Introduction to Markov Decision Processes

Martin L. Puterman and Timothy C. Y. Chan

June 19, 2023

Chapter 1

Infinite Horizon Expected Total Reward Models

"Starting a long way off the true point, and proceeding by loops and zigzags, we now and then arrive just where we ought to be[1]*."*
George Eliot - British novelist, 1819-1880

(I am thinking of moving positive and negative models stuff to appendix since most (all?) interesting finite state and action models are transient or ssp.)

1.1 Introduction

In the spirit of the above quote from *Middlemarch*, this chapter focuses on models with a zero-reward destination state that we seek to reach, or on the other hand avoid as long as possible. Such models use the expected total reward criteria which raises a range of technical issues, most notably, the infinite horizon expected total reward

$$v^{\pi}(s) = \lim_{N\to\infty} E^{\pi}\left\{ \sum_{n=1}^{N} r(X_n, Y_n) \,|\, X_1 = s \right\} \tag{1.1}$$

[1]From the novel *Middlemarch*.

may be unbounded or even not exist for some policies. While most of the challenges arise in countable state[2] models, finite state and action models still require some subtle analyses.

Originally, expected total reward models were carefully formulated and analyzed by mathematicians for their intrinsic complexity. But they also provided a unified framework to study colorful and significant applications including

- optimal stopping,
- sequential statistical hypothesis testing,
- finding the shortest path in a network with random transitions, and
- gambling strategy.

The recent use of Markov decision process and reinforcement learning methods in *task-oriented* applications such as gaming, autonomous vehicle guidance and robotic control, has rekindled interest in such models and necessitates this chapter.

1.1.1 Motivating examples

To see some of the challenges we face when using the expected total reward criterion we discuss three examples.

First consider the 2-state model in Section **??**. In that model there are two stationary policies (those choosing action $a_{2,2}$ in s_2) with expected total reward $+\infty$ and two stationary policies (those choosing action $a_{2,1}$ in s_2) with expected total reward reward $-\infty$. Hence the expected total reward criterion would not be useful for distinguishing policies when we maximizing rewards or minimizing costs.

The grid world navigation model of Section **??** provides an interesting application in which to explore the use of the expected total reward criterion. This is an example of what is often referred to as an *episodic* model. That is the decision process terminates at the end of an episode where episode length varies among policies. As another example, in the game of golf, an elite player may usually require 4 shots on a hole but on some occasions may require many more.

Example 1.1. Consider the grid world navigation model of Section **??**. We focus on the task of delivering coffee from the coffee room (cell 13) to the professor's office (cell 1). In this model, the decision process terminates when the robot successfully delivers coffee or falls down the stairs (cell 7). The goal in this application is to choose a series of actions for the robot so as to maximize the expected total reward consisting of a positive reward for delivering the coffee, a negative reward (cost) incurred if the robot falls down the stairs and a negative reward (cost) per transition.

[2]See Chapter 7 in Puterman [1994].

In the presence of randomness in the robot's motion in all cells, under any policy, the decision process terminates at a random time so that a finite horizon model with a fixed horizon would not apply. Discounting makes little sense in this application since the time scale is so short and averaging is inappropriate because the process terminates when the task is completed so the average reward of any policy would be zero.

On the other hand, when the model is deterministic there exist polices which never terminate so that their expected total reward equals $-\infty$. However even in this case, there are also polices with finite total reward.

The following example illustrates further pitfalls that arise when using the expected total reward criterion.

Example 1.2. Let $S = \{s_1, s_2\}$, $A_{s_i} = \{a_{i,1}, a_{i,2}\}$ for $i = 1, 2$, $r(s_1, a_{1,1}) = -1$, $r(s_1, a_{1,2}) = 1$, $r(s_2, a_{2,1}) = 0$, $r(s_2, a_{2,2}) = -1$ and $p(s_1|s_1, a_{1,1}) = p(s_2|s_1, a_{1,2}) = p(s_2|s_2, a_{2,1}) = p(s_1|s_2, a_{2,2}) = 1$. (See Figure 1.1).

In this model there are four stationary policies; $d_i^\infty, i = 1, 2, 3, 4$ where $d_1 = (a_{1,1}, a_{2,1}), d_2 = (a_{1,1}, a_{2,2}), d_3 = (a_{1,2}, a_{2,1})$ and $d_4 = (a_{1,2}, a_{2,2})$. For these we have

$$\mathbf{v}^{d_1^\infty} = \begin{bmatrix} -\infty \\ 0 \end{bmatrix}, \quad \mathbf{v}^{d_2^\infty} = \begin{bmatrix} -\infty \\ -\infty \end{bmatrix}, \quad \mathbf{v}^{d_3^\infty} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and for d_4^∞ when $X_1 = s_1$,

$$\liminf_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = 0 \quad \text{and} \quad \limsup_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = 1$$

and when $X_1 = s_2$,

$$\liminf_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = -1 \quad \text{and} \quad \limsup_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = 0.$$

If we seek to maximize expected total reward we would prefer d_3^∞. However this example shows that there are stationary policies for which the expected total reward is $-\infty$ and stationary policies for which the limit implicit in the definition of the expected total reward does not exist.

Suppose now we subtract -1 from all rewards so that $r(s, a) \leq 0$ for all states and actions. Then in the resulting model, all policies have expected total reward $-\infty$. Thus something is clearly special about the existence of policies with absorbing states with zero reward.

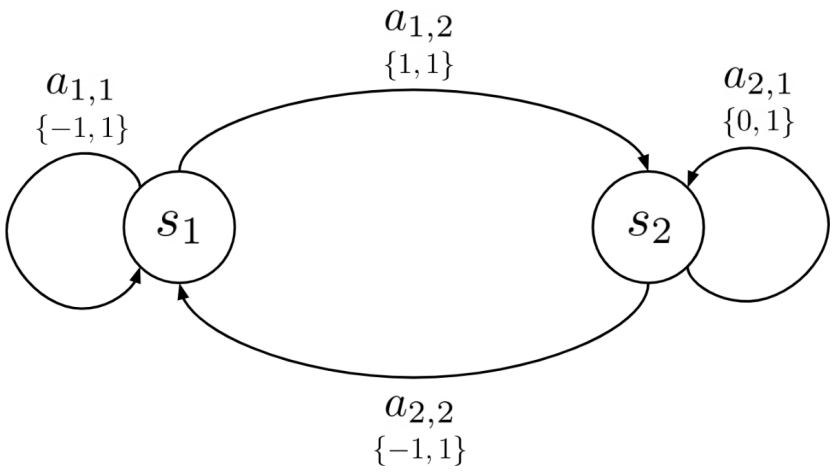

Figure 1.1: Symbolic representation of model in Example 1.2.

1.2 Model Classification

We classify modes in two distinct ways:

1. On the basis of the chain structure of Markov chains generated by stationary policies, and

2. on the basis of the sign of the rewards.

Historically, the first research on expected total reward models focused on models with either positive or negative rewards. It did so to ensure that the expected total reward of stationary policies was well defined. Most of this research was concerned with the existence of optimal policies in countable state models. A distinct research stream considered finite state models containing zero-reward absorbing states. Such models are more relevant for modern applications and moreover possess some nice theoretical properties.

1.2.1 Some Assumptions

We assume throughout this chapter stationary rewards and transition probabilities and that the rewards do not depend on the subsequent state[3].

For each $s \in S$ and $a \in A_s$, define

$$r_+(s,a) := \max\{r(s,a), 0\} \quad \text{and} \quad r_-(s,a) := \min\{r(s,a), 0\}. \tag{1.2}$$

[3]If they do, replace $r(s,a,j)$ by $r(s,a) = \sum_{j\in S} r(s,a,j)p(j|s,a)$.

and for each $\pi \in \Pi^{HR}$ define[4]

$$v_+^\pi(s) := E^\pi\left\{ \sum_{n=1}^{\infty} r_+(X_n, Y_n) \middle| X_1 = s \right\} \tag{1.3}$$

and

$$v_-^\pi(s) := E^\pi\left\{ \sum_{n=1}^{\infty} r_-(X_n, Y_n) \middle| X_1 = s \right\}. \tag{1.4}$$

Since

$$v^\pi(s) = v_+^\pi(s) + v_-^\pi(s), \tag{1.5}$$

the limit defining $v^\pi(s)$ exists whenever at **least one** of $v_+^\pi(s)$ and $v_-^\pi(s)$ is finite.

Returning to Example 1.2, we observe that

$$v_+^{d_4^\infty}(s) = +\infty \quad \text{and} \quad v_-^{d_4^\infty}(s) = -\infty$$

so that

$$v^{d_4^\infty}(s) = \lim_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n))$$

does not exist.

1.2.2 Transient Models

We first distinguish a class of models which we refer to as *transient*[5] Transient models are characterized by the structure of Markov chains corresponding to stationary policies (See Appendix **??** for a review of relevant Markov chain concepts and results).

A *transient model* satisfies the property that there exists an absorbing state $\Delta \in S$ and a single action $\delta \in A_\Delta$ for which:

1. $r(\Delta, \delta) = 0$ and $p(\Delta|\Delta, \delta) = 1$ and
2. for every stationary policy d^∞,

$$\lim_{N\to\infty} P^{d^\infty}(X_N = \Delta | X_1 = s) = 1 \tag{1.6}$$

for all $s \in S$.

The only assumptions on rewards in a transient model are that $r(\Delta, \delta) = 0$. consequently a transient model can have both positive and negative rewards. The condition that there is a single state Δ satisfying conditions 1. and 2. can be relaxed as follows:

[4]The limits implicit in defining $v_+^\pi(s)$ and $v_-^\pi(s)$ always exist but may be infinite.

[5]The expression transient model originates with A.F. Veinott [1969] however he defines this concept in terms of transition probability matrices of stationary policies. Derman [1970] referred to such models as optimal first passage models. Other refer to such models as *contracting*

1. Δ may represent a *set* of absorbing states with the property that for each $s' \in \Delta$, $A_{s'}$ contains a single action a' for which $r(s', a') = 0$ and $p(s'|s', a') = 1$.

2. When the sets of absorbing states varies between stationary policies, i.e, suppose for $d \in D^{MD}$, the set of its absorbing states is denoted Δ_d. In this case, we can add a zero-reward absorbing state Δ that can be reached in one zero-reward transition from each $s \in \Delta_d$.

The most important consequence of this definition is that under every stationary policy the corresponding Markov chain is *unichain* with a single absorbing state Δ and a set of transient states $S - \Delta$. This means that the transition probability and reward corresponding to decision rule d can be written as

$$\mathbf{P}_d = \begin{bmatrix} \mathbf{Q}_d & \mathbf{R}_d \\ \mathbf{0} & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{r}_d = \begin{bmatrix} (\mathbf{r}_d)_T \\ 0 \end{bmatrix} \tag{1.7}$$

where $\mathbf{Q}_d$ is an $|S-1| \times |S-1|$ matrix which represents the probability of transitions between transient states in $S - \Delta$, $\mathbf{R}_d$ represents the probability of transitions from $S - \Delta$ to Δ. The last row of the matrix corresponds to transition from state Δ. Since it is absorbing it cannot reach states in $S - \Delta$, instead in remains in Δ in perpetuity which corresponds to the entry 1 in the lower right-hand corner of the matrix. The rewards can be partitioned with $(\mathbf{r}_d)_T$ representing $\mathbf{r}_d$ restricted to states in $S - \Delta$ and a reward of 0 in state Δ.

Since in a transient model, the system reaches state Δ with certainty under each stationary policy, at least one of the row sums of $\mathbf{Q}_d$ must be strictly less than 1[6]. Consequently $\mathbf{Q}_d^n \to \mathbf{0}$. It follows from Lemma **??** that $(\mathbf{I} - \mathbf{Q}_d)^{-1}$ exists. Hence

$$\mathbf{v}^{d^\infty} = \sum_{n=1}^{\infty} \mathbf{P}_d^{n-1} \mathbf{r}_d = \sum_{n=1}^{\infty} \mathbf{Q}_d^{n-1} (\mathbf{r}_d)_T = (\mathbf{I} - \mathbf{Q}_d)^{-1} (\mathbf{r}_d)_T. \tag{1.8}$$

Moreover, the *spectral radius*[7] of $\mathbf{Q}_d$, denoted by $\rho(\mathbf{Q}_d)$ satisfies[8]

$$\rho(\mathbf{Q}_d) < 1. \tag{1.9}$$

As a further consequence of condition in 2. in the definition of a transient model, if τ_Δ denotes the random time the system reaches state Δ, then (1.6) implies that

$$E^{d^\infty}\{\tau_\Delta | X_1 = s\} < \infty. \tag{1.10}$$

Since

$$v^{d^\infty}(s) = E^{d^\infty}\left\{ \sum_{n=1}^{\tau_\Delta} r(X_n, d(X_n)) \middle| X_1 = s \right\}$$

[6]Otherwise $S - \Delta$ would be a closed class and not transient.

[7]The spectral radius of a finite square matrix is its largest eigenvalue in absolute value.

[8]See A.F. Veinott [1969] or p.223 in Berman and Plemmons [1979]

it follows that

$$-\infty < E^{d^\infty}\{\tau_\Delta | X_1 = s\} r_{\min} \leq v^{d^\infty}(s) \leq E^{d^\infty}\{\tau_\Delta | X_1 = s\} r_{\max} < \infty.$$

where

$$r_{\min} = \min_{a \in A_s, s \in S} r(s,a) \quad \text{and} \quad r_{\max} = \max_{a \in A_s, s \in S} r(s,a).$$

The finiteness of r_{min} and r_{max} follow from the assumption of a finite state and action model. Thus in a transient model, the expected total reward of every stationary policy exists and is finite.

As noted in Section **??**, a discounted model may be regarded as transient model by interpreting discounting as random termination at a geometrically distributed stopping time independent of the policy. Conversely we will see that transient models inherit many of the properties of discounted reward models.

1.2.3 Stochastic shortest path models

Stochastic shortest path[9] (SSP) models generalize transient models by allowing the possibility that some stationary policies generate Markov chains with more than one closed class. However they exclude the possibility that such a policy can be optimal by requiring that the value of any such policy be equal to $-\infty$ in some state.

A *Stochastic shortest path model* is characterized by a distinguished state Δ and a single action $\delta \in A_\Delta$ for which:

1. $r(\Delta, \delta) = 0$ and $p(\Delta|\Delta, \delta) = 1$,
2. the existence of at least one stationary policy d^∞ for which

$$\lim_{N \to \infty} P^{d^\infty}(X_N = \Delta | X_1 = s) = 1$$

   for all $s \in S$ and
3. Suppose

$$\lim_{N \to \infty} P^{d^\infty}(X_N = \Delta | X_1 = s) < 1 \tag{1.11}$$

   for some stationary policy d' and $s \in S$ then $v^{(d')^\infty}(s) = -\infty$.

