I\}$. The equilibrium probability $\pi_i(\overline{R})$ is positive only if state i is recurrent under policy $\overline{R}$. Multiplying both sides of the policy-improvement inequality by $\pi_i(\overline{R})$, summing over i and noting that $\sum_{i\in I}\pi_i(\overline{R}) = 1$, we get

$$\sum_{i\in I}\pi_i(\overline{R})c_i(\overline{R}_i) - g + \sum_{i\in I}\pi_i(\overline{R})\sum_{j\in I}p_{ij}(\overline{R}_i)v_j \leq \sum_{i\in I}\pi_i(\overline{R})v_i,$$

where the strict inequality sign holds if there is strict inequality in the policy-improvement inequality for some state i with $\pi_i(\overline{R}) > 0$. Interchanging the order of summation in the double sum and using the equilibrium equations $\pi_j(\overline{R}) = \sum_{i\in I}\pi_i(\overline{R})p_{ij}(\overline{R}_i)$ together with $g(\overline{R}) = \sum_{i\in I}\pi_i(\overline{R})c_i(\overline{R}_i)$, we find

$$g(\overline{R}) - g + \sum_{j\in I}\pi_j(\overline{R})v_j \leq \sum_{i\in I}\pi_i(\overline{R})v_i,$$

where the strict inequality sign holds if there is strict inequality in the policy-improvement inequality for some state i which is recurrent for policy $\overline{R}$. This completes the proof.

Remark 1. An examination of the proof shows that Theorem 3 remains valid when all inequality signs are reversed. As a consequence, a stationary policy R with average cost $g(R)$ and relative values $v_i(R)$, $i \in I$ is average cost optimal if

$$c_i(a) - g(R) + \sum_{j \in I} p_{ij}(a) v_j(R) \leq v_i(R) \quad \text{for all } i \in I \text{ and } a \in A(i).$$

Then $g(R)$ and $v_i(R)$, $i \in I$ satisfy the so-called *average cost optimality equation*

$$v_i = \min_{a \in A(i)} \left\{ c_i(a) - g + \sum_{j \in I} p_{ij}(a) v_j \right\} \quad \text{for } i \in I.$$

4. Policy-iteration algorithm

After the preparatory analysis in the previous section, we can formulate the policy-iteration algorithm:

Step 0 (initialization). Choose a stationary policy R.

Step 1 (value-determination step). For the current rule R, compute the unique solution $\{g(R), v_i(R)\}$ to the following system of linear equations:

$$v_i = c_i(R_i) - g + \sum_{j \in I} p_{ij}(R_i) v_j \quad \text{for } i \in I,$$

$$v_s = 0,$$

where s is an arbitrarily chosen state.

Step 2 (policy-improvement step). For each state $i \in I$, determine an action a_i yielding the minimum in

$$\min_{a \in A(i)} \left\{ c_i(a) - g(R) + \sum_{j \in I} p_{ij}(a) v_j(R) \right\}.$$

The new stationary policy $\overline{R}$ is obtained by choosing $\overline{R}_i = a_i$ for all $i \in I$ with the convention that $\overline{R}_i$ is chosen equal to the old action R_i when this action minimizes the policy-improvement quantity.

Step 3 (convergence test). If the new policy $\overline{R} = R$, then the algorithm is stopped with policy R. Otherwise, go to step 1 with R replaced by $\overline{R}$.

The policy-iteration algorithm can be shown to converge after a finite number of iterations to an average cost optimal policy. The policy-iteration algorithm is empirically found to be a remarkably robust algorithm that converges very fast in specific problems. The number of iterations is *practically independent* of the number of states and varies typically between 3 and 15 (say).

