2
Contractive Models

Contents

In this chapter we consider the abstract DP model of Section 1.2 under the most favorable assumptions: monotonicity and weighted sup-norm contraction. Important special cases of this model are the discounted problems with bounded cost per stage (Example 1.2.1-1.2.5), the stochastic shortest path problem of Example 1.2.6 in the case where all policies are proper, as well as other problems involving special structures.

We first provide some basic analytical results and then focus on two types of algorithms: *value iteration and policy iteration*. In addition to exact forms of these algorithms, we discuss combinations and approximate versions, as well as asynchronous distributed versions.

2.1 BELLMAN'S EQUATION AND OPTIMALITY CONDITIONS

In this section we recall the abstract DP model of Section 1.2, and derive some of its basic properties under the monotonicity and contraction assumptions of Section 1.3. We consider a set X of states and a set U of controls, and for each $x \in X$, a nonempty control constraint set $U(x) \subset U$. We denote by $\mathcal{M}$ the set of all functions $\mu : X \mapsto U$ with $\mu(x) \in U(x)$ for all $x \in X$, which we refer to as *policies* (or "stationary policies," when we want to emphasize the distinction from nonstationary policies, to be discussed later).

We denote by $\mathcal{R}(X)$ the set of real-valued functions $J : X \mapsto \Re$. We have a mapping $H : X \times U \times \mathcal{R}(X) \mapsto \Re$ and for each policy $\mu \in \mathcal{M}$, we consider the mapping $T_\mu : \mathcal{R}(X) \mapsto \mathcal{R}(X)$ defined by

$$(T_\mu J)(x) = H\big(x, \mu(x), J\big), \qquad \forall\ x \in X.$$

We also consider the mapping T defined by

$$(TJ)(x) = \inf_{u \in U(x)} H(x, u, J) = \inf_{\mu \in \mathcal{M}} (T_\mu J)(x), \qquad \forall\ x \in X.$$

[We will use frequently the second equality above, which holds because $\mathcal{M}$ can be viewed as the Cartesian product $\Pi_{x \in X} U(x)$.] We want to find a function $J^* \in \mathcal{R}(X)$ such that

$$J^*(x) = \inf_{u \in U(x)} H(x, u, J^*), \qquad \forall\ x \in X,$$

i.e., to find a fixed point of T within $\mathcal{R}(X)$. We also want to obtain a policy $\mu^* \in \mathcal{M}$ such that $T_{\mu^*} J^* = TJ^*$.

Let us restate for convenience the contraction and monotonicity assumptions of Section 1.2.2.

Assumption 2.1.1: (Monotonicity) If $J, J' \in \mathcal{R}(X)$ and $J \leq J'$, then

$$H(x, u, J) \leq H(x, u, J'), \qquad \forall\ x \in X,\ u \in U(x).$$

Note that the monotonicity assumption implies the following properties, for all $J, J' \in \mathcal{R}(X)$ and $k = 0, 1, \ldots$, which we will use extensively:

$$J \leq J' \qquad \Rightarrow \qquad T^k J \leq T^k J', \qquad T_\mu^k J \leq T_\mu^k J', \quad \forall\ \mu \in \mathcal{M},$$

$$J \leq TJ \qquad \Rightarrow \qquad T^k J \leq T^{k+1} J, \qquad T_\mu^k J \leq T_\mu^{k+1} J, \quad \forall\ \mu \in \mathcal{M}.$$

Here T^k and T_μ^k denotes the k-fold composition of T and T_μ, respectively.

For the contraction assumption, we introduce a function $v : X \mapsto \Re$ with

$$v(x) > 0, \qquad \forall\ x \in X.$$

We consider the weighted sup-norm

$$\|J\| = \sup_{x \in X} \frac{\big|J(x)\big|}{v(x)}$$

on $\mathcal{B}(X)$, the space of real-valued functions J on X such that $J(x)/v(x)$ is bounded over $x \in X$ (see Appendix B for a discussion of the properties of this space).

Assumption 2.1.2: (Contraction) For all $J \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$, the functions $T_\mu J$ and TJ belong to $\mathcal{B}(X)$. Furthermore, for some $\alpha \in (0, 1)$, we have

$$\|T_\mu J - T_\mu J'\| \leq \alpha \|J - J'\|, \qquad \forall\ J, J' \in \mathcal{B}(X),\ \mu \in \mathcal{M}.$$

The classical DP models where both the monotonicity and contraction assumptions are satisfied are the discounted finite-state Markovian decision problem of Example 1.2.2, and the stochastic shortest path problem of Example 1.2.6 in the special case where all policies are proper; see the textbook [Ber12a] for an extensive discussion. In the context of these problems, the fixed point equation $J = TJ$ is called *Bellman's equation*, a term that we will use more generally in this book as well. The following proposition summarizes some of the basic consequences of the contraction assumption.

Proposition 2.1.1: Let the contraction Assumption 2.1.2 hold. Then:

(a) The mappings T_μ and T are contraction mappings with modulus α over $\mathcal{B}(X)$, and have unique fixed points in $\mathcal{B}(X)$, denoted J_μ and J^*, respectively.

(b) For any $J \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$,

$$\lim_{k\to\infty} \|J^* - T^k J\| = 0, \qquad \lim_{k\to\infty} \|J_\mu - T_\mu^k J\| = 0.$$

(c) We have $T_\mu J^* = TJ^*$ if and only if $J_\mu = J^*$.

(d) For any $J \in \mathcal{B}(X)$,

$$\|J^* - J\| \leq \frac{1}{1-\alpha}\|TJ - J\|, \qquad \|J^* - TJ\| \leq \frac{\alpha}{1-\alpha}\|TJ - J\|.$$

(e) For any $J \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$,

$$\|J_\mu - J\| \leq \frac{1}{1-\alpha}\|T_\mu J - J\|, \qquad \|J_\mu - T_\mu J\| \leq \frac{\alpha}{1-\alpha}\|T_\mu J - J\|.$$

Proof: We showed in Section 1.2.2 that T is a contraction with modulus α over $\mathcal{B}(X)$. Parts (a) and (b) follow from Prop. B.1 of Appendix B.

To show part (c), note that if $T_\mu J^* = TJ^*$, then in view of $TJ^* = J^*$, we have $T_\mu J^* = J^*$, which implies that $J^* = J_\mu$, since J_μ is the unique fixed point of T_μ. Conversely, if $J^* = J_\mu$, we have $T_\mu J^* = T_\mu J_\mu = J_\mu = J^* = TJ^*$.

To show part (d), we use the triangle inequality to write for every k,

$$\|T^k J - J\| \leq \sum_{\ell=1}^{k} \|T^\ell J - T^{\ell-1} J\| \leq \sum_{\ell=1}^{k} \alpha^{\ell-1}\|TJ - J\|.$$

Taking the limit as $k \to \infty$ and using part (b), the left-hand side inequality follows. The right-hand side inequality follows from the left-hand side and the contraction property of T. The proof of part (e) is similar to part (d) [indeed it is the special case of part (d) where T is equal to T_μ, i.e., when $U(x) = \{\mu(x)\}$ for all $x \in X$]. **Q.E.D.**

Part (c) of the preceding proposition shows that there exists a $\mu \in \mathcal{M}$ such that $J_\mu = J^*$ if and only if the minimum of $H(x, u, J^*)$ over $U(x)$ is attained for all $x \in X$. Of course the minimum is attained if $U(x)$ is finite for every x, but otherwise this is not guaranteed in the absence of additional assumptions. Part (d) provides a useful error bound: we can evaluate the proximity of any function $J \in \mathcal{B}(X)$ to the fixed point J^* by applying T to J and computing $\|TJ - J\|$. The left-hand side inequality of part (e) (with $J = J^*$) shows that for every $\epsilon > 0$, there exists a $\mu_\epsilon \in \mathcal{M}$ such that $\|J_{\mu_\epsilon} - J^*\| \leq \epsilon$, which may be obtained by letting $\mu_\epsilon(x)$ minimize $H(x, u, J^*)$ over $U(x)$ within an error of $(1-\alpha)\epsilon\, v(x)$, for all $x \in X$.

The preceding proposition and some of the subsequent results may also be proved if $\mathcal{B}(X)$ is replaced by a closed subset $\overline{\mathcal{B}}(X) \subset \mathcal{B}(X)$. This is because the contraction mapping fixed point theorem (Prop. B.1) applies to closed subsets of complete spaces. For simplicity, however, we will disregard this possibility in the present chapter.

An important consequence of monotonicity of H, when it holds in addition to contraction, is that it implies that J^*, the unique fixed point of T, is the infimum over $\mu \in \mathcal{M}$ of J_μ, the unique fixed point of T_μ.

Proposition 2.1.2: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold. Then

$$J^*(x) = \inf_{\mu \in \mathcal{M}} J_\mu(x), \qquad \forall\ x \in X.$$

Furthermore, for every $\epsilon > 0$, there exists $\mu_\epsilon \in \mathcal{M}$ such that

$$J^*(x) \leq J_{\mu_\epsilon}(x) \leq J^*(x) + \epsilon, \qquad \forall\ x \in X. \tag{2.1}$$

Proof: We note that the right-hand side of Eq. (2.1) holds by Prop. 2.1.1(e) (see the remark following its proof). Thus $\inf_{\mu \in \mathcal{M}} J_\mu(x) \leq J^*(x)$ for all $x \in X$. To show the reverse inequality as well as the left-hand side of Eq. (2.1), we note that for all $\mu \in \mathcal{M}$, we have $TJ^* \leq T_\mu J^*$, and since $J^* = TJ^*$, it follows that $J^* \leq T_\mu J^*$. By applying repeatedly T_μ to both sides of this inequality and by using the monotonicity Assumption 2.1.1, we obtain $J^* \leq T_\mu^k J^*$ for all $k > 0$. Taking the limit as $k \to \infty$, we see that $J^* \leq J_\mu$ for all $\mu \in \mathcal{M}$, so that $J^*(x) \leq \inf_{\mu \in \mathcal{M}} J_\mu(x)$ for all $x \in X$. **Q.E.D.**

Note that without monotonicity, we may have $\inf_{\mu \in \mathcal{M}} J_\mu(x) < J^*(x)$ for some x. This is illustrated by the following example.

Example 2.1.1 (Counterexample Without Monotonicity)

Let $X = \{x_1, x_2\}$, $U = \{u_1, u_2\}$, and let

$$H(x_1, u, J) = \begin{cases} -\alpha J(x_2) & \text{if } u = u_1, \\ -1 + \alpha J(x_1) & \text{if } u = u_2, \end{cases} \quad H(x_2, u, J) = \begin{cases} 0 & \text{if } u = u_1, \\ B & \text{if } u = u_2, \end{cases}$$

where B is a positive scalar. Then it can be seen that

$$J^*(x_1) = -\frac{1}{1-\alpha}, \qquad J^*(x_2) = 0,$$

and $J_{\mu^*} = J^*$ where $\mu^*(x_1) = u_2$ and $\mu^*(x_2) = u_1$. On the other hand, for $\mu(x_1) = u_1$ and $\mu(x_2) = u_2$, we have $J_\mu(x_1) = -\alpha B$ and $J_\mu(x_2) = B$, so $J_\mu(x_1) < J^*(x_1)$ for B sufficiently large.

Optimality over Nonstationary Policies

The connection with DP motivates us to consider the set Π of all sequences $\pi = \{\mu_0, \mu_1, \ldots\}$ with $\mu_k \in \mathcal{M}$ for all k (nonstationary policies in the DP context), and define

$$J_\pi(x) = \limsup_{k\to\infty} (T_{\mu_0} \cdots T_{\mu_k} \bar{J})(x), \qquad \forall\ x \in X,$$

with $\bar{J}$ being some function in $\mathcal{B}(X)$, where $T_{\mu_0} \cdots T_{\mu_k} J$ denotes the composition of the mappings $T_{\mu_0}, \ldots, T_{\mu_k}$ applied to J, i.e.,

$$T_{\mu_0} \cdots T_{\mu_k} J = T_{\mu_0}\big(T_{\mu_1} \cdots (T_{\mu_{k-1}}(T_{\mu_k} J)) \cdots\big).$$

Note that under the contraction Assumption 2.1.2, *the choice of $\bar{J}$ in the definition of J_π does not matter*, since for any two $J, J' \in \mathcal{B}(X)$, we have

$$\|T_{\mu_0} T_{\mu_1} \cdots T_{\mu_k} J - T_{\mu_0} T_{\mu_1} \cdots T_{\mu_k} J'\| \le \alpha^{k+1} \|J - J'\|,$$

so the value of $J_\pi(x)$ is independent of $\bar{J}$. Since by Prop. 2.1.1(b), $J_\mu(x) = \lim_{k\to\infty} (T_\mu^k J)(x)$ for all $\mu \in \mathcal{M}$, $J \in \mathcal{B}(X)$, and $x \in X$, in the DP context we recognize J_μ as the cost function of the stationary policy $\{\mu, \mu, \ldots\}$.

We now claim that under the monotonicity and contraction Assumptions 2.1.1 and 2.1.2, *J^*, which was defined as the unique fixed point of T, is equal to the optimal value of J_π*, i.e.,

$$J^*(x) = \inf_{\pi\in\Pi} J_\pi(x), \qquad \forall\ x \in X.$$

Indeed, since $\mathcal{M}$ defines a subset of Π, we have from Prop. 2.1.2,

$$J^*(x) = \inf_{\mu\in\mathcal{M}} J_\mu(x) \ge \inf_{\pi\in\Pi} J_\pi(x), \qquad \forall\ x \in X,$$

while for every $\pi \in \Pi$ and $x \in X$, we have

$$J_\pi(x) = \limsup_{k\to\infty} (T_{\mu_0} T_{\mu_1} \cdots T_{\mu_k} \bar{J})(x) \ge \lim_{k\to\infty} (T^{k+1} \bar{J})(x) = J^*(x)$$

[the monotonicity Assumption 2.1.1 can be used to show that

$$T_{\mu_0} T_{\mu_1} \cdots T_{\mu_k} \bar{J} \ge T^{k+1} \bar{J},$$

and the last equality holds by Prop. 2.1.1(b)]. Combining the preceding relations, we obtain $J^*(x) = \inf_{\pi\in\Pi} J_\pi(x)$.

Thus, in DP terms, *we may view J^* as an optimal cost function over all policies*, including nonstationary ones. At the same time, Prop. 2.1.2 states that stationary policies are sufficient in the sense that *the optimal cost can be attained to within arbitrary accuracy with a stationary policy* [uniformly for all $x \in X$, as Eq. (2.1) shows].

Error Bounds and Other Inequalities

The analysis of abstract DP algorithms and related approximations requires the use of some basic inequalities that follow from the assumptions of contraction and monotonicity. We have obtained two such results in Prop. 2.1.1(d),(e), which assume only the contraction assumption. These results can be strengthened if in addition to contraction, we have monotonicity. To this end we first show the following useful characterization.

Proposition 2.1.3: The monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold if and only if for all $J, J' \in \mathcal{B}(X)$, $\mu \in \mathcal{M}$, and scalar $c \geq 0$, we have

$$J \leq J' + c\,v \quad \Rightarrow \quad T_\mu J \leq T_\mu J' + \alpha c\,v, \tag{2.2}$$

where v is the weight function of the weighted sup-norm $\|\cdot\|$.

Proof: Let the contraction and monotonicity assumptions hold. If $J' \leq J + c\,v$, we have

$$H(x,u,J') \leq H(x,u,J+c\,v) \leq H(x,u,J) + \alpha c\,v(x), \quad \forall\, x \in X,\, u \in U(x), \tag{2.3}$$

where the left-side inequality follows from the monotonicity assumption and the right-side inequality follows from the contraction assumption, which together with $\|v\| = 1$, implies that

$$\frac{H(x,u,J+c\,v) - H(x,u,J)}{v(x)} \leq \alpha \|J + c\,v - J\| = \alpha c.$$

The condition (2.3) implies the desired condition (2.2). Conversely, condition (2.2) for $c = 0$ yields the monotonicity assumption, while for $c = \|J' - J\|$ it yields the contraction assumption. **Q.E.D.**

We can now derive the following useful variant of Prop. 2.1.1(d),(e), which involves one-sided inequalities. This variant will be used in the derivation of error bounds for various computational methods.

Proposition 2.1.4: (Error Bounds Under Contraction and Monotonicity) Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold. Then:

(a) For any $J \in \mathcal{B}(X)$ and $c \geq 0$, we have

$$TJ \leq J + c\,v \quad \Rightarrow \quad J^* \leq J + \frac{c}{1-\alpha}v,$$

$$J \leq TJ + c\,v \quad \Rightarrow \quad J \leq J^* + \frac{c}{1-\alpha}v.$$

(b) For any $J \in \mathcal{B}(X)$, $\mu \in \mathcal{M}$, and $c \geq 0$, we have

$$T_\mu J \leq J + c\,v \quad \Rightarrow \quad J_\mu \leq J + \frac{c}{1-\alpha}v,$$

$$J \leq T_\mu J + c\,v \quad \Rightarrow \quad J \leq J_\mu + \frac{c}{1-\alpha}v.$$

(c) For all $J \in \mathcal{B}(X)$, $c \geq 0$, and $k = 0, 1, \ldots$, we have

$$TJ \leq J + c\,v \qquad \Rightarrow \qquad J^* \leq T^k J + \frac{\alpha^k c}{1-\alpha}v,$$

$$J \leq TJ + c\,v \qquad \Rightarrow \qquad T^k J \leq J^* + \frac{\alpha^k c}{1-\alpha}v.$$

Proof: (a) We show the first relation. Applying Eq. (2.2) with J' and J replaced by J and TJ, respectively, and taking infimum over $\mu \in \mathcal{M}$, we see that if $TJ \leq J + c\,v$, then $T^2 J \leq TJ + \alpha c\,v$. Proceeding similarly, it follows that

$$T^\ell J \leq T^{\ell-1} J + \alpha^{\ell-1} c\,v.$$

We now write for every k,

$$T^k J - J = \sum_{\ell=1}^{k} (T^\ell J - T^{\ell-1} J) \leq \sum_{\ell=1}^{k} \alpha^{\ell-1} c\,v,$$

from which, by taking the limit as $k \to \infty$, we obtain $J^* \leq J + \big(c/(1-\alpha)\big)v$. The second relation follows similarly.

(b) This part is the special case of part (a) where T is equal to T_μ.

(c) We show the first relation. From part (a), the inequality $TJ \leq J + c\,v$ implies that

$$J^* \leq J + \frac{c}{1-\alpha}v.$$

Applying T^k to both sides of this inequality, and using the monotonicity and fixed point property of T^k, we have

$$J^* \leq T^k \left(J + \frac{c}{1-\alpha} v \right).$$

Using Eq. (2.2) with T_μ and α replaced by T^k and α^k, respectively, we obtain

$$T^k \left(J + \frac{c}{1-\alpha} v \right) \leq T^k J + \frac{\alpha^k c}{1-\alpha} v,$$

and the first relation to be shown follows from the preceding two relations. The second relation follows similarly. **Q.E.D.**

2.2 LIMITED LOOKAHEAD POLICIES

In this section, we discuss a basic building block in the algorithmic methodology of abstract DP. Given some function $\tilde{J}$ that approximates J^*, we obtain a policy by solving a finite-horizon problem where $\tilde{J}$ is the terminal cost function. The simplest possibility is a *one-step lookahead policy* $\overline{\mu}$ defined by

$$\overline{\mu}(x) \in \arg\min_{u \in U(x)} H(x, u, \tilde{J}), \qquad x \in X. \tag{2.4}$$

The following proposition gives some bounds for its performance.

Proposition 2.2.1: (One-Step Lookahead Error Bounds) Let the contraction Assumption 2.1.2 hold, and let $\overline{\mu}$ be a one-step lookahead policy obtained by minimization in Eq. (2.4), i.e., satisfying $T_{\overline{\mu}} \tilde{J} = T\tilde{J}$. Then

$$\|J_{\overline{\mu}} - T\tilde{J}\| \leq \frac{\alpha}{1-\alpha} \|T\tilde{J} - \tilde{J}\|, \tag{2.5}$$

where $\|\cdot\|$ denotes the weighted sup-norm. Moreover

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{2\alpha}{1-\alpha} \|\tilde{J} - J^*\|, \tag{2.6}$$

and

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{2}{1-\alpha} \|T\tilde{J} - \tilde{J}\|. \tag{2.7}$$

Proof: Equation (2.5) follows from the second relation of Prop. 2.1.1(e) with $J = \tilde{J}$. Also from the first relation of Prop. 2.1.1(e) with $J = J^*$, we

have

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{1}{1-\alpha}\|T_{\overline{\mu}}J^* - J^*\|.$$

By using the triangle inequality, and the relations $T_{\overline{\mu}}\tilde{J} = T\tilde{J}$ and $J^* = TJ^*$, we obtain

$$\begin{aligned}\|T_{\overline{\mu}}J^* - J^*\| &\leq \|T_{\overline{\mu}}J^* - T_{\overline{\mu}}\tilde{J}\| + \|T_{\overline{\mu}}\tilde{J} - T\tilde{J}\| + \|T\tilde{J} - J^*\| \\ &= \|T_{\overline{\mu}}J^* - T_{\overline{\mu}}\tilde{J}\| + \|T\tilde{J} - TJ^*\| \\ &\leq \alpha\|J^* - \tilde{J}\| + \alpha\|\tilde{J} - J^*\| \\ &= 2\alpha\,\|\tilde{J} - J^*\|,\end{aligned}$$

and Eq. (2.6) follows by combining the preceding two relations.

Also, from the first relation of Prop. 2.1.1(d) with $J = \tilde{J}$,

$$\|J^* - \tilde{J}\| \leq \frac{1}{1-\alpha}\|T\tilde{J} - \tilde{J}\|. \tag{2.8}$$

Thus

$$\begin{aligned}\|J_{\overline{\mu}} - J^*\| &\leq \|J_{\overline{\mu}} - T\tilde{J}\| + \|T\tilde{J} - \tilde{J}\| + \|\tilde{J} - J^*\| \\ &\leq \frac{\alpha}{1-\alpha}\|T\tilde{J} - \tilde{J}\| + \|T\tilde{J} - \tilde{J}\| + \frac{1}{1-\alpha}\|T\tilde{J} - \tilde{J}\| \\ &= \frac{2}{1-\alpha}\|T\tilde{J} - \tilde{J}\|,\end{aligned}$$

where the second inequality follows from Eqs. (2.5) and (2.8). This proves Eq. (2.7). **Q.E.D.**

Equation (2.5) provides a computable bound on the cost function $J_{\overline{\mu}}$ of the one-step lookahead policy. The bound (2.6) says that if the one-step lookahead approximation $\tilde{J}$ is within ϵ of the optimal, the performance of the one-step lookahead policy is within $2\alpha\epsilon/(1-\alpha)$ of the optimal. Unfortunately, this is not very reassuring when α is close to 1, in which case the error bound is large relative to ϵ. Nonetheless, the following example from [BeT96], Section 6.1.1, shows that this bound is tight in the sense that for any $\alpha < 1$, there is a problem with just two states where the error bound is satisfied with equality. What is happening is that an $O(\epsilon)$ difference in single stage cost between two controls can generate an $O\big(\epsilon/(1-\alpha)\big)$ difference in policy costs, yet it can be "nullified" in the fixed point equation $J^* = TJ^*$ by an $O(\epsilon)$ difference between J^* and $\tilde{J}$.