We say[10] that a stationary policy is *proper* if it satisfies

$$\lim_{N \to \infty} P^{d^\infty}(X_N = \Delta | X_1 = s) = 1.$$

[9] The nomenclature stochastic shortest path model and the associated terminology originates with Bertsekas and Tsitsiklis [1991] in the context of cost minimization models. Since we formulate our models in terms of rewards we might think of them as stochastic longest path models but sill still refer to them as stochastic shortest path models to be consistent with the literature.

[10] Following the terminology of Bertsekas and Tsitsiklis [1991].

on the other hand (1.11) holds, the policy is said to be *improper*. Thus an SSP generalizes a transient model by allowing improper policies, but adds the condition that the expected total reward of an improper policy must have value $-\infty$ for at least one state. Consequently an improper policy cannot be optimal. Note that improper policies will contain closed classes of states in which there are negative rewards. On the other hand, an SSP model in which every stationary policy is proper is a transient model.

1.2.4 Positive models

we discuss positive and negative models in order to enable you to refer to the expected total reward literature. Moreover derivations of results for these models provide additional insight into the structure of Markov decision process models.

Positive and negative models are based on the decomposition of $v^\pi(s)$ in (1.5). We begin with positive models which were first formulated as those in which $r(s,a) \geq 0$ for all $s \in S$ and $a \in A_s$. The following definition is slightly more general and isolates the critical components of the model.

A *positive model* satisfies

1. $v_+^\pi(s) < \infty$ for all $s \in S$ and $\pi \in \Pi^{HR}$, and
2. for each $s \in S$, $r(s,a) \geq 0$ for some $a \in A_s$.

The first condition ensures that there are no policies with infinite expected total reward . Thus in a positive model it is possible to distinguish among policies with non-negative expected total reward. The second condition ensures that there is at least one (stationary) policy with non-negative expected total reward. Note that in this model, some policies may have $v^\pi(s) = -\infty$ but it follows from (1.5) that the first condition excludes the possibility that the limit defining the expected total reward does not exist.

1.2.5 Negative models

Negative models were first formulated as ones in which $r(s,a) \leq 0$ for all $s \in S$ and $a \in A_s$. The following definition is more general.

A *negative model* satisfies

1. $v_+^\pi(s) = 0$ for all $s \in S$ and $\pi \in \Pi^{HR}$, and
2. there exists a $\pi \in \Pi^{HR}$ for which $v_-^\pi(s) > -\infty$ for all $s \in S$.

The first condition ensures that $v^\pi(s) \leq 0$ for all policies and the second condition ensures there is at least one policy with a finite non-positive reward. Observe also that the limit defining $v^\pi(s)$ exists for all $\pi \in \Pi^{HR}$. Note that a subclass of models

with non-negative rewards in some states can be transformed to a negative model. We discuss this point below in the context of the grid-world navigation model.

1.2.6 Comparison of model classes

At first glance it might appear that by changing signs of rewards, positive and negative models are equivalent. This is not case because with an objective of maximizing the expected total reward, in positive models the decision maker seeks a policy with its expected total reward as far above zero as possible while in a negative model the decision maker seeks a policy with its expected total reward as close to zero as possible. By regarding negative rewards as costs, maximizing the expected total reward in a negative model is equivalent to minimizing the expected total cost.

Transient and SSP models allow more general cost structures than positive and negative models however there are two special cases to consider.

1. When all rewards are less than or equal zero, transient and SSP models are negative models.

2. an SSP model in which at least one stationary policy has a non-negative reward is equivalent to a positive model.

We now return to our examples.

Example 1.1 revisited: Recall that in the grid-world model, there is a reward of $B > 0$ for successfully delivering the coffee cup, a penalty of $X > 0$ (reward of $-X$) for falling down the stairs and a cost of c (reward of $-c$) for each transition. Since the (expected) positive reward for delivering the coffee can only be obtained from cells 2 and 4 (see Figure **??** in Section **??**), the condition that for each $s \in S$ there exists an $a \in A_s$ for which $r(s, a) \geq 0$ doesn't hold. Moreover the condition that $v_+^\pi(s) = 0$ in the negative model definition doesn't hold.

However, this model can be transformed into a negative model by subtracting B from all rewards (or just from the reward associated with delivering the coffee and adding it to the penalty for falling down the stairs). On the other hand two special cases are noteworthy, when $B = 0$, that is there is no reward for delivering coffee, this is negative model and when the penalty cost $X = 0$ and there is no transition cost, the model becomes a positive model. If, $B = 1$ and all other costs are 0, maximizing the expected total reward is equivalent to maximizing the probability that the robot succeeds at its task of delivering the coffee. So we see that positive models include those which maximize the probability of reaching a distinguished set of states.

On the other hand, there are two zero-reward absorbing states so that the model doesn't satisfy the first condition in the definition of a transient model. However by adding a zero-reward absorbing state Δ and deterministic zero-reward transitions from cells 1 and 7 to Δ, (See Figure **??** below), the first transient model condition is satisfied. When transitions are random, all stationary policies are proper so that the model is transient. When transitions are deterministic, infinite loops are possible so that there are improper stationary policies. However, since each transition has a reward of $-c$, when $c \neq 0$ such improper policies have expected total reward equal to $-\infty$ so that it is an SSP model. Thus in this example, the SSP classification allows deterministic actions while the transient model classification does not.

Example 1.2 revisited: Classifying the model in Example 1.2 is more problematic. Because of $v_+^{d_4^\infty}(s) = +\infty$ and $v_-^{d_4^\infty}(s) = -\infty$ the model is neither a positive or negative model. Moreover subtracting 1 from all rewards satisfies condition 1. of a negative model but after this transformation, $v_-^\pi(s) = -\infty$ for each policy so condition 2 does not hold.

Moreover it is not an SSP model because the improper policy d_3^∞ does not have an expected total reward equal to $-\infty$.

1.3 Optimal policies and the Bellman Equation

Define the optimal value function $v^*(s)$ for all $s \in S$ by

$$v^*(s) := \sup_{\pi \in \Pi^{HR}} v^\pi(s). \tag{1.12}$$

What this condition means is that given $\epsilon > 0$ for each $s \in S$, there exists a $\pi^* \in \Pi^{HR}$ (which can vary with s) for which

$$v^{\pi^*}(s) \geq v^*(s) - \epsilon.$$

While the above definition implicitly requires that $v^\pi(s)$ exist for each $\pi \in \Pi^{HR}$, it can be relaxed to require that $v^*(s)$ be the smallest $v(s)$ that satisfies

$$v^*(s) \geq \limsup_{N \to \infty} E^\pi \left\{ \sum_{n=1}^{\infty} r(X_n, Y_n) \middle| X_1 = s \right\} \tag{1.13}$$

for all $\pi \in \Pi^{HR}$. Although this more general definition allows inclusion of examples such Example 1.2, we will assume the following from here so as to exclude this possibility.

Assumption 6.1 - Either $v_+^\pi(s)$ or $v_-^\pi(s)$ is finite for all $s \in S$ for all $\pi \in \Pi^{HR}$.

Clearly this assumption is satisfied in transient, SSP and positive and negative models.

Following a similar argument to Theorem **??** we obtain

Proposition 1.1. For each $s \in S$,

$$v^*(s) = \sup_{\pi \in \Pi^{MR}} v^\pi(s). \tag{1.14}$$

This proposition establishes that for each state, the supremum over randomized history dependent policies equals the supremum over randomized Markovian policies. Recall that the underlying result used to prove this proposition was the key Lemma **??** which showed that for any history dependent policy, for each state there exists a randomized Markovian policy with the same transition probabilities.

1.3.1 The Bellman equation in an expected total reward model

In this section we investigate properties of the Bellman equation for expected total reward models. In component notation, the Bellman equation can be expressed as:

$$v(s) = \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) v(j) \right\} \quad \text{for all } s \in S. \tag{1.15}$$

In vector notation it is given by:

$$\mathbf{v} = \underset{d\in D^{MD}}{\text{c-max}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}\} \tag{1.16}$$

Observe that the Bellman equation in expected total reward models is identical to that in the discounted model with $\lambda = 1$.

In the special case that D^{MD} consists of a single decision rule d, (1.15) becomes the (deterministic stationary) policy evaluation equation:

$$v(s) = r(s, d(s)) + \sum_{j\in S} p(j|s, d(s)v(j) \tag{1.17}$$

defined for all $s \in S$. In vector notation (1.16) reduces to the policy evaluation equation:

$$\mathbf{v} = \mathbf{r}_d + \mathbf{P}_d\mathbf{v}. \tag{1.18}$$

As before, we define the *Bellman operator*, L and the *policy evaluation* operator, L_d defined for $\mathbf{v} \in V$[11] and taking values in V by

$$L\mathbf{v} := \underset{d\in D^{MD}}{\text{c-max}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}\} \quad \text{and} \quad L_d\mathbf{v} := \mathbf{r}_d + \mathbf{P}_d\mathbf{v}. \tag{1.19}$$

Consequently the Bellman equation and policy evaluation equations can be written in vector notation as:

$$\mathbf{v} = L\mathbf{v} \quad \text{and} \quad \mathbf{v} = L_d\mathbf{v}. \tag{1.20}$$

(Do we need the following in the sequel?)

We restate Lemma **??** when $\lambda = 1$ for convenience as follows:

Lemma 1.1. For any $\mathbf{v} \in V$,

$$\underset{d\in D^{\text{MD}}}{\text{c-max}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}\} = \underset{d\in D^{\text{MR}}}{\text{c-sup}}\{\mathbf{r}_d + bP_d\mathbf{v}\}. \tag{1.21}$$

It enables us to restrict attention to Markovian deterministic decision rules when defining the operator L.

1.3.2 The transient Bellman equation

In transient models, the Bellman equation is only of interest on the set of transient states $S - \Delta$. since the definition of a transient model implies $v^{d^\infty}(\Delta) = 0$ for any

[11] Recall that V denotes the set of real valued function on S.

$d \in D^{MD}$. Let $\mathbf{v}_T$ denote the vector of components of $\mathbf{v}$ restricted to transient states. and let V_T denoted the set of bounded functions on $S - \Delta$. Then using the matrix partitioning in (1.7)the Bellman equation can be expressed as:

$$\begin{bmatrix} \mathbf{v}_T \\ v(\Delta) \end{bmatrix} = \underset{d \in D^{MD}}{\text{c-max}} \left\{ \begin{bmatrix} (\mathbf{r}_d)_T \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{Q}_d & \mathbf{R}_d \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{v}_T \\ v(\Delta) \end{bmatrix} \right\}$$

Since $v(\Delta) = 0$, the Bellman equation in a transient model reduces to :

$$\mathbf{v}_T = \underset{d \in D^{MD}}{\text{c-max}} \{(\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}_T\}. \tag{1.22}$$

We refer to (1.33) as the *transient Bellman equation.*

Let V_T denoted the set of bounded functions on $S - \Delta$ and define the operator $L_T : V_T \to V_T$ by

$$L_T \mathbf{v} := \underset{d \in D^{MD}}{\text{c-max}} \{(\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}_T\}. \tag{1.23}$$

Then the transient Bellman equation can be represented in operator form by:

$$\mathbf{v}_T = L_T \mathbf{v}_T. \tag{1.24}$$

By replacing D^{MD} by a single decision rule d, the policy evaluation equation in a transient model reduces to:

$$\mathbf{v}_T = (\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}_T =: (L_d)_T \mathbf{v}_T. \tag{1.25}$$

We will establish below that as a consequence of the observation in a transient model that at least one row sum of Q_d is strictly less than 1, the transient Bellman operator is a contraction mapping so that it can be anlayzed using a similar approach used for discounted models.

1.3.3 Solutions of the Bellman equation in expected total reward models

In this section we will establish some properties of the Bellman operator that apply for any model class. We will show that:

1. the operator L is monotone,
2. the solution of the Bellman equation is not unique, and
3. the value function satisfies the Bellman equation.

In contrast to the discounted model, without further assumptions, the Bellman operator L is not a contraction mapping so we cannot rely on properties of contraction mappings to establish existence of solutions to the Bellman equation.

We begin with the following easily proved result.

Proposition 1.2. Let $\mathbf{u}$ and $\mathbf{v}$ be elements of V. Then

1. For any scalar c, $L(\mathbf{v} + c\mathbf{e}) = L\mathbf{v} + c\mathbf{e}$.
2. If $\mathbf{u} \geq \mathbf{v}$, $L\mathbf{u} \geq L\mathbf{v}$.

Proof. To prove the first part,

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

The second part follows by noting that there exists a $d \in D^{MD}$ for which

$$L\mathbf{v} = \mathbf{r}_d + \mathbf{P}_d\mathbf{v} \leq \mathbf{r}_d + \mathbf{P}_d\mathbf{u} \leq L\mathbf{u}.$$

□

Next we show that $v^*(s)$ is a solution of the optimality equation. This proof also applies when $\lambda < 1$ however we used a different approach in that case. **(We prove this under Assumption 6.1? Is it needed?)**

Theorem 1.1. Suppose Assumption 6.1 holds. Then the optimal value function $\mathbf{v}^*$ is a solution of the Bellman equation.

Proof. We first show that $v^* \geq L\mathbf{v}^*$. From the definition of "sup", given $\epsilon > 0$ for each $s \in S$, there exists a $\pi_s^* \in \Pi^{HR}$ for which

$$v^{\pi_s^*}(s) \geq v^*(s) - \epsilon. \tag{1.26}$$

Since for all $a \in A_s$

$$v^*(s) \geq r(s,a) + \sum_{j \in S} p(j|s,a) v^{\pi_j^*}(j)$$

It follows from the monotonicity of L (part 2. of Proposition 1.2)

$$v^*(s) \geq \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) v^{\pi_j^*}(j) \right\} \geq \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) v^*(j) \right\} - \epsilon$$

Since the result holds for every $\epsilon > 0$, $v^*(s) \geq Lv^*(s)$ for all $s \in S$.

Now we show $\mathbf{v}^* \leq L\mathbf{v}^*$. Writing π_s^* in (1.26), as $\pi_s^* = (a_s, \pi_s')$, that is in state s it chooses action a_s at the first decision epoch and then uses the history dependent randomized policy, π_s', it follows that

$$v^*(s) - \epsilon \leq v^{\pi_s^*}(s) = r(s,a_s) + \sum_{j \in S} p(j|s,a_s) v^{\pi_j'}(j) \leq \max_{a \in A_S} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a) v^*(j) \right\}.$$

Since ϵ was arbitrary, it follows that $v^*(s) \leq L\mathbf{v}^*(s)$.