Remark 2. (*one-step policy improvement heuristic*) The policy-iteration algorithm has the remarkable feature that it achieves the largest improvements in costs in the first few iterations.These findings underlie a heuristic approach for Markov decision problems with a *multi-dimensional* state space. In such decision problems it is usually not feasible to solve the value-determination equations. However, a policy-improvement step offers in general no computational difficulties. This suggests a heuristic approach that determines first a good estimate for the relative values and next applies only *one* policy-improvement step. By the nature of the policy-iteration algorithm one might expect to obtain a good decision rule by the heuristic approach. How to compute the relative values to be used in the policy-improvement step typically depends on the specific application.

5. The Odoni bounds for value-iteration algorithm

The value-iteration algorithm computes recursively a sequence of value functions approximating the minimum average cost per time unit. The idea of the policy-improvement step enables us to extract from the value functions lower and upper bounds both on the average cost of the generated policies and. These bounds are the so-called Odoni bounds The bounds converge to the minimum average cost under an aperiodicity condition. The value-iteration algorithm endowed with these lower and upper bounds is in general the best computational method for solving large-scale Markov decision problems. Using Theorem 3 in section 2, we will give a simple derivation of these bounds. Before doing this, let us first formulate the value-iteration algorithm. The value-iteration algorithm computes recursively for $n = 1, 2, ...$ the value function $V_n(i)$ from

$$V_n(i) = \min_{a \in A(i)} \Big\{ c_i(a) + \sum_{j \in I} p_{ij}(a) V_{n-1}(j) \Big\} \quad \text{for } i \in I,$$

starting with an arbitrarily chosen function $V_0(i)$, $i \in I$. The quantity $V_n(i)$ can be interpreted as the minimum total expected costs with n periods left to the time horizon when the current state is i and a terminal cost of $V_0(j)$ is incurred when the system ends up at state j Intuitively, one might expect that the one-step difference $V_n(i) - V_{n-1}(i)$ will come very close to the minimum average cost per time unit and that the stationary policy whose actions minimize the right side of the equation for $V_n(i)$ for all i will be very close in cost to the minimum average cost when n is large enough. The recursion equation for $V_n(i)$ suggests to investigate the operator T that adds to each function $\mathbf{v} = (v_i,\, i \in I)$ a function $T\mathbf{v}$ whose ith component $(T\mathbf{v})_i$ is defined by

$$(T\mathbf{v})_i = \min_{a \in A(i)} \left\{ c_i(a) + \sum_{j \in I} p_{ij}(a) v_j \right\} \quad \text{for } i \in I.$$

Note that $(T\mathbf{v})_i = V_n(i)$ if $v_i = V_{n-1}(i)$, $i \in I$. The following theorem plays a key role in the value-iteration algorithm.

Theorem 4. Let $\mathbf{v} = (v_i, i \in I)$ *be a given function. Define the stationary policy* $R(\mathbf{v})$ *as a policy which adds to each state* $i \in I$ *an action* $a = R_i(\mathbf{v})$ *that minimizes the right-hand side of the equation for* $(T\mathbf{v})_i$. *Then,*

$$\min_{i \in I} \{(Tv)_i - v_i\} \leq g^* \leq g(R(\mathbf{v})) \leq \max_{i \in I} \{(Tv)_i - v_i\}$$

where g^* *is the minimum long-run average cost per unit time.*

Proof. To verify the bounds on g^*, take any stationary policy R. By the definition of $(T\mathbf{v})_i$, we have for any state $i \in I$ that

$$(T\mathbf{v})_i \leq c_i(a) + \sum_{j \in I} p_{ij}(a) v_j \quad \text{for all } a \in A(i),$$

where the equality sign holds for $a = R_i(\mathbf{v})$. Choosing $a = R_i$ gives

$$(Tv)_i \leq c_i(R_i) + \sum_{i \in I} p_{ij}(R_i) v_j \quad \text{for } i \in I.$$

