ISyE 6664: Stochastic Optimization

Rui Gong

October 1, 2024

Acknowledgements

These notes are based on the ISyE 6664 lectures given by Professor *Hayriye Ayhan* in Fall 2024 at Georgia Institute of Technology and the course notes of *Weiwei (William) Kong*.

Contents

1 Markov Decision Processes (MDPs)

We study *sequential decision making process:* a Markov Process, where the set of available actions, the rewards and transition probabilities depend on the current state and the action taken that state. It has the following ingredients:

- Decision epoch
- State space
- Actions space
- Rewards
- Transition probabilities

Example 1.1.

- Inventory Model: A warehouse manager observes his on hand inventory at the end of each month. Based on how many units he has, he decides to purchase new items or not to order anything at all.
  - the demand is random.
  - purchase cost
  - holding cost
  - reveneu from sales
  - pending cost for shortage
- Machine Replacement: A machine deteriorates over time. The decision maker checks the condition of the machine at the end of everyday and decides to keep or replace the machine.
  - state dependent income
  - state dependent cost
  - replacement cost
- Admission Control: Consider a system with k servers, i.e. the capacity is k, with service times following $\exp(\mu)$. One type of calls enters at a Poisson rate with parameter λ_1 and reward r_1 and another type of calls enters at a Poisson rate with parameter λ_2 and reward r_2 with $r_1 > r_2$.
  You should always accept the higher reward customers, and only reject the other set when as a number of servers greater M has filled up, where M is to be determined.

1.1 Modeling MDPs

Definition 1.1: Ingredients of a MDP

- Decision Epochs: T: *set of decision epochs,* $T = \{1, \ldots, N\}$ *where* $N - 1$ *is the time of last decision, and* N *is the time with a determined reward.* $T = \{1, 2, \ldots\}$ *if there are infinitely many epochs.*
- State Space (of the Markov Chain): S
- Action Space: A_s*: the set of possible actions in state* $s \in S$*, and the total action space is*

  $$A = \cup_{s \in S} A_s.$$

  We can choose actions deterministically or randomly. Let us define

  $$P(A_s) \text{ : collection of probability distributions on subsets of } A_s$$

  and $q(\cdot) \in P(A_s)$*. Basically, when you are in state* s*, you choose a particular action* a *with probability* $q(a)$*.*
- Rewards: $r_t(s, a)$ *is the reward received when action* a *is chosen in state* s *at time* t*.* $r_t(s, a, j)$ *is the reward earned when action* a *is chosen in state* s *at epoch* t *and the state is* j *at epoch* $t + 1$*, then*

  $$r_t(s, a) = \sum_{j \in S} P_t(j \mid s, a) r_t(s, a, j).$$

  For T *being finite, the terminal reward* $r_N(s)$ *is the reward earned at decision epoch* N *if the state is* s *at time* N*.*
- Transition Probability:

  $$p_t(j \mid s, a) : \text{probability of being in state } j \text{ at decision epoch } t + 1$$
  $$\text{given that } a \text{ is chosen in state } s \text{ at decision epoch } t.$$

The five-tuple

$$\{T, S, A, p(\cdot \mid \cdot, \cdot), r(\cdot, \cdot)\}$$

forms a Markov decision process (MDP).

Definition 1.2: Decision Rules and Policy

A decision rule *prescribes a procedure for action selection at a specified decision epoch.*
Markovian Deterministic Decision Rule:

$$d_t : S \mapsto A \text{ where } d_t(s) \in A_s,$$

where d_t represents the decision rules at decision epoch t.
Markovian Randomized Decision Rule

$$d_t : S \mapsto P(A) \text{ where } q_{d_t(s)}(\cdot) \in P\left(A_s\right),$$

where the decision rule in state s_t tells you a probability of possible actions.
History Dependent Deterministic Decision Rule H_t : *the set of all histories at decision epoch t, where $h_t \in H_t$ is a specific instance of history such that*

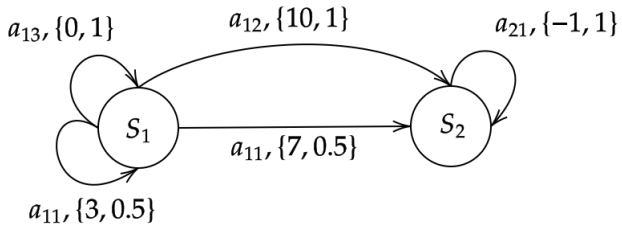
$$h_t = (s_1, a_1, s_2, a_2, \ldots, s_{t-1}, a_{t-1}, s_t) = (h_{t-1}, a_{t-1}, s_t),$$

then the rule can be represented as

$$d_t : H_t \to A, \text{ where } d_t(h_t) \in A_{s_t}.$$

History Dependent Randomized Decision Rule: $d_t : H_t \mapsto P(A)$.
Policy: *A policy π is a sequence of decision of rules. For finite epochs $T = \{1, \ldots, N\}$, $\pi = (d_1, d_2, \ldots, d_N)$; $T = \{1, 2, \ldots\}$, $pi = (d_1, d_2 \ldots)$. If $d_t = d$ for all $t \in T$, then $\pi = (d, d, \ldots) := d^\infty$ is called a stationary policy.*

Example 1.2. Consider the following plot of an MDP:



where for action a_{11}, it goes to S_2 with reward 7 and probability 0.5 OR go to S_1 with reward 3 with probability 0.5; similar interpretations for a_{12}, a_{13}, a_{21}. Specifically, $S = \{S_1, S_2\}, A_{S_1} = \{a_{11}, a_{12}, a_{13}\}, A_{S_2} = \{a_{21}\}, T = \{1, 2, 3\}$. For example, $P(S_1 \mid S_1, a_{11}) = 0.5, P(S_2 \mid S_1, a_{11}) = 0.5, r_{S_1, a_{11}, S_1)} = 3, r(S_1, a_{12}) = 10$.

Example 1.3 (Continued). A Markovian deterministic decision rule: $d_1(S_1) = a_{11}, d_1(S_2) = a_{21}, d_2(S_1) = a_{12}, d_2(S_2) = a_{21}$.
A history dependent deterministic decision rule: $d_1(S_1) = a_{11}, d_1(S_2) = a_{21}, d_2((S_1, a_{11}, S_1)) = a_{13}, d_2((S_1, a_{11}, S_2)) = a_{21}, d_2((S_1, a_{21}, S_2)) = a_{21}$

A Markovian randomized decision rule: $P(d_1(S_1) = a_{11}) = 0.6, P(d_1(S_1) = a_{13}) = 0.4, P(d_1(S_2) = a_{21}) = 1, P(d_2(S_1) = a_{11}) = 0.4, P(d_2(S_1) = a_{12}) = 0.6, P(d_2(S_2) = a_{21}) = 1.$

Example 1.4. An inventory manager checks his on-hand inventory at the end of each month. Depending on how many units he has on hand, he decides whether or not order new units from a supplier. Assume that newly purchased units arrive before the start of next month. Demand arrives during the month but orders are filled at the end of the month. Assume no backlogs are allowed, i.e., orders are lost if not enough inventories, and the warehouse has a capacity of M units. Let D_t be the monthly demand during month t and

$$P\left(D_t = j\right) = p_j \text{ for } j = 0, 1, 2, \ldots$$

Assume that if j units are purchased, the purchase cost is $C(j)$. The holding cost for u units is $h(u)$ and the revenue obtained from j units is $\rho(j)$. Finally, let $O(u)$ denote the wholesale purchase cost when u units are purchased and

$$O(u) = \begin{cases} k + C(u), & \text{if } u > 0 \\ 0, & \text{otherwise.} \end{cases}$$

The inventory manager would like to maximize his expected profit for the next N months. Let $g_N(s)$ be the terminal reward if there are s units left at time N.
Modeling this as a MDP, we have

$$\begin{aligned} T &= \{1, 2, \ldots, N\} \\ S &= \{0, 1, \ldots, M\} \\ A_s &= \{0, 1, \ldots, M - s\},\ \forall s \in S \\ p_t(j \mid s, a) &= \begin{cases} 0, & \text{if } j > s + a \\ p_{s+a-j}, & \text{if } 0 < j \leq s + a \\ \sum_{j=s+a}^{\infty} p_j & \text{if } j = 0 \end{cases} \\ r_t(s, a) &= -O(a) - h(s + a) + \sum_{j=0}^{s+a} \rho(j) p_j + \sum_{k=s+a+1}^{\infty} \rho(s + 1) p_j, \text{ for } t = 1, \ldots, N - 1 \\ r_N(s) &= g_N(s). \end{aligned}$$

Example 1.5. The condition of a machine used in a manufacturing process deteriorates over time. The condition of the machine is checked at predetermined discrete decision epochs. Let $S = \{0, 1, \ldots\}$ denote the state of the machine at each decision epoch. The higher the value of s is, the worse the condition of the machine. At each decision epoch, you can choose either to replace or keep as it is. Suppose replacements happen instantaneously. We assume in each period, the machine deteriorates by i states with probability $p(i)$. There is a fixed income of R units per period, a state dependent operating cost $h(s)$ where s is the state at the beginning of the period, and a replacement cost of K units. Suppose the objective is to maximize the long-run average

profit. Modeling this as a MDP, we have

$$\begin{aligned} T &= \{1, 2, \ldots\} \\ S &= \{0, 1, \ldots\} \\ A_s &= \{0, 1\}, \text{ where 1 indicates a replacement action} \\ p_t(j \mid s, 0) &= \begin{cases} 0, & \text{if } j < s \\ p(j-s), & \text{if } j \geq s \end{cases} \\ p_t(j \mid s, 1) &= p(j) \\ r_t(s, 0) &= R - h(s) \\ r_t(s, 1) &= R - K - h(0) \end{aligned}$$

1.2 Finite Horizon MDPs

Throughout this subsection, let $T = \{1, \ldots, N\}$ and $\pi = (d_1, d_2, \ldots, d_{N-1})$. Let $V_N^\pi(s)$ be the total expected reward for an N period problem under policy π when the system state at the first decision epoch is s.