Thus $L\mathbf{v}^* = \mathbf{v}^*$. □

Hence we conclude that that $\mathbf{v}^*$ is a solution of the Bellman equation, however as a consequence of part 1. of Proposition 1.2, the solution is not unique. Note that Theorem 1.1 also holds when

$$v^*(s) = \sup_{\pi \in \Pi^{HR}} \limsup_{N\to\infty} E^\pi \left\{ \sum_{n=1}^{N} r(X_n, Y_n) \right\}$$

but the proof is a bit more subtle and we leave it as an exercise.

As an immediate corollary we have

Corollary 1.1. For every $d \in D^{MR}$, $L_d \mathbf{v}^{d^\infty} = \mathbf{v}^{d^\infty}$.

1.3.4 Existence of optimal stationary policies in expected total reward models*

In this section we show that under mild assumptions, in a finite state and action Markov decision process, there exists a stationary policy that maximizes the expected total reward. Our proof is rather technical and uses the result (Theorem ??) that in a discounted model, for each non-negative $\lambda < 1$ there exists an optimal stationary policy.

We begin with a technical result. It's proof is a direct application of Abel's Theorem[12] which appears as Theorem 8.2 in Rudin [1964].

Lemma 1.2. Suppose for some $\pi \in \Pi^{HR}$ and $s \in S$

$$\lim_{N\to\infty} E_s^\pi \left\{ \sum_{n=0}^{N} r(X_n, Y_n) \,\middle|\, X_1 = s \right\} = v^\pi(s) \tag{1.27}$$

exists. Then

$$\lim_{\lambda\uparrow 1} v_\lambda^\pi(s) = v^\pi(s). \tag{1.28}$$

Note that the assumption that the limit in (1.27) exists cannot be easily relaxed. To see this we return briefly to Example 1.2.

[12] Abel's Theorem states that if $\sum_{n=0}^{\infty} x_n$ converges, and $f(\lambda) = \sum_{n=0}^{\infty} \lambda^n x_n$ converges for $|\lambda| < 1$, then $\lim_{\lambda\uparrow 1} f(\lambda) = \sum_{n=0}^{\infty} x_n$.

Example 1.2 revisited: Consider stationary policy d_4^∞. Letting $\lambda < 1$, we have that

$$v_\lambda^{d_4^\infty}(s_1) = 1 - \lambda + \lambda^2 - \lambda^3 + ... = \frac{1}{1+\lambda}.$$

Hence $\lim_{\lambda\uparrow 1} v_\lambda^{d_4^\infty}(s_1) = \frac{1}{2}$ but $v^{d_4^\infty}(s_1)$ does not exist. Moreover

$$\liminf_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = 0 < \lim_{\lambda\uparrow 1} v_\lambda^{d_4^\infty}(s_1) < \limsup_{N\to\infty} \sum_{n=1}^{N} r(X_n, d_4(X_n)) = 1,$$

so that the limit of $v_\lambda^{d_4^\infty}(s_1)$ provides no insight about the "lim inf" or "lim sup" of the partial sums of (expected) rewards.

Theorem 1.2. Suppose in an expected total reward model, the limit in (1.27) exists for all $\pi \in \Pi^{HR}$ and $s \in S$. Then there exists an optimal stationary deterministic policy.

Proof. From Theorem ?? there exists an optimal deterministic stationary policy for all non-negative $\lambda < 1$ in a discounted model. Let $\{\lambda_n\}$ denote a sequence of λ's which converges monotonically to 1. Then for each λ_n there exists an optimal deterministic stationary policy. Since there are only finitely many deterministic stationary policies, there must exist a deterministic stationary policy $(d^*)^\infty$ and a subsequence $\{\lambda_{n_k}\}$ for which $(d^*)^\infty$ is optimal.

Therefore for all $\pi \in \Pi^{HR}$ and $s \in S$,

$$v_{\lambda_{n_k}}^{\pi}(s) \le v_{\lambda_{n_k}}^{(d^*)^\infty}(s).$$

Hence from Lemma 1.2, both limits below exist so that

$$v^{\pi}(s) = \lim_{k\to\infty} v_{\lambda_{n_k}}^{\pi}(s) \le \lim_{k\to\infty} v_{\lambda_{n_k}}^{(d^*)^\infty}(s) = v^{(d^*)^\infty}(s). \tag{1.29}$$

From this we conclude that $(d^*)^\infty$ is an optimal policy. □

We have stated Theorem 1.2 in considerable generality, clearly the hypothesis holds under Assumption 6.1. The following corollary distinguishes some models in which this condition holds but also extends to SSP models for the reasons discussed below.

Corollary 1.2. There exists an optimal deterministic stationary policy in

1. a positive model,
2. a negative model,
3. a transient model, and

4. an SSP model.

Note that the the hypothesis of Theorem 1.2 excludes SSP models with improper policies, but since such policies cannot be optimal, it is sufficient to restrict attention to proper policies for which (1.27) exists. Hence the result also applies to SSP models.

1.3.5 Identification of optimal policies

Next we discuss how to identify an optimal policy. Analogously to discounted models, we would suspect that if

$$d^* \in \underset{d\in D^{MD}}{\text{c-max}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}^*\}. \tag{1.30}$$

then d^∞ was optimal. The following simple example shows that is not case without adding further conditions such as those in Theorem 1.3 below.

Example 1.3. Let $S = \{s_1, s_2\}, A_{s_1} = \{a_{1,1}, a_{1,2}\}, A_{s_2} = \{a_{2,1}\}, r(s_1, a_{1,1}) = 0, r(s_1, a_{1,2}) = 1, r(s_2, a_{2,1}) = 0$ and $p(s_1|s_1, a_{1,1}) = p(s_2|s_1, a_{1,2}) = p(s_2|s_2, a_{2,1}) = 1$. This is an example of a positive model. If it is modified by adding a zero-reward absorbing state Δ and changing transition probabilities for actions $a_{1,1}$ and $a_{2,1}$ to $p(\Delta|s_1, a_{1,1}) = p(\Delta|s_2, a_{2,1}) = 1$, it can be classified as a transient model.

The Bellman equation for this model may be written as:

$$v(s_1) = \max\{v(s_1), 1 + v(s_2)\} \quad \text{and} \quad v(s_2) = v(s_2).$$

Since $v^*(s_1) = 1$ and $v^*(s_2) = 0$, it follows that both $d_1 = \{a_{1,1}, a_{2,1}\}$ and $d_2 = \{a_{1,2}, a_{2,1}\}$ are in $\arg\,\text{c-max}_{d\in D^{MD}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}^*\}$ but only d_2^∞ is optimal.

Our key result follows.

Theorem 1.3. Suppose d^* satisfies (1.30) then $\mathbf{v}^{(d^*)^\infty} = \mathbf{v}^*$ if

$$\limsup_{N\to\infty} \mathbf{P}_{d^*}^N \mathbf{v}^* \le 0. \tag{1.31}$$

Proof. Iteratively applying Theorem 1.1 and (1.30), yields

$$\mathbf{v}^* = \mathbf{r}_{d^*} + \mathbf{P}_{d^*}\mathbf{v}^* = \mathbf{r}_{d^*} + \mathbf{P}_{d^*}\mathbf{r}_{d^*} + \mathbf{P}_{d^*}^2\mathbf{v}^* = \cdots = \sum_{n=1}^{N} \mathbf{P}_{d^*}^{n-1} + \mathbf{P}_{d^*}^N\mathbf{v}^*. \tag{1.32}$$

Hence, the result holds if (1.31) holds. □

We note that the form of the rewards and/or the structure of transition probabilities corresponding to d^* determines whether (1.31) holds. We summarize these results in the following corollary:

Corollary 1.3. Suppose d^* satisfies (1.30). Then $(d^*)^\infty$ is optimal in:

1. a negative model,
2. a positive model in which (1.31) holds, or
3. a transient model, and
4. in an SSP model in which d^* is proper.

Returning to Example 1.3, since the model can be classified as a positive model, it requires the addition of condition (1.31) to establish the optimality of d_2^∞ since $\mathbf{P}_{d_1}^N \mathbf{v}^*(s_1) = 1$ for all N. On the other hand, under the modification used to obtain a transient model described in the example, only d_1^∞ attains the maximum in the Bellman equation.

Now we return to Example 1.2.

Example 1.2 revisited: Recall that the model in Example 1.2 fell under none of the model classifications. Its Bellman equation may be written as:

$$v(s_1) = \max\{-1 + v(s_1), 1 + v(s_2)\} \quad \text{and} \quad v(s_2) = \max\{v(s_2), -1 + v(s_1)\}$$

and clearly $v^*(s_1) = 1$ and $v^*(s_2) = 0$ (taking into consideration that the limit defining $v^{d_4^\infty}(s)$ does not exist). Observe that in state s_2 both actions achieve the maximum on the right hand side of the Bellman equation, but only d_3 which satisfies (1.31) is optimal. Thus this condition can be used to identify optimal policies even in the case when none of the model classifications apply.

1.4 Value Iteration

Next we explore computation in expected total reward models. First we consider value iteration which we express in vector notation as:

Value iteration for an expected total reward model

1. Specify $\mathbf{v}$ and $\epsilon > 0$.
2. While $\|\mathbf{v}' - \mathbf{v}\| > \epsilon$:
   (a) $\mathbf{v}' \leftarrow L\mathbf{v} = \text{c-max}_{d \in D^{MD}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}\}$
   (b) $\mathbf{v} \leftarrow \mathbf{v}'$
3. Return $\mathbf{v}_\epsilon = \mathbf{v}$ and
$$d_\epsilon \in \arg\underset{d \in D^{MD}}{\text{c-max}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}_\epsilon\}$$

Note that we represent value iteration with a norm based stopping criterion. We found in most examples that the norm based criterion gave the same results as a span based criterion (see Section **??**)[13]. Note also that when applying value iteration to transient models, we base recursions on the transient Bellman operator L_T.

1.4.1 Examples

Before analyzing convergence properties of value iteration, we consider a few examples.

Example 1.3 revisited: The value iteration recursion for this model may be written in component form as:

$$v'(s_1) = \max\{v(s_1), 1 + v(s_2)\} \quad \text{and} \quad v'(s_2) = v(s_2).$$

So starting with $v(s_1) = v(s_2) = 0$, we find that after one iteration, $v'(s_1) = v^*(s_1) = 1$ and $v'(s_1) = v^*(s_2) = 0$.

Example 1.2 revisited: Recall that the model in Example 1.2 did not fall into any of the specified model classes. Since the value iteration recursion may be written as:

$$v'(s_1) = \max\{-1 + v(s_1), 1 + v(s_2)\} \quad \text{and} \quad v'(s_2) = \max\{v(s_2), -1 + v(s_1)\}$$

starting value iteration at $v(s_1) = v(s_2) = 0$, we see that at the next and all subsequent iterations, $v'(s_1) = 1$ and $v'(s_2) = 0$ so that $\mathbf{v}' = \mathbf{v}^*$.

Next we consider the more interesting grid world model.

[13] We observed that the iterates remain unchanged on states that lead to absorbing states, for example, if in state $s \in S$ there is a single action leading to zero-reward absorbing state Δ with reward c, then value iteration will eventually set $v(s) = c$ so that the quantity involved in defining the span, $\min_{s \in S}\{v(s) - v'(s)\} = 0$ so that $sp(\mathbf{v} - \mathbf{v}') = \|\mathbf{v} - \mathbf{v}'|$.

Example 1.1 revisited: We will focus primarily on computational issues here, we will explore the structure of policies below. We apply value iteration in three instances:

Instance 1: $B = 50, X = 200$ and movement cost $c = 1$,

Instance 2: $B = 0, X = 150$ and $c = 1$, and

Instance 3: $B = 1, X = 0$ and $c = 0$.

As noted above Instance 1 can be transformed into a negative model, Instance 2 is a negative model and Instance 3 is a positive model. On the other hand, when the probability the robot moves in the intended direction, $p < 1$, the model is transient. When this probability is 1, it is an SSP model.

We express the value iteration recursion in selected states as follows. We let $q = 1 - p$ and assume the only actions are "left", "up" and "right". We leave it as an exercise to write out the entire recursion.

$$
\begin{aligned}
v(1) &= v(1) \\
v(2) &= -c + \max\{p[B + v(1)] + \frac{1}{2}q[v(3) + v(5)], pv(3) + \frac{1}{2}q[B + v(1) + v(5)]\}, \\
v(3) &= -c + pv(2) + qv(6), \\
v(8) &= -c + \max\{p[-X + v(7)] + \frac{1}{3}q[v(5) + v(9) + v(11)], \\
&\qquad pv(5) + \frac{1}{3}q[-X + v(7) + v(9) + v(11)], \\
&\qquad\quad pv(9) + \frac{1}{3}q[-X + v(7) + v(5) + v(11)]\}.
\end{aligned}
$$

These recursions are obtained by noting that in cell 2 there are 2 possible actions, "left" or "right", in cell 3, there is only one action "left", and in cell 8, all three actions are possible, however "left" would be inadvisable for large p.

We start value iteration from $\mathbf{v} = \mathbf{0}$ and use a stopping criterion $\|\mathbf{v}' - \mathbf{v}\| < 0.0001$. We observe that in all instances value iteration converges; the number of iterations to achieve the stopping criterion varies with p and instance as shown in Table 1.1. We observed that:

1. For each p the number iterations varied by instance with Instance 1 and 2 the most similar. Moreover convergence was faster in Instance 3 for all values of p.

2. In each instance the number of iterations to achieve convergence was decreasing with p.

3. In instance 1, the successive values of $v(13)$ were non-monotone when $p \geq 0.6$. In Instance 2, the successive values of $v(13)$ were monotone non-increasing and in Instance 3 the successive values of $v(13)$ were monotone non-decreasing for all values of p.

p	Instance 1	Instance 2	Instance 3
0	480	481	305
0.25	199	201	101
0.50	114	124	81
0.75	51	44	38
0.95	21	19	17

Table 1.1: Columns 2 - 4 give number of iterations to achieve stopping criterion $\|\mathbf{v}' - \mathbf{v}\| < 0.0001$ in grid world model as a function of p and instance.

1.4.2 Convergence of value iteration

We focus on transient and SSP models in which value iteration is based on the transient Bellman operator. In such models we can construct a norm under which the transient Bellman operator L_T is a contraction on V_T. Hence, as in the case of discounted models, we can use the Banach fixed point theorem (Theorem **??** to establish convergence and error bounds. In addition it also guarantees existence of a solution of the transient Bellman equation. Subsequently we establish convergence of value iteration in positive and negative models using monotonicity properties of the Bellman operator however in such models, no error bounds are available.

1.4.3 Value iteration in transient models

In our analysis of the grid-world model we found in all instances that its rate of convergence depended on p, the probability the robot moves in the intended direction. Hence there appears to be some underlying contraction properties in play. Delving more deeply into this issue, we saw that the spectral radius of $\mathbf{Q}_d$[14] was smallest when p was closest to 1 in absolute value. Our analysis of the convergence of value iteration in transient models provides insight into this issue.