Let $m = \min_{i \in I}\{(Tv)_i - v_i\}$. Noting that $m \leq (Tv)_i - v_i$ for all i and using the above inequality, we get $m + v_i \leq c_i(R_i) + \sum_{j \in I} p_{ij}(R_i) v_j$ for all $i \in I$, and so

$$c_i(R_i) - m + \sum_{j \in I} p_{ij}(R_i) v_j \geq v_i \quad \text{for } i \in I.$$

An application of Theorem 3 now gives that $g(R) \geq m$. This inequality holds for each policy R and so $g^* = \min_R g(R) \geq m$. The derivation of the upper bound for $g(R(\mathbf{v}))$ is very similar. By the definition of policy $R(\mathbf{v})$,

$$(T\mathbf{v})_i = c_i(R_i(\mathbf{v})) + \sum_{j \in I} p_{ij}(R_i(\mathbf{v})) v_j \quad \text{for } i \in I.$$

Let $M = \max_{i \in I} \{(T\mathbf{v})_i - v_i\}$. Since $M \geq (T\mathbf{v})_i - v_i$ for all $i \in I$, we get

$$c_i(R_i(\mathbf{v})) - M + \sum_{j \in I} p_{ij}(R_i(\mathbf{v})) v_j \leq v_i \quad \text{for } i \in I.$$

Hence, by Theorem 3, $g(R(\mathbf{v})) \leq M$ for all $i \in I$. This completes the proof.

How do the bounds in Theorem 4 work out for the value-iteration algorithm? Let $R^{(n)}$ be any stationary policy such that the action $a = R_i^{(n)}$ minimizes the right-hand side of the recursive equation for $V_n(i)$ for all $i \in I$. Define the Odoni bounds

$$m_n = \min_{i \in I} \{V_n(i) - V_{n-1}(i)\} \ \text{ and } M_n = \max_{i \in I} \{V_n(i) - V_{n-1}(i)\}.$$

Then, $m_n \leq g^* \leq g(R^{(n)}) \leq M_n$. It is no restriction to assume that $c_i(a) > 0$ for all i, a; otherwise, add a same sufficiently large constant c to each $c_i(a)$. We then have $m_n > 0$ so that the bounds imply that

$$0 \leq \frac{g(R^{(n)}) - g^*}{g^*} \leq \frac{M_n - m_n}{m_n}.$$

The value-iteration algorithm is typically stopped as soon as $0 \leq M_n - m_n \leq \varepsilon m_n$, where $\varepsilon > 0$ is a small pre-specified number, e.g. $\varepsilon = 10^{-3}$. The remaining question is whether the lower and upper bounds m_n and M_n converge to the same limit so that the algorithm will be stopped after finitely many iterations. The answer to this question is only in the affirmative if a certain *aperiodicity* condition is satisfied for the underlying Markov chains. A sufficient condition is that the Markov chain $\{X_n\}$ is aperiodic for each average cost optimal stationary policy. Then the monotone sequences $\{m_n\}$ and $\{M_n\}$ have the same limit g^* and the convergence to this limit can be shown to be exponentially fast

The aperiodicity requirement is no problem for the value-iteration method. The periodicity issue can be circumvented by a classical *perturbation* of the one-step transition probabilities of a Markov chain. If the one-step transition probabilities p_{ij} of a Markov chain are perturbed as

$$\overline{p}_{ij} = \tau p_{ij} \text{ for } j \neq i \text{ and } \overline{p}_{ii} = \tau p_{ii} + 1 - \tau$$

for some constant τ with $0 < \tau < 1$, the perturbed Markov chain with one-step transition probabilities $\overline{p}_{ij}$ is aperiodic and has the same equilibrium probabilities as the original Markov chain. Thus a Markov decision model involving periodicity may be perturbed as follows. Choose some constant τ with $0 < \tau < 1$ and let the state space, the action sets, and the one-step costs unchanged. For any $i \in I$ and $a \in A(i)$, the one-step transition probabilities of the perturbed Markov decision model are defined by

$$p_{ii}(a) = \tau p_{ij}(a) + 1 - \tau \ \text{ and } \ p_{ij}(a) = \tau p_{ij}(a) \ \text{ for } j \neq i.$$