Suppose π is a randomized history dependent policy and

$$\begin{aligned} X_t &: \text{ state at time } t \\ Y_t &: \text{ action chosen at time } t, \end{aligned}$$

where $\{X_t\}$ is the Markov Chain representing state under policy π. Then,

$$V_N^\pi(s) = \mathbb{E}^\pi \left[\sum_{t=1}^{N-1} r_t\left(X_t, Y_t\right) + r_N\left(X_N\right) \mid X_1 = s \right]$$

If, $\pi = (d_1, \ldots, d_{N-1})$ is a deterministic Markovian policy, then

$$V_N^\pi(s) = \mathbb{E}^\pi \left[\sum_{t=1}^{N-1} r_t\left(X_t, d_t\left(X_t\right)\right) + r_N\left(X_N\right) \mid X_1 = s \right]$$

If instead, $\pi = (d_1, \ldots, d_{N-1})$ is a history dependent deterministic policy, then

$$V_N^\pi(s) = \mathbb{E}^\pi \left[\sum_{t=1}^{N-1} r_t\left(X_t, d_t\left(h_t\right)\right) + r_N\left(X_N\right) \mid X_1 = s \right] \text{ with } h_t = \left(h_{t-1}, a_{t-1}, X_t\right),$$

where $|r(s, a) < M$ for all $a \in A_s, s \in S$.
If there exists $0 < \lambda < 1$ as a discount factor, then

$$V_N^\pi(s) = \mathbb{E}^\pi \left[\sum_{t=1}^{N-1} \lambda^{t-1} r_t\left(X_t, d_t\left(h_t\right)\right) + \lambda^{N-1} r_N\left(X_N\right) \mid X_1 = s \right] \text{ with } h_t = \left(h_{t-1}, a_{t-1}, X_t\right).$$

Define Π : the set of all possible history dependent randomized policies. Our objective is to find π^* (among all history dependent randomized policies) such that

$$V_N^{\pi^*}(s) \geq V_N^\pi(s), \text{ for all } \pi \in \Pi$$

and we would also like to compute

$$V_N^*(s) = \sup_{\pi \in \Pi} V_N^\pi(s),$$

where $V_N^*(s) = \max_{\pi \in \Pi} V_N^\pi(s)$ if the supremum is attained.

Now for a policy $\pi = (d_1, d_2, \ldots, d_{N-1})$, let us define the total expected reward from t to $N-1$, given h_t, as

$$u_t^\pi(h_t) = \mathbb{E}\left[\sum_{n=t}^{N-1} r_n(X_n, d_n(h_n)) + r_n(X_N) \mid H_t = h_t\right]$$

for $t = 1, \ldots, N-1$ and $u_N(h_N) = r_N(s_N)$ for all $h_N = (h_{N-1}, a_{N-1}, s_N)$. If π is Markovian deterministic, then

$$u_t^\pi(s_t) = \mathbb{E}\left[\sum_{n=t}^{N-1} r_n(X_n, d_n(X_n)) + r_N(X_n) \mid X_t = s_t\right]$$

If $h_1 = s$, then

$$u_1^\pi(s) = V_N^\pi(s) = \text{ total expected reward}$$

Note that $V_N^\pi(s)$ is not dependent on t. From recursively figuring out $V_N^\pi(s)$ by calculating $u_t^\pi(h_t)$, we can compute $V_N^\pi(s)$.

1.2.1 Backward Dynamic Programming for Computing the Expected Reward for a Finite Horizon Problem

1. Set $t = N$ and $u_N^\pi(h_N) = r_N(s_N)$, the terminal reward, for all $h_N = (h_{N-1}, a_{N-1}, s_N)$. Go to Step 2.
2. If $t = 1$, stop; otherwise go to Step 3 .
3. Substitute $t-1$ for t and compute $u_t^\pi(h_t)$ as

$$u_t^\pi(h_t) = r_t(s_t, d_t(h_t)) + \sum_{j \in S} p_t(j \mid s_t, d_t(h_t))\, u_{t+1}^\pi(\underbrace{h_t, d_t(h_t), j}_{h_{t+1}})$$

4. Return to Step 2.

For Markovian deterministic π, we have

$$u_t^\pi(h_t) = \underbrace{r_t(s_t, d_t(h_t))}_{\text{immediate reward}} + \underbrace{\sum_{j \in S} p(j \mid s_t, d_t(h_t))\, u_{t+1}^\pi(j)}_{\mathbb{E}_{h_t}^\pi[u_{t+1}]}$$

Theorem 1.3

Suppose that $\pi = (d_1, \ldots, d_{N-1})$ is a history dependent deterministic policy and u_t^π is obtained by the backward dynamic programming. Then for all $t \leq N$,

$$u_t^\pi(h_t) = \mathbb{E}_{h_t}\left[\sum_{n=t}^{N-1} r_n(X_n, d_n(h_n)) + r_N(X_N)\right]$$

and $V_N^\pi(s) = u_1^\pi(h_1)$ for $h_1 = s$.

Proof. Let $t = N$, $u_N^\pi(h_N) = r_N(s_N)$ for all $h_N = (h_{N-1}, a_{N-1}, s_N)$. Suppose the result holds for $n = t+1, \ldots, N$ and we will prove that it holds for $n = t$.

$$\begin{aligned}
u_t^\pi(h_t) &= r_t(s_t, d_t(h_t)) + \sum_{j \in S} p(j \mid s_t, d_t(h_t))\, u_{t+1}^\pi(h_t, d_t(h_t), j) \\
&= r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}\left[\mathbb{E}_{h_{t+1}}\left[\sum_{n=t+1}^{N-1} r_n(X_n, d_n(h_n)) + r_N(X_N)\right]\right] \\
&= r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}\left[\sum_{n=t+1}^{N-1} r_n(X_n, d_n(h_n)) + r_N(X_N)\right] \\
&= \mathbb{E}_{h_t}\left[\sum_{n=t}^{N-1} r_n(X_n, d_n(h_n)) + r_N(X_N)\right]
\end{aligned}$$

□

Suppose π were a randomized history dependent policy, then

$$u_t^\pi(h_t) = \sum_{a \in A_t} p(d_t(h_t) = a)\left(r_t(s_t, a) + \sum_{j \in S} p(j \mid s_t, a)\, u_{t+1}^\pi(h_t, a, j)\right)$$

1.2.2 Optimality Equations

We have

$$u_t^*(h_t) = \sup_{\pi \in \Pi} u_t(h_t), \; h_1 = s_1$$

where π belongs to the set of history dependent deterministic policies.

Lemma 1.4

Let w be a real valued function on an arbitrary discrete set W and let $q(\cdot)$ be a probability distribution on W. Then $\sup_{u \in W} w(u) \geq \sum_{u \in W} q(u) w(u)$

Proof. Let $w^* = \sup_{u \in W} w(u)$. Then

$$w^* = \sum_{u \in W} q(u) w^* \geq \sum_{u \in W} q(u) w(u)$$

□

That is, there is always a deterministic rule that performs as well/better than all randomized ones.

Optimality Equations for the N Period Problem Define

$$u_t(h_t) = \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a) u_{t+1}(h_t, a, j) \right\}$$

for $t = 1, \ldots, N-1$ and for $u_N(h_N) = r_N(s_N)$ for $h_N = (h_{N-1}, a_{N-1}, s_N)$.
If the supremum is obtained,

$$u_t(h_t) = \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a) u_{t+1}(h_t, a, j) \right\}$$

Recall that

$$u_t^*(h_t) = \sup_{\pi} u_t^{\pi}(h_t) \quad \text{and} \quad u_1^{\pi}(s) = V_N^{\pi}(s)$$

so by computing $u_1^*(s)$ like this, we will compute $V_N^*(s)$. In fact, we will show that, if we compute $u_t(h_t)$ as above, then it is actually $u_t^*(h_t)$ and hence we have $u_1(s_1) = V_N^*(s_1)$.