Example 2.2.1

Consider a discounted optimal control problem with two states, 1 and 2, and deterministic transitions. State 2 is absorbing, but at state 1 there are two possible decisions: move to state 2 (policy μ^*) or stay at state 1 (policy μ). The cost of each transition is 0 except for the transition from 1 to itself under

policy μ, which has cost $2\alpha\epsilon$, where ϵ is a positive scalar and $\alpha \in [0, 1)$ is the discount factor. The optimal policy μ^* is to move from state 1 to state 2, and the optimal cost-to-go function is $J^*(1) = J^*(2) = 0$. Consider the vector $\tilde{J}$ with $\tilde{J}(1) = -\epsilon$ and $\tilde{J}(2) = \epsilon$, so that

$$\|\tilde{J} - J^*\| = \epsilon,$$

as assumed in Eq. (2.6) (cf. Prop. 2.2.1). The policy μ that decides to stay at state 1 is a one-step lookahead policy based on $\tilde{J}$, because

$$2\alpha\epsilon + \alpha\tilde{J}(1) = \alpha\epsilon = 0 + \alpha\tilde{J}(2).$$

We have

$$J_\mu(1) = \frac{2\alpha\epsilon}{1-\alpha} = \frac{2\alpha}{1-\alpha}\|\tilde{J} - J^*\|,$$

so the bound of Eq. (2.6) holds with equality.

Multistep Lookahead Policies with Approximations

Let us now consider a more general form of lookahead involving multiple stages as well as other approximations of the type that we will consider later in the implementation of various approximate value and policy iteration algorithms. In particular, we will assume that given any $J \in \mathcal{B}(X)$, we cannot compute exactly TJ, but instead we can compute $\tilde{J} \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$ such that

$$\|\tilde{J} - TJ\| \leq \delta, \qquad \|T_\mu J - TJ\| \leq \epsilon, \tag{2.9}$$

where δ and ϵ are nonnegative scalars. These scalars are usually unknown, so the resulting analysis will have a mostly qualitative character.

The case $\delta > 0$ arises when the state space is either infinite or it is finite but very large. Then instead of calculating $(TJ)(x)$ for all states x, one may do so only for some states and estimate $(TJ)(x)$ for the remaining states x by some form of interpolation. Alternatively, one may use simulation data [e.g., noisy values of $(TJ)(x)$ for some or all x] and some kind of least-squares error fit of $(TJ)(x)$ with a function from a suitable parametric class. The function $\tilde{J}$ thus obtained will satisfy $\|\tilde{J} - TJ\| \leq \delta$ with $\delta > 0$. Note that δ may not be small in this context, and the resulting performance degradation may be a primary concern.

Cases where $\epsilon > 0$ may arise when the control space is infinite or finite but large, and the minimization involved in the calculation of $(TJ)(x)$ cannot be done exactly. Note, however, that it is possible that

$$\delta > 0, \qquad \epsilon = 0,$$

and in fact this occurs often in practice. In an alternative scenario, we may first obtain the policy μ subject to a restriction that it belongs to a certain subset of structured policies, so it satisfies

$$\|T_\mu J - TJ\| \leq \epsilon$$

for some $\epsilon > 0$, and then we may set $\tilde{J} = T_\mu J$. In this case we have $\epsilon = \delta$ in Eq. (2.9).

In a multistep method with approximations, we are given a positive integer m and a lookahead function J_m, and we successively compute (backwards in time) $J_{m-1}, \ldots, J_0$ and policies $\mu_{m-1}, \ldots, \mu_0$ satisfying

$$\|J_k - TJ_{k+1}\| \leq \delta, \quad \|T_{\mu_k} J_{k+1} - TJ_{k+1}\| \leq \epsilon, \qquad k = 0, \ldots, m-1. \tag{2.10}$$

Note that in the context of MDP, J_k can be viewed as an approximation to the optimal cost function of an $(m-k)$-stage problem with terminal cost function J_m. We have the following proposition.

Proposition 2.2.2: (Multistep Lookahead Error Bound) Let the contraction Assumption 2.1.2 hold. The periodic policy

$$\pi = \{\mu_0, \ldots, \mu_{m-1}, \mu_0, \ldots, \mu_{m-1}, \ldots\}$$

generated by the method of Eq. (2.10) satisfies

$$\|J_\pi - J^*\| \leq \frac{2\alpha^m}{1-\alpha^m}\|J_m - J^*\| + \frac{\epsilon}{1-\alpha^m} + \frac{\alpha(\epsilon + 2\delta)(1-\alpha^{m-1})}{(1-\alpha)(1-\alpha^m)}. \tag{2.11}$$

Proof: Using the triangle inequality, Eq. (2.10), and the contraction property of T, we have for all k

$$\begin{aligned} \|J_{m-k} - T^k J_m\| &\leq \|J_{m-k} - TJ_{m-k+1}\| + \|TJ_{m-k+1} - T^2 J_{m-k+2}\| \\ &\quad + \cdots + \|T^{k-1} J_{m-1} - T^k J_m\| \\ &\leq \delta + \alpha\delta + \cdots + \alpha^{k-1}\delta, \end{aligned} \tag{2.12}$$

showing that

$$\|J_{m-k} - T^k J_m\| \leq \frac{\delta(1-\alpha^k)}{1-\alpha}, \qquad k = 1, \ldots, m. \tag{2.13}$$

From Eq. (2.10), we have $\|J_k - T_{\mu_k} J_{k+1}\| \leq \delta + \epsilon$, so for all k

$$\begin{aligned}\|J_{m-k} - T_{\mu_{m-k}} \cdots T_{\mu_{m-1}} J_m\| &\leq \|J_{m-k} - T_{\mu_{m-k}} J_{m-k+1}\| \\ &\quad + \|T_{\mu_{m-k}} J_{m-k+1} - T_{\mu_{m-k}} T_{\mu_{m-k+1}} J_{m-k+2}\| \\ &\quad + \cdots \\ &\quad + \|T_{\mu_{m-k}} \cdots T_{\mu_{m-2}} J_{m-1} - T_{\mu_{m-k}} \cdots T_{\mu_{m-1}} J_m\| \\ &\leq (\delta + \epsilon) + \alpha(\delta + \epsilon) + \cdots + \alpha^{k-1}(\delta + \epsilon),\end{aligned}$$

showing that

$$\|J_{m-k} - T_{\mu_{m-k}} \cdots T_{\mu_{m-1}} J_m\| \leq \frac{(\delta + \epsilon)(1 - \alpha^k)}{1 - \alpha}, \qquad k = 1, \ldots, m. \tag{2.14}$$

Using the fact $\|T_{\mu_0} J_1 - T J_1\| \leq \epsilon$ [cf. Eq. (2.10)], we obtain

$$\begin{aligned}\|T_{\mu_0} \cdots T_{\mu_{m-1}} J_m - T^m J_m\| &\leq \|T_{\mu_0} \cdots T_{\mu_{m-1}} J_m - T_{\mu_0} J_1\| \\ &\quad + \|T_{\mu_0} J_1 - T J_1\| + \|T J_1 - T^m J_m\| \\ &\leq \alpha \|T_{\mu_1} \cdots T_{\mu_{m-1}} J_m - J_1\| + \epsilon + \alpha \|J_1 - T^{m-1} J_m\| \\ &\leq \epsilon + \frac{\alpha(\epsilon + 2\delta)(1 - \alpha^{m-1})}{1 - \alpha},\end{aligned}$$

where the last inequality follows from Eqs. (2.13) and (2.14) for $k = m - 1$.

From this relation and the fact that $T_{\mu_0} \cdots T_{\mu_{m-1}}$ and T^m are contractions with modulus α^m, we obtain

$$\begin{aligned}\|T_{\mu_0} \cdots T_{\mu_{m-1}} J^* - J^*\| &\leq \|T_{\mu_0} \cdots T_{\mu_{m-1}} J^* - T_{\mu_0} \cdots T_{\mu_{m-1}} J_m\| \\ &\quad + \|T_{\mu_0} \cdots T_{\mu_{m-1}} J_m - T^m J_m\| + \|T^m J_m - J^*\| \\ &\leq 2\alpha^m \|J^* - J_m\| + \epsilon + \frac{\alpha(\epsilon + 2\delta)(1 - \alpha^{m-1})}{1 - \alpha}.\end{aligned}$$

We also have using Prop. 2.1.1(e), applied in the context of the multistep mapping of Example 1.3.1,

$$\|J_\pi - J^*\| \leq \frac{1}{1 - \alpha^m} \|T_{\mu_0} \cdots T_{\mu_{m-1}} J^* - J^*\|.$$

Combining the last two relations, we obtain the desired result. **Q.E.D.**

Note that for $m = 1$ and $\delta = \epsilon = 0$, i.e., the case of one-step lookahead policy $\overline{\mu}$ with lookahead function J_1 and no approximation error in the minimization involved in TJ_1, Eq. (2.11) yields the bound

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{2\alpha}{1 - \alpha} \|J_1 - J^*\|,$$

which coincides with the bound (2.6) derived earlier.

Also, in the special case where $\epsilon = \delta$ and $J_k = T_{\mu_k} J_{k+1}$ (cf. the discussion preceding Prop. 2.2.2), the bound (2.11) can be strengthened somewhat. In particular, we have for all k, $J_{m-k} = T_{\mu_{m-k}} \cdots T_{\mu_{m-1}} J_m$, so the right-hand side of Eq. (2.14) becomes 0 and the preceding proof yields, with some calculation,

$$\begin{aligned}\|J_\pi - J^*\| &\leq \frac{2\alpha^m}{1-\alpha^m}\|J_m - J^*\| + \frac{\delta}{1-\alpha^m} + \frac{\alpha\delta(1-\alpha^{m-1})}{(1-\alpha)(1-\alpha^m)} \\ &= \frac{2\alpha^m}{1-\alpha^m}\|J_m - J^*\| + \frac{\delta}{1-\alpha}.\end{aligned}$$

We finally note that Prop. 2.2.2 shows that as the lookahead size m increases, the corresponding bound for $\|J_\pi - J^*\|$ tends to $\epsilon + \alpha(\epsilon + 2\delta)/(1-\alpha)$, or

$$\limsup_{m\to\infty} \|J_\pi - J^*\| \leq \frac{\epsilon + 2\alpha\delta}{1-\alpha}.$$

We will see that this error bound is superior to corresponding error bounds for approximate versions of value and policy iteration by essentially a factor $1/(1-\alpha)$. In practice, however, periodic suboptimal policies, as required by Prop. 2.2.2, are typically not used.

There is an alternative and often used form of *on-line* multistep lookahead, whereby at the current state x we compute a multistep policy $\{\mu_0, \ldots, \mu_{m-1}\}$, we apply the first component $\mu_0(x)$ of that policy at state x, then at the next state $\bar{x}$ we recompute a new multistep policy $\{\bar{\mu}_0, \ldots, \bar{\mu}_{m-1}\}$, apply $\bar{\mu}_0(\bar{x})$, etc. However, no error bound similar to the one of Prop. 2.2.2 is currently known for this type of lookahead.

2.3 VALUE ITERATION

In this section, we discuss value iteration (VI for short), the algorithm that starts with some $J \in \mathcal{B}(X)$, and generates $TJ, T^2J, \ldots$. Since T is a weighted sup-norm contraction under Assumption 2.1.2, the algorithm converges to J^*, and the rate of convergence is governed by

$$\|T^k J - J^*\| \leq \alpha^k \|J - J^*\|, \qquad k = 0, 1, \ldots.$$

Similarly, for a given policy $\mu \in \mathcal{M}$, we have

$$\|T_\mu^k J - J_\mu\| \leq \alpha^k \|J - J_\mu\|, \qquad k = 0, 1, \ldots.$$

From Prop. 2.1.1(d), we also have the error bound

$$\|T^{k+1} J - J^*\| \leq \frac{\alpha}{1-\alpha}\|T^{k+1} J - T^k J\|, \qquad k = 0, 1, \ldots.$$

This bound does not rely on the monotonicity Assumption 2.1.1.

The VI algorithm is often used to compute an approximation $\tilde{J}$ to J^*, and then to obtain a policy $\overline{\mu}$ by minimizing $H(x,u,\tilde{J})$ over $u \in U(x)$ for each $x \in X$. In other words $\tilde{J}$ and $\overline{\mu}$ satisfy

$$\|\tilde{J} - J^*\| \leq \gamma, \qquad T_{\overline{\mu}}\tilde{J} = T\tilde{J},$$

where γ is some positive scalar. Then by using Eq. (2.6), we have

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{2\alpha\gamma}{1-\alpha}. \tag{2.15}$$

If the set of policies is finite, this procedure can be used to compute an optimal policy with a finite but sufficiently large number of exact VI, as shown in the following proposition.

Proposition 2.3.1: Let the contraction Assumption 2.1.2 hold and let $J \in \mathcal{B}(X)$. If the set of policies $\mathcal{M}$ is finite, there exists an integer $\overline{k} \geq 0$ such that $J_{\mu^*} = J^*$ for all μ^* and $k \geq \overline{k}$ with $T_{\mu^*}T^kJ = T^{k+1}J$.

Proof: Let $\tilde{\mathcal{M}}$ be the set of policies such that $J_\mu \neq J^*$. Since $\tilde{\mathcal{M}}$ is finite, we have

$$\inf_{\mu \in \tilde{\mathcal{M}}} \|J_\mu - J^*\| > 0,$$

so by Eq. (2.15), there exists sufficiently small $\beta > 0$ such that

$$\|\tilde{J} - J^*\| \leq \beta \ \text{ and } \ T_\mu\tilde{J} = T\tilde{J} \quad \Rightarrow \quad \|J_\mu - J^*\| = 0 \quad \Rightarrow \quad \mu \notin \tilde{\mathcal{M}}. \tag{2.16}$$

It follows that if k is sufficiently large so that $\|T^kJ - J^*\| \leq \beta$, then $T_{\mu^*}T^kJ = T^{k+1}J$ implies that $\mu^* \notin \tilde{\mathcal{M}}$ so $J_{\mu^*} = J^*$. **Q.E.D.**

2.3.1 Approximate Value Iteration

We will now consider situations where the VI method may be implementable only through approximations. In particular, given a function J, assume that we may only be able to calculate an approximation $\tilde{J}$ to TJ such that

$$\big\|\tilde{J} - TJ\big\| \leq \delta,$$

where δ is a given positive scalar. In the corresponding approximate VI method, we start from an arbitrary bounded function J_0, and we generate a sequence $\{J_k\}$ satisfying

$$\|J_{k+1} - TJ_k\| \leq \delta, \qquad k = 0, 1, \ldots. \tag{2.17}$$

This approximation may be the result of representing J_{k+1} compactly, as a linear combination of basis functions, through a projection or aggregation process, as is common in approximate DP (cf. the discussion of Section 1.2.4).

We may also simultaneously generate a sequence of policies $\{\mu^k\}$ such that

$$\|T_{\mu^k} J_k - T J_k\| \leq \epsilon, \qquad k = 0, 1, \ldots, \tag{2.18}$$

where ϵ is some scalar [which could be equal to 0, as in case of Eq. (2.10), considered earlier]. The following proposition shows that the corresponding cost functions J_{μ^k} "converge" to J^* to within an error of order $O\big(\delta/(1-\alpha)^2\big)$ [plus a less significant error of order $O\big(\epsilon/(1-\alpha)\big)$].

Proposition 2.3.2: (Error Bounds for Approximate VI) Let the contraction Assumption 2.1.2 hold. A sequence $\{J_k\}$ generated by the approximate VI method (2.17)-(2.18) satisfies

$$\limsup_{k\to\infty} \|J_k - J^*\| \leq \frac{\delta}{1-\alpha}, \tag{2.19}$$

while the corresponding sequence of policies $\{\mu^k\}$ satisfies

$$\limsup_{k\to\infty} \|J_{\mu^k} - J^*\| \leq \frac{\epsilon}{1-\alpha} + \frac{2\alpha\delta}{(1-\alpha)^2}. \tag{2.20}$$

Proof: Using the triangle inequality, Eq. (2.17), and the contraction property of T, we have

$$\begin{aligned} \|J_k - T^k J_0\| &\leq \|J_k - T J_{k-1}\| \\ &\quad + \|T J_{k-1} - T^2 J_{k-2}\| + \cdots + \|T^{k-1} J_1 - T^k J_0\| \\ &\leq \delta + \alpha\delta + \cdots + \alpha^{k-1}\delta, \end{aligned}$$

and finally

$$\|J_k - T^k J_0\| \leq \frac{(1-\alpha^k)\delta}{1-\alpha}, \qquad k = 0, 1, \ldots. \tag{2.21}$$

By taking limit as $k \to \infty$ and by using the fact $\lim_{k\to\infty} T^k J_0 = J^*$, we obtain Eq. (2.19).

We also have using the triangle inequality and the contraction property of T_{μ^k} and T,

$$\begin{aligned} \|T_{\mu^k} J^* - J^*\| &\leq \|T_{\mu^k} J^* - T_{\mu^k} J_k\| + \|T_{\mu^k} J_k - T J_k\| + \|T J_k - J^*\| \\ &\leq \alpha\|J^* - J_k\| + \epsilon + \alpha\|J_k - J^*\|, \end{aligned}$$

while by using also Prop. 2.1.1(e), we obtain

$$\|J_{\mu^k} - J^*\| \leq \frac{1}{1-\alpha}\|T_{\mu^k}J^* - J^*\| \leq \frac{\epsilon}{1-\alpha} + \frac{2\alpha}{1-\alpha}\|J_k - J^*\|.$$

By combining this relation with Eq. (2.19), we obtain Eq. (2.20). **Q.E.D.**

The error bound (2.20) relates to stationary policies obtained from the functions J_k by one-step lookahead. We may also obtain an m-step periodic policy π from J_k by using m-step lookahead. Then Prop. 2.2.2 shows that the corresponding bound for $\|J_\pi - J^*\|$ tends to $\epsilon + 2\alpha\delta/(1-\alpha)$ as $m \to \infty$, which improves on the error bound (2.20) by a factor $1/(1-\alpha)$.

Finally, let us note that the error bound of Prop. 2.3.2 is predicated upon generating a sequence $\{J_k\}$ satisfying $\|J_{k+1} - TJ_k\| \leq \delta$ for all k [cf. Eq. (2.17)]. Unfortunately, some practical approximation schemes guarantee the existence of such a δ only if $\{J_k\}$ is a bounded sequence. The following example from [BeT96], Section 6.5.3, shows that boundedness of the iterates is not automatically guaranteed, and is a serious issue that should be addressed in approximate VI schemes.

Example 2.3.1 (Error Amplification in Approximate Value Iteration)

Consider a two-state α-discounted MDP with states 1 and 2, and a single policy. The transitions are deterministic: from state 1 to state 2, and from state 2 to state 2. These transitions are also cost-free. Thus we have $(TJ(1) = (TJ)(2) = \alpha J(2)$, and $J^*(1) = J^*(2) = 0$.

We consider a VI scheme that approximates cost functions within the one-dimensional subspace of linear functions $S = \big\{(r, 2r) \mid r \in \Re\big\}$ by using a weighted least squares minimization; i.e., we approximate a vector J by its weighted Euclidean projection onto S. In particular, given $J_k = (r_k, 2r_k)$, we find $J_{k+1} = (r_{k+1}, 2r_{k+1})$, where for weights $\xi_1, \xi_2 > 0$, r_{k+1} is obtained as

$$r_{k+1} \in \arg\min_r \Big[\xi_1\big(r - (TJ_k)(1)\big)^2 + \xi_2\big(2r - (TJ_k)(2)\big)^2\Big].$$

Since for a zero cost per stage and the given deterministic transitions, we have $TJ_k = (2\alpha r_k, 2\alpha r_k)$, the preceding minimization is written as

$$r_{k+1} \in \arg\min_r \big[\xi_1(r - 2\alpha r_k)^2 + \xi_2(2r - 2\alpha r_k)^2\big],$$

which by writing the corresponding optimality condition yields $r_{k+1} = \alpha\beta r_k$, where $\beta = 2(\xi_1 + 2\xi_2)(\xi_1 + 4\xi_2) > 1$. Thus if $\alpha > 1/\beta$, the sequence $\{r_k\}$ diverges and so does $\{J_k\}$. Note that in this example the optimal cost function $J^* = (0, 0)$ belongs to the subspace S. The difficulty here is that the approximate VI mapping that generates J_{k+1} as the weighted Euclidean projection of TJ_k is not a contraction (this is a manifestation of an important issue in approximate DP and projected equation approximation, namely that the projected mapping ΠT need not be a contraction even if T is a sup-norm contraction; see [DFV00], [Ber12b] for examples and related discussions). At the same time there is no δ such that $\|J_{k+1} - TJ_k\| \leq \delta$ for all k, because of error amplification in each approximate VI.

2.4 POLICY ITERATION

In this section, we discuss policy iteration (PI for short), an algorithm whereby we maintain and update a policy μ^k, starting from some initial policy μ^0. The typical iteration has the following form (see Fig. 2.4.1 for a one-dimensional illustration).

Policy iteration given the current policy μ^k:

Policy evaluation: We compute J_{μ^k} as the unique solution of the equation

$$J_{\mu^k} = T_{\mu^k} J_{\mu^k}.$$

Policy improvement: We obtain a policy μ^{k+1} that satisfies

$$T_{\mu^{k+1}} J_{\mu^k} = T J_{\mu^k}.$$

We assume that the minimum of $H(x, u, J_{\mu^k})$ over $u \in U(x)$ is attained for all $x \in X$, so that the improved policy μ^{k+1} is defined (we use this assumption for all the PI algorithms of the book). The following proposition establishes a basic cost improvement property, as well as finite convergence for the case where the set of policies is finite.