For each stationary policy, the associated Markov chain in the perturbed model is aperiodic, while the average cost per unit time in the perturbed model is the same as that in the original model. In specific problems involving periodicity the 'optimal' value of τ is usually not clear beforehand; empirical investigations indicate that $\tau = \frac{1}{2}$ is usually a satisfactory choice. It is noted that this transformation technique can be generalized to transform any semi-Markov decision model into an equivalent discrete-time Markov decision model so that the value-iteration method can also be applied to the semi-Markov decision model, see Tijms (2003) for details.

Finally, it is pointed out that the unichain assumption from section 1 can be weakened. In practical applications of the average cost Markov decision model, there may be non-optimal policies for which the associated Markov chains have multiple recurrent classes, but the unichain property typically holds for the average cost optimal policies. For the value-iteration method, it suffices to require that the Markov chains associated with the average cost optimal policies are unichain (and aperiodic), see Tijms (2003) for details.

5. Numerical illustration of the algorithms

In this section the policy-iteration algorithm and the value-iteration algorithm are illustrated with the following example.

Maintenance problem: At the beginning of each day a piece of equipment is inspected to reveal its actual working condition. The equipment will be found in one of the working conditions $i = 1, ..., N$, where the working condition i is better than the working condition $i + 1$. The equipment deteriorates in time. If the present working condition is i and no repair is done, then at the beginning of the next day the equipment has working condition j with probability q_{ij}. It is assumed that $q_{ij} = 0$ for $j < i$ and $\sum_{j \geq i} q_{ij} = 1$. The working condition $i = N$ represents a malfunction that requires an enforced repair taking two days. For the intermediate states i with $1 < i < N$ there is a choice between preventively repairing the equipment and letting the equipment operate for the present day. A preventive repair takes only one day. A repaired system has the working condition $i = 1$. The cost of an enforced repair upon failure is C_f and the cost of a preemptive repair in working condition i is C_{pi}. The goal is to find a maintenance rule that minimizes the long-run average cost per day.

Since an enforced repair takes two days and the state of the system has to be defined at the beginning of each day, we need an auxiliary state for the situation in which an enforced repair is in progress already for one day. Thus the set of possible states of the system is chosen as $I = \{1, 2, ..., N, N + 1\}$. State i with $1 \leq i \leq N$ corresponds to the situation in which an inspection reveals working condition i, while state $N + 1$ corresponds to the situation in which an enforced repair is in progress already for one day. Define the actions

$$a = \begin{cases} 0 & \text{if no repair is done} \\ 1 & \text{if a preventive repair is done} \\ 2 & \text{if an enforced repair is done.} \end{cases}$$

The set of possible actions in state i is chosen as

$$A(1) = \{0\}, \quad A(i) = \{0, 1\} \text{ for } 1 < i < N, \quad A(N) = A(N+1) = \{2\}.$$

The one-step transition probabilities $p_{ij}(a)$ are given by

$$p_{ij}(0) = q_{ij} \quad \text{for } 1 \leq i < N,\ p_{i1}(1) = 1 \quad \text{for } 1 < i < N,$$
$$p_{N,N+1}(2) = p_{N+1,1}(2) = 1,$$

and the other $p_{ij}(a) = 0$. The one-step costs $c_i(a)$ are given by

$$c_i(0) = 0, \quad c_i(1) = C_{pi}, \quad c_N(2) = C_f \quad \text{and} \quad c_{N+1}(2) = 0.$$

The following numerical data are taken

$$N = 5,\ C_f = 10,\ C_{p2} = C_{p3} = 7, \text{ and } C_{p4} = 5.$$

The deterioration probabilities q_{ij} are given by

$i\backslash j$	1	2	3	4	5
1	0.90	0.10	0	0	0
2	0	0.80	0.10	0.05	0.05
3	0	0	0.70	0.10	0.20
4	0	0	0	0.50	0.50