Looking more carefully at the iterates of value iteration in the grid-world model we observe that there is no decision in cells 7 and 1 so $v^n(1) = v^n(7) = 0$ for $n \geq 1$. Hence our computations above are in essence applying value iteration only on transient states.

Without loss of generality assume that all deterministic stationary policies have a single reward-free absorbing state Δ. Let $\mathbf{v}_T$ and $\mathbf{v}'_T$ denote the vector of components of of $\mathbf{v}$ and $\mathbf{v}'$ restricted to transient states, that is $S - \Delta$. we base value iteration on the transient Bellman operator as follows:

[14]Recall that $\mathbf{Q}_d$ denotes the restriction of the transition matrix to transient states of the Markov chain with transition matrix $\mathbf{P}_d$.

$$\mathbf{v}'_T = \underset{d\in D^{MD}}{\text{c-max}}\{(\mathbf{r}_d)_T + \mathbf{Q}_d\mathbf{v}_T\} := L_T\mathbf{v}_T. \tag{1.33}$$

Hence analysis of value iteration in transient models depends on the properties of $\mathbf{Q}_d$.

Theorem 1.1 showed that in a model with expected total reward criterion, $\mathbf{v}^*$ was a solution of the Bellman equation. Under the additional restriction to transient models, we obtain the following easily proved result.

Theorem 1.4. In a transient model, $\mathbf{v}^*_T$ is the unique solution of the transient Bellman equation $L_T\mathbf{v} = \mathbf{v}$.

Proof. From part 1. of Proposition 1.2, $\mathbf{v}^* + c\mathbf{e}$ satisfies the Bellman equation for any scalar c. Hence for some $d' \in D^{MD}$,

$$\mathbf{v}^*_T + c\mathbf{e} = L_T(\mathbf{v}^* + c\mathbf{e}) \leq L_T\mathbf{v}^* + \underset{d\in D}{\text{c-max}}\, c\mathbf{Q}_d\mathbf{e} = \mathbf{v}^*_T + c\mathbf{Q}_{d'}\mathbf{e}.$$

For some $s \in S$, the row sum of $\mathbf{Q}_{d'}$ must be strictly less than 1. Denoting such a row sum by k where $0 \leq k < 1$ the above inequality becomes $c \leq ck$ which implies $c = 0$. □

Establishing convergence

To justify the observed fast convergence of value iteration in a transient model, we show below that:

1. L^N_T is a contraction mapping for some $N > 0$,
2. there is an equivalent Markov decision process in which L_T is a contraction mapping, or
3. L_T is a contraction mapping with respect to a specific weighted max norm.

In each of these cases, we can apply the Banach fixed-point Theorem (Theorem **??**) to establish:

Theorem 1.5. In a transient model value iteration converges geometrically to the unique fixed point of L_T.

Note that the convergence rate is based on either regarding L_T as an N-stage contraction or a weighted max norm.

N stage contractions

By the defining property of a transient model, for each $d \in D^{MD}$ from some state in $S - \Delta$ there is a positive probability of reaching Δ in one transition. Hence for all states in $S - \Delta$, there is positive probability of reaching Δ in *at most* $N = |S - \Delta|$ transitions. Therefore there exist a positive $\alpha < 1$ for which $\|\mathbf{Q}_d^N\| < \alpha < 1$ for all $d \in D^{MD}$. Hence the operator L_T satisfies

$$\|L_T^N \mathbf{v}_T - L_T^N \mathbf{v}'_T\| \leq \alpha \|\mathbf{v}_T - \mathbf{v}'_T\|$$

for any $\mathbf{v}_T$ and $\mathbf{v}'_T$ in V_T.

Positively similar Markov decision processes *

In this section we describe Veinott's elegant analysis of the transient model. In particular we show that value iteration convergences geometrically.

Since for any matrix A, $\rho(A) \leq \|A\|$ it follows that $\rho(\mathbf{Q}_d^N) \leq \|\mathbf{Q}^N\| < 1$ for all $d \in D^{MD}$. But the largest eigenvalue of $\mathbf{Q}_d^N = \lambda_d^N$ where λ_d denotes the largest eigenvalue of $\mathbf{Q}_d$. Hence $\rho(\mathbf{Q}_d) < 1$. Thus it follows that;

$$\max_{d \in D} \rho(\mathbf{Q}_d) := \alpha' < 1. \tag{1.34}$$

It is not true that $\rho(\mathbf{Q}_d) \geq \|\mathbf{Q}_d\|$[15] but we can construct a positively similar Markov decision process for which this almost holds.

Two Markov decision processes with the same state and action spaces are said to be *positively similar* if there exists a positive diagonal matrix $\mathbf{B}$ such that $\tilde{\mathbf{r}}_d = \mathbf{B}\mathbf{r}_d$ and $\tilde{\mathbf{P}}_d = \mathbf{B}\mathbf{P}_d\mathbf{B}^{-1}$. The consequence of this definition is that when two Markov decision processes are positively similar:

1. they have the same sets of transient policies,
2. they have the same optimal policy,
3. in a transient model, the iterates of value iteration converge geometrically to the value function.

We construct a positively similar Markov decision process by formulating and solving an expected total reward Markov decision process that maximizes the number of transitions to reach Δ from each $s \neq \Delta$ as follows.

Consider a transient Markov decision process for which $(\mathbf{r}_d)_T = \mathbf{1}$ for all $d \in D^{MD}$. Then its transient Bellman equation becomes

$$\mathbf{v}_T = \underset{d \in D^{MD}}{\text{c-max}} \{\mathbf{1} + \mathbf{Q}_d \mathbf{v}_T\}. \tag{1.35}$$

---

[15] As an example where this inequality does not hold, consider the transient matrix $\mathbf{Q} = \begin{bmatrix} 0 & \frac{1}{4} \\ 1 & 0 \end{bmatrix}$. Then $\rho(\mathbf{Q}) = \frac{1}{2}$ and $\|\mathbf{Q}\| = 1$.

For any $d \in D^{MD}$, $\mathbf{v}_T^{d^\infty} \geq \mathbf{1}$ so we conclude that $\mathbf{v}_T \geq \mathbf{1}$. When $\mathbf{Q}_d \neq \mathbf{0}$, this inequality is strict[16].

Let $\hat{\mathbf{w}}$ denote the unique solution of (1.35) and set $\mathbf{W}$ equal to the $|S-\Delta| \times |S-\Delta|$ diagonal matrix with entries $1/\hat{w}(s)$. Then multiplying both sides of (1.35) on the left by $\mathbf{W}$ we obtain

$$\mathbf{W}\hat{\mathbf{w}} - \mathbf{W}\mathbf{1} = \underset{d \in D^{MD}}{\text{c-max}}\, \mathbf{W}\mathbf{Q}_d\hat{\mathbf{w}} = \underset{d \in D^{MD}}{\text{c-max}}\, \mathbf{W}\mathbf{Q}_d\mathbf{W}^{-1}\mathbf{1}$$

where the last equality follows by noting that $\hat{\mathbf{w}} = \mathbf{W}^{-1}\mathbf{1}$. Rewriting this in component notation gives

$$1 - \frac{1}{\hat{w}(s)} = \max_{a \in A_s}\left\{\frac{\sum_{j \in S-\Delta} p(j|s,a)\hat{w}(j)}{\hat{w}(s)}\right\} = \max_{a \in A_s}\{\sum_{j \in S-\Delta} \tilde{p}(j|s,a)\}$$

where

$$\tilde{p}(j|s,a) := \frac{\sum_{j \in S-\Delta} p(j|s,a)\hat{w}(j)}{\hat{w}(s)}.$$

Hence there exists an $\alpha > 0$

$$1 > \alpha = \max_{s \in S-\Delta}\left\{1 - \frac{1}{\hat{w}(s)}\right\} = \max_{a \in A_s}\max_{s \in S-\Delta}\{\sum_{j \in S-\Delta} \tilde{p}(j|s,a)\} = \underset{d \in D^{MD}}{\text{c-max}}\, \|\tilde{\mathbf{Q}}_d\| \qquad (1.36)$$

where $\tilde{\mathbf{Q}}_d$ denotes the matrix with components $\tilde{p}(j|s,d(s))$.

Thus if we take our original Markov decision process to be one with rewards $\mathbf{r}_d$ and transition probabilities $\mathbf{P}_d$ and extend the definition of $\hat{w}(s)$ from $S-\Delta$ to S by setting $\hat{w}(\Delta) = 1$, we have that (1.36) holds for the positively similar Markov decision process with rewards $\tilde{\mathbf{r}}_d = \mathbf{W}\mathbf{r}_d$ and transition probability matrix $\tilde{\mathbf{P}}_d = \mathbf{W}\mathbf{P}_d\mathbf{W}^{-1}$ defined for all $d \in D^{MD}$. We leave it as an exercise to show that $\tilde{L}_T\mathbf{v} := \text{c-max}_{d \in D^{MD}}\{\tilde{\mathbf{r}}_d + \tilde{\mathbf{Q}}_d\mathbf{v}\}$ is a contraction mapping in the transformed Markov decision process. Hence by the Banach fixed point theorem (Theorem **??**), it follows that that

Theorem 1.6. $\tilde{L}_T$ has a unique fixed point $\tilde{\mathbf{v}}^*$ and for any $\mathbf{v} : S - \Delta \to \Re$, $\tilde{L}_T^n\mathbf{v}$ converges geometrically to $\tilde{\mathbf{v}}^*$.

Thus it follows that:

Corollary 1.4. L_T has a unique fixed point $\mathbf{v}^*$ and for any $\mathbf{v} : S - \Delta \to \Re$, $L_T^n\mathbf{v}$ converges geometrically to $\mathbf{v}^*$.

[16]For example when $|S-\Delta| = 1$ and $\mathbf{Q}_d = 0, \mathbf{v}^{d^\infty} = 1$.

A weighted norm approach*

? and later Bertsekas and Tsitsiklis [1991] extend Veinott's approach in several ways. ? use the *weighted max-norm*[17]. on real-valued functions on S (or on $S - \Delta$) by

$$\|\mathbf{v}\|_{\mathbf{w}} := \max_{s \in S} \left| \frac{v(s)}{w(s)} \right| \tag{1.37}$$

where $w(s) > 0$ for all $s \in S$. Let V_w denote the set of real valued functions on $|S|$ that are bounded in the weighted max-norm. F or each $\mathbf{v} \in V_w$ there exists a $\|\mathbf{v}\|_{\mathbf{w}} \leq M$ for some $M > 0$. Consequently for all $s \in S$.

$$|v(s)| \leq M w(s)$$

Thus boundedness in this norm implies a state-dependent bound. If S is ordinal, this corresponds to a bounded growth rate

The corresponding *subordinate* matrix norm, defined on $|S| \times |S|$ matrices $\mathbf{A}$ is given by

$$\|\mathbf{A}\|_b w := \max_{s \in S} \sum_{j \in S} \frac{|a(j|s)w(j)|}{w(s)}. \tag{1.38}$$

When $\|\mathbf{A}\|_w \leq M'$,

$$\sum_{j \in S} w(j)a(j|s) \leq M' w(s).$$

Hence the boundedness in the weighted max norm imposes growth conditions on values and structure on matrices. Weighted supremum norms also play an important role in countable states models and approximate dynamic programs.

We now show how this is related to the construction of a positively similar Markov decision process above. It is easy to see that

$$\|Q_d\|_w = \|\tilde{\mathbf{Q}}_d\|. \tag{1.39}$$

where the norm on the left is the weighted max-norm on the original Markov decision process while the norm on the right is the *unweighted* max-norm applied to the constructed positively similar Markov decision process.

Hence setting $\mathbf{w} = \hat{\mathbf{w}}$ and noting (1.36) it follows that in transient model:

Proposition 1.3. L_T is a contraction mapping with respect to the weighted max norm $\|\cdot\|_w b$.

Hence it follows that in the **original** model value iteration converges geometrically in the weighted max norm. What this means is that

$$\|L_T^n \mathbf{v}^0 - \mathbf{v}^*\|_{\hat{\mathbf{w}}} \leq \alpha^n \|\mathbf{v}^0 - \mathbf{v}^*\|_{\hat{\mathbf{w}}}. \tag{1.40}$$

[17]Usually it is called a supremum norm but since we restrict attention to finite-state models in which the maximum is attained, we refer to it as a max norm

SSP models

The above analysis shows that value iteration converges in an SSP model in which every policy is proper. Bertsekas and Tsitsiklis [1991] show this also happens in an SSP model with improper policies however the convergence may not be at a geometric rate.

1.4.4 Convergence of value iteration in positive and negative models.*

First we note that if we begin value iteration with $\mathbf{v}^0 = \mathbf{0}$, then the sequence of iterates of value iteration equals the value of a deterministic Markovian policy in a finite horizon problem with terminal reward zero[18]. That is

Lemma 1.3. Suppose $\{\mathbf{v}^n\}$ is generated by value iteration with $\mathbf{v}^0 = \mathbf{0}$. Then for each $N \geq 1$,

$$\mathbf{v}^n = \mathbf{v}_N^\pi = \sum_{n=1}^{N} \mathbf{P}_{d_1} ... \mathbf{P}_{d_n} \mathbf{r}_{d_n} \tag{1.41}$$

where $\pi \in \Pi^{MD}$ and $\pi = (d_1, d_2, ...)$.

We distinguish analysis by model class.

Positive models

We observe in our analysis of the grid-world example that convergence was monotone in Instance 3 which was an example of a positive model. We show that this monotonicity holds for any positive model and consequently provides the basis for a convergence proof.

We begin with a technical result that summarizes useful properties of the Bellman operator L in positive models. To do so, we introduce the set of non-negative values, $V^+ := \{\mathbf{v} : S \to \Re : \mathbf{v} \geq \mathbf{0}\}$.

Proposition 1.4. In a positive model:

1. There exists a $d \in D^{MD}$ for which $\mathbf{v}^{d^\infty} \geq \mathbf{0}$,
2. $L\mathbf{0} \geq \mathbf{0}$,
3. $\mathbf{v}^* \geq \mathbf{0}$,
4. $L : V^+ \to V^+$, and

[18] This policy is also optimal in the corresponding finite horizon problem.

5. for any $d \in D^{MD}$, $r_d(s) \leq 0$ for all s in each recurrent set of $\mathbf{P}_d$.

Proof. Part 1. is an immediate consequence of condition 2. in the positive model definition. Parts 2. and 3. follow immediately from Part 1. Part 4. is an immediate consequence of Proposition 1.2 and Part 5. is an immediate consequence of the the first condition defining a positive model. □

If we return to Example 1.2 and delete action $a_{2,2}$, then we see that the model exhibits all of the results in the above proposition. Note that under action $a_{1,1}$, s_1 is recurrent and it reward is negative. This model would also satisfy the hypotheses of an SSP model since the policy using this action would be improper.

