Dynamic Programming: Major Algorithms

John Stachurski

March 2024

Topics

- Introduction to the problem
- Lifetime value
- Optimality
- Value function iteration
- Howard policy iteration
- Optimistic policy iteration

Set Up

We take as given

1. a finite set X called the **state space** and
2. a finite set A called the **action space**

We study a controller who, at each integer $t \geqslant 0$

1. observes the current state $X_t \in \mathsf{X}$
2. responds with an action $A_t \in \mathsf{A}$

Her aim is to maximize

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t r(X_t, A_t) \quad \text{given } X_0 = x_0$$

Actions restricted by a **feasible correspondence** Γ

- $\Gamma(x)$ is a nonempty subset of A for each $x \in \mathsf{X}$
- interpretation: $\Gamma(x) =$ actions available in state x

Reward $r(x, a)$ is received at feasible state-action pair (x, a)

Let P denote transition probabilities:

$$P(x, a, x') = \text{ prob of transitioning to } x' \text{ given } x, a$$

MDP dynamics:

$t \leftarrow 0$

input X_0

while $t < \infty$ **do**

  observe X_t

  choose action A_t from $\Gamma(X_t)$

  receive reward $r(X_t, A_t)$

  draw X_{t+1} from $P(X_t, A_t, \cdot)$

  $t \leftarrow t + 1$

end

The **Bellman equation** is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x,a,x') \right\}$$

Policies

A **feasible policy** is a map σ from X to A such that

$$\sigma(x) \in \Gamma(x) \text{ for all } x \in \mathsf{X}$$

- Let $\Sigma :=$ the set of all feasible policies

Choosing $\sigma \in \Sigma \implies$

respond to state X_t with action $A_t := \sigma(X_t)$ at <u>all</u> $t \geqslant 0$

If $A_t := \sigma(X_t)$ at all $t \geqslant 0$, then

$$X_{t+1} \sim P(X_t, \sigma(X_t), \cdot) \quad \text{for all } t \geqslant 0$$

Thus, X_t is P_σ-Markov for

$$P_\sigma(x, x') := P(x, \sigma(x), x') \qquad (x, x' \in \mathsf{X})$$

- Fixing a policy "closes the loop" in the state dynamics

Lifetime rewards

Under the policy σ, rewards at x given by $r(x, \sigma(x))$

Let

- $r_\sigma(x) := r(x, \sigma(x))$
- $\mathbb{E}_x := \mathbb{E}[\,\cdot \mid X_0 = x]$

Now

$$\mathbb{E}_x \, r(X_t, A_t) = \mathbb{E}_x \, r_\sigma(X_t) = \sum_{x'} r_\sigma(x') P_\sigma^t(x, x') = (P_\sigma^t \, r_\sigma)(x)$$

The **lifetime value of σ** starting from x is

$$
\begin{aligned}
v_\sigma(x) &:= \mathbb{E}_x \sum_{t \geqslant 0} \beta^t r_\sigma(X_t) \\
&= \sum_{t \geqslant 0} \mathbb{E}_x \left[\beta^t r_\sigma(X_t)\right] \\
&= \sum_{t \geqslant 0} \beta^t (P_\sigma^t \, r_\sigma)(x)
\end{aligned}
$$

By the Neumann (geometric) series lemma,

$$
v_\sigma = \sum_{t \geqslant 0} (\beta P_\sigma)^t \, r_\sigma = (I - \beta P_\sigma)^{-1} \, r_\sigma
$$

Policy Operators

The **policy operator** corresponding to σ is

$$(T_\sigma \, v)(x) = r(x, \sigma(x)) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, \sigma(x), x')$$

In vector notation, we can write

$$T_\sigma \, v = r_\sigma + \beta P_\sigma v$$

- **Fact.** T_σ is a contraction map

Fact. v_σ is the unique fixed point of T_σ in $\mathbb{R}^n$

Proof: Since $\beta < 1$, we have

$$v = T_\sigma v \iff v = r_\sigma + \beta P_\sigma v$$

$$\iff v = (I - \beta P_\sigma)^{-1} r_\sigma$$

$$\iff v = v_\sigma$$

Hence

$$v \text{ is a fixed point of } T_\sigma \iff v = v_\sigma$$

Fact. v_σ is the unique fixed point of T_σ in $\mathbb{R}^n$

<u>Proof</u>: Since $\beta < 1$, we have

$$
\begin{aligned}
v = T_\sigma \, v \iff & \; v = r_\sigma + \beta P_\sigma v \\
\iff & \; v = (I - \beta P_\sigma)^{-1} \, r_\sigma \\
\iff & \; v = v_\sigma
\end{aligned}
$$

Hence

$$
v \text{ is a fixed point of } T_\sigma \quad \iff \quad v = v_\sigma
$$

Greedy Policies

Fix $v \in \mathbb{R}^n$

A policy σ is called v-**greedy** if

$$\sigma(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x'} v(x') P(x,a,x') \right\}$$

for all $x \in \mathsf{X}$

Ex. Prove: at least one v-greedy policy exists in Σ

The Bellman Operator

Recall: the Bellman equation is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x'} v(x') P(x,a,x') \right\}$$

The **Bellman operator** is defined by

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x'} v(x') P(x,a,x') \right\}$$

By construction, $Tv = v \iff v$ satisfies the Bellman equation

Optimality

The **value function** is defined by $v^* := \bigvee_{\sigma \in \Sigma} v_\sigma$

More explicitly,

$$v^*(x) := \max_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X})$$

Thus, $v^*(x) =$ <u>maximal lifetime value</u> from state x

A policy $\sigma \in \Sigma$ is called **optimal** if

$$v_\sigma = v^*$$

Thus, σ is optimal $\iff$ lifetime value is maximal at each state

Theorem. For the DP model described above,

1. v^* is the unique fixed point of T in $\mathbb{R}^n$
2. A feasible policy is optimal if and only it is v^*-greedy
3. At least one optimal policy exists

Remark: Point (2) is called **Bellman's principle of optimality**

Algorithms

Previously we used value function iteration (VFI) to solve optimal stopping problems

Here we

1. present a generalization suitable for arbitrary MDPs
2. introduce two other important methods

The two other methods are called

1. Howard policy iteration (HPI) and
2. Optimistic policy iteration (OPI)

Algorithm 1: VFI for MDPs

input $v_0 \in \mathbb{R}^n$, an initial guess of v^*

input τ, a tolerance level for