Chapter 7

Infinite Horizon Models: Long-Run Average Reward

This material will be published by Cambridge University Press as "Markov Decision Processes and Reinforcement Learning" by Martin L. Puterman and Timothy C. Y. Chan. This pre-publication version is free to view and download for personal use only. Not for re-distribution, re-sale, or use in derivative works. ©Martin L. Puterman and Timothy C. Y. Chan, 2025.

> *The field of average cost optimization is a strange one. Counterexamples exist to almost all natural conjectures, yet these conjectures are the basis of a proper intuition and are valid if reformulated right or if natural conditions are imposed*[1].
>
> Peter Whittle, Mathematician and statistician, 1927-2021.

This chapter focuses on infinite horizon Markov decision processes with the long-run average reward criterion, referred to as *average reward models*. The average reward criterion applies to non-terminating systems, such as controlled queues, in which decisions are made frequently and in perpetuity. In such models neither discounted or expected total reward models are appropriate but for different reasons: discounting trades off recent decisions with future decisions while the expected total reward does not discriminate between policies.

Average reward models pose numerous analytical challenges, as noted by Whittle in the quote above. This is because the existence and structure of the average reward depends on the *limiting* properties[2] of underlying Markov chains. In contrast to discounted models in which Markov chain properties do not impact results, they are fundamental here. Challenges faced include:

[1] Whittle (1983), p. 118.

[2] Limiting properties refer to the behavior of large powers of transition probability matrices.

1. For stationary policies with periodic transition probabilities, limits need not exist due to oscillation of the powers of transition probability matrices.

2. In finite state models, limits needed to define the average reward for history-dependent and Markovian policies need not exist.

3. In countable state models, the limit of transition probabilities may exist but it need not be a transition probability matrix.

4. The form of the Bellman equation depends on whether the average reward is constant or state-dependent. This is determined by the class structure of underlying Markov chains.

In light of these challenges, this chapter will focus primarily on finite state and actions models that are aperiodic and have constant (state-independent) average reward, a point that will be emphasized throughout. The reader is encouraged to review Appendix B for background on Markov chains relevant for Markov decision process.

7.1 Preliminaries

This chapter assumes:

Stationary rewards: The (expected) reward function does not vary from epoch to epoch and does not depend on the subsequent state[3]. It will be written as $r(s, a)$, independent of n.

Bounded rewards: There is a finite W such that $|r(s, a)| \leq W$ for all $a \in A_s, s \in S$.

Stationary transition probabilities: The transition probabilities will be written $p(j|s, a)$, independent of n.

This chapter examines how the form of the average reward is influenced by the structural properties of Markov chains induced by stationary policies. Key concepts of Markov chains are explored and illustrated to provide the necessary foundation. Several constructs unique to Markov decision processes are introduced, including *partial Laurent expansions*, *bias vectors*, and *relative value vectors*, which are essential for deriving and determining properties of the Bellman equation. These constructs underlie the three primary algorithms discussed: value iteration, policy iteration, and linear programming. The analysis and application of Gauss-Seidel iteration, modified policy iteration, and action elimination are left to the reader.

[3]As in previous chapters, when the reward depends on the subsequent state, that is it is given by $r(s, a, j)$, it is replaced by the expected reward $r(s, a) = \sum_{j \in S} r(s, a, j)p(j|s, a)$.

7.2 The long-run average reward or gain

At first glance, one might consider defining the *long-run average reward* or *gain* of a policy $\pi \in \Pi^{\text{HR}}$ by

$$g^{\pi}(s) := \lim_{N\to\infty} \frac{1}{N} E^{\pi}\left[\sum_{n=1}^{N} r(X_n, Y_n)\middle| X_1 = s\right]. \tag{7.1}$$

It will be shown below that the limit in this definition exists for stationary policies, however, it need not exist for non-stationary policies. This necessitates a more robust definition of the long-run average reward of a policy.

7.2.1 The gain of a stationary policy

For a stationary policy $\pi = d^{\infty}$ where $d \in D^{\text{MR}}$, (7.1) becomes

$$g^{d^{\infty}}(s) = \lim_{N\to\infty} \frac{1}{N} E^{d^{\infty}}\left[\sum_{n=1}^{N} r(X_n, Y_n)\middle| X_1 = s\right]. \tag{7.2}$$

Recalling the notation

$$r_d(s) = \sum_{a\in A_s} w_d(a|s) r(s,a) \quad \text{and} \quad p_d(j|s) = \sum_{a\in A_s} w_d(a|s) p(j|s,a), \tag{7.3}$$

equation (7.2) can be expressed in vector notation as

$$\mathbf{g}^{d^{\infty}} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathbf{P}_d^{n-1} \mathbf{r}_d, \tag{7.4}$$

where $\mathbf{r}_d$ denotes a vector with components $r_d(s)$, $\mathbf{P}_d$ denotes the matrix with components $P_d(j|s)$, and $\mathbf{P}_d^0 = \mathbf{I}$.

Applying the first result of Theorem B.1 in Appendix B provides the following representation for the gain.

Theorem 7.1. Let S be finite and let $\pi = d^{\infty}$ where $d \in D^{\text{MR}}$. Then the limit in (7.1) exists and satisfies

$$\mathbf{g}^{d^{\infty}} = \mathbf{P}_d^* \mathbf{r}_d \tag{7.5}$$

where

$$\mathbf{P}_d^* := \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathbf{P}_d^{n-1}. \tag{7.6}$$

Note that when $\mathbf{P}_d^n$ converges to $\mathbf{P}_d^*$, which occurs for example when the underlying Markov chain is regular[4], the gain may be also regarded as the *steady state reward*. A fundamental result in Markov chain theory also establishes that for regular chains, $\mathbf{P}_d^n$ converges to $\mathbf{P}_d^*$ at an *exponential* rate.

7.2.2 The gain of a non-stationary policy need not exist

The following example[5] shows that the limit in (7.1) need not exist for history-dependent and non-stationary Markovian policies. The analysis is quite subtle and can be skipped, but the conclusion is important and justifies the need for a more robust definition of the gain of a policy.

Let $S = \{s_1, s_2\}$, $A_{s_i} = \{a_{i,1}, a_{i,2}\}$ for $i = 1, 2$, $p(s_1|s_1, a_{1,1}) = 1$, $p(s_2|s_1, a_{1,2}) = 1$, $p(s_1|s_2, a_{2,1}) = 1$ and $p(s_2|s_2, a_{2,2}) = 1$ and $r(s_1, a_{1,1}) = r(s_1, a_{1,2}) = 1$ and $r(s_2, a_{2,1}) = r(s_2, a_{2,2}) = 0$. (See Figure 7.1).

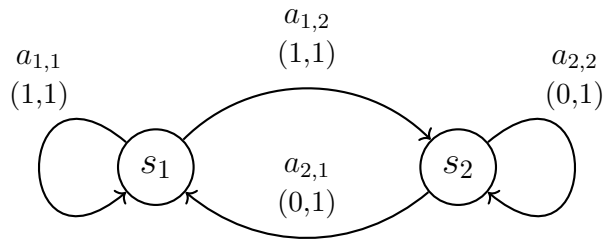


Figure 7.1: Symbolic representation of model in Section 7.2.2. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

A problematic non-stationary policy

Consider the deterministic history-dependent policy that when starting in state s_1 chooses the following sequence of actions:

$$a_{1,2}, a_{2,1}, a_{1,2}, a_{2,1}, a_{1,1}, a_{1,2}, a_{2,2}, a_{2,1}, a_{1,1}, a_{1,1}, a_{1,1}, a_{1,2}, a_{2,2}, a_{2,2}, a_{2,2}, a_{2,1}, \ldots$$

and generates the following sequence of rewards where parentheses correspond to blocks of length $k = 2, 2, 4, 8, 16, \ldots$ consisting of $k/2$ ones followed by $k/2$ zeroes.

$$(1, 0), (1, 0), (1, 1, 0, 0), (1, 1, 1, 1, 0, 0, 0, 0), \ldots.$$

[4]See Appendix B for definitions of relevant Markov chain concepts. Note this result holds for other aperiodic Markov chains as well.

[5]This elegant example and its analysis was adopted from Section A.4 in Sennott [1999]. Example 8.1.1 in Puterman [1994] provides a model with similar properties.

The corresponding sequence of averages w_n is given by

$$\left(1, \frac{1}{2}\right), \left(\frac{2}{3}, \frac{2}{4}\right), \left(\frac{3}{5}, \frac{4}{6}, \frac{4}{7}, \frac{4}{8}\right), \left(\frac{5}{9}, \frac{6}{10}, \frac{7}{11}, \frac{8}{12}, \frac{8}{13}, \frac{8}{14}, \frac{8}{15}, \frac{8}{16}\right), \ldots$$

Therefore

$$1/2 = \liminf_{n\to\infty} w_n < \limsup_{n\to\infty} w_n = 2/3$$

where the lim inf corresponds to the subsequence that chooses the last 0 in each block and the lim sup corresponds to the subsequence that chooses the last 1 in each block. Since a sequence converges when every subsequence has the same limit, this analysis shows that the limit in (7.1) **does not** exist for this history-dependent policy starting in state s_1.

When the system starts in state s_2, it chooses the actions

$$a_{2,2}, a_{2,1}, a_{1,2}, a_{2,1}, a_{1,1}, a_{1,2}, a_{2,2}, a_{2,1}, a_{1,1}, a_{1,1}, a_{1,1}, a_{1,2}, a_{2,2}, a_{2,2}, a_{2,2}, a_{2,1}, \ldots$$

that are identical to that in s_1 after the first decision epoch. For states not visited by this deterministic policy at a decision epoch, action choice is arbitrary and deterministic.

As a consequence of Lemma 5.9 for s_1 and s_2 there exist Markovian randomized policies that have the same state-action probabilities and generate the same sequence of rewards after the first decision epoch. Moreover, after defining $d_1(s_1) = a_{1,2}$ and $d_1(s_2) = a_{2,2}$ the Markovian policies are identical and deterministic with $d_2(s_2) = a_{2,1}$, $d_3(s_1) = a_{1,2}, \ldots$. Hence, in fact, there is a Markovian deterministic policy for which the limit in (7.1) does not exist.

Stationary policies

Now consider stationary policies. Simple calculations show:

action in s_1	action in s_2	gain in s_1	gain in s_2
$a_{1,1}$	$a_{2,2}$	1	0
$a_{1,1}$	$a_{2,1}$	1	1
$a_{1,2}$	$a_{2,2}$	0	0
$a_{1,2}$	$a_{2,1}$	1/2	1/2

Since the largest one-step reward is 1, the stationary policy, $(d^*)^\infty$ that uses $a_{1,1}$ in s_1 and $a_{2,1}$ in s_2 is optimal with respect to the average reward criterion.

Observe also that for the policy $d'(s_1) = a_{1,1}$ and $d'(s_2) = a_{2,2}$, the gain varies with the starting state. This is because this policy generates a Markov chain with two closed classes[6] .

[6]This issue will be discussed in the next section on model classification.

7.2.3 A more general definition of the gain

The above example illustrates the need for a more rigorous definition of the gain that as well applies to non-stationary policies. For any $\pi = (d_1, d_2, \ldots) \in \Pi^{\text{HR}}$, let

$$v_N^\pi(s) := E^\pi \left[\sum_{n=1}^{N} r(X_n, Y_n) \middle| X_1 = s \right]. \tag{7.7}$$

In vector notation,

$$\mathbf{v}_N^\pi = \sum_{n=1}^{N} \mathbf{P}_\pi^{n-1} \mathbf{r}_{d_n}, \tag{7.8}$$

where $\mathbf{P}_\pi^0 = \mathbf{I}$ and $\mathbf{P}_\pi^n = \mathbf{P}_{d_1} \mathbf{P}_{d_2} \cdots \mathbf{P}_{d_n}$.

Definition 7.1. For $\pi \in \Pi^{\text{HR}}$ define the *lim inf average reward* $g_-^\pi(s)$ and *lim sup average reward* $g_+^\pi(s)$ by:

$$g_-^\pi(s) := \liminf_{N\to\infty} \frac{1}{N} v_N^\pi(s) \quad \text{and} \quad g_+^\pi(s) := \limsup_{N\to\infty} \frac{1}{N} v_N^\pi(s) \tag{7.9}$$

for all $s \in S$.

Clearly $g_-^\pi(s) \le g_+^\pi(s)$. The example in the previous section shows that this inequality may be strict. When they are equal the limit in (7.1) exists, so that:

Definition 7.2. Whenever $g_-^\pi(s) = g_+^\pi(s)$ for all $s \in S$, the *average reward* of policy $\pi \in \Pi^{\text{HR}}$ is defined by:

$$g^\pi(s) := \lim_{N\to\infty} \frac{1}{N} v_N^\pi(s). \tag{7.10}$$

Applying Lemma 5.9, gives the following important result which simplifies subsequent analyses in this chapter in the same way as Theorem 5.1 did in Chapter 5. It says that given any history-dependent policy, there exists a Markov randomized policy with the same lim inf average reward and lim sup average reward.

Theorem 7.2. Let $\pi \in \Pi^{\text{HR}}$. Then for each $s \in S$, there exists a $\pi' \in \Pi^{\text{MR}}$ for which

$$g_-^{\pi'}(s) = g_-^{\pi}(s) \quad \text{and} \quad g_+^{\pi'}(s) = g_+^{\pi}(s). \tag{7.11}$$

Furthermore if $g_-^{\pi}(s) = g_+^{\pi}(s)$,

$$g^{\pi'}(s) = g^{\pi}(s). \tag{7.12}$$

7.2.4 Average optimality

It would be desirable to say that a policy π^* is optimal with respect to the average reward criterion whenever

$$g^{\pi^*}(s) \geq g^{\pi}(s) \tag{7.13}$$

for all $\pi \in \Pi^{\text{HR}}$ and $s \in S$. Unfortunately, as shown above, the limits implicit in defining the quantities in (7.13) need not exist, even when S is finite. Thus the expressions in (7.9) are needed to define optimality. (Recall the quote at the beginning of this chapter.)

Hence one must be content (at least until establishing the existence of optimal stationary policies) to adopt the following more rigorous notion of optimality.

Definition 7.3. A policy π^* is *average optimal* (*gain optimal*) if for all $s \in S$

$$g_-^{\pi^*}(s) \geq g_+^{\pi}(s) \tag{7.14}$$

for all $\pi \in \Pi^{\text{HR}}$.

What this definition means is that a policy is average optimal if its worst possible value (corresponding to its smallest limit point) is at least as large as the best possible value (corresponding to the largest limit point) of any other policy.

The example in Section 7.2.2 provides an illustration of this definition in action. That example showed that there exists a non-stationary Markovian policy π' for which

$$g_-^{\pi'}(s) = 1/2 \quad \text{and} \quad g_+^{\pi'}(s) = 2/3$$

for $s \in S$. Moreover from Theorem 7.1 for any stationary policy,

$$g^{d^\infty}(s) = g_+^{d^\infty}(s) = g_-^{d^\infty}(s).$$

Thus it follows for the stationary policy $(d^*)^\infty$ where $d^*(s_1) = a_{1,1}$ and $d^*(s_2) = a_{2,1}$ that:

$$g^{(d^*)^\infty}(s) = g_-^{(d^*)^\infty}(s) \geq g_+^{\pi'}(s)$$

for all $s \in S$. Noting that $g_+^{\pi}(s) \leq 1$ for all $\pi \in \Pi^{\text{HR}}$ and $s \in S$ and $g^{(d^*)^\infty}(s) = 1$ for all $s \in S$, confirms the optimality of d^* under this more general definition.

Optimal value functions

Defining optimal value functions is more subtle than in discounted and expected total reward models because as illustrated by the example in Section 7.2.2, the limit corresponding to the average reward of a policy need not exist. In light of this result, the *optimal value function* $g^*(s)$ is defined by[7]:

Definition 7.4. An *optimal value function* $g^*(s)$ satisfies

$$g^*(s) := \sup_{\pi \in \Pi^{\text{HR}}} g_-^\pi(s) = \sup_{\pi \in \Pi^{\text{HR}}} \liminf_{N \to \infty} \frac{1}{N} v_N^\pi(s) \tag{7.15}$$

This means that $g^*(s) \geq g_-^\pi(s)$ for all $\pi \in \Pi^{\text{HR}}$ so that if a policy attains this value, it has the best worst case behavior among all policies[8].

Later in this chapter, it will be shown that there exists a stationary deterministic policy that is optimal in certain average reward models so that

$$g^*(s) = \max_{d \in D^{\text{SD}}} g^{d^\infty}(s). \tag{7.16}$$

In this case, a policy π is average optimal whenever $g^\pi(s) = g^*(s)$ for all $s \in S$.

7.3 Chain structure

In contrast to discounted models, in average reward models transition properties of Markov chains generated by stationary policies impact the form of the average reward and the structure of the evaluation and Bellman equation. Material in Appendix B is fundamental here.

[7] Not this definition differs from that in Puterman [1994]

[8] In a minimization problem, define $g^*(s) := \inf_{\pi \in \Pi^{\text{HR}}} g_+^\pi(s)$.

Definition 7.5. A Markov decision process is said to be:

- *Regular* if the Markov chain corresponding to every Markovian deterministic decision rule is irreducible and aperiodic, in other words, consists of a single closed aperiodic class and no transient states.
- *Recurrent* or *ergodic* if the Markov chain corresponding to every Markovian deterministic decision rule consists of a single closed class and no transient states. Note that a regular model is a special case of a recurrent model in which every stationary deterministic policy generates an aperiodic Markov chain.
- *Unichain* if the Markov chain corresponding to every Markovian deterministic decision rule consists of a single closed class and the Markov chain corresponding to at least one decision rule contains a non-empty set of transient states[a].
- *Multi-chain* if the Markov chain corresponding to at least one Markovian deterministic decision rule contains two or more closed classes and possibly transient states[b].
- *Communicating* if for each pair of states s and j in S, there exists a Markovian deterministic decision rule d for which j is accessible from s, that is $p_d^n(j|s) > 0$ for some $n > 0$.
- *Weakly communicating* if the state space can be decomposed into two disjoint sets of states, one containing states that are transient under every deterministic decision rule and the other which forms a communicating Markov decision process[c].

[a] In Puterman [1994], a unichain model need not have any transient states in which case recurrent models and regular models are special cases. The definition here requires that at least one policy have transient states.

[b] This definition does not preclude the possibility of a model in which some decision rules generate Markov chains with a single closed class.

[c] Some authors refer to such a model as *weakly accessible*.

The first four categories depend on the chain structure of *every* Markovian deterministic decision rule. On the other hand, determining whether a model is communicating or weakly communicating requires checking whether every pair of states is accessible from each other under some policy. Note that regular and recurrent models are by definition communicating, a unichain model is necessarily weakly communicating and a multi-chain model may or may not be communicating or weakly communicating.

This classification scheme must be taken into consideration because it determines whether the value function is constant, as well as the appropriate form of the evaluation

equations and the Bellman equation. These relationships are summarized in Table 7.1.

Model class	Evaluation equations	Gain of a stationary policy	Gain of optimal policy
Regular or Recurrent	(7.43)	constant	constant
Unichain	(7.43)	constant	constant
Multi-chain	(7.35)	constant or non-constant	constant or non-constant
Communicating	(7.35)	constant or non-constant	constant
Weakly communicating	(7.35)	constant or non-constant	constant

Table 7.1: Relationship between chain structure and model features.

Communicating models

Note that in the definition of a communicating model, a different stationary deterministic decision rule may be required to reach states j and j' from s. The following result shows that this is not a limitation and moreover establishes the form of the optimal gain in a communicating model.

Proposition 7.1. In a communicating model:

1. There exists a randomized decision rule under which all pairs of states are accessible from each other.

2. The optimal average reward is constant.

Proof. The first part follows by noting that if a different decision rule is required to reach j and j' from state s, then randomizing between the action chosen by these two policies in state s generates a single randomized policy in which both j and j' are accessible from s.

To prove the second part, suppose there exists a stationary policy with closed[9] classes $C_1, C_2, \ldots, C_m$ with different average rewards on each. Assume further that C_1 has the greatest average reward g_1. From the first part of this proposition, there exists a randomized stationary policy under which there is a positive probability of reaching C_1 from each state in $C_2, \ldots, C_m$. Hence under this policy, all states in $C_2, \ldots, C_m$ are transient and have average reward g_1. □

Note that the presence of transient states requires only slight modification to the above proof to obtain the following corollary.

[9]Meaning that states in this class cannot be reached by states outside this class under the policy (see Appendix for formal definition).

Corollary 7.1. In a weakly communicating model, the optimal average reward is constant.

As noted in Table 7.1 communicating and weakly communicating models have constant gain but require the multi-chain evaluation equations, (7.35), because some policies may have two or more closed classes.

Chain structure examples

The following examples illustrate these properties.

Example 7.1. Consider the two-state model in Section 2.5. In it there are four deterministic decision rules. The Markov chain corresponding to decision rule $d(s_1) = a_{1,1}$ and $d(s_2) = a_{2,1}$ has a unichain transition matrix with state s_1 transient and state s_2 being a closed class (absorbing state). Also, all other deterministic decision rules correspond to regular Markov chains. Thus this model is unichain. Moreover it is easy to see that it is communicating as well.

A variant: Suppose a third action $a_{1,3}$ is added in s_1 with $p(s_1|s_1, a_{1,3}) = 1$, $p(s_2|s_1, a_{1,3}) = 0$ and arbitrary reward k (see Figure 7.2). Then as shown in Figure 7.3 the Markov chain corresponding to decision rule $d'(s_1) = a_{1,3}$ and $d'(s_2) = a_{2,1}$ has two closed classes. Hence with this additional action the model is multi-chain and communicating. This illustrates the interesting property that a model remains communicating when additional actions become available.

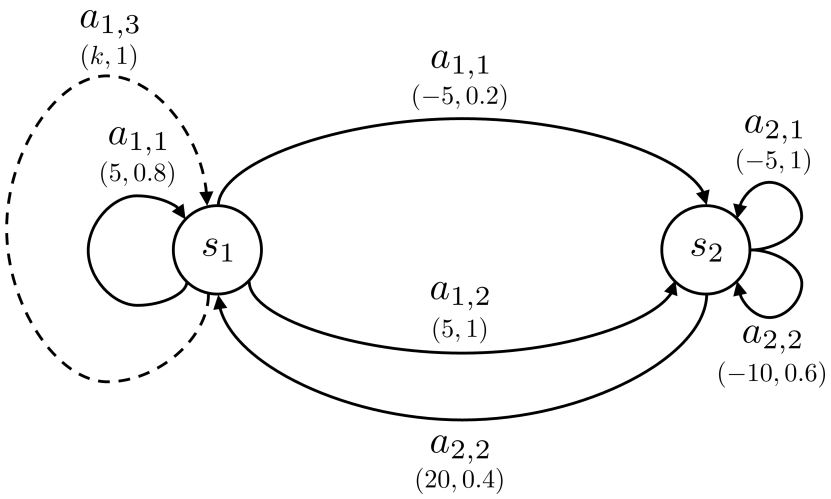
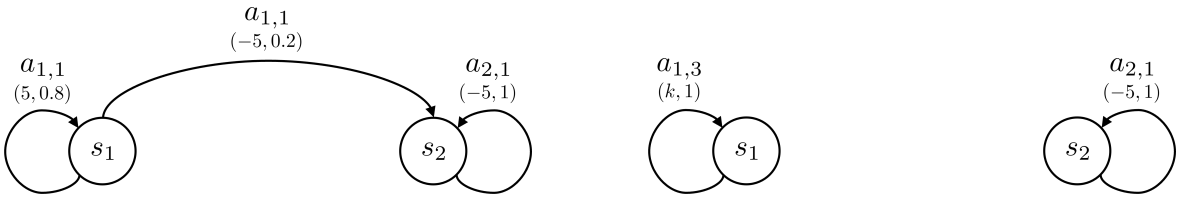


Figure 7.2: Model variant discussed in Example 7.1. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.



(a) Chain structure under policy d^∞ (b) Chain structure under policy $(d')^\infty$

Figure 7.3: Symbolic representation of the two policies discussed in Example 7.1. Observe that under d^∞, s_1 is transient and s_2 is recurrent so that this policy generates a unichain model while under policy $(d')^\infty$, s_1 and s_2 each form closed classes so that the model is multi-chain. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

Example 7.2. This example considers a modified version of the Gridworld model in Section 3.2 in which the robot starts in the coffee room, cell 13, with a full cup of coffee and seeks to deliver it to the office, cell 1. In this model, $S = \{1, \ldots, 15\}$, where each state represents the location of the robot. Recall that actions specify an intended direction for the robot. Letting p denote the probability the robot moves in the intended direction and $(1-p)/(k-1)$ the probability it moves in each of the remaining $k-1$ directions including the possibility of remaining in the same cell.

In this model, cells 1 (deliver coffee) and 7 (fall down the stairs), each correspond to a closed class consisting of one absorbing state. When $p < 1$ the remaining set of states are transient under any decision rule since the robot reaches cells 1 or 7 with probability 1 under any stationary policy. When $p = 1$, decision rules can generate other closed classes. For example, by choosing the action R (right) in cell 2 and L (left) in cell 3 the robot will cycle between cells 2 and 3 forever and never reach cell 1 or cell 7. Moreover when $p = 1$ and it costs 1 unit per transition, then the long run average reward on this closed class is -1 while the average reward on any policy that reaches cells 1 or 7 is zero. Hence in either case the model is multi-chain but when $p < 1$, the only closed classes are $\{1\}$ and $\{7\}$.