Theorem 1.7. Suppose in a positive model the sequence $\{\mathbf{v}^n\}$ is generated by value iteration with $\mathbf{v}^0 = 0$, then $\mathbf{v}^n$ convergences monotonically and in norm to $\mathbf{v}^*$.

Proof. Since $\mathbf{v}^1 = L\mathbf{0} \geq \mathbf{0} = \mathbf{v}^0$, it follows from part 2. of Proposition 1.2 that $\mathbf{v}^n$ is monotonically non-decreasing. Thus since $\mathbf{v}^n$ is the return of a policy, condition 1. in the definition of positive models implies that it is bounded above. Hence by a standard result in analysis[19] $v^n(s)$ converges for each $s \in S$. Call the limit $v'(s)$. Since S is finite, $\|\mathbf{v}^n - \mathbf{v}'\| \to 0$.

Since $\mathbf{v}^n$ is monotonically non-decreasing $L\mathbf{v}' \geq \mathbf{v}^n$ for all $n \geq 0$ so that $L\mathbf{v}' \geq \mathbf{v}'$. On the other hand, for any $d \in D^{MD}$

$$L_d\mathbf{v}^n \leq L\mathbf{v}^n = \mathbf{v}^{n+1} \leq \mathbf{v}'.$$

Since $L_d\mathbf{v}$ is linear in $\mathbf{v}$, $L_d\mathbf{v}^n$ converges to $L_d\mathbf{v}'$ so that $L_d\mathbf{v}' \leq \mathbf{v}'$. Hence

$$L\mathbf{v}' = \operatorname*{c\text{-}max}_{d \in D^{MD}} L_d\mathbf{v}' \leq \mathbf{v}'.$$

Therefore, $L\mathbf{v}' = \mathbf{v}'$.

Since for every $n \geq 0, \mathbf{v}^n$ is the value of a policy, it follows from Theorem 1.1 that $\mathbf{v}' = \mathbf{v}^*$. □

Negative models

In the grid-world example above, Instance 2 was a negative model in which we observed that value iteration generated a convergent monotonically non-increasing sequence of iterates. We now establish that result in greater generality.

We begin with a technical result that summarizes useful properties of the Bellman operator L in negative models. To do so, we introduce the set of non-positive values, $V^- := \{\mathbf{v} : S \to \Re : \mathbf{v} \leq \mathbf{0}\}$.

[19] Theorem 3.14 in Rudin [1964].

Proposition 1.5. In a negative model:

1. For all $d \in D^{MD}$, $0 \geq \mathbf{v}^{d^\infty}$,
2. $L\mathbf{0} \leq \mathbf{0}$,
3. $\mathbf{v}^* \leq \mathbf{0}$,
4. $L : V^- \to V^-$, and
5. for any $d \in D^{MD}$ satisfying $\mathbf{v}^{d^\infty}(s) > -\infty$ for a $s \in S$, $r_d(s) = 0$ for all s in each recurrent set of $\mathbf{P}_d$.

Proof. Condition 1. in the definition of a negative model implies that $r(s,a) \leq 0$ for $a \in A_s, s \in S$. From this observation, results 1. - 4. follow immediately. Part 5. follows by noting that if $r_d(s') < 0$ for some s' in a recurrent set of $\mathbf{P}_d$, then $v_d(s) = -\infty$ for all s in a recurrent set of P_d which would be a contradiction. □

In the proof of convergence of value iteration below we will need the following useful result.

Proposition 1.6. Suppose there exists a $\mathbf{v} \in V^-$ for which $L\mathbf{v} \geq \mathbf{v}$. Then $\mathbf{v} \leq \mathbf{v}^*$.

Proof. Since $L\mathbf{v} \leq \mathbf{v}$, there exists a $d_1 \in D^{MD}$ for which:

$$\mathbf{r}_{d_1} + \mathbf{P}_{d_1}\mathbf{v} \geq \mathbf{v}.$$

But since $v \in V^-$, $\mathbf{P}_{d_1}\mathbf{v} \leq \mathbf{0}$ so that $\mathbf{r}_{d_1} \geq \mathbf{v}$. Applying L to both sides of this inequality implies that there exists a $d_2 \in D^{MD}$ for which

$$\mathbf{r}_{d_2} + \mathbf{P}_{d_2}\mathbf{r}_{d_1} = L\mathbf{r}_{d_1} \geq L\mathbf{v} \geq \mathbf{v}.$$

Applying this argument repeatedly we conclude that there exists a a policy $\pi = (d_1, d_2, \ldots) \in \Pi^{MD}$ for which $\mathbf{v}^\pi \geq \mathbf{v}$. Hence $\mathbf{v}^* \geq \mathbf{v}$. □

We now establish convergence of value iteration. The proof is a bit more complicated than that for positive models.

Theorem 1.8. Suppose in a negative model the sequence $\{\mathbf{v}^n\}$ is generated by value iteration with $\mathbf{v}^0 = 0$, then $\mathbf{v}^n$ convergences monotonically and in norm to $\mathbf{v}^*$.

Proof. Since $\mathbf{v}^1 = L\mathbf{0} \leq \mathbf{0} = \mathbf{v}^0$, it follows from part 2. of Proposition 1.2 that $\mathbf{v}^n$ is monotonically non-increasing. Since $\mathbf{v}^* \leq \mathbf{0}$ and satisfies the optimality equation, it follows from the monotonicity of L that for any $n \geq 1$

$$\mathbf{v}^* = L^n\mathbf{v}^* \leq L^n\mathbf{0} = \mathbf{v}^{n+1}.$$

Hence by the same argument in the positive case, $\mathbf{v}^n$ converges to a limit $\mathbf{v}'$ which satisfies $L\mathbf{v}' \leq \mathbf{v}'$ and $\mathbf{v}' \geq \mathbf{v}^*$. Since S is finite, $\|\mathbf{v}' - \mathbf{v}^n\| \to 0$.

Next we establish the reverse result. Since

$$\begin{aligned}
\mathbf{v}' = \lim_{n\to\infty} \mathbf{v}^n = \lim_{n\to\infty} L\mathbf{v}^n &= \lim_{n\to\infty} \text{c-max}_{d\in D^{MD}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}^n\} \\
&= \lim_{n\to\infty} \text{c-max}_{d\in D^{MD}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}' + \mathbf{P}_d(\mathbf{v}^n - \mathbf{v}')\} \\
&\leq \lim_{n\to\infty} \text{c-max}_{d\in D^{MD}}\{\mathbf{r}_d + \mathbf{P}_d\mathbf{v}' + \|\mathbf{v}^n - \mathbf{v}'\|\mathbf{e}\} \\
&= L\mathbf{v}' + \epsilon\mathbf{e}
\end{aligned}$$

where the last equality follows by noting that for any $\epsilon > 0$ and n sufficiently large $\|\mathbf{v}' - \mathbf{v}^n\| \leq \epsilon$. Since ϵ was arbitrary, $\mathbf{v}' \leq L\mathbf{v}'$ so that it follows from Proposition 1.6, that $\mathbf{v}' \leq \mathbf{v}*$.

Thus combining the two parts of the proof, we conclude that $\mathbf{v}' = \mathbf{v}^*$. □

1.5 Policy Iteration

We now discuss convergence of policy iteration in expected total reward models. We restrict our analysis to transient and SSP models so that we base the algorithm on solving the transient Bellman equation. We state the algorithm in vector form as follows.

Policy iteration for a transient model: vector notation

1. Choose $d \in D^{MD}$.
2. Until $d' = d$:
   (a) (**Evaluation**) Find $\mathbf{v}'$ by solving
   $$\mathbf{v}'_T = (\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}'_T. \tag{1.42}$$
   (b) (**Improvement**) Select
   $$d' \in \arg\operatorname*{c\text{-}max}_{d'' \in D^{MD}} \{(\mathbf{r}_{d''})_T + \mathbf{Q}_{d''} \mathbf{v}'_T\} \tag{1.43}$$
   setting $d' = d$ if possible.

We emphasize that the improvement step is executed component-wise as follows. For each $s \in S$ choose

$$a'_s \in \arg\max_{a \in A_s} \{r(s,a) + \sum_{j \in S} p(j|s,a) v'(j)\} \tag{1.44}$$

setting $a'_s = d(s)$ if possible.

The conditions "setting $d' = d$ if possible" or "setting $a'_s = d(s)$ if possible" are referred to as *anti-cycling rules*. They apply to the case when there is more than one decision rule or action that attains the max in the improvement step. The inclusion of this stipulation ensures that the stopping criterion is eventually satisfied.

Moreover, the evaluation step can be implemented by the applying

$$\begin{aligned} \mathbf{v}'_T &\leftarrow (\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}_T \\ \mathbf{v}_T &\leftarrow \mathbf{v}'_T \end{aligned}$$

to a high degree of precision. By the argument in the previous section, this iterative scheme converges to $\mathbf{v}^{d^\infty}$ in a transient model.

1.5.1 Transient models

We now analyze policy iteration in transient models. our development parallels that in the discounted case (Section **??**).

We now show that the iterates of policy iteration have a nice "Newton" representation[20].

[20]To be consistent with Chapter **??** we would modify (1.45) by changing the sign in the inverse to obtain $\mathbf{v}'_T = \mathbf{v}_T - (\mathbf{Q}_{d_v} - \mathbf{I})^{-1} B_T \mathbf{v}_T$.

Proposition 1.7. Let $\mathbf{v}_T$ and $\mathbf{v}'_T$ be successive iterates of policy iteration (restricted to the set of transient states). Then

$$\mathbf{v}'_T = \mathbf{v}_T + (\mathbf{I} - \mathbf{Q}_{d_v})^{-1} B_T \mathbf{v}_T \tag{1.45}$$

where $d_v \in \arg\,\text{c-max}_{d' \in D^{MD}} \{(\mathbf{r}_{d'})_T + \mathbf{Q}_{d'} \mathbf{v}_T\}$.

Proof. Since

$$\begin{aligned}
\mathbf{v}'_T &= (\mathbf{I} - \mathbf{Q}_{d_v})^{-1} (\mathbf{r}_{d_v})_T = (\mathbf{I} - \mathbf{Q}_{d_v})^{-1} (\mathbf{r}_{d_v})_T - \mathbf{v}_T + \mathbf{v}_T \\
&= (\mathbf{I} - \mathbf{Q}_{d_v})^{-1} \Big[(\mathbf{r}_{d_v})_T + (\mathbf{Q}_{d_v} - \mathbf{I}) \mathbf{v}_T \Big] + \mathbf{v}_T \\
&= \mathbf{v}_T + (\mathbf{I} - \mathbf{Q}_{d_v})^{-1} B_T \mathbf{v}_T,
\end{aligned}$$

the result follows from the last equality by changing signs and rearranging terms. □

This result leads to an easy proof of the convergence of policy iteration in transient models.

Theorem 1.9. In a transient model, policy iteration converges monotonically and in a finite number of iterations to the optimal value function $\mathbf{v}^*_T$ and an optimal stationary policy.

Proof. By construction

$$\mathbf{v}'_T = (\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}'_T \leq \underset{d'' \in D^{MD}}{\text{c-max}} \{(\mathbf{r}_{d''})_T + \mathbf{Q}_{d''} \mathbf{v}'_T\} = L_T \mathbf{v}'_T$$

with strict inequality whenever $d' \neq d$ where d' satisfies (1.43). Therefore either $B_T \mathbf{v} = L_T \mathbf{v} - \mathbf{v} = \mathbf{0}$ or $B_T \mathbf{v}(s) > 0$ for some $s \in S - \Delta$. In the former case $\mathbf{v}$ solves the transient Bellman equation and in the latter case it follows immediately from (1.45) and the observation that $(\mathbf{I} - \mathbf{Q}_d)^{-1} \geq \mathbf{I}$ that $v'_T(s) > v_T(s)$ for some $s \in S - \Delta$. Since there are only finitely many policies, the stopping criterion must be satisfied for some finite N at which point $\mathbf{v}_T$ satisfies the Bellman equation. From Corollary (1.3), d^∞ is an optimal stationary policy. □

Example 1.1 revisited: We now solve Instance 1 of the grid world model using policy iteration. We choose the initial decision rule as "go up" when possible otherwise "go right" in cell 2 and " go left" in cell 3[a] To simplify coding we implemented the evaluation step iteratively.
The number of iterations (improvement steps) required for convergence varied with p as shown in Table 1.2. **(ideally table would go in textbox and paragraphs would be indented)**
Note that when $p = 0$ or $p = 1$ the stationary policy corresponding to the initial decision rule was improper because it cycled between cells 2 and 3 forming a closed class consisting of cells 2 and 3. Since the cost associated with these states is negative infinity the model can be classified as an SSP model. Hence, as noted below, in such a case, we need to start the algorithm with a proper policy to ensure convergence.

[a]To simplify coding one could allow all actions in all cells but set $q(s, a)$ to a large negative value when action a was impossible in state s, Therefore we could have started policy iteration with the decision rule "go up" in all states.

p	0.01	0.1	0.5	0.9	0.99
iterations	5	4	3	3	6

Table 1.2: Number of iterations for policy iteration to converge in grid world model as a function of p.

1.5.2 Stochastic shortest path models

The above result extends directly to SSP models in which policy iteration starts with a proper policy. The argument used to prove Theorem 1.9 ensures that the an improper policy cannot be identified at a subsequent iterate of policy iteration. Hence we have:

Corollary 1.5. In a stochastic shortest path model, suppose policy iteration begins with a *proper* policy. Then it converges monotonically to $\mathbf{v}_T^*$ in a finite number of iterations to the optimal value function and an optimal stationary policy.

Numerical examples such as that in Example 1.5 below, show that when we start with an improper policy, d, (1.42) will not have a solution.

1.6 Modified Policy Iteration

Since we saw that transient models behave like discounted models, we can also consider the modified policy iteration (MPI) algorithm.

Modified policy iteration in a transient model: vector notation

1. **Initialize:** Specify a decision rule $d' \in D^{\mathrm{MD}}$, $\mathbf{v}_T = \mathbf{0}$, a sequence of non-negative integers $\{m_n \; n = 0, 1, \ldots$ and a stopping criterion $\epsilon > 0$. Set $n = 0$.

2. **Check stopping criterion:** Until $\|\mathbf{v}_T - \mathbf{v}'_T\| < \epsilon$:

   (a) **Evaluate:**

      i. $k \leftarrow 0$ and $\mathbf{u}_T = \mathbf{v}_T$.

      ii. Until $k = m_n$.

         A. $\mathbf{u}_T \leftarrow (\mathbf{r}_{d'})_T + \mathbf{Q}_{d'} d\mathbf{u}_T$

         B. $k \leftarrow k + 1$

      iii. $\mathbf{v}'_T \leftarrow \mathbf{u}_T$.