Theorem 1.5

Suppose that u_t is a solution to the optimality equations for $t = 1, \ldots, N-1$ with $u_N(s_N) = r_N(s_N)$. Then,
(a) $u_t(h_t) = u_t^*(h_t)$ for $t = 1, \ldots, N-1$
(b) $u_1(s_1) = V_N^*(s_1)$

Proof. We will first try to show taht $u_n(h_n) \geq u_n^*(h_n)$ for all $n = 1, \ldots, N$.
For $n = N$, $u_n(h_n) = r_N(s_N) = u_N^{\pi}(h_N)$ for all $\pi \in \Pi$ and $h_N = (h_{N_1}, a_{N-1}, s_N)$. Thus, the result holds for $n = N$. Assume it holds for $t = n+1, \ldots, N$, we will show that it holds for $t = n$ as well. Let $\pi = (d_1, \ldots, d_{N-1})$ be an arbitrary policy.

$$\begin{aligned}
u_n(h_n) &= \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_j(j \mid s_n, a) u_{n+1}(h_n, a, j) \right\} \\
&\geq \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_j(j \mid s_n, a) u_{n+1}^*(h_n, a, j) \right\} \\
&\geq r_n(s_n, d_n(h_n)) + \sum_{j \in S} P(j \mid s_n, d_n(h_n)) u_{n+1}^{\pi}(h_n, d_n(h_n), j) \\
&= u_n^{\pi}(h_n)
\end{aligned}$$

Since π is arbitrary,

$$u_n(h_n) \geq \sup_{\pi \in \Pi} u_n^{\pi}(h_n).$$

We will next show that for each $\epsilon > 0$, there exists π' such that

$$u_n^{\pi'}(h_n) + (N-n)\epsilon \geq u_n(h_n).$$

We will construct such a policy $\pi' = (d_1', d_2', \ldots, d_{N-1}')$ by choosing $d_n'(h_n)$ such that

$$r_n(s_n, d_n'(h_n)) + \sum_{j \in S} P_n(j \mid s_n, d_n'(h_n)) u_{n+1}^{\pi'}(h_n, d_n'(h_n), j) + \epsilon \geq u_n(h_n).$$

This π' exists by the definition of $u_n(h_n)$. Note $u_N^{\pi'} = r_N(s_N) = u_N(s_N)$ for $h_N = (h_{N-1}, a_{N-1}, s_N)$. Suppose the result holds for $t = n+1, \ldots, N$, then

$$\begin{aligned}
u_n^{\pi'}(h_n) &= r_n(s_n, d_n'(h_n)) + \sum_{j \in S} P_n(j \mid s_n, d_n'(h_n)) u_{n+1}^{\pi'}(h_n, d_n'(h_n), j) \\
&\geq r_n(s_n, d_n'(h_n)) + \sum_{j \in S} P_n(j \mid s_n, d_n'(h_n)) \left(u_{n+1}(h_n, d_n'(h_n), j) - (N-n-1)\epsilon\right) \\
&\geq r_n(s_n, d_n'(h_n)) + \left(\sum_{j \in S} P_n(j \mid s_n, d_n'(h_n)) u_{n+1}(h_n, d_n'(h_n), j)\right) + \epsilon - (N-n)\epsilon \\
&\geq u_n(h_n) - (N-n)\epsilon
\end{aligned}$$

But then for each n, we have

$$u_n^*(h_n) + (N-n)\epsilon \geq u_n^{\pi'}(h_n) + (N-n)\epsilon \geq u_n(h_n) \geq u_n^*(h_n),$$

which impllies $u_n(h_n) = u_n^*(h_n)$. □

Now the above theorem shows us a way to iteratively compute $u_n^*(h_n)$ and hence $V_N^*(s_1)$ in the end. Now, the following theorem will show that in fact, we are able to compute an optimal policy based on the iterations.

Theorem 1.6

Suppose that u_t^* for $t = 1, \ldots, N$ are solutions to the optimality equations subject to the boundary condition and the policy $\pi^* = \left(d_1^*, \ldots, d_{N-1}^*\right)$ satisfies

$$\begin{aligned}
&r_t\left(s_t, d_t^*\left(h_t\right)\right) + \sum_{j \in S} p_t\left(j \mid s_t, d_t^*\left(h_t\right)\right) u_{t+1}^*\left(h_t, d_t^*\left(h_t\right), j\right) \\
&= \max_{a \in A_{s_t}} \left\{ r_t\left(s_t, a\right) + \sum_{j \in S} p_t\left(j \mid s_t, a\right) u_{t+1}^*\left(h_t, a, j\right) \right\} \text{ for } t = 1, \ldots, N-1
\end{aligned}$$

(a) $u_t^*\left(h_t\right) = u_t^{\pi^*}\left(h_t\right)$

(b) π^* *is an optimal policy and* $V_N^{\pi^*}(s) = V_N^*(s)$.

Proof.

(a) Trivially

$$u_N^*(s_N) = r_N(s_N) = u_N^{\pi^*}(s_n)$$

Suppose that this holds for $n = t+1, \ldots, N$. We will show that it also holds for $n = t$. We have

$$\begin{aligned} u_t^*(h_t) &= \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(h_t, a, j) \right\} \\ &= r_t(s_t, d_t^*(h_t)) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^{\pi^*}(h_t, d_t^*(h_t), j) \\ &= u_t^{\pi^*}(h_t) \end{aligned}$$

(b) We have

$$V_N^{\pi^*}(s) = u_1^{\pi^*}(s) = u_1^*(s) = V_N^*(s)$$

Hence, the optimal policy $\pi^* = \left(d_1^*, \ldots, d_{N-1}^*\right)$ is defined as

$$d_t(h_t) \in \arg\max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(h_t, a, j) \right\}$$

□

That is, when we do the iteration, if we always pick the action maximizing $u_t(h_t)$, we get an optimal policy. Now we show that we actually only need s_t rather than h_t and there exists a deterministic policy if all $u_t(h_t)$ are attained by a deterministic action.

Theorem 1.7

Let u_t^* for $t = 1, \ldots, N$ be the solution to the optimality equations together with the boundary conditions.

(a) For each $t = 1, \ldots, N$, $u_t^*(h_t)$ depends on h_t only through s_t.

(b) If there exists $a^1 \in A_{s_t}$ such that

$$\begin{aligned} & r_t\left(s_t, a^1\right) + \sum_{j \in S} p\left(j \mid s_t, a^1\right) u_{t+1}^*\left(h_t, a^1, j\right) \\ = & \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(h_t, a, j) \right\} \end{aligned}$$

for all $t = 1, \ldots, N-1$ then there exists an optimal policy that is Markovian deterministic.

Proof.

(a) We have

$$u_N^*(h_N) = u_N^*(h_{N-1}, a_{N-1}, s_N) = r_N(s_N).$$

Thus, u_N^* depends on h_N only though s_N. The result holds for $n = N$. Let us assume it holds for $n = t+1, \ldots, N$ and we proceed to show that it also holds for $n = t$. We have

$$\begin{aligned} u_t^*(h_t) &= \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(h_t, a, j) \right\} \\ &= \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(j) \right\} \end{aligned}$$

and the result holds for $n = t$.

(b) Given policy $\pi^* = \left(d_1^*, \ldots, d_{N-1}^*\right)$ we have, from a previous result,

$$\begin{aligned} & r_t(s_t, d_t^*(h_t)) + \sum_{j \in S} p_t(j \mid s_t, d_t^*(h_t))\, u_{t+1}^{\pi^*}(j) \\ = & \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(j) \right\} \end{aligned}$$

□

Corollary 1.8

Let

π^{HR} : *set of history dependent randomized policies*

π^{MD} : *set of Markovian deterministic policies.*

Then,

$$V_N^*(s) = \sup_{\pi \in \pi^{HR}} V_N^\pi(s) = \sup_{\pi \in \pi^{MD}} V_N^\pi(s)$$

Proposition 1.9

Assume that S is finite or countable and if either one of the following conditions hold:

(a) A_s *is finite for each* $s \in S$.

(b) A_s *is compact for each* $s \in S$ *and*

$$r_t(s,a) \text{ is continuous in } a \text{ for all } s \in S$$
$$|r_t(s,a)| \leq M \text{ for all } a \in A_s, s \in S$$
$$p_t(j \mid s,a) \text{ is continuous in } a \text{ for each } j \in S, s \in S$$

(c) A_s *is compact for each* $s \in S$ *and*
$r_t(s,a)$ *is upper semicontinuous in* a *for all* $s \in S$,
$|r_t(s,a)| \leq M$ *for all* $a \in A_s, s \in S$,
$p_t(j \mid s,a)$ *is lower semicontinuous in* a *for each* $j \in S, s \in S$

then there exists a deterministic Markovian policy which is optimal.

Backward Induction Algorithm for the optimal policy and optimal total expected reward

(1) Set $t = N$ and $u_N^*(s_N) = r_N(s_N)$.

(2) Substitute $t-1$ for t and compute $u_t^*(s_t)$ for each $s_t \in S$ by

$$u_t^*(s_t) = \max_{a \in A_{s_t}} \left\{ r_t(s_t,a) + \sum_{j \in S} p_t(j \mid s_t,a)\, u_{t+1}^*(j) \right\}$$

and set

$$A_{s_t}^* = \arg\max_{a \in A_{s_t}} \left\{ r_t(s_t,a) + \sum_{j \in S} p_t(j \mid s_t,a)\, u_{t+1}^*(j) \right\}$$

(3) If $t = 1$ then stop. Otherwise go to step 2 .