A variant: Suppose, as shown in Figure 7.4, the model is augmented by adding an absorbing state Δ and an action a_Δ in states 1 and 7, which results in a transition to Δ with certainty so that $p(\Delta|1, a_\Delta) = p(\Delta|7, a_\Delta) = 1$. Moreover if the system reaches Δ it remains there forever and receives no reward. This can be represented by adding an action a' for which $p(\Delta|\Delta, a') = 1$. Moreover $r(1, a_\Delta, \Delta) = r(7, a_\Delta, \Delta) = r(\Delta, a', \Delta) = 0$. When $p < 1$, $\{\Delta\}$ is the only closed class, all other states are transient and the model is unichain.

When $p < 1$ in both the first formulation above and its variant, the average

reward of every policy is zero (since the robot reaches a zero-reward absorbing state), so the average reward criterion does not distinguish policies. The expected total reward[a], considered in the previous chapter, is a more appropriate optimality criterion for this model.

Another variant: Suppose instead of being absorbed in Δ, an action is added that loops to cell 13 after reaching cell 1 or 7. Then when $p < 1$, the model is regular. For the same reasons as above, when $p = 1$ the model is multi-chain and communicating.

The case of $p = 1$ can also be used to illustrate the construction in the proof of Proposition 7.1. Starting in cell 4, it is easy to see that a different policy is required to reach cells 1 and 6 but by randomizing action choice in cell 4 between the actions "Up" and "Right", the same randomized policy can be used to reach these two cells.

[a] In this model, it is equivalent to the *bias*, a concept defined below.

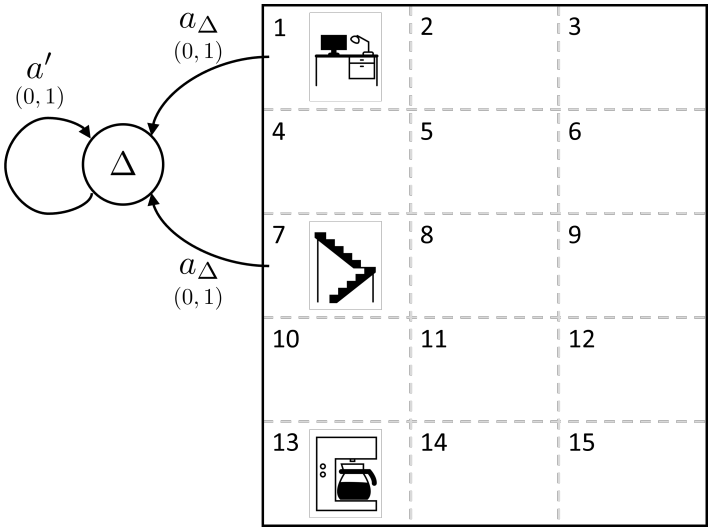


Figure 7.4: Variant of coffee delivering robot model with added absorbing state. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

In other common models, such as the queuing admission control model in Exercise 12, one may encounter real applications that are multi-chain and (weakly) communicating.

7.4 Bias of a stationary policy

In order to develop efficient algorithms to find average optimal policies, this section introduces an additional related quantity referred to as the *bias*. Definitions of the bias require distinguishing between two cases that will be described in separate subsections. The latter is an "edge case" that does not apply to most real applications and can be ignored when reading this for the first time.

1. **Aperiodic case:** When the sub-chains[10] corresponding to all closed classes are aperiodic.
2. **Periodic case:** When at least one closed class has a periodic sub-chain.

The aperiodic case

Definition 7.6. The *bias* of a stationary policy $d^\infty \in \Pi^{\text{SR}}$ in the aperiodic case is defined to be:

$$h^{d^\infty}(s) := E^{d^\infty}\left[\sum_{n=1}^{\infty}\left(r(X_n, Y_n) - g^{d^\infty}(X_n)\right)\middle| X_1 = s\right]. \tag{7.17}$$

In the above definition, the expectation of the infinite sum represents the limit of expectations of partial sums.

Equivalently, the bias of a stationary policy can be written in vector notation as

$$\mathbf{h}^{d^\infty} = \sum_{n=1}^{\infty}\left(\mathbf{P}_d^{n-1}\mathbf{r}_d - \mathbf{g}^{d^\infty}\right) = \sum_{n=1}^{\infty}\left(\mathbf{P}_d^{n-1} - \mathbf{P}_d^*\right)\mathbf{r}_d, \tag{7.18}$$

where the second equality follows from (7.5).

The following frequently used closed form representation for $\mathbf{h}^{d^\infty}$ holds in general, that is for **both** periodic and aperiodic models. It follows directly from Proposition B.1 in Appendix B:

[10] The expression *sub-chain* refers to a Markov chain on a closed class of the original Markov chain.

Theorem 7.3. Let $d \in D^{\text{MR}}$. Then

$$\mathbf{h}^{d^\infty} = \mathbf{H}_d \mathbf{r}_d, \tag{7.19}$$

where[a]

$$\mathbf{H}_d := (\mathbf{I} - (\mathbf{P}_d - \mathbf{P}_d^*))^{-1}(\mathbf{I} - \mathbf{P}_d^*).$$

[a]Note that $\mathbf{H}_d$ refers to the deviation matrix $\mathbf{H}$ introduced in Appendix B applied to the transition probability matrix $\mathbf{P}_d$.

Although Theorems 7.1 and 7.3 provide closed form representations for $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$, they will not be used for computation except in simple examples. Instead, a system of equations will be derived that can be solved to find these quantities.

The periodic case*

Example B.4 describes a Markov chain where the following more general definitions are necessary:

Definition 7.7. The *bias* of a stationary policy $d^\infty \in \Pi^{\text{SR}}$ in the periodic case is defined to be:

$$h^{d^\infty}(s) := \lim_{N\to\infty} \frac{1}{N} \sum_{j=1}^{N} E^{d^\infty} \left[\sum_{n=1}^{j} \left(r(X_n, Y_n) - g^{d^\infty}(X_n) \right) \middle| X_1 = s \right]. \tag{7.20}$$

It is left to the reader to show that when the limit in (7.17) exists the two definitions are equivalent. However, when this limit does not exist, the second definition is necessary.

In the periodic case, when the limit in (7.18) does not exist, the equivalent matrix representation is

$$\mathbf{h}^{d^\infty} = \lim_{N\to\infty} \frac{1}{N} \sum_{j=1}^{N} \sum_{n=1}^{j} \left(\mathbf{P}_d^{n-1} \mathbf{r}_d - \mathbf{g}^{d^\infty} \right) = \lim_{N\to\infty} \frac{1}{N} \sum_{j=1}^{N} \sum_{n=1}^{j} \left(\mathbf{P}_d^{n-1} - \mathbf{P}_d^* \right) \mathbf{r}_d. \tag{7.21}$$

7.4.1 Interpreting the bias

The bias will be explained from three perspectives:

1. Directly from its definition.
2. In relation to the discounted reward as $\lambda \uparrow 1$.
3. In relation to expected total reward.

Using the definition of the bias

The nth term in the sum (7.17), $r(X_n, Y_n) - g^{d^\infty}(X_n)$, represents the difference in the reward received in period n and the gain. Hence the bias represents the expected total sum of these differences.

Turning to the matrix representation for the bias in (7.18), it follows that when $\mathbf{P}_d$ is regular, part 6 of Theorem B.1 implies that $\mathbf{P}_d^n - \mathbf{P}_d^*$ converges to $\mathbf{0}$ exponentially quickly, so that terms in the sum become very small quickly. Hence, the bias may be regarded as the total *initial* difference between the expected reward and the steady state reward. For this reason, the bias is sometimes referred to as the *transient reward*.

Example 7.3. Consider the stationary policy d^∞ where $d(s_1) = a_{1,1}$ and $d(s_2) = a_{2,2}$ in the model in Section 2.5. For this policy

$$\mathbf{P}_d = \begin{bmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{bmatrix} \quad \text{and} \quad \mathbf{r}_d = \begin{bmatrix} 3 \\ 2 \end{bmatrix}.$$

Since $\mathbf{P}_d$ is regular, $\mathbf{P}^n \to \mathbf{P}_d^*$. Define the largest component difference between $\mathbf{P}^n$ and $\mathbf{P}_d^*$ by

$$\Delta_n := \max_{(s,j)\in S\times S} |p_d^n(j|s) - p_d^*(j|s)|.$$

Direct computation shows that $\Delta_2 = 0.1067$, $\Delta_3 = 0.0427$, $\Delta_4 = 0.0171$, ... indicative of the exponential convergence of $\mathbf{P}_d^n$ to $\mathbf{P}_d^*$. Because the eigenvalues of $\mathbf{P}_d$ are 1 and 0.4 the convergence is exponential at the rate of the second eigenvalue as noted in the Markov chain appendix.

Moreover,

$$\mathbf{P}_d^* = \begin{bmatrix} 2/3 & 1/3 \\ \\ 2/3 & 1/3 \end{bmatrix} \quad \text{and} \quad \mathbf{g}^{d^\infty} = \begin{bmatrix} 8/3 \\ \\ 8/3 \end{bmatrix}.$$

Observe that the rows of $\mathbf{P}_d^*$ are equal and so are the components of $\mathbf{g}^{d^\infty}$.

Evaluating the bias of d^∞ yields:

$$\mathbf{H}_d = \begin{bmatrix} 5/9 & -5/9 \\ \\ -10/9 & 10/9 \end{bmatrix} \quad \text{and} \quad \mathbf{h}^{d^\infty} = \mathbf{H}_d\mathbf{r}_d = \begin{bmatrix} 5/9 \\ \\ -10/9 \end{bmatrix}.$$

Let

$$\mathbf{S}_N := \sum_{n=1}^{N} \left(\mathbf{P}_d^{n-1}\mathbf{r}_d - \mathbf{g}^{d^\infty}\right).$$

denote the N-th partial sum in (7.18). Table 7.2 below provides the first few terms and the limit of $\mathbf{S}_N$. It shows that $\mathbf{S}_N \to \mathbf{h}^{d^\infty} = (5/9, -10/9)$. Observe that starting in s_1 the bias is positive since the expected reward $r_d(s_1)$ exceeds $g^{d^\infty}(s_1)$ and negative when starting in s_2 since $r_d(s_2)$ is less than $g^{d^\infty}(s_2)$.

	$N=1$	$N=2$	$N=3$	...	$N=\infty$
$S_N(s_1)$	0.33	0.47	0.52	...	0.5556
$S_N(s_2)$	−0.67	−0.93	1.04	...	−1.1111

Table 7.2: Terms in $\mathbf{S}_N$.

Relationship to the discounted reward

The most elegant analysis of finite state average reward models uses the relationship between the discounted reward and the average reward referred to as the *vanishing discount rate* approach. It is based on the following key result. The straightforward proof, provided in the aperiodic case[11], is informative as it includes several calculations that will be used below.

Theorem 7.4. Let S be finite, $d \in D^{\mathrm{MR}}$, and $0 \le \lambda < 1$. Then

$$\mathbf{v}_\lambda^{d^\infty} = \frac{\mathbf{g}^{d^\infty}}{1-\lambda} + \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda), \tag{7.22}$$

where $\mathbf{h}^{d^\infty} = \mathbf{H}_d\mathbf{r}_d$ and $\mathbf{f}(\lambda)$ is a vector that converges to $\mathbf{0}$ as $\lambda \uparrow 1$.

Proof. From (5.29) the expected discounted reward of stationary policy d^∞ may be written as

$$\mathbf{v}_\lambda^{d^\infty} = \sum_{n=0}^{\infty} \lambda^n \mathbf{P}_d^n \mathbf{r}_d.$$

Adding and subtracting $\mathbf{P}_d^*$ inside the summation yields:

$$\sum_{n=0}^{\infty} \lambda^n \mathbf{P}_d^n \mathbf{r}_d = \sum_{n=0}^{\infty} \lambda^n (\mathbf{P}_d^* + \mathbf{P}_d^n - \mathbf{P}_d^*)\mathbf{r}_d$$

[11] A proof in the periodic case requires additional averaging as noted in the definitions above.

$$
\begin{aligned}
&= \sum_{n=0}^{\infty} \lambda^n \mathbf{P}_d^* \mathbf{r}_d + \left(\mathbf{I} - \mathbf{P}_d^* + \sum_{n=1}^{\infty} \lambda^n (\mathbf{P}_d^n - \mathbf{P}_d^*) \right) \mathbf{r}_d \\
&= \frac{\mathbf{P}_d^* \mathbf{r}_d}{1-\lambda} + \left(\sum_{n=0}^{\infty} \lambda^n (\mathbf{P}_d - \mathbf{P}_d^*)^n - \mathbf{P}_d^* \right) \mathbf{r}_d \qquad (7.23) \\
&= \frac{\mathbf{P}_d^* \mathbf{r}_d}{1-\lambda} + \left((\mathbf{I} - \lambda(\mathbf{P}_d - \mathbf{P}_d^*))^{-1} - \mathbf{P}_d^* \right) \mathbf{r}_d \qquad (7.24) \\
&= \frac{\mathbf{P}_d^* \mathbf{r}_d}{1-\lambda} + \mathbf{H}_d \mathbf{r}_d + \left((\mathbf{I} - \lambda(\mathbf{P}_d - \mathbf{P}_d^*))^{-1} - \mathbf{P}_d^* - \mathbf{H}_d \right) \mathbf{r}_d \qquad (7.25) \\
&= \frac{\mathbf{g}^{d^\infty}}{1-\lambda} + \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda),
\end{aligned}
$$

where

$$
\mathbf{f}(\lambda) := \left((\mathbf{I} - \lambda(\mathbf{P}_d - \mathbf{P}_d^*))^{-1} - \mathbf{P}_d^* - \mathbf{H}_d \right) \mathbf{r}_d.
$$

An explanation of some key steps above follows. That $\mathbf{P}_d^n - \mathbf{P}_d^* = (\mathbf{P}_d - \mathbf{P}_d^*)^n$ for $n \geq 1$ in (7.23) follows from properties of $\mathbf{P}_d^*$ in part 2 of Theorem B.1. Its derivation is left as an exercise. The existence and representation for the inverse in (7.24) for $\lambda \leq 1$ follows from Proposition B.1 in Appendix B. Equation (7.25) follows by adding and subtracting $\mathbf{H}_d$ in (7.24) and the final equality follows from the representations for $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$ in Theorems 7.1 and 7.3. Lastly, noting that $\mathbf{H}_d$ can be defined equivalently as $(\mathbf{I} - (\mathbf{P}_d - \mathbf{P}_d^*))^{-1} - \mathbf{P}_d^*$ (see Proposition B.1), $\mathbf{f}(\lambda)$ converges to $\mathbf{0}$, completing the proof. □

The expression (7.22) is referred to as the *partial Laurent expansion* of the expected discounted reward. When $\mathbf{f}(\lambda)$ is written as an infinite series, the representation is referred to as the *Laurent expansion* of the expected discounted reward. It is the basis for the concept of sensitive discount optimality[12], which will not be discussed in this book.

Note that the above result provides the approximation for the bias:

$$
\mathbf{h}^{d^\infty} \approx \mathbf{v}_\lambda^{d^\infty} - \frac{\mathbf{g}^{d^\infty}}{1-\lambda}. \qquad (7.26)
$$

This means that the bias of a stationary policy approximately equals the difference between its expected discounted reward and the expected discounted reward in a system that accrues the average reward every period.

The above proof also provides the matrix relationship

$$
\sum_{n=0}^{\infty} \lambda^n \mathbf{P}_d^n = \frac{\mathbf{P}_d^*}{1-\lambda} + \mathbf{H}_d + \mathbf{F}(\lambda), \qquad (7.27)
$$

where all of the components of the matrix $\mathbf{F}(\lambda)$ converge to 0 as $\lambda \uparrow 1$.

[12] See Chapter 10 in Puterman 1994

The following corollary uses Theorem 7.4 to relate the discounted reward to the average reward[13].

Corollary 7.2. Let $d \in D^{\mathrm{MR}}$. Then

$$\lim_{\lambda \uparrow 1} (1-\lambda)\mathbf{v}_\lambda^{d^\infty} = \mathbf{g}^{d^\infty} \tag{7.28}$$

The following table continues Example 7.3 by illustrating the calculations implicit in Theorem 7.4 and Corollary 7.2. Observe that approximations are accurate to two decimal places for $\lambda = 0.99$ and three decimal places for $\lambda = 0.999$.

λ	$(1-\lambda)\mathbf{v}_\lambda^{d^\infty}$	$\mathbf{v}_\lambda^{d^\infty} - \left((1-\lambda)^{-1}\mathbf{g}^{d^\infty} - \mathbf{h}^\infty\right)$
0.9	$(2.719, 2.563)$	$(-0.0347, 0.0694)$
0.95	$(2.694, 2.613)$	$(-0.0179, 0.0358)$
0.99	$(2.672, 2.656)$	$(-0.0037, 0.0074)$
0.999	$(2.667, 2.666)$	$(-0.0004, 0.0007)$

Table 7.3: Illustration of the approximation to $\mathbf{g}^{d^\infty} = (2.667, 2.667)$ based on Corollary 7.2 and the accuracy of the partial Laurent series approximation in Theorem 7.4 as a function of λ for the policy in Example 7.3.

Relationship to the expected total reward

The following theorem describes the relationship between the average reward and the expected total reward of a stationary policy in the aperiodic case. A representation like (7.5) in the periodic case requires averaging (Exercise 4).

[13]Results of this kind are often called Tauberian theorems because they relate two different ways of summing a divergent infinite series (see Derman [1970] or Sennott [1999]). For a sequence of real numbers $a_n, n = 0, 1, 2, \ldots$ it is analogous to

$$\lim_{\lambda \uparrow 1} (1-\lambda) \sum_{n=0}^{\infty} a_n \lambda^n = \lim_{N \to \infty} \sum_{n=1}^{N} a_n$$

Theorem 7.5. Let $d \in D^{\text{MR}}$. Then if the Markov chain corresponding to $\mathbf{P}_d$ has no closed periodic classes

$$\mathbf{v}_N^{d^\infty} = N\mathbf{g}^{d^\infty} + \mathbf{h}^{d^\infty} + \mathbf{o}(1), \tag{7.29}$$

where $\mathbf{o}(1)$ is a vector that converges to $\mathbf{0}$ as $N \to \infty$.

Proof. Adding and subtracting $\mathbf{P}_d^*$ from the matrix representation for the expected total reward over N decision epochs yields:

$$\begin{aligned}
\mathbf{v}_N^{d^\infty} &= \sum_{n=1}^{N} \mathbf{P}_d^{n-1}\mathbf{r}_d \qquad (7.30)\\
&= \sum_{n=1}^{N} (\mathbf{P}_d^* + \mathbf{P}_d^{n-1} - \mathbf{P}_d^*)\mathbf{r}_d \\
&= N\mathbf{P}_d^*\mathbf{r}_d + \sum_{n=1}^{N} (\mathbf{P}_d^{n-1} - \mathbf{P}_d^*)\mathbf{r}_d \\
&= N\mathbf{P}_d^*\mathbf{r}_d + \sum_{n=1}^{N} (\mathbf{P}_d^{n-1} - \mathbf{P}_d^*)\mathbf{r}_d + \sum_{n=N+1}^{\infty} (\mathbf{P}_d^{n-1} - \mathbf{P}_d^*)\mathbf{r}_d - \sum_{n=N+1}^{\infty} (\mathbf{P}_d^{n-1} - \mathbf{P}_d^*)\mathbf{r}_d \\
&= N\mathbf{P}_d^*\mathbf{r}_d + \mathbf{H}_d\mathbf{r}_d - \left(\sum_{n=N+1}^{\infty} (\mathbf{P}_d^{n-1} - \mathbf{P}_d^*) \right) \mathbf{r}_d, \qquad (7.31)
\end{aligned}$$

where the last expression in (7.31) is $\mathbf{o}(1)$ because $\sum_{n=1}^{N}(\mathbf{P}_d^{n-1} - \mathbf{P}_d^*)$ converges to $\mathbf{H}_d$ as a consequence of Proposition B.1. □

This result means that for each $s \in S$, $v_N^{d^\infty}(s)$ is *approximately linear* in N with slope $g^{d^\infty}(s)$ and intercept $h^{d^\infty}(s)$ (see Figure 7.5). Furthermore this theorem provides another interpretation for the bias as

$$\mathbf{h}^{d^\infty} \approx \mathbf{v}_N^{d^\infty} - N\mathbf{g}^{d^\infty}. \tag{7.32}$$

An immediate consequence of Theorem 7.5 follows.

Corollary 7.3. Let $d \in D^{\text{MR}}$. Then

$$\mathbf{g}^{d^\infty} = \lim_{N\to\infty} \frac{1}{N}\mathbf{v}_N^{d^\infty}. \tag{7.33}$$

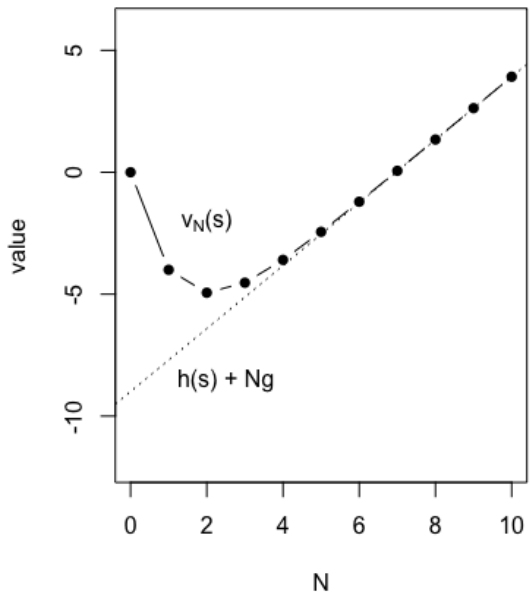


Figure 7.5: Sample plot of $v_N^{d^\infty}(s)$ vs. $h^{d^\infty}(s) + Ng^{d^\infty}$.

Observe that Theorem 7.5 also suggests the following approximation

$$\mathbf{g}^{d^\infty} \approx \mathbf{v}_N^{d^\infty} - \mathbf{v}_{N-1}^{d^\infty}, \tag{7.34}$$

which will be very useful when analyzing value iteration in average reward models.

The following discussion illustrates these results in the context of the decision rule in Example 7.3. Table 7.4 shows that $\mathbf{v}_N^{d^\infty} - \mathbf{v}_{N-1}^{d^\infty}$ better approximates $\mathbf{g}^{d^\infty}$ than $\mathbf{v}_N^{d^\infty}/N$. In fact the approximation by $\mathbf{v}_N^{d^\infty} - \mathbf{v}_{N-1}^{d^\infty}$ is accurate to three decimal places when $N = 11$. Note also that the approximation of the expected total reward based on (7.29) is very accurate even for small N.

7.4.2 Computing the gain and bias of a stationary policy

This section shows how to compute the gain and bias without evaluating $\mathbf{P}_d^*$ and $\mathbf{H}_d$. It derives a system of equations that when solved yields the gain and bias, and moreover can be generalized to obtain the Bellman equation for average reward models. The derivation is motivated in two ways: using the partial Laurent expansion of the expected discounted reward and using the approximation based on the finite horizon expected reward described in the previous section.

It will show that the gain and bias satisfy the following system of matrix equations[14]:

[14]Many of you may be familiar with this system without the first equation. This is because in most

N	$\frac{1}{N}\mathbf{v}_N^{d^\infty}$	$\mathbf{v}_N^{d^\infty} - \mathbf{v}_{N-1}^{d^\infty}$	$\mathbf{v}_N^{d^\infty} - \left(N\mathbf{g}^{d^\infty} - \mathbf{h}^\infty\right)$
5	$(2.777, 2.447)$	$(2.675, 2.644)$	$(-0.006, 0.011)$
10	$(2.722, 2.556)$	$(2.667, 2.666)$	$(-0.0001, 0.0001)$
20	$(2.694, 2.611)$	$(2.667, 2.667)$	*
50	$(2.678, 2.644)$	$(2.667, 2.667)$	*

Table 7.4: Illustration of the approximation to $\mathbf{g}^{d^\infty} = (2.667, 2.667)$ based on Corollary 7.3 and (7.34). The last column shows the accuracy of the approximation in Theorem 7.5 as a function of N for the policy in Example 7.3. The entries denoted by * are less than 10^{-8}.

Average reward evaluation equations

$$\mathbf{g} = \mathbf{P}_d\mathbf{g} \quad \text{and} \quad \mathbf{h} = \mathbf{r}_d - \mathbf{g} + \mathbf{P}_d\mathbf{h}. \tag{7.35}$$

These equations are sometimes written as

$$(\mathbf{I} - \mathbf{P}_d)\mathbf{g} = \mathbf{0} \quad \text{and} \quad \mathbf{g} + (\mathbf{I} - \mathbf{P}_d)\mathbf{h} = \mathbf{r}_d, \tag{7.36}$$

with all variables on the left hand side. For a deterministic decision rule, (7.35) can be expressed in component notation as:

$$g(s) = \sum_{s\in S} p(j|s, d(s))g(j) \quad \text{and} \quad h(s) = r(s, d(s)) - g(s) + \sum_{j\in S} p(j|s, d(s))h(j). \tag{7.37}$$

An obvious modification is required for a randomized decision rule.

models in the literature, $\mathbf{g}^{d^\infty}$ is a constant vector so this equation becomes redundant. This issue will be discussed below.

Heuristic derivation based on the partial Laurent series approximation to the discounted reward.