   (b) **Improve:** Choose

$$d' \in \arg\operatorname*{c\text{-}max}_{d \in D^{\mathrm{MD}}} \{(\mathbf{r}_d)_T + \mathbf{Q}_d \mathbf{v}'_T\}. \tag{1.46}$$

   (c) **Iterate:** $n \leftarrow n + 1$ and $\mathbf{v}_T \leftarrow \mathbf{v}'_T$.

3. **Return value and decision rule:** Set $(\mathbf{v}_\epsilon)_T = \mathbf{v}'_T$ and $d_\epsilon = d'$.

Some comments of about the algorithm follow:

1. We could instead start the algorithm from a value $\mathbf{v}_T^0$ (or equivalently set $m_n = 0$) and proceed directly to the improvement step.

2. The evaluation step begins with the previous estimate of the value $\mathbf{v}_T$ except at the first iteration which starts the evaluation at $\mathbf{v}_T = \mathbf{0}$. As a result of starting with $\mathbf{v}_T = \mathbf{0}$, all iterates of MPI correspond to the value of some policy over a finite horizon. In contrast, policy iteration with iterative evaluation starts with each evaluation with $\mathbf{u}_T = \mathbf{0}$. In a discounted model, the discount factor damps out the effect of earlier estimates while in a transient model, contraction properties plays this role.

3. The index n specifies the order m_n.

4. Setting $m_n = 0$ for all n is identical to value iteration.

5. The stopping criterion is value based while that of policy iteration is decision rule based.

Note that the iterates of modified policy iteration can be written as:

$$\mathbf{v}'_T \leftarrow L_{d'}^{m_n+1} \mathbf{v}_T \tag{1.47}$$

Consequently it follows that

$$\mathbf{v}_T^n = L_{d_n}^{m_n+1} \dots L_{d_0}^{m_0+1} \mathbf{0} \tag{1.48}$$

where $\mathbf{v}_T^n$ represents the value obtained after the nth evaluation and d_n denotes the decision rule obtained after the $(n-1)$th evaluation.

Convergence

We outline a proof of convergence of MPI from the discounted case under the assumption that $L\mathbf{v}_T^0 \geq \mathbf{v}_T^0$.

Theorem 1.10. Suppose

$$L\mathbf{v}_T^0 \geq \mathbf{v}_T^0$$

where $\mathbf{v}_T^0$ denotes the value obtained after the first evaluation step[a]. Then the iterates of modified policy iteration converge monotonically and in norm to $\mathbf{v}_T^*$.

[a] Or instead MPI is started in the improvement step with $\mathbf{v}_T^0$ satisfying the above condition.

Proof. Denote the iterates of MPI as $\mathbf{v}_T^n$ and let $\mathbf{u}_T^n$ denote the iterates of value iteration starting with $\mathbf{u}_T^0 = \mathbf{v}_T^0$ for $n = 0, 1, \dots$ Then we can show inductively that

$$\mathbf{u}_T^n \leq \mathbf{v}_T^n \leq \mathbf{v}_T^*.$$

Since $\mathbf{u}_T^n$ converges monotonically to the optimal value function $\mathbf{v}_T^*$ when $L\mathbf{u}_T^0 \geq \mathbf{u}_T^0$, the above inequalities imply the result. □

An example

We now use modified policy iteration in the grid world model to find an ϵ optimal policy .

Example 1.1 revisited:
We focus on Instance 1 with $p = 0.1$ and $\epsilon = 0.0001$. We choose p small so that value iteration converges slowly and we can distinguish different configurations for . We initiate the algorithm with the same policy used to start policy iteration in the previous section. We consider m_n, constant, increasing and decreasing. Table 1.3 reports computational results. In this example each maximization requires evaluating 28 state action pairs while each evaluation step required updating 13 values so that a maximization step is "roughly" twice as costly as an evaluation step. The results below suggest that a moderate fixed value of m_n or an increasing sequence is preferable.

m_n	Improvement steps (maximizations)	Evaluation equivalents	Iteration at which optimal policy first appears
0	251	502	31
2	88	352	7
10	21	252	2
20	13	286	2
50	7	364	1
n^2	10	405	3
$\max(40-n, 0)$	9	666	4

Table 1.3: Grid world modified policy iteration computational results for various specifications for m_n. "Evaluation equivalents" equals the sum of the number of evaluations plus 2 times the number of improvements.

MPI in SSP models

In an SSP model, MPI converges when

1. initiated with a proper policy, or
2. it begins in the improvement step, or
3. when $m_0 < \infty$.

In the latter case, $\mathbf{v}'_T$, will be finite so at the subsequent improvement step, a proper policy will be identified.

1.7 Linear Programming in transient models

We now discuss the formulation and properties of linear programs in the expected total reward model.

1.7.1 The primal linear program

If $v(s)$ satisfies the transient Bellman equation

$$v(s) = \max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S-\Delta} p(j|s,a) v(j) \right\}$$

for all $s \in S - \Delta$, then it satisfies

$$v(s) \geq \; r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v(j) \tag{1.49}$$

for all $a \in A_s$ and $s \in S - \Delta$.

By a generalization of Corollary **??** in the case of discounted models, any $v(s)$ satisfying (**??**) for all $a \in A_s$ and $s \in S$ is an upper bound on $v^*(s)$. Hence a solution of the Bellman equation may be thought of as the *smallest* (in a component-wise sense) $v(s)$ that satisfies (1.49) for all $a \in A_s$ and $s \in S - \Delta$. To find such a $v(s)$, we choose $\alpha(s) > 0$ for all $s \in S_\Delta$ and solve the following linear program.

Primal LP formulation of a transient model

$$\text{minimize} \quad \sum_{s \in S-\Delta} \alpha(s) v(s) \tag{1.50a}$$

$$\text{subject to} \quad v(s) - \sum_{j \in S} p(j|s,a) v(j) \geq r(s,a), \quad a \in A_s,\ s \in S - \Delta \tag{1.50b}$$

Because the above system has many more rows than columns, we prefer to analyze the model through its dual linear program in which dual variables have an interesting interpretation.

1.7.2 The dual linear program and its properties

The corresponding dual linear program may be written as:

Dual LP formulation of a transient model

$$\text{maximize} \quad \sum_{s \in S-\Delta} \sum_{a \in A_s} r(s,a) x(s,a) \tag{1.51a}$$

$$\text{subject to} \quad \sum_{a \in A_s} x(s,a) - \sum_{j \in S-\Delta} \sum_{a \in A_j} p(s|j,a) x(j,a) = \alpha(s), \quad s \in S - \Delta \tag{1.51b}$$

$$x(s,a) \geq 0, \quad a \in A_s, s \in S - \Delta \tag{1.51c}$$

Using the same approach as in the discounted model, the theorem below shows that there is a one-to-one relationship between:

1. feasible solutions to the dual problem and randomized stationary policies
2. *basic* feasible solutions to the dual problem and deterministic stationary policies.

As in Theorem **??** in Chapter 5, we have that

Theorem 1.11.

a. For each $d \in D^{MR}$

$$x_d(s,a) := \sum_{j \in S-\Delta} \alpha(j) \sum_{n=1}^{\infty} P^{d^\infty}(X_n = s, Y_n = a \,|\, X_1 = j) \tag{1.52}$$

is a feasible solution to the dual linear program, that is it satisfies the constraints (1.51b) and (1.51c).

b. Suppose for each $s \in S-\Delta$, $\alpha(s) > 0$ and let $x(s,a)$ denote a feasible solution to the dual linear program. Let d_x denote the randomized decision rule which selects action $a \in A_s$ in state $s \in S - \Delta$ with probability

$$w_{d_x}(s,a) : \frac{x(s,a)}{\sum_{a \in A_s} x(s,a)}. \tag{1.53}$$

Then $x_{d_x}(s,a)$ defined by (1.52) is a feasible solution to the dual linear program and $x_{d_x}(s,a) = x(s,a)$ for all $a \in A_s$, $s \in S - \Delta$.

Noting that there are $|S-1|$ dual constraints[21] and assuming $\alpha(s) > 0$, for each $s \in S - \Delta$ a basic feasible solution has $x(s,a') > 0$ for exactly one $a' \in A_s$. Hence we have:

Corollary 1.6. Suppose $\alpha(s) > 0$ for all $s \in S_\Delta$. Then

a. For each $d \in D^{MD}$ and $s \in S - \Delta$

$$x_d(s,a) := \begin{cases} \sum_{j \in S-\Delta} \alpha(j) \sum_{n=1}^{\infty} P^{d^\infty}(X_n = s, Y_n = a \,|\, X_1 = j) & \text{for} \quad a = d(s) \\ 0 & \text{for} \quad a \neq d(s) \end{cases} \tag{1.54}$$

is a basic feasible solution of the dual linear program.

b. Define $d_x \in D^{MD}$ by $d_x(s) = a_s$ if $x(s,a_s) > 0$ for $s \in S - \Delta$. Then $x_{d_x}(s,a)$ is a basic feasible solution to the dual linear program and $x_{d_x}(s,a) = x(s,a)$ for all $a \in A_s$, $s \in S - \Delta$.

Finally we have the following result.

Corollary 1.7. There exists an optimal basic feasible solution $x^*(s,a)$ to the dual linear program and an optimal stationary deterministic policy $d^\infty_{x^*}$ where $d_{x^*}(s) = a^*_s$ if $x^*(s,a^*_s) > 0$.

[21] Not including the non-negativity constraints,

We observe that when $\sum_{s\in S-\Delta}\alpha(s)=1$, the expression on the right hand side of (1.52) represents the expected number of times (under policy d^∞) that the system with initial state chosen with probability $\alpha(s)$ visits state s and chooses action a[22] Hence the quantity

$$\sum_{s\in S-\Delta}\sum_{a\in A_s} r(s,a)x_d(s,a)=\sum_{s\in s-\Delta}\alpha(s)v^{d^\infty}(s).$$

This means that when $\sum_{s\in S-\Delta}\alpha(s)=1$, the objective function value when $x(s,a)=x_d(s,a)$ is the expected total reward corresponding to policy d^∞ averaged with respect to initial distribution $\alpha(s)$.

1.7.3 Examples

We consider two examples.

Example 1.4. A very simple transient example.
Suppose $S=\{s',\Delta\}$ and $A_{s'}=\{a_1,a_2\}$ with $r(s',a_1)=5$ and $p(s'|s',a_1)=0.2$ and $r(s',a_2)=3$ and $p(s'|s',a_2)=0.5$.

Setting $\alpha(s')=1$ yields the primal linear program:

$$\begin{aligned} \text{minimize} \quad & v(s') \\ \text{subject to} \quad & v(s')-0.2v(s')\geq 5, \\ & v(s')-0.5v(s')\geq 3. \end{aligned}$$

Rewriting the constraints as $v(s')\geq 6.25$ and $v(s')\geq 6$. Therefore the solution $v*(s')=6.25$ and action $d^*(s')=a_1$ is optimal since equality holds in the first constraint.

Next we formulate the dual as

$$\begin{aligned} \text{maximize} \quad & 5x(s',a_1)+3x(s',a_2) \\ \text{subject to} \quad & x(s',a_1)+x(s',a_2)-0.2x(s,a_1)-0.5x(s',a_2)=1 \\ & x(s',a_1)\geq 0 \text{ and } x(s',a_2)\geq 0. \end{aligned}$$

The constraint can be written as $0.8x(s',a_1)+0.5x(s',a_2)=1$. There are two basic feasible solutions; $x_1(s',a_1)=1.25$, $x_1(s',a_2)=0$ and $x_2(s',a_1)=0$, $x_2(s',a_2)=2$ and clearly the first is optimal with an objective function value of 6.25 in agreement with the solution of the primal linear program. Hence the deterministic stationary

[22]To see this note that

$$\sum_{n=1}^{\infty}P^{d^\infty}(X_n=s,Y_n=a\,|\,X_1=j)=E\{\sum_{n=1}^{\infty}I_{\{X_n=s,Y_n=a\}}\,|\,X_1=j\}.$$

policy derived from $d^*(s') = a_1$ is optimal and we see that under this policy, on average, the system spends 1.25 decision epochs in state s' prior to absorption in Δ.

Example 1.5. An SSP model
Consider the model depicted in Figure 1.2 with $S = \{s_1, s_2, \Delta\}$, $A_{s_1} = \{a_{1,1}, a_{1,2}\}$ and $A_{s_2} = \{a_{2,1}, a_{2,2}\}$ with rewards $r(s_1, a_{1,1}) = -3$, $r(s_1, a_{1,2}) = 1$, $r(s_2, a_{2,1}) = 1$ and $r(s_2, a_{2,2}) = -2$ and non-zero transition probabilities $p(s_1|s_1, a_{1,1}) = p(s_2|s_1, a_{1,1}) = 0.5$, $p(s_2|s_1, a_{1,2}) = 1$, $p(\Delta|s_2, a_{2,1}) = 1$ and $p(s_1|s_2, a_{2,2}) = 1$. Then the model is not transient since the stationary policy based on the decision rule $d'(s_1) = a_{1,2}, d'(s_2) = a_{2,2}$ is improper with $v^{(d')^\infty}(s_1) = v^{(d')^\infty}(s_1) = -\infty$. Note also that the policy based on $d''(s_1) = a_{1,1}$ and $d''(s_2) = a_{2,2,}$ is also improper. Hence the model satisfies the hypotheses of a stochastic shortest path model.

Taking $\alpha(s_1) = \alpha(s_2) = 0.5$, the primal linear program becomes

$$\begin{aligned} \text{minimize} \quad & 0.5v(s_1) + 0.5v(s_2) \\ \text{subject to} \quad & v(s_1) - 0.5v(s_1) - 0.5v(s_2) \geq -3, \\ & v(s_1) - v(s_2) \geq 1, \\ & v(s_2) \geq 1, \\ & -v(s_1) + v(s_2) \geq -2. \end{aligned}$$

The corresponding dual linear program is

$$\begin{aligned} \text{maximize} \quad & -3x(s_1, a_{1,1}) + x(s_1, a_{1,2}) + x(s_2, a_{2,1}) - 2x(s_2, a_{2,2}) \\ \text{subject to} \quad & x(s_1, a_{1,1}) + x(s_1, a_{1,2}) - 0.5x(s_1, a_{1,1}) - x(s_2, a_{2,2}) = 0.5 \\ & x(s_2, a_{2,1}) + x(s_2, a_{2,2}) - 0.5x(s_1, a_{1,1}) - x(s_1, a_{1,2}) = 0.5 \\ \text{and} \quad & x(s, a) \geq 0 \text{ for } a \in A_s, s \in S - \Delta. \end{aligned}$$

Solving the dual, we find that $x(s_1, a_{1,2}) = 0.5$, $x(s_2, a_{2,1}) = 1$ and the remaining $x(s, a) = 0$. Hence the optimal policy d^* uses decision rule $d^*(s_1) = a_{1,2}$ and $d^*(s_2) = a_{2,1}$.