Proposition 2.4.1: (Convergence of PI) Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, and let $\{\mu^k\}$ be a sequence generated by the PI algorithm. Then for all k, we have $J_{\mu^{k+1}} \leq J_{\mu^k}$, with equality if and only if $J_{\mu^k} = J^*$. Moreover,

$$\lim_{k \to \infty} \|J_{\mu^k} - J^*\| = 0,$$

and if the set of policies is finite, we have $J_{\mu^k} = J^*$ for some k.

Proof: We have

$$T_{\mu^{k+1}} J_{\mu^k} = T J_{\mu^k} \leq T_{\mu^k} J_{\mu^k} = J_{\mu^k}.$$

Applying $T_{\mu^{k+1}}$ to this inequality while using the monotonicity Assumption 2.1.1, we obtain

$$T^2_{\mu^{k+1}} J_{\mu^k} \leq T_{\mu^{k+1}} J_{\mu^k} = T J_{\mu^k} \leq T_{\mu^k} J_{\mu^k} = J_{\mu^k}.$$

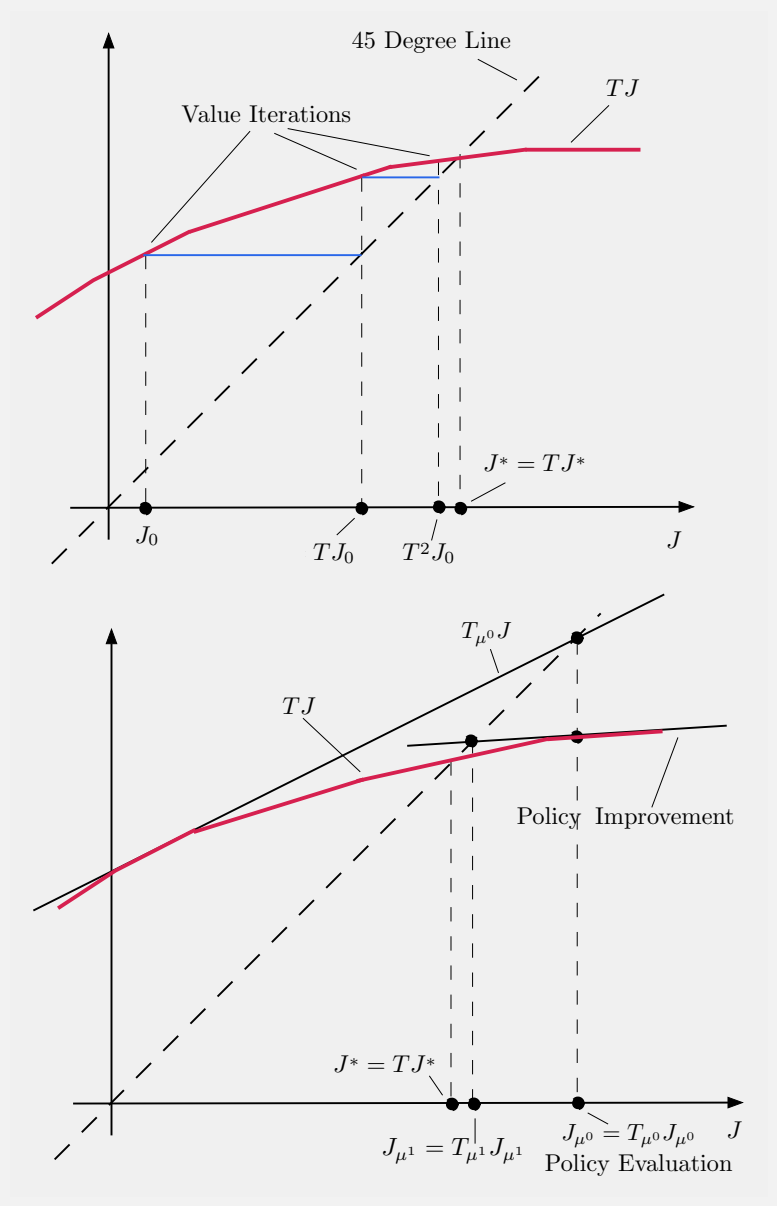

Figure 2.4.1 Geometric interpretation of PI and VI in one dimension (a single state). Each policy μ defines the mapping T_μ, and TJ is the function $\min_\mu T_\mu J$. When the number of policies is finite, TJ a piecewise linear concave function, with each piece being a linear function $T_\mu J$ that corresponds to a policy μ. The optimal cost function J^* satisfies $J^* = TJ^*$, so it is obtained from the intersection of the graph of TJ and the 45 degree line shown. Similarly J_μ is the intersection of the graph of $T_\mu J$ and the 45 degree line. The VI sequence is indicated in the top figure by the staircase construction, which asymptotically leads to J^*. A single policy iteration is illustrated in the bottom figure.

Similarly, we have for all $m > 0$,

$$T^m_{\mu^{k+1}} J_{\mu^k} \leq T J_{\mu^k} \leq J_{\mu^k},$$

and by taking the limit as $m \to \infty$, we obtain

$$J_{\mu^{k+1}} \leq T J_{\mu^k} \leq J_{\mu^k}, \qquad k = 0, 1, \ldots. \tag{2.22}$$

If $J_{\mu^{k+1}} = J_{\mu^k}$, it follows that $T J_{\mu^k} = J_{\mu^k}$, so J_{μ^k} is a fixed point of T and must be equal to J^*. Moreover by using induction, Eq. (2.22) implies that

$$J_{\mu^k} \leq T^k J_{\mu^0}, \qquad k = 0, 1, \ldots.$$

Since

$$J^* \leq J_{\mu^k}, \qquad \lim_{k\to\infty} \|T^k J_{\mu^0} - J^*\| = 0,$$

it follows that $\lim_{k\to\infty} \|J_{\mu^k} - J^*\| = 0$.

Finally, if the number of policies is finite, Eq. (2.22) implies that there can be only a finite number of iterations for which $J_{\mu^{k+1}}(x) < J_{\mu^k}(x)$ for some x. Thus we must have $J_{\mu^{k+1}} = J_{\mu^k}$ for some k, at which time $J_{\mu^k} = J^*$ as shown earlier [cf. Eq. (2.22)]. **Q.E.D.**

In the case where the set of policies is infinite, we may assert the convergence of the sequence of generated policies under some compactness and continuity conditions. In particular, we will assume that the state space is finite, $X = \{1, \ldots, n\}$, and that each control constraint set $U(x)$ is a compact subset of $\Re^m$. We will view a cost function J as an element of $\Re^n$, and a policy μ as an element of the set $U(1) \times \cdots \times U(n) \subset \Re^{mn}$, which is compact. Then $\{\mu^k\}$ has at least one limit point $\overline{\mu}$, which must be an admissible policy. The following proposition guarantees, under an additional continuity assumption for $H(x, \cdot, \cdot)$, that every limit point $\overline{\mu}$ is optimal.

Assumption 2.4.1: (Compactness and Continuity)

(a) The state space is finite, $X = \{1, \ldots, n\}$.

(b) Each control constraint set $U(x)$, $x = 1, \ldots, n$, is a compact subset of $\Re^m$.

(c) Each function $H(x, \cdot, \cdot)$, $x = 1, \ldots, n$, is continuous over $U(x) \times \Re^n$.

Proposition 2.4.2: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, together with Assumption 2.4.1, and let $\{\mu^k\}$ be a sequence generated by the PI algorithm. Then for every limit point $\overline{\mu}$ of $\{\mu^k\}$, we have $J_{\overline{\mu}} = J^*$.

Proof: We have $J_{\mu^k} \to J^*$ by Prop. 2.4.1. Let $\overline{\mu}$ be the limit of a subsequence $\{\mu^k\}_{k\in\mathcal{K}}$. We will show that $T_{\overline{\mu}}J^* = TJ^*$, from which it follows that $J_{\overline{\mu}} = J^*$ [cf. Prop. 2.1.1(c)]. Indeed, we have $T_{\overline{\mu}}J^* \geq TJ^*$, so we focus on showing the reverse inequality. From the equation

$$T_{\mu^k}J_{\mu^{k-1}} = TJ_{\mu^{k-1}},$$

we have

$$H\big(x,\mu^k(x),J_{\mu^{k-1}}\big) \leq H(x,u,J_{\mu^{k-1}}), \qquad x=1,\ldots,n,\ u\in U(x).$$

By taking limit in this relation as $k \to \infty$, $k \in \mathcal{K}$, and by using the continuity of $H(x,\cdot,\cdot)$ [cf. Assumption 2.4.1(c)], we obtain

$$H\big(x,\overline{\mu}(x),J^*\big) \leq H(x,u,J^*), \qquad x=1,\ldots,n,\ u\in U(x).$$

By taking the minimum of the right-hand side over $u \in U(x)$, we obtain $T_{\overline{\mu}}J^* \leq TJ^*$. **Q.E.D.**

2.4.1 Approximate Policy Iteration

We now consider the PI method where the policy evaluation step and/or the policy improvement step of the method are implemented through approximations. This method generates a sequence of policies $\{\mu^k\}$ and a corresponding sequence of approximate cost functions $\{J_k\}$ satisfying

$$\|J_k - J_{\mu^k}\| \leq \delta, \qquad \|T_{\mu^{k+1}}J_k - TJ_k\| \leq \epsilon, \qquad k=0,1,\ldots, \tag{2.23}$$

where δ and ϵ are some scalars, and $\|\cdot\|$ denotes the weighted sup-norm (the one used in the contraction Assumption 2.1.2). The following proposition provides an error bound for this algorithm.

Proposition 2.4.3: (Error Bound for Approximate PI) Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold. The sequence $\{\mu^k\}$ generated by the approximate PI algorithm (2.23) satisfies

$$\limsup_{k\to\infty} \|J_{\mu^k} - J^*\| \leq \frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}. \tag{2.24}$$

The essence of the proof is contained in the following proposition, which quantifies the amount of approximate policy improvement at each iteration.

Proposition 2.4.4: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold. Let J, $\overline{\mu}$, and μ satisfy

$$\|J - J_\mu\| \leq \delta, \qquad \|T_{\overline{\mu}} J - TJ\| \leq \epsilon,$$

where δ and ϵ are some scalars. Then

$$\|J_{\overline{\mu}} - J^*\| \leq \alpha \|J_\mu - J^*\| + \frac{\epsilon + 2\alpha\delta}{1-\alpha}. \tag{2.25}$$

Proof: We denote by v the weight function corresponding to the weighted sup-norm. Using the contraction property of T and $T_{\overline{\mu}}$, which implies that $\|T_{\overline{\mu}} J_\mu - T_{\overline{\mu}} J\| \leq \alpha\delta$ and $\|TJ - TJ_\mu\| \leq \alpha\delta$, and hence $T_{\overline{\mu}} J_\mu \leq T_{\overline{\mu}} J + \alpha\delta\, v$ and $TJ \leq TJ_\mu + \alpha\delta\, v$, we have

$$T_{\overline{\mu}} J_\mu \leq T_{\overline{\mu}} J + \alpha\delta\, v \leq TJ + (\epsilon + \alpha\delta)\, v \leq TJ_\mu + (\epsilon + 2\alpha\delta)\, v. \tag{2.26}$$

Since $TJ_\mu \leq T_\mu J_\mu = J_\mu$, this relation yields

$$T_{\overline{\mu}} J_\mu \leq J_\mu + (\epsilon + 2\alpha\delta)\, v,$$

and applying Prop. 2.1.4(b) with $\mu = \overline{\mu}$, $J = J_\mu$, and $c = \epsilon + 2\alpha\delta$, we obtain

$$J_{\overline{\mu}} \leq J_\mu + \frac{\epsilon + 2\alpha\delta}{1-\alpha}\, v. \tag{2.27}$$

Using this relation, we have

$$J_{\overline{\mu}} = T_{\overline{\mu}} J_{\overline{\mu}} = T_{\overline{\mu}} J_\mu + (T_{\overline{\mu}} J_{\overline{\mu}} - T_{\overline{\mu}} J_\mu) \leq T_{\overline{\mu}} J_\mu + \frac{\alpha(\epsilon + 2\alpha\delta)}{1-\alpha}\, v,$$

where the inequality follows by using Prop. 2.1.3 and Eq. (2.27). Subtracting J^* from both sides, we have

$$J_{\overline{\mu}} - J^* \leq T_{\overline{\mu}} J_\mu - J^* + \frac{\alpha(\epsilon + 2\alpha\delta)}{1-\alpha}\, v. \tag{2.28}$$

Also by subtracting J^* from both sides of Eq. (2.26), and using the contraction property

$$TJ_\mu - J^* = TJ_\mu - TJ^* \leq \alpha \|J_\mu - J^*\|\, v,$$

we obtain

$$T_{\overline{\mu}} J_\mu - J^* \leq TJ_\mu - J^* + (\epsilon + 2\alpha\delta)\, v \leq \alpha \|J_\mu - J^*\|\, v + (\epsilon + 2\alpha\delta)\, v.$$

Combining this relation with Eq. (2.28), yields

$$J_{\overline{\mu}} - J^* \leq \alpha \|J_\mu - J^*\| \, v + \frac{\alpha(\epsilon + 2\alpha\delta)}{1-\alpha} \, v + (\epsilon + \alpha\delta) e = \alpha \|J_\mu - J^*\| \, v + \frac{\epsilon + 2\alpha\delta}{1-\alpha} \, v,$$

which is equivalent to the desired relation (2.25). **Q.E.D.**

Proof of Prop. 2.4.3: Applying Prop. 2.4.4, we have

$$\|J_{\mu^{k+1}} - J^*\| \leq \alpha \|J_{\mu^k} - J^*\| + \frac{\epsilon + 2\alpha\delta}{1-\alpha},$$

which by taking the lim sup of both sides as $k \to \infty$ yields the desired result. **Q.E.D.**

We note that the error bound of Prop. 2.4.3 is tight, as can be shown with an example from [BeT96], Section 6.2.3. The error bound is comparable to the one for approximate VI, derived earlier in Prop. 2.3.2. In particular, the error $\|J_{\mu^k} - J^*\|$ is asymptotically proportional to $1/(1-\alpha)^2$ and to the approximation error in policy evaluation or value iteration, respectively. This is noteworthy, as it indicates that contrary to the case of exact implementation, approximate PI need not hold a convergence rate advantage over approximate VI, despite its greater overhead per iteration.

Note that when $\delta = \epsilon = 0$, Eq. (2.25) yields

$$\|J_{\mu^{k+1}} - J^*\| \leq \alpha \|J_{\mu^k} - J^*\|.$$

Thus in the case of an infinite state space and/or control space, exact PI converges at a geometric rate under the contraction and monotonicity assumptions of this section. This rate is the same as the rate of convergence of exact VI. It follows that judging solely from the point of view of rate of convergence estimates, exact PI holds an advantage over exact VI only when the number of states is finite. This raises the question what happens when the number of states is finite but very large. However, this question is not very interesting from a practical point of view, since for a very large number of states, neither VI or PI can be implemented in practice without approximations (see the discussion of Section 1.2.4).

2.4.2 Approximate Policy Iteration Where Policies Converge

Generally, the policy sequence $\{\mu^k\}$ generated by approximate PI may oscillate between several policies. However, under some circumstances this sequence may be guaranteed to converge to some $\overline{\mu}$, in the sense that

$$\mu^{\overline{k}+1} = \mu^{\overline{k}} = \overline{\mu} \qquad \text{for some } \overline{k}. \tag{2.29}$$

An example arises when the policy sequence $\{\mu^k\}$ is generated by *exact PI* applied with a *different* mapping $\tilde{H}$ in place of H, but the policy evaluation and policy improvement error bounds of Eq. (2.23) are satisfied. The mapping $\tilde{H}$ may for example correspond to an approximation of the original problem (as in the aggregation methods of Example 1.2.10; see [Ber11c] and [Ber12a] for further discussion). In this case we can show the following bound, which is much more favorable than the one of Prop. 2.4.3.

Proposition 2.4.5: (Error Bound for Approximate PI when Policies Converge) Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, and assume that the approximate PI algorithm (2.23) terminates with a policy $\overline{\mu}$ that satisfies condition (2.29). Then we have

$$\|J_{\overline{\mu}} - J^*\| \leq \frac{\epsilon + 2\alpha\delta}{1-\alpha}. \tag{2.30}$$

Proof: Let $\tilde{J}$ be the cost function obtained by approximate policy evaluation of $\overline{\mu}$ [i.e., $\tilde{J} = J_{\overline{k}}$, where $\overline{k}$ satisfies the condition (2.29)]. Then we have

$$\|\tilde{J} - J_{\overline{\mu}}\| \leq \delta, \qquad \|T_{\overline{\mu}}\tilde{J} - T\tilde{J}\| \leq \epsilon, \tag{2.31}$$

where the latter inequality holds since we have

$$\|T_{\overline{\mu}}\tilde{J} - T\tilde{J}\| = \|T_{\mu^{\overline{k}+1}}J_{\overline{k}} - TJ_{\overline{k}}\| \leq \epsilon,$$

cf. Eq. (2.23). Using Eq. (2.31) and the fact $J_{\overline{\mu}} = T_{\overline{\mu}}J_{\overline{\mu}}$, we have

$$\begin{aligned} \|TJ_{\overline{\mu}} - J_{\overline{\mu}}\| &\leq \|TJ_{\overline{\mu}} - T\tilde{J}\| + \|T\tilde{J} - T_{\overline{\mu}}\tilde{J}\| + \|T_{\overline{\mu}}\tilde{J} - J_{\overline{\mu}}\| \\ &= \|TJ_{\overline{\mu}} - T\tilde{J}\| + \|T\tilde{J} - T_{\overline{\mu}}\tilde{J}\| + \|T_{\overline{\mu}}\tilde{J} - T_{\overline{\mu}}J_{\overline{\mu}}\| \\ &\leq \alpha\|J_{\overline{\mu}} - \tilde{J}\| + \epsilon + \alpha\|\tilde{J} - J_{\overline{\mu}}\| \\ &\leq \epsilon + 2\alpha\delta. \end{aligned} \tag{2.32}$$

Using Prop. 2.1.1(d) with $J = J_{\overline{\mu}}$, we obtain the error bound (2.30). **Q.E.D.**

The preceding error bound can be extended to the case where two successive policies generated by the approximate PI algorithm are "not too different" rather than being identical. In particular, suppose that μ and $\overline{\mu}$ are successive policies, which in addition to

$$\|\tilde{J} - J_{\mu}\| \leq \delta, \qquad \|T_{\overline{\mu}}\tilde{J} - T\tilde{J}\| \leq \epsilon,$$

[cf. Eq. (2.23)], also satisfy

$$\|T_{\mu}\tilde{J} - T_{\overline{\mu}}\tilde{J}\| \leq \zeta,$$

where ζ is some scalar (instead of $\mu = \overline{\mu}$, which is the case where policies converge exactly). Then we also have

$$\|T\tilde{J} - T_\mu \tilde{J}\| \leq \|T\tilde{J} - T_{\overline{\mu}}\tilde{J}\| + \|T_{\overline{\mu}}\tilde{J} - T_\mu \tilde{J}\| \leq \epsilon + \zeta,$$

and by replacing ϵ with $\epsilon + \zeta$ and $\overline{\mu}$ with μ in Eq. (2.32), we obtain

$$\|J_\mu - J^*\| \leq \frac{\epsilon + \zeta + 2\alpha\delta}{1-\alpha}.$$

When ζ is small enough to be of the order of $\max\{\delta, \epsilon\}$, this error bound is comparable to the one for the case where policies converge.

2.5 OPTIMISTIC POLICY ITERATION AND λ-POLICY ITERATION

In this section, we discuss some variants of the PI algorithm of the preceding section, where the policy evaluation

$$J_{\mu^k} = T_{\mu^k} J_{\mu^k}$$

is approximated by using VI. The most straightforward of these methods is *optimistic PI* (also called "modified" PI, see e.g., [Put94]), where a policy μ^k is evaluated approximately, using a finite number of VI. Thus, starting with a function $J_0 \in \mathcal{B}(X)$, we generate sequences $\{J_k\}$ and $\{\mu^k\}$ with the algorithm

$$T_{\mu^k} J_k = TJ_k, \qquad J_{k+1} = T_{\mu^k}^{m_k} J_k, \qquad k = 0, 1, \ldots, \tag{2.33}$$

where $\{m_k\}$ is a sequence of positive integers (see Fig. 2.5.1, which shows one iteration of the method where $m_k = 3$). There is no systematic guideline for selecting the integers m_k. Usually their best values are chosen empirically, and tend to be considerably larger than 1 (in the case where $m_k \equiv 1$ the optimistic PI method coincides with the VI method). The convergence of this method is discussed in Section 2.5.1.

Variants of optimistic PI include methods with approximations in the policy evaluation and policy improvement phases (Section 2.5.2), and methods where the number m_k is randomized (Section 2.5.3). An interesting advantage of the latter methods is that *they do not require the monotonicity Assumption 2.1.1* for convergence in problems with a finite number of policies.

A method that is conceptually similar to the optimistic PI method is the λ-PI method defined by

$$T_{\mu^k} J_k = TJ_k, \qquad J_{k+1} = T_{\mu^k}^{(\lambda)} J_k, \qquad k = 0, 1, \ldots, \tag{2.34}$$

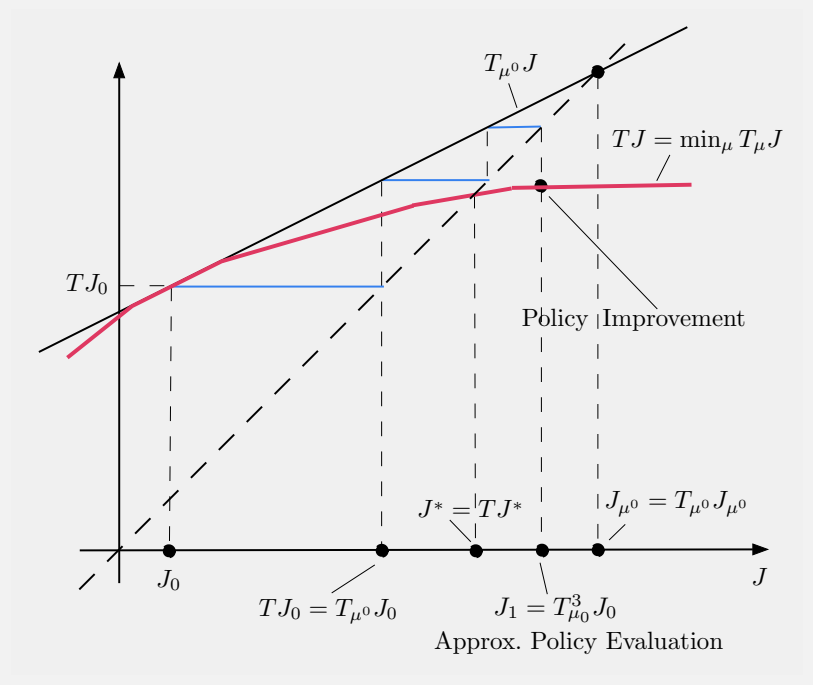

Figure 2.5.1 Illustration of optimistic PI in one dimension. In this example, the policy μ^0 is evaluated approximately with just three applications of T_{μ^0} to yield $J_1 = T^3_{\mu^0}J_0$.

where J_0 is an initial function in $\mathcal{B}(X)$, and for any policy μ and scalar $\lambda \in (0,1)$, $T_\mu^{(\lambda)}$ is the multistep mapping defined by

$$T_\mu^{(\lambda)}J = (1-\lambda)\sum_{\ell=0}^{\infty}\lambda^\ell T_\mu^{\ell+1}J, \qquad J \in \mathcal{B}(X),$$

(cf. Section 1.2.5). To compare optimistic PI and λ-PI, note that they both involve multiple applications of the VI mapping T_{μ^k}: a fixed number m_k in the former case, and a geometrically weighted number in the latter case. In fact, we may view the λ-PI iterate $T_{\mu^k}^{(\lambda)}J_k$ as the expected value of the optimistic PI iterate $T_{\mu^k}^{m_k}J_{\mu^k}$ when m_k is chosen by a geometric probability distribution with parameter λ.

One of the reasons that make λ-PI interesting is its relation with TD(λ) and other temporal difference methods on one hand, and the proximal algorithm on the other. In particular, in λ-PI a policy evaluation is performed with a single iteration of an extrapolated proximal algorithm; cf. the discussion of Section 1.2.5 and Exercise 1.2. Thus implementation

of λ-PI can benefit from the rich methodology that has developed around temporal difference and proximal methods.

Generally the optimistic and λ-PI methods have similar convergence properties. In this section, we focus primarily on optimistic PI, and we discuss briefly λ-PI in Section 2.5.3, where we will prove convergence for a randomized version. For a convergence proof of λ-PI without randomization in discounted stochastic optimal control and stochastic shortest path problems, see the paper [BeI96] and the book [BeT96] (Section 2.3.1).

2.5.1 Convergence of Optimistic Policy Iteration

We will now focus on the optimistic PI algorithm (2.33). The following two propositions provide its convergence properties.

Proposition 2.5.1: (Convergence of Optimistic PI) Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, and let $\{(J_k, \mu^k)\}$ be a sequence generated by the optimistic PI algorithm (2.33). Then

$$\lim_{k\to\infty} \|J_k - J^*\| = 0,$$

and if the number of policies is finite, we have $J_{\mu^k} = J^*$ for all k greater than some index $\bar{k}$.

Proposition 2.5.2: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, together with Assumption 2.4.1, and let $\{(J_k, \mu^k)\}$ be a sequence generated by the optimistic PI algorithm (2.33). Then for every limit point $\overline{\mu}$ of $\{\mu^k\}$, we have $J_{\overline{\mu}} = J^*$.

We develop the proofs of the propositions through four lemmas. The first lemma collects some properties of monotone weighted sup-norm contractions, variants of which we noted earlier and we restate for convenience.

Lemma 2.5.1: Let $W : \mathcal{B}(X) \mapsto \mathcal{B}(X)$ be a mapping that satisfies the monotonicity assumption

$$J \leq J' \quad \Rightarrow \quad WJ \leq WJ', \qquad \forall\ J, J' \in \mathcal{B}(X),$$

and the contraction assumption

$$\|WJ - WJ'\| \leq \alpha \|J - J'\|, \qquad \forall\ J, J' \in \mathcal{B}(X),$$

for some $\alpha \in (0, 1)$.

(a) For all $J, J' \in \mathcal{B}(X)$ and scalar $c \geq 0$, we have

$$J \geq J' - c\,v \qquad \Rightarrow \qquad WJ \geq WJ' - \alpha c\,v. \tag{2.35}$$

(b) For all $J \in \mathcal{B}(X)$, $c \geq 0$, and $k = 0, 1, \ldots$, we have

$$J \geq WJ - c\,v \qquad \Rightarrow \qquad W^k J \geq J^* - \frac{\alpha^k}{1-\alpha} c\,v, \tag{2.36}$$

$$WJ \geq J - c\,v \qquad \Rightarrow \qquad J^* \geq W^k J - \frac{\alpha^k}{1-\alpha} c\,v, \tag{2.37}$$

where J^* is the fixed point of W.

Proof: The proof of part (a) follows the one of Prop. 2.1.4(b), while the proof of part (b) follows the one of Prop. 2.1.4(c). **Q.E.D.**

Lemma 2.5.2: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, let $J \in \mathcal{B}(X)$ and $c \geq 0$ satisfy

$$J \geq TJ - c\,v,$$

and let $\mu \in \mathcal{M}$ be such that $T_\mu J = TJ$. Then for all $k > 0$, we have

$$TJ \geq T_\mu^k J - \frac{\alpha}{1-\alpha} c\,v, \tag{2.38}$$

and

$$T_\mu^k J \geq T(T_\mu^k J) - \alpha^k c\,v. \tag{2.39}$$

Proof: Since $J \geq TJ - c\,v = T_\mu J - c\,v$, by using Lemma 2.5.1(a) with $W = T_\mu^j$ and $J' = T_\mu J$, we have for all $j \geq 1$,

$$T_\mu^j J \geq T_\mu^{j+1} J - \alpha^j c\,v. \tag{2.40}$$

By adding this relation over $j = 1, \ldots, k-1$, we have

$$TJ = T_\mu J \geq T_\mu^k J - \sum_{j=1}^{k-1} \alpha^j c\,v = T_\mu^k J - \frac{\alpha - \alpha^k}{1-\alpha} c\,v \geq T_\mu^k J - \frac{\alpha}{1-\alpha} c\,v,$$

showing Eq. (2.38). From Eq. (2.40) for $j = k$, we obtain

$$T_\mu^k J \geq T_\mu^{k+1} J - \alpha^k c\, v = T_\mu(T_\mu^k J) - \alpha^k c\, v \geq T(T_\mu^k J) - \alpha^k c\, v,$$

showing Eq. (2.39). **Q.E.D.**

The next lemma applies to the optimistic PI algorithm (2.33) and proves a preliminary bound.

Lemma 2.5.3: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, let $\{(J_k, \mu^k)\}$ be a sequence generated by the optimistic PI algorithm (2.33), and assume that for some $c \geq 0$ we have

$$J_0 \geq TJ_0 - c\, v.$$

Then for all $k \geq 0$,

$$TJ_k + \frac{\alpha}{1-\alpha}\beta_k c\, v \geq J_{k+1} \geq TJ_{k+1} - \beta_{k+1} c\, v, \tag{2.41}$$

where β_k is the scalar given by

$$\beta_k = \begin{cases} 1 & \text{if } k = 0, \\ \alpha^{m_0 + \cdots + m_{k-1}} & \text{if } k > 0, \end{cases} \tag{2.42}$$

with m_j, $j = 0, 1, \ldots$, being the integers used in the algorithm (2.33).

Proof: We prove Eq. (2.41) by induction on k, using Lemma 2.5.2. For $k = 0$, using Eq. (2.38) with $J = J_0$, $\mu = \mu^0$, and $k = m_0$, we have

$$TJ_0 \geq J_1 - \frac{\alpha}{1-\alpha} c\, v = J_1 - \frac{\alpha}{1-\alpha}\beta_0 c\, v,$$

showing the left-hand side of Eq. (2.41) for $k = 0$. Also by Eq. (2.39) with $\mu = \mu^0$ and $k = m_0$, we have

$$J_1 \geq TJ_1 - \alpha^{m_0} c\, v = TJ_1 - \beta_1 c\, v.$$

showing the right-hand side of Eq. (2.41) for $k = 0$.

Assuming that Eq. (2.41) holds for $k - 1 \geq 0$, we will show that it holds for k. Indeed, the right-hand side of the induction hypothesis yields

$$J_k \geq TJ_k - \beta_k c\, v.$$

Using Eqs. (2.38) and (2.39) with $J = J_k$, $\mu = \mu^k$, and $k = m_k$, we obtain

$$TJ_k \geq J_{k+1} - \frac{\alpha}{1-\alpha}\beta_k c\, v,$$

and

$$J_{k+1} \geq TJ_{k+1} - \alpha^{m_k}\beta_k c\, v = TJ_{k+1} - \beta_{k+1} c\, v,$$

respectively. This completes the induction. **Q.E.D.**

The next lemma essentially proves the convergence of the optimistic PI (Prop. 2.5.1) and provides associated error bounds.

Lemma 2.5.4: Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, let $\{(J_k, \mu^k)\}$ be a sequence generated by the optimistic PI algorithm (2.33), and let $c \geq 0$ be a scalar such that

$$\|J_0 - TJ_0\| \leq c. \tag{2.43}$$

Then for all $k \geq 0$,

$$J_k + \frac{\alpha^k}{1-\alpha} c\, v \geq J_k + \frac{\beta_k}{1-\alpha} c\, v \geq J^* \geq J_k - \frac{(k+1)\alpha^k}{1-\alpha} c\, v, \tag{2.44}$$

where β_k is defined by Eq. (2.42).

Proof: Using the relation $J_0 \geq TJ_0 - c\, v$ [cf. Eq. (2.43)] and Lemma 2.5.3, we have

$$J_k \geq TJ_k - \beta_k c\, v, \quad k = 0, 1, \ldots.$$

Using this relation in Lemma 2.5.1(b) with $W = T$ and $k = 0$, we obtain

$$J_k \geq J^* - \frac{\beta_k}{1-\alpha} c\, v,$$

which together with the fact $\alpha^k \geq \beta_k$, shows the left-hand side of Eq. (2.44).

Using the relation $TJ_0 \geq J_0 - c\, v$ [cf. Eq. (2.43)] and Lemma 2.5.1(b) with $W = T$, we have

$$J^* \geq T^k J_0 - \frac{\alpha^k}{1-\alpha} c\, v, \qquad k = 0, 1, \ldots. \tag{2.45}$$

Using again the relation $J_0 \geq TJ_0 - c\, v$ in conjunction with Lemma 2.5.3, we also have

$$TJ_j \geq J_{j+1} - \frac{\alpha}{1-\alpha}\beta_j c\, v, \qquad j = 0, \ldots, k-1.$$

Applying T^{k-j-1} to both sides of this inequality and using the monotonicity and contraction properties of T^{k-j-1}, we obtain

$$T^{k-j}J_j \geq T^{k-j-1}J_{j+1} - \frac{\alpha^{k-j}}{1-\alpha}\beta_j c\, v, \qquad j = 0, \ldots, k-1,$$

cf. Lemma 2.5.1(a). By adding this relation over $j = 0, \ldots, k-1$, and using the fact $\beta_j \leq \alpha^j$, it follows that

$$T^k J_0 \geq J_k - \sum_{j=0}^{k-1} \frac{\alpha^{k-j}}{1-\alpha} \alpha^j c\, v = J_k - \frac{k\alpha^k}{1-\alpha} c\, v. \tag{2.46}$$

Finally, by combining Eqs. (2.45) and (2.46), we obtain the right-hand side of Eq. (2.44). **Q.E.D.**

Proof of Props. 2.5.1 and 2.5.2: Let c be a scalar satisfying Eq. (2.43). Then the error bounds (2.44) show that $\lim_{k\to\infty} \|J_k - J^*\| = 0$, i.e., the first part of Prop. 2.5.1. To show the second part (finite termination when the number of policies is finite), let $\widehat{\mathcal{M}}$ be the finite set of nonoptimal policies. Then there exists $\epsilon > 0$ such that $\|T_{\hat{\mu}} J^* - TJ^*\| > \epsilon$ for all $\hat{\mu} \in \widehat{\mathcal{M}}$, which implies that $\|T_{\hat{\mu}} J_k - TJ_k\| > \epsilon$ for all $\hat{\mu} \in \widehat{\mathcal{M}}$ and k sufficiently large. This implies that $\mu^k \notin \widehat{\mathcal{M}}$ for all k sufficiently large. The proof of Prop. 2.5.2 follows using the compactness and continuity Assumption 2.4.1, and the convergence argument of Prop. 2.4.2. **Q.E.D.**

Convergence Rate Issues

Let us consider the convergence rate bounds of Lemma 2.5.4 for optimistic PI, and write them in the form

$$\|J_0 - TJ_0\| \leq c \quad \Rightarrow \quad J_k - \frac{(k+1)\alpha^k}{1-\alpha} c\, v \leq J^* \leq J_k + \frac{\alpha^{m_0+\cdots+m_k}}{1-\alpha} c\, v. \tag{2.47}$$

We may contrast these bounds with the ones for VI, where

$$\|J_0 - TJ_0\| \leq c \quad \Rightarrow \quad T^k J_0 - \frac{\alpha^k}{1-\alpha} c\, v \leq J^* \leq T^k J_0 + \frac{\alpha^k}{1-\alpha} c\, v \tag{2.48}$$

[cf. Prop. 2.1.4(c)].

In comparing the bounds (2.47) and (2.48), we should also take into account the associated overhead for a single iteration of each method: optimistic PI requires at iteration k a single application of T and $m_k - 1$ applications of T_{μ^k} (each being less time-consuming than an application of T), while VI requires a single application of T. It can then be seen that the upper bound for optimistic PI is better than the one for VI (same bound for less overhead), while the lower bound for optimistic PI is worse than the one for VI (worse bound for more overhead). This suggests that the choice of the initial condition J_0 is important in optimistic PI, and in particular it is preferable to have $J_0 \geq TJ_0$ (implying convergence to J^* from above) rather than $J_0 \leq TJ_0$ (implying convergence to J^* from below). This is consistent with the results of other works, which indicate that the convergence properties of the method are fragile when the condition $J_0 \geq TJ_0$ does not hold (see [WiB93], [BeT96], [BeY10], [BeY12], [YuB13a]).

2.5.2 Approximate Optimistic Policy Iteration

We will now derive error bounds for the case where the policy evaluation and policy improvement operations are approximate, similar to the nonoptimistic PI case of Section 2.4.1. In particular, we consider a method that generates a sequence of policies $\{\mu^k\}$ and a corresponding sequence of approximate cost functions $\{J_k\}$ satisfying

$$\|J_k - T_{\mu^k}^{m_k} J_{k-1}\| \leq \delta, \qquad \|T_{\mu^{k+1}} J_k - T J_k\| \leq \epsilon, \qquad k = 0, 1, \ldots, \quad (2.49)$$

[cf. Eq. (2.23)]. For example, we may compute (perhaps approximately, by simulation) the values $(T_{\mu^k}^{m_k} J_{k-1})(x)$ for a subset of states x, and use a least squares fit of these values to select J_k from some parametric class of functions.

We will prove the same error bound as for the nonoptimistic case, cf. Eq. (2.24). However, for this we will need the following condition, which is stronger than the contraction and monotonicity conditions that we have been using so far.

Assumption 2.5.1: (Semilinear Monotonic Contraction) For all $J \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$, the functions $T_\mu J$ and TJ belong to $\mathcal{B}(X)$. Furthermore, for some $\alpha \in (0,1)$, we have for all $J, J' \in \mathcal{B}(X)$, $\mu \in \mathcal{M}$, and $x \in X$,

$$\frac{(T_\mu J')(x) - (T_\mu J)(x)}{v(x)} \leq \alpha \sup_{y \in X} \frac{J'(y) - J(y)}{v(y)}. \quad (2.50)$$

This assumption implies both the monotonicity and contraction Assumptions 2.1.1 and 2.1.2, as can be easily verified. Moreover the assumption is satisfied in the discounted DP examples of Section 1.2, as well as the stochastic shortest path problem of Example 1.2.6. It holds if T_μ is a linear mapping involving a matrix with nonnegative components that has spectral radius less than 1 (or more generally if T_μ is the minimum or the maximum of a finite number of such linear mappings).

For any function $y \in \mathcal{B}(X)$, let us use the notation

$$M(y) = \sup_{x \in X} \frac{y(x)}{v(x)}.$$

Then the condition (2.50) can be written for all $J, J' \in \mathcal{B}(X)$, and $\mu \in \mathcal{M}$ as

$$M(T_\mu J - T_\mu J') \leq \alpha M(J - J'), \quad (2.51)$$

and also implies the following multistep versions, for $\ell \geq 1$,

$$T_\mu^\ell J - T_\mu^\ell J' \leq \alpha^\ell M(J - J')v, \quad M(T_\mu^\ell J - T_\mu^\ell J') \leq \alpha^\ell M(J - J'), \quad (2.52)$$

which can be proved by induction using Eq. (2.51). We have the following proposition.

Proposition 2.5.3: (Error Bound for Optimistic Approximate PI) Let Assumption 2.5.1 hold, in addition to the monotonicity and contraction Assumptions 2.1.1 and 2.1.2. Then the sequence $\{\mu^k\}$ generated by the optimistic approximate PI algorithm (2.49) satisfies

$$\limsup_{k\to\infty} \|J_{\mu^k} - J^*\| \leq \frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}.$$

Proof: Let us fix $k \geq 1$, and for simplicity let us assume that $m_k \equiv m$ for some m, and denote

$$\underline{J} = J_{k-1}, \qquad J = J_k, \qquad \mu = \mu^k, \qquad \bar{\mu} = \mu^{k+1},$$

$$s = J_\mu - T_\mu^m \underline{J}, \qquad \bar{s} = J_{\bar{\mu}} - T_{\bar{\mu}}^m J, \qquad t = T_\mu^m \underline{J} - J^*, \qquad \bar{t} = T_{\bar{\mu}}^m J - J^*.$$

We have

$$J_\mu - J^* = J_\mu - T_\mu^m \underline{J} + T_\mu^m \underline{J} - J^* = s + t. \quad (2.53)$$

We will derive recursive relations for s and t, which will also involve the residual functions

$$r = T_\mu \underline{J} - \underline{J}, \qquad \bar{r} = T_{\bar{\mu}} J - J.$$

We first obtain a relation between r and $\bar{r}$. We have

$$\begin{aligned}
\bar{r} &= T_{\bar{\mu}} J - J \\
&= (T_{\bar{\mu}} J - T_\mu J) + (T_\mu J - J) \\
&\leq (T_{\bar{\mu}} J - TJ) + \big(T_\mu J - T_\mu(T_\mu^m \underline{J})\big) + (T_\mu^m \underline{J} - J) + \big(T_\mu^m(T_\mu \underline{J}) - T_\mu^m \underline{J}\big) \\
&\leq \epsilon v + \alpha M(J - T_\mu^m \underline{J})v + \delta v + \alpha^m M(T_\mu \underline{J} - \underline{J})v \\
&\leq (\epsilon + \delta)v + \alpha\delta v + \alpha^m M(r)v,
\end{aligned}$$

where the first inequality follows from $T_{\bar{\mu}} J \geq TJ$, and the second and third inequalities follow from Eqs. (2.49) and (2.52). From this relation we have

$$M(\bar{r}) \leq \big(\epsilon + (1+\alpha)\delta\big) + \beta M(r),$$

where $\beta = \alpha^m$. Taking lim sup as $k \to \infty$ in this relation, we obtain

$$\limsup_{k\to\infty} M(r) \leq \frac{\epsilon + (1+\alpha)\delta}{1-\beta}. \tag{2.54}$$

Next we derive a relation between s and r. We have

$$\begin{aligned} s &= J_\mu - T_\mu^m \underline{J} \\ &= T_\mu^m J_\mu - T_\mu^m \underline{J} \\ &\leq \alpha^m M(J_\mu - \underline{J})v \\ &\leq \frac{\alpha^m}{1-\alpha} M(T_\mu \underline{J} - \underline{J})v \\ &= \frac{\alpha^m}{1-\alpha} M(r)v, \end{aligned}$$

where the first inequality follows from Eq. (2.52) and the second inequality follows by using Prop. 2.1.4(b). Thus we have $M(s) \leq \frac{\alpha^m}{1-\alpha} M(r)$, from which by taking lim sup of both sides and using Eq. (2.54), we obtain

$$\limsup_{k\to\infty} M(s) \leq \frac{\beta\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)(1-\beta)}. \tag{2.55}$$

Finally we derive a relation between t, $\bar{t}$, and r. We first note that

$$\begin{aligned} TJ - TJ^* &\leq \alpha M(J - J^*)v \\ &= \alpha M(J - T_\mu^m \underline{J} + T_\mu^m \underline{J} - J^*)v \\ &\leq \alpha M(J - T_\mu^m \underline{J})v + \alpha M(T_\mu^m \underline{J} - J^*)v \\ &\leq \alpha\delta v + \alpha M(t)v. \end{aligned}$$

Using this relation, and Eqs. (2.49) and (2.52), we have

$$\begin{aligned} \bar{t} &= T_{\bar{\mu}}^m J - J^* \\ &= (T_{\bar{\mu}}^m J - T_{\bar{\mu}}^{m-1} J) + \cdots + (T_{\bar{\mu}}^2 J - T_{\bar{\mu}} J) + (T_{\bar{\mu}} J - TJ) + (TJ - TJ^*) \\ &\leq (\alpha^{m-1} + \cdots + \alpha) M(T_{\bar{\mu}} J - J)v + \epsilon v + \alpha\delta v + \alpha M(t)v, \end{aligned}$$

so finally

$$M(\bar{t}) \leq \frac{\alpha - \alpha^m}{1-\alpha} M(\bar{r}) + (\epsilon + \alpha\delta) + \alpha M(t).$$

By taking lim sup of both sides and using Eq. (2.54), it follows that

$$\limsup_{k\to\infty} M(t) \leq \frac{(\alpha-\beta)\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)^2(1-\beta)} + \frac{\epsilon + \alpha\delta}{1-\alpha}. \tag{2.56}$$

We now combine Eqs. (2.53), (2.55), and (2.56). We obtain

$$\begin{aligned}\limsup_{k\to\infty} M(J_{\mu^k} - J^*) &\le \limsup_{k\to\infty} M(s) + \limsup_{k\to\infty} M(t)\\ &\le \frac{\beta\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)(1-\beta)} + \frac{(\alpha-\beta)\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)^2(1-\beta)} + \frac{\epsilon+\alpha\delta}{1-\alpha}\\ &= \frac{\big(\beta(1-\alpha) + (\alpha-\beta)\big)\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)^2(1-\beta)} + \frac{\epsilon+\alpha\delta}{1-\alpha}\\ &= \frac{\alpha\big(\epsilon + (1+\alpha)\delta\big)}{(1-\alpha)^2} + \frac{\epsilon+\alpha\delta}{1-\alpha}\\ &= \frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}.\end{aligned}$$

This proves the result, since in view of $J_{\mu^k} \ge J^*$, we have $M(J_{\mu^k} - J^*) = \|J_{\mu^k} - J^*\|$. **Q.E.D.**

A remarkable fact is that approximate VI, approximate PI, and approximate optimistic PI have very similar error bounds (cf. Props. 2.3.2, 2.4.3, and 2.5.3). Approximate VI has a slightly better bound, but insignificantly so in practical terms. When approximate PI produces a convergent sequence of policies, the associated error bound is much better (cf. Prop. 2.4.5). However, special conditions are needed for convergence of policies in approximate PI. These conditions are fulfilled in some cases, notably including schemes where aggregation is used for policy evaluation (cf. Section 1.2.4). In other cases, including some where the projected equation is used for policy evaluation, approximate PI (both optimistic and nonoptimistic) will typically generate a cycle of policies satisfying the bound of Prop. 2.4.3; see Section 3.6 of the PI survey paper [Ber11c], or Chapter 6 of the book [Ber12a].

2.5.3 Randomized Optimistic Policy Iteration

We will now consider a randomized version of the optimistic PI algorithm where the number m_k of VI iterations in the kth policy evaluation is random, while the monotonicity assumption need not hold. We assume, however, that each policy mapping is a contraction in a suitable space, that the number of policies is finite, and that $m_k = 1$ with positive probability (these assumptions can be modified and/or generalized in ways suggested by the subsequent line of proof). In particular, for each positive integer j, we have a probability $p(j) \ge 0$, where

$$p(1) > 0, \qquad \sum_{j=1}^{\infty} p(j) = 1.$$

We consider the algorithm

$$T_{\mu^k} J_k = T J_k, \qquad J_{k+1} = T_{\mu^k}^{m_k} J_k, \qquad k = 0, 1, \ldots, \tag{2.57}$$

where m_k is chosen randomly according to the distribution $p(j)$,

$$P(m_k = j) = p(j), \qquad j = 1, 2, \ldots. \tag{2.58}$$

The selection of m_k is independent of previous selections. We will assume the following.

Assumption 2.5.2: Let $\|\cdot\|$ be a norm on some complete space of real-valued functions over X, denoted $\mathcal{F}(X)$, and assume the following.

(a) The set of policies $\mathcal{M}$ is finite.

(b) The mappings T_μ, $\mu \in \mathcal{M}$, and T are contraction mappings from $\mathcal{F}(X)$ into $\mathcal{F}(X)$.

The preceding assumption requires that the number of policies is finite, but does not require any monotonicity condition (cf. Assumption 2.1.1), while its contraction condition (b) is weaker than the contraction Assumption 2.1.2 since $\mathcal{F}(X)$ is a general complete normed space, not necessarily $\mathcal{B}(X)$. This flexibility may be useful in algorithms that involve cost function approximation within a subspace of basis functions. For such algorithms, however, T does not necessarily have a unique fixed point, as discussed in Section 1.2.4. By contrast since $\mathcal{F}(X)$ is assumed complete, Assumption 2.5.2 implies that T_μ and T have unique fixed points, which we denote by J_μ and J^*, respectively.

An important preliminary fact (which relies on the finiteness of $\mathcal{M}$) is given in the following proposition. The proposition implies that near J^* the generated policies μ^k are "optimal" in the sense that $J_{\mu^k} = J^*$, so the algorithm does not tend to cycle. †

Proposition 2.5.4: Let Assumption 2.5.2 hold, and let $\mathcal{M}^*$ be the subset of all $\mu \in \mathcal{M}$ such that $T_\mu J^* = T J^*$. Then for all $\mu \in \mathcal{M}^*$, we have $J_\mu = J^*$. Moreover, there exists an $\epsilon > 0$ such that for all J with $\|J - J^*\| < \epsilon$ we have $T_\mu J = T J$ only if $\mu \in \mathcal{M}^*$.

Proof: If $\mu \in \mathcal{M}^*$, we have $T_\mu J^* = T J^* = J^*$. Thus J^* is the unique fixed point J_μ of T_μ, and we have $J_\mu = J^*$.

† Note that without monotonicity, J^* need not have any formal optimality properties (cf. the discussion of Section 2.1 and Example 2.1.1).

To prove the second assertion, we argue by contradiction, so we assume that there exist a sequence of scalars $\{\epsilon_k\}$ and a sequence of policies $\{\mu^k\}$ such that $\epsilon_k \downarrow 0$ and

$$\mu^k \notin \mathcal{M}^*, \qquad T_{\mu^k} J_k = TJ_k, \qquad \|J_k - J^*\| < \epsilon_k, \qquad \forall\ k = 0, 1, \ldots.$$

Since $\mathcal{M}$ is finite, we may assume without loss of generality that for some $\overline{\mu} \notin \mathcal{M}^*$, we have $\mu^k = \overline{\mu}$ for all k, so from the preceding relation we have

$$T_{\overline{\mu}} J_k = TJ_k, \qquad \|J_k - J^*\| < \epsilon_k, \qquad \forall\ k = 0, 1, \ldots.$$

Thus $\|J_k - J^*\| \to 0$, and by the contraction Assumption 2.5.2(b), we have

$$\|T_{\overline{\mu}} J_k - T_{\overline{\mu}} J^*\| \to 0, \qquad \|TJ_k - TJ^*\| \to 0.$$

Since $T_{\overline{\mu}} J_k = TJ_k$, the limits of $\{T_{\overline{\mu}} J_k\}$ and $\{TJ_k\}$ are equal, i.e., $T_{\overline{\mu}} J^* = TJ^* = J^*$. Since $J_{\overline{\mu}}$ is the unique fixed point of $T_{\overline{\mu}}$ over $\mathcal{F}(X)$, it follows that $J_{\overline{\mu}} = J^*$, contradicting the earlier hypothesis that $\overline{\mu} \notin \mathcal{M}^*$. **Q.E.D.**

The preceding proof illustrates the key idea of the randomized optimistic PI algorithm, which is that for $\mu \in \mathcal{M}^*$, the mappings $T_\mu^{m_k}$ have a common fixed point that is equal to J^*, the fixed point of T. Thus within a distance ϵ from J^*, the iterates (2.57) aim consistently at J^*. Moreover, because the probability of a VI (an iteration with $m_k = 1$) is positive, the algorithm is guaranteed to eventually come within ϵ from J^* through a sufficiently long sequence of contiguous VI iterations. For this we need the sequence $\{J_k\}$ to be bounded, which will be shown as part of the proof of the following proposition.

Proposition 2.5.5: Let Assumption 2.5.2 hold. Then for any starting point $J_0 \in \mathcal{F}(X)$, a sequence $\{J_k\}$ generated by the randomized optimistic PI algorithm (2.57)-(2.58) belongs to $\mathcal{F}(X)$ and converges to J^* with probability one.

Proof: We will show that $\{J_k\}$ is bounded by showing that for all k, we have

$$\max_{\mu \in \mathcal{M}} \|J_k - J_\mu\| \leq \rho^k \max_{\mu \in \mathcal{M}} \|J_0 - J_\mu\| + \frac{2}{1-\rho} \max_{\mu, \mu' \in \mathcal{M}} \|J_\mu - J_{\mu'}\|, \tag{2.59}$$

where ρ is a common contraction modulus of T_μ, $\mu \in \mathcal{M}$, and T. Indeed, we have for all $\mu \in \mathcal{M}$

$$\begin{aligned}
\|J_k - J_\mu\| &\leq \|J_k - J_{\mu^{k-1}}\| + \|J_{\mu^{k-1}} - J_\mu\| \\
&= \|T_{\mu^{k-1}}^{m_{k-1}} J_{k-1} - J_{\mu^{k-1}}\| + \|J_{\mu^{k-1}} - J_\mu\| \\
&\leq \rho^{m_{k-1}} \|J_{k-1} - J_{\mu^{k-1}}\| + \|J_{\mu^{k-1}} - J_\mu\| \\
&\leq \rho^{m_{k-1}} \big(\|J_{k-1} - J_\mu\| + \|J_\mu - J_{\mu^{k-1}}\|\big) + \|J_{\mu^{k-1}} - J_\mu\| \\
&\leq \rho \max_{\mu \in \mathcal{M}} \|J_{k-1} - J_\mu\| + 2 \max_{\mu, \mu' \in \mathcal{M}} \|J_\mu - J_{\mu'}\|,
\end{aligned}$$

and finally, for all k,

$$\max_{\mu\in\mathcal{M}} \|J_k - J_\mu\| \leq \rho \max_{\mu\in\mathcal{M}} \|J_{k-1} - J_\mu\| + 2 \max_{\mu,\mu'\in\mathcal{M}} \|J_\mu - J_{\mu'}\|.$$

From this relation, we obtain Eq. (2.59) by induction.

Thus in conclusion, we have $\{J_k\} \subset D$, where D is the bounded set

$$D = \left\{ J \;\middle|\; \max_{\mu\in\mathcal{M}} \|J - J_\mu\| \leq \max_{\mu\in\mathcal{M}} \|J_0 - J_\mu\| + \frac{2}{1-\rho} \max_{\mu,\mu'\in\mathcal{M}} \|J_\mu - J_{\mu'}\| \right\}.$$

We use this fact to argue that with enough contiguous value iterations, i.e., iterations where $m_k = 1$, J_k can be brought arbitrarily close to J^*, and once this happens, the algorithm operates like the ordinary VI algorithm.

Indeed, each time the iteration $J_{k+1} = TJ_k$ is performed (i.e., when $m_k = 1$), the distance of the iterate J_k from J^* is reduced by a factor ρ, i.e., $\|J_{k+1} - J^*\| \leq \rho\|J_k - J^*\|$. Since $\{J_k\}$ belongs to the bounded set D, and our randomization scheme includes the condition $p(1) > 0$, the algorithm is guaranteed (with probability one) to eventually execute a sufficient number of contiguous iterations $J_{k+1} = TJ_k$ to enter a sphere

$$S_\epsilon = \big\{ J \in \mathcal{F}(X) \mid \|J - J^*\| < \epsilon \big\}$$

of small enough radius ϵ to guarantee that the generated policy μ^k belongs to $\mathcal{M}^*$, as per Prop. 2.5.4. Once this happens, all subsequent iterations reduce the distance $\|J_k - J^*\|$ by a factor ρ at every iteration, since

$$\big\|T_\mu^m J - J^*\| \leq \rho\|T_\mu^{m-1} J - J^*\| \leq \rho\|J - J^*\|, \quad \forall\ \mu \in \mathcal{M}^*,\ m \geq 1,\ J \in S_\epsilon.$$

Thus once $\{J_k\}$ enters S_ϵ, it stays within S_ϵ and converges to J^*. **Q.E.D.**

A Randomized Version of λ-Policy Iteration

We now turn to the λ-PI algorithm. Instead of the nonrandomized version

$$T_{\mu^k} J_k = TJ_k, \qquad J_{k+1} = T_{\mu^k}^{(\lambda)} J_k, \qquad k = 0, 1, \ldots,$$

cf. Eq. (2.34), we consider a randomized version that involves a fixed probability $p \in (0, 1)$. It has the form

$$T_{\mu^k} J_k = TJ_k, \qquad J_{k+1} = \begin{cases} TJ_k & \text{with probability } p, \\ T_{\mu^k}^{(\lambda)} J_k & \text{with probability } 1 - p. \end{cases} \tag{2.60}$$

The idea of the algorithm is similar to the one of the randomized optimistic PI algorithm (2.57)-(2.58). Under the assumptions of Prop.

2.5.5, the sequence $\{J_k\}$ generated by the randomized λ-PI algorithm (2.60) belongs to $\mathcal{F}(X)$ and converges to J^* with probability one. The reason is that the contraction property of T_μ over $\mathcal{F}(X)$ with respect to the norm $\|\cdot\|$ implies that $T_\mu^{(\lambda)}$ is well-defined, and also implies that $T_\mu^{(\lambda)}$ is a contraction over $\mathcal{F}(X)$. The latter assertion follows from the calculation

$$\begin{aligned}\left\|T_\mu^{(\lambda)}J - J_\mu\right\| &= \left\|(1-\lambda)\sum_{\ell=0}^{\infty}\lambda^\ell T_\mu^{\ell+1}J - (1-\lambda)\sum_{\ell=0}^{\infty}\lambda^\ell J_\mu\right\| \\ &\le (1-\lambda)\sum_{\ell=0}^{\infty}\lambda^\ell\left\|T_\mu^{\ell+1}J - J_\mu\right\| \\ &\le (1-\lambda)\sum_{\ell=0}^{\infty}\lambda^\ell\rho^{\ell+1}\|J - J_\mu\| \\ &= \rho\,\|J - J_\mu\|,\end{aligned}$$

where the first inequality follows from the triangle inequality, and the second inequality follows from the contraction property of T_μ. Given that $T_\mu^{(\lambda)}$ is a contraction, the proof of Prop. 2.5.5 goes through with minimal changes. The idea again is that $\{J_k\}$ remains bounded, and through a sufficiently long sequence of contiguous iterations where the iteration $x_{k+1} = TJ_k$ is performed, it enters the sphere S_ϵ, and subsequently stays within S_ϵ and converges to J^*.

The convergence argument just given suggests that the choice of the randomization probability p is important. If p is too small, convergence may be slow because oscillatory behavior may go unchecked for a long time. On the other hand if p is large, a correspondingly large number of fixed point iterations $x_{k+1} = TJ_k$ may be performed, and the hoped for benefits of the use of the proximal iterations $x_{k+1} = T_{\mu^k}^{(\lambda)}J_k$ may be lost. Adaptive schemes that adjust p based on algorithmic progress may address this issue. Similarly, the choice of the probability $p(1)$ is significant in the randomized optimistic PI algorithm (2.57)-(2.58).

2.6 ASYNCHRONOUS ALGORITHMS

In this section, we extend further the computational methods of VI and PI for abstract DP models, by embedding them within an asynchronous computation framework.

2.6.1 Asynchronous Value Iteration

Each VI of the form given in Section 2.3 applies the mapping T defined by

$$(TJ)(x) = \inf_{u\in U(x)} H(x,u,J), \qquad \forall\ x \in X,$$

for all states simultaneously, thereby producing the sequence $TJ, T^2J, \ldots$ starting with some $J \in \mathcal{B}(X)$. In a more general form of VI, at any one iteration, $J(x)$ may be updated and replaced by $(TJ)(x)$ only for a subset of states. An example is the Gauss-Seidel method for the finite-state case, where at each iteration, $J(x)$ is updated only for a single selected state $\overline{x}$ and $J(x)$ is left unchanged for all other states $x \neq \overline{x}$ (see [Ber12a]). In that method the states are taken up for iteration in a cyclic order, but more complex iteration orders are possible, deterministic as well as randomized.

Methods of the type just described are called *asynchronous VI methods* and may be motivated by several considerations such as:

(a) *Faster convergence*. Generally, computational experience with DP as well as analysis, have shown that convergence is accelerated by incorporating the results of VI updates for some states as early as possible into subsequent VI updates for other states. This is known as the *Gauss-Seidel effect*, which is discussed in some detail in the book [BeT89].

(b) *Parallel and distributed asynchronous computation*. In this context, we have several processors, each applying VI for a subset of states, and communicating the results to other processors (perhaps with some delay). One objective here may be faster computation by taking advantage of parallelism. Another objective may be computational convenience in problems where useful information is generated and processed locally at geographically dispersed points. An example is data or sensor network computations, where nodes, gateways, sensors, and data collection centers collaborate to route and control the flow of data, using DP or shortest path-type computations.

(c) *Simulation-based implementations*. In simulation-based versions of VI, iterations at various states are often performed in the order that the states are generated by some form of simulation.

With these contexts in mind, we introduce a model of asynchronous distributed solution of abstract fixed point problems of the form $J = TJ$. Let $\mathcal{R}(X)$ be the set of real-valued functions defined on some given set X and let T map $\mathcal{R}(X)$ into $\mathcal{R}(X)$. We consider a partition of X into disjoint nonempty subsets $X_1, \ldots, X_m$, and a corresponding partition of J as $J = (J_1, \ldots, J_m)$, where J_ℓ is the restriction of J on the set X_ℓ. Our computation framework involves a network of m processors, each updating corresponding components of J. In a (synchronous) distributed VI algorithm, processor ℓ updates J_ℓ at iteration t according to

$$J_\ell^{t+1}(x) = T(J_1^t, \ldots, J_m^t)(x), \qquad \forall\ x \in X_\ell,\ \ell = 1, \ldots, m.$$

Here to accommodate the distributed algorithmic framework and its overloaded notation, we will use superscript t to denote iterations/times where

some (but not all) processors update their corresponding components, reserving the index k for computation stages involving all processors, and also reserving subscript ℓ to denote component/processor index.

In an asynchronous VI algorithm, processor ℓ updates J_ℓ only for t in a selected subset $\mathcal{R}_\ell$ of iterations, and with components J_j, $j \neq \ell$, supplied by other processors with communication "delays" $t - \tau_{\ell j}(t)$,

$$J_\ell^{t+1}(x) = \begin{cases} T\left(J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\right)(x) & \text{if } t \in \mathcal{R}_\ell,\ x \in X_\ell, \\ J_\ell^t(x) & \text{if } t \notin \mathcal{R}_\ell,\ x \in X_\ell. \end{cases} \tag{2.61}$$

Communication delays arise naturally in the context of asynchronous distributed computing systems of the type described in many sources (an extensive reference is the book [BeT89]). Such systems are interesting for solution of large DP problems, particularly for methods that are based on simulation, which is naturally well-suited for distributed computation. On the other hand, if the entire algorithm is centralized at a single physical processor, the algorithm (2.61) ordinarily will not involve communication delays, i.e., $\tau_{\ell j}(t) = t$ for all ℓ, j, and t.

The simpler case where X is a finite set and each subset X_ℓ consists of a single element ℓ arises often, particularly in the context of simulation. In this case we may simplify the notation of iteration (2.61) by writing J_ℓ^t in place of the scalar component $J_\ell^t(\ell)$, as we do in the following example.

Example 2.6.1 (One-State-at-a-Time Iterations)

Assuming $X = \{1, \ldots, n\}$, let us view each state as a processor by itself, so that $X_\ell = \{\ell\}$, $\ell = 1, \ldots, n$. Consider a VI algorithm that executes one-state-at-a-time, according to some state sequence $\{x^0, x^1, \ldots\}$, which is generated in some way, possibly by simulation. Thus, starting from some initial vector J^0, we generate a sequence $\{J^t\}$, with $J^t = (J_1^t, \ldots, J_n^t)$, as follows:

$$J_\ell^{t+1} = \begin{cases} T(J_1^t, \ldots, J_n^t)(\ell) & \text{if } \ell = x^t, \\ J_\ell^t & \text{if } \ell \neq x^t, \end{cases}$$

where $T(J_1^t, \ldots, J_n^t)(\ell)$ denotes the ℓ-th component of the vector

$$T(J_1^t, \ldots, J_n^t) = TJ^t,$$

and for simplicity we write J_ℓ^t instead of $J_\ell^t(\ell)$. This algorithm is a special case of iteration (2.61) where the set of times at which J_ℓ is updated is

$$\mathcal{R}_\ell = \{t \mid x^t = \ell\},$$

and there are no communication delays (as in the case where the entire algorithm is centralized at a single physical processor).

Note also that if X is finite, we can assume without loss of generality that each state is assigned to a separate processor. The reason is that a physical processor that updates a group of states may be replaced by a group of fictitious processors, each assigned to a single state, and updating their corresponding components of J simultaneously.

We will now discuss the convergence of the asynchronous algorithm (2.61). To this end we introduce the following assumption.

Assumption 2.6.1: (Continuous Updating and Information Renewal)

(1) The set of times $\mathcal{R}_\ell$ at which processor ℓ updates J_ℓ is infinite, for each $\ell = 1, \ldots, m$.

(2) $\lim_{t\to\infty} \tau_{\ell j}(t) = \infty$ for all $\ell, j = 1, \ldots, m$.

Assumption 2.6.1 is natural, and is essential for any kind of convergence result about the algorithm. † In particular, the condition $\tau_{\ell j}(t) \to \infty$ guarantees that outdated information about the processor updates will eventually be purged from the computation. It is also natural to assume that $\tau_{\ell j}(t)$ is monotonically increasing with t, but this assumption is not necessary for the subsequent analysis.

We wish to show that $J_\ell^t \to J_\ell^*$ for all ℓ, and to this end we employ the following convergence theorem for totally asynchronous iterations from the author's paper [Ber83], which has served as the basis for the treatment of totally asynchronous iterations in the book [BeT89] (Chapter 6), and their application to DP (i.e., VI and PI), and asynchronous gradient-based optimization. For the statement of the theorem, we say that a sequence $\{J^k\} \subset \mathcal{R}(X)$ *converges pointwise* to $J \in \mathcal{R}(X)$ if

$$\lim_{k\to\infty} J^k(x) = J(x)$$

for all $x \in X$.

Proposition 2.6.1 (Asynchronous Convergence Theorem): Let T have a unique fixed point J^*, let Assumption 2.6.1 hold, and assume that there is a sequence of nonempty subsets $\big\{S(k)\big\} \subset \mathcal{R}(X)$ with

† Generally, convergent distributed iterative asynchronous algorithms are classified in *totally and partially asynchronous* [cf. the book [BeT89] (Chapters 6 and 7), or the more recent survey in the book [Ber16c] (Section 2.5)]. In the former, there is no bound on the communication delays, while in the latter there must be a bound (which may be unknown). The algorithms of the present section are totally asynchronous, as reflected by Assumption 2.6.1.

$$S(k+1) \subset S(k), \quad k = 0, 1, \ldots,$$

and is such that if $\{V^k\}$ is any sequence with $V^k \in S(k)$, for all $k \geq 0$, then $\{V^k\}$ converges pointwise to J^*. Assume further the following:

(1) *Synchronous Convergence Condition:* We have

$$TJ \in S(k+1), \qquad \forall\ J \in S(k),\ k = 0, 1, \ldots.$$

(2) *Box Condition:* For all k, $S(k)$ is a Cartesian product of the form

$$S(k) = S_1(k) \times \cdots \times S_m(k),$$

where $S_\ell(k)$ is a set of real-valued functions on X_ℓ, $\ell = 1, \ldots, m$.

Then for every $J^0 \in S(0)$, the sequence $\{J^t\}$ generated by the asynchronous algorithm (2.61) converges pointwise to J^*.

Proof: To explain the idea of the proof, let us note that the given conditions imply that updating any component J_ℓ, by applying T to a function $J \in S(k)$, while leaving all other components unchanged, yields a function in $S(k)$. Thus, once enough time passes so that the delays become "irrelevant," then after J enters $S(k)$, it stays within $S(k)$. Moreover, once a component J_ℓ enters the subset $S_\ell(k)$ and the delays become "irrelevant," J_ℓ gets permanently within the smaller subset $S_\ell(k+1)$ at the first time that J_ℓ is iterated on with $J \in S(k)$. Once each component J_ℓ, $\ell = 1, \ldots, m$, gets within $S_\ell(k+1)$, the entire function J is within $S(k+1)$ by the Box Condition. Thus the iterates from $S(k)$ eventually get into $S(k+1)$ and so on, and converge pointwise to J^* in view of the assumed properties of $\{S(k)\}$.

With this idea in mind, we show by induction that for each $k \geq 0$, there is a time t_k such that:

(1) $J^t \in S(k)$ for all $t \geq t_k$.

(2) For all ℓ and $t \in \mathcal{R}_\ell$ with $t \geq t_k$, we have

$$\left(J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\right) \in S(k).$$

[In words, after some time, all fixed point estimates will be in $S(k)$ and all estimates used in iteration (2.61) will come from $S(k)$.]

The induction hypothesis is true for $k = 0$ since $J^0 \in S(0)$. Assuming it is true for a given k, we will show that there exists a time t_{k+1} with the required properties. For each $\ell = 1, \ldots, m$, let $t(\ell)$ be the first element of $\mathcal{R}_\ell$ such that $t(\ell) \geq t_k$. Then by the Synchronous Convergence Condition,

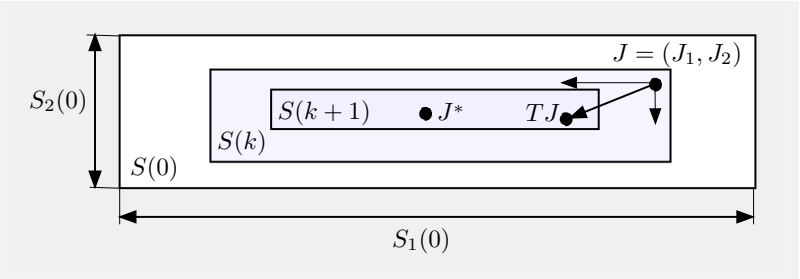

Figure 2.6.1 Geometric interpretation of the conditions of asynchronous convergence theorem. We have a nested sequence of boxes $\{S(k)\}$ such that $TJ \in S(k+1)$ for all $J \in S(k)$.

we have $TJ^{t(\ell)} \in S(k+1)$, implying (in view of the Box Condition) that

$$J_\ell^{t(\ell)+1} \in S_\ell(k+1).$$

Similarly, for every $t \in \mathcal{R}_\ell$, $t \geq t(\ell)$, we have $J_\ell^{t+1} \in S_\ell(k+1)$. Between elements of $\mathcal{R}_\ell$, J_ℓ^t does not change. Thus,

$$J_\ell^t \in S_\ell(k+1), \qquad \forall\, t \geq t(\ell)+1.$$

Let $t'_k = \max_\ell \big\{t(\ell)\big\} + 1$. Then, using the Box Condition we have

$$J^t \in S(k+1), \qquad \forall\, t \geq t'_k.$$

Finally, since by Assumption 2.6.1, we have $\tau_{\ell j}(t) \to \infty$ as $t \to \infty$, $t \in \mathcal{R}_\ell$, we can choose a time $t_{k+1} \geq t'_k$ that is sufficiently large so that $\tau_{\ell j}(t) \geq t'_k$ for all ℓ, j, and $t \in \mathcal{R}_\ell$ with $t \geq t_{k+1}$. We then have, for all $t \in \mathcal{R}_\ell$ with $t \geq t_{k+1}$ and $j = 1, \ldots, m$, $J_j^{\tau_{j\ell}(t)} \in S_j(k+1)$, which (by the Box Condition) implies that

$$\Big(J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\Big) \in S(k+1).$$

The induction is complete. **Q.E.D.**

Figure 2.6.1 illustrates the assumptions of the preceding convergence theorem. The challenge in applying the theorem is to identify the set sequence $\big\{S(k)\big\}$ and to verify the assumptions of Prop. 2.6.1. In abstract DP, these assumptions are satisfied in two primary contexts of interest. The first is when $S(k)$ are weighted sup-norm spheres centered at J^*, and can be used in conjunction with the contraction framework of the preceding section (see the following proposition). The second context is based on monotonicity conditions. It will be used in Section 3.6 in conjunction

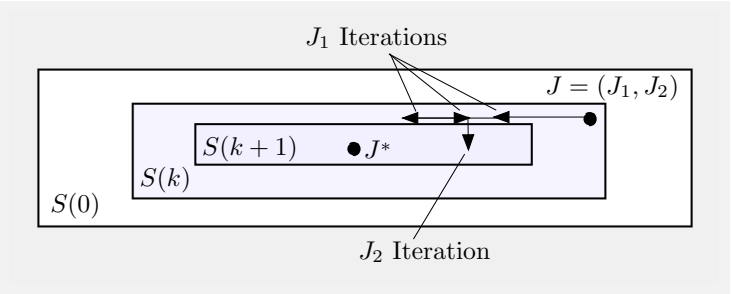

Figure 2.6.2 Geometric interpretation of the mechanism for asynchronous convergence. Iteration on a single component of a function $J \in S(k)$, say J_ℓ, keeps J in $S(k)$, while it moves J_ℓ into the corresponding component $S_\ell(k+1)$ of $S(k+1)$, where it remains throughout the subsequent iterations. Once all components J_ℓ have been iterated on at least once, the iterate is guaranteed to be in $S(k+1)$.

with semicontractive models for which there is no underlying sup-norm contraction. It is also relevant to the noncontractive models of Section 4.3 where again there is no underlying contraction. Figure 2.6.2 illustrates the mechanism by which asynchronous convergence is achieved.

We note a few extensions of the theorem. It is possible to allow T to be time-varying, so in place of T we operate with a sequence of mappings T_k, $k = 0, 1, \ldots$. Then if all T_k have a common fixed point J^*, the conclusion of the theorem holds (see Exercise 2.2 for a more precise statement). This extension is useful in some of the algorithms to be discussed later. Another extension is to allow T to have multiple fixed points and introduce an assumption that roughly says that $\cap_{k=0}^{\infty} S(k)$ is the set of fixed points. Then the conclusion is that any limit point (in an appropriate sense) of $\{J^t\}$ is a fixed point.

We now apply the preceding convergence theorem to the totally asynchronous VI algorithm under the contraction assumption. Note that the monotonicity Assumption 2.1.1 is not necessary (just like it is not needed for the synchronous convergence of $\{T^k J\}$ to J^*).

Proposition 2.6.2: Let the contraction Assumption 2.1.2 hold, together with Assumption 2.6.1. Then if $J^0 \in \mathcal{B}(X)$, a sequence $\{J^t\}$ generated by the asynchronous VI algorithm (2.61) converges pointwise to J^*.

Proof: We apply Prop. 2.6.1 with

$$S(k) = \big\{J \in \mathcal{B}(X) \mid \|J^k - J^*\| \leq \alpha^k \|J^0 - J^*\|\big\}, \qquad k = 0, 1, \ldots.$$

Since T is a contraction with modulus α, the synchronous convergence

condition is satisfied. Since T is a weighted sup-norm contraction, the box condition is also satisfied, and the result follows. **Q.E.D.**

2.6.2 Asynchronous Policy Iteration

We will now develop asynchronous PI algorithms that have comparable properties to the asynchronous VI algorithm of the preceding subsection. The processors collectively maintain and update an estimate J^t of the optimal cost function, and an estimate μ^t of an optimal policy. The local portions of J^t and μ^t of processor ℓ are denoted J_ℓ^t and μ_ℓ^t, respectively, i.e., $J_\ell^t(x) = J^t(x)$ and $\mu_\ell^t(x) = \mu^t(x)$ for all $x \in X_\ell$.

For each processor ℓ, there are two disjoint subsets of times $\mathcal{R}_\ell, \overline{\mathcal{R}}_\ell \subset \{0, 1, \ldots\}$, corresponding to policy improvement and policy evaluation iterations, respectively. At the times $t \in \mathcal{R}_\ell \cup \overline{\mathcal{R}}_\ell$, the local cost function J_ℓ^t of processor ℓ is updated using "delayed" local costs $J_j^{\tau_{\ell j}(t)}$ of other processors $j \neq \ell$, where $0 \leq \tau_{\ell j}(t) \leq t$. At the times $t \in \mathcal{R}_\ell$ (the local policy improvement times), the local policy μ_ℓ^t is also updated. For various choices of $\mathcal{R}_\ell$ and $\overline{\mathcal{R}}_\ell$, the algorithm takes the character of VI (when $\mathcal{R}_\ell = \{0, 1, \ldots\}$), and PI (when $\overline{\mathcal{R}}_\ell$ contains a large number of time indices between successive elements of $\mathcal{R}_\ell$). As before, we view $t - \tau_{\ell j}(t)$ as a "communication delay," and we require Assumption 2.6.1. †

In a natural asynchronous version of optimistic PI, at each time t, each processor ℓ does one of the following:

(a) *Local policy improvement*: If $t \in \mathcal{R}_\ell$, processor ℓ sets for all $x \in X_\ell$,

$$J_\ell^{t+1}(x) = \min_{u \in U(x)} H\big(x, u, J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\big), \tag{2.62}$$

$$\mu_\ell^{t+1}(x) \in \arg\min_{u \in U(x)} H\big(x, u, J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\big). \tag{2.63}$$

(b) *Local policy evaluation*: If $t \in \overline{\mathcal{R}}_\ell$, processor ℓ sets for all $x \in X_\ell$,

$$J_\ell^{t+1}(x) = H\big(x, \mu^t(x), J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)}\big), \tag{2.64}$$

and leaves μ_ℓ unchanged, i.e., $\mu_\ell^{t+1}(x) = \mu_\ell^t(x)$ for all $x \in X_\ell$.

(c) *No local change*: If $t \notin \mathcal{R}_\ell \cup \overline{\mathcal{R}}_\ell$, processor ℓ leaves J_ℓ and μ_ℓ unchanged, i.e., $J_\ell^{t+1}(x) = J_\ell^t(x)$ and $\mu_\ell^{t+1}(x) = \mu_\ell^t(x)$ for all $x \in X_\ell$.

Unfortunately, even when implemented without the delays $\tau_{\ell j}(t)$, the preceding PI algorithm is unreliable. The difficulty is that the algorithm

† As earlier in all PI algorithms we assume that the infimum over $u \in U(x)$ in the policy improvement operation is attained, and we write min in place of inf.

involves a mix of applications of T and various mappings T_μ that have different fixed points, so in the absence of some systematic tendency towards J^* there is the possibility of oscillation (see Fig. 2.6.3). While this does not happen in synchronous versions (cf. Prop. 2.5.1), asynchronous versions of the algorithm (2.33) may oscillate unless J^0 satisfies some special condition (examples of this type of oscillation have been constructed in the paper [WiB93]; see also [Ber10], which translates an example from [WiB93] to the notation of the present book).

In this subsection and the next we will develop two distributed asynchronous PI algorithms, each embodying a distinct mechanism that precludes the oscillatory behavior just described. In the first algorithm, there is a simple randomization scheme, according to which a policy evaluation of the form (2.64) is replaced by a policy improvement (2.62)-(2.63) with some positive probability. In the second algorithm, given in Section 2.6.3, we introduce a mapping F_μ, which has a common fixed point property: its fixed point is related to J^* and is the same for all μ, so the anomaly illustrated in Fig. 2.6.3 cannot occur. The first algorithm is simple but requires some restrictions, including that the set of policies is finite. The second algorithm is more sophisticated and does not require this restriction. Both of these algorithms *do not require the monotonicity assumption*.

An Optimistic Asynchronous PI Algorithm with Randomization

We introduce a randomization scheme for avoiding oscillatory behavior. It is defined by a small probability $p > 0$, according to which a policy evaluation iteration is replaced by a policy improvement iteration with probability p, independently of the results of past iterations. We model this randomization by assuming that before the algorithm is started, we restructure the sets $\mathcal{R}_\ell$ and $\overline{\mathcal{R}}_\ell$ as follows: we take each element of each set $\overline{\mathcal{R}}_\ell$, and with probability p, remove it from $\overline{\mathcal{R}}_\ell$, and add it to $\mathcal{R}_\ell$ (independently of other elements). We will assume the following:

Assumption 2.6.2:

(a) The set of policies $\mathcal{M}$ is finite.

(b) There exists an integer $B \geq 0$ such that

$$(\mathcal{R}_\ell \cup \overline{\mathcal{R}}_\ell) \cap \{\tau \mid t < \tau \leq t + B\} \neq \emptyset, \qquad \forall\ t,\ \ell.$$

(c) There exists an integer $B' \geq 0$ such that

$$0 \leq t - \tau_{\ell j}(t) \leq B', \qquad \forall\ t,\ \ell,\ j.$$

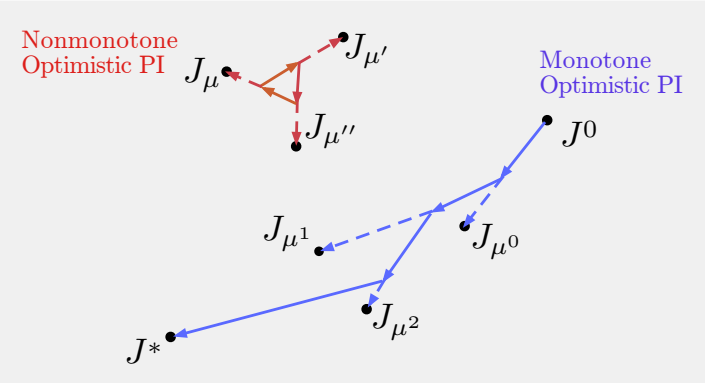

Figure 2.6.3 Illustration of optimistic asynchronous PI under the monotonicity and the contraction assumptions. When started with J^0 and μ^0 satisfying

$$J^0 \geq TJ^0 = T_{\mu^0} J^0,$$

the algorithm converges monotonically to J^* (see the trajectory on the right). However, for other initial conditions, there is a possibility for oscillations, since with changing values of μ, the mappings T_μ have different fixed points and "aim at different targets" (see the trajectory on the left, which illustrates a cycle between three policies μ, μ', μ''). It turns out that such oscillations are not possible when the algorithm is implemented synchronously (cf. Prop. 2.5.1), but may occur in asynchronous implementations.

Assumption 2.6.2 guarantees that each processor ℓ will execute at least one policy evaluation or policy improvement iteration within every block of B consecutive iterations, and places a bound B' on the communication delays. The convergence of the algorithm is shown in the following proposition.

Proposition 2.6.3: Under the contraction Assumption 2.1.2, and Assumptions 2.6.1, and 2.6.2, for the preceding algorithm with randomization, we have

$$\lim_{t\to\infty} J^t(x) = J^*(x), \qquad \forall\, x \in X,$$

with probability one.

Proof: Let J^* and J_μ be the fixed points of T and T_μ, respectively, and denote by $\mathcal{M}^*$ the set of optimal policies:

$$\mathcal{M}^* = \{\mu \in \mathcal{M} \mid J_\mu = J^*\} = \{\mu \in \mathcal{M} \mid T_\mu J^* = TJ^*\}.$$

We will show that the algorithm eventually (with probability one) enters a small neighborhood of J^* within which it remains, generates policies in $\mathcal{M}^*$, becomes equivalent to asynchronous VI, and therefore converges to J^* by Prop. 2.6.2. The idea of the proof is twofold; cf. Props. 2.5.4 and 2.5.5.

(1) There exists a small enough weighted sup-norm sphere centered at J^*, call it $\mathcal{S}^*$, within which policy improvement generates only policies in $\mathcal{M}^*$, so policy evaluation with such policies as well as policy improvement keep the algorithm within $\mathcal{S}^*$ if started there, and reduce the weighted sup-norm distance to J^*, in view of the contraction and common fixed point property of T and T_μ, $\mu \in \mathcal{M}^*$. This is a consequence of Prop. 2.3.1 [cf. Eq. (2.16)].

(2) With probability one, thanks to the randomization device, the algorithm will eventually enter permanently $\mathcal{S}^*$ with a policy in $\mathcal{M}^*$.

We now establish (1) and (2) in suitably refined form to account for the presence of delays and asynchronism. As in the proof of Prop. 2.5.5, we can prove that given J^0, we have that $\{J^t\} \subset D$, where D is a bounded set that depends on J^0. We define

$$S(k) = \big\{J \mid \|J - J^*\| \leq \alpha^k c\big\},$$

where c is sufficiently large so that $D \subset S(0)$. Then $J^t \in D$ and hence $J^t \in S(0)$ for all t.

Let k^* be such that

$$J \in S(k^*) \ \text{ and } \ T_\mu J = TJ \qquad \Rightarrow \qquad \mu \in \mathcal{M}^*. \tag{2.65}$$

Such a k^* exists in view of the finiteness of $\mathcal{M}$ and Prop. 2.3.1 [cf. Eq. (2.16)].

We now claim that with probability one, for any given $k \geq 1$, J^t will eventually enter $S(k)$ and stay within $S(k)$ for at least B' additional consecutive iterations. This is because our randomization scheme is such that for any t and k, with probability at least $p^{k(B+B')}$ the next $k(B+B')$ iterations are policy improvements, so that

$$J^{t+k(B+B')-\xi} \in S(k)$$

for all ξ with $0 \leq \xi < B'$ [if $t \geq B' - 1$, we have $J^{t-\xi} \in S(0)$ for all ξ with $0 \leq \xi < B'$, so $J^{t+B+B'-\xi} \in S(1)$ for $0 \leq \xi < B'$, which implies that $J^{t+2(B+B')-\xi} \in S(2)$ for $0 \leq \xi < B'$, etc].

It follows that with probability one, for some $\overline{t}$ we will have $J^\tau \in S(k^*)$ for all τ with $\overline{t} - B' \leq \tau \leq \overline{t}$, as well as $\mu^{\overline{t}} \in \mathcal{M}^*$ [cf. Eq. (2.65)]. Based on property (2.65) and the definition (2.63)-(2.64) of the algorithm, we see that at the next iteration, we have $\mu^{\overline{t}+1} \in \mathcal{M}^*$ and

$$\|J^{\overline{t}+1} - J^*\| \leq \|J^{\overline{t}} - J^*\| \leq \alpha^{k^*} c,$$

so $J^{\overline{t}+1} \in S(k^*)$; this is because in view of $J_{\mu^{\overline{t}}} = J^*$, and the contraction property of T and $T_{\mu^{\overline{t}}}$, we have

$$\frac{\left|J_\ell^{\overline{t}+1}(x) - J_\ell^*(x)\right|}{v(x)} \leq \alpha \|J^{\overline{t}} - J^*\| \leq \alpha^{k^*+1} c, \tag{2.66}$$

for all $x \in X_\ell$ and ℓ such that $\overline{t} \in \overline{\mathcal{R}}_\ell \cup \mathcal{R}_\ell$, while

$$J^{\overline{t}+1}(x) = J^{\overline{t}}(x)$$

for all other x. Proceeding similarly, it follows that for all $t > \overline{t}$ we will have

$$J^\tau \in S(k^*), \qquad \forall\ t \text{ with } t - B' \leq \tau \leq t,$$

as well as $\mu^t \in \mathcal{M}^*$. Thus, after at most B iterations following $\overline{t}$ [after all components J_ℓ are updated through policy evaluation or policy improvement at least once so that

$$\frac{\left|J_\ell^{t+1}(x) - J_\ell^*(x)\right|}{v(x)} \leq \alpha \|J^{\overline{t}} - J^*\| \leq \alpha^{k^*+1} c,$$

for every i, $x \in X_\ell$, and some t with $\overline{t} \leq t < \overline{t} + B$, cf. Eq. (2.66)], J^t will enter $S(k^*+1)$ permanently, with $\mu^t \in \mathcal{M}^*$ (since $\mu^t \in \mathcal{M}^*$ for all $t \geq \overline{t}$ as shown earlier). Then, with the same reasoning, after at most another $B' + B$ iterations, J^t will enter $S(k^*+2)$ permanently, with $\mu^t \in \mathcal{M}^*$, etc. Thus J^t will converge to J^* with probability one. **Q.E.D.**

The proof of Prop. 2.6.3 shows that eventually (with probability one after some iteration) the algorithm will become equivalent to asynchronous VI (each policy evaluation will produce the same results as a policy improvement), while generating optimal policies exclusively. However, the expected number of iterations for this to happen can be very large. Moreover the proof depends on the set of policies being finite. These observations raise questions regarding the practical effectiveness of the algorithm. However, it appears that for many problems the algorithm works well, particularly when oscillatory behavior is a rare occurrence.

A potentially important issue is the choice of the randomization probability p. If p is too small, convergence may be slow because oscillatory behavior may go unchecked for a long time. On the other hand if p is large, a correspondingly large number of policy improvement iterations may be performed, and the hoped for benefits of optimistic PI may be lost. Adaptive schemes which adjust p based on algorithmic progress may be an interesting possibility for addressing this issue.

2.6.3 Optimistic Asynchronous Policy Iteration with a Uniform Fixed Point

We will now discuss another approach to address the convergence difficulties of the "natural" asynchronous PI algorithm (2.62)-(2.64). As illustrated in Fig. 2.6.3 in connection with optimistic PI, the mappings T and T_μ have different fixed points. As a result, optimistic and distributed PI, which involve an irregular mixture of applications of T_μ and T, do not have a "consistent target" at which to aim.

With this in mind, we introduce a new mapping that is parametrized by μ and has *a common fixed point for all* μ, which in turn yields J^*. This mapping is a weighted sup-norm contraction with modulus α, so it may be used in conjunction with asynchronous VI and PI. An additional benefit is that the monotonicity Assumption 2.1.1 is not needed to prove convergence in the analysis that follows; the contraction Assumption 2.1.2 is sufficient (see Exercise 2.3 for an application).

The mapping operates on a pair (V, Q) where:

- V is a function with a component $V(x)$ for each x (in the DP context it may be viewed as a cost function).
- Q is a function with a component $Q(x, u)$ for each pair (x, u) [in the DP context $Q(x, u)$, is known as a *Q-factor*].

The mapping produces a pair

$$\big(MF_\mu(V, Q), F_\mu(V, Q)\big),$$

where

- $F_\mu(V, Q)$ is a function with a component $F_\mu(V, Q)(x, u)$ for each (x, u), defined by
$$F_\mu(V, Q)(x, u) = H\big(x, u, \min\{V, Q_\mu\}\big), \tag{2.67}$$
where for any Q and μ, we denote by Q_μ the function of x defined by
$$Q_\mu(x) = Q\big(x, \mu(x)\big), \qquad x \in X, \tag{2.68}$$
and for any two functions V_1 and V_2 of x, we denote by $\min\{V_1, V_2\}$ the function of x given by
$$\min\{V_1, V_2\}(x) = \min\big\{V_1(x), V_2(x)\big\}, \qquad x \in X.$$

- $MF_\mu(V, Q)$ is a function with a component $\big(MF_\mu(V, Q)\big)(x)$ for each x, where M denotes minimization over u, so that
$$\big(MF_\mu(V, Q)\big)(x) = \min_{u \in U(x)} F_\mu(V, Q)(x, u). \tag{2.69}$$

Example 2.6.2 (Asynchronous Optimistic Policy Iteration for Discounted Finite-State MDP)

Consider the special case of the finite-state discounted MDP of Example 1.2.2. We have

$$H(x,u,J) = \sum_{y=1}^{n} p_{xy}(u)\big(g(x,u,y) + \alpha J(y)\big),$$

and

$$\begin{aligned} F_\mu(V,Q)(x,u) &= H\big(x,u,\min\{V,Q_\mu\}\big) \\ &= \sum_{y=1}^{n} p_{xy}(u)\Big(g(x,u,y) + \alpha \min\big\{V(y), Q(y,\mu(y))\big\}\Big), \end{aligned}$$

$$\big(MF_\mu(V,Q)\big)(x) = \min_{u\in U(x)} \sum_{y=1}^{n} p_{xy}(u)\Big(g(x,u,y) + \alpha \min\big\{V(y), Q(y,\mu(y))\big\}\Big),$$

[cf. Eqs. (2.67)-(2.69)]. Note that $F_\mu(V,Q)$ is the mapping that defines Bellman's equation for the Q-factors of a policy μ in an optimal stopping problem where the stopping cost at state y is equal to $V(y)$.

We now consider the mapping G_μ given by

$$G_\mu(V,Q) = \big(MF_\mu(V,Q), F_\mu(V,Q)\big), \tag{2.70}$$

and show that it has a uniform contraction property and a corresponding uniform fixed point. To this end, we introduce the norm

$$\big\|(V,Q)\big\| = \max\big\{\|V\|, \|Q\|\big\}$$

in the space of (V,Q), where $\|V\|$ is the weighted sup-norm of V, and $\|Q\|$ is defined by

$$\|Q\| = \sup_{x\in X,\, u\in U(x)} \frac{\big|Q(x,u)\big|}{v(x)}.$$

We have the following proposition.

Proposition 2.6.4: Let the contraction Assumption 2.1.2 hold. Consider the mapping G_μ defined by Eqs. (2.67)-(2.70). Then for all μ:

(a) (J^*, Q^*) is the unique fixed point of G_μ, where Q^* is defined by

$$Q^*(x,u) = H(x,u,J^*), \qquad x \in X,\ u \in U(x). \tag{2.71}$$

(b) The following uniform contraction property holds for all (V,Q) and $(\tilde V, \tilde Q)$:

$$\big\|G_\mu(V,Q) - G_\mu(\tilde V, \tilde Q)\big\| \le \alpha \big\|(V,Q) - (\tilde V, \tilde Q)\big\|.$$

Proof: (a) Using the definition (2.71) of Q^*, we have

$$J^*(x) = (TJ^*)(x) = \inf_{u \in U(x)} H(x,u,J^*) = \inf_{u \in U(x)} Q^*(x,u), \qquad \forall\ x \in X,$$

so that

$$\min\big\{J^*(x),\, Q^*\big(x,\mu(x)\big)\big\} = J^*(x), \qquad \forall\ x \in X,\ \mu \in \mathcal{M}.$$

Using the definition (2.67) of F_μ, it follows that $F_\mu(J^*,Q^*) = Q^*$ and also that $MF_\mu(J^*,Q^*) = J^*$, so (J^*,Q^*) is a fixed point of G_μ for all μ. The uniqueness of this fixed point will follow from the contraction property of part (b).

(b) We first show that for all (V,Q) and $(\tilde V,\tilde Q)$, we have

$$\begin{aligned}\big\|F_\mu(V,Q) - F_\mu(\tilde V,\tilde Q)\big\| &\le \alpha\big\|\min\{V,Q_\mu\} - \min\{\tilde V,\tilde Q_\mu\}\big\| \\ &\le \alpha \max\big\{\|V-\tilde V\|,\, \|Q-\tilde Q\|\big\}.\end{aligned} \tag{2.72}$$

Indeed, the first inequality follows from the definition (2.67) of F_μ and the contraction Assumption 2.1.2. The second inequality follows from a nonexpansiveness property of the minimization map: for any J_1, J_2, $\tilde J_1$, $\tilde J_2$, we have

$$\big\|\min\{J_1,J_2\} - \min\{\tilde J_1,\tilde J_2\}\big\| \le \max\big\{\|J_1-\tilde J_1\|, \|J_2-\tilde J_2\|\big\}; \tag{2.73}$$

[to see this, write for every x,

$$\frac{J_m(x)}{v(x)} \le \max\big\{\|J_1-\tilde J_1\|, \|J_2-\tilde J_2\|\big\} + \frac{\tilde J_m(x)}{v(x)}, \qquad m = 1,2,$$

take the minimum of both sides over m, exchange the roles of J_m and $\tilde J_m$, and take supremum over x]. Here we use the relation (2.73) for $J_1 = V$, $\tilde J_1 = \tilde V$, and $J_2(x) = Q\big(x,\mu(x)\big)$, $\tilde J_2(x) = \tilde Q\big(x,\mu(x)\big)$, for all $x \in X$.

We next note that for all $Q, \tilde Q$, †

$$\|MQ - M\tilde Q\| \le \|Q - \tilde Q\|,$$

† For a proof, we write

$$\frac{Q(x,u)}{v(x)} \le \|Q-\tilde Q\| + \frac{\tilde Q(x,u)}{v(x)}, \qquad \forall\ u \in U(x),\ x \in X,$$

take infimum of both sides over $u \in U(x)$, exchange the roles of Q and $\tilde Q$, and take supremum over $x \in X$ and $u \in U(x)$.

which together with Eq. (2.72) yields

$$\max\big\{\big\|MF_\mu(V,Q)-MF_\mu(\tilde V,\tilde Q)\big\|,\big\|F_\mu(V,Q)-F_\mu(\tilde V,\tilde Q)\big\|\big\} \leq \alpha\max\big\{\|V-\tilde V\|,\,\|Q-\tilde Q\|\big\},$$

or equivalently $\big\|G_\mu(V,Q)-G_\mu(\tilde V,\tilde Q)\big\|\leq\alpha\big\|(V,Q)-(\tilde V,\tilde Q)\big\|$. **Q.E.D.**

Because of the uniform contraction property of Prop. 2.6.4(b), a distributed fixed point iteration, like the VI algorithm of Eq. (2.61), can be used in conjunction with the mapping (2.70) to generate asynchronously a sequence $\big\{(V^t,Q^t)\big\}$ that is guaranteed to converge to (J^*,Q^*) for any sequence $\{\mu^t\}$. This can be verified using the proof of Prop. 2.6.2 (more precisely, a proof that closely parallels the one of that proposition); the mapping (2.70) plays the role of T in Eq. (2.61). †

Asynchronous PI Algorithm

We now describe a PI algorithm, which applies asynchronously the components $MF_\mu(V,Q)$ and $F_\mu(V,Q)$ of the mapping $G_\mu(V,Q)$ of Eq. (2.70). The first component is used for local policy improvement and makes a local update to V and μ, while the second component is used for local policy evaluation and makes a local update to Q. The algorithm draws its validity from the weighted sup-norm contraction property of Prop. 2.6.4(b) and the asynchronous convergence theory (Prop. 2.6.2 and Exercise 2.2).

The algorithm is a modification of the "natural" asynchronous PI algorithm (2.63)-(2.64) [without the "communication delays" $t-\tau_{\ell j}(t)$]. It generates sequences $\{V^t,Q^t,\mu^t\}$, which will be shown to converge, in the sense that $V^t\to J^*$, $Q^t\to Q^*$. Note that this is not the only distributed iterative algorithm that can be constructed using the contraction property of Prop. 2.6.4, because this proposition allows a lot of freedom of choice for the policy μ. The paper by Bertsekas and Yu [BeY12] provides an extensive discussion of alternative possibilities, including stochastic simulation-based iterative algorithms, and algorithms that involve function approximation.

To define the asynchronous computation framework, we consider again m processors, a partition of X into sets $X_1,\ldots,X_m$, and assignment of each subset X_ℓ to a processor $\ell\in\{1,\ldots,m\}$. For each ℓ, there are two infinite disjoint subsets of times $\mathcal{R}_\ell,\overline{\mathcal{R}}_\ell\subset\{0,1,\ldots\}$, corresponding to policy improvement and policy evaluation iterations, respectively. Each processor ℓ operates on $V^t(x)$, $Q^t(x,u)$, and $\mu^t(x)$, only for the states x within its "local" state space X_ℓ. Moreover, to execute the steps (a) and (b) of the algorithm, processor ℓ needs only the values $Q^t\big(x,\mu^t(x)\big)$ of Q^t [which are

† Because F_μ and G_μ depend on μ, which changes as the algorithm progresses, it is necessary to use a minor extension of the asynchronous convergence theorem, given in Exercise 2.2, for the convergence proof.

equal to $Q^t_{\mu^t}(x)$; cf. Eq. (2.68)]. In particular, at each time t, each processor ℓ does one of the following:

(a) *Local policy improvement*: If $t \in \mathcal{R}_\ell$, processor ℓ sets for all $x \in X_\ell$, †

$$V^{t+1}(x) = \min_{u \in U(x)} H\big(x, u, \min\{V^t, Q^t_{\mu^t}\}\big) = \big(MF_{\mu^t}(V^t, Q^t)\big)(x),$$

sets $\mu^{t+1}(x)$ to a u that attains the minimum, and leaves Q unchanged, i.e., $Q^{t+1}(x,u) = Q^t(x,u)$ for all $x \in X_\ell$ and $u \in U(x)$.

(b) *Local policy evaluation*: If $t \in \overline{\mathcal{R}}_\ell$, processor ℓ sets for all $x \in X_\ell$ and $u \in U(x)$,

$$Q^{t+1}(x,u) = H\big(x, u, \min\{V^t, Q^t_{\mu^t}\}\big) = F_{\mu^t}(V^t, Q^t)(x,u),$$

and leaves V and μ unchanged, i.e., $V^{t+1}(x) = V^t(x)$ and $\mu^{t+1}(x) = \mu^t(x)$ for all $x \in X_\ell$.

(c) *No local change*: If $t \notin \mathcal{R}_\ell \cup \overline{\mathcal{R}}_\ell$, processor ℓ leaves Q, V, and μ unchanged, i.e., $Q^{t+1}(x,u) = Q^t(x,u)$ for all $x \in X_\ell$ and $u \in U(x)$, $V^{t+1}(x) = V^t(x)$, and $\mu^{t+1}(x) = \mu^t(x)$ for all $x \in X_\ell$.

Note that while this algorithm does not involve the "communication delays" $t - \tau_{\ell j}(t)$, it can clearly be extended to include them. The reason is that our asynchronous convergence analysis framework in combination with the uniform weighted sup-norm contraction property of Prop. 2.6.4 can tolerate the presence of such delays.

Reduced Space Implementation

The preceding PI algorithm may be used for the calculation of both J^* and Q^*. However, if the objective is just to calculate J^*, a simpler and more efficient algorithm is possible. To this end, we observe that the preceding algorithm can be operated so that it does not require the maintenance of the entire function Q. The reason is that the values $Q^t(x,u)$ with $u \neq \mu^t(x)$ do not appear in the calculations, and hence we need only the values $Q^t_{\mu^t}(x) = Q\big(x, \mu^t(x)\big)$, which we store in a function J^t:

$$J^t(x) = Q\big(x, \mu^t(x)\big).$$

This observation is the basis for the following algorithm.

At each time t and for each processor ℓ:

(a) *Local policy improvement*: If $t \in \mathcal{R}_\ell$, processor ℓ sets for all $x \in X_\ell$,

$$J^{t+1}(x) = V^{t+1}(x) = \min_{u \in U(x)} H\big(x, u, \min\{V^t, J^t\}\big), \tag{2.74}$$

† As earlier we assume that the infimum over $u \in U(x)$ in the policy improvement operation is attained, and we write min in place of inf.

and sets $\mu^{t+1}(x)$ to a u that attains the minimum.

(b) *Local policy evaluation*: If $t \in \overline{\mathcal{R}}_\ell$, processor ℓ sets for all $x \in X_\ell$,

$$J^{t+1}(x) = H\big(x, \mu^t(x), \min\{V^t, J^t\}\big), \tag{2.75}$$

and leaves V and μ unchanged, i.e., for all $x \in X_\ell$,

$$V^{t+1}(x) = V^t(x), \qquad \mu^{t+1}(x) = \mu^t(x).$$

(c) *No local change*: If $t \notin \mathcal{R}_\ell \cup \overline{\mathcal{R}}_\ell$, processor ℓ leaves J, V, and μ unchanged, i.e., for all $x \in X_\ell$,

$$J^{t+1}(x) = J^t(x), \quad V^{t+1}(x) = V^t(x), \quad \mu^{t+1}(x) = \mu^t(x).$$

Example 2.6.3 (Asynchronous Optimistic Policy Iteration for Discounted Finite-State MDP - Continued)

As an illustration of the preceding reduced space implementation, consider the special case of the finite-state discounted MDP of Example 2.6.2. Here

$$H(x,u,J) = \sum_{y=1}^{n} p_{xy}(u)\big(g(x,u,y) + \alpha J(y)\big),$$

and the mapping $F_\mu(V,Q)$ given by

$$F_\mu(V,Q)(x,u) = \sum_{y=1}^{n} p_{xy}(u)\Big(g(x,u,y) + \alpha \min\big\{V(y), Q(y,\mu(y))\big\}\Big),$$

defines the Q-factors of μ in a corresponding stopping problem. In the PI algorithm (2.74)-(2.75), policy evaluation of μ aims to solve this stopping problem, rather than solve a linear system of equations, as in classical PI. In particular, the policy evaluation iteration (2.75) is

$$J^{t+1}(x) = \sum_{y=1}^{n} p_{xy}\big(\mu^t(x)\big)\Big(g\big(x,\mu^t(x),y\big) + \alpha \min\big\{V^t(y), J^t(y)\big\}\Big),$$

for all $x \in X_\ell$. The policy improvement iteration (2.74) is a VI for the stopping problem:

$$J^{t+1}(x) = V^{t+1}(x) = \min_{u \in U(x)} \sum_{y=1}^{n} p_{xy}(u)\Big(g(x,u,y) + \alpha \min\big\{V^t(y), J^t(y)\big\}\Big),$$

for all $x \in X_\ell$, while the current policy is locally updated by

$$\mu^{t+1}(x) \in \arg\min_{u \in U(x)} \sum_{y=1}^{n} p_{xy}(u)\Big(g(x,u,y) + \alpha \min\big\{V^t(y), J^t(y)\big\}\Big),$$

for all $x \in X_\ell$. The "stopping cost" $V^t(y)$ is the most recent cost value, obtained by local policy improvement at y.

Example 2.6.4 (Asynchronous Optimistic Policy Iteration for Minimax Problems and Dynamic Games)

Consider the optimistic PI algorithm (2.74)-(2.75) for the case of the minimax problem of Example 1.2.5 of Chapter 1, where

$$H(x,u,J)=\sup_{w\in W(x,u)}\Big[g(x,u,w)+\alpha J\big(f(x,u,w)\big)\Big].$$

Then the local policy evaluation step [cf. Eq. (2.75)] is written as

$$J^{t+1}(x)=\sup_{w\in W(x,\mu^t(x))}\Big[g\big(x,\mu^t(x),w\big)+\alpha\min\Big\{V^t\big(f(x,\mu^t(x),w)\big),J^t\big(f(x,\mu^t(x),w)\big)\Big\}\Big].$$

The local policy improvement step [cf. Eq. (2.74)] takes the form

$$J^{t+1}(x)=V^{t+1}(x)=\min_{u\in U(x)}\sup_{w\in W(x,u)}\Big[g(x,u,w)+\alpha\min\Big\{V^t\big(f(x,u,w)\big),J^t\big(f(x,u,w)\big)\Big\}\Big],$$

and sets $\mu^{t+1}(x)$ to a u that attains the minimum.

Similarly for the discounted dynamic game problem of Example 1.2.4 of Chapter 1, a local policy evaluation step [cf. Eq. (2.75)] consists of a local VI for the maximizer's DP problem assuming a fixed policy for the minimizer, and a stopping cost V^t as per Eq. (2.75). A local policy improvement step [cf. Eq. (2.74)] at state x consists of the solution of a static game with a payoff matrix that also involves $\min\{V^t,J^t\}$ in place of J^t, as per Eq. (2.74).

A Variant with Interpolation

While the use of $\min\{V^t,J^t\}$ (rather than J^t) in Eq. (2.75) provides a convergence enforcement mechanism for the algorithm, it may also become a source of inefficiency, particularly when $V^t(x)$ approaches its limit $J^*(x)$ from lower values for many x. Then $J^{t+1}(x)$ is set to a lower value than the iterate

$$\hat{J}^{t+1}(x)=H\big(x,\mu^t(x),J^t\big),\tag{2.76}$$

given by the "standard" policy evaluation iteration, and in some cases this may slow down the algorithm.

A possible way to address this is to use an algorithmic variation that modifies appropriately Eq. (2.75), using interpolation with a parameter $\gamma_t\in(0,1]$, with $\gamma_t\to 0$. In particular, for $t\in\overline{\mathcal{R}}_\ell$ and $x\in X_\ell$, we calculate the values $J^{t+1}(x)$ and $\hat{J}^{t+1}(x)$ given by Eqs. (2.75) and (2.76), and if

$$J^{t+1}(x)<\hat{J}^{t+1}(x),\tag{2.77}$$

we reset $J^{t+1}(x)$ to

$$(1-\gamma_t)J^{t+1}(x)+\gamma_t\hat{J}^{t+1}(x). \tag{2.78}$$

The idea of the algorithm is to aim for a larger value of $J^{t+1}(x)$ when the condition (2.77) holds. Asymptotically, as $\gamma_t \to 0$, the iteration (2.77)-(2.78) becomes identical to the convergent update (2.75). For a more detailed analysis of an algorithm of this type, we refer to the paper by Bertsekas and Yu [BeY10].

2.7 NOTES, SOURCES, AND EXERCISES

Section 2.1: The abstract contractive DP model of this chapter was introduced by Denardo [Den67], who assumed an unweighted sup-norm, proved the basic results of Section 2.1, and described some of their applications. Section 2.1 has extended the analysis and results of [Den67] to the case of weighted sup-norm contractions.

Section 2.2: The abstraction of the computational methodology for finite-state discounted MDP within the broader framework of weighted sup-norm contractions and an infinite state space (Sections 2.2-2.6) follows the author's survey [Ber12b], and relies on several earlier analyses that use more specialized assumptions.

Section 2.3: The multistep error bound of Prop. 2.2.2 follows the work of Scherrer [Sch12], which explores the use of periodic policies in approximate VI and PI in finite-state discounted MDP (see also Scherrer and Lesner [ShL12], who give an example showing that the bound for approximate VI of Prop. 2.3.2 is essentially sharp for discounted finite-state MDP). For a related discussion of approximate VI, including the error amplification phenomenon of Example 2.3.1, and associated error bounds, see de Farias and Van Roy [DFV00].

Section 2.4: The error bound and approximate PI analysis of Section 2.4.1 (Prop. 2.4.3) extends the one of Bertsekas and Tsitsiklis [BeT96] (Section 6.2.2), which was given for finite-state discounted MDP, and was shown to be tight by an example.

Section 2.5: Optimistic PI has received a lot of attention in the literature, particularly for finite-state discounted MDP, and it is generally thought to be computationally more efficient in practice than ordinary PI (see e.g., Puterman [Put94], who refers to the method as "modified PI"). The convergence analysis of the synchronous optimistic PI (Section 2.5.1) follows Rothblum [Rot79], who considered the case of an unweighted sup-norm ($v = e$); see also Canbolat and Rothblum [CaR13], which extends some of the results of [Rot79]. The error bound for optimistic PI (Section 2.5.2) is due to Thierry and Scherrer [ThS10b], given for the case of a finite-state

discounted MDP. We follow closely their line of proof. Related error bounds and analysis are given by Scherrer [Sch11].

The λ-PI method [cf. Eq. (2.34)] was introduced by Bertsekas and Ioffe [BeI96], and was also presented in the book [BeT96], Section 2.3.1. It was studied further in approximate DP contexts by Thierry and Scherrer [ThS10a], Bertsekas [Ber11b], and Scherrer [Sch11]. An extension of λ-PI, called Λ-PI, uses a different parameter λ_i for each state i, and is discussed in Section 5 of the paper by Yu and Bertsekas [YuB12]. Based on the discussion of Section 1.2.5 and Exercise 1.2, Λ-PI may be viewed as a diagonally scaled version of the proximal algorithm, i.e., one that uses a different penalty parameter for each proximal term.

When the state and control spaces are finite, and cost approximation over a subspace $\{\Phi r \mid r \in \Re^s\}$ is used (cf. Section 1.2.4), a prominent approximate PI approach is to replace the exact policy evaluation equation $J_{\mu^k} = T_{\mu^k} J_{\mu^k}$ with an approximate version of the form

$$\Phi r_k = W T_{\mu^k}(\Phi r_k), \tag{2.79}$$

where W is some $n \times n$ matrix, where n is the number of states. For example the projected and aggregation equations, described in Section 1.2.4, have this form. The next policy μ^{k+1} is obtained using the policy improvement equation

$$T_{\mu^{k+1}}(\Phi r_k) = T(\Phi r_k). \tag{2.80}$$

Critical issues for the validity of such a method is whether the approximate Bellman equations

$$\Phi r = W T(\Phi r), \qquad \Phi r = W T_\mu(\Phi r), \quad \mu \in \mathcal{M},$$

have a unique solution. This is true if the composite mappings $W \circ T \circ \Phi$ and $W \circ T_\mu \circ \Phi$ are contractions over $\Re^n$. In particular, in the case of an aggregation equation, where $W = \Phi D$, the rows of Φ and D are probability distributions, and T_μ, $\mu \in \mathcal{M}$, are monotone sup-norm contractions, the mappings $W \circ T \circ \Phi$ and $W \circ T_\mu \circ \Phi$ are also monotone sup-norm contractions. However, in other cases, including when policy evaluation is done using the projected equation, $W \circ T \circ \Phi$ need not be monotone or be a contraction of any kind, and the approximate PI algorithm (2.79)-(2.80) may lead to systematic oscillations, involving cycles of policies (see related discussions in [BeT96], [Ber11c], and [Ber12a]). This phenomenon has been known since the early days of approximate DP ([Ber96] and the book [BeT96]), but its practical implications have not been fully assessed. Generally, the line of analysis of Section 2.5.3, which does not require monotonicity or sup-norm contraction properties of the composite mappings $W \circ T \circ \Phi$ and $W \circ T_\mu \circ \Phi$, can be applied to the approximate PI algorithm (2.79)-(2.80), but only in the case where these mappings are contractions over $\Re^n$ with respect to a common norm $\|\cdot\|$; see Exercise 2.6 for further discussion.

Section 2.6: Asynchronous VI (Section 2.6.1) for finite-state discounted MDP and games, shortest path problems, and abstract DP models, was proposed in the author's paper on distributed DP [Ber82]. The asynchronous convergence theorem (Prop. 2.6.1) was first given in the author's paper [Ber83], where it was applied to a variety of algorithms, including VI for discounted and undiscounted DP, and gradient methods for unconstrained optimization (see also Bertsekas and Tsitsiklis [BeT89], where a textbook account is presented). Earlier references on distributed asynchronous iterative algorithms include the work of Chazan and Miranker [ChM69] on Gauss-Seidel methods for solving linear systems of equations (who attributed the original idea to Rosenfeld [Ros67]), and also Baudet [Bau78] on sup-norm contractive iterations. We refer to [BeT89] for detailed references.

Asynchronous algorithms have also been studied and applied to simulation-based DP, particularly in the context of Q-learning, first proposed by Watkins [Wat89], which may be viewed as a stochastic version of VI, and is a central algorithmic concept in approximate DP and reinforcement learning. Two principal approaches for the convergence analysis of asynchronous stochastic algorithms have been suggested. The first approach, initiated in the paper by Tsitsiklis [Tsi94], considers the totally asynchronous computation of fixed points of abstract sup-norm contractive mappings and monotone mappings, which are defined in terms of an expected value. The algorithm of [Tsi94] contains as special cases Q-learning algorithms for finite-spaces discounted MDP and SSP problems. The analysis of [Tsi94] shares some ideas with the theory of Section 2.6.1, and also relies on the theory of stochastic approximation methods. For a subsequent analysis of the convergence of Q-learning for SSP, which addresses the issue of boundedness of the iterates, we refer to Yu and Bertsekas [YuB13b]. The second approach, treats asynchronous algorithms of the stochastic approximation type under some restrictions on the size of the communication delays or on the time between consecutive updates of a typical component. This approach was initiated in the paper by Tsitsiklis, Bertsekas, and Athans [TBA86], and was also developed in the book by Bertsekas and Tsitsiklis [BeT89] for stochastic gradient optimization methods. A related analysis that uses the ODE approach for more general fixed point problems was given in the paper by Borkar [Bor98], and was refined in the papers by Abounadi, Bertsekas, and Borkar [ABB02], and Borkar and Meyn [BoM00], which also considered applications to Q-learning. We refer to the monograph by Borkar [Bor08] for a more comprehensive discussion.

The convergence of asynchronous PI for finite-state discounted MDP under the condition $J^0 \geq T_{\mu^0} J^0$ was shown by Williams and Baird [WiB93], who also gave examples showing that without this condition, cycling of the algorithm may occur. The asynchronous PI algorithm with a uniform fixed point (Section 2.6.3) was introduced in the papers by Bertsekas and Yu [BeY10], [BeY12], [YuB13a], in order to address this difficulty. Our

analysis follows the analysis of these papers.

In addition to resolving the asynchronous convergence issue, the asynchronous PI algorithm of Section 2.6.3, obviates the need for minimization over all controls at every iteration (this is the generic computational efficiency advantage that optimistic PI typically holds over VI). Moreover, the algorithm admits a number of variations thanks to the fact that Prop. 2.6.4 asserts the contraction property of the mapping G_μ *for all* μ. This can be used to prove convergence in variants of the algorithm where the policy μ^t is updated more or less arbitrarily, with the aim to promote some objective. We refer to the paper [BeY12], which also derives related asynchronous simulation-based Q-learning algorithms with and without cost function approximation, where μ^t is replaced by a randomized policy to enhance exploration.

The randomized asynchronous optimistic PI algorithm of Section 2.6.2, introduced in the first edition of this monograph, also resolves the asynchronous convergence issue. The fact that this algorithm does not require the monotonicity assumption may be useful in nonDP algorithmic contexts (see [Ber16b] and Exercise 2.6).

In addition to discounted stochastic optimal control, the results of this chapter find application in the context of the stochastic shortest path problem of Example 1.2.6, when all policies are proper. Then, under some additional assumptions, it can be shown that T and T_μ are weighted sup-norm contractions with respect to a special norm. It follows that the analysis and algorithms of this chapter apply in this case. For a detailed discussion, we refer to the monograph [BeT96] and the survey [Ber12b]. For extensions to the case of countable state space, see the textbook [Ber12a], Section 3.6, and Hinderer and Waldmann [HiW05].

EXERCISES

2.1 (Periodic Policies)

Consider the multistep mappings $\overline{T}_\nu = T_{\mu_0} \cdots T_{\mu_{m-1}}$, $\nu \in \mathcal{M}_m$, defined in Exercise 1.1 of Chapter 1, where $\mathcal{M}_m$ is the set of m-tuples $\nu = (\mu_0, \ldots, \mu_{m-1})$, with $\mu_k \in \mathcal{M}$, $k = 1, \ldots, m-1$, and m is a positive integer. Assume that the mappings T_μ satisfy the monotonicity and contraction Assumptions 2.1.1 and 2.1.2, so that the same is true for the mappings $\overline{T}_\nu$ (with the contraction modulus of $\overline{T}_\nu$ being α^m, cf. Exercise 1.1).

(a) Show that the unique fixed point of $\overline{T}_\nu$ is J_π, where π is the nonstationary but periodic policy $\pi = \{\mu_0, \ldots, \mu_{m-1}, \mu_0, \ldots, \mu_{m-1}, \ldots\}$.

(b) Show that the multistep mappings $T_{\mu_0} \cdots T_{\mu_{m-1}}$, $T_{\mu_1} \cdots T_{\mu_{m-1}} T_{\mu_0}$, $\ldots$, $T_{\mu_{m-1}} T_{\mu_0} \cdots T_{\mu_{m-2}}$, have unique corresponding fixed points $J_0, J_1, \ldots$,

J_{m-1}, which satisfy

$$J_0 = T_{\mu_0} J_1, \;\; J_1 = T_{\mu_1} J_2, \; \ldots \; , \; J_{\mu_{m-2}} = T_{\mu_{m-2}} J_{\mu_{m-1}}, \;\; J_{\mu_{m-1}} = T_{\mu_{m-1}} J_0.$$

Hint: Apply T_{μ_0} to the fixed point relation

$$J_1 = T_{\mu_1} \cdots T_{\mu_{m-1}} T_{\mu_0} J_1$$

to show that $T_{\mu_0} J_1$ is the fixed point of $T_{\mu_0} \cdots T_{\mu_{m-1}}$, i.e., is equal to J_0. Similarly, apply T_{μ_1} to the fixed point relation

$$J_2 = T_{\mu_2} \cdots T_{\mu_{m-1}} T_{\mu_0} T_{\mu_1} J_2,$$

to show that $T_{\mu_1} J_2$ is the fixed point of $T_{\mu_1} \cdots T_{\mu_{m-1}} T_{\mu_0}$, etc.

Solution: (a) Let us define

$$J_0 = \lim_{k \to \infty} \overline{T}_\nu^k J', \quad J_1 = \lim_{k \to \infty} \overline{T}_\nu^k (T_{\mu_0} J'), \; \ldots, \; J_{m-1} = \lim_{k \to \infty} \overline{T}_\nu^k (T_{\mu_0} \cdots T_{\mu_{m-2}} J'),$$

where J' is some function in $\mathcal{B}(X)$. Since $\overline{T}_\nu$ is a contraction mapping, $J_0, \ldots, J_{m-1}$ are all equal to the unique fixed point of $\overline{T}_\nu$. Since $J_0, \ldots, J_{m-1}$ are all equal, they are also equal to J_π (by the definition of J_π). Thus J_π is the unique fixed point of $\overline{T}_\nu$.

(b) Follow the hint.

2.2 (Asynchronous Convergence Theorem for Time-Varying Maps)

In reference to the framework of Section 2.6.1, let $\{T_t\}$ be a sequence of mappings from $\mathcal{R}(X)$ to $\mathcal{R}(X)$ that have a common unique fixed point J^*, let Assumption 2.6.1 hold, and assume that there is a sequence of nonempty subsets $\big\{S(k)\big\} \subset \mathcal{R}(X)$ with $S(k+1) \subset S(k)$ for all k, and with the following properties:

(1) *Synchronous Convergence Condition:* Every sequence $\{J^k\}$ with $J^k \in S(k)$ for each k, converges pointwise to J^*. Moreover, we have

$$T_t J \in S(k+1), \qquad \forall \; J \in S(k), \; k, \, t = 0, 1, \ldots .$$

(2) *Box Condition:* For all k, $S(k)$ is a Cartesian product of the form

$$S(k) = S_1(k) \times \cdots \times S_m(k),$$

where $S_\ell(k)$ is a set of real-valued functions on X_ℓ, $\ell = 1, \ldots, m$.

Then for every $J^0 \in S(0)$, the sequence $\{J^t\}$ generated by the asynchronous algorithm

$$J_\ell^{t+1}(x) = \begin{cases} T_t(J_1^{\tau_{\ell 1}(t)}, \ldots, J_m^{\tau_{\ell m}(t)})(x) & \text{if } t \in \mathcal{R}_\ell, \; x \in X_\ell, \\ J_\ell^t(x) & \text{if } t \notin \mathcal{R}_\ell, \; x \in X_\ell, \end{cases}$$

[cf. Eq. (2.61)] converges pointwise to J^*.

Solution: A straightforward adaptation of the proof of Prop. 2.6.1.

2.3 (Nonmonotonic Contractive Models – Fixed Points of Concave Sup-Norm Contractions [Ber16b])

The purpose of this exercise is to make a connection between our abstract DP model and the problem of finding the fixed point of a (not necessarily monotone) mapping that is a sup-norm contraction and has concave components. Let $T : \Re^n \mapsto \Re^n$ be a real-valued function whose n scalar components are concave. Then the components of T can be represented as

$$(TJ)(x) = \inf_{u \in U(x)} \Big\{ F(x,u) - J'u \Big\}, \qquad x = 1, \ldots, n, \tag{2.81}$$

where $u \in \Re^n$, $J'u$ denotes the inner product of J and u, $F(x,\cdot)$ is the conjugate convex function of the convex function $-(TJ)(x)$, and $U(x) = \Big\{ u \in \Re^n \mid F(x,u) < \infty \Big\}$ is the effective domain of $F(x,\cdot)$ (for the definition of these terms, we refer to books on convex analysis, such as [Roc70] and [Ber09]). Assuming that the infimum in Eq. (2.81) is attained for all x, show how the VI algorithm of Section 2.6.1 and the PI algorithm of Section 2.6.3 can be used to find the fixed point of T in the case where T is a sup-norm contraction, but not necessarily monotone. *Note*: For algorithms that relate to the context of this exercise and are inspired by approximate PI, see [Ber16b].

Solution: The analysis of Sections 2.6.1 and 2.6.3 does not require monotonicity of the mapping T_μ given by

$$(T_\mu J)(x) = F\big(x, \mu(x)\big) - J'\mu(x).$$

2.4 (Discounted Problems with Unbounded Cost per Stage)

Consider a countable-state MDP, where $X = \{1, 2, \ldots\}$, the discount factor is $\alpha \in (0,1)$, the transition probabilities are denoted $p_{xy}(u)$ for $x, y \in X$ and $u \in U(x)$, and the expected cost per stage is denoted by $g(x,u)$, $x \in X$, $u \in U(x)$. The constraint set $U(x)$ may be infinite. For a positive weight sequence $v = \big\{v(1), v(2), \ldots\big\}$, we consider the space $\mathcal{B}(X)$ of sequences $J = \big\{J(1), J(2), \ldots\big\}$ such that $\|J\| < \infty$, where $\|\cdot\|$ is the corresponding weighted sup-norm. We assume the following.

(1) The sequence $G = \{G_1, G_2, \ldots\}$, where

$$G_x = \sup_{u \in U(x)} \big|g(x,u)\big|, \qquad x \in X,$$

belongs to $\mathcal{B}(X)$.

(2) The sequence $V = \{V_1, V_2, \ldots\}$, where

$$V_x = \sup_{u \in U(x)} \sum_{y \in X} p_{xy}(u)\, v(y), \qquad x \in X,$$

belongs to $\mathcal{B}(X)$.

(3) We have

$$\frac{\sum_{y\in X} p_{xy}(u)v(y)}{v(x)} \leq 1, \qquad \forall\ x \in X.$$

Consider the monotone mappings T_μ and T, given by

$$(T_\mu J)(x) = g\big(x,\mu(x)\big) + \alpha \sum_{y\in X} p_{xy}\big(\mu(x)\big)J(y), \qquad x \in X,$$

$$(TJ)(x) = \inf_{u\in U(x)} \left[g(x,u) + \alpha \sum_{y\in X} p_{xy}(u)J(y) \right], \qquad x \in X.$$

Show that T_μ and T map $\mathcal{B}(X)$ into $\mathcal{B}(X)$, and are contraction mappings with modulus α.

Solution: We have

$$\frac{\big|(T_\mu J)(x)\big|}{v(x)} \leq \frac{G_x}{v(x)} + \alpha \sum_{y\in X} \frac{p_{xy}\big(\mu(x)\big)\, v(y)}{v(x)} \frac{\big|J(y)\big|}{v(y)}, \qquad \forall\ x \in X,\ \mu \in \mathcal{M},$$

from which, using assumptions (1) and (2),

$$\frac{\big|(T_\mu J)(x)\big|}{v(x)} \leq \|G\| + \|V\|\,\|J\|, \qquad \forall\ x \in X,\ \mu \in \mathcal{M}.$$

A similar argument shows that

$$\frac{\big|(TJ)(x)\big|}{v(x)} \leq \|G\| + \|V\|\,\|J\|, \qquad \forall\ x \in X.$$

It follows that $T_\mu J \in \mathcal{B}(X)$ and $TJ \in \mathcal{B}(X)$ if $J \in \mathcal{B}(X)$.

For any $J, J' \in \mathcal{B}(X)$ and $\mu \in \mathcal{M}$, we have

$$\begin{aligned}
\|T_\mu J - T_\mu J'\| &= \sup_{x\in X} \frac{\Big|\alpha \sum_{y\in X} p_{xy}\big(\mu(x)\big)\big(J(x) - J'(x)\big)\Big|}{v(x)} \\
&\leq \sup_{x\in X} \frac{\Big|\alpha \sum_{y\in X} p_{xy}\big(\mu(x)\big)v(y)\big(|J(y) - J'(y)|/v(y)\big)\Big|}{v(x)} \\
&\leq \sup_{x\in X} \alpha \frac{\Big|\sum_{y\in X} p_{xy}\big(\mu(x)\big)v(y)\Big|}{v(x)} \|J - J'\| \\
&\leq \alpha \|J - J'\|,
\end{aligned}$$

where the last inequality follows from assumption (3). Hence T_μ is a contraction of modulus α.

To show that T is a contraction, we note that

$$\frac{(T_\mu J)(x)}{v(x)} \leq \frac{(T_\mu J')(x)}{v(x)} + \alpha \|J - J'\|, \qquad x \in X,\ \mu \in \mathcal{M},$$

so by taking infimum over $\mu \in \mathcal{M}$, we obtain

$$\frac{(TJ)(x)}{v(x)} \leq \frac{(TJ')(x)}{v(x)} + \alpha \|J - J'\|, \qquad x \in X.$$

Similarly,

$$\frac{(TJ')(x)}{v(x)} \leq \frac{(TJ)(x)}{v(x)} + \alpha \|J - J'\|, \qquad x \in X,$$

and by combining the last two relations the contraction property of T follows.

2.5 (Solution by Mathematical Programming)

Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold. Show that if $J \leq TJ$ and $J \in \mathcal{B}(X)$, then $J \leq J^*$. Use this fact to show that if $X = \{1, \ldots, n\}$ and $U(i)$ is finite for each $i = 1, \ldots, n$, then $J^*(1), \ldots, J^*(n)$ solves the following problem (in $z_1, \ldots, z_n$):

$$\begin{aligned} &\text{maximize} \quad \sum_{i=1}^{n} z_i \\ &\text{subject to} \quad z_i \leq H(i, u, z), \qquad i = 1, \ldots, n, \quad u \in U(i), \end{aligned}$$

where $z = (z_1, \ldots, z_n)$. *Note*: This is a linear or nonlinear program (depending on whether H is linear in J or not) with n variables and as many as $n \times m$ constraints, where m is the maximum number of elements in the sets $U(i)$.

Solution: If $J \leq TJ$, by monotonicity we have $J \leq \lim_{k\to\infty} T^k J = J^*$. Any feasible solution z of the given optimization problem satisfies $z_i \leq H(i, u, z)$ for all $i = 1, \ldots, n$ and $u \in U(i)$, so that $z \leq Tz$. It follows that $z \leq J^*$, which implies that J^* solves the optimization problem.

2.6 (Conditions for Convergence of PI with Cost Function Approximation [Ber11c])

Let the monotonicity and contraction Assumptions 2.1.1 and 2.1.2 hold, and assume that there are n states, and that $U(x)$ is finite for every x. Consider a PI method that aims to approximate a fixed point of T on a subspace $S = \{\Phi r \mid r \in \Re^s\}$, where Φ is an $n \times s$ matrix, and evaluates a policy $\mu \in \mathcal{M}$ with a solution $\tilde{J}_\mu$ of the following fixed point equation in the vector $J \in \Re^n$:

$$J = W T_\mu J \tag{2.82}$$

where $W : \Re^n \mapsto \Re^n$ is some mapping (possibly nonlinear, but independent of μ), whose range is contained in S. Examples where W is linear include policy

evalution using the projected and aggregation equations; see Section 1.2.4. The algorithm is given by

$$\Phi r_k = WT_{\mu^k}(\Phi r_k), \qquad T_{\mu^{k+1}}(\Phi r_k) = T(\Phi r_k); \tag{2.83}$$

[cf. Eqs. (2.79)-(2.80)]. We assume the following:

(1) For each $J \in \Re^n$, there exists $\mu \in \mathcal{M}$ such that $T_\mu J = TJ$.

(2) For each $\mu \in \mathcal{M}$, Eq. (2.82) has a unique solution that belongs to S and is denoted $\tilde{J}_\mu$. Moreover, for all J such that $WT_\mu J \leq J$, we have

$$\tilde{J}_\mu = \lim_{k\to\infty} (WT_\mu)^k J.$$

(3) For each $\mu \in \mathcal{M}$, the mappings W and WT_μ are monotone in the sense that

$$WJ \leq WJ', \quad WT_\mu J \leq WT_\mu J', \qquad \forall\ J, J' \in \Re^n \text{ with } J \leq J'. \tag{2.84}$$

Note that conditions (1) and (2) guarantee that the iterations (2.83) are well-defined. Assume that the method is initiated with some policy in $\mathcal{M}$, and it is operated so that it terminates when a policy $\overline{\mu}$ is obtained such that $T_{\overline{\mu}}\tilde{J}_{\overline{\mu}} = T\tilde{J}_{\overline{\mu}}$. Show that the method terminates in a finite number of iterations, and the vector $\tilde{J}_{\overline{\mu}}$ obtained upon termination is a fixed point of WT. *Note*: Condition (2) is satisfied if WT_μ is a contraction, while condition (b) is satisfied if W is a matrix with nonnegative components and T_μ is monotone for all μ. For counterexamples to convergence when the conditions (2) and/or (3) are not satisfied, see [BeT96], Section 6.4.2, and [Ber12a], Section 2.4.3.

Solution: Similar to the standard proof of convergence of (exact) PI, we use the policy improvement equation $T_{\overline{\mu}}\tilde{J}_\mu = T\tilde{J}_\mu$, the monotonicity of W, and the policy evaluation equation to write

$$WT_{\overline{\mu}}\tilde{J}_\mu = WT\tilde{J}_\mu \leq WT_\mu\tilde{J}_\mu = \tilde{J}_\mu.$$

By iterating with the monotone mapping $WT_{\overline{\mu}}$ and by using condition (2), we obtain

$$\tilde{J}_{\overline{\mu}} = \lim_{k\to\infty} (WT_{\overline{\mu}})^k \tilde{J}_\mu \leq \tilde{J}_\mu.$$

There are finitely many policies, so we must have $\tilde{J}_{\overline{\mu}} = \tilde{J}_\mu$ after a finite number of iterations, which using the policy improvement equation $T_{\overline{\mu}}\tilde{J}_\mu = T\tilde{J}_\mu$, implies that $T_{\overline{\mu}}\tilde{J}_{\overline{\mu}} = T\tilde{J}_{\overline{\mu}}$. Thus the algorithm terminates with $\overline{\mu}$, and since $\tilde{J}_{\overline{\mu}} = WT_{\overline{\mu}}\tilde{J}_{\overline{\mu}}$, it follows that $\tilde{J}_{\overline{\mu}}$ is a fixed point of WT.