Week 13

Nicolo Michelusi

I. Approximate Dynamic Programming (Bertsekas Vol. II, Ch. 6)

- Let us focus on the discounted cost MDP.
- Solving a Markov decision process requires finding a fixed point of

$$J^* = T(J^*),$$

where

$$T(J)(i) = \min_{u \in \mathcal{U}(i)} \bar{c}(i,u) + \alpha \sum_{j=1}^{n} P_{j|i,u} J(j),\ \forall i = 1, \ldots, n.$$

Generally, solving this problem requires policy evaluation, i.e. to evaluate the performance of a certain stationary policy μ by iteratively solving

$$J_{k+1} = T_\mu(J_k),$$

(this converges to J_μ when $k \to \infty$); and a policy improvement step, which aims to find an improved policy as solution of

$$\hat{\mu}(i) = \arg \min_{u \in \mathcal{U}(i)} \bar{c}(i,u) + \alpha \sum_{j=1}^{n} P_{j|i,u} J(j).$$

The optimal policy μ^* and optimal cost J^* are found by iterating these two steps until convergence

- However, in many practical systems, the state space is so large that it is not practical to compute J_μ in all states, let alone to store the vector J_μ in a look-up table.

- In these cases, we are interested to develop techniques to approximate J_μ by

$$J_\mu \approx \tilde{J}_\mu(r),$$

where r is a low-dimensional parameter vector (much smaller than the number of states), which we want to find, for instance, by minimizing

$$\min_r \|J_\mu - \tilde{J}_\mu(r)\|_2.$$

Once a good approximation $J_\mu(\hat{r})$ is found, one can execute the improvement step as

$$\hat{\mu}(i) = \arg \min_{u \in \mathcal{U}(i)} \bar{c}(i,u) + \alpha \sum_{j=1}^{n} P_{j|i,u} J_\mu(j;\hat{r}).$$

- Note that, with this approximation method, there is no need to store the entire n-dimensional vector J_μ; instead, we can store the low-dimensional vector r, and compute on the fly $J_\mu(j;\hat{r})$ based on the approximation architecture employed.

- There are many architectures that define the approximation function $J_\mu(r)$, for instance:
– DNN: $J_\mu(r)$ could represent a deep neural network, parameterized by r
– Linear model: $J_\mu(i) = \phi(i)^T r$, where $\phi(i)$ is a given feature vector associated to state i, capturing relevant features of the state.
– etc.
– We will focus on the linear case $J_\mu(i) = \phi(i)^T r$, but many extensions are possible.

- Additionally, it is often even difficult to compute expectations, due to the complexity of the system dynamics. In these cases, it is convenient to use Monte Carlo methods that simulate the process $\{(X_k, C_k), k \geq 0\}$. Roughly speaking, if we have a very long sequence $\{(X_k, C_k), k \geq 0\}$, we can estimate transition probabilities and costs in each states; we can then use these estimates in the policy evaluation step.

- Example: Tetris (Bertsekas Vol. II, P.393): we can model the problem of finding an optimal tetris playing strategy as a stochastic shortest path problem (it can be shown that the game terminates w.p. 1)

  State: the board configuration, a binary vector of size 200 (a standard board is width 10 x height 20); $x_i \in \{0, 1\}$ represents whether pixel i is occupied or not; hence there are 2^{200} states!

  Action: choose the position and rotation of each falling block, given the state x and the shape of the falling block

  Reward: when a solid horizontal line with no gaps is formed, it disappears and any blocks above it fall down to fill the space. The number of disappearing lines is the reward accumulated.

  Let J^* be the optimal reward vector; this is huge, since $J^*(x)$ defines the optimal reward starting from state x and there are 2^{200} states. Given J^* and the current state x and shape s, one can apply the control u specifying the position p and rotation r of the block; the optimal control is determined as

  $$(p, r)^* = \arg\min_{p,r} g(x, s, p, r) + J^*(f(x, s, p, r)),$$

  where $g()$ is the number of solid lines, and $f(x, s, p, r)$ is the next state.

  However, there is no way one can store the optimal reward vector $J^* \in \mathbb{R}^{2^{200}}$, let alone compute it; therefore , we need approximation methods.

  J^* has been successfully approximated in practice by low-dimensional linear architectures

  $$J = \Phi\mathbf{r}.$$

  For example, the following features have been used: the heights of the columns (10 columns), the height differentials of adjacent columns (9 of them); the maximum wall height, the number of holes of the board, and the vector $\mathbf{1}$, adding to 22 features [BeI96]. These are readily recognized by tetris players as capturing important aspects of the board position.

The $2^{200} \times 22$ feature matrix Φ ($J = \Phi\mathbf{r}$) cannot be stored in a computer, but for any board position, the corresponding vector of features can be easily generated and this is sufficient for implementation of the associated approximate DP algorithms explained next.

II. Direct Approximation

- Consider a stationary policy μ. Under such policy, let J be the optimal cost vector. This satisfies

$$J = \bar{c} + \alpha \mathbf{P} J,$$

where $\mathbf{P}$ is the transition probability matrix (under the given policy μ) and $\bar{c}$ is the cost vector (under the given policy μ). Solving we obtain

$$J = (\mathbf{I} - \alpha \mathbf{P})^{-1} \bar{c}.$$

- Direct approx aims to directly approximating the value function J
- We wish to approximate J with a linear approximation

$$J \approx \Phi \mathbf{r},$$

where Φ is a given feature matrix, and $\mathbf{r}$ is a parameter vector, to be designed.

- To this end, we wish to solve

$$r^* = \arg\min \|J - \Phi\mathbf{r}\|_\sigma^2,$$

where $\|x\|_\sigma$ is the weighted norm, defined as

$$\|x\|_\sigma = \sqrt{\sum_{i=1}^{n} \sigma_i x_i^2},$$

with weight vector $\sigma > 0$.

- Assume that Φ is $n \times s$, with $s < n$, and rank s. Then, r^* can be computed in closed form as (set gradient to zero and solve):

$$\mathbf{r}^* = (\Phi^T \Sigma \Phi)^{-1} \Phi^T \Sigma J,$$

where $\Sigma = \mathrm{diag}(\sigma)$

- There are two problems:

  - Computing operations with huge matrices Φ may be unfeasible;

  - J may not be available to start with.

- To approach the above problems, we may use Monte Carlo methods: we can simulate a very long sequence $\{(X_k, C_k), 0 \leq k \leq N-1\}$ of states and costs according to the dynamics determined by state transitions and policy μ;

Using this sequence, we note that

$$J(X_k) \approx \sum_{t=k}^{N-1} \alpha^{t-k} C_t, \ \forall k = 0, \dots, N-1$$

(indeed, the expectation of the left hand side is equal to $J(X_k)$ when $N \to \infty$)

Then, we can define r^* by minimizing

$$\min_r \frac{1}{2} \sum_{k=0}^{N-1} \left(\phi(X_k)^T \mathbf{r} - \sum_{t=k}^{N-1} \alpha^{t-k} C_t \right)^2 .$$

The solution is

$$\begin{aligned} \mathbf{r}_N^* &= \left(\sum_{k=0}^{N-1} \phi(X_k)\phi(X_k)^T \right)^{-1} \sum_{k=0}^{N-1} \sum_{t=k}^{N-1} \alpha^{t-k} \phi(X_k) C_t \\ &= B_N^{-1} \frac{1}{N} \sum_{t=0}^{N-1} z_t C_t \end{aligned}$$

where we have defined

$$z_t = \sum_{k=0}^{t} \alpha^{t-k} \phi(X_k), \ B_N = \frac{1}{N} \sum_{k=0}^{N-1} \phi(X_k)\phi(X_k)^T.$$

Note that, when $N \to \infty$, letting $\xi_i = \mathbb{P}(X_k = i)$ at steady-state (steady-state probability of being in state X_k), and $\bar{c}(i) = \mathbb{E}[C_k | X_k]$, then (Take the expectation and then $N \to \infty$)

$$B_N \to \sum_{i=1}^{n} \xi_i \phi(i)\phi(i)^T = \Phi^T \Xi \Phi$$

$$\frac{1}{N} \sum_{t=0}^{N-1} z_t C_t \to \lim_{N \to \infty} \frac{1}{N} \sum_{i,j} \xi_i P_{i,j}^{(t-k)} \sum_{k=0}^{N-1} \sum_{t=k}^{N-1} \alpha^{t-k} \phi(i) \bar{c}(j) = \Phi^T \Xi [\mathbf{I} - \alpha \mathbf{P}]^{-1} \bar{c} = \Phi^T \Xi J$$

where $P_{i,j}^{(q)}$ is the probability of going from state i to state j in q steps, also computed as $P_{i,j}^{(q)} = [\mathbf{P}^q]_{i,j}$, and $\mathbf{P}$ is the one-step transition probability; above, we have defined the column vector $\bar{c}$, and noticed that

$$J = [\mathbf{I} - \alpha \mathbf{P}]^{-1} \bar{c}.$$

Therefore, when $N \to \infty$,

$$\mathbf{r}_N^* \to (\Phi^T \Xi \Phi)^{-1} \Phi^T \Xi J = \arg\min \|\Phi \mathbf{r} - J\|_\xi.$$

- Note that $\mathbf{r}_N^*$ can be computed online as new samples are collected, starting from $\mathbf{r}_0^*$, $z_0 = \phi(X_0)$, $B_0 = 0$, $S_0 = 0$ as, for $N \geq 0$:

$$B_{N+1} = \left(1 - \frac{1}{N+1}\right) B_N + \frac{1}{N+1} \phi(X_N) \phi(X_N)^T$$

$$S_{N+1} = \left(1 - \frac{1}{N+1}\right) S_N + \frac{1}{N+1} z_N C_N$$

$$z_{N+1} = \alpha z_N + \phi(X_{N+1}).$$

$$\mathbf{r}_{N+1}^* = B_{N+1}^{-1} S_{N+1}$$

III. Projected equation methods for policy evaluation

- This method aims at solving a projected form of Bellman's equation
- Again, let

$$J = (\mathbf{I} - \alpha\mathbf{P})^{-1}\bar{c}.$$

- Note that J satisfies Bellman's equation under policy μ,

$$J = T_\mu(J)$$

- We want to approximate $J \approx \Phi\mathbf{r}$ in such a way that the approximation solves a projected form of Bellman's equation, i.e.

$$\Phi\mathbf{r}^* = \Pi[T_\mu(\Phi\mathbf{r}^*)],$$

where $\Pi[x]$ is the projection onto the subspace spanned by Φ under a given norm $\|\cdot\|$, i.e.

$$\Pi[x] = \Phi\mathbf{r}^*, \text{ where } \mathbf{r}^* = \arg\min_r \|\Phi\mathbf{r} - x\|.$$

The significance of this equation is as follows: $\Phi\mathbf{r}^*$ (approximation of J) is first applied to the operator T_μ; this operation might bring $T_\mu(\Phi\mathbf{r}^*)$ outside of the subspace spanned by Φ; we bring it back to that subspace via projection; this whole operation must define a fixed point, the same as J is a fixed point of $J = T_\mu(J)$.

- **Assumptions**: we make some simplifying assumptions:

- ξ_i (steady-state probability of visiting state i) satisfies $\xi_i > 0, \forall i$ (i.e., all states are visited infinitely often)

- Φ is $n \times s$ with $s < n$, and has rank s

- We use the weighted norm $\|\cdot\|_\xi$ with weight vector ξ to define the projection operation Π_ξ

- Note that the vector ξ satisfies $\xi^T\mathbf{P} = \xi^T$, i.e.

$$\xi_i = \sum_{j=1}^{n} \xi_j \mathbb{P}(X_{k+1} = i | X_k = j)$$

- Under these assumptions we obtain:

$$\mathbf{r}^* = \arg\min_{\mathbf{r}} \|\Phi\mathbf{r} - T_\mu(\Phi\mathbf{r}^*)\|_\xi = (\Phi^T\Xi\Phi)^{-1}\Phi^T\Xi[\bar{c} + \alpha\mathbf{P}\Phi\mathbf{r}^*],$$

or equivalently

$$\mathbf{r}^* = B^{-1}d,$$

where we have defined

$$B = \Phi^T \Xi \left(\mathbf{I} - \alpha \mathbf{P} \right) \Phi, \; d = \Phi^T \Xi \bar{c}$$

- Monte Carlo method: We consider Monte Carlo methods to estimate B, d: consider a very long sequence $\{(X_k, C_k), 0 \leq k \leq N-1\}$ of states and costs according to the dynamics determined by state transitions and policy μ. Then,

$$d \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t) C_t \triangleq d_k$$

$$B \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t) \left[\phi(X_t) - \alpha \phi(X_{t+1}) \right]^T \triangleq B_k$$

(to see this, compute the expectation of the left-hand side with respect to X_t, X_{t+1} at steady-state).

Then, since $\mathbf{r}^* = B^{-1} d$, one can estimate it as

$$\mathbf{r}_k = B_k^{-1} d_k,$$

where, starting from $B_0 = 0$, $d_0 = 0$,

$$B_{k+1} = \left(1 - \frac{1}{k+1} \right) B_k + \frac{1}{k+1} \phi(X_k) \left[\phi(X_k) - \alpha \phi(X_{k+1}) \right]^T$$

$$d_{k+1} = \left(1 - \frac{1}{k+1} \right) d_k + \frac{1}{k+1} \phi(X_k) C_k$$

When $k \to \infty$, $B_k \to B$, $d_k \to d$ and $\mathbf{r}_k \to \mathbf{r}^*$ (provided that the respective empirical frequences of occurrence of states and transitions asymptotically converge to ξ_i and $\mathbf{P}_{i,j}, \forall i, j$)

- Projected Value Iteration (PVI): we consider an iterative method to find the fixed point $\mathbf{r}^*$ of $\Phi\mathbf{r}^* = \Pi_\xi[T_\mu(\Phi\mathbf{r}^*)]$; similarly, to value iteration, we consider updates of the type

$$\Phi\mathbf{r}_{k+1} = \Pi_\xi[T_\mu(\Phi\mathbf{r}_k)]$$

starting from any initialization $\mathbf{r}_0$. Given $\mathbf{r}_k$, we obtain

$$\mathbf{r}_{k+1} = (\Phi^T\Xi\Phi)^{-1}\Phi^T\Xi[\bar{c} + \alpha\mathbf{P}\Phi\mathbf{r}_k] = \mathbf{r}_k - (\Phi^T\Xi\Phi)^{-1}\left[B\mathbf{r}_k - d\right]$$

Does this converge to $\mathbf{r}^*$ when $k \to \infty$? Yes, based on the following theorem:

Theorem 1. *Let Π_ξ be the projection with respect to the weighted norm with weight vector ξ. Then, under the above assumptions, $\Pi_\xi[T_\mu(\cdot)]$ is a contraction with modulus α, with respect to the weighted norm $\|\cdot\|_\xi$, i.e.,*

$$\|\Pi_\xi[T_\mu(J_1)] - \Pi_\xi[T_\mu(J_2)]\|_\xi \leq \alpha\|J_1 - J_2\|_\xi.$$

Proof. First, note that Projection reduces distance:

$$\|\Pi_\xi[T_\mu(J_1)] - \Pi_\xi[T_\mu(J_2)]\|_\xi \leq \|T_\mu(J_1) - T_\mu(J_2)\|_\xi = \alpha\|\mathbf{P}(J_1 - J_2)\|_\xi.$$

Hence, it is sufficient to prove that

$$\|\mathbf{P}x\|_\xi \leq \|x\|_\xi.$$

Indeed,

$$\|\mathbf{P}x\|_\xi^2 = \sum_{i=1}^n \xi_i \left[\sum_{j=1}^n \mathbf{P}_{i,j}x_j\right]^2 \leq \sum_{i=1}^n \xi_i \sum_{j=1}^n \mathbf{P}_{i,j}\left[x_j\right]^2 = \sum_{i=1}^n \xi_j \left[x_j\right]^2 = \|x\|_\xi^2$$

□

Then, we obtain

$$\|\Phi\mathbf{r}_{k+1} - \Phi\mathbf{r}^*\|_\xi = \|\Pi_\xi[T_\mu(\Phi\mathbf{r}_k)] - [T_\mu(\Phi\mathbf{r}^*)]\|_\xi \leq \alpha\|\Phi\mathbf{r}_k - \Phi\mathbf{r}^*\|_\xi \leq \cdots \leq \alpha^{k+1}\|\Phi\mathbf{r}_0 - \Phi\mathbf{r}^*\|_\xi.$$

Then, taking the limit $k \to \infty$, we obtain

$$\|\Phi\mathbf{r}_k - \Phi\mathbf{r}^*\|_\xi \to 0.$$

- Monte Carlo methods: Note that PVI is of the form

$$\mathbf{r}_{k+1} = \mathbf{r}_k - (\Phi^T \Xi \Phi)^{-1} [B\mathbf{r}_k - d]$$

We consider Monte Carlo methods to estimate these updates: consider a very long sequence $\{(X_k, C_k), 0 \leq k \leq N-1\}$ of states and costs according to the dynamics determined by state transitions and policy μ. Then,

$$\Phi^T \Xi \Phi \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t)\phi(X_t)^T \triangleq G_k$$

$$d \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t) C_t \triangleq d_k$$

$$B \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t) \left[\phi(X_t) - \alpha\phi(X_{t+1})\right]^T \triangleq B_k$$

(to see this, compute the expectation of the left-hand side with respect to X_t, X_{t+1} at steady-state).

Then, we can approximate $\mathbf{r}_{k+1}$ as

$$\mathbf{r}_{k+1} = \mathbf{r}_k - G_k^{-1} [B_k \mathbf{r}_k - d_k]$$

where, starting from $B_0 = 0$, $d_0 = 0$,

$$B_{k+1} = \left(1 - \frac{1}{k+1}\right) B_k + \frac{1}{k+1} \phi(X_k) \left[\phi(X_k) - \alpha\phi(X_{k+1})\right]^T$$

$$d_{k+1} = \left(1 - \frac{1}{k+1}\right) d_k + \frac{1}{k+1} \phi(X_k) C_k$$

This method is called Least squares policy evaluation (LSPE).

Clearly, $G_k \to \Phi^T \Xi \Phi$, $B_k \to B$, $d_k \to d$ as $k \to \infty$, hence $\mathbf{r}_k \to \mathbf{r}^*$, provided that the respective empirical frequences of occurrence of states and transitions asymptotically converge to ξ_i and $\mathbf{P}_{i,j}, \forall i, j$.

- Temporal Difference method TD(0)

  Note that $\mathbf{r}^*$ is a fixed point of

$$\mathbf{r} = \mathbf{r} - \gamma(B\mathbf{r} - d),\ \gamma > 0.$$

It can be shown that, for γ small enough, $(\mathbf{I} - \gamma B)$ has eigenvalues strictly within the unit circle, hence $Q[x] \triangleq (\mathbf{I} - \gamma B)x + \gamma d$ is a contraction. As a result, the iterative algorithm

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma(B\mathbf{r}_k - d)$$

converges to the fixed point $\mathbf{r}^* = B^{-1}d$. Again, we want to develop a simulation based method to do this. We can use this algorithm (TD(0)):

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma_k \phi(X_k) q_k,$$

where we have defined the temporal difference

$$q_k \triangleq \phi(X_k)^T \mathbf{r}_k - C_k - \alpha\phi(X_{k+1})^T \mathbf{r}_k,$$

and γ_k is now a diminishing step-size, to combat randomness. To see this, note that

$$\mathbb{E}[q_k|\mathbf{r}_k, X_k = i] = \phi(i)^T \mathbf{r}_k - \bar{c}(i) - \alpha \sum_{j=1}^{n} \mathbf{P}_{i,j}\phi(j)^T \mathbf{r}_k = \tilde{J}(i) - \bar{c}(i) - \alpha \sum_{j=1}^{n} \mathbf{P}_{i,j}\tilde{J}(j)$$

hence at steady-state

$$\begin{aligned}\mathbb{E}[\phi(X_k)q_k|\mathbf{r}_k] &= \sum_{i=1}^{n} \phi(i)\xi_i\phi(i)^T \mathbf{r}_k - \sum_{i=1}^{n} \phi(i)\xi_i\bar{c}(i) - \alpha \sum_{i,j=1}^{n} \phi(i)\xi_i \mathbf{P}_{i,j}\phi(j)^T \mathbf{r}_k \\ &= \Phi^T \Xi \left(\mathbf{I} - \alpha\mathbf{P}\right) \Phi \mathbf{r}_k - \Phi^T \Xi \bar{c} = B\mathbf{r}_k - d\end{aligned}$$

TD(0) is typically slower to converge (needs a diminishing step-size), but has very low complexity and require minimal memory requirements: the update requires only to compute the temporal difference q_k and update

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma_k \phi(X_k) q_k,$$

IV. Multistep simulation based methods

- We are now going to generalized the above techniques. Again, we consider the policy evaluation problem under a fixed policy μ.
- Note that the cost-to-go J under policy μ is the fixed point of $J = T_\mu(J)$; hence, we also have $J = T_\mu(J) = T_\mu[T_\mu(J)]$, and by induction,

$$J = T_\mu^k(J), \ \forall k \geq 1,$$

where T_μ^k is the operator obtained by applying k times the operator T_μ to J.
- Hence, consider the operator, for $\lambda \in [0, 1)$,

$$T_\mu^{(\lambda)} = (1 - \lambda) \sum_{k=0}^{\infty} \lambda^k T_\mu^{k+1}.$$

Then, it follows that

$$T_\mu^{(\lambda)}(J) = (1 - \lambda) \sum_{k=0}^{\infty} \lambda^k T_\mu^{k+1}(J) = (1 - \lambda) \sum_{k=0}^{\infty} \lambda^k J = J,$$

i.e., J is also a fixed point of $J = T_\mu^{(\lambda)}(J)$.
- We can express $T_\mu^k(x)$ explicitly as

$$T_\mu(x) = \bar{c} + \alpha \mathbf{P} x,$$

$$T_\mu^2(x) = \bar{c} + \alpha \mathbf{P} T_\mu(x) = \bar{c} + \alpha \mathbf{P} \bar{c} + \alpha^2 \mathbf{P}^2 x,$$

hence (as can be easily seen by induction)

$$T_\mu^k(x) = \sum_{t=0}^{k-1} \alpha^t \mathbf{P}^t \bar{c} + \alpha^k \mathbf{P}^k x, \ \forall k \geq 1.$$

Hence, we can express $T_\mu^{(\lambda)}(x)$ explicitly as

$$T_\mu^{(\lambda)}(x) = (1-\lambda) \sum_{k=0}^{\infty} \lambda^k T_\mu^{k+1}(x) = (1-\lambda) \sum_{k=0}^{\infty} \lambda^k \sum_{t=0}^{k} \alpha^t \mathbf{P}^t \bar{c} + (1-\lambda) \sum_{k=0}^{\infty} \lambda^k \alpha^{k+1} \mathbf{P}^{k+1} x = \bar{c}_\lambda + \alpha_\lambda \mathbf{P}_\lambda x$$

where we have defined

$$\bar{c}_\lambda = \sum_{t=0}^{\infty} [\lambda \alpha]^t \mathbf{P}^t \bar{c} = [\mathbf{I} - \lambda \alpha \mathbf{P}]^{-1} \bar{c},$$

$$\mathbf{P}_\lambda = (1 - \lambda \alpha) \sum_{k=0}^{\infty} \lambda^k \alpha^k \mathbf{P}^{k+1},$$

and

$$\alpha_\lambda = \frac{(1 - \lambda)\alpha}{1 - \lambda \alpha}.$$

Therefore, we can interpret $T_\mu^{(\lambda)}$ as a MDP with cost-per-stage vector $\bar{c}_\lambda$, one-step transition probability $\mathbf{P}_\lambda$ (indeed, it is easy to verify that $\mathbf{P}_\lambda$ is a transition probability with steady-state distribution ξ), and discount factor α_λ.

Note that we recover the previous section by letting $\lambda = 0$.

Note also that $\alpha_\lambda < \alpha,\ \forall \lambda > 0$, so that this operation corresponds to a stronger discounting.

- We can now apply the same algorithms as we used before for projected equation methods: we want to find a fixed point $\mathbf{r}^*$ of
$$\Phi\mathbf{r}^* = \Pi_\xi[T_\mu^{(\lambda)}(\Phi\mathbf{r}^*)].$$
This is given by
$$\mathbf{r}^* = B_\lambda^{-1} d_\lambda,$$
where we have defined
$$B_\lambda = \Phi^T \Xi \left(\mathbf{I} - \alpha_\lambda \mathbf{P}_\lambda\right)\Phi,\ d_\lambda = \Phi^T \Xi \bar{c}_\lambda.$$
It will be useful to express B_λ and d_λ as
$$B_\lambda = \Phi^T \Xi \sum_{t=0}^{\infty} \lambda^t \alpha^t \mathbf{P}^t [\mathbf{I} - \alpha \mathbf{P}]\Phi$$
$$d_\lambda = \Phi^T \Xi \sum_{t=0}^{\infty} [\lambda\alpha]^t \mathbf{P}^t \bar{c}$$
- Monte Carlo method: We consider Monte Carlo methods to estimate B_λ, d_λ: consider a very long sequence $\{(X_k, C_k), 0 \leq k \leq N-1\}$ of states and costs according to the dynamics determined by state transitions and policy μ. Then,
$$d_\lambda \approx \frac{1}{k}\sum_{\tau=0}^{k-1} z_\tau C_\tau \triangleq d_{\lambda,k}$$
$$B_\lambda \approx \frac{1}{k}\sum_{\tau=0}^{k-1} z_\tau [\phi(X_\tau) - \alpha\phi(X_{\tau+1})]^T \triangleq B_{\lambda,k}$$
where we have defined the eligibility vector
$$z_\tau \triangleq \sum_{t=0}^{\tau} [\lambda\alpha]^{\tau - t} \phi(X_t)$$

.

To see this, note that

$$\mathbb{E}[z_\tau | X_0 = i] = \sum_j \sum_{t=0}^{\tau} [\lambda\alpha]^t [\mathbf{P}^{\tau-t}]_{i,j} \phi(j)$$

hence

$$\mathbb{E}[d_{\lambda,k}] = \frac{1}{k} \sum_{\tau=0}^{k-1} \sum_{t=0}^{\tau} [\lambda\alpha]^t \sum_{i,j} \phi(i)\xi_i [\mathbf{P}^t]_{i,j} \bar{c}_j = \frac{1}{k} \sum_{\tau=0}^{k-1} \sum_{t=0}^{\tau} [\lambda\alpha]^t \Phi^T \Xi \mathbf{P}^t \bar{c}$$

$$\to \sum_{t=0}^{\infty} [\lambda\alpha]^t \Phi^T \Xi \mathbf{P}^t \bar{c} = d_\lambda \text{ for } k \to \infty$$

Similarly,

$$\mathbb{E}\left[B_{\lambda,k}\right] = \sum_i \phi(i)\xi_i \frac{1}{k} \sum_{\tau=0}^{k-1} \sum_{t=0}^{\tau} \lambda^t \alpha^t \left[\sum_j [\mathbf{P}^t]_{i,j} \phi(j) - \alpha \sum_m [\mathbf{P}^{t+1}]_{i,m} \phi(m) \right]^T$$

$$= \frac{1}{k} \sum_{\tau=0}^{k-1} \sum_{t=0}^{\tau} [\lambda\alpha]^t \Phi^T \Xi \mathbf{P}^t [\mathbf{I} - \alpha \mathbf{P}] \Phi$$

$$\to \sum_{t=0}^{\infty} [\lambda\alpha]^t \Phi^T \Xi \mathbf{P}^t [\mathbf{I} - \alpha \mathbf{P}] \Phi = B_\lambda \text{ for } k \to \infty.$$

It is then clear that $\mathbb{E}[d_{\lambda,k}] \to d_\lambda$ and $\mathbb{E}[B_{\lambda,k}] \to B_\lambda$ as $k \to \infty$.

Then, since $\mathbf{r}^* = B_\lambda^{-1} d_\lambda$, one can estimate it as

$$\mathbf{r}_k = B_{\lambda,k}^{-1} d_{\lambda,k},$$

where, starting from $B_{\lambda,0} = 0$, $d_{\lambda,0} = 0$, $z_0 = \phi(X_0)$,

$$d_{\lambda,k+1} = \left(1 - \frac{1}{k+1}\right) d_{\lambda,k} + \frac{1}{k+1} z_k C_k$$

$$B_{\lambda,k+1} = \left(1 - \frac{1}{k+1}\right) B_{\lambda,k} + \frac{1}{k+1} z_k [\phi(X_k) - \alpha\phi(X_{k+1})]^T,$$

$$z_{k+1} = \lambda\alpha z_k + \phi(X_{k+1}).$$

- Projected Value Iteration PVI(λ): Similarly, we consider a generalization of PVI for $\lambda > 0$. We consider an iterative method to find the fixed point $\mathbf{r}^*$ of $\Phi\mathbf{r}^* = \Pi_\xi[T_\mu^{(\lambda)}(\Phi\mathbf{r}^*)]$ of the form

$$\Phi\mathbf{r}_{k+1} = \Pi_\xi[T_\mu^{(\lambda)}(\Phi\mathbf{r}_k)]$$

starting from any initialization $\mathbf{r}_0$. Given $\mathbf{r}_k$, we obtain

$$\mathbf{r}_{k+1} = (\Phi^T \Xi \Phi)^{-1} \Phi^T \Xi [\bar{c}_\lambda + \alpha_\lambda \mathbf{P}_\lambda \Phi \mathbf{r}_k] = \mathbf{r}_k - (\Phi^T \Xi \Phi)^{-1} \left[B_\lambda \mathbf{r}_k - d_\lambda\right]$$

Does this converge to $\mathbf{r}^*$ when $k \to \infty$? Yes, based on the following theorem:

Theorem 2. *Let Π_ξ be the projection with respect to the weighted norm with weight vector ξ. Then, under the above assumptions, $\Pi_\xi[T_\mu^{(\lambda)}(\cdot)]$ is a contraction with modulus α_λ, with respect to the weighted norm $\|\cdot\|_\xi$, i.e.,*

$$\|\Pi_\xi[T_\mu(J_1)] - \Pi_\xi[T_\mu(J_2)]\|_\xi \leq \alpha \|J_1 - J_2\|_\xi.$$

Then, we obtain

$$\|\Phi\mathbf{r}_{k+1} - \Phi\mathbf{r}^*\|_\xi \leq \alpha_\lambda^{k+1} \|\Phi\mathbf{r}_0 - \Phi\mathbf{r}^*\|_\xi.$$

Then, taking the limit $k \to \infty$, we obtain

$$\|\Phi\mathbf{r}_k - \Phi\mathbf{r}^*\|_\xi \to 0.$$

- Monte Carlo methods: Note that PVI(λ) is of the form

$$\mathbf{r}_{k+1} = \mathbf{r}_k - (\Phi^T \Xi \Phi)^{-1} \left[B_\lambda \mathbf{r}_k - d_\lambda \right]$$

We consider Monte Carlo methods to estimate these updates: consider a very long sequence $\{(X_k, C_k), 0 \le k \le N-1\}$ of states and costs according to the dynamics determined by state transitions and policy μ. Then,

$$\Phi^T \Xi \Phi \approx \frac{1}{k} \sum_{t=0}^{k-1} \phi(X_t)\phi(X_t)^T \triangleq G_k$$

$$d_\lambda \approx \frac{1}{k} \sum_{\tau=0}^{k-1} z_\tau C_\tau \triangleq d_{\lambda,k}$$

$$B_\lambda \approx \frac{1}{k} \sum_{\tau=0}^{k-1} z_\tau [\phi(X_\tau) - \alpha\phi(X_{\tau+1})]^T \triangleq B_{\lambda,k}$$

where we have defined the eligibility vector

$$z_\tau \triangleq \sum_{t=0}^{\tau} [\lambda\alpha]^{\tau-t} \phi(X_t)$$

.

Then, we can approximate $\mathbf{r}_{k+1}$ as

$$\mathbf{r}_{k+1} = \mathbf{r}_k - G_k^{-1} \left[B_{\lambda,k} \mathbf{r}_k - d_{\lambda,k} \right]$$

where, starting from $B_{\lambda,0} = 0$, $d_{\lambda,0} = 0$, $z_0 = \phi(X_0)$,

$$d_{\lambda,k+1} = \left(1 - \frac{1}{k+1}\right) d_{\lambda,k} + \frac{1}{k+1} z_k C_k$$

$$B_{\lambda,k+1} = \left(1 - \frac{1}{k+1}\right) B_{\lambda,k} + \frac{1}{k+1} z_k [\phi(X_k) - \alpha\phi(X_{k+1})]^T,$$

$$z_{k+1} = \lambda\alpha z_k + \phi(X_{k+1}).$$

This method is called Least squares policy evaluation LSPE(λ).

Clearly, $G_k \to \Phi^T \Xi \Phi$, $B_k \to B$, $d_k \to d$ as $k \to \infty$, hence $\mathbf{r}_k \to \mathbf{r}^*$.

- Temporal Difference method TD(λ)

  Note that $\mathbf{r}^*$ is a fixed point of

  $$\mathbf{r} = \mathbf{r} - \gamma(B_\lambda \mathbf{r} - d_\lambda),\ \gamma > 0.$$

  It can be shown that, for γ small enough, $(\mathbf{I} - \gamma B_\lambda)$ has eigenvalues strictly within the unit circle, hence $Q[x] \triangleq (\mathbf{I} - \gamma B_\lambda)x + \gamma d$ is a contraction. As a result, the iterative algorithm

  $$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma(B_\lambda \mathbf{r}_k - d_\lambda)$$

  converges to the fixed point $\mathbf{r}^* = B_\lambda^{-1} d_\lambda$. Again, we want to develop a simulation based method to do this. We can use this algorithm (TD(λ)):

  $$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma_k z_k q_k,$$

  where we have defined the temporal difference

  $$q_k \triangleq \phi(X_k)^T \mathbf{r}_k - C_k - \alpha\phi(X_{k+1})^T \mathbf{r}_k$$

  and γ_k is now a diminishing step-size, to combat randomness (note that q_k is very noisy across time). To see this, note that

  $$\begin{aligned}\mathbb{E}[z_k q_k | \mathbf{r}] &= \sum_{i,j}\sum_{t=0}^{k}[\lambda\alpha]^{k-t}\phi(i)\xi_i\{[\mathbf{P}^{k-t}]_{i,j}\phi(j)^T\mathbf{r} - [\mathbf{P}^{k-t}]_{i,j}\bar{c}_j - \alpha[\mathbf{P}^{k+1-t}]_{i,j}\phi(j)\mathbf{r}\} \\ &= \sum_{t=0}^{k}[\lambda\alpha]^{k-t}\Phi^T\Xi\mathbf{P}^{k-t}\left[(\mathbf{I} - \alpha\mathbf{P})\Phi^T\mathbf{r} - \bar{c}\right].\end{aligned}$$

  hence taking the limit $k \to \infty$

  $$\begin{aligned}\lim_{k\to\infty}\mathbb{E}[z_k q_k|\mathbf{r}] &= \Phi^T\Xi[\mathbf{I} - \lambda\alpha\mathbf{P}]^{-1}\left[(\mathbf{I} - \alpha\mathbf{P})\Phi^T\mathbf{r} - \bar{c}\right] \\ &= B_\lambda\mathbf{r} - d_\lambda\end{aligned}$$

TD(λ) is typically slower to converge (needs a diminishing step-size), but has very low complexity and require minimal memory requirements: the update requires only to compute the update

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \gamma_k z_k q_k,$$

where q_k is the <u>temporal difference</u>

$$q_k \triangleq \phi(X_k)^T \mathbf{r}_k - C_k - \alpha\phi(X_{k+1})^T \mathbf{r}_k,$$

and z_k is the <u>eligibility vector</u> with $z_0 = \phi(X_0)$,

$$z_{k+1} = \lambda\alpha z_k + \phi(X_{k+1}).$$

- Note that TD(0) is recovered by letting $\lambda = 0$

- It has been shown that there is a bias-variance trade-off as λ is varied. When λ is small, PVI(λ) converges to a solution with larger error (bias), but is overall less noisy; on the other hand, when λ approaches one, PVI(λ) converges to a solution with smaller error (bias), but is overall more noisy. To see the bias of the solution, consider the following theorem:

Theorem 3. *Let $\Pi_\xi(J_\mu)$ be the projection of J_μ into the subspace spanned by Φ, based on the weighted norm with weights ξ (this is the smallest approximation error on J_μ). Let $\mathbf{r}^*_\lambda$ be the fixed point of PVI. Then*

$$\|J_\mu - \Pi_\xi(J_\mu)\|_\xi \leq \|J_\mu - \Phi\mathbf{r}^*_\lambda\|_\xi \leq \frac{1}{\sqrt{1-\alpha_\lambda^2}}\|J_\mu - \Pi_\xi(J_\mu)\|_\xi,$$

where

$$\alpha_\lambda = \frac{(1-\lambda)\alpha}{1-\lambda\alpha}.$$

Proof. Clearly,

$$\|J_\mu - \Pi_\xi(J_\mu)\|_\xi \leq \|J_\mu - \Phi\mathbf{r}^*_\lambda\|_\xi$$

by definition of projection.

Now, consider the second inequality:

$$\|J_\mu - \Phi\mathbf{r}^*_\lambda\|_\xi^2 = \|J_\mu - \Pi_\xi(J_\mu) + \Pi_\xi(J_\mu) - \Phi\mathbf{r}^*_\lambda\|_\xi^2 = \|J_\mu - \Pi_\xi(J_\mu)\|_\xi^2 + \|\Pi_\xi(J_\mu) - \Phi\mathbf{r}^*_\lambda\|_\xi^2$$

$$= \|J_\mu - \Pi_\xi(J_\mu)\|_\xi^2 + \|\Pi_\xi(T^{(\lambda)}(J_\mu)) - \Pi_\xi(T^{(\lambda)}(\Phi\mathbf{r}^*_\lambda))\|_\xi^2 \leq \|J_\mu - \Pi_\xi(J_\mu)\|_\xi^2 + \alpha_\lambda^2\|J_\mu - \Phi\mathbf{r}^*_\lambda\|_\xi^2$$

and therefore

$$\|J_\mu - \Phi\mathbf{r}^*_\lambda\|_\xi^2 \leq \frac{1}{1-\alpha_\lambda^2}\|J_\mu - \Pi_\xi(J_\mu)\|_\xi^2$$

□

Clearly, in the limit $\lambda \to 1$, we obtain $\alpha_\lambda \to 0$, hence $\Phi\mathbf{r}^*_\lambda \to \Pi_\xi(J_\mu)$ (i.e., the solution becomes unbiased with respect to the best approximation as $\lambda \to 1$).

- Other important aspects we are not going to consider in this class (but you might be interested to investigate further):

  - Policy iteration: when embedding these schemes into a policy iteration framework, we encounter some difficulties related to the fact that some states/actions pairs may not be visited sufficiently often; this leads to poor quality of the estimate of J_μ, hence poor convergence properties; we thus need to enhance exploration by appropriate design of exploration strategies

  - Aggregation methods for approximate DP: the idea is to simplify the MDP by considering a smaller MDP obtained by aggregating similar states together. If you are interested to further develop this topic, please refer to Bertsekas Vol. II, Ch. 6.