We noted that the same policy was optimal for all values of $r(s_2, a_{1,1})$. On theother hand, when either $r(s_2, a_{2,2}) > -1$, $r(s_1, a_{1,2}) > 2$ or $r(s_1, a_{1,1}) > 1$, the primal linear program was infeasible[a]. In all of these cases the condition that value of an improper policy equal $-\infty$ were violated.

[a] You can easily confirm this by plotting the feasible region of the primal.

We leave solving gridworld using linear programming as an exercise.

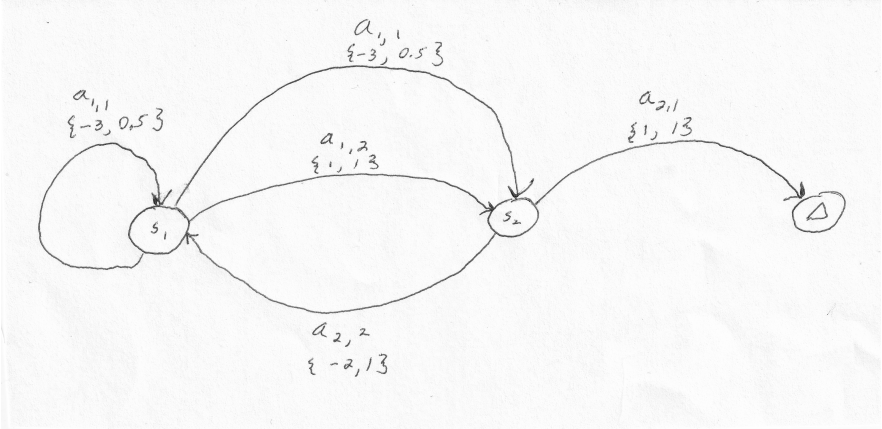

Figure 1.2: Symbolic representation of model in Example 1.5

1.8 Optimality of structured policies

(do we need this.)

In this section we extend results in Section **??** to expected total reward models in which value iteration is convergent.

1.8.1 The fundamental result

Define V^F to be a subset of V with a specific form (such as convex or monotone) and define D^F to be a subset D^{MD} in which all decision rules have a particular structure. We say that D^F and V^F are *compatible* if $\mathbf{v} \in V^F$ implies that there exists a

$$d' \in \arg\operatorname*{c\text{-}max}_{d \in D^{\text{MD}}} \{\mathbf{r}_d + \mathbf{P}_d \mathbf{v}\}$$

that is in D^F. For example in an optimal stopping problem, V^F might denote the set of non-decreasing functions and D^F might denote decision rules which stop after a threshold is reached.

Theorem 1.12. Suppose:

1. V^F is non-empty,
2. D^F is non-empty and compatible with V^F,
3. $\mathbf{v} \in V^F$ implies $L\mathbf{v} \in V^F$, and
4. if $\lim_{n\to\infty} \|\mathbf{v}^n - \mathbf{v}^*\| = 0$ and $\mathbf{v}^n \in V^F$ for all $n = 0, 1, \ldots$ then $\mathbf{v}^* \in V^F$.

Then there exists an optimal stationary policy $(d^*)^\infty$ with $d^* \in D^F$.

The proof is identical to that of Theorem **??** and we omit it. Note that for hypothesis 4. to apply requires at a minimum the convergence of value iteration which holds in all the models considered previously in this section.

1.9 Applications

1.9.1 Grid world navigation

In the computational examples above we focused on algorithmic convergence. Now we turn on our attention to the properties and structure of solutions and how they are impacted by model parameters. We consider only Instance 1 with a reward of 50 units when successfully delivering the coffee, a cost of 200 units if the robot falls down the stairs and a cost per step of 1 unit. Recall that the only available actions are "up", "left" or "right", with not every action available in each cell.

When $p = 1$ this is a deterministic shortest path problem. We find the optimal policy starting in cell 13 is

$$13 \to 10 \text{ or } 14 \to 11 \to 8 \to 5 \to 2 \to 1.$$

Both policies yield $v^*(13) = 50 - 5 = 44$. Moreover, solving this problem also provides the shortest path from each cell not on the shortest path to the office.

When $p = 0.95$ starting in cell 13, the optimal path becomes

$$13 \to 14 \to 15 \to 12 \to 9 \to 6 \to 3 \to 2 \to 1.$$

For such a policy, the expected value starting in cell 13 is $v^*(13) = 40.96$. Observe that this policy is conservative in that it aims to take a longer path so as to avoid the costly cell 7. Moreover if the robot enters cell 11, the optimal policy moves it to cell 12 for the same reason.

Note that when $p = 0.65$, the optimal policy remains the same as when $p = 0.95$ but $v^*(13) = -4.63$ implying that even under the optimal policy there is a non-negligible probability of falling down the stairs.

For small values of p such as $p = 0.2$, the robot's movement becomes unreliable and the optimal policy becomes quite strange. In such cases the robot must aim in the "wrong" direction so as to actually head in the right direction. The optimal policy when $p = 0.2$ appears in Table 1.4:

Observe that in cell 8, the optimal policy aims the robot in the counter-intuitive direction of cell 7. This is because the probability it moves in some other direction is 0.8. Note that $v^*(13) = -150.26$ corresponding to the high probability of falling down the stairs when the robot is unreliable.

Interesting variants to consider include models with cell-dependent movement probabilities or a tilted grid where it is more likely to move in one direction than the other.

state	intended cell
2	3
3	2
4	5
5	6
6	3
8	7
9	8
10	11
11	10
12	11
13	10
14	13
15	14

Table 1.4: State and optimal direction when $p = 0.2$.

1.9.2 Optimal stopping

We use the above results to analyze a stationary version of the optimal stopping problem (Section **??** in which the system moves probabilistically between states in S' until the decision maker decides to stop. Thus $S = S' \cup \{\Delta\}$ where Δ denotes the stopped state. For each state in S' the decision maker can either continue (action C) or stop (action Q). Therefore the action sets may be written as

$$A_s = \begin{cases} \{C, Q\} & s \in S' \\ \{\delta\} & s = \Delta. \end{cases}$$

For simplicity we assume a fixed continuation cost c and a reward $g(s)$ for stopping in state $s \in S'$. Hence the rewards are given by

$$r(s,a) = \begin{cases} -c & s \in S', a = C \\ g(s) & s \in S', a = Q \\ 0 & s = \Delta, a = \delta. \end{cases}$$

Note that the reward $g(s)$ is only received upon stopping at which time the system moves to the zero reward absorbing state Δ.

We assume an underlying $|S'| \times |S'|$ transition probability matrix $\mathbf{B}$ with components $b(j|s)$ so that the transition probabilities for the corresponding Markov decision process can be expressed as

$$p(j|s,a) = \begin{cases} b(j|s) & s \in S', a = C, j \in S' \\ 1 & s \in S', a = Q, j = \Delta \text{ or } s = j = \Delta, a = C \\ 0 & \text{otherwise.} \end{cases}$$

First observe that the when $g(s) \geq 0$, this model satisfies the hypotheses of a positive model. Observe also that stationary policy $(d')^\infty$ where $d'(s) = C$ for all $s \in S'$ is has $v^{(d')^\infty}(s) = -\infty$ for all $s \in S'$ so if $g(s) \leq 0$ for all $s \in S'$, it is a negative model. The model is not transient since under $(d')^\infty$ the state Δ cannot be reached. Moreover if S' is a closed class under $\mathbf{B}$, it is an SSP model where the policy $(d')^\infty$ is improper but satisfies condition 3. in the definition of an SSP model.

We express the Bellman equations in component notation as:

$$v(s) = \max\{-c + \sum_{j \in S'} b(j|s)v(j), g(s) + v(\Delta)\} \quad s \in S' \quad \text{and} \quad v(\Delta) = v(\Delta). \tag{1.2}$$

Since $v(\Delta) = 0$ they can be rewritten as:

$$v(s) = \max\{-c + \sum_{j \in S'} b(j|s)v(j), g(s)\} \tag{1.3}$$

which is equivalent to the transient Bellman equations defined above.

They can be solved using any iterative algorithm or the (primal) linear program:

$$\text{minimize} \sum_{s \in S'} v(s) \tag{1.4}$$

subject to

$$v(s) - \sum_{j \in S'} b(j|s)v(j) \geq -c \tag{1.5}$$

$$v(s) \geq g(s) \tag{1.6}$$

Let $\mathbf{v}^*$ denote the optimal value function. Since an optimal policy must stop in some state[23] it follows from Theorem 1.3 that the policy d^* defined below is optimal.

$$d^*(s) = \begin{cases} Q & \text{if} \quad v^*(s) = g(s) \\ C & \text{if} \quad v^*(s) > g(s). \end{cases}$$

Optimal stopping on a random walk

We apply this to optimal stopping in a finite random walk on the integers with reflecting barriers at 1 and N.

Example 1.6. Optimal stopping in a random walk.

In this example, $S = \{1, \ldots, N\}$, c denotes the continuation cost, $g(s) = \alpha s^2$ and p denotes the probability of a transition from s to $s+1$ except in state N where p denotes the probability of remaining in state N. Setting $q = 1 - p$, we let it denote the probability of a transition from state s to $s-1$ except in state

[23] Consequently $\mathbf{P}_{d^*}^N$ must converge to a matrix which has entries equal to zero in the first $|S' - \Delta|$ columns and ones in the column corresponding to Δ. Since $v^*(\Delta) = 0$, (1.31) holds.

1 where it denotes the probability of staying there. Thus the (transient) Bellman equations become

$$\begin{aligned} v(1) &= \max\{-c + qv(1) + pv(2),\ g(1)\}, \\ v(s) &= \max\{-c + qv(s-1) + pv(s+1),\ g(s)\}, \quad \text{for} \quad s = 2, \ldots, N-1 \\ v(N) &= \max\{-c + qv(N-1) + pv(N),\ g(N)\}. \end{aligned}$$

The decision maker trades off the cost of continuing versus the cost of reaching the high reward states. When $\alpha > 0$ these are states with large values of s and when $\alpha < 0$ they correspond to small values of s.

Since this is an SSP model, we can solve it numerically using value iteration[a] and a "sup norm" stopping criterion with $\epsilon = 0.000001$. For simplicity we choose $v^0(s) = 0$.

We consider the following instances:

Instance 1: $N = 25$, $\alpha = 0.2$, $p = 0.65$ and $c = 2$,

Instance 2: $N = 25$, $\alpha = 0.2$, $p = 0.5$ and $c = 2$,

Instance 3: $N = 500$, $\alpha = 0.05$, $p = 0.65$ and $c = 10$, and

Instance 4: $N = 500$, $\alpha = -0.05$, $p = 0.65$ and $c = 10$.

Instance 5: $N = 500$, $\alpha = -0.05$, $p = 0.35$ and $c = 10$.

Note that Instances 1-3 represent reward maximization problems and Instances 4-5 represent cost minimization problems. The parameters were chosen so as to obtain "interesting" policies.

Table 1.5 provides the number of iterations to converge and the *continuation region* in which the optimal policy chooses the action C for each instance. We observe that:

1. Value iteration converged in all instances.
2. In all instances $v^*(s)$ was a monotone function of s.
3. With the exception of Instance 2 in which $v^*(s) = g(s)$ for $S = 1, \ldots, 25$, and Instance 4 where the continuation region is a singleton, the continuation region is a set of consecutive states.
4. In instance 2, value iteration converges in 2 iterations since with $v^0(s) = 0$, $v^n(s) = g(s)$ for all $n \geq 1$.
5. In Instances 1 and 3, the optimal policy stops for s less than some threshold and when $s = N$.

6. In Instance 4 when the $g(s) < 0$ for all $s \in S$, and there is a probability of 0.65 of moving to a higher cost state, it is better to stop except unless the system reaches the highest cost state.

7. In Instance 5, in which the system moves to a lower cost state with probability 0.65, it is optimal to continue in high cost states.

We hypothesize that one can prove analytically that when $g(s)$ is monotone and p is constant, that $v^*(s)$ is monotone. Hence under further conditions on $g(s)$, the on can prove that the continuation region is either empty, or a set of consecutive states. We leave this as an exercise.

[a] This is the easiest to code and converges quickly for small N.

Instance	Iterations	Continuation region
1	249	$\{8, \ldots, 24\}$
2	2	none
3	1777	$\{299, \ldots, 499\}$
4	1287	$\{500\}$
4	1677	$\{334, \ldots, 500\}$

Table 1.5: Computational results for Example 1.6.

Selling an asset*

We now consider a stationary version of the asset selling problem introduced in Section **??**. Assume you are trying to sell an asset such as a home. Offers arrive daily and you can either accept the offer or wait another day in anticipation of a better offer. If you do not accept the current offer, it is withdrawn at the end of the day.

Let the random variable X denote the offer amount. We assume offers are discrete and independent between days. Let $p(n) := p[X = n]$, n, $0 \leq n \leq N$. Furthermore we assume a cost c for holding the asset for an additional day.

We first model a related problem in which all past offers remain available in perpetuity. We refer to it as the *modified asset selling problem*.

In it $S = S' \cup \{\Delta\}$, $S' = \{0, \ldots, N\}$ and Δ denotes the stopped state. The set of actions $A_s = \{C, Q\}$ for $s \in S'$ and $A_\Delta = \{Q\}$. Transition probabilities satisfy $p(\Delta|s, Q) = 1$ for all $s \in S$ and

$$p(j|s, C) = \begin{cases} 0 & 0 \leq j < s \\ \sum_{k \leq s} p(k) & j = s \\ p(j) & s < j \leq N \end{cases}$$

and rewards satisfy

$$r(s,a) = \begin{cases} -c & s \in S', a = C \\ s & s \in S, a = Q \\ 0 & s = \Delta, a = Q \end{cases}.$$

Again this can be classified as a positive or an SSP model. The (transient) Bellman equation can be written[24] as:

$$v(s) = \max\{s, -c + v(s)\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{N} p(j)v(j)\} \tag{1.7}$$

Instead of solving this equation numerically, the following theorem provides a relationship that can be used to easily compute the optimal policy. We prove it using policy iteration.

We let μ denote the expected offer size, that is,

$$\mu = \sum_{n=1}^{N} np(n)$$

. We assume $\mu > 0$ and define

$$F(s) := \sum_{j=s+1}^{N} (j-s)p(j)$$

for $s = -1, \ldots, N$. Then it's easy to see that $F(-1) = \mu$, $F(N) = 0$ and $F(s)$ is non-increasing in s.