Theorem 1.10

Suppose $\pi_t^*,\ t = 1, \ldots, N$ *and* $A_{s_t}^*$ *are obtained using backward dynamic programming.*

(i) For $t = 1, \ldots, N$ *and* $h_t = (h_{t-1}, a_t, s_t)$,

$$u_t^*(s_t) = \sup_{\pi \in \Pi} u_t^{\pi}(h_t),$$

where Π *is the set of all history dependent randomized policies.*

(ii) Let $d_t^*(s_t) \in A_{s_t}^*$, *for all* $s_t \in S,\ t = 1, \ldots, N-1$ *and* $\pi^* = (d_1^*, d_2^*, \ldots, d_{N-1}^*)$. *The* π^* *is optimal,*

$$u_1^*(s) = V_N^*(s) = V_N^{\pi^*}(s).$$

Example 1.6 (Inventory problem revisited). Consider the setup

$$M = 3, h(u) = u, \rho(u) = 8u, N = 4, T = \{1, 2, 3, 4\}$$
$$A_s = \{0, \dots, 3 - s\}$$

and

$$O(u) = \begin{cases} 4 + 2u, & u > 0 \\ 0, & u = 0 \end{cases}$$

with

$$P(D = 0) = \frac{1}{4}, P(D = 1) = \frac{1}{2}, P(D = 2) = \frac{1}{4}$$
$$r_N(0) = r_N(1) = r_N(2) = r_N(3) = 0$$

Now,

$$u_4^*(0) = u_4^*(1) = u_4^*(2) = u_4^*(3) = 0$$

and since

$$u_t^*(s_t) = \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j \mid s_t, a)\, u_{t+1}^*(j) \right\}$$

then

$$r(0,1) = -O(1) - h(1) + \rho(1)P(D = 1 \cup D = 2) = -6 - 1 + 8 \cdot \frac{3}{4} = -1$$
$$r(0,2) = -12 - 2 + 16 \cdot \frac{1}{4} + 8 \cdot \frac{1}{2} = -2$$
$$r(0,3) = -10 - 3 + 16 \cdot \frac{1}{4} + 8 \cdot \frac{1}{2} = -5$$
$$u_3^*(0) = \max\{0 + 1 \cdot 0, \underbrace{-1}_{=r(0,1)} + 0, \underbrace{-2}_{=r(0,2)}, \underbrace{-5}_{=r(0,3)}\} = 0, d_3^*(0) = 0$$

and continuing in this fashion, we will get

$$u_3^*(1) = 5, u_3^*(2) = 6, u_3^*(3) = 5$$
$$d_3^*(1) = 0, d_3^*(2) = 0, d_3^*(3) = 0$$

Next,

$$\begin{aligned} u_2^*(0) &= \max \left\{ 0, \underbrace{-6 - 1 + 8 * \frac{3}{4}}_{\text{reward}} + \underbrace{\frac{3}{4} * 0}_{\text{demand} \geq 1, u_3^*(0)} + \underbrace{\frac{1}{4} * 5}_{\text{demand}=0, u_3^*(1)}, 2, \frac{1}{2} \right\} \\ &= \max \left\{ 0, \frac{1}{4}, 2, \frac{1}{2} \right\} \\ &= 2 \end{aligned}$$

and $d_2^*(0) = 2$. Continuing, we will get

$$d_1^*(s) = \begin{cases} 3, & s = 0 \\ 0, & \text{otherwise} \end{cases}, d_2^*(s) = \begin{cases} 2, & s = 0 \\ 0, & \text{otherwise} \end{cases}$$

and $d_3^*(s) = 0$ for all $s \in \{1, 2, 3\}$. Finishing, we will get

$$v_4^*(0) = \frac{67}{16}, v_4^*(1) = \frac{129}{16}, v_4^*(2) = \frac{97}{8}, v_4^*(3) = \frac{227}{16}$$

1.3 Optimality of Monotone Policies

Consider

$$u_t^*(s) = \max_{a \in A_s} \left\{ r_t(s, a) + \sum_{j \in S} p_t(j \mid s, a) u_{t+1}^*(j) \right\}$$

Definition 1.11

We say that $g(\cdot, \cdot)$ for $x^+ \geq x^-$ in X and $y^+ \geq y^-$ in Y is superadditive if

$$g\left(x^+, y^+\right) + g\left(x^-, y^-\right) \geq g\left(x^+, y^-\right) + g\left(x^-, y^+\right)$$

If $-g(\cdot, \cdot)$ is superadditive then $g(\cdot, \cdot)$ is subadditive.

Example 1.7. $g(x, y) = h(x) + f(y)$ is both superadditive and subadditive.

Lemma 1.12

Suppose that g is a superadditive function in $X \times Y$ and for each $x \in X, \max_{y \in Y} g(x, y)$ exists. Then,

$$\rho(x) = \max\{y \in \arg\max_{y \in Y} g(x, y)\}$$

is monotone non-decreasing in X.

Proof. Let $x^+ \geq x^-$ and choose $y \leq \rho\left(x^-\right)$. Then,

$$g\left(x^-, \rho\left(x^-\right)\right) - g\left(x^-, y\right) \geq 0$$

Since g is superadditive,

$$\begin{aligned} & g\left(x^+, \rho(x^-)\right) + g\left(x^-, y\right) \geq g\left(x^+, y\right) + g\left(x^-, \rho\left(x^-\right)\right) \\ \Longrightarrow\ & g\left(x^+, \rho\left(x^-\right)\right) \geq \underbrace{\left[g\left(x^-, \rho(x^-)\right) - g\left(x^-, y\right)\right]}_{\geq 0} + g\left(x^+, y\right) \\ \Longrightarrow\ & g\left(x^+, \rho\left(x^-\right)\right) \geq g\left(x^+, y\right), \forall y \leq \rho(x^-). \end{aligned}$$

then by definition, $\rho\left(x^+\right) \geq \rho\left(x^-\right)$ since

$$g\left(x^+, \rho\left(x^+\right)\right) \geq g\left(x^+, \rho\left(x^-\right)\right) \text{ and } g\left(x^+, y\right) \leq g\left(x^+, \rho\left(x^-\right)\right),\ \forall y \leq \rho\left(x^-\right).$$

if $\rho(x^+) < \rho(x^-)$, then $g\left(x^+, \rho\left(x^+\right)\right) = g\left(x^+, \rho\left(x^-\right)\right)$, but then by the definition of $\rho(x^+)$, we have $\rho(x^+) \geq \rho(x^-)$. □

Lemma 1.13

Let $\{x_j\}, \{x'_j\}$ *be real-valued sequences satisfying*

$$\sum_{j=k}^{\infty} x_j \geq \sum_{j=k}^{\infty} x'_j$$

for all $k \geq 0$ *with equality holding for* $k = 0$. *Suppose* $v_{j+1} \geq v_j$ *for all* $j = 0, 1, \ldots$. *Then,*

$$\sum_{j=0}^{\infty} x_j v_j \geq \sum_{j=0}^{\infty} x'_j v_j$$

Proof. Set $v_{-1} = 0$. Then,

$$\begin{aligned}
\sum_{j=0}^{\infty} v_j x_j &= \sum_{j=0}^{\infty} x_j \sum_{i=0}^{j} (v_i - v_{i-1}) \\
&= \sum_{j=0}^{\infty} (v_j - v_{j-1}) \sum_{i=j}^{\infty} x_j \\
&= \sum_{j=1}^{\infty} (v_j - v_{j-1}) \sum_{i=j}^{\infty} x_j + v_0 \sum_{i=0}^{\infty} x_i \\
&\geq \sum_{j=1}^{\infty} (v_j - v_{j-1}) \sum_{i=j}^{\infty} x'_j + v_0 \sum_{i=0}^{\infty} x'_i \\
&= \sum_{j=0}^{\infty} v_j x'_j.
\end{aligned}$$

□

Note. A classical way to apply this lemma is that given two variables X, Y such that $P(X \geq a) \geq P(Y \geq a), \forall a$, then $\mathbb{E}[f(X)] \geq \mathbb{E}[f(y)]$ for every nondecreasing f.

Theorem 1.14

Assume that

1. $S = \{0, 1, \ldots\}$
2. $A_s = A$ *for all* $s \in S$

Suppose that

1. $r_t(s, a)$ *is non-decreasing (non-increasing) in* s *for all* $a \in A$ *and* $t = 1, \ldots, N-1$.
2. $\sum_{j=k}^{\infty} p_t(j \mid s, a)$ *is non-decreasing in* s *for all* $k \in S, a \in A$ *and* $t = 1, \ldots, N-1$.
3. $r_N(s)$ *is non-decreasing (non-increasing) in* s.

Then $u_t^*(s)$ *is non-decreasing (non-increasing) in* s *for all* $t = 1, \ldots, N$.

Proof. We know $u_N^*(s) = r_N(s)$ and thus the result holds for $t = N$. Now assume it holds for $n = t+1, \ldots, N$ and note that for $n = t$ we have

$$\begin{aligned} u_t^* &= \max_{a \in A_s} \left\{ r_t(s, a) + \sum_{j \in S} p_t(j \mid s, a) u_{t+1}^*(j) \right\} \\ &= r_t\left(s, a_s^*\right) + \sum_{j \in S} p_t\left(j \mid s, a_s^*\right) u_{t+1}^*(j) \end{aligned}$$

Suppose that $s' \geq s$. We need to show $u_t^*\left(s'\right) \geq u_t^*(s)$. Now

$$\begin{aligned} u_t^*(s) &= r_t\left(s, a_s^*\right) + \sum_{j \in S} p_t\left(j \mid s, a_s^*\right) u_{t+1}^*(j) \\ &\leq r_t\left(s', a_s^*\right) + \sum_{j \in S} p_t\left(j \mid s', a_s^*\right) u_{t+1}^*(j) \\ &\leq \max_{a \in A} \left\{ r_t\left(s', a\right) + \sum_{j \in S} p_t\left(j \mid s', a\right) u_{t+1}^*(j) \right\} \\ &= u_t^*\left(s'\right) \end{aligned}$$

which follows from the assumptions of the theorem, induction hypothesis and the earlier lemma. □

Theorem 1.15

Assume that

1. $S = \{0, 1, \ldots\}$
2. $A_s = A$ *for all* $s \in S$.

Suppose that

(1) $r_t(s, a)$ *is non-decreasing in* s *for all* $a \in A$ *and* $t = 1, \ldots, N-1$.

(2) $\sum_{j=k}^{\infty} p_t(j \mid s, a)$ *is non-decreasing in* s *for all* $k \in S, a \in A$ *and* $t = 1, \ldots, N-1$.

(3) $r_t(s, a)$ *is a superadditive function on* $S \times A$.

(4) $\sum_{j=k}^{\infty} p_t(j \mid s, a)$ *is a superadditive function on* $S \times A$.

(5) $r_N(s)$ *is non-decreasing in* s.

Then there exists an decision rules $d_t^*(s)$ *which are non-decreasing in* s *for all* $t = 1, \ldots, N-1$.

Proof. From 1, 2, and 5, we know that $u_t^*(s)$ is non-decreasing in s for all $t = 1, \ldots, N$ and so

$$\sum_{j=k}^{\infty} \left[p_t\left(j \mid s^+, a^+\right) + p_t\left(j \mid s^-, a^-\right)\right] \geq \sum_{j=k}^{\infty} \left[p_t\left(j \mid s^+, a^-\right) + p_t\left(j \mid s^-, a^+\right)\right]$$

for $s^+ \geq s^-, a^+ \geq a^-$, which implies, from the previous theorem, that

$$\sum_{j=0}^{\infty} \left[p_t\left(j \mid s^+, a^+\right) + p_t\left(j \mid s^-, a^-\right)\right] u_{t+1}^*(j) \geq \sum_{j=0}^{\infty} \left[p_t\left(j \mid s^+, a^-\right) + p_t\left(j \mid s^-, a^+\right)\right] u_{t+1}^*(j),$$

so $\sum_{j=0}^{\infty} p_t(j \mid s, a) u_{t+1}^*(j)$ is superadditive on $S \times A$. Since the sum of two superadditive functions is superadditive, then

$$r_t(s, a) + \sum_{j=0}^{\infty} p_t(j \mid s, a) u_{t+1}^*(j)$$

is superadditive and the result holds. □

Theorem 1.16

Suppose for $t = 1, \ldots, N-1$ *that*

(1) $r_t(s,a)$ *is non-increasing in* s *for all* $a \in A$ *and* $t = 1, \ldots, N-1$.

(2) $\sum_{j=k}^{\infty} p_t(j \mid s,a)$ *is non-decreasing in* s *for all* $k \in S, a \in A$ *and* $t = 1, \ldots, N-1$.

(3) $r_t(s,a)$ *is a superadditive function on* $S \times A$.

(4) $\sum_{j=0}^{\infty} p_t(j \mid s,a)$ *is a superadditive function on* $S \times A$.

(5) $r_N(s)$ *is non-increasing in* s.

Then there exists an optimal decision rules $d_t^*(s)$ *which are non-decreasing in* s *for all* $t = 1, \ldots, N-1$.

Proof. From (1), (2), and (5) we have $u_t^*(s)$ non-increasing in s. Then from (3) and (4), we have

$$r_t(s,a) + \sum_{j=0}^{\infty} p_t(j \mid s,a) u_t^*(j)$$

superadditive on $S \times A$. □

Backward Dynamic Programming for finding Monotone Optimal Policies Suppose that $S = \{0, 1, \ldots, M\}$ and $A_s = A$ for all $s \in S$.

1. Set $t = N$ and $u_N^*(s) = r_N(s)$ for all $s \in S$.
2. Substitute $t-1$ for t, set $s = 0$ and $A_0 = A$.
   (a) Set
   $$u_t^*(s) = \max_{a \in A_s} \left\{ r_t(s,a) + \sum_{j \in S} p_t(j \mid s,a) u_{t+1}^*(j) \right\}$$
   (b) Set
   $$A_{s,t}^* = \arg\max_{a \in A_s} \left\{ r_t(s,a) + \sum_{j \in S} p_t(j \mid s,a) u_{t+1}^*(j) \right\}$$
   (c) If $s = M$ go to step 3, otherwise set
   $$A_{s+1} = \left\{ a \in A : a \geq \max \left\{ a' \in A_{s,t}^* \right\} \right\}$$
   (d) Substitute $s+1$ for s and return to (a).
3. If $t = 1$, stop; otherwise go to Step 2.

Example 1.8. Given $S = \{0, 1, \ldots\}$, the higher the worse the equipment is. From one decision epoch to the next, the equipment deteriorates i states with probability $p(i)$. We are also given, $A_s = \{0, 1\}$ where 0 is "do nothing" and 1 is replacing, R is the fixed income per period, $h(s)$ is the operating cost if the equipment is in state s, K is the replacement cost, $r_N(s)$ is the salvage of the equipment if it is in state s at time N. Assume $h(s)$ is non-decreasing in s and $r_N(s)$ is non-increasing in s. Let $T = \{1, \ldots, N\}$.

We have:

$$p(j \mid s, 0) = \begin{cases} 0, & \text{if } j < s \\ p(j-s), & \text{if } j \geq s \end{cases} \text{ and } p(j \mid s, 1) = p(j), i = 0, 1, 2, \ldots$$

and

$$r(s, 0) = R - h(s) \text{ and } r(s, 1) = R - K - h(0)$$

1. $r(s, a)$ is non-increasing in s. Clearly this holds for the rewards.
2. $r_N(s)$ is non-increasing in s.
3. $\sum_{j=k}^{\infty} p_t(j \mid s, a)$ is non-decreasing in s for all $k \in S$ and $a \in A$ since when we replace,

$$\sum_{j=k+1}^{\infty} p(j \mid s+1, 1) - \sum_{j=k}^{\infty} p(j \mid s, 1) = \sum_{j=k}^{\infty} p(j) - \sum_{j=k}^{\infty} p(j) = 0$$

Now when we do not replace, for $k > s$,

$$\sum_{j=k}^{\infty} p(j \mid s+1, 0) - \sum_{j=k}^{\infty} p(j \mid s, 0) = \sum_{j=k}^{\infty} p(j-s-1) - \sum_{j=k}^{\infty} p(j-s) = p(k-s-1) \geq 0$$

and for $k \leq s$, we have

$$\sum_{j=k}^{\infty} p(j \mid s+1, 0) - \sum_{j=k}^{\infty} p(j \mid s, 0) = \sum_{j=s+1}^{\infty} p(j-s-1) - \sum_{j=s}^{\infty} p(j-s) = 0$$

4. $r(s, a)$ is superadditive on $S \times A$:

$$\begin{aligned} & r(s+1, 1) + r(s, 0) \geq r(s, 1) + r(s+1, 0) \\ \Longleftrightarrow & R - K - h(0) + R - h(s) \geq R - K - h(0) + R - h(s+1) \\ \Longleftrightarrow & h(s+1) - h(s) \geq 0 \end{aligned}$$

5. $\sum_{j=0}^{\infty} p(j \mid s,a)u(j)$ is superadditive on $S \times A$ for any non-increasing function u:

$$\begin{aligned}
&\sum_{j=0}^{\infty} p(j \mid s+1,1)u(j) + \sum_{j=0}^{\infty} p(j \mid s,0)u(j) \geq \sum_{j=0}^{\infty} p(j \mid s,1)u(j) + \sum_{j=0}^{\infty} p(j \mid s+1,0)u(j) \\
\iff &\sum_{j=0}^{\infty} p(j)u(j) + \sum_{j=s}^{\infty} p(j-s)u(j) \geq \sum_{j=0}^{\infty} p(j)u(j) + \sum_{j=s+1}^{\infty} p(j-s-1)u(j) \\
\iff &\sum_{j=s}^{\infty} p(j-s)u(j) \geq \sum_{j=s+1}^{\infty} p(j-s-1)u(j) \\
\iff &\sum_{j=s}^{\infty} p(j-s)u(j) - \sum_{j=s}^{\infty} p(j-s)u(j+1) \geq 0
\end{aligned}$$

since u is non-creasing.

6. $d_t^*(s) = \begin{cases} 0, \text{ if } & s \leq s_t^* \\ 1, \text{ if } & s > s_t^*, \forall t = 1, \ldots, N-1 \end{cases}$

2 Infinite Horizon MDPs

We assume:

- Transition probabilities and rewards are stationary and $|r(s,a)| \leq M$
- We are given a discount factor $0 < \lambda < 1$.
- $\pi = (d_1, d_2, \ldots)$ is Markovian deterministic.
- $T = \{1, 2, 3, \ldots\}$.
- $v_\lambda^\pi(s)$: total expected discounted reward under policy π when the initial state is s and the discount factor is λ. Let $\{X_t : t \geq 1\}$ is the Markov Chain under policy π,

$$v_\lambda^\pi(s) = \mathbb{E}_s \left[\sum_{t=1}^{\infty} \lambda^{t-1} r\left(X_t, d_t\left(X_t\right)\right) \right]$$

- r_d : vector of rewards under decision rule d

$$r_{d_1} = \begin{bmatrix} r(s_1, d_1(s_1)) \\ r(s_2, d_1(s_2)) \\ \vdots \end{bmatrix}$$

- P_d : probability transition matrix under decision rule d

Let us denote $v_\lambda^*(s) = \sup_\pi v_\lambda^\pi(s)$. If it exists, we would also like to find π^* where

$$v_\lambda^*(s) = v_\lambda^{\pi^*}(s).$$

If v_λ^π is the vector of total expected rewards, then

$$\begin{aligned} v_\lambda^\pi &= r_{d_1} + \lambda P_{d_1} r_{d_2} + \lambda^2 P_{d_1} P_{d_2} r_{d_3} + \ldots \\ &= \sum_{t=1}^{\infty} \lambda^{t-1} P_d^{t-1} r_d \\ &= r_{d_1} + \lambda P_{d_1} \left(r_{d_2} + \lambda P_{d_2} r_{d_3} + \ldots\right) \\ &= r_{d_1} + \lambda P_{d_1} v_\lambda^{\pi'} \end{aligned}$$

where $\pi' = (d_2, d_3, \ldots)$. Now if π is stationary, then

$$v_\lambda^\pi = r_d + \lambda P_d v_\lambda^\pi \implies v_\lambda^\pi = \left(I - \lambda P_d\right)^{-1} r_d$$

Theorem 2.1

For any stationary policy $\pi = d^\infty$, $v_\lambda^{d^\infty}$ is the unique solution of

$$v = r_d + \lambda P_d v$$

and furthermore, v_λ^∞ can be written as

$$v_\lambda^{d^\infty} = (I - \lambda P_d)^{-1} r_d = \sum_{t=1}^{\infty} \lambda^{t-1} P_d^{t-1} r_d = L_d v_\lambda^{d^\infty}$$

where $L_d(v) := r_d + \lambda P_d v$. Note the inverse exists because $\lambda < 1$.

Example 2.1. Consider a simple system with $S = \{s_1, s_2\}$ and $A_{s_1} = \{a_{11}, a_{12}\}$ and $A_{s_2} = \{a_{21}\}$. We have $p(s_1 \mid s_1, a_{11}) = 0.5$, $p(s_2 \mid s_1, a_{11}) = 0.5, p(s_2 \mid s_1, a_{12}) = 1$, and $p(s_2 \mid s_2, a_{21})$. Finally, $r(s_1, a_{11}, s_1) = 5, r(s_1, a_{11}, s_2) = 5, r(s_1, a_{12}) = 10$ and $r(s_2, a_{21}) = -1$. Consider the stationary policy that uses the decision rule $d(s_1) = a_{11}$ and $d(s_2) = a_{21}$. Compute $v_\lambda^{d^\infty}(s_1)$ and $v_\lambda^{d^\infty}(s_2)$.

We have $r_d = \begin{bmatrix} 5 \\ -1 \end{bmatrix}$ and

$$v_\lambda^{d^\infty}(s_1) = 5 + \lambda\left(0.5 v_\lambda^{d^\infty}(s_1) + 0.5 v_\lambda^{d^\infty}(s_2)\right)$$
$$v_\lambda^{d^\infty}(s_2) = -1 + \lambda v_\lambda^{d^\infty}(s_2) \implies v_\lambda^{d^\infty} = \frac{-1}{1-\lambda}$$

and so after substitution,

$$v_\lambda^{d^\infty}(s_1) = \frac{5 - 5.5\lambda}{(1-\lambda)(1-0.5\lambda)}.$$

Lemma 2.2

Suppose $0 \le \lambda < 1$. Then for any Markovian deterministic decision rule d,

(i) If $u \ge 0$ then $(I - \lambda P_d)^{-1} u \ge 0$ and $(I - \lambda P_d)^{-1} u \ge u$.

(ii) If $u \ge v$ then $(I - \lambda P_d)^{-1} u \ge (I - \lambda P_d)^{-1} v$.

(iii) If $u \ge 0$ then $u^T (I - \lambda P_d)^{-1} \ge 0$.

Proof. (i) and (iii): directly by

$$(I - \lambda P_d)^{-1} u = \sum_{t=1}^{\infty} \lambda^{t-1} P_d^{t-1} u \ge u \ge 0$$

(ii): follows from (i) by replacing u with $u - v$ □

2.1 Optimality Equations

$$v_n^*(s) = \sup_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} \lambda p(j \mid s,a) v_{n+1}^*(j) \right\}$$

by taking the limit as $n \to \infty$ on both sides,

$$v_\lambda^*(s) = \underbrace{\sup_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} \lambda p(j \mid s,a) v^*(j) \right\}}_{\mathcal{L}}$$

If v^* is the vector of $v^*(s)$ for $s \in S$, then $v^* = \mathcal{L}v^*$.
If the supremum is attained,

$$v_\lambda^*(s) = \underbrace{\max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} \lambda p(j \mid s,a) v^*(j) \right\}}_{L}$$

Theorem 2.3

Suppose that there exists a v such that

(i) $v \geq \mathcal{L}v$ *then* $v \geq v_\lambda^*$

(ii) $v \leq \mathcal{L}v$ *then* $v \leq v_\lambda^*$

(iii) $v = \mathcal{L}v$ *then* $v = v_\lambda^*$

Proof. (i) Let $\pi = (d_1, d_2, \ldots)$ and let us use the notation

$$\mathcal{L}v = \sup_\alpha \{r_\alpha + \lambda P_\alpha v^*\}$$
$$Lv = \max_\alpha \{r_\alpha + \lambda P_\alpha v\}$$

Then

$$\begin{aligned} v \geq & \sup_\alpha \{r_\alpha + \lambda P_d v^*\} = \mathcal{L}v = r_{d_1} + \lambda P_{d_1} v \\ \geq & r_{d_1} + \lambda P_{d_1} \left(r_{d_2} + \lambda P_{d_2} v\right) \\ & \vdots \\ \geq & r_{d_1} + \lambda P_{d_1} r_{d_2} + \lambda^2 P_{d_1} P_{d_2} r_{d_3} + \ldots + \lambda^{n-1} P_{d_1} \ldots P_{d_{n-1}} r_{d_n} + \lambda^n \underbrace{P_{d_1} \ldots P_{d_n}}_{P_\pi^n} v \end{aligned}$$

and also since

$$v_\lambda^\pi = r_{d_1} + \lambda P_{d_1} r_{d_2} + \ldots + \sum_{k=2}^{\infty} \lambda^k P_{d_1} \ldots P_{d_k} r_{d_{k+1}}$$

then

$$v - v_\lambda^\pi \geq \lambda^n P_{d_1} \dots P_{d_n} v - \sum_{k=n}^{\infty} \lambda^k P_{d_1} \dots P_{d_k} r_{d_{k+1}}$$

Next, if we define $\|v\| = \sup_{s \in S} |v(s)|$ then $\|\lambda^n P^n v\| \leq \lambda^n \|v\|$ then we can choose $\epsilon > 0$ such that there exists n sufficiently larges such that

$$-\frac{\epsilon}{2} e \leq \lambda^n P_{d_1} \dots P_{d_n} v \leq \frac{\epsilon}{2} e$$

where e is a vector of ones. Hence,

$$-\frac{\lambda^n M e}{(1-\lambda)} \leq \sum_{k=n}^{\infty} \lambda^k P_{d_1} \dots P_{d_k} r_{d_{k+1}} \leq \frac{\lambda^n M e}{(1-\lambda)}$$

by $|r_{d_{k+1}}| \leq Me$, and so with can find n sufficiently large so that

$$v - v_\lambda^\pi \geq \varepsilon e \implies v \geq \sup_\pi v_\lambda^\pi = v_\lambda^*$$

(ii) From the definition of $\mathcal{L}$, we know that for all $\epsilon > 0$ there exists α such that

$$v \leq r_\alpha + \lambda P_\alpha v + \epsilon e$$

which implies

$$\begin{aligned} &(I - \lambda P_\alpha) v \leq r_\alpha + \epsilon e \\ \implies &v \leq (I - \lambda P_\alpha)^{-1} (r_\alpha + \epsilon e) \\ \implies &v \leq (I - \lambda P_\alpha)^{-1} r_\alpha + (I - \lambda P_\alpha)^{-1} \epsilon e \end{aligned}$$

and hence

$$\begin{aligned} v &\leq v_\lambda^{d^\infty} + \epsilon \sum_{k=1}^{\infty} \lambda^{k-1} P_\alpha^{k-1} e \\ &= v_\lambda^{d^\infty} + \frac{\epsilon e}{1-\lambda} \\ &\leq \sup_\pi v_\lambda^\pi = v_\lambda^* \end{aligned}$$

where the last inequality is by pushing ϵ to 0.

(iii) Trivial.

□

Definition 2.4

Let U be a Banach space (complete normed linear space, e.g. $\mathbb{R}^n$). The operator $T : U \to U$ is a contraction mapping if $\exists \lambda$ with $0 \leq \lambda < 1$ such that

$$\|Tv - Tu\| \leq \lambda \|v - u\|$$

Theorem 2.5: Fixed Point Theorem

Suppose U is Banach space and $T : U \mapsto U$ is a contraction mapping. Then,

1. *$\exists v^* \in U$ unique such that $Tv^* = v^*$*
2. *for arbitrary $v^0 \in U$, the sequence $\{v^n\}$ defined by $v^{n+1} = Tv^n$ converges to v^*.*

Proof. (a) Directly

$$\begin{aligned}
\left\|v^{n+m} - v^n\right\| &= \left\|\sum_{k=0}^{m-1} v^{n+k+1} - \sum_{k=0}^{m-1} v^{n+k}\right\| \\
&\leq \sum_{k=0}^{m-1} \left\|v^{n+k+1} - v^{n+k}\right\| \\
&= \sum_{k=0}^{m-1} \left\|T^{n+k}v^1 - T^{n+k}v^0\right\| \\
&\leq \sum_{k=0}^{m-1} \lambda^{n+k} \left\|v^1 - v^0\right\| \\
&= \left\|v^1 - v^0\right\| \cdot \frac{\lambda^n (1 - \lambda^m)}{1 - \lambda}
\end{aligned}$$

and so $\{v^n\}$ is a Cauchy sequence and $\exists v^*$ such that $v^n \to v^*$. It remains to be seen that $Tv^* = v^*$. We have

$$\begin{aligned}
0 \leq \|Tv^* - v^*\| &\leq \|Tv^* - v^n\| - \|v^n - v^*\| \\
&\leq \left\|Tv^* - Tv^{n-1}\right\| - \|v^n - v^*\| \\
&\leq \lambda \left\|v^* - v^{n-1}\right\| - \|v^n - v^*\| .
\end{aligned}$$

Since $v^n \to v^*$ the the right hand side can be made arbitrarily small by picking large enough n. Hence $\|Tv^* - v^*\| = 0$ and $Tv^* = v^*$.
Suppose there exists v' such that $Tv' = v'$. Then,

$$\|v^* - v'\| = \|Tv^* - Tv'\| \leq \lambda \|v^* - v'\|$$

which is only possible if $\|v^* - v'\| = 0 \implies v^* = v'$.

□

Proposition 2.6

For $0 \leq \lambda < 1$, L and $\mathcal{L}$ are contraction mappings.

Proof. Let u and v be such that $Lv(s) \geq Lu(s)$ for $s \in S$ and

$$\max_{a \in A_s} \left\{ r(s,a) + \lambda \sum p(j \mid s,a) v(j) \right\} \geq \max_{a \in A_s} \left\{ r(s,a) + \lambda \sum p(j \mid s,a) u(j) \right\}$$

and suppose that

$$a_s^* \in \underset{a \in A_s}{argmax} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v(j) \right\}$$

Then

$$\begin{aligned} 0 \leq Lv(s) - Lu(s) &\leq r\left(s, a_s^*\right) + \lambda \sum_{j \in S} p\left(j \mid s, a_s^*\right) v(j) - r\left(s, a_s^*\right) - \lambda \sum_{j \in S} p\left(j \mid s, a_s^*\right) u(j) \\ &= \lambda \sum_{j \in S} p\left(j \mid s, a_s^*\right) [v(j) - u(j)] \\ &\leq \lambda \sum_{j \in S} p\left(j \mid s, a_s^*\right) \|v - u\| \\ &= \lambda \|v - u\| \end{aligned}$$

and we can have the similar result for $Lu(s) \geq Lv(s)$. Therefore,

$$|Lv(s) - Lu(s)| \leq \lambda \|v - u\| \implies \|Lv - Lu\| \leq \lambda \|v - u\|$$

and a similar argument can be made for $\mathcal{L}$. Note that L_d, through the same arguments, is also a contraction mapping. □

Theorem 2.7

1. *There exists a unique v^* satisfying $Lv^* = v^*$ ($\mathcal{L}v^* = v^*$) and hence $v_\lambda^* = v^*$.*
2. *For each d satisfying $L_d v = v$, there exists a unique solution $v = v_\lambda^\pi$ where $\pi = (d, d, \ldots)$. [$L_d v = r_d + \lambda P_d v$]*

Theorem 2.8

A policy π^* is optimal if and only if $v_\lambda^{\pi^*}$ is a solution to the optimality equations.

Proof. If π^* is optimal then $v_\lambda^* = v_\lambda^{\pi^*}$ and hence $Lv_\lambda^{\pi^*} = v_\lambda^{\pi^*}$. If $Lv_\lambda^{\pi^*} = v_\lambda^{\pi^*}$ then $v_\lambda^{\pi^*} = v_\lambda^*$ by the above theorem and hence π^* is optimal. □

Theorem 2.9

Suppose d is such that

$$L_{d^*} v_\lambda^* = r_{d^*} + \lambda P_{d^*} v_\lambda^* = v_\lambda^*$$

or $d^* \in \operatorname{argmax}\{r_d + \lambda P_d v_\lambda^*\}$ where we say that d^* is a conserving decision rule. Then, $(d^*)^\infty$ is an optimal decision policy and $v_\lambda^{(d^*)^\infty} = v_\lambda^*$.

Proof.

$$v_\lambda^* = L v_\lambda^* = r_{d^*} + \lambda P_{d^*} v_\lambda^* = v_\lambda^{(d^*)^\infty}$$

□

Theorem 2.10

Suppose there exists an optimal policy, then there exists an optimal stationary policy.

Proof. Given $\pi^* = (d_1, d_2, \ldots)$ and $\pi^* = (d_1, \pi')$. Then,

$$\begin{aligned} v_\lambda^{\pi^*} &= r_{d_1} + \lambda P_{d_1} v_\lambda^{\pi'} \\ &\leq r_{d_1} + \lambda P_{d_1} v_\lambda^{\pi^*} \\ &\leq \sup_d \left\{ r_d + \lambda P_d v_\lambda^{\pi^*} \right\} \\ &= \mathcal{L} v_\lambda^{\pi^*} = v_\lambda^{\pi^*} \end{aligned}$$

and d_1 is a conserving decision rule which means it is an optimal decision rule. □

2.2 Algorithms

We will be considering:

1. Value Iteration
2. Policy Iteration
3. Linear Programming

Theorem 2.11

Suppose that S is countable. Then there exists a stationary optimal policy if

(a) A_s is finite for each $s \in S$, or

(b) A_s is compact for each $s \in S, r(s,a)$ is continuous in a for each s, and $p(j \mid s, a)$ is continuous in a for each $j \in S$ and $s \in S$, or

(c) A_s is compact for each $s \in S, r(s,a)$ is upper semicontinuous in a for each s, and $p(j \mid s, a)$ is lower semicontinuous in a for each $j \in S$ and $s \in S$.

2.2.1 Value Iteration

We wish to find a policy π_ϵ such that $v_\lambda^{\pi_\epsilon} \geq v_\lambda^*(s) - \epsilon$.

(1) Select $v^0 \in V, \epsilon > 0$ and set $n = 0$

(2) For each $s \in S$, compute $v^{n+1}(s)$ as

$$v^{(n+1)}(s) = \max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v^n(j) \right\}$$

(3) If $\|v^{n+1} - v^n\| \leq \frac{\epsilon(1-\lambda)}{2\lambda}$ then go to step 4. Otherwise, increment n by 1 and go to step (2).

(4) For each $s \in S$, choose

$$d_\epsilon(s) \in \underset{a \in A_s}{arg\,max} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v^{n+1}(j) \right\}$$

Theorem 2.12

For value iteration, we have

(1) v^n converges to v_λ^*

(2) Stationary policy $(d_\epsilon)^\infty$ is an ϵ-optimal policy

Proof.

(1) Trivial, from fixed point theorem.

(2) We need to show that $\left\| v_\lambda^{(d_\epsilon)^\infty} - v_\lambda^* \right\| \leq \epsilon$, where $v_\lambda^{(d_\epsilon)^\infty}$ is the expected reward under the stationary policy $(d_\epsilon)^\infty$ satisfying $L_{(d_\epsilon)^\infty} v_\lambda^{(d_\epsilon)^\infty} = v_\lambda^{(d_\epsilon)^\infty}$. Note that

$$\left\| v_\lambda^{(d_\epsilon)^\infty} - v_\lambda^* \right\| \leq \left\| v_\lambda^{(d_\epsilon)^\infty} - v^{n+1} \right\| + \left\| v^{n+1} - v_\lambda^* \right\|.$$

First, we have

$$\begin{aligned}
\left\|v^{n+1} - v_\lambda^*\right\| &= \left\|\sum_{k=n+1}^{\infty} v^k - v^{k+1}\right\| \\
&\leq \sum_{k=n+1}^{\infty} \left\|v^k - v^{k+1}\right\| \\
&= \sum_{k=0}^{\infty} \left\|v^{k+n+1} - v^{k+n+2}\right\| \\
&= \sum_{k=0}^{\infty} \left\|L^{k+1}v^n - L^{k+1}v^{n+1}\right\| \\
&\leq \sum_{k=0}^{\infty} \lambda^{k+1} \left\|v^n - v^{n+1}\right\| \\
&\leq \sum_{k=0}^{\infty} \lambda^{k+1} \frac{\epsilon(1-\lambda)}{2\lambda} \\
&= \frac{\lambda}{1-\lambda} \cdot \frac{\epsilon(1-\lambda)}{2\lambda} \\
&= \frac{\epsilon}{2}
\end{aligned}$$

and

$$\begin{aligned}
\left\|v_\lambda^{(d_\epsilon)^\infty} - v^{n+1}\right\| &= \left\|L_{d_\epsilon} v_\lambda^{(d_\epsilon)^\infty} - v^{n+1}\right\| \\
&\leq \left\|L_{d_\epsilon} v_\lambda^{(d_\epsilon)^\infty} - Lv^{n+1}\right\| + \left\|Lv^{n+1} - v^{n+1}\right\| \\
&= \left\|L_{d_\epsilon} v_\lambda^{(d_\epsilon)^\infty} - L_{d_\epsilon} v^{n+1}\right\| + \left\|Lv^{n+1} - Lv^n\right\| \\
&\leq \lambda \left\|v_\lambda^{(d_\epsilon)^\infty} - v^{n+1}\right\| + \lambda \left\|v^{n+1} - v^n\right\| \\
\implies (1-\lambda) \left\|v_\lambda^{(d_\epsilon)^\infty} - v^{n+1}\right\| &\leq \lambda \left\|v^{n+1} - v^n\right\| \leq \frac{\epsilon(1-\lambda)}{2} \\
\implies \left\|v_\lambda^{(d_\epsilon)^\infty} - v^{n+1}\right\| &\leq \frac{\epsilon}{2}
\end{aligned}$$

where from the second to the third line, we first use the fact that

$$Lv^{n+1}(s) = \max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j \mid s, a) v^{n+1}(j) \right\}$$

and then the definition $d_\epsilon(s) = \underset{a \in A_s}{arg\,max} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j \mid s, a) v^{n+1}(j) \right\}$, which implies that $Lv^{n+1}(s)$ is obtained under policy $d_\epsilon(s)$, that is, $L_{d_\epsilon} v^{n+1}(s) = Lv^{n+1}(s)$. Combine the two results together, we get what we want.

□

Proposition 2.13

(1) Suppose $v \geq u$. Then $Lv \geq Lu$.

(2) Suppose that for some N, $Lv^N \leq (\geq) v^N$. Then $v^{N+m+1} \leq (\geq) v^{N+m}$ for all $m \geq 0$.

Proof.

1. Let $d' \in \operatorname{argmax}\{r_d + \lambda P_d u\}$. Then,

$$Lu = r_{d'} + \lambda P_{d'} u \leq r_{d'} + \lambda P_{d'} v \leq \max\{r_d + \lambda P_d v\} = Lv,$$

where the first inequality is by the fact that $P_{d'}$ is a nonnegative matrix.

2. Directly,

$$v^{N+m+1} = L^m L v^N \geq L^m v^N = v^{N+m}$$

and likewise for the $(\leq)$ case.

□

For the second property of the proposition above, it says that if such N exists, then from N, such property holds for all iterations after that. So if $v^1 \geq v^0$ in value iteration, then $\{v^n\} \to v_\lambda^*$ is monotone decreasing. For example, if $Lv^0 \geq v^0$, then $v^{n+1} \geq v^n$ for all n, similar for $\leq$, but for some problems, v^1, v^0 might not be comparable.

Definition 2.14

Let $y_n \to y^*$, so $\lim \|y_n - y^*\| = 0$. We say y_n converges of order α *if there exists a $k > 0$ such that*

$$\|y_{n+1} - y^*\| \leq k \|y_n - y^*\|^\alpha.$$

Theorem 2.15

(i) Convergence rate of value iteration is linear in λ.

(ii)

$$\limsup_{n\to\infty} \left[\frac{\|v^n - v_\lambda^*\|}{\|v^0 - v_\lambda^*\|}\right]^{\frac{1}{n}} \leq \lambda$$

(iii) For every n,

$$\|v^n - v_\lambda^*\| \leq \frac{\lambda^n}{1-\lambda} \|v^1 - v^0\|$$

(iv) For every $d_\epsilon = \arg\max\{r_d + \lambda P_d v^n\}$,

$$\|v^{(d_n)^\infty} - v_\lambda^*\| \leq \frac{2\lambda^n}{1-\lambda} \|v^1 - v^0\|$$

Proof. (i) $\|v^{n+1} - v_\lambda^*\| = \|Lv^n - Lv_\lambda^*\| \leq \lambda \|v^n - v_\lambda^*\|$

(ii) Directly from (i)

(iii) Similar to the first part of the proof of Theorem 2.12.

(iv) Similar to the proof of Theorem 2.12.

□

2.2.2 Policy Iteration

(a) Set $n = 0$ and select arbitrary decision rule d_0

(b) (Policy Evaluation) Obtain v^n by solving

$$(I - \lambda P_{d_n})\, v^n = r_{d_n}$$

(c) (Policy Increment) Choose d_{n+1} such that

$$d_{n+1} \in \underset{d}{argmax}\, \{r_d + \lambda P_d v^n\}$$

and setting $d_{n+1} = d_n$ if possible. That is, if d_n is in the $\arg\max$ above, always pick $d_{n+1} = d_n$.

(d) If $d_{n+1} = d_n$ then stop and return $d^* = d_n$, otherwise increment n by 1 and return to (b)

- Advantages: Works well for solving d^* and even 1 iteration is a good heuristic.
- Disadvantages: Computing step (b)

Proposition 2.16

Let v^n, v^{n+1} be successive values generated by policy iteration. Then $v^{n+1} \geq v^n$.

Proof. Directly

$$\begin{aligned} & r_{d_{n+1}} + \lambda P_{d_{n+1}} v^n \geq r_{d_n} + \lambda P_{d_n} v^n = v^n \\ \Longrightarrow\ & r_{d_{n+1}} \geq \left(I - \lambda P_{d_{n+1}}\right) v^n \\ \Longrightarrow\ & \left(I - \lambda P_{d_{n+1}}\right)^{-1} r_{d_{n+1}} \geq v^n \\ \Longrightarrow\ & v^{n+1} \geq v^n \end{aligned}$$

□

Theorem 2.17

For a finite state and action space, policy iteration terminates after a finite number of step with a stationary (discounted) optimal policy $(d^*)^\infty$

That is, when we stop, our v^n solves the optimality equations and d_n is a conserving decision rule. It is finite because we have a finite number of actions and states.

Example 2.2. Recall example with

$$S = \{s_1, s_2\}, A_{s_1} = \{a_{11}, a_{12}\}, A_{s_2} = \{a_{21}\}$$

and

$$\begin{aligned} p(s_1 \mid s_1, a_{11}) &= \frac{1}{2} \\ p(s_2 \mid s_1, a_{11}) &= \frac{1}{2} \\ p(s_2 \mid s_1, a_{12}) &= 1 \\ p(s_2 \mid s_2, a_{21}) &= 1 \end{aligned}$$

and general $\lambda \in [0, 1)$. We also have

$$r(s_1, a_{11}) = 5, r(s_1, a_{12}) = 10, r(s_2, a_{21}) = -1$$

The policy iteration is:

(1) Let $d_0(s_1) = a_{11}$ and $d_0(s_2) = a_{21}$

(2) $\equiv$ (b) Get

$$v_\lambda^{(d_0)^\infty}(s_1) = \frac{5 - 5.5\lambda}{(1 - 0.5\lambda)(1 - \lambda)} \text{ and } v_\lambda^{(d_0)^\infty}(s_2) = \frac{-1}{1 - \lambda}$$

(3) $\equiv$ (c) Get

$$\begin{aligned} d_1(s_1) &\in \operatorname{argmax}\left\{5 + \frac{1}{2}v_\lambda^{(d_0)^\infty}(s_1) + \frac{1}{2}v_\lambda^{(d_0)^\infty}(s_2), 10 + v_\lambda^{(d_0)^\infty}(s_2)\right\} \\ \implies d_1(s_1) &\in \arg\max\left\{\frac{(5 - 5.5\lambda)}{(1 - 0.5\lambda)(1 - \lambda)}, \frac{2(5 - 5.5\lambda)}{1 - \lambda}\right\} \end{aligned}$$

Now if $\lambda > \frac{10}{11}$, we have $d_1(s_1) = a_{11}$, otherwise we have $d_1(s_1) = a_{12}$.
For example, let $\lambda = 0.95$ and $d_0(s_1) = a_{12}, d_0(s_2) = a_{21}$. Then

$$\begin{aligned} v_0 &= r_{d_0} + \lambda P_{d_0} v_0 \\ v_0(s_1) &= 10 + 0.95 v_0(s_2) \implies v_0(s_1) = -9 \\ v_0(s_2) &= -1 + 0.95 v_0(s_2) \implies v_0(s_2) = -20 \end{aligned}$$

And hence

$$d_s(1) = \arg\max\{\underbrace{5 + 0.95(0.5(-9) + 0.5 * 20)}_{a_{11}}, \underbrace{10 + 0.95(-20)}_{a_{12}}\} = \arg\max\{-8.775, -9\} = a_{11}.$$

Hence, $d_1(s_1) = a_{11}, d_1(s_2) = a_{21}$ which is different from d_0, we need to run the iteration again and we go back to the analysis above.

2.2.3 Modified Policy Iteration

Let $\{m_n\}$ be a sequence of non-negative integers.

(1) Select v^0, specify $\epsilon > 0$, and set $n = 0$.

(2) (Policy Improvement) Choose d_{n+1} to satisfy

$$d_{n+1} \in \underset{d}{argmax} \{r_d + \lambda P_d v^n\}$$

and setting $d_{n+1} = d_n$ if possible (when $n > 0$).

(3) (Partial Policy Evaluation)

a. Set $k = 0$ and

$$u_n^0 = \max_{d \in D} \{r_d + \lambda P_d v^n\}$$

or equivalently,

$$u_n^0(s) = \max_{a \in A_s} \left\{ r_d(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v^n(j) \right\}$$

b. If $\|u_n^0 - v^n\| < \frac{\epsilon(1-\lambda)}{2\lambda}$ go to step (4). Otherwise go to c.

c. If $k = m_n$ go to e., otherwise compute u_n^{k+1} by

$$u_n^{k+1} = r_{d_{n+1}} + \lambda P_{d_{n+1}} u_n^k = L_{d_{n+1}} u_n^k$$

d. Increment k by 1 and return to c.

e. Set $v^{n+1} = u_n^{m_n}$, increment n by 1 and go to step (2).

(4) Set $d_\epsilon = d_{n+1}$.

2.3 Linear Programming

If $v \geq Lv$ then $v \geq v_\lambda^*$ by Proposition 2.13. For each $j \in S$ pick $\alpha(j) > 0$ and consider the primal LP:

$$\begin{aligned} &\min_v \sum_{j \in S} \alpha(j) v(j) \\ &\text{s.t. } v(s) \geq r(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v(j), \forall s \in S, \forall a \in A_s \end{aligned}$$

where the constraint is equivalent to

$$v(s) - \lambda \sum_{j \in S} p(j \mid s,a) v(j) \geq r(s,a), \forall s \in S, \forall a \in A_s.$$

Also, please note that the constraint is equivalent to

$$v(s) \geq \max_{a \in A_s} \{r(s,a) + \lambda \sum_{j \in S} p(j \mid s,a) v(j)\} \iff v \geq Lv \implies v \geq v_\lambda^*.$$