Policy iteration

The policy-iteration algorithm is initialized with the policy $R^{(1)} = (0, 0, 0, 0, 2, 2)$ which prescribes repair only in the states 5 and 6. In the calculations below the policy-improvement quantity is abbreviated as

$$T_i(a, R) = c_i(a) - g(R) + \sum_{j \in I} p_{ij}(a) v_j(R)$$

when the current policy is R. Note that

$$T_i(a, R) = v_i(R) \quad \text{for } a = R_i.$$

Iteration 1
Step 1 (value determination). The average cost and the relative values of policy $R^{(1)} = (0, 0, 0, 0, 2, 2)$ are computed by solving the linear equation.

$$\begin{aligned}
v_1 &= 0 - g + 0.9v_1 + 0.1v_2 \\
v_2 &= 0 - g + 0.8v_2 + 0.1v_3 + 0.05v_4 + 0.05v_5 \\
v_3 &= 0 - g + 0.7v_3 + 0.1v_4 + 0.2v_5 \\
v_4 &= 0 - 9 + 0.5v_4 + 0.5v_5 \\
v_5 &= 10 - g + v_6 \\
v_6 &= 0 - g + v_1 \\
v_6 &= 0,
\end{aligned}$$

where state $s = 6$ is chosen for the normalizing equation $v_s = 0$. The solution of these linear equations is given by

$$\begin{aligned}
g(R^{(1)}) &= 0.5128,\ v_1(R^{(1)}) = 0.5128,\ v_2(R^{(1)}) = 5.6410,\ v_3(R^{(1)}) = 7.4359, \\
v_4(R^{(1)}) &= 8.4615,\ v_5(R^{(1)}) = 9.4872,\ v_6(R^{(1)}) = 0.
\end{aligned}$$

Step 2 (policy improvement). The test quantity $T_i(a, R)$ has the values

$$\begin{aligned}
T_2(0, R^{(1)}) &= 5.6410,\ T_2(1, R^{(1)}) = 7.0000,\ T_3(0, R^{(1)}) = 7.4359, \\
T_3(1, R^{(1)}) &= 7.0000\ T_4(0, R^{(1)}) = 9.4872,\ T_4(1, R^{(1)}) = 5.0000.
\end{aligned}$$

This yields the new policy $R^{(2)} = (0, 0, 1, 1, 2, 2)$ by choosing for each state i the action a that minimizes $T_i(a, R^{(1)})$.

Step 3 (convergence test). The new policy $R^{(2)}$ is different from the previous policy $R^{(1)}$ and hence a next iteration is performed.

Iteration 2

Step 1 (value determination). The average cost and the relative values of policy $R^{(2)} = (0, 0, 1, 1, 2, 2)$ are computed by solving the linear equations

$$\begin{aligned}
v_1 &= 0 - g + 0.9v_1 + 0.1v_2 \\
v_2 &= 0 - g + 0.8v_2 + 0.1v_3 + 0.05v_4 + 0.05v_5 \\
v_3 &= 7 - g + v_1 \\
v_4 &= 5 - g + v_1 \\
v_5 &= 10 - g + v_6 \\
v_6 &= 0 - g + v_1 \\
v_6 &= 0.
\end{aligned}$$

The solution of these linear equations is given by

$$\begin{aligned} g(R^{(2)}) &= 0.4462,\ v_1(R^{(2)}) = 0.4462,\ v_2(R^{(2)}) = 4.9077,\ v_3(R^{(2)}) = 7.000, \\ v_4(R^{(2)}) &= 5.0000,\ v_5(R^{(2)}) = 9.5538,\ v_6(R^{(2)}) = 0. \end{aligned}$$

Step 2 (policy improvement). The test quantity $T_i(a, R^{(2)})$ has the values

$$\begin{aligned} T_2(0, R^{(2)}) &= 4.9077,\ T_2(1, R^{(2)}) = 7.0000\ T_3(0, R^{(2)}) = 6.8646, \\ T_3(1, R^{(2)}) &= 7.0000,\ T_4(0, R^{(2)}) = 6.8307,\ T_4(1, R^{(2)}) = 5.0000. \end{aligned}$$

This yields the new policy $R^{(3)} = (0, 0, 0, 1, 2, 2)$.

Step 3 (convergence test). The new policy $R^{(3)}$ is different from the previous policy $R^{(2)}$ and hence a next iteration is performed.

Iteration 3
Step 1 (value determination). The average cost and the relative values of policy $R^{(3)} = (0, 0, 0, 1, 2, 2)$ are computed by solving the linear equations

$$\begin{aligned} v_1 &= 0 - g + 0.9v_1 + 0.1v_2 \\ v_2 &= 0 - g + 0.8v_2 + 0.1v_3 + 0.05v_4 + 0.05v_5 \\ v_3 &= 0 - g + 0.7v_3 + 0.1v_4 + 0.2v_5 \\ v_4 &= 5 - g + v_1 \\ v_5 &= 10 - g + v_6 \\ v_6 &= 0 - g + v_1 \\ v_6 &= 0. \end{aligned}$$

The solution of these linear equations is given by

$$\begin{aligned} g(R^{3)}) &= 0.4338,\ v_1(R^{(3)}) = 0.4338,\ v_2(R^{(3)}) = 4.7717,\ v_3(R^{(3)}) = 6.5982, \\ v_4(R^{(3)}) &= 5.0000,\ v_5(R^{(3)}) = 9.5662,\ v_6(R^{(3)}) = 0. \end{aligned}$$

Step 2 (policy improvement). The test quantity $T_i(a, R^{(3)})$ has the values

$$\begin{aligned} T_2(0, R^{(3)}) &= 4.7717,\ T_2(1, R^{(3)}) = 7,\ T_3(0, R^{(3)}) = 6.5987, \\ T_3(1, R^{(3)}) &= 7.0000\ T_4(0, R^{(3)}) = 6.8493,\ T_4^{(1)}(1, R^{(3)}) = 5.0000. \end{aligned}$$

This yields the new policy $R^{(4)} = (0, 0, 0, 1, 2, 2)$.

Step 3 (convergence test). The new policy $R^{(4)}$ is identical to the previous policy $R^{(3)}$ and is thus average cost optimal. The minimal average cost is 0.4338 per day.

Value iteration

For the maintenance problem the recursion equation becomes

$$
\begin{aligned}
&V_n(1) = 0 + \sum_{j=1}^{N} q_{1j} V_{n-1}(j),\\
&V_n(i) = \min\left\{0 + \sum_{j=i}^{N} q_{ij} V_{n-1}(j),\ C_{pi} + V_{n-1}(1)\right\} \quad \text{for } 1 < i < N,\\
&V_n(N) = C_f + V_{n-1}(N+1),\ V_n(N+1) = 0 + V_{n-1}(1).
\end{aligned}
$$

For each stationary policy the associated Markov chain $\{X_n\}$ is aperiodic. Taking $V_0(i) = 0$ for all i and the accuracy number $\varepsilon = 10^{-3}$, the algorithm is stopped after $n = 28$ iterations with the stationary policy $R(n) = (0, 0, 0, 1, 2, 2)$ together with the lower and upper lower bounds $m_n = 0.4336$ and $M_n = 0.4340$. The average cost of policy $R(n)$ is estimated by $\frac{1}{2}(m_n + M_n) = 0.4338$ and this cost cannot deviate more than 0.1% from the theoretically minimal average cost. In fact policy $R(n)$ is optimal as we know from previous results obtained by policy iteration.

References

1. Tijms, H. C. (2003). *A First Course in Stochastic Models*, Wiley, New York.