Theorem 1.13. Let

$$s^* = \min\{s \in S' : F(s) < c\}. \tag{1.8}$$

Then the stationary policy $(d^*)^\infty$ derived from:

$$d^*(s) = \begin{cases} C & s < s^* \\ Q & s \geq s^*. \end{cases} \tag{1.9}$$

is an optimal policy for the modified asset selling problem. Moreover if $\mu < c$, it is optimal to accept the first offer.

[24]Note typo in the Bellman equation on p.307 in Puterman [1994]. We could have avoided this typo by carefully writing out the transition probabilities.

Proof. Begin policy iteration with $d(s) = Q$ for all $s \in S'$. Then $v(s) = s$ for all $s \in S'$. The improvement step chooses a $d'(s)$ as

$$d'(s) \in \arg\max\{s, -c + s\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{N} jp(j)\}$$

$$= \arg\max\{s, -c + s + \sum_{j=s+1}^{N} (j-s)p(j)\} = \arg\max\{0, -c + \sum_{j=s+1}^{N} (j-s)p(j)\}$$

where the equality follows by adding and subtracting $s\sum_{j=s+1}^{N} p(j)$ in the second expression. Therefore

$$d'(s) = \begin{cases} Q & \text{if} \quad F(s) - c < 0 \\ \{Q, C\} & \text{if} \quad F(s) - c = 0 \\ C & \text{if} \quad F(s) - c > 0. \end{cases}$$

Let $d''(s)$ denote the decision rule chosen at the next iteration of policy iteration. When $\mu < c$, $d''(s) = Q$ for all $s \in S'$ so policy iteration terminates because $d''(s) = d'(s)$ for all $s \in S$.

Now assume $\mu \geq c$. Consequently $s^* > 0$, so that we know without computation that the value of $v'(s) := v^{(d')^\infty}(s) = s$ for $s \geq s^*$ and $v'(s) > s$ for $s < s^*$ (Why?).

Now assume $s \geq s^*$. Then

$$-c + v'(s)\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{N} v'(j)p(j) = -c + s\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{N} jp(j)$$

$$= -c + \sum_{j=s+1}^{N} (j-s)p(j) + s = -c + F(s) + s < s.$$

Therefore $d''(s) = Q = d'(s)$ for all $s \geq s^*$.

Now assume $s < s^*$. Then

$$-c + v'(s)\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{N} v'(j)p(j)$$

$$= -c + v'(s)\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{s^*} v'(j)p(j) + \sum_{j=s^*+1}^{N} v'(j)p(j)$$

$$\geq -c + s\sum_{j=0}^{s} p(j) + \sum_{j=s+1}^{s^*} jp(j) + \sum_{j=s^*+1}^{N} jp(j)$$

$$= -c + \sum_{j=s+1}^{N} (j-s)p(j) + s = -c + F(s) + s > s.$$

Therefore $d''(s) = C = d'(s)$. Hence policy iteration terminates and the stationary policy derived from d' is optimal. □

Since the above policy is admissible in the (unmodified) asset selling problem and and the value of the modified asset selling problem is greater than or equal the value of the (unmodified) asset selling problem, we obtain the following result as a corollary.

Corollary 1.8. The stationary policy derived from the decision rule $d^*(s)$ defined by (1.9) is optimal in the (unmodified) asset selling problem.

Note this result extends to continuous distributions and those on $0, 1, \ldots$ but it requires theoretical extension to continuous and countable state spaces.

Example 1.7. Suppose the offer size is generated from a truncated (at $N = 20$) Poisson with parameter 10. Figure 1.3 shows that when $c = 2$ that $s^* = 8$. Moreover, we see from this figure that if $c > 9.98$ (the mean of the truncated Poisson), it would be optimal to stop after receiving the first offer and if $c = 0$ it would be optimal to continue until we receive the maximum offer.

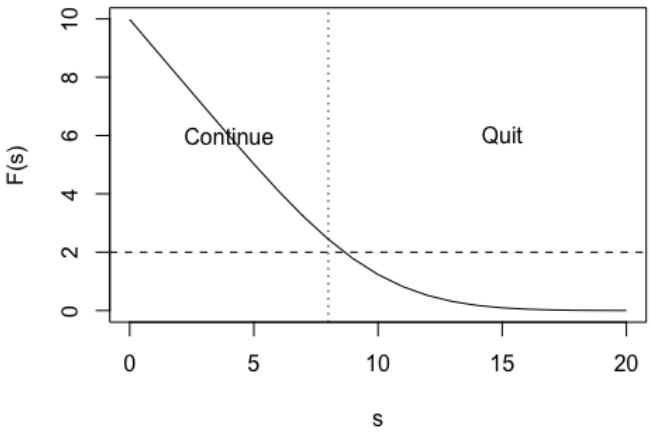

Figure 1.3: Plot of $F(s)$ vs s in asset selling problem when offers follow a truncated Poisson distribution with parameter 10. Vertical line shows $s^* = 8$.

1.9.3 Optimal parking

(This is actually a finite horizon problem which we can assign or solve in Chapter 4)

1.10 Bibliographic remarks

This chapter analyzes finite state and action expected total reward models focusing primarily on transient and SSP models which arise naturally in modern applications. On the other hand, Chapter 7 in Puterman [1994] considers countable state models and restricts attention to positive and negative models which present considerable analytical challenges.

Foundational papers include Strauch [1966]'s study of negative models, Blackwell [1967]'s study of positive models, and A.F. Veinott [1969]'s study of transient models.

Transient models were first referred to as *optimal first passage problems* and studied by Eaton and Zadeh [1962] and Derman [1970]. Bertsekas and Tsitsiklis [1991] extended them to include stochastic shortest path models and provide a rich bibliographic background.

The definition of a transient model in A.F. Veinott [1969] differs from ours in that he requires at least one row sum of $\mathbf{P}_d$ be strictly less than 1. Thus he doesn't explicitly require the existence of a single reward-free absorbing state. Wessels [1977] introduced the use of the weighted supremum norm for analyzing value iteration in expected total reward models.

We draw on Çinlar [1975] for optimal stopping and Karlin [1962] for asset selling. Puterman [1994] analyzes an optimal parking problem where it is formulated as a countable state model and Ross [1983] analyzes an application in gambling. The tennis example in the exercises is adapted from Norman [1985].

1.11 Exercises

1.11.1 Computational Exercises

1. Consider the following deterministic expected total reward model with $S = \{s_1, s_2, s_3\}$; $A_{s_1} = \{a_{1,1}\}$,$A_{s_2} = \{a_{2,1,}, a_{2,2}\}$,$A_{s_3} = \{a_{3,1}\}$; $r(s_1, a_{1,1}) = 1$, $r(s_2, a_{2,1}) = 0$,$r(s_2, a_{2,2}) = c$, $r(s_3, a_{3,1}) = 1$ and $p(s_2|s_1, a_{1,1}) = 1$, $p(s_3|s_2, a_{2,1}) = 0$, $p(s_1|s_2, a_{2,2}) = 1$, $p(s_3|s_3, a_{3,1}) = 1$.

   (a) Represent the model graphically.

   (b) Classify the model when $c = -2, -1, 0, 1$.

   (c) Find the value of each deterministic stationary policy when $c = -2, -1, 0, 1$. What is $\mathbf{v}^*$ in each case?

   (d) Write out and solve the Bellman equation in each case.

(e) Investigate the convergence of value iteration for each value of c.

2. Solve the grid world example in this chapter using linear programming.

3. Consider a version of the grid world navigation problem in Section 1.9.1 in which an additional column has been added to the right of grid displayed in Figure **??**. Show how the optimal policy changes as a function of the probability, p of successfully exercising the intended movement in the following two cases:

   (a) There is a penalty for falling down the stairs combined with a reward for successfully delivering the coffee, and

   (b) the robot seeks to maximize the probability of successfully delivering the coffee.

4. Formulate and solve a version of the grid world navigation problem in Section 1.9.1 in which the surface is tilted being lower on the left and higher on the right. For example, if the robot is in cell and intends to move to cell 8, it reaches cell 8 with probability p, and reaches cells 10 and 14 with equal probability.

5. **The rational burglar:** Consider a Markov chain on $S = \{0, 1, \ldots, N\}$ with probability p of moving from s to $s+1$ and a probability of $1-p$ of moving from s to 0. Suppose in addition that states 0 and N are absorbing.

   (a) Formulate this as an optimal stopping problem in which the decision maker receives a reward of $g(s) = s$ if he decides to stop in state s.

   (b) Why do you think this is called the burgalr problem?

   (c) Classify the model.

   (d) Find an optimal policy numerically for $p = 0.01, 0.25, .5, .750.99$ using policy iteration or linear programming.

   (e) Describe the structure of the optimal policy and how it varies with p.

   (f) *Formally prove the optimality of the structured policy.

6. **A sequential gambling model:** As long as you have some money to bet, you can repeatably play a game of chance that has a win probability of p. If your fortune reaches 0 you can no longer play.

   Prior to each round of the game you may place a bet of b. If you win you receive $2b$ (the amount bet and an equal pay out) and if you lose, you receive 0. You must decide how much to bet on each round of the game given that your fortune can assume any (integer) value in between 0 and N dollars.

   (a) Formulate this as expected total reward model in which you seek to develop a betting strategy that maximizes the probability of reaching N.

(b) In what ways is this similar to the previous problem? In what ways is it different?

(c) Classify this model.

(d) Solve the model numerically for $N = 100$ and $p = 0.01, 0.25, 0.5, 0.75, 0.99$ and describe how the optimal strategy varies as a function of p.

(e) *Hypothesize and prove the form of the optimal strategy as a function of p.

7. Determine the optimal policy in the asset selling problem when the demand distribution is binomially and normally distributed.

8. **Lion Hunting Behavior** Consider a simplified infinite horizon version of the lion hunting behavior model in Section **??** in which the lion seeks a hunting policy to maximize survival time. Assume a lion's energy capacity $C = 30$ kgs, that a lion requires $d = 4$ kgs of energy per day to survive and hunting requires 1 additional kg of energy per day. A successful hunt which occurs with probability 0.25 yields 12 kgs of energy.

   (a) Formulate this as an expected total reward model in which the objective is to maximize the expected number of days of survival and the decision is whether to hunt or not.

   (b) Classify this model.

   (c) Provide the Bellman equations and transient Bellman equations.

   (d) Find and interpret the optimal policy and the expected survival time if the lion starts the planning horizon at its highest energy level.

   (e) Suppose there are two other species to hunt, one yields 7 kgs of energy with a catch probability of 0.5 and another which yields 20 kg of energy with a catch probability of 0.15. Reformulate and solve this more general model where the objective is to decide whether or not to hunt and if so, what species to hunt.

9. **Tetris** Formulate the game of Tetris as an expected total reward Markov decision processin which the objective to maximize the length of the game.

10. **Serving in Tennis** We consider the problem of choosing the type of serve to use at each point in a game of tennis. Tennis scoring is described in Section **??**, the salient point is that whenever the game reaches *deuce*, the game continues until a player wins two points in a row[25] Moreover at each point, if the first serve is "out", that is, it lands out of bounds, the server has second serve. If the second serve is out, the server loses the point.

[25]Deuce corresponds to a score of $2-2$, or $30-30$ in tennis jargon.

From the perspective of this chapter, a tennis game must be analyzed using an infinite horizon expected total reward model because the length of the game is random.

Consider the simplified situation where the server can use either a "fast" or "slow" serve. A fast serve lands in bounds with probability p_f and the server wins the point with probability q_f. Similarly a slow serve lands in bounds with probability p_s and the server wins the point with probability q_s. To be realistic, $q_f > q_s$ and $p_f < p_s$.

The objective is choose a service strategy that maximizes the probability of winning the game as a function of the score and whether it is a first serve or second serve.

(a) Formulate this as infinite horizon expected total reward model.

(b) Provide the Bellman equation.

(c) Find an optimal policy numerically when $p_f = 0.4$, $p_s = 0.8$, $q_f = 0.9$ and $q_s = 0.5$.

(d) Characterize the optimal strategy as a function of the model parameters.

(e) Reformulate and solve the problem when the parameters depend on the game score. Use your knowledge of tennis (or empirical data) to determine realistic parameter choice.

1.11.2 Theoretical Exercises

1. Show that if value iteration converges to $\mathbf{v}^*$ with $\mathbf{v}^0 = \mathbf{0}$, it converges to $\mathbf{v}^* + c\mathbf{e}$ if $\mathbf{v}^0 = c\mathbf{e}$.

2. Show using result in Section ?? that L_T is a contraction mapping.

3. Prove that if two Markov decision processes are positively singular, then they have the same sets of transient policies, they have the same optimal policy and value iteration converges at the same rate in each.

4. Prove in an optimal stopping problem on a random walk that when $g(s)$ is monotone and p is constant, that $v^*(s)$ is monotone and the continuation region is either empty, or a set of consecutive states.

5. Complete the details of the proof of Theorem 1.10.

6. Show by example that in an SSP, the evaluation equations will not have a solution when you choose an improper policy.

Bibliography

Jr. A.F. Veinott. On discrete dynamic programming programming with sensitive discount optimality criteria. *Ann. Math. Stat.*, 40:1635–1660, 1969.

A. Berman and R. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Academic Press, 1979.

D. Bertsekas and J. Tsitsiklis. An analysis of stochastic shortest path problems. *Math. Oper. Res.*, 16:580–595, 1991.

D. Blackwell. Positive dynamic programming. In *Proceedings of the 5th Berkeley Symposium on Probability and Statistics, Volume 1*, pages 415–418, 1967.

E. Çinlar. *Introduction to Stochastic Processes*. Prentice Hall, Englewood Cliffs, N.J., 1975.

C. Derman. *Finite State Markovian Decision Processes*. Academic Press, Newyork, NY., 1970.

J.H. Eaton and L.A. Zadeh. Optimal pursuit strategies in discrete state probabilistic systems. *Trans. ASME, Series D*, 84:23–29, 1962.

S. Karlin. Stochastic models and optimal policies for selling an asset. In K. Arrow, S.Karlin, and H. Scarf, editors, *Studies in Applied Probability and Management Science*, pages 148–158. Stanford University Press, Stanford,CA, 1962.

J.M. Norman. Dynamic programming in tennis - when to use a fast serve. *J. Opl. Res. Soc.*, 36:75–77, 1985.

M. L. Puterman. *Markov Decision Processes:Discrete Stochastic Dynamic Programming*. John Wiley & Sons., 1994.

S. Ross. *Introduction to Stochastic Dynamic Programming*. Academic Press, New York, 1983.

W. Rudin. *Principles of Mathematical Analysis, 2nd Edition*. McGraw-Hill, Inc., 1964.

R. Strauch. Negative dynamic programming. *Ann. Math. Stat.*, 37:871–890, 1966.

J. Wessels. Markov programming by successive approximations with respect to weighted supremum norms. *J. Math. Anal. Appl.*, 58:326–335, 1977.

Chapter 2

Index

1. xx