As a result of equation (5.31), the expected discounted reward of stationary policy d^∞, $\mathbf{v}_\lambda^{d^\infty}$ satisfies

$$\mathbf{v} = \mathbf{r}_d + \lambda \mathbf{P}_d \mathbf{v}.$$

Substituting (7.22) into both sides of this equation yields

$$\begin{aligned}\frac{\mathbf{g}^{d^\infty}}{1-\lambda} + \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda) &= \mathbf{r}_d + \lambda \mathbf{P}_d \left(\frac{\mathbf{g}^{d^\infty}}{1-\lambda} + \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda) \right) \\ &= \mathbf{r}_d + \frac{\mathbf{P}_d \mathbf{g}^{d^\infty}}{1-\lambda} - \mathbf{P}_d \mathbf{g}^{d^\infty} + \lambda \mathbf{P}_d \mathbf{h}^{d^\infty} + \lambda \mathbf{P}_d \mathbf{f}(\lambda).\end{aligned}$$

For this last relationship to be valid in the limit as $\lambda \uparrow 1$ requires that $\mathbf{g}^{d^\infty} = \mathbf{P}_d \mathbf{g}^{d^\infty}$. When this is the case[15]

$$\begin{aligned}\mathbf{h}^{d^\infty} &= \mathbf{r}_d - \mathbf{P}_d \mathbf{g}^{d^\infty} + \lambda \mathbf{P}_d \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda) \\ &= \mathbf{r}_d - \mathbf{g}^{d^\infty} + \lambda \mathbf{P}_d \mathbf{h}^{d^\infty} + \mathbf{f}(\lambda).\end{aligned}$$

Letting $\lambda \uparrow 1$ shows that $\mathbf{h}^{d^\infty} = \mathbf{r}_d - \mathbf{g}^{d^\infty} + \mathbf{P}_d \mathbf{h}^{d^\infty}$. Hence $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$ must be solutions of (7.35).

Heuristic derivation based on the expected total reward approximation.

From equation (6.24), subject to $\mathbf{v}_0 = \mathbf{0}$, the expected total reward over n periods of stationary policy d^∞, $\mathbf{v}_n^{d^\infty}$, satisfies

$$\mathbf{v}_{n+1} = \mathbf{r}_d + \mathbf{P}_d \mathbf{v}_n$$

for $n = 0, 1, \ldots$. Substituting (7.29) into both sides of this equation yields

$$(n+1)\mathbf{g}^{d^\infty} + \mathbf{h}^{d^\infty} + \mathbf{o}(1) = \mathbf{r}_d + \mathbf{P}_d \left(n\mathbf{g}^{d^\infty} + \mathbf{h}^{d^\infty} + \mathbf{o}(1) \right)$$

so that

$$n\mathbf{g}^{d^\infty} + \mathbf{h}^{d^\infty} = \mathbf{r}_d - \mathbf{g}^{d^\infty} + n\mathbf{P}_d \mathbf{g}^{d^\infty} + \mathbf{P}_d \mathbf{h}^{d^\infty} + \mathbf{o}(1).$$

For this to hold for all $n = 0, 1, \ldots$ requires that the expressions multiplied by n be equal. Hence $\mathbf{g}^{d^\infty} = \mathbf{P}_d \mathbf{g}^{d^\infty}$. If this holds it follows that $\mathbf{h}^{d^\infty} = \mathbf{r}_d - \mathbf{g}^{d^\infty} + \mathbf{P}_d \mathbf{h}^{d^\infty}$ as $n \to \infty$. Consequently $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$ must be solutions of (7.35).

[15] The precise form of $\mathbf{f}(\lambda)$ is not required, only that it converges to $\mathbf{0}$ as $\lambda \uparrow 1$. So analogously to "small o notation" $\mathbf{f}(\lambda)$ represents $\lambda \mathbf{P}_d \mathbf{f}(\lambda) - \mathbf{f}(\lambda)$.

An example

Before describing some important properties of the system of equations (7.35), an analysis of Example 7.3 illustrates the subtleties involved in solving this system.

For the decision rule d^∞ analyzed in Example 7.3, (7.35) becomes:

$$\begin{bmatrix} g(s_1) \\ g(s_2) \end{bmatrix} = \begin{bmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{bmatrix} \begin{bmatrix} g(s_1) \\ g(s_2) \end{bmatrix}$$

and

$$\begin{bmatrix} h(s_1) \\ h(s_2) \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix} - \begin{bmatrix} g(s_1) \\ g(s_2) \end{bmatrix} + \begin{bmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{bmatrix} \begin{bmatrix} h(s_1) \\ h(s_2) \end{bmatrix}.$$

Expressing this system of equations as

$$(\mathbf{I} - \mathbf{P}_d)\mathbf{g} = \mathbf{0} \quad \text{and} \quad \mathbf{I}\mathbf{g} + (\mathbf{I} - \mathbf{P}_d)\mathbf{h} = \mathbf{r}_d \tag{7.38}$$

they can be combined and written in terms of a partitioned matrix as:

$$\begin{bmatrix} \mathbf{I} - \mathbf{P}_d & \mathbf{0} \\ \mathbf{I} & \mathbf{I} - \mathbf{P}_d \end{bmatrix} \begin{bmatrix} \mathbf{g} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{r}_d \end{bmatrix}. \tag{7.39}$$

Substituting appropriate values, this matrix equation becomes

$$\begin{bmatrix} 0.2 & -0.2 & 0 & 0 \\ -0.4 & 0.4 & 0 & 0 \\ 1 & 0 & 0.2 & -0.2 \\ 0 & 1 & -0.4 & 0.4 \end{bmatrix} \begin{bmatrix} g(s_1) \\ g(s_2) \\ h(s_1) \\ h(s_2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 3 \\ 2 \end{bmatrix}. \tag{7.40}$$

Applying Gauss-Jordan elimination[16] reduces this equation to

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} g(s_1) \\ g(s_2) \\ h(s_1) \\ h(s_2) \end{bmatrix} = \begin{bmatrix} 8/3 \\ 0 \\ 8/3 \\ 5/3 \end{bmatrix}. \tag{7.41}$$

Hence $g(s_1) = g(s_2) = 8/3$ and $h(s_1) - h(s_2) = 5/3$, so calculations in Example 7.3 show that $\mathbf{g} = \mathbf{g}^{d^\infty}$. Moreover for an arbitrary constant c,

$$\mathbf{h} = \begin{bmatrix} 5/3 \\ 0 \end{bmatrix} + c \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

where the vector $(5/3, 0)$ corresponds to the solution when $h(s_2) = c = 0$. By choosing $c = -10/9$ it follows that $\mathbf{h} = \mathbf{h}^{d^\infty}$.

[16] Some might refer to this as Gaussian elimination.

Making this formal

This above example illustrates the following key features of the system of equations (7.35):

1. When $\mathbf{g}$ satisfies these equations, $\mathbf{g} = \mathbf{g}^{d^\infty}$.
2. These equations do not uniquely determine $\mathbf{h}$. As can be seen directly from (7.40) or from the observation that the second row of the reduced matrix (7.41) contains all zero entries, the partitioned matrix in (7.39) is not full rank[17].
3. The singular matrix $\mathbf{I} - \mathbf{P}_d$ plays a key role in finding $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$.

These observations are formalized and extended in the following important theorem.

Theorem 7.6. Let $d \in D^{\text{MR}}$.

1. Then $\mathbf{g}^{d^\infty} = \mathbf{P}_d^* \mathbf{r}_d$ and $\mathbf{h}^{d^\infty} = \mathbf{H}_d \mathbf{r}_d$ are solutions of (7.35).
2. Suppose $(\mathbf{g}, \mathbf{h})$ is a solution of (7.35). Then $\mathbf{g} = \mathbf{g}^{d^\infty}$ and $\mathbf{h} = \mathbf{h}^{d^\infty} + \mathbf{u}$ where $(\mathbf{I} - \mathbf{P}_d)\mathbf{u} = \mathbf{0}$.
3. Suppose $(\mathbf{g}, \mathbf{h})$ is a solution of (7.35) and $\mathbf{P}_d^* \mathbf{h} = \mathbf{0}$. Then $\mathbf{g} = \mathbf{g}^{d^\infty}$ and $\mathbf{h} = \mathbf{h}^{d^\infty}$.

Proof. Since $\mathbf{P}_d^* = \mathbf{P}_d \mathbf{P}_d^*$, that $\mathbf{g}^{d^\infty}$ satisfies the first equation in (7.35) follows from noting that

$$(\mathbf{I} - \mathbf{P}_d)\mathbf{g}^{d^\infty} = (\mathbf{I} - \mathbf{P}_d)\mathbf{P}_d^* \mathbf{r}_d = \mathbf{0}.$$

To establish the second equality in (7.35), note that

$$\begin{aligned} \mathbf{g}^{d^\infty} + (\mathbf{I} - \mathbf{P}_d)\mathbf{h}^{d^\infty} &= \mathbf{P}_d^* \mathbf{r}_d + (\mathbf{I} - \mathbf{P}_d)\mathbf{H}_d \mathbf{r}_d \\ &= \mathbf{P}_d^* \mathbf{r}_d + (\mathbf{I} - \mathbf{P}_d^*)\mathbf{r}_d \\ &= \mathbf{r}_d, \end{aligned}$$

where the second equality follows from the expression $(\mathbf{I} - \mathbf{P}_d)\mathbf{H}_d = \mathbf{I} - \mathbf{P}_d^*$ in (B.16) in Appendix B. Hence part 1 is proved.

To prove[18] the result in part 2 let $(\mathbf{g}_1, \mathbf{h}_1)$ and $(\mathbf{g}_2, \mathbf{h}_2)$ denote two solutions of (7.35) and $\Delta\mathbf{g} := \mathbf{g}_1 - \mathbf{g}_2$ and $\Delta\mathbf{h} := \mathbf{h}_1 - \mathbf{h}_2$. Then it is easy to see that

$$(\mathbf{I} - \mathbf{P}_d)\Delta\mathbf{g} = \mathbf{0} \quad \text{and} \quad \Delta\mathbf{g} = (\mathbf{P}_d - \mathbf{I})\Delta\mathbf{h}. \tag{7.42}$$

[17] In the example it has three independent rows and one redundant row and $\mathbf{I} - \mathbf{P}_d$ has one redundant row.

[18] An alternative proof that $\Delta\mathbf{g} = \mathbf{0}$ is obtaining by multiplying the second equation in (7.42) by $\mathbf{P}_d^*$, to obtain $\mathbf{P}_d^* \Delta\mathbf{g} = \mathbf{0}$ and then adding this to the first expression to obtain $(\mathbf{I} - (\mathbf{P}_d - \mathbf{P}_d^*))\Delta\mathbf{g} = \mathbf{0}$. Since this matrix is invertible, $\Delta\mathbf{g} = \mathbf{0}$.

Multiplying both sides of the second equation on the left by $\mathbf{P}_d$ and noting the first equation gives

$$\Delta\mathbf{g} = \mathbf{P}_d\Delta\mathbf{g} = (\mathbf{P}_d^2 - \mathbf{P}_d)\Delta\mathbf{h}.$$

Repeating this argument (or applying induction) establishes that for all $n \geq 0$,

$$\Delta\mathbf{g} = (\mathbf{P}_d^n - \mathbf{P}_d^{n-1})\Delta\mathbf{h}.$$

Adding these expressions for $\Delta\mathbf{g}$ together yields

$$n\Delta\mathbf{g} = (\mathbf{P}_d^n - \mathbf{I})\Delta\mathbf{h}.$$

Dividing both sides by n and take the limit as $n \to \infty$ establishes that

$$\Delta\mathbf{g} = \lim_{n\to\infty} \frac{1}{n}(\mathbf{P}_d^n - \mathbf{I})\Delta\mathbf{h} = \mathbf{0}.$$

Hence $\mathbf{g}_1 = \mathbf{g}_2 = \mathbf{g}^{d^\infty}$, where the last equality follows from part 1. Substituting $\Delta\mathbf{g} = \mathbf{0}$ into the second equation in (7.42) shows that $(\mathbf{I} - \mathbf{P}_d)\Delta\mathbf{h} = \mathbf{0}$. Part 1 established that $\mathbf{h}^{d^\infty}$ satisfies

$$\mathbf{r}_d = \mathbf{g}^{d^\infty} + (\mathbf{I} - \mathbf{P}_d)\mathbf{h}.$$

So setting $\mathbf{h}_1 = \mathbf{h}^{d^\infty}$ implies that $\mathbf{h}_2 = \mathbf{h}^{d^\infty} + \Delta\mathbf{h}$, which completes the proof of part 2.

To establish part 3 add $\mathbf{P}_d^*\mathbf{h} = \mathbf{0}$ to the right hand side of $\mathbf{r}_d = \mathbf{g}^{d^\infty} + (\mathbf{I} - \mathbf{P}_d)\mathbf{h}$ and note that $\mathbf{g}^{d^\infty} = \mathbf{P}_d^*\mathbf{r}_d$ and rearrange terms to obtain

$$(\mathbf{I} - \mathbf{P}_d^*)\mathbf{r}_d = (\mathbf{I} - \mathbf{P}_d + \mathbf{P}_d^*)\mathbf{h}.$$

Multiplying both sides on the left by $(\mathbf{I} - \mathbf{P}_d + \mathbf{P}_d^*)^{-1}$ yields

$$\mathbf{h} = (\mathbf{I} - \mathbf{P}_d + \mathbf{P}_d^*)^{-1}(\mathbf{I} - \mathbf{P}_d^*)\mathbf{r}_d = \mathbf{H}_d\mathbf{r}_d = \mathbf{h}^{d^\infty}$$

completing the proof. □

Some implications of this result follow:

1. It will be shown below that the system of equations (7.35) provides the basis for the average reward Bellman equation. Consequently, understanding its properties is fundamental.

2. Theorem 7.6 establishes that (7.35) uniquely determines $\mathbf{g}^{d^\infty}$ and determines $\mathbf{h}^{d^\infty}$ up to a vector in the null space[19] of $\mathbf{I} - \mathbf{P}_d$. Part 3 provides a way of identifying $\mathbf{h}^{d^\infty}$ but it requires determining $\mathbf{P}_d^*$, which can be burdensome.

[19]The vector $\mathbf{x} \neq \mathbf{0}$ is in the *null space* of the matrix $\mathbf{A}$ if $\mathbf{A}\mathbf{x} = \mathbf{0}$.

3. The equation $(\mathbf{I}-\mathbf{P}_d)\mathbf{g}=\mathbf{0}$ shows only that $\mathbf{g}$ is an element of the null space of $\mathbf{I}-\mathbf{P}_d$. Since the null space of $\mathbf{I}-\mathbf{P}_d$ is equivalent to the space of right eigenvectors of $\mathbf{P}_d$ corresponding to the eigenvalue 1, this means that this equation determines $\mathbf{g}$ up to m arbitrary constants where m represents the number of closed classes of $\mathbf{P}_d$. Special attention will be paid to the case $m=1$ where this system of equations simplifies further as the corollary below shows.

4. As an alternative to adding the condition $\mathbf{P}^*\mathbf{h}=\mathbf{0}$ to uniquely determine $\mathbf{h}^{d^\infty}$ one can instead add the *third* equation

$$-\mathbf{h}+(\mathbf{P}_d-\mathbf{I})\mathbf{w}=\mathbf{0}$$

to the system (7.35). Solving this system of three matrix equations uniquely determines $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$ and determines $\mathbf{w}$ up to an element of the null space of $\mathbf{I}-\mathbf{P}_d$. When $\mathbf{w}$ is uniquely determined by the condition $\mathbf{P}_d^*\mathbf{w}=\mathbf{0}$ or by adding a *fourth* equation, it corresponds to the third term in the Laurent series[20] expansion of $\mathbf{v}_\lambda^{d^\infty}$.

Simplifications for regular and unichain models

The following corollary provides simplified evaluation equations that apply when a Markovian decision rule has a single closed class, that is, it is recurrent or unichain. It follows from the previous theorem by noting for such a model that $\mathbf{P}_d^*$ has equal rows so that all components of g^{d^∞} are equal, and so that the equation $(\mathbf{I}-\mathbf{P}_d)\mathbf{g}=\mathbf{0}$ is superfluous. Note this is the form of the evaluation you are most likely to come across in the literature.

Corollary 7.4. Suppose $\mathbf{P}_d$ is the transition probability matrix of a Markovian decision rule d in a recurrent or unichain model.

1. If $(\mathbf{I}-\mathbf{P}_d)\mathbf{g}=\mathbf{0}$, then $\mathbf{g}=g\mathbf{e}$, for some scalar g[a].

2. Suppose $(g,\mathbf{h})$ satisfy
$$g\mathbf{e}+(\mathbf{I}-\mathbf{P}_d)\mathbf{h}=\mathbf{r}_d. \tag{7.43}$$
Then $g\mathbf{e}=\mathbf{g}^{d^\infty}$ and $\mathbf{h}=\mathbf{h}^{d^\infty}+k\mathbf{e}$, where k is a scalar.

3. Suppose $(g,\mathbf{h})$ satisfy
$$g\mathbf{e}+(\mathbf{I}-\mathbf{P}_d)\mathbf{h}=\mathbf{r}_d$$
and $\mathbf{P}_d^*\mathbf{h}=\mathbf{0}$. Then $g\mathbf{e}=\mathbf{g}^{d^\infty}$ and $\mathbf{h}=\mathbf{h}^{d^\infty}$.

[a] Recall that $\mathbf{e}$ denotes a vector with all components equal to one.

[20] See Theorem 8.2.8 and Chapter 9 in Puterman [1994] for details on this concept.

Hence for recurrent and unichain models, $\mathbf{g}^{d^\infty}$ and $\mathbf{h}^{d^\infty}$, up to a constant, can be obtained by solving (7.43). For $d \in D^{\mathrm{MD}}$ and all $s \in S$, (7.43) can be expressed in component notation as:

$$g + h(s) - \sum_{j \in S} p(j|s, d(s))h(j) = r(s, d(s)) \tag{7.44}$$

Of course the standard modification applies when $d \in D^{\mathrm{MR}}$.

Example 7.4. This examples continues Example 7.3. For the stationary policy analyzed there, $\mathbf{P}_d$ is recurrent (in fact it is regular) so that results in Corollary 7.4 can be used to find its gain and bias. Equation (7.44) becomes

$$\begin{aligned} g + 0.2h(s_1) - 0.2h(s_2) &= 3 \\ g - 0.4h(s_1) + 0.4h(s_2) &= 2. \end{aligned}$$

Note that this under-determined system has three variables and two unknowns.

Multiplying the first equation by 2 and adding it to the second equation shows that $g = 8/3$. Substituting this value back into the first (or second) equation shows that $h(s_1) - h(s_2) = 5/3$. Thus $\mathbf{h}$ is unique up to an additive constant and agrees with previous analyses.

Since

$$\mathbf{P}_d^* = \begin{bmatrix} 2/3 & 1/3 \\ \\ 2/3 & 1/3 \end{bmatrix},$$

the condition $\mathbf{P}_d^*\mathbf{h} = \mathbf{0}$ implies that $2h(s_1)/3 + h(s_2)/3 = 0$ or equivalently $h(s_2) = -2h(s_1)$. Hence $h^{d^\infty}(s_1) = 5/9$ and $h^{d^\infty}(s_2) = -10/9$.

Alternatively other particular solutions can be obtained by setting $h(s_2) = 0$ to yield $h(s_1) = 5/3$ or setting $h(s_1) = 0$ to yield $h(s_2) = -5/3$.These solutions will be referred to as relative values and defined formally below.

Relative values

It is sometimes sufficient in a recurrent or unichain model to distinguish specific representations for $\mathbf{h}$, referred to as *relative values*.

Definition 7.8. The *relative value* of d^∞ for a specified $s^* \in S$, is the $\mathbf{h}$ part of the solution of (7.43) subject to the condition $h(s^*) = 0$. It is denoted by $\mathbf{h}_{\mathrm{rel}}^{d^\infty}$.

The calculation in the previous section distinguished relative values with respect to

$s^* = s_1$ and $s^* = s_2$. More generally if $(g, \mathbf{h})$ satisfy (7.43), then

$$h_{\text{rel}}^{d^\infty}(s) = h(s) - h(s^*). \tag{7.45}$$

In some applications, for instance in single-server queuing control models or single-product inventory models, it is convenient to set $s^* = 0$ when computing the relative values. Most importantly it will be seen below that relative values suffice for optimization, so from this perspective it is not necessary to undertake the extra effort to compute $\mathbf{h}^{d^\infty}$ (unless you want to).

7.5 The Bellman equation and its properties

As in the case of other optimality criteria, the Bellman equation plays a fundamental role in average reward Markov decision processes. The key difference is that multi-chain models require a pair of nested Bellman equations, while recurrent and unichain models require only a single (vector) equation. Because of this, this section will focus on recurrent and unichain models. A brief discussion of the multi-chain Bellman equation will be included for completeness[21].

The main results in this section are that:

1. The Bellman equation has a solution.
2. The first component of the solution corresponds to the optimal gain[22].
3. The Bellman equation provides the basis for algorithms for finding average optimal policies.

7.5.1 The Bellman equation in recurrent and unichain models

The Bellman equation for a recurrent or unichain model is given in component form by:

$$h(s) = \max_{a \in A_s} \left\{ r(s,a) - g + \sum_{j \in S} p(j|s,a) h(j) \right\}, \quad \forall s \in S \tag{7.46}$$

or equivalently in vector form as:

[21] See Chapter 8 in Puterman [1994] for a rigorous analysis of the multi-chain Bellman equation.

[22] It does not determine the optimal bias. Instead it determines the gain of an average optimal policy and its bias up to a constant. A further optimality equation is required to determine the optimal bias. Moreover as Example 7.7 shows, there may exist average optimal policies with gain and bias that do not satisfy the Bellman equation.

$$\mathbf{h} = \underset{d \in D^{\mathrm{MD}}}{\text{c-max}} \{\mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d \mathbf{h}\}. \tag{7.47}$$

This equation (in either form), is sometimes referred to as the *average optimality equation.*

Example 7.5. The Bellman equation (7.46) for the unichain model in Section 2.5 becomes

$$h(s_1) = \max\{3 - g + 0.8h(s_1) + 0.2h(s_2), 5 - g + h(s_2)\} \tag{7.48}$$
$$h(s_2) = \max\{-5 - g + h(s_2), 2 - g + 0.4h(s_1) + 0.6h(s_2)\}. \tag{7.49}$$

Observe that there are two equations with three variables so the solution is not unique.

In most instances, solving the Bellman equation requires value iteration, policy iteration or linear programming. This example solves it by using the result (which is established below) that there exists a deterministic stationary policy that is optimal. Hence by enumerating the stationary policies and computing g^{d^∞} for each, it follows that $(d^*)^\infty$ with $d^*(s_1) = a_{1,2}$ and $d^*(s_2) = a_{2,2}$ is optimal and has gain $g^{(d^*)^\infty} = 20/7 = 2.857$.

Solving the evaluation equations (7.44) subject to the condition $\mathbf{P}_{d^*}\mathbf{h} = \mathbf{0}$ gives the solution

$$g^{(d^*)^\infty} = \frac{20}{7}, \quad h^{(d^*)^\infty}(s_1) = \frac{75}{49} \quad \text{and} \quad h^{d^\infty}(s_2) = -\frac{30}{49}.$$

If instead the evaluation equations were solved subject to $h(s_2) = 0$, the solution is $g^{(d^*)^\infty} = 20/7$ together with the relative values $h_{\mathrm{rel}}^{(d^*)^\infty}(s_1) = 15/7$ and $h_{\mathrm{rel}}^{(d^*)^\infty}(s_2) = 0$. Finally, if $h(s_1)$ and $h(s_2)$ together with $g^{(d^*)^\infty}$ solves the evaluation equations, then $g^{(d^*)^\infty}$ and $h(s_1) + k$ and $h(s_2) + k$ for a constant k also satisfies this system of equations.

To verify that these values satisfy the Bellman equation it is easiest to do so for the relative values. Consider equation (7.48)

$$h(s_1) = \max\{3 - g + 0.8h(s_1) + 0.2h(s_2), 5 - g + h(s_2)\}.$$

Since $h_{\mathrm{rel}}^{(d^*)^\infty}(s_1) = 5 - g^{(d^*)^\infty} + h_{\mathrm{rel}}^{(d^*)^\infty}(s_2)$, it is only necessary to check the first expression in the "max". Since $3 - g^{(d^*)^\infty} + 0.8h_{\mathrm{rel}}^{(d^*)^\infty}(s_1) + 0.2h_{\mathrm{rel}}^{(d^*)^\infty}(s_2) = 13/7 < 15/7 = h_{\mathrm{rel}}^{(d^*)^\infty}(s_1)$, equation (7.48) holds.

Now consider the equation (7.49)

$$h(s_2) = \max\{-5 - g + h(s_2), 2 - g + 0.4h(s_1) + 0.6h(s_2)\}.$$

Since $h_{\text{rel}}^{(d^*)^\infty}(s_2) = 2 - g^{(d^*)^\infty} + 0.4h_{\text{rel}}^{(d^*)^\infty}(s_1) + 0.6h_{\text{rel}}^{(d^*)^\infty}(s_2)$, it is necessary only to check the first equation in the "max". Since $-5 - g^{(d^*)^\infty} + h_{\text{rel}}^{(d^*)^\infty}(s_2) = -49/7 < 0 = h_{\text{rel}}^{(d^*)^\infty}(s_2)$, equation (7.49) holds. Hence

$$g^{(d^*)^\infty} = \frac{20}{7}, \quad h^{(d^*)^\infty}(s_1) = \frac{15}{7} \quad \text{and} \quad h^{(d^*)^\infty}(s_2) = 0$$

is a solution to the Bellman equation.

7.5.2 Bounds on the optimal gain

The following result establishes that the first component of the solution to the Bellman equation equals the optimal gain. It does so by first establishing upper bounds on the lim sup average reward and lower bounds on the lim inf average reward of each policy. Its instructive proof provides insight into some fundamental Markov decision process concepts especially how the Bellman equation, which is expressed in terms of Markovian deterministic decision rules, accounts for history-dependent randomized policies.

Theorem 7.7. Let g be a scalar and $\mathbf{h}$ a bounded $|S|$-dimensional vector.

1. If
$$\mathbf{h} \geq \underset{d \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h}\}, \tag{7.50}$$
then $g\mathbf{e} \geq \mathbf{g}_+^\pi$ for all $\pi \in \Pi^{\text{HR}}$.

2. If
$$\mathbf{h} \leq \underset{d \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h}\}, \tag{7.51}$$
then $g\mathbf{e} \leq \mathbf{g}_-^\pi$ for all $\pi \in \Pi^{\text{HR}}$.

3. If
$$\mathbf{h} = \underset{d \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h}\}, \tag{7.52}$$
then $g\mathbf{e} = \mathbf{g}^*$.

Proof. Let $\pi = (d_1, d_2, \ldots)$ denote an arbitrary policy in Π^{MR}. As a consequence of Lemma 2.43 and equation (7.50),

$$g\mathbf{e} \geq \underset{d \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_d + (\mathbf{P}_d - \mathbf{I})\mathbf{h}\} = \underset{d \in D^{\text{MR}}}{\text{c-max}}\{\mathbf{r}_d + (\mathbf{P}_d - \mathbf{I})\mathbf{h}\}.$$

This means that

$$g\mathbf{e} \geq \mathbf{r}_d + (\mathbf{P}_d - \mathbf{I})\mathbf{h} \tag{7.53}$$

for any $d \in D^{\text{MR}}$. Setting $d = d_2$ gives

$$g\mathbf{e} \geq \mathbf{r}_{d_2} + (\mathbf{P}_{d_2} - \mathbf{I})\mathbf{h}.$$

Multiplying this expression by $\mathbf{P}_{d_1}$ and noting that $\mathbf{P}_{d_1}\mathbf{e} = \mathbf{e}$, it follows that

$$g\mathbf{e} \geq \mathbf{P}_{d_1}\mathbf{r}_{d_2} + \mathbf{P}_{d_1}(\mathbf{P}_{d_2} - \mathbf{I})\mathbf{h}.$$

Adding this to (7.53) applied to d_1 implies that

$$\begin{aligned} 2g\mathbf{e} &\geq \mathbf{r}_{d_1} + \mathbf{P}_{d_1}\mathbf{r}_{d_2} + (\mathbf{P}_{d_1}\mathbf{P}_{d_2} - \mathbf{P}_{d_1} + \mathbf{P}_{d_1} - \mathbf{I})\mathbf{h} \\ &= \sum_{n=1}^{2} \mathbf{P}_\pi^{n-1}\mathbf{r}_{d_n} + (\mathbf{P}_\pi^2 - \mathbf{I})\mathbf{h} \end{aligned}$$

Hence by induction it can be shown that for $N \geq 1$

$$Ng\mathbf{e} \geq \sum_{n=1}^{N} \mathbf{P}_\pi^{n-1}\mathbf{r}_{d_n} + (\mathbf{P}_\pi^N - \mathbf{I})\mathbf{h}$$

where $\mathbf{P}_\pi^0 = \mathbf{I}$ and $\mathbf{P}_\pi^n = \mathbf{P}_{d_1}\mathbf{P}_{d_2}\cdots\mathbf{P}_{d_n}$. From this it follows that

$$g\mathbf{e} \geq \limsup_{N\to\infty} \frac{1}{N}\left[\sum_{n=1}^{N} \mathbf{P}_\pi^{n-1}\mathbf{r}_{d_n} + (\mathbf{P}_\pi^N - \mathbf{I})\mathbf{h}\right] = \limsup_{N\to\infty} \frac{1}{N}\mathbf{v}_N^\pi = \mathbf{g}_+^\pi,$$

where the first equality follows from the boundedness of $\mathbf{h}$. Thus, $g\mathbf{e} \geq \mathbf{g}_+^\pi$ for all $\pi \in \Pi^{\text{MR}}$. As a consequence of Theorem 7.12, this is also valid for all $\pi \in \Pi^{\text{HR}}$ establishing part 1.

The proof of part 2 is more direct. From (7.51) there exists a $d^* \in D^{\text{MD}}$ for which

$$g\mathbf{e} \leq \mathbf{r}_{d^*} + (\mathbf{P}_{d^*} - \mathbf{I})\mathbf{h}.$$

Repeating the argument in the proof of part 1 with the single decision rule d^* and accounting for the reversed direction of the inequality implies that

$$g\mathbf{e} \leq \lim_{N\to\infty} \frac{1}{N}\mathbf{v}_N^{(d^*)^\infty} = \liminf_{N\to\infty} \frac{1}{N}\mathbf{v}_N^{(d^*)^\infty} \leq \sup_{\pi\in\Pi^{\text{HR}}} \left\{ \liminf_{N\to\infty} \frac{1}{N}\mathbf{v}_N^{d^\infty} \right\} = \sup_{\pi\in\Pi^{\text{HR}}} \mathbf{g}_-^\pi = \mathbf{g}^*,$$

where the limit in the first inequality exits and equals its lim inf because $(d^*)^\infty$ is stationary. The last equality follows from Definition 7.4.

The third result follows by combining the first two to obtain

$$g\mathbf{e} \leq \mathbf{g}^* = \sup_{\pi\in\Pi^{\text{HR}}} \mathbf{g}_-^\pi \leq \sup_{\pi\in\Pi^{\text{HR}}} \mathbf{g}_+^\pi \leq g\mathbf{e}.$$

□

Some comments on this result follow:

1. Part 3 expresses the most important result, namely that the existence of a solution to the Bellman equation (7.47) identifies the optimal average reward.

2. Parts 1 and 2 hold regardless of chain structure.

3. Note that no assumptions are made on the chain structure. However part 3 may be vacuous when the optimal gain is not constant because there need not exist a scalar g and vector $\mathbf{h}$ that satisfy this specification of the Bellman equation. The following example illustrates this situation.

Example 7.6. Consider the deterministic model shown in Figure 7.6 with $S = \{s_1, s_2\}$, $A_{s_1} = \{a_{1,1}, a_{1,2}\}$ and $A_{s_2} = \{a_{2,1}\}$ with $r(s_1, a_{1,1}) = 3, r(s_1, a_{1,1}) = 1$ and $r(s_2, a_{2,1}) = 2$. Clearly the model is multi-chain and not communicating. Because $g^\pi(s_2) = 2$ and $g^\pi(s_1) \leq 3$ for any policy π, the policy d^∞ with $d(s_1) = a_{1,1}$ and $d(s_2) = a_{2,1}$ is optimal with values $g^*(s_1) = 3$ and $g^*(s_2) = 2$.

Note $g = 3$ and $h(s_1) = h(s_2) = 0$ satisfy (7.50) (expressed in component notation):

$$\begin{aligned} h(s_1) &\geq \max\{3 - g + h(s_1), 1 - g + h(s_2)\} \\ h(s_2) &\geq 2 - g + h(s_2) \end{aligned}$$

Hence $g \geq g_+^\pi(s)$ for $s \in S$ and $\pi \in \Pi^{\text{HR}}$ in agreement with part 1 of Theorem 7.7.

Moreover $g' = 2$ and $h'(s_1) = h'(s_2) = 0$ satisfy (7.51) so that $g' \leq g_-^\pi(s)$ for $s \in S$ and $\pi \in \Pi^{\text{HR}}$ in agreement with part 2. This establishes that

$$2\mathbf{e} \leq \mathbf{g}^* \leq 3\mathbf{e}.$$

Writing out (7.52):

$$\begin{aligned} h(s_1) &= \max\{3 - g + h(s_1), 1 - g + h(s_2)\} \\ h(s_2) &= 2 - g + h(s_2). \end{aligned}$$

The second equation suggests $g = 2$, but then the first equation becomes $h(s_1) = \max\{1 + h(s_1), -1 + h(s_2)\}$, which has no solution. Thus part 3 of the theorem does not apply for this multi-chain model.

If on the other hand $r(s_2, a_{2,1}) = 3$, then part 3 would be valid even though the model is multi-chain.

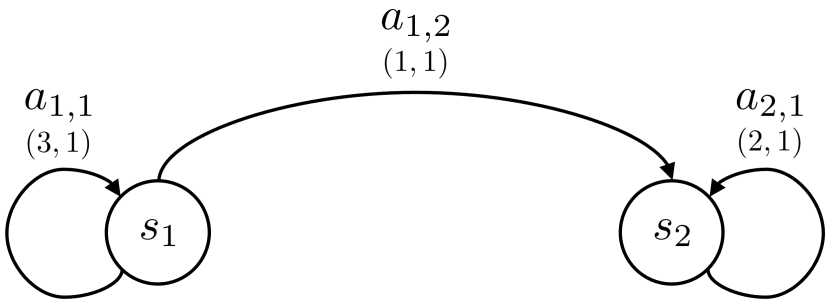


Figure 7.6: Model in Example 7.6. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

7.5.3 Existence of solutions of the Bellman equation

This section establishes the existence of a solution to the unichain optimality equation. The proof formalizes the use of the partial Laurent series of the expected discounted reward to link the gain and bias of a policy to its discounted reward. The fundamental idea is that in a model with finitely many deterministic stationary policies, there exists a sequence of discount factors converging to one for which the *same* policy is discount optimal[23].

Theorem 7.8. In a finite state, finite action recurrent or unichain model there exist:

1. A unique scalar g and a vector $\mathbf{h}$ that satisfy the Bellman equation (7.47)
2. A stationary deterministic optimal policy.

Proof. Choose a sequence of discount factors $\lambda'_n, n = 1, 2, \ldots$ for which $\lambda'_n \uparrow 1$. For each n there exists a stationary deterministic discount optimal policy by Theorem 5.4. Since the set of stationary policies is finite, there exists a subsequence of λ'_n, denoted by $\lambda_n, n = 1, 2, \ldots$ for which a specific policy δ^∞ is optimal. Hence $\mathbf{v}_{\lambda_n}^{\delta^\infty} = \mathbf{v}_{\lambda_n}^*$ for $n \geq 1$.

This means that $\mathbf{v}_{\lambda_n}^{\delta^\infty}$ satisfies the discounted Bellman equation (5.48):

$$\mathbf{0} = \underset{d' \in D^{\mathrm{MD}}}{\text{c-max}}\{\mathbf{r}_{d'} + (\lambda_n \mathbf{P}_{d'} - \mathbf{I})\mathbf{v}\}$$

for all $n \geq 1$. Thus it follows that

$$\mathbf{0} = \mathbf{r}_\delta + (\lambda_n \mathbf{P}_\delta - \mathbf{I})\mathbf{v}_{\lambda_n}^{\delta^\infty} = \underset{d' \in D^{\mathrm{MD}}}{\text{c-max}}\{\mathbf{r}_{d'} + (\lambda_n \mathbf{P}_{d'} - \mathbf{I})\mathbf{v}_{\lambda_n}^{\delta^\infty}\} \geq \mathbf{r}_d + (\lambda_n \mathbf{P}_d - \mathbf{I})\mathbf{v}_{\lambda_n}^{\delta^\infty} \tag{7.54}$$

[23] The stronger concept of *Blackwell optimality* is relevant here. A policy π is said to be *Blackwell optimal* if there exists a λ^* for which π is discount optimal for all $\lambda \in [\lambda^*, 1)$.

for all $d \in D^{\text{MD}}$.

From Theorem 7.4, there exist g^{δ^∞} and $\mathbf{h}^{\delta^\infty}$ for which

$$\mathbf{v}_{\lambda_n}^{\delta^\infty} = \frac{g^{\delta^\infty}\mathbf{e}}{1-\lambda_n} + \mathbf{h}^{\delta^\infty} + \mathbf{f}(\lambda_n), \tag{7.55}$$

where $\mathbf{f}(\lambda_n) \to \mathbf{0}$ as $n \to \infty$.

Now substitute (7.55) into the right hand side of (7.54) to obtain for *any* $d' \in D^{\text{MD}}$ that

$$\begin{aligned}\mathbf{r}_{d'} + (\lambda_n \mathbf{P}_{d'} - \mathbf{I})\mathbf{v}_{\lambda_n}^{\delta^\infty} &= \mathbf{r}_{d'} + (\lambda_n \mathbf{P}_{d'} - \mathbf{I})\left(\frac{g^{\delta^\infty}\mathbf{e}}{1-\lambda_n} + \mathbf{h}^{\delta^\infty} + \mathbf{f}(\lambda_n)\right) \\ &= \mathbf{r}_{d'} - g^{\delta^\infty}\mathbf{e} + (\lambda_n \mathbf{P}_{d'} - \mathbf{I})\mathbf{h}^{\delta^\infty} + \mathbf{f}'(\lambda_n), \end{aligned} \tag{7.56}$$

where $\mathbf{f}'(\lambda_n) \to \mathbf{0}$ as $n \to \infty$. Choose d' equal to δ and d, substitute this expression into (7.54) and let $n \to \infty$ to yield

$$\mathbf{0} = \mathbf{r}_\delta - g^{\delta^\infty}\mathbf{e} + (\mathbf{P}_\delta - \mathbf{I})\mathbf{h}^{\delta^\infty} \geq \mathbf{r}_d - g^{\delta^\infty}\mathbf{e} + (\mathbf{P}_d - \mathbf{I})\mathbf{h}^{\delta^\infty}$$

for all $d \in D^{\text{MD}}$. From this it follows that:

$$\mathbf{0} = \underset{d' \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_{d'} - g^{\delta^\infty}\mathbf{e} + (\mathbf{P}_{d'} - \mathbf{I})\mathbf{h}^{\delta^\infty}\}. \tag{7.57}$$

Hence part 1 follows. Moreover since the pair $(g^{\delta^\infty}, \mathbf{h}^{\delta^\infty})$ satisfy the optimality equation, part 3 of Theorem 7.7 implies that $g^{\delta^\infty} = g^*$, so δ^∞ is average optimal. □

7.5.4 The Bellman equation and optimal policies

The following result shows how to use solutions of the Bellman equation to identify optimal policies.

Theorem 7.9. Suppose $(g^*, \mathbf{h}^*)$ are solutions of the Bellman equation and for all $s \in S$

$$d^*(s) \in \underset{a \in A_s}{\arg\max}\left\{r(s,a) + \sum_{j \in S} p(j|s,a)h^*(j)\right\} \tag{7.58}$$

or equivalently

$$d^* \in \underset{d \in D^{\text{MD}}}{\arg\,\text{c-max}}\left\{\mathbf{r}_d + \mathbf{P}_d\mathbf{h}^*\right\}. \tag{7.59}$$

Then $(d^*)^\infty$ is average optimal.

Proof. From (7.59)

$$\mathbf{h}^* = \max_{d \in D^{\text{MD}}}\{\mathbf{r}_d - g^*\mathbf{e} + \mathbf{P}_d\mathbf{h}^*\} = \mathbf{r}_{d^*} - g^*\mathbf{e} + \mathbf{P}_{d^*}\mathbf{h}^*.$$

As a consequence of Corollary 7.4, $g^* = g^{(d^*)^\infty}$ establishing the result. □

Some comments about this result follow:

1. In a finite action model, there always exists such a d^*. This result holds in greater generality but requires some technical considerations.

2. Surprisingly, neither (7.58) nor (7.59) includes g^* directly. However, they do so indirectly since the pair $(g^*, \mathbf{h}^*)$ satisfies the Bellman equation. Moreover, h^* is defined using g^*.

3. Some authors refer to any decision rule d that satisfies

$$d \in \arg\max_{d \in D^{\text{MD}}} \{\mathbf{r}_d + \mathbf{P}_d \mathbf{h}\}$$

   as *greedy* with respect to $\mathbf{h}$.

4. As (7.58) and (7.59) suggest, d^* need not be unique.

Does the gain and bias of every average optimal policy satisfy the Bellman equation?

Theorem 7.9 does *not* imply that the gain and bias of *every* average optimal policy satisfies the Bellman equation. In particular, if d^* is average optimal, then it must satisfy (7.58) for all recurrent states but need not satisfy the optimality equation for transient states. This is because the reward accumulated in transient states, which in expectation are only visited a finite number of times, does not affect the overall average reward of a policy. Thus, average optimal decision rules do not discriminate on the basis of behavior in transient states.

The following example sheds some light on the relationship between solutions of the average optimality equation and average optimal policies. It shows that there exist average optimal policies with gain and bias that do not satisfy the Bellman equation. In other words, the Bellman equation is sufficient, but not necessary, for optimality, in contrast to previous chapters.

Moreover the following example delves into why the gain and bias of certain average optimal policies satisfy the Bellman equation while others do not.

Example 7.7. Let $S = \{s_1, s_2\}$; $A_{s_1} = \{a_{1,1}, a_{1,2}\}$, $A_{s_2} = \{a_{2,1}\}$; $p(s_1|s_1, a_{1,1}) = p(s_2|s_1, a_{1,1}) = 0.5$, $p(s_2|s_1, a_{1,2}) = 1$, $p(s_2|s_2, a_{2,1}) = 1$; and $r(s_1, a_{1,1}) = 5$, $r(s_1, a_{1,2}) = 10$ and $r(s_2, a_{2,1}) = 1$. Rewards need not be defined for zero probability transitions. See Figure 7.7[a].

Let δ denote the decision rule that uses action $a_{1,1}$ in s_1 and γ denote the decision rule which uses action $a_{1,2}$ in s_1. There is no action choice in s_2. This model is unichain with s_2 absorbing and s_1 transient under both stationary

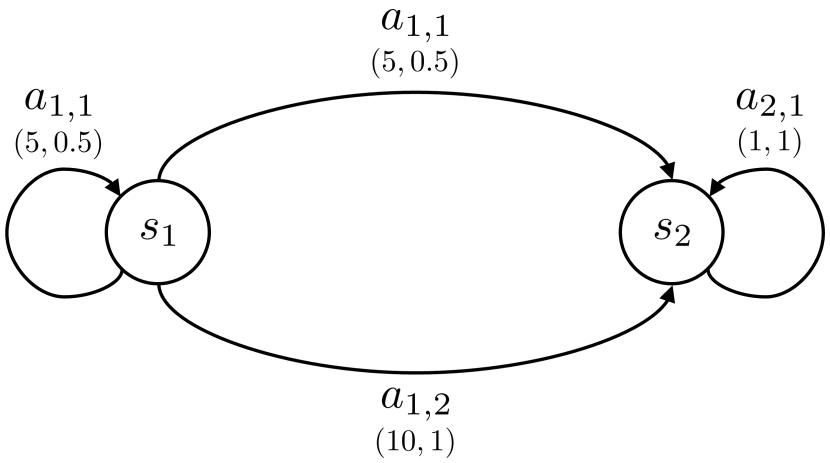


Figure 7.7: Graphical representation of model in Example 7.7. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

policies. Clearly

$$g^{\delta^\infty} = g^{\gamma^\infty} = g^* = 1$$

so that *both* deterministic stationary policies are average optimal.

To find the gain and bias of each stationary policy solve

$$\mathbf{h} = \mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h}$$

subject to $\mathbf{P}_d^*\mathbf{h} = \mathbf{0}$. Since

$$\mathbf{P}_\delta^* = \mathbf{P}_\gamma^* = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$

$h^{\delta^\infty}(s_2) = h^{\gamma^\infty}(s_2) = 0$ so that it follows that $h^{\delta^\infty}(s_1) = 8$ and $h^{\gamma^\infty}(s_1) = 9$. From the Bellman equation,

$$\begin{aligned} h(s_1) &= \max\{5 - g + 0.5h(s_1) + 0.5h(s_2), 10 - g + h(s_2)\} \\ h(s_2) &= 1 - g + h(s_2) \end{aligned}$$

it follows that $(g^{\gamma^\infty}, \mathbf{h}^{\gamma^\infty})$ satisfies the Bellman equation but $(g^{\delta^\infty}, \mathbf{h}^{\delta^\infty})$ does not.

[a] A variant of this example is analyzed throughout Puterman [1994], most notably in slightly modified form as Example 8.4.3.

In this example, both policies maximize the gain. Moreover, starting in state s_1, each accumulates an expected total reward of 10 prior to reaching s_2. But one would believe that a decision maker would prefer the policy γ^∞ to δ^∞ because the reward of

10 would be received sooner and with certainty.

This anomaly can be understood in two ways:

1. The proof of Theorem 7.8 constructs a solution to the Bellman equation using the partial Laurent series expansion of a policy that is discount optimal for a sequence of λ converging to one. Thus the solution of the Bellman equation corresponds to the gain and bias (up to a constant) of a discount optimal policy for λ near 1. It is easy to prove[24] that γ^∞ is discount optimal for λ near 1, while δ^∞ is not.

2. Computing the bias of each policy in s_1 is constructive. From the definition of bias in (7.17);

$$h^{\gamma^\infty}(s_1) = r(s_1, a_{1,2}) - g^{\gamma^\infty} = 10 - 1 = 9$$
$$h^{\delta^\infty}(s_1) = (5-1) + 0.5(5-1) + (0.5)^2(5-1) + \ldots = 8.$$

   Hence the bias of γ^∞ is greater than that of δ^∞.

As a consequence of the second observation above, one might conjecture that the solution of the Bellman equation also maximizes the bias over all average optimal policies. This is not true in general[25].

Note that the above example might seem a bit contrived, however the same issue arises in many practical problems such as queuing admission control[26] and (s, S)-inventory models. In such models good policies drive the system to a set of recurrent states where it remains in perpetuity so that actions chosen on transient states do not affect the average reward.

7.5.5 State-action value functions

As in the cases of previously considered optimality criteria, in average reward models the policy evaluation equation and Bellman equation can be expressed in terms of state-action value functions. This representation will not be central to this chapter, but will be fundamental in Chapter 10 which studies simulation methods for average reward models. Discussion will be restricted to recurrent/unichain models and be brief.

Let g^* denote the optimal average reward and π^* denote an optimal policy. Define the *optimal state-action value function* as follows.

[24] Apply policy iteration starting from policy γ^∞.

[25] See Chapter 10 of Puterman [1994] for a detailed discussion of this issue.

[26] See Haviv and Puterman [1998] for an analysis of this issue in a continuous time queuing model.

Definition 7.9. The *optimal state-action value function*, $q^*(s,a)$, for all $a \in A_s$ and $s \in S$, is defined as

$$q^*(s,a) := E^{\pi^*}\left[\sum_{n=1}^{\infty}\left(r(X_n,Y_n)-g^*\right)\middle| X_1=s, Y_1=a\right]. \tag{7.60}$$

Separating out the first term it follows that

$$\begin{aligned} q^*(s,a) &= r(s,a)-g^*+E^{\pi}\left[\sum_{n=2}^{\infty}\left(r(X_n,Y_n)-g^*\right)\middle| X_1=s, Y_1=a\right] \\ &= r(s,a)-g^*+\sum_{j\in S}p(j|s,a)h^*(j) \end{aligned} \tag{7.61}$$

where the scalar g^* and $h^*(s)$ for all $s \in S$ solve the recurrent/unichain optimality equation (7.46). Since $h^*(s)$ is unique up to a scalar factor, the state-action value function depends on its specification. The most common choices are the bias or a relative value.

For a stationary policy[27] d^∞, the *policy-specific state-action value function* is defined by

$$q^{d^\infty}(s,a) := r(s,a)-g^{d^\infty}+\sum_{j\in S}p(j|s,a)h^{d^\infty}(j) \tag{7.62}$$

for all $a \in A_s$ and $s \in S$ where g^{d^∞} and $h^{d^\infty}(s)$ solve the policy evaluation equations defined in Theorem 7.6. Again note that $h^{d^\infty}(s)$ is unique up to a constant.

Bellman equation for state-action value functions

As noted above, g^* and $h^*(s)$ are solutions of the Bellman equation in recurrent and unichain models. Hence

$$h^*(s)=\max_{a\in A_s}\left\{r(s,a)-g^*+\sum_{j\in S}p(j|s,a)h^*(j)\right\}.$$

Setting

$$q^*(s,a)=r(s,a)-g^*+\sum_{j\in S}p(j|s,a)h^*(j) \tag{7.63}$$

it follows from above that

$$h^*(s)=\max_{a\in A_s}q^*(s,a).$$

[27] Note the state-action value functions can be defined for arbitrary policies but this level of generality will not be needed.

Hence substituting this representation into (7.63) establishes that the Bellman equation expressed in terms of state-action value functions is equivalent to finding a scalar g^* and a function $q^*(s,a)$ defined for $s \in S$ and $a \in A_s$ that satisfy:

Average reward Bellman equation for state-action value functions:

$$q(s,a) = r(s,a) - g + \sum_{j \in S} p(j|s,a) \max_{a' \in A_s} q(j,a'). \tag{7.64}$$

When g^* and $q^*(s,a)$ are solutions of (7.64), choosing $d^*(s) \in \arg\max_{a' \in A_s} q^*(s,a')$ for all $\in S$ identifies an optimal stationary policy.

Note that:

1. Equation (7.64) involves action choice at *both* the current decision epoch and the subsequent decision epoch.

2. The solution of (7.64) need not be unique. One way of resolving this is to add the extra condition that $q(s^*,a^*) = 0$ for some designated state-action pair (s^*,a^*). This approach is generalized when discussing relative Q-learning in Chapter 10.

3. To evaluate the deterministic stationary policy d^∞, the analogous evaluation equation would be

$$q(s,a) = r(s,a) - g + \sum_{j \in S} p(j|s,a) q(j,d(s)). \tag{7.65}$$

Written as an expected value, (7.64) is equivalent to

$$q(s,a) = E^{\pi^*}\left[r(X,Y) - g + \max_{a \in A_{X'}} q(X',a) \,\middle|\, X = s, Y = a \right], \tag{7.66}$$

where π^* is an optimal policy and X' is a random variable distributed according to $p(\cdot|s,a)$. The previous equation extends to models in which $r(s,a,j)$ is the model primitive as follows:

$$q(s,a) = E^{\pi^*}\left[r(X,Y,X') - g + \max_{a \in A_{X'}} q(X',a) \,\middle|\, X = s, Y = a \right]. \tag{7.67}$$

Similar to state-action value functions in discounted Markov decisions processes, these expressions differ from the usual Bellman equation in that the maximization is *inside* the expectation. This will be important in Chapter 10, where expectations are estimated by simulated or realized rewards and transitions.

7.5.6 The Bellman equation in multi-chain models*

For completeness, this brief section[28] provides the Bellman equation for multi-chain models.

In component notation there are two equations, given by

$$g(s) = \max_{a \in A_s} \left\{ \sum_{j \in S} p(j|s,a) g(j) \right\} \tag{7.68}$$

$$h(s) = \max_{a \in G_s} \left\{ r(s,a) - g(s) + \sum_{j \in S} p(j|s,a) h(j) \right\} \tag{7.69}$$

where

$$G_s := \left\{ a' \in A_s \;\middle|\; g(s) = \sum_{j \in S} p(j|s,a') g(j) \right\}$$

for all $s \in S$.

Note that:

1. This pair of equations is said to be *nested* because the maximization in the second equation is over actions that achieve the maximum in the first equation.

2. The equations uniquely determine $g(s)$ and determine $h(s)$ up to one or more constants. The number of constants depends on the chain structure of optimal policies.

3. In a recurrent/unichain model, the gain of every policy is constant so that the first equation only guarantees that $g(s)$ is constant, in which case these reduce to recurrent/unichain Bellman equation.

4. The multi-chain equations are required in communicating or weakly communicating models because in such models, some policies might have non-constant gain.

5. The most direct way to solve these equations is by policy iteration. The proof of convergence is quite involved.

7.6 Value iteration

Discussion now shifts to algorithms for solving average reward models. Average reward value iteration is based on the following recursion expressed in component notation as

[28] A comprehensive discussion of multi-chain average reward models appears in Chapter 9 of Puterman [1994].

$$v'(s) \leftarrow \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) v(j) \right\} \tag{7.70}$$

or equivalently in vector form as:

$$\mathbf{v}' \leftarrow L\mathbf{v} := \underset{d \in D^{\mathrm{MD}}}{\text{c-max}} \{ \mathbf{r}_d + \mathbf{P}_d \mathbf{v} \}. \tag{7.71}$$

The astute reader will note that these recursions have a similar form to value iteration in expected total reward models. In that case, a convergent algorithm was constructed by restricting attention to positive, negative, transient and stochastic shortest path models. A relevant explicit or implicit feature of these cases was that in all them, the system reached zero-reward absorbing states so that the optimal average reward equaled zero. In this chapter, models have non-zero average rewards so different analyses are required.

Consequently, several challenges emerge to create a convergent algorithm from these recursions.

1. The average reward g does not appear in the above expressions; a formal algorithm requires a method to approximate the optimal gain from a sequence of iterates.

2. The iterates of an algorithm based on these recursions may diverge; a modifications is required to obtain a convergent sequence.

3. The iterates of an algorithm based on these recursions may oscillate. Models that have this property must be characterized so that approaches to address this difficulty can be developed.

4. Span-based stopping criteria may not result in algorithm termination in multichain models.

7.6.1 Examples

The following two examples illustrate the challenges of using value iteration in average reward models.

A straightforward example

The first example applies value iteration to the two-state model to illustrate that values may diverge.

Example 7.8. Consider the two-state example from Section 2.5. Begin the iterative scheme implicit in (7.70) or (7.71) at $\mathbf{v}^0 = \mathbf{0}$ to obtain the sequence of iterates $\mathbf{v}^n$ reported in Table 7.5.

Observe that the iterates diverge but that $\mathrm{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n)$ converges to zero exponentially quickly. Observe further that for $s \in S$, $v^{10}(s) - v^9(s) = 2.857 = g^*$ in agreement with Example 7.5. See (7.34) for an indication why this might be the case. This will be made precise below.

n	$v^n(s_1)$	$v^n(s_2)$	$\mathrm{sp}(\mathbf{v}^n - \mathbf{v}^{n-1})$
0	0.00000	0.00000	n/a
1	5.00000	2.00000	3.0000000
2	7.40000	5.20000	0.8000000
3	10.20000	8.08000	0.0800000
4	13.08000	10.92800	0.0320000
5	15.92800	13.78880	0.0128000
6	18.78880	16.64448	0.0051200
7	21.64448	19.50221	0.0020480
8	24.50221	22.35912	0.0008192
9	27.35912	25.21635	0.0003277
10	30.21635	28.07346	0.0001311

Table 7.5: Iterates of value iteration obtained using the methods in Example 7.8.

A periodic example

The following example provides an "edge-case" in which the iterates of value iteration oscillate.

Example 7.9. Consider a two-state periodic model with $S = \{s_1, s_2\}$ and a single decision rule $d \in D^{\mathrm{MD}}$ with reward and transition probabilities given by:

$$\mathbf{r}_d = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{P}_d = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Set $\mathbf{v}^0 = \begin{bmatrix} a \\ b \end{bmatrix}$. Since $\mathbf{v}^n = \mathbf{P}_d^n \mathbf{v}^0$,

$$\mathbf{v}^n = \begin{bmatrix} a \\ b \end{bmatrix} \quad \text{for } n \text{ even and} \quad \mathbf{v}^n = \begin{bmatrix} b \\ a \end{bmatrix} \quad \text{for } n \text{ odd.}$$

Hence, unless $a = b$, the iterates oscillate. Moreover, $\mathrm{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) = 2|a - b|$.

Since

$$\mathbf{P}_d^* = \begin{bmatrix} 1/2 & 1/2 \\ \\ 1/2 & 1/2 \end{bmatrix},$$

$g^{d^\infty} = 0$.

These observations suggest that in a periodic model value iteration oscillates and need not converge with respect to the span semi-norm.

7.6.2 Removing periodicity

Periodicity is the most significant challenge to developing a convergent value iteration algorithm. In practice most applications will be aperiodic. However, a model with periodic policies can be transformed into an aperiodic model by a simple transformation that preserves values (up to constant) and optimality by a simple perturbation.

To do so, for $0 < \tau < 1$ define a transformed model in which rewards and transition probabilities are denoted by a tilde as follows:

$$\tilde{r}(s,a) = \tau r(s,a) \quad \text{and} \quad \tilde{p}(j|s,a) = (1-\tau)\delta(j|s) + \tau p(j|s,a) \tag{7.72}$$

for $s \in S$, $a \in A_s$ and $j \in S$ where $\delta(j|s) = 1$ if $j = s$ and 0 otherwise.

Alternatively in vector notation this transformation can be expressed for $d \in D^{\text{MD}}$ by:

$$\tilde{\mathbf{r}}_d = \tau \mathbf{r}_d \quad \text{and} \quad \tilde{\mathbf{P}}_d = (1-\tau)\mathbf{I} + \tau \mathbf{P}_d. \tag{7.73}$$

This transformation adds a self-transition to the model so that no Markovian decision rule has a periodic transition matrix. Another way to think about this transformation is to regard it as the result of inspecting the state more frequently in a model in which transitions occur uniformly throughout the period between two decision epochs. For example if $\tau = 0.5$, the system state is viewed twice as often. Consequently at such an inspection time, a transition may have occurred with probability τ and the system remains in the same state with probability $1 - \tau$.

Example 7.10. Apply the transformation to the model in Example 7.9 above to obtain:

$$\tilde{\mathbf{r}}_d = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \tilde{\mathbf{P}}_d = \begin{bmatrix} 1-\tau & \tau \\ \tau & 1-\tau \end{bmatrix}.$$

The transformed model has period 1 so that it is aperiodic.

Since the iterates of value iteration satisfy $\mathbf{v}^n = \tilde{\mathbf{P}}_d^n \mathbf{v}^0$, an eigenvalue decomposition of $\tilde{\mathbf{P}}_d$ simplifies calculations. Applying the eigenvalue decomposition Theorem B.2 in Appendix B results in

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & 0 \\ 0 & 1-2\tau \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \text{and} \quad \mathbf{W}^{-1} = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix}.$$

So,

$$\tilde{\mathbf{P}}_d^n = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^{-1} = \frac{1}{2}\begin{bmatrix} 1+(1-2\tau)^n & 1-(1-2\tau)^n \\ 1-(1-2\tau)^n & 1+(1-2\tau)^n \end{bmatrix}.$$

Consequently for $\mathbf{v}^0 = \begin{bmatrix} a \\ b \end{bmatrix}$

$$\mathbf{v}^{n+1} - \mathbf{v}^n = \tilde{\mathbf{P}}_d^n(\tilde{\mathbf{P}}_d - \mathbf{I})\mathbf{v}_0 = \tau\tilde{\mathbf{P}}_d^n \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{v}^0 = \tau(1-2\tau)^n \begin{bmatrix} b-a \\ a-b \end{bmatrix}.$$

Therefore $\text{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) = 2\tau(1-2\tau)^n|b-a|$ converges to 0 as $n \to \infty$ so that in this modified model, value iteration is convergent with respect to the span semi-norm.

The following result shows that this transformation preserves solutions of the optimality equation and consequently optimal policies.

Proposition 7.2. Suppose that the pair $(g^*, \mathbf{h}^*)$ satisfies (7.47). Then

1. The pair $(\tau g^*, \mathbf{h}^*)$ satisfies

$$\mathbf{h} = \underset{d \in D^{\text{MD}}}{\text{c-max}} \{\tilde{\mathbf{r}}_d - g\mathbf{e} + \tilde{\mathbf{P}}_d \mathbf{h}\}. \tag{7.74}$$

2. The same policy is optimal for both models.

Proof. Rewriting (7.47),

$$\mathbf{h}^* + g^*\mathbf{e} = \underset{d \in D^{\text{MD}}}{\text{c-max}} \{\mathbf{r}_d + \mathbf{P}_d \mathbf{h}^*\}.$$

Multiply this equation by τ and add $(1-\tau)\mathbf{h}^*$ to both sides to obtain

$$\mathbf{h}^* + \tau g^*\mathbf{e} = \underset{d \in D^{\text{MD}}}{\text{c-max}} \{\tau\mathbf{r}_d + (1-\tau)\mathbf{I}\mathbf{h}^* + \tau\mathbf{P}_d\mathbf{h}^*\} = \underset{d \in D^{\text{MD}}}{\text{c-max}} \{\tilde{\mathbf{r}}_d + \tilde{\mathbf{P}}_d\mathbf{h}^*\},$$

which establishes the result. □

Observe that the above result holds regardless how $\mathbf{h}^*$ is specified. Moreover, this result is consistent with interpreting the aperiodic transformation as increased model inspection. When the state is inspected twice as often, that is $\tau = 0.5$, the per period reward is half as great, that is $0.5g^*$

Replacing D^{MD} by a single policy establishes:

Corollary 7.5. Suppose for some $d \in D^{\text{MD}}$ that the pair $(g^{d^\infty}, \mathbf{h}^{d^\infty})$ satisfy (7.43), then the pair $(\tau g^{d^\infty}, \mathbf{h}^{d^\infty})$ satisfy

$$\mathbf{h} = \tilde{\mathbf{r}}_d - g\mathbf{e} + \tilde{\mathbf{P}}_d\mathbf{h}.$$

Since this transformation applies to any model, the remainder of this section assumes that *all policies are aperiodic*. On occasion, this will be restated for emphasis.

7.6.3 A value iteration algorithm

This section states and investigates the properties of the following value iteration algorithm expressed in component form:

Algorithm 7.1. Value iteration for an average reward model: component form

1. **Initialize:** Specify $v'(s)$ for all $s \in S$, $\epsilon > 0$ and $\sigma > \epsilon$.
2. **Iterate:** While $\sigma \geq \epsilon$:
   (a) $v(s) \leftarrow v'(s)$ for all $s \in S$.
   (b) For all $s \in S$,
   $$v'(s) \leftarrow \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a)v(j) \right\}. \tag{7.75}$$
   (c) $\sigma \leftarrow \text{sp}(\mathbf{v}' - \mathbf{v})$.
3. **Terminate:** For all $s \in S$, return
   $$d_\epsilon(s) \in \arg\max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a)v(j) \right\}$$
   and
   $$g^\epsilon = \frac{1}{2}\left(\max_{s \in S}\{v'(s) - v(s)\} + \min_{s \in S}\{v'(s) - v(s)\} \right). \tag{7.76}$$

Note that the recursion (7.75) is identical to that in the expected total reward model, however the above algorithm uses a span-based stopping criterion and returns an estimate of the average reward.

Convergence of value iteration in unichain and recurrent models*

In the absence of contraction properties that were applicable in discounted models and in transient expected total reward models, a proof that value iteration converges and eventually achieves the span-based stopping criterion requires subtle analysis. For ease of exposition, this convergence is established under the assumption that the transition probabilities of every stationary policy are aperiodic and *recurrent or unichain*. Almost all of the results hold under the weaker condition of communicating or weakly communicating models, which ensure that the optimal gain is constant[29].

A key technical result

A proof of the convergence of value iteration in an average reward model is based on the following technical result.

Proposition 7.3. Assume an aperiodic recurrent or unichain model and let $v^n(s)$ denote a sequence of iterates generated by the average reward value iteration algorithm. Then

$$\lim_{n\to\infty} \left(v^n(s) - ng^* - h^*(s)\right) \tag{7.77}$$

exists where g^* and $h^*(s)$ are solutions of the Bellman equation (7.46).

A formal proof is omitted but follows from the steps outlined below.

1. Establish upper and lower bounds on $e_n(s) := v^n(s) - ng^* - h^*(s)$ for each $s \in S$.
2. Use the well-known result that every bounded infinite sequence contains a convergent subsequence[30].
3. Show that $\lim_{n\to\infty} e_n(s)$ exists for s in the recurrent states of an optimal policy by proving that the $\liminf_{n\to\infty} e_n(s) = \limsup_{n\to\infty} e_n(s)$.
4. Show that the above limit exists on transient states by combining a modified argument to that on recurrent states and then appealing to the result on recurrent states.

[29] This development is adapted from the analysis of multi-chain value iteration in Section 9.4 of Puterman [1994].

[30] Known as the Bolzano-Weierstrass Theorem.

The hypothesis of this proposition can be weakened to "there exists an optimal policy with an aperiodic transition probability matrix". Alternatively, by applying the aperiodicity transformation (7.72) prior to analysis, this applies to all recurrent and unichain models. Note that this result is valid in multi-chain models as well but requires a subtle intermediate step[31].

Consequences

When the limit in (7.77) exists, the following important results follows easily.

Proposition 7.4. Suppose in an aperiodic recurrent or unichain model that the sequence $v^n(s)$ is generated by (7.75) with $v^0(s)$ arbitrary and that the limit in (7.77) exists.

Then for all $s \in S$,

$$\lim_{n\to\infty} (v^{n+1}(s) - v^n(s)) = g^* \tag{7.78}$$

and

$$\lim_{n\to\infty} \operatorname{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) = 0. \tag{7.79}$$

Proof. To prove the first result choose $s' \in S$ and denote the limit in (7.77) by K. Then given $\epsilon > 0$ there exists an N such that for $n \geq N$

$$|v^n(s') - ng^* - h^*(s') - K| < \epsilon.$$

Hence

$$\begin{aligned}
&|v^{n+1}(s') - v^n(s') - g^*| \\
&\quad = |(v^{n+1}(s') - (n+1)g^* - h^*(s') - K) - (v^n(s') - ng^* - h^*(s') - K)| \\
&\quad \leq |v^{n+1}(s') - (n+1)g^* - h^*(s') - K| + |v^n(s') - ng^* - h^*(s') - K| \\
&\quad < 2\epsilon.
\end{aligned}$$

Since s' and ϵ were arbitrary, the result follows.

To prove the second result, let

$$s^+ = \arg\max_{s\in S}\{v^{n+1}(s) - v^n(s)\} \text{ and } s^- = \arg\min_{s\in S}\{v^{n+1}(s) - v^n(s)\}.$$

Since g^* is constant, it follows from the first result that

$$\operatorname{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) = \left(v^{n+1}(s^+) - v^n(s^+)\right) - \left(v^{n+1}(s^-) - v^n(s^-)\right) \tag{7.80}$$

$$= \left(v^{n+1}(s^+) - v^n(s^+) - g^*\right) - \left(v^{n+1}(s^-) - v^n(s^-) - g^*\right) \tag{7.81}$$

[31] Details appear in Section 9.4.1 in Puterman [1994].

$$\begin{aligned} &< |v^{n+1}(s^+) - v^n(s^+) - g^*| + |v^{n+1}(s^-) - v^n(s^-) - g^*| \qquad (7.82)\\ &< 4\epsilon \end{aligned}$$

Hence the second result follows. Note that the assumption of constant g^* was crucial to expression (7.81). □

The following result provides bounds on the optimal gain. It can be used to identify an ϵ-optimal policy when using value iteration and also estimate the optimal gain at termination. Moreover it may be useful for estimating the optimal gain when using simulation methods to solve average reward problems.

Proposition 7.5. In an aperiodic recurrent or unichain model, for any $v(s)$,

$$\min_{s'\in S}\{L\mathbf{v}(s') - v(s')\} \leq g^{d^\infty} \leq g^* \leq \max_{s'\in S}\{L\mathbf{v}(s') - v(s')\}, \qquad (7.83)$$

where

$$d(s) \in \arg\max_{a\in A_s}\left\{r(s,a) + \sum_{j\in S} p(j|s,a)v(j)\right\}. \qquad (7.84)$$

Proof. First establish the lower bound. Suppose d satisfies (7.84) and let R_d denote its set of recurrent states. Then since $\mathbf{P}_d^* = \mathbf{P}_d^*\mathbf{P}_d$

$$\begin{aligned} g^{d^\infty}\mathbf{e} = \mathbf{P}_d^*\mathbf{r}_d = \mathbf{P}_d^*(\mathbf{r}_d + \mathbf{P}_d\mathbf{v} - \mathbf{v}) &= \mathbf{P}_d^*(L\mathbf{v} - \mathbf{v})\\ &\geq \min_{s\in R_d}\{L\mathbf{v}(s) - v(s)\}\mathbf{e} \geq \min_{s\in S}\{L\mathbf{v}(s) - v(s)\}\mathbf{e}. \end{aligned}$$

Note that the minimum in the first inequality above is over R_d because columns corresponding to transient states of $\mathbf{P}_d$ will have zeroes in columns of $\mathbf{P}_d^*$ corresponding to transient states.

To establish the upper bound, from Theorem 7.8, there exists a g^* and a decision rule d^* for which

$$\begin{aligned} g^*\mathbf{e} = \mathbf{P}_{d^*}^*\mathbf{r}_{d^*} = \mathbf{P}_{d^*}^*(\mathbf{r}_{d^*} + \mathbf{P}_{d^*}\mathbf{v} - \mathbf{v}) &= \mathbf{P}_{d^*}^*(L\mathbf{v} - \mathbf{v})\\ &\leq \max_{s\in R_{d^*}}\{L\mathbf{v}(s) - v(s)\}\mathbf{e} \leq \max_{s\in S}\{L\mathbf{v}(s) - v(s)\}\mathbf{e}. \end{aligned}$$

□

Combining the above propositions gives the main result of this section. The second and third parts follow immediately from Proposition 7.5.

Theorem 7.10. Let $v^n(s)$ denote the sequence of iterates of value iteration in an aperiodic recurrent or unichain model. Then:

1. Value iteration converges and the stopping criterion in step 2 of the algorithm is satisfied for n sufficiently large.

2. For every $n > 0$,

$$\min_{s\in S}\left\{v^{n+1}(s) - v^n(s)\right\} \leq g^{d^\infty} \leq g^* \leq \max_{s\in S}\left\{v^{n+1}(s) - v^n(s)\right\} \tag{7.85}$$

   where

$$d(s) \in \arg\max_{a\in A_s}\left\{r(s,a) + \sum_{j\in S} p(j|s,a)v^n(j)\right\}. \tag{7.86}$$

3. If $\text{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) < \epsilon$ for some $\epsilon > 0$ then

$$\left|\max_{s\in S}\left\{v^{n+1}(s) - v^n(s)\right\} + \min_{s\in S}\left\{v^{n+1}(s) - v^n(s)\right\} - 2g\right| < \epsilon \tag{7.87}$$

   for $g = g^*$ or $g = g^{d^\infty}$ where d satisfies (7.86).

Some comments about the above results follow:

1. Part 1 of Theorem 7.10 gives the most significant result. Namely in a model in which $g^*(s)$ is constant, the value iteration algorithm converges and the span-based stopping criterion is achieved. Aperiodicity is crucial for this result. Without it, the iterates could oscillate and not satisfy the stopping criterion.

2. In a model in which optimal policies are multi-chain, the algorithm will converge, but since the optimal average reward need not be constant, the span need not converge to zero and hence does not provide a valid stopping rule.

3. As noted above, the assumption that the model be aperiodic is not limiting. It can always be achieved by applying the aperiodicity transformation described in the preceding section.

4. The proof of part 2 of Proposition 7.4 holds under the weaker assumption that the optimal gain is constant. This is true when either:

   (a) The model is regular or unichain.

   (b) The model is communicating.

   (c) The model is weakly communicating.

   Thus this result holds in considerable generality.

5. Part 1 of Proposition 7.4 justifies the approximation

$$g^*(s) \approx v^{n+1}(s) - v^n(s).$$

Part 3 of Theorem 7.10 makes it precise and justifies the calculation in the step "Approximate the optimal average reward" in the value iteration algorithm statements.

6. Part 2 of Theorem 7.10 shows how to obtain an ϵ-optimal policy d^∞ when the condition

$$\text{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n) < \epsilon$$

holds. It provides the basis for the estimate in (7.76).

Referring back to Example 7.8, Table 7.5 shows that $\text{sp}(\mathbf{v}^{n+1} - \mathbf{v}^n)$ converges to zero. At termination, (7.76) gives the estimate $g^* = 2.857$.

7.6.4 Relative value iteration

Observe that in Example 7.8 the iterates of value iteration diverge. While this does not impact theory, it might become problematic in practice, especially in large models. To overcome this and as well provide a basis for a simulation approach, *relative value iteration* can be used instead. Relative value iteration distinguishes a state $\bar{s}$ and normalizes by subtracting $v(\bar{s})$ from $v(s)$ after each iteration. A formal statement of an algorithm follows.

Algorithm 7.2. Relative value iteration: component form

1. **Initialize:** Specify a state $\bar{s}$, $\epsilon > 0$, $\sigma > \epsilon$ and $w'(s)$ satisfying $w'(\bar{s}) = 0$.
2. **Iterate:** While $\sigma \geq \epsilon$:
   (a) For all $s \in S$, $w(s) \leftarrow w'(s)$.
   (b) For all $s \in S$

$$v(s) \leftarrow \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) w(j) \right\}. \tag{7.88}$$

   (c) **Normalize:** For all $s \in S$,

$$w'(s) \leftarrow v(s) - v(\bar{s}).$$

   (d) $\sigma \leftarrow \text{sp}(\mathbf{w}' - \mathbf{w})$.

3. **Terminate:** For all $s \in S$, return

$$d_\epsilon(s) \in \arg\max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) w(j) \right\}$$

and

$$g^\epsilon = \frac{1}{2} \left(\max_{s \in S} \{v(s) - w(s)\} + \min_{s \in S} \{v(s) - w(s)\} \right).$$

To illustrate this algorithm, apply relative value iteration to the model in Example 7.8 normalized by setting $w(s_2) = 0$. Comparing Table 7.6 to Table 7.5 shows that the span expressed in terms of $\mathbf{w}^n$ agrees with that expressed in terms of $\mathbf{v}^n$. (Why?) Moreover the estimates of g^* converge quickly. Note also that $w^n(s_1)$ converges to $h^*_{\text{rel}}(s_1) = 15/7$ (see Example 7.5). Thus with obvious modifications, Theorem 7.10 applies to relative value iteration and moreover, $w^n(s)$ converges to $h^*_{\text{rel}}(s)$.

n	$w^n(s_1)$	$w^n(s_2)$	$\text{sp}(\mathbf{w}^n - \mathbf{w}^{n-1})$	Estimate of g^*
0	0.00000	0	n/a	n/a
1	3.00000	0	3.0	3.5
2	2.20000	0	0.8	2.8
3	2.12000	0	0.08	2.84
4	2.15200	0	0.032	2.864
5	2.13900	0	0.0128	2.8544
6	2.14432	0	0.00512	2.85824
7	2.14227	0	0.002048	2.85670
8	2.14309	0	0.000819	2.85732
9	2.14276	0	0.0003278	2.85707
10	2.14290	0	0.0001311	2.85717

Table 7.6: Iterates of relative value iteration for Example 7.8.

Some concluding remarks regarding average reward value iteration

Given the complexity of this analysis of value iteration, a concluding recommendation may be helpful. In practice, one should use relative value iteration. An aperiodicity transformation can be applied if necessary to ensure convergence in most cases, the exception being multi-chain models in which the optimal policy does *not* have constant gain. Convergent rate estimates require additional analyses beyond the scope of this book.

7.7 Policy iteration

This section describes a policy iteration algorithm for recurrent and unichain models, illustrates its application in an example, proves its convergence and comments on some salient features.

7.7.1 An algorithm

The following policy iteration algorithm, expressed in vector notation, provides a high-level perspective on its key steps.

Algorithm 7.3. Policy iteration for recurrent and unichain models: vector form

1. **Initialize:** Specify $d \in D^{\text{MD}}$ and set $\Gamma \leftarrow S$.
2. While $\Gamma \neq \emptyset$:
   (a) **Evaluate:** Obtain g' and $\mathbf{h}'$ by solving
   $$\mathbf{h} = \mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h}. \tag{7.89}$$
   (b) **Improve:** Choose
   $$d' \in \arg\underset{\delta \in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_\delta + \mathbf{P}_\delta \mathbf{h}'\}. \tag{7.90}$$
   setting $d' = d$ if possible.
   (c) $\Gamma \leftarrow \{s \in S \,|\, d'(s) \neq d(s)\}$.
   (d) **Update:** $d \leftarrow d'$.
3. **Terminate:** Return $d^* = d$, $g^* = g'$ and $\mathbf{h}^* = \mathbf{h}'$.

Its restatement in component notation better describes the details of an implementable algorithm. This description emphasizes that the improvement step is implemented state by state.

Algorithm 7.4. Policy iteration for recurrent and unichain models: component form

1. **Initialize:** Specify $a_s \in A_s$ for all $s \in S$ and set $\Gamma \leftarrow S$.
2. While $\Gamma \neq \emptyset$:

(a) **Evaluate:** Obtain g' and $h'(s)$ for all $s \in S$ by solving the system of linear equations

$$h(s) = r(s, a_s) - g + \sum_{j \in S} p(j|s, a_s)h(j). \tag{7.91}$$

(b) **Improve:** For all $s \in S$, choose

$$a'_s \in \arg\max_{a \in A_s} \left\{ r(s, a) + \sum_{j \in S} p(j|s, a)h'(j) \right\}, \tag{7.92}$$

setting $a'_s = a_s$ if possible.

(c) $\Gamma \leftarrow \{s \in S \,|\, a'_s \neq a_s\}$.

(d) **Update:** For all $s \in S$, $a_s \leftarrow a'_s$.

3. **Terminate:** Return $d^*(s) = a_s$ and $h^*(s) = h'(s)$ for all $s \in S$, and $g^* = g'$.

Some comments follow:

1. The evaluation step uniquely determines g and determines $h(s)$ up to a constant. This arbitrary constant does not impact decision rule or action choice in the subsequent improvement step, however its specification is required to implement it numerically. The easiest approach is to set $h(s') = 0$ for some distinguished state s'. In this case the evaluation step generates relative values. Note this does not apply to multi-chain models.

2. The value of g does not enter into the improvement step. Therefore $h(s)$ contains all of the information required to possibly find an improved decision rule.

3. This algorithm is not impacted by periodicity since solution of the evaluation equation is exact.

4. The improvement step requires the same effort as one pass through value iteration. It is implemented component-wise to avoid enumerating all Markovian deterministic decision rules. Modified policy iteration, which is not described here, replaces this step with an iterative method for approximating the value function of $(d')^\infty$.

5. The instruction "set $d'' = d'$ if possible" prevents cycling when the maximizer is non-unique in some states. It is also fundamental to enable the stopping condition to be satisfied. Without this rule the stopping criterion can be replaced by one based on the value functions being equal at two iterations. However, one must account for the numerical precision to which these quantities have been computed. Alternatively a span-based stopping condition could be used.

7.7.2 An example

This section solves the two-state example from Section 2.5 using the above policy iteration algorithm expressed in component form. It is easy to see that this model is unichain because state s_2 is absorbing under action $a_{2,1}$. This implementation arbitrarily sets $h(s_2) = 0$ for convenience.

1. Choose $a'_{s_1} = a_{1,2}$ and $a'_{s_2} = a_{2,1}$.

2. The evaluation equations (7.91) become:

$$\begin{aligned} h(s_1) &= 5 - g + h(s_2) \\ h(s_2) &= -5 - g + h(s_2). \end{aligned}$$

   Setting $h(s_2) = 0$ gives $g = -5$ and $h(s_1) = 10$.

3. In the improvement step (7.92)

$$\begin{aligned} a''_{s_1} &\in \arg\max_{i \in \{1,2\}} \left\{ r(s_1, a_{1,i}) + \sum_{j=1}^{2} p(s_j|s_1, a_{1,i}) h(s_j) \right\} \\ &= \arg\max\{3 + 0.8h(s_1) + 0.2h(s_2), 5 + h(s_2)\} \\ &= \arg\max\{11, 5\} = a_{1,1} \end{aligned}$$

   and

$$\begin{aligned} a''_{s_2} &\in \arg\max_{i \in \{1,2\}} \left\{ r(s_2, a_{2,i}) + \sum_{j=1}^{2} p(s_j|s_2, a_{2,i}) h(s_j) \right\} \\ &= \arg\max\{-5 + h(s_2), 2 + 0.4h(s_1) + 0.6h(s_2))\} \\ &= \arg\max\{-5, 6\} = a_{2,2} \end{aligned}$$

4. Since $a''_s \neq a'_s$ for both s, replace a'_s by a''_s for all $s \in S$ and return to the evaluation step.

5. Evaluating this decision rule ($(a'_{s_1}, a'_{s_2}) = (a_{1,1}, a_{2,2})$) using (7.91) together with $h(s_2) = 0$ yields $g = 8/3 = 2.6667$ and $h(s_1) = 5/3 = 1.6667$. Hence g has increased.

6. Applying the improvement step gives

$$\begin{aligned} a''_{s_1} &\in \arg\max\{13/3, 5\} = a_{1,2} \\ a''_{s_2} &\in \arg\max\{-5, 8/3\} = a_{2,2}. \end{aligned}$$

7. Since $a''_s \neq a'_s$ for s_1, replace a'_s by a''_s for all $s \in S$ and return to the evaluation step.

8. Evaluating this decision rule ($(a'_{s_1}, a'_{s_2}) = (a_{1,2}, a_{2,2})$) using (7.91) together with $h(s_2) = 0$ yields $g = 20/7 = 2.857$ and $h(s_1) = 15/7 = 2.143$. Hence g has increased.

9. Applying the improvement step again gives

$$a''_{s_1} \in \arg\max\{29.735, 30.147\} = a_{1,2}$$
$$a''_{s_2} \in \arg\max\{20.146, 27.941\} = a_{2,2}.$$

10. Since $a''_s = a'_s$ for all $s \in S$, stop.

The above calculations showed that it required two improvement steps to find the optimal policy and an additional improvement step to confirm its optimality. This is because the algorithm was initiated with the worst policy. A better starting value would have led to faster convergence.

Observe that the algorithm terminates with:

1. the optimal policy d^*,
2. the optimal gain,
3. the relative values[32] of d^*.
4. a solution of the Bellman equation.

Moreover g increased from iteration to iteration. This need not be the case in general; at some iterations g may remain the same and h increase. This observation will be the basis for a convergence proof for policy iteration.

7.7.3 Convergence of recurrent and unichain policy iteration

This section establishes that policy iteration finds an optimal policy in a finite number of iterations. The proof is based on the following observations:

1. If there is a strict improvement in a recurrent state of a decision rule obtained in the improvement step, then the gain increases.
2. If there is a strict improvement in a transient state of a decision rule obtained in the improvement step and no change in action in any recurrent state, then the gain remains the same but the bias increases.

[32] Note that if the condition $h(s_2) = 0$ was replaced by the more computationally intensive condition $\mathbf{P}_{d^*}\mathbf{h} = \mathbf{0}$, $h(s)$ would equal the bias of d^*.

The consequence of this is that the iterates of policy iteration are monotone in a *lexicographic* (or dictionary-ordering) sense[33]. What this means that if the solution of the evaluation equations is expressed as a vector $(g, \mathbf{h})$ then either the first component g increases or g remains the same and $\mathbf{h}$ increases. Note that there are several other ways of proving this result[34]; we believe this approach is the most informative.

Improvements on recurrent states

The following proposition establishes the first statement above. It expresses the gain of a stationary policy δ^∞ in terms of a one-step "Bellman update" of the gain and bias of a different stationary policy d^∞. It then explores the implications of δ^∞ representing a strict improvement of d^∞ in the recurrent states of δ^∞. Note the result applies to any scalar g and $\mathbf{h} \in \mathbb{R}^{|S|}$ but it is stated to be compatible with the average reward policy iteration algorithm. It is often referred to as a "comparison lemma".

Proposition 7.6. Suppose in a recurrent or unichain model that for some $d \in D^{\mathrm{MD}}$, g and $\mathbf{h}$ satisfy

$$\mathbf{0} = \mathbf{r}_d - g\mathbf{e} + \mathbf{P}_d\mathbf{h} - \mathbf{h}. \tag{7.93}$$

Then for any $\delta \in D^{\mathrm{MD}}$:

1. $$g^{\delta^\infty}\mathbf{e} = g\mathbf{e} + \mathbf{P}^*_\delta(\mathbf{r}_\delta - g\mathbf{e} + (\mathbf{P}_\delta - \mathbf{I})\mathbf{h}). \tag{7.94}$$

2. Suppose
$$r(s,\delta(s)) - g + \sum_{j\in S} p(j|s,\delta(s))h(j) - h(s) \geq 0 \tag{7.95}$$
for all recurrent states under $\mathbf{P}_\delta$ with strict inequality for some s that is recurrent under $\mathbf{P}_\delta$, then $g^{\delta^\infty} > g$.

3. If
$$r(s,\delta(s)) - g + \sum_{j\in S} p(j|s,\delta(s))h(j) - h(s) = 0 \tag{7.96}$$
for all s that are recurrent under δ, then $g^{\delta^\infty} = g$.

Proof. Since $g^{\delta^\infty}\mathbf{e} = \mathbf{P}^*_\delta\mathbf{r}_\delta$, $\mathbf{P}^*_\delta(\mathbf{P}_\delta - \mathbf{I}) = \mathbf{0}$ and $\mathbf{P}^*_\delta\mathbf{e} = \mathbf{e}$, it follows that

$$\begin{aligned} g^{\delta^\infty}\mathbf{e} =& \mathbf{P}^*_\delta\mathbf{r}_\delta + g\mathbf{e} - g\mathbf{e} + \mathbf{P}^*_\delta(\mathbf{P}_\delta - \mathbf{I})\mathbf{h} \\ =& g\mathbf{e} + \mathbf{P}^*_\delta(\mathbf{r}_\delta - g\mathbf{e} + (\mathbf{P}_\delta - \mathbf{I})\mathbf{h}) \end{aligned}$$

[33] A vector $\mathbf{a}$ is lexicographicaly larger (smaller) than a vector $\mathbf{b}$ of the same length if the first non-zero component of $\mathbf{a} - \mathbf{b}$ is positive (negative).

[34] One can use the partial Laurent expansion or the representation $\mathbf{v}_n = ng\mathbf{e} + \mathbf{h} + \mathbf{o}(1)$.

from which part 1 follows.

Parts 2 and 3 follow immediately from Lemma B.4 in Appendix B, which shows that the components of the limiting matrix $\mathbf{P}^*_\delta$ satisfy $p^*(j|s) > 0$ when both s and j are recurrent states and $p^*(j|s) = 0$ for all s and all transient states j .

□

Some comments follow:

1. The statement of the above proposition combines vector and component notation. This is intended to make the results as transparent as possible.

2. In the context of policy iteration, δ in part 1. will be chosen according to

$$\delta \in \arg\max_{d' \in D^{\text{MD}}} \{\mathbf{r}_{d'} + \mathbf{P}_{d'}\mathbf{h}\}. \tag{7.97}$$

3. Since $p^*_\delta(j|s) = 0$ when j is transient under δ, changes on transient states of δ do not effect the average reward. Note this structure was fundamental to the analysis of transient expected total reward models.

4. In a recurrent model the above result is sufficient to establish the convergence of policy iteration.

5. This result can be generalized to multi-chain models[35].

Improvement on transient states

The following proposition provides the basis for the statement that there is a strict improvement in a transient state of a greedy decision rule and no change in action in any recurrent state, then the gain remains the same but the bias increases. the beginning of this section. Since it is concerned with behavior on transient states, it is vacuous when applied to a recurrent model. The proof relies heavily on the structure of $\mathbf{P}^*_\delta$ for unichain policies.

The proposition statement is simplified by using the following additional notation. For a unichain decision rule δ, let R_δ and T_δ denote the recurrent and transient states of $\mathbf{P}_\delta$, for an arbitrary vector $\mathbf{v}$ let $\mathbf{v}_R$ and $\mathbf{v}_T$ respectively denote the sub-vectors of $\mathbf{v}$ restricted to R_δ and T_δ and let $(\mathbf{P}_\delta)_{TT}$ denote the sub-matrix of $\mathbf{P}_\delta$ corresponding to transitions between the transient states of $\mathbf{P}_\delta$. The proof is based on applying Theorem 7.3, partitioning

$$\mathbf{H}_d = \begin{bmatrix} (\mathbf{H}_d)_{RR} & (\mathbf{H}_d)_{RT} \\ (\mathbf{H}_d)_{TR} & (\mathbf{H}_d)_{TT} \end{bmatrix} \tag{7.98}$$

for a Markovian decision rule d, and using the properties of the sub-matrices of $\mathbf{H}_d$.

[35] See Lemma 9.2.5 in Puterman [1994].

Proposition 7.7. Suppose in a unichain model that for some $d \in D^{\mathrm{MD}}$, g and $\mathbf{h}$ satisfy (7.93) subject to $\mathbf{P}_d^* \mathbf{h} = \mathbf{0}$. Then for any $\delta \in D^{\mathrm{MD}}$:

1. $$\mathbf{h}^{\delta^\infty} = (\mathbf{I} - \mathbf{P}_\delta^*)\mathbf{h} + \mathbf{H}_\delta(\mathbf{r}_\delta - g\mathbf{e} + \mathbf{P}_\delta \mathbf{h} - \mathbf{h}). \tag{7.99}$$

2. Suppose $\delta(s) = d(s)$ for $s \in R_d$. Then $\mathbf{h}_R^{\delta^\infty} = \mathbf{h}_R$ and
$$\mathbf{h}_T^{\delta^\infty} = \mathbf{h}_T + (\mathbf{I} - \mathbf{P}_\delta)_{TT}^{-1}(\mathbf{r}_\delta - g\mathbf{e} + \mathbf{P}_\delta \mathbf{h} - \mathbf{h})_T. \tag{7.100}$$

3. Suppose
$$r(s, \delta(s)) - g + \sum_{j \in S} p(j|s, \delta(s))h(j) - h(s) = 0 \text{ for } s \in R_\delta \tag{7.101}$$
and
$$r(s, \delta(s)) - g + \sum_{j \in S} p(j|s, \delta(s))h(j) - h(s) \geq 0 \text{ for } s \in T_\delta \tag{7.102}$$
with strict inequality for some $s \in T_\delta$.
Then $g^{\delta^\infty} = g$ and $h^{\delta^\infty}(s) \geq h(s)$ for all $s \in S$ with $h^{\delta^\infty}(s') > h(s')$ for some $s' \in S$.

4. Suppose
$$r(s, \delta(s)) - g + \sum_{j \in S} p(j|s, \delta(s))h(j) - h(s) = 0 \tag{7.103}$$
for all $s \in S$. Then $g^{\delta^\infty} = g$ and $\mathbf{h}^{\delta^\infty} = \mathbf{h}$.

Proof. Since $\mathbf{h}^{\delta^\infty} = \mathbf{H}_\delta \mathbf{r}_\delta$, $\mathbf{H}_\delta \mathbf{e} = \mathbf{0}$ and $\mathbf{H}_\delta(\mathbf{P}_\delta - \mathbf{I}) = \mathbf{P}_\delta^* - \mathbf{I}$ it follows after a bit of manipulation that

$$\begin{aligned}\mathbf{h}^{\delta^\infty} &= \mathbf{H}_\delta \mathbf{r}_\delta + \mathbf{H}_\delta\big((\mathbf{P}_\delta - \mathbf{I})\mathbf{h} - (\mathbf{P}_\delta - \mathbf{I})\mathbf{h}\big) \\ &= (\mathbf{I} - \mathbf{P}_\delta^*)\mathbf{h} + \mathbf{H}_\delta\big(\mathbf{r}_\delta - g\mathbf{e} + (\mathbf{P}_\delta - \mathbf{I})\mathbf{h}\big)\end{aligned}$$

establishing part 1.

To prove part 2, suppose $\delta(s) = d(s)$ when s is recurrent under d, then $(\mathbf{r}_\delta - g\mathbf{e} + \mathbf{P}_\delta \mathbf{h} - \mathbf{h})_R = \mathbf{0}$ by the hypothesis that g and $\mathbf{h}$ satisfy (7.93) subject to $\mathbf{P}_d^* \mathbf{h} = \mathbf{0}$. By the properties of limiting matrices, $\mathbf{P}_d^* = \mathbf{P}_\delta^*$ so it follows that $\mathbf{P}_\delta^* \mathbf{h} = \mathbf{P}_d^* \mathbf{h} = \mathbf{0}$. From Lemma B.5, $(\mathbf{H}_\delta)_{RT} = \mathbf{0}$ and $(\mathbf{H}_\delta)_{TT} = (\mathbf{I} - \mathbf{P}_\delta)_{TT}^{-1}$. Hence part 2. follows by substituting these observations into (7.99).

To prove part 3, note that under (7.101), it follows from Proposition 7.6 that $g^{\delta^\infty} = g$. Since $(\mathbf{I} - \mathbf{P}_\delta)_{TT}^{-1} \geq \mathbf{I}$ from part 3 of Lemma B.5, combining (7.102) with representation (7.100) establishes the result.

Part 4 follows immediately from part 2. For both parts 3 and 4, $g^{\delta^\infty} = g$ follows from Proposition 7.6. □

Some comments about this proposition follow:

1. The additional hypothesis that $\mathbf{P}_d^* \mathbf{h} = \mathbf{0}$ implies that $\mathbf{h} = \mathbf{h}^{\delta^\infty}$. It is not required to derive (7.99) but is fundamental for deriving (7.100).

2. Note that the increase in $\mathbf{h}$ in part 3 occurs in transient states of $\mathbf{P}_\delta$.

3. The above propositions do not account for the eventuality that

$$r(s, \delta(s)) - g + \sum_{j \in S} p(j|s, \delta(s))h(j) - h(s)$$

is positive on some states and negative in others. When δ is derived from d in the policy iteration improvement step, this is impossible.

4. It might at first appear that these propositions do not consider the the possibility that δ represents an improvement in **both** recurrent and transient states. However as a consequence of Proposition 7.6, this would result in an increase in the average reward. This is not of concern since it will not impact the proof of convergence of policy iteration below.

Convergence of policy iteration

The following theorem applies Propositions 7.6 and 7.7 to establish convergence of policy iteration.

Theorem 7.11. In a recurrent or unichain model, policy iteration converges in a finite number of iterations to a solution $(g^*, \mathbf{h})$ of the Bellman equation (7.47). Moreover if

$$d^* \in \arg \operatorname*{c\text{-}max}_{d \in D^{\mathrm{MD}}} \{\mathbf{r}_d + \mathbf{P}_d \mathbf{h}\} \tag{7.104}$$

then $(d^*)^\infty$ is average optimal.

Proof. The proof uses the vector version of the recurrent and unichain policy iteration algorithm above. Let $d' \in D^{\mathrm{MD}}$. Clearly

$$\operatorname*{c\text{-}max}_{d \in D^{\mathrm{MD}}} \left\{ \mathbf{r}_d - g^{(d')^\infty} \mathbf{e} + \mathbf{P}_d \mathbf{h}^{(d')^\infty} - \mathbf{h}^{(d')^\infty} \right\} \geq \mathbf{r}_{d'} - g^{(d')^\infty} \mathbf{e} + \mathbf{P}_{d'} \mathbf{h}^{(d')^\infty} - \mathbf{h}^{(d')^\infty} = \mathbf{0}.$$

Thus when d'' is chosen in the improvement step using (7.90),

$$\mathbf{r}_{d''} - g^{(d')^\infty} \mathbf{e} + \mathbf{P}_{d''} \mathbf{h}^{(d')^\infty} - \mathbf{h}^{(d')^\infty} \geq \mathbf{0}. \tag{7.105}$$

If equality holds in (7.105), $(g^{(d')^\infty}, \mathbf{h}^{(d')^\infty})$ is a solution of the unichain Bellman equation, completing the proof. If not, and there is strict inequality in (7.105), for some state in $R_{d''}$ then $g^{(d'')^\infty} > g^{(d')^\infty}$ from Proposition 7.6.

Finally, if equality holds in (7.105) on $R_{d''}$ and there is strict inequality for some state in $T_{d''}$, Proposition 7.7 implies $g^{(d'')^\infty} = g^{(d')^\infty}$ and $h^{(d'')^\infty}(s) \geq h^{(d')^\infty}(s)$ for all s with strict inequality in some state. Since there are only finitely many stationary deterministic policies, only a finite number of such increases are possible. Thus, the result follows. □

Some comments follow:

1. The above result shows the average reward policy iteration algorithm finds a solution of the optimality equation, an optimal policy and its average reward.

2. Depending on the specification used to uniquely determine $\mathbf{h}$, it finds either the bias or relative values of an optimal policy.

3. If policy iteration is implemented with $h(s') = 0$ for some distinguished state s', the above proof still holds because Proposition 7.7 shows the bias increases when the only strict improvement is in a transient state.

4. The above theorem also provides a constructive proof of the existence of a solution for the Bellman equation (7.47).

A curious example

Note that the above analysis does not conclude anything regarding optimality with respect to the bias. The following example shows that policy iteration finds an average optimal policy but not one with optimal bias among average optimal policies.

Example 7.11. Let $S = \{s_1, s_2\}$; $A_{s_1} = \{a_{1,1}, a_{1,2}\}$, $A_{s_2} = \{a_{2,1}\}$; $r(s_1, a_{1,1}) = 2$, $r(s_1, a_{1,2}) = 0$, $r(s_2, a_{2,1}, s_1) = 6$, $r(s_2, a_{2,1}, s_2) = 0$; $p(s_1|s_1, a_{1,1}) = 1$, $p(s_2|s_1, a_{1,2}) = 1, p(s_1|s_2, a_{2,1}) = p(s_1|s_2, a_{2,1}) = 0.5$. Note that $r(s_2, a_{2,1}) = 3$ and the model is unichain. See Figure 7.8.

Since actions are only chosen in state s_1, Markovian deterministic decision rules can be specified by their choice of action in s_1. Let d denote the deterministic decision rule that uses action $a_{1,1}$ and f denote the deterministic decision rule that uses action $a_{1,2}$.

Suppose policy iteration begins with policy f^∞. Evaluating this policy gives $g^{f^\infty} = 2$, $h^{f^\infty}(s_1) = -4/3$ and $h^{f^\infty}(s_2) = 2/3$. The improvement step in s_1 gives

$$\max_{a \in A_{s_1}} \left\{ r(s_1, a) + p(s_1|s_1, a) h^{f^\infty}(s_1) + p(s_2|s_1, a) h^{f^\infty}(s_2) \right\}$$

$$= \max\left\{2 + h^{f^\infty}(s_1), h^{f^\infty}(s_2)\right\} = \max\left\{\frac{2}{3}, \frac{2}{3}\right\}.$$

Thus, policy iteration terminates with policy f^∞.

Evaluating d^∞ gives $g^{d^\infty} = 2$, $h^{d^\infty}(s_1) = 0$ and $h^{d^\infty}(s_2) = 2$. Therefore the bias of d^∞ exceeds that of f^∞. Consequently in this example, policy iteration finds an average optimal policy but not a policy with optimal bias among average optimal policies. This is because d and f have different sets of recurrent states.

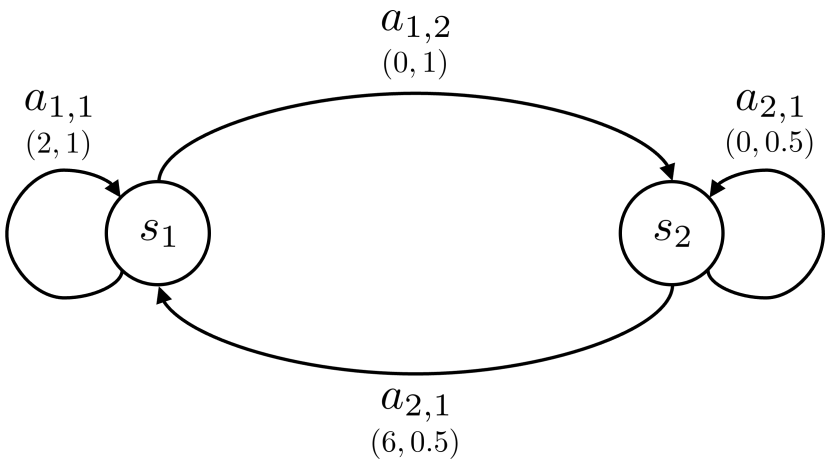


Figure 7.8: Graphical representation of model in Example 7.11 illustrating that policy iteration need not find a policy with optimal bias among average optimal policies. The labels (r, p) refer to the reward and transition probabilities, respectively, associated with using the specified action in the given state.

7.8 Linear programming

The linear programming formulation of the average reward Markov decision process is noteworthy because:

1. Dual variables represent state-action probabilities generated by randomized policies, and

2. The model allows direct inclusion of meaningful and practical constraints.

This section includes a detailed analysis of linear programming in recurrent models followed by generalization to unichain models. Linear programming in multi-chain models is more complicated and its analysis is beyond the scope of this book.

7.8.1 Linear programming formulation

In both recurrent and unichain models, in which the average reward is constant, a solution of the *single* Bellman equation (7.46) can be used to identify optimal policies. As a consequence of part 1 of Theorem 7.7, if there exist a scalar g and vector $\mathbf{h}$ such that

$$h(s) \geq r(s,a) - g + \sum_{j \in S} p(j|s,a)h(j), \tag{7.106}$$

for all $a \in A_s$ and $s \in S$, then $g \geq g^*$. Hence g^* represents the smallest g for which there exists $h(s)$ satisfying (7.106) for all $s \in S$ and $a \in A_s$. This observation gives rise to the following primal linear program for finding such a quantity.

Primal LP formulation of average reward model

$$\underset{g,\mathbf{h}}{\text{minimize}} \quad g \tag{7.107a}$$

$$\text{subject to} \quad g + h(s) - \sum_{j \in S} p(j|s,a)h(j) \geq r(s,a), \quad a \in A_s, s \in S \tag{7.107b}$$

The corresponding dual linear program is given by:

Dual LP formulation of average reward model

$$\text{maximize} \quad \sum_{s \in S} \sum_{a \in A_s} r(s,a)x(s,a) \tag{7.108a}$$

$$\text{subject to} \quad \sum_{a \in A_j} x(j,a) - \sum_{s \in S} \sum_{a \in A_s} p(j|s,a)x(s,a) = 0, \quad j \in S, \tag{7.108b}$$

$$\sum_{s \in S} \sum_{a \in A_s} x(s,a) = 1. \tag{7.108c}$$

$$x(s,a) \geq 0 \text{ for } a \in A_s \text{ and } s \in S. \tag{7.108d}$$

Note that one of the constraints is redundant. Hence the set of constraints of the dual linear program have at most $|S|$ linearly independent rows. To see this, sum (7.108b) over $j \in S$ to obtain

$$\sum_{j \in S} \sum_{a \in A_j} x(j,a) - \sum_{j \in S} \sum_{s \in S} \sum_{a \in A_s} p(j|s,a)x(s,a)$$
$$= \sum_{j \in S} \sum_{a \in A_j} x(j,a) - \sum_{s \in S} \sum_{a \in A_s} x(s,a) = 0.$$

The following example illustrates the linear programming formulations of the two-state model.

Example 7.12.
The primal LP for the two-state model in Section 2.5 is

$$\begin{aligned} \text{minimize} \quad & g \\ \text{subject to} \quad & g + h(s_1) - 0.8h(s_1) - 0.2h(s_2) \geq 3 \\ & g + h(s_1) - h(s_2) \geq -5 \\ & g + h(s_2) - h(s_2) \geq -5 \\ & g + h(s_2) - 0.4h(s_1) - 0.6(s_2) \geq 2 \end{aligned}$$

Observe that the primal LP has three variables, four constraints (one for each state-action pair) and no non-negativity constraints. After simplification the constraints can be written as

$$\begin{aligned} & g + 0.2\big(h(s_1) - h(s_2)\big) \geq 3 \\ & g + \big(h(s_1) - h(s_2)\big) \geq -5 \\ & g \geq -5 \\ & g - 0.4\big(h(s_1) - h(s_2)\big) \geq 2. \end{aligned}$$

These inequalities show that $h(s_1)$ and $h(s_2)$ cannot be independently determined.

The dual LP is given by

$$\begin{aligned} \text{maximize} \quad & 3x(s_1, a_{1,1}) - 5x(s_1, a_{1,2}) - 5x(s_2, a_{2,1}) + 2x(s_2, a_{2,2}) \\ \text{subject to} \quad & x(s_1, a_{1,1}) + x(s_1, a_{1,2}) - 0.8x(s_1, a_{1,1}) - 0.4x(s_2, a_{2,2}) = 0 \\ & x(s_2, a_{2,1}) + x(s_2, a_{2,2}) - 0.2x(s_1, a_{1,1}) \\ & \quad - x(s_1, a_{1,2}) - x(s_2, a_{2,1}) - 0.6x(s_2, a_{2,2}) = 0 \\ & x(s_1, a_{1,1}) + x(s_1, a_{1,2}) + x(s_2, a_{2,1}) + x(s_2, a_{2,2}) = 1 \\ & x(s_1, a_{1,1}), x(s_1, a_{1,2}), x(s_2, a_{2,1}), x(s_2, a_{2,2}) \geq 0. \end{aligned}$$

The dual linear program has four variables (one for each state-action pair), three equality constraints (one for each of the two state-action pairs and one corresponding to the sum of the dual variables equaling one), and non-negativity constraints.

Rearranging terms in the first two constraints of the dual shows that one of them is redundant. Hence the dual has a total of two independent constraints.

As in Chapters 5 and 6, an analysis of the dual LP is the most insightful. Recurrent and unichain models require distinct analyses.

7.8.2 Recurrent models

This section establishes for recurrent models that:

1. There is a one-to-one relationship between randomized stationary policies and feasible solutions of the dual linear program,
2. There is a one-to one relationship between deterministic stationary policies and basic feasible solutions of the dual linear program, and
3. An optimal basic feasible solution to the dual linear program exists. From it an optimal deterministic policy can be derived.

Feasible solutions and randomized policies

The following theorem relates feasible[36] solutions of the dual LP to the stationary probability that a Markovian randomized decision rule occupies state $s \in S$ and chooses action $a \in A_s$. Recall that $w_d(a|s)$ denotes the probability that Markovian randomized decision rule d chooses action a in state s and for a Markovian randomized decision rule d, that $p_d(j|s) = \sum_{a \in A_s} p(j|s,a) w_d(a|s)$.

The proof of the theorem below uses part 3 of Theorem B.1 in Appendix B, which is restated here.

Proposition 7.8. In a recurrent model, the stationary distribution $q_d(s)$ of the Markov chain corresponding to Markovian decision rule d is the unique non-negative solution of

$$\sum_{j \in S} q(s) p_d(j|s) = q(s) \quad \text{for } s \in S \tag{7.109}$$

subject to $\sum_{s \in S} q(s) = 1$. Moreover, $q_d(s) > 0$ for all $s \in S$.

Note that in a unichain model, $q_d(s)$ is positive on the recurrent states of $\mathbf{P}_d$ and zero on its transient states.

[36] Recall that a *feasible solution* of a linear program is one that satisfies all of its constraints.

Theorem 7.12. In a recurrent model:

1. For each $d \in D^{\mathrm{MR}}$,
$$x_d(s,a) := w_d(a|s)q_d(s) \tag{7.110}$$
is a feasible solution to the dual linear program.

2. (a) Let $x(s,a)$ be a feasible solution to the dual linear program. Then for each $s \in S$,
$$\sum_{a\in A_s} x(s,a) > 0. \tag{7.111}$$

   (b) Suppose the Markovian randomized decision rule $d_{\mathbf{x}}$ has action choice probability
$$w_{d_{\mathbf{x}}}(a|s) := \frac{x(s,a)}{\sum_{a\in A_s} x(s,a)}. \tag{7.112}$$
   Then $x_{d_{\mathbf{x}}}(s,a) = w_{d_{\mathbf{x}}}(a|s)q_{d_{\mathbf{x}}}(s)$ is a feasible solution to the dual linear program.

   (c) For each $s \in S$ and $a \in A_s$,
$$x_{d_{\mathbf{x}}}(s,a) = x(s,a). \tag{7.113}$$

Proof. To prove part 1 substitute $x_d(s,a)$ into the left hand side of (7.108b). As a consequence of (7.109), that $p_d(j|s) = \sum_{a\in A_s} p(j|s,a)w_d(a|s)$ and $\sum_{a\in A_s} w_d(a|s) = 1$, it follows for all $s \in S$ that

$$\sum_{a\in A_s} w_d(a|s)q_d(s) - \sum_{s'\in S}\sum_{a\in A_{s'}} p(s|s',a)w_d(a|s')q_d(s') = q_d(s) - \sum_{j\in S} p_d(j|s)q_d(s) = 0.$$

Moreover it is easy to see that $x_d(s,a)$ satisfies (7.108c) and (7.108d) so the first result follows.

To prove 2(a) define $f(s) := \sum_{a\in A_s} x(s,a)$ and let $S' \subseteq S$ denote the set of $s \in S$ for which $f(s) > 0$. To show that $S' = S$, note first that

$$\begin{aligned}
\sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)x(s,a) &= \sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)x(s,a)\frac{f(s)}{f(s)} \\
&= \sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)\frac{x(s,a)}{\sum_{a\in A_s} x(s,a)} f(s) \\
&= \sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)w_{d'}(a|s)f(s) = \sum_{s\in S'} p_{d'}(j|s)f(s),
\end{aligned}$$

where the randomized decision rule $d'(s)$ selects action $a \in A_s$ for $s \in S'$ with proba-

bility[37]

$$w_{d'}(a|s) := \frac{x(s,a)}{\sum_{a\in A_s} x(s,a)}.$$

From (7.108b) and the above equality it follows that for $j \in S'$,

$$\begin{aligned} 0 &= f(j) - \sum_{s\in S}\sum_{a\in A_s} p(j|s,a)x(s,a) \\ &= f(j) - \sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)x(s,a) - \sum_{s\in S\setminus S'}\sum_{a\in A_s} p(j|s,a)x(s,a) \\ &= f(j) - \sum_{s\in S'}\sum_{a\in A_s} p(j|s,a)x(s,a) \\ &= f(j) - \sum_{s\in S'} p_{d'}(j|s)f(s), \end{aligned} \tag{7.114}$$

where the third equality follows from the fact that $f(s) = 0$ for $s \in S\setminus S'$ and $x(s,a) \geq 0$ together imply $x(s,a) = 0$ for all $a \in A_s, s \in S \setminus S'$. Moreover, because $f(s) = 0$ for $s \in S \setminus S'$ and (7.108c),

$$\sum_{s\in S'} f(s) = \sum_{s\in S} f(s) = \sum_{s\in S}\sum_{a\in A_s} x(s,a) = 1. \tag{7.115}$$

Since $f(s) > 0$ solves (7.114) and (7.115), it follows from Proposition 7.8, $f(s)$ is the stationary distribution of d', that is $f(s) = q_{d'}(s)$ for all $s \in S'$ and by (7.115), $\sum_{s\in S'} q_{d'}(s) = 1$. Since in a recurrent model, all states are recurrent under every stationary policy, $q_{d'}(s) > 0$ for all $s \in S$. Hence, $S' = S$, proving 2(a).

As a consequence of 2(a), $w_{d_{\mathbf{x}}}(a|s) \geq 0$ is well defined for all $s \in S$ and $a \in A_s$, non-negative and $\sum_{a\in A_s} w_{d_{\mathbf{x}}}(a|s) = 1$. Therefore $w_{d_{\mathbf{x}}}(a|s)$ is a probability distribution on A_s for each $s \in S$ so that $d_{\mathbf{x}}$ is a Markovian randomized decision rule. Hence from part 1 $x_{d_{\mathbf{x}}}(s,a)$ satisfies the dual linear program.

Finally

$$x(s,a) = x(s,a)\frac{f(s)}{\sum_{a\in A_s} x(s,a)} = w_{d_{\mathbf{x}}}(a|s)q_{d_{\mathbf{x}}}(s) = x_{d_{\mathbf{x}}}(s,a),$$

where the second equality was established in the proof of 2(a). □

The following comments provide insight into the significance of the above result:

1. From an application perspective, (7.112) is fundamental. It shows how to derive a randomized stationary policy from a feasible solution of the dual linear program.

[37] Note that this is the same action choice probability of the decision rule $d_{\mathbf{x}}(s)$ defined in (7.112) but it is not yet determined that $S' = S$ so $d'(s)$ is only defined on S'.

2. As a consequence of part 2(c) of the above theorem, $x(s,a)$ is the joint stationary probability that the system corresponding to $d_{\mathbf{x}}$ occupies state s **and** chooses action a in steady state.

3. The previous comment implies that the dual objective function equals the stationary reward corresponding to the policy $(d_{\mathbf{x}})^{\infty}$. That is,

$$\sum_{s\in S}\sum_{a\in A_s} r(s,a)x(s,a) = g^{(d_{\mathbf{x}})^{\infty}}$$

4. The proof of 2(a) of Theorem 7.12 is quite subtle. It exploits the property that in a recurrent Markov chain, the stationary distribution is strictly positive in every state.

Basic feasible solutions and deterministic policies.

This section explores the relationship between basic feasible solutions[38] and deterministic stationary policies. The next result follows immediately from the previous theorem and the definition of a basic feasible solution.

Theorem 7.13. In a recurrent model:

1. Suppose $d \in D^{\mathrm{MD}}$. Then

$$x_d(s,a) := \begin{cases} q_d(s) & \text{when } d(s) = a \\ 0 & \text{when } d(s) \neq a \end{cases} \tag{7.116}$$

   is a basic feasible solution of the dual linear program.

2. Let $x(s,a)$ denote a basic feasible solution of the dual linear program.

   (a) Then $x(s,a) > 0$ for exactly one $a \in A_s$ for each $s \in S$.

   (b) For each $s \in S$, $d_{\mathbf{x}}(s) = a$ corresponds to $x(s,a) > 0$ so that $d_{\mathbf{x}} \in D^{\mathrm{MD}}$.

Proof. Part 1 follows immediately from part 1 of the preceding theorem.

As shown above, the dual has at most $|S|$ linearly independent rows. Thus, a basic feasible solution has at most $|S|$ positive components. Since part 2(a) of Theorem 7.12 shows that $\sum_{a\in A_s} x(s,a) > 0$ for each $s \in S$, each basic feasible solution has exactly one positive $x(s,a)$ for each $s \in S$. Hence Part 2(a) holds. Hence from (7.112), $w_{d_{\mathbf{x}}}(a|s) = 1$ for exactly one $a \in A_s$ for each $s \in S$ from which part 2(b) follows. □

[38] The proof of the following theorem uses the result that in a linear program with m constraints and n variables with $n > m$, a *basic feasible solution* has at most m non-zero entries. A formal definition of this concept appears in Appendix C.

This theorem represents the next building block in analyzing the dual linear program in recurrent average reward models, namely:

1. The assumption of a recurrent model is crucial to this result. It is used to show that $\sum_{a\in A_s} x(s,a) > 0$ for each $s \in S$. The next section considers transient models in which for some $s' \in S$, $x(s',a) = 0$ for all $a \in A_{s'}$.

2. Part 2(b) shows how to identify deterministic policies from basic feasible solutions. Namely for each state, choose the action a for which $x(s,a) > 0$.

3. Part 1 shows that the solution of the dual gives the stationary probability distribution for the Markov chain corresponding to $d_{\mathbf{x}}$.

Optimal solutions and optimal policies

Finally, this section relates optimal solutions to the dual with optimal deterministic stationary policies.

Theorem 7.14. Suppose in a recurrent model that $r(s,a)$ is bounded for all $s \in S$, and $a \in A_s$. Then there exists an optimal basic feasible solution $x^*(s,a)$ to the dual linear program. Moreover $(d_{\mathbf{x}^*})^\infty$ is an optimal deterministic stationary policy where $d_{\mathbf{x}^*}(s) = a$ when $x^*(s,a) > 0$.

Proof. Since $r(s,a)$ is bounded, Theorems C.1, C.2 and C.4 together imply that the dual LP has at least one optimal basic feasible solution. Let $x^*(s,a)$ denote one.

For any $d \in D^{\mathrm{MR}}$, $x_d(s,a)$ defined in part 1 of Theorem 7.12 is a feasible solution of the dual linear program so that by the optimality of $x^*(s,a)$,

$$\sum_{s\in S}\sum_{a\in A_s} r(s,a)x^*(s,a) \geq \sum_{s\in S}\sum_{a\in A_s} r(s,a)x_d(s,a).$$

Hence from Theorem 7.13 there exists a deterministic decision rule $d_{\mathbf{x}^*}(s)$ corresponding to $x^*(s,a)$ for which

$$\begin{aligned} g^{(d_{\mathbf{x}^*})^\infty} &= \sum_{s\in S}\sum_{a\in A_s} r(s,a)x_{d_{\mathbf{x}^*}}(s,a) = \sum_{s\in S}\sum_{a\in A_s} r(s,a)x^*(s,a) \\ &\geq \sum_{s\in S}\sum_{a\in A_s} r(s,a)x_d(s,a) = g^{d^\infty} \end{aligned}$$

for any $d \in D^{\mathrm{MR}}$. Hence $(d_{\mathbf{x}^*})^\infty$ is an optimal policy. □

Note that this theorem guarantees the existence of an optimal stationary deterministic policy but it does not exclude the possibility that there exist multiple optimal policies. In such a case there also exist optimal randomized policies.

Moreover Theorem C.6 guarantees the existence of an optimal solution $(g^*, \mathbf{h}^*)$ to the primal LP with $g^* = \sum_{s\in S}\sum_{a\in A_s} r(s,a)x^*(s,a)$. Hence the above analysis provides an alternative approach to establishing the existence of a solution to the average reward Bellman equation.

7.8.3 Unichain models

The situation is more complex in unichain models. The following is the most significant result; it is stated without proof[39].

Theorem 7.15. In a unichain model with $r(s,a)$ is bounded for all $a \in A_s$, $s \in S$:

1. There exists a bounded optimal basic feasible solution $x^*(s,a)$ to the dual linear program.

2. Let $S_{\mathbf{x}^*}$ denote the set of $s \in S$ for which $x^*(s,a) > 0$ for some $a \in A_s$. Then the policy $d^\infty_{\mathbf{x}^*}$ defined by

$$d_{\mathbf{x}^*}(s) = \begin{cases} a & \text{if } x^*(s,a) > 0, s \in S_{\mathbf{x}^*} \\ \text{arbitrary} & s \in S \setminus S_{\mathbf{x}^*} \end{cases} \tag{7.117}$$

is optimal and $S_{\mathbf{x}^*}$ equals its set of recurrent states.

Note that in a recurrent model, $S_{\mathbf{x}^*} = S$, so there is no arbitrariness in selecting an optimal policy to correspond to the solution. The following example illustrates the implications of this result by solving the dual linear program formulated in Example 7.12. Recall that the two-state model in Section 2.5 is unichain so that Theorem 7.15 applies.

Example 7.13. This example solves the dual linear program for the model in Example 7.12 and a variant. The analysis is divided into two cases, one in which the optimal policy is recurrent and the other in which it is unichain.
Case 1: Rewrite the dual LP as

$$\begin{aligned} \text{maximize} \quad & 3x(s_1,a_{1,1}) - 5x(s_1,a_{1,2}) - 5x(s_2,a_{2,1}) + 2x(s_2,a_{2,2}) \\ \text{subject to} \quad & 0.2x(s_1,a_{1,1}) + x(s_1,a_{1,2}) - 0.4x(s_2,a_{2,2}) = 0 \\ & -0.2x(s_1,a_{1,1}) - x(s_1,a_{1,2}) + 0.4x(s_2,a_{2,2}) = 0 \end{aligned}$$

[39] See Section 8.8.2 in Puterman [1994].

$$x(s_1, a_{1,1}) + x(s_1, a_{1,2}) + x(s_2, a_{2,1}) + x(s_2, a_{2,2}) = 1$$
$$x(s_1, a_{1,1}), x(s_1, a_{1,2}), x(s_2, a_{2,1}), x(s_2, a_{2,2}) \geq 0.$$

Note that $x(s_2, a_{2,1})$ does not appear in the first or second constraint and that one of those two constraints is redundant (one is a multiple of the other).

Solving this linear program using the simplex algorithm gives the optimal basic feasible solution $x(s_1, a_{1,1}) = 0.6667$, $x(s_1, a_{1,2}) = 0$, $x(s_2, a_{2,1}) = 0$ and $x(s_2, a_{2,2}) = 0.3333$ with objective function value of 2.6667. By Theorem 7.15, $S_{\mathbf{x}^*} = S$ and the stationary policy $d_{\mathbf{x}^*}(s_1) = a_{1,1}$ and $d_{\mathbf{x}^*}(s_1) = a_{2,2}$ is optimal.

Case 2: Suppose instead that $r(s_2, a_{2,1}) = 4$. In this case, the optimal solution is $x(s_1, a_{1,1}) = 0, x(s_1, a_{1,2}) = 0, x(s_2, a_{2,1}) = 1, x(s_2, a_{2,2}) = 0$ with an the objective function value of 4. Thus $S_{\mathbf{x}^*} = \{s_2\}$ and an optimal policy uses action $a_{2,1}$ in state s_2. Since $x(s_1, a_{1,1}) = x(s_1, a_{1,2}) = 0$, s_1 is transient and either action in s_1 gives the same average reward. Which action would you choose in s_1? (Why?)

7.8.4 Constrained Markov decision processes

Because variables in the dual linear program represent the stationary probability that the system is in state $s \in S$ and chooses action $a \in A_s$, the dual linear program provides a natural framework for adding one or more constraints of the form

$$\sum_{s \in S} \sum_{a \in A_s} c(s, a) x(s, a) \leq C. \tag{7.118}$$

The following examples show why such constraints could be useful.

Example 7.14. Consider the queuing service rate control model of Section 3.4.1. To model the constraint that the percentage of time that the fastest service rate a_f is used is at most α, (7.118) becomes

$$\sum_{s \in S} x(s, a_f) \leq \alpha,$$

corresponding to $c(s, a) = 1$ if $a = a_f$ and 0 otherwise.

Example 7.15. Consider the inventory model of Section 3.1. To add a constraint that the percentage of time that an order is backlogged[a] is less than or equal to α, (7.118) can be written as

$$\sum_{s \leq 0} \sum_{a \in A_s} x(s, a) \leq \alpha,$$

corresponding to $c(s,a) = 1$ if $s \leq 0$ and 0 otherwise.

[a]Recall that *backlogging* means an inventory level of 0 or below.

The following example examines the impact of adding a constraint in the two-state example.

Example 7.16. Consider the model analyzed as Case 1 in Example 7.13. Limiting the frequency at which the system occupies state s_1 to be at most 0.5 in steady state is captured through the constraint

$$x(s_1, a_{1,1}) + x(s_1, a_{1,2}) \leq 0.5.$$

Solving the model with this extra constraint yields $x^c(s_1, a_{1,1}) = 0.375$, $x^c(s_1, a_{1,2}) = 0.125$, $x^c(s_2, a_{2,1}) = 0$ and $x^c(s_2, a_{2,2}) = 0.5$. Since there are three non-zero variables, this is not a basic feasible solution for the unconstrained problem, but of course it is a basic feasible solution for the constrained problem.

Through (7.112) it generates the *randomized* policy $(d_{\mathbf{x}^c})^\infty$ with the following action selection probabilities:

$$w_{d_{\mathbf{x}^c}}(a_{1,1}|s_1) = \frac{0.375}{0.375 + 0.125} = 0.75, \quad w_{d_{\mathbf{x}^c}}(a_{1,2}|s_1) = \frac{0.125}{0.375 + 0.125} = 0.25,$$

$$w_{d_{\mathbf{x}^c}}(a_{2,1}|s_2) = 0, \qquad w_{d_{\mathbf{x}^c}}(a_{2,2}|s_2) = 1.$$

Moreover $g^{(d_{\mathbf{x}^c})^\infty} = 1.5$, which is considerably less than its value of 2.667 in the unconstrained model. Also, note that when $\alpha \geq 0.6667$ the model has the same solution as the unconstrained model since this constraint becomes non-binding.

The takeaway from this example is that in a constrained Markov decision process with one constraint, the optimal stationary policy may be randomized. In particular, when the constraint is binding, as when $\alpha = 0.5$ above, the optimal policy randomizes in a single state. In general, when there are K constraints, an optimal policy will randomize in at most K states.

7.9 Optimality of structured policies

The optimality of a structured policy in discounted and expected total reward models in Chapters 5 and 6 was a consequence of showing that:

1. the form of the value function (such as convexity) is preserved under application of the Bellman operator,

2. the limit of functions of the specific form retains the same form, and

3. greedy decision rules with respect to a function of this form has the desired structure.

A proof of this result was based on inductively showing that if value iteration were started with a value function of a particular form, all iterates of value iteration would retain this form as would the limits of the iterates. Since the limit is the optimal value function and greedy decision rules correspond to optimal stationary policies, the optimal stationary policy has the desired structure.

Since in an average reward model the iterates of value iteration diverge, the above analysis requires some modification. Instead, analysis can be based on the iterates of relative value iteration. To do so, distinguish a particular state s' and form iterates $w^n(s)$ based on the recursions:

$$v^{n+1}(s) = \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) w^n(j) \right\} \quad \text{and} \quad w^{n+1}(s) = v^{n+1}(s) - v^{n+1}(s').$$

A formal statement of the relevant result follows. Note that it requires assumptions regarding the model class that ensure convergence of relative value iteration.

Let V^F_{rel} denote the set of real-valued functions $\mathbf{w}$ on S for which $w(s') = 0$ for some designated $s' \in S$ and let D^F denote a set of decision rules that are compatible with V^F_{rel}.

Theorem 7.16. In an aperiodic recurrent, unichain, communicating or weakly communicating model, suppose that

1. V^F_{rel} is non-empty,

2. $\mathbf{w} \in V^F_{\text{rel}}$ implies $\mathbf{w}'$ defined by

$$\mathbf{v} = \underset{d \in D^{\text{MD}}}{\text{c-max}} \{\mathbf{r}_d + \mathbf{P}_d \mathbf{w}\} \text{ and } \mathbf{w}' = \mathbf{v} - v(s')\mathbf{e}$$

   is in V^F_{rel}, and

3. $\mathbf{w}^n \in V^F_{\text{rel}}$ for $n = 1, 2, \ldots$ and $\lim_{n \to \infty} \|\mathbf{w}^n - \mathbf{w}^*\| = 0$ implies $\mathbf{w}^* \in V^F_{\text{rel}}$.

Then if D^F is compatible with V^F_{rel}, there is an optimal stationary policy $(d^*)^\infty$ with $d^* \in D^F$.

Applying this result to the equipment replacement model in Section 5.10.2 is left as an exercise.

7.10 Queuing service rate control

This section focuses on numerical solutions in a infinite horizon average cost version of the discrete time service rate control model described in Section 3.4.1 and previously analyzed in the finite horizon (Section 4.3.1) and discounted (Section 5.11) cases. This section illustrates some of the challenges faced when solving average reward models.

7.10.1 The model

Suppose the controller chooses among three service probabilities $a_1 = 0.2$, $a_2 = 0.4$ and $a_3 = 0.6$ and the arrival probability is $b = 0.2$. (Note that if $b > 0.4$, the row sums of transition probabilities when using a_3 will exceed 1.) Assume further that the delay cost is $f(s) = s^2$ and the cost per period of serving at rate a_k is $m(a_k) = 5k^3$. To simplify notation set $c(s, a_k) := f(s) + m(a_k)$.

To obtain a finite state space, truncate the state space at N and assume that the transition probabilities in state N for $k = 1, 2, 3$ are given by:

$$p(j|N, a_k) = \begin{cases} a_k & j = N-1 \\ 1 - a_k & j = N. \end{cases}$$

Since every state is accessible from every other state (see Exercise 19), the model is recurrent and aperiodic (regular) so that the average reward of every policy is constant. Hence the Bellman equation for this model becomes

$$h'(s) = \min_{k=1,2,3} \begin{cases} c(0, a_k) - g + (1-b)h(0) + bh(1) & s = 0 \\ c(s, a_k) - g + a_k h(s-1) + (1-b-a_k)h(s) + bh(s+1)) & 1 \le s \le N-1 \\ c(N, a_k) - g + a_k h(N-1) + (1-a_k)h(N) & s = N. \end{cases}$$

Since $f(0) = 0$ and the choice of service rate has no impact on transition behavior in state 0, the optimal decision in this state is to choose the least expensive service rate with cost $m(a_1)$. Hence, the Bellman equation in state 0 simplifies to

$$h'(0) = m(a_1) - g + (1-b)h(0) + bh(1).$$

7.10.2 Value iteration

The beauty of this model is that as a result of the simple transition structure, when implementing value iteration there is no need to write out the entire transition matrix. Coding is simplified when value iteration is expressed in two steps as follows.

Given $v(s)$, define $q(s, a)$ by

$$q(s, a_k) = \begin{cases} c(0, a_k) + (1-b)v(0) + bv(1) & s = 0 \\ c(s, a_k) + a_k v(s-1) + (1-b-a_k)v(s) + bv(s+1) & 1 \le s \le N-1 \\ c(N, a_k) + a_k v(N-1) + (1-a_k)v(N) & s = N. \end{cases}$$

Then

$$v'(s) = \min_{k=1,2,3} q(s, a_k) \quad \text{for all } s \in S.$$

Value iteration was applied to models with $N = 50, 200, 500$, and 1,000 setting $v^0(s) = 0$ for all $s \in S$ and $\epsilon = 0.0001$. Convergence required $350, 796, 1661$, and 3055 iterations, respectively.

The left-hand figure in Figure 7.9 shows the ϵ-optimal policy when $N = 50$. Observe that it is monotone and uses action a_1 for $s = 0, 1, 2$, action a_2 when $s = 3, 4, \ldots, 8$ and action a_3 for $s \geq 9$. Moreover the same policy is optimal for all choices of N with the estimate $g^* = 19.4247$. That g does not vary with N is a result of the fact that under the optimal policy, the steady state probability that there are 10 or more jobs in the system is less than 0.001. Thus, under the optimal policy the system remains in the low-occupancy states[40] regardless of the value of N.

Because the values diverged, for example when $N = 1{,}000$, $v(1{,}000) = 834, 930, 712$ at termination, relative value iteration was applied setting $w^n(s) = v^n(s) - v^n(0)$ after each iteration. Convergence required the same number steps as above, but as noted in the text $w^n(s)$ converged. The right-hand image in Figure 7.9 depicts $w^n(s)$ at termination. Observe that it is convex and non-decreasing.

Using policy iteration the next section shows the ϵ-optimal policy found by value iteration is indeed optimal and moreover when $N = 50$, $w(s)$ and $h_{\text{rel}}(s)$ differ by at most by 0.002. Thus relative value iteration well approximates the relative value function.

Since under the optimal policy the system spends most of it time in states 0 to 9, for large s one can view the relative value function $h_{\text{rel}}(s)$ as the expected total excess cost associated with starting in state s.

7.10.3 Policy iteration

This section solves several instances of this model using policy iteration. For each problem size, iteration is initialized with the following "silly" stationary policy: use action a_1 in state 0, a_2 in state 1, a_3 in state 2 and repeat this pattern up to state N.

Policy iteration is implemented by adding the constraint $h(0) = 0$ so as to uniquely specify $h(s)$ for all $s \in S$. Consequently the evaluation equation can be expressed in matrix notation as

$$\begin{bmatrix} \mathbf{e} & \mathbf{I} - \mathbf{P}_d \\ 0 & \mathbf{e}_1^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} g \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_d \\ 0 \end{bmatrix}, \tag{7.119}$$

where $\mathbf{e}$ is a vector of length $|S|$ with all entries equal to 1 and $\mathbf{e}_1^{\mathsf{T}}$ is a row vector of length $|S|$ with a 1 in the first component (corresponding to state 0) and the remaining entries equal to zero. Note the last row of this equation corresponds to the constraint

[40] This will provide a challenge when using simulation based methods because there will be few observations in high occupancy states.

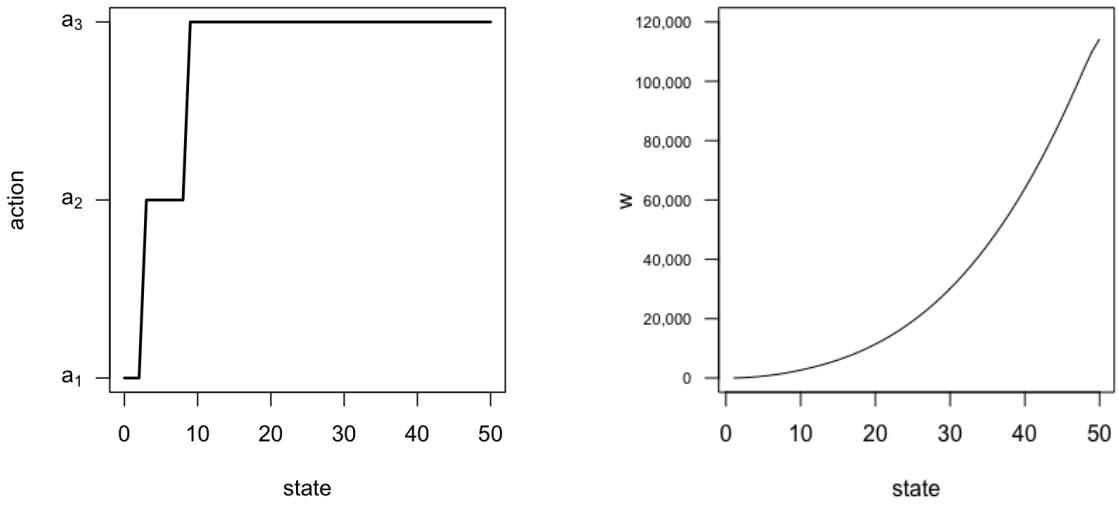


Figure 7.9: The ϵ-optimal stationary policy d^∞ (left) identified by value iteration and relative value iteration and the iterate $w^n(s)$ (right) of relative value iteration at termination for the queuing service rate control model with $N = 50$.

$h(0) = 0$[41]. As a result of this specification, the solution will give the gain and relative value function for each policy.

The model was solved for $N = 50, 200, 500$, and 1,000. It was observed that:

1. For each value of N, policy iteration required 3 iterations to satisfy the termination criterion $d' = d$. Computation time was negligible on a MacBook Pro M3. The resulting optimal policy was identical to that found using value iteration with $g^* = 19.4247$.

2. The optimal policy for this model agrees with the optimal policy for a discounted model with $\lambda = 0.999$. This observation is consistent with the logic used to prove Theorem 7.8[42].

Policy iteration was also used to solve a queuing service rate control model with 6 actions and 5,000 states. It required 4 iterations and 47.2 seconds on a MacBook Pro M3. Again the optimal policy was monotone non-decreasing in the state and the relative value function was convex non-decreasing. It used the fastest rate when $s > 22$.

[41] Expressing the matrix in this format is convenient for using the solver in R or other software.

[42] This observation is a consequence of Blackwell optimality, namely that in a finite state and action model there exists a stationary policy that is optimal for all λ sufficiently close to 1.

7.10.4 Linear programming

Formulation of the linear programming model for solution by a solver[43] requires specification of

1. a matrix representing constraints,
2. the right hand side vector, and
3. objective function coefficients.

Moreover, since this model seeks to minimize costs, the dual becomes a minimization instead of a maximization problem[44].

In a model with $S = \{0, 1, \ldots, N-1\}$, there are $3N$ dual variables so that the constraint matrix is $(N+1) \times 3N$. The rows of the matrix correspond to the N constraints involving the transition probabilities plus an additional constraint that the sum of the dual variables equals 1. Although one of the first N constraints is redundant, it is left to the solver to address.

It is convenient to order the dual variables as $x(0,a_1), \ldots, x(N-1,a_1), x(0,a_2), \ldots, x(N-1,a_2), x(0,a_3), \ldots, x(N-1,a_3)$ so the constraint matrix can be constructed by concatenating matrices corresponding to each action. Let $\mathbf{P}_{a_k}$ denote the $N \times N$ transition matrix that uses action a_k in every state. That is

$$\mathbf{P}_{a_k} = \begin{bmatrix} 1-b & b & 0 & 0 & \ldots & 0 & 0 & 0 \\ a_k & 1-b-a_k & b & 0 & \ldots & 0 & 0 & 0 \\ 0 & a_k & 1-b-a_k & b & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & \ldots & \ldots & 0 & \ldots & 0 & a_k & 1-a_k \end{bmatrix}.$$

Thus the portion of the constraint matrix representing transition probabilities can be written as

$$\left[(\mathbf{I} - \mathbf{P}_{a_1})^\mathsf{T} \quad (\mathbf{I} - \mathbf{P}_{a_2})^\mathsf{T} \quad (\mathbf{I} - \mathbf{P}_{a_3})^\mathsf{T} \right].$$

The vector of objective function coefficients is given by

$$(c(0,a_1), \ldots, c(N-1,a_1), c(0,a_2), \ldots, c(N-1,a_2), c(0,a_3), \ldots, c(N-1,a_3))$$

and the right hand side vector equals $\mathbf{e}_{N+1}$ (a vector with a 1 in the $N+1$-th component and zeroes elsewhere).

Solving the model with $N = 20$ and noting (7.112) gives the same solution as that obtained by value iteration and policy iteration. Note that when using this solution, the system use action a_3 with probability less than 0.0002. This is because most of the time the system remains in low occupancy states.

[43] Computations here used the package lpSolver in R.

[44] As always one can formulate it as a maximization problem with negative rewards.

If instead the problem is solved with arrival probability equal to 0.35, the average optimal policy is $(d^*)^\infty$ where

$$d^*(s) = \begin{cases} a_1 & s = 0, 1 \\ a_2 & s = 2, 3, 4 \\ a_3 & s = 5, 6, \ldots . \end{cases}$$

Because arrivals occur more often, the system uses higher service rates at lower occupancy levels than when $b = 0.2$. In fact the system now uses action a_3 with probability 0.195 and optimal average cost equals to 60.27.

Adding constraints

To illustrate the use of constraints, consider imposing the condition that *the system use action a_3 at most* 15% *of the time in steady state*. This corresponds to the constraint:

$$\sum_{s \in S} x(s, a_3) \leq 0.15.$$

Solving the model with this constraint results in $x(s, a_1) > 0$ for $s = 0, 1$, $x(s, a_2) > 0$ for $s = 2, \ldots, 6$ and $x(s, a_3) > 0$ for $s \geq 6$. Thus since $x(6, a_2) = 0.0158$ and $x(6, a_3) = 0.053$ the optimal policy to the constrained model differs from the above policy by randomizing action choice when $s = 6$. Using (7.112) gives

$$w_{d^*}(a_2|6) = \frac{0.016}{0.016 + 0.053} = 0.229 \quad \text{and} \quad w_{d^*}(a_3|6) = 0.771.$$

The optimal average reward to the constrained problem is 60.46, only slightly greater than that of the unconstrained solution. Since this constraint is binding, the system uses action a_3 with probability 0.15.

If instead of 15% the constraint was set at 10%, the optimal policy changes and uses action a_1 with certainty when $s = 0$, a_2 with certainty when $s = 1, \ldots, 6$ and randomizes action choice between a_2 and a_3 when $s = 7$ and chooses a_3 with certainty when $s \geq 8$. Note that $w_{d^*}(a_2|7) = 0.795$ and the optimal average cost equals 62.85.

When trying to solve larger problems (say $N = 51$) using linear programming it was challenging to identify the optimal policy because it was difficult to distinguish true zeroes from those due to underflow, even when the right hand side of the equality constraint was scaled considerably.

Thus, the main advantage of using linear programming is the ability to add constraints. However, this may be outweighed by

1. the challenge of identifying the optimal policy in moderate sized problems,
2. the work required to generate the constraint matrix, and
3. the need to have an in depth understanding of linear programming to apply it successfully.

7.10.5 Concluding remarks on computation

Current trends suggest that Markov decision process users will want to solve much larger problems than those considered in this chapter. Consequently, when a model is available, relative value iteration or policy iteration are the most attractive computational approaches. Although not considered here, modified policy iteration might be an attractive alternative because it requires less computational effort than (relative) value iteration because it avoids many unnecessary maximizations and avoids generating the transition probability matrix that is required to evaluate a policy using policy iteration. Linear programming is not recommended for most computational studies but is discussed here because of its elegant theory and historical significance. Moreover it has proved useful in approximate dynamic programming (Chapter 9).

Bibliographic remarks

Our development closely follows Puterman [1994] but omits many technical issues specifically around multi-chain models.

Howard was the first to study average reward Markov decision problems although there are apparently antecedents in the game theory literature. The monograph Howard [1960], based on his MIT doctoral dissertation, formulates the model, introduces policy iteration and observe that distinct analyses were required for unichain and multi-chain models.

Howard's work motivated the seminal paper Blackwell [1962], which studies the average reward as the limit of the discounted reward as the discount rate approaches 1 and relies heavily on the partial Laurent expansion (see Theorem 7.4). He shows that in finite state and action models multi-chain policy iteration converges and moreover there exists a stationary policy that is optimal for all discount rates sufficiently close to 1, a concept that is now referred to as Blackwell optimality. The classic book Kemeny and Snell [1960] describes the concepts of limiting and fundamental matrices that underlie Blackwell's paper.

As noted above, Howard pointed out the need for distinguishing between unichain and multi-chain models. However Bather [1973] showed that classification excludes many applications. He introduces a class of models referred to as *communicating*, which include a wide-range of applications and in which the optimal gain is constant. Platzman [1977] generalizes this concept to the class of weakly communicating models.

The proof of existence of solutions to the unichain Bellman equation uses concepts from Blackwell [1962]. Derman [1970] and Sennott [1999] provide an alternative approach.

The use of value iteration in average reward models has been extensively studied. Howard [1960] conjectured that the limit in (7.77) exists. Theorem 7.10 shows that when this limit exists, value iteration converges. However, it required many papers to establish the existence of this limit. Schweitzer and Federgruen [1977] provide a

comprehensive analysis of this problem that applies also to countable state models. The discussion of the aperiodicity transformation as proposed by Schweitzer [1971] follows Section 6.6 in Sennott [1999]. Relative value iteration is due to White [1963].

The use of linear programming in average reward models originates with Ghelinick [1960] and Manne [1960] who focus on regular models. This work was extended to more general models by Denardo and Fox [1968]. Derman [1970] studies the use of linear programming in constrained models and Kallenberg [1983] provides a comprehensive overview and development in his thesis.

Exercises

1. Consider a stationary version of the three-state model from exercise 1 of Chapter 2.

   (a) Verify that the model is regular.

   (b) Give the Bellman equation for this model.

   (c) Find an average ϵ-optimal policy using value iteration and relative value iteration with $\epsilon = 0.0001$. Compare the number of iterations each require to achieve the same degree of precision.

   (d) Find an average optimal policy using policy iteration starting with the decision rule that uses a_2 in each state.

   (e) Find an average optimal policy using linear programming.

   (f) For what values of λ does the average optimal policy agree with the discount optimal policy?

2. Consider the decision rule $d'(s_i) = a_2$ for $i = 1, 2, 3$ in the model in Problem 1. Investigate the quality of the approximation $\mathbf{v}_n \approx ng\mathbf{e}+\mathbf{h}$ and $\mathbf{v}_\lambda \approx (1-\lambda)^{-1}g\mathbf{e}+\mathbf{h}$ for the stationary policy $(d')^\infty$. Show your results graphically for each state.

3. Classify the model in Problem 1 with:

   (a) the addition of the action $a_{1,3}$ with transition probability $p(s_1|s_1, a_{1,3}) = 1$ and $r(s_1, a_{1,3}) = 2$.

   (b) the addition of the action $a_{2,3}$ with transition probability $p(s_2|s_2, a_{2,3}) = 1$ and $r(s_2, a_{2,3}) = 4$.

   Use value iteration to find a 0.0001-optimal policy for the model with the two additional actions. What is the chain structure of the transition probability matrix corresponding to an optimal policy? What is the form of the ϵ-optimal average reward?

4. Consider a Markovian decision rule with periodic transition matrix

$$\mathbf{P}_d = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{r}_d = \begin{bmatrix} a \\ b \end{bmatrix}.$$

   (a) Show numerically that (7.29) is not valid when $a \neq b$.

   (b) Derive a relationship similar to (7.29) in terms of

$$\frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{j} \mathbf{v}_i^{d^\infty}$$

   by modifying the proof of Theorem 7.5.

   (c) Show numerically that this new expression holds for $a \neq b$.

5. Show that the limit in (7.1) exists for all stationary policies in the example in Section 7.2.2.

6. (This example was suggested to us by Nicolas Gast in a personal communication.) Consider a three state deterministic model with $S = \{s_1, s_2, s_3\}$ and $A_{s_1} = \{a_{1,1}\}, A_{s_2} = \{a_{2,1}, a_{2,2}\}, A_{s_3} = \{a_{3,1}\}$ where $p(s_2|s_1, a_{1,1}) = p(s_1|s_2, a_{2,1}) = p(s_3|s_2, a_{2,2}) = p(s_2|s_3, a_{3,1}) = 1$ and $r(s_1, a_{1,1}) = 0$, $r(s_2, a_{2,1}) = 2$, $r(s_2, a_{2,2}) = 2$, $r(s_1, a_{1,1}) = 4$.

   (a) Show this model is communicating.

   (b) Show that there exists no Markovian deterministic policy that generates a recurrent Markov chain.

   (c) Find an optimal Markovian deterministic policy and show that it is unichain.

7. Prove in general that in a communicating model there exists a Markovian randomized decision rule that generates a Markov chain with a single recurrent class. (Note that since optimizing over randomized decision rules is equivalent to optimizing over deterministic decision rules, this implies the optimality of a deterministic Markovian policy.)

8. Find 0.001-optimal and optimal policies for the replacement model from Exercise 4 from Chapter 5. Compare your results to those in the discounted case.

9. Find 0.001-optimal and optimal policies for the inventory model in Exercise 3 from Chapter 5. Compare your results to those in the discounted case.

10. Find an average optimal policy for an infinite horizon average reward variant of the Gridworld navigation problem of Section 3.2 in which after delivering the coffee (cell 1) or falling down the stairs (cell 7), the robot begins the subsequent decision epoch in the coffee room (cell 13). Assume the system starts in the

coffee room. Choose appropriate values for the parameters. How does the optimal policy for this model compare with the optimal solution to the single pass problem?

11. Consider the following **deterministic** stationary model formulated in Table 7.7:

State	Action	Next state	Reward
s_1	$a_{1,1}$	s_3	4
s_1	$a_{1,2}$	s_2	2
s_2	$a_{2,1}$	s_3	3
s_3	$a_{3,1}$	s_3	0

Table 7.7: Description of transitions and rewards for Exercise 11.

(a) Depict the model graphically and classify its states.

(b) Find all policies that maximize the long-run average reward. What does this imply about the long-run average reward criterion?

(c) Find all policies that maximize the expected total reward.

(d) Find all policies that maximize the discounted reward for λ close to 1.

(e) Suppose action $a_{3,1}$ is replaced by a zero-reward transition to s_1. Find all policies that maximize the average reward for this modified problem. How is this policy related to those above?

(f) How does this observation apply to the previous problem?

12. Consider an admission control queuing model (Section 3.4.2) with finite buffer in which at the start of each period a job arrives with certainty and is processed in that period with probability p and not processed with probability $1-p$. Assume further that when the buffer is full, no jobs arrive.

(a) Show that there exist stationary policies with two or more closed classes.

(b) Show that the model is communicating.

13. Consider an inventory model with a capacity of 3 units in which the probability that the demand in any period is 1 unit equals p and the probability demand is 0 equals $1-p$. Assume unfilled demand is lost so that $S = \{0, 1, 2, 3\}$.

(a) Show that there exists a stationary deterministic policy that has a transition probability matrix with two closed classes so that the model is multi-chain.

(b) Show that the model is communicating.

(c) What algorithm would you use to find an optimal policy?

14. Provide an example of a weakly communicating Markov decision process and prove that the optimal reward in a weakly communicating model is constant.

15. In Example 7.5, show that g and $\mathbf{h}$ as given in the example satisfy the optimality equation.

16. Verify all calculations in Example 7.7. Show that when $r(s_2, a_{2,1}, s_2) = 0$ that the gain and bias of both stationary policies satisfies the Bellman equation.

17. Consider a deterministic decision rule $d \in D^{\text{MD}}$ with

$$\mathbf{r}_d = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \text{and} \quad \mathbf{P}_d = \begin{bmatrix} 0 & 1/2 & 1/2 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

    (a) Verify that $\mathbf{P}_d$ is periodic. What is its period?

    (b) Find g^{d^∞} and $\mathbf{h}^{d^\infty}$ by solving (7.43).

    (c) Show through calculations that the result in Corollary 7.5 holds.

    (d) Transform $\mathbf{P}_d$ to $\tilde{\mathbf{P}}_d$ using aperiodicity transformation (7.73).

    (e) Provide a graphical representation of the transformed model.

    (f) Compare the eigenvalues of $\mathbf{P}_d$ to those of $\tilde{\mathbf{P}}_d$ when $\tau = 0.05$. What difference do you observe?

18. Verify all calculations and statements in Example 7.10.

19. Show that the queuing service rate control model from Section 7.10 is communicating.

20. Prove that an optimal policy of the equipment replacement model is a control limit policy.

21. Show by induction on n or direct expansion of the left hand side that for $n \geq 1$, $\mathbf{P}_d^n - \mathbf{P}_d^* = (\mathbf{P}_d - \mathbf{P}_d^*)^n$.

22. Prove that $\mathbf{P}_d^* = \tilde{\mathbf{P}}_d^*$ where $\tilde{\mathbf{P}}_d^*$ is defined by (7.73).

23. Assuming that (7.77) is valid, show that all parts of Theorem 7.10 hold.

24. Write out one step of relative value iteration for a fixed decision rule in the form $\mathbf{w}' = \mathbf{Q}\mathbf{v}'$ where $\mathbf{Q}$ is a rank-one matrix that subtracts the first component from all other elements of $\mathbf{v}'$ where $\mathbf{v}' = \mathbf{r}_d + \mathbf{P}_d\mathbf{w}$.

    (a) Show by example that when $\mathbf{P}_d$ is regular, the largest eigenvalue of $\mathbf{Q}\mathbf{P}_d$ is strictly less than one.

    (b) Prove this result.

25. Develop, apply and demonstrate the convergence of Gauss-Seidel and modified policy iteration algorithms for unichain average reward models.

26. Create an alternative non-stationary policy to the one analyzed in Section 7.2.2 that has a different lim inf and lim sup average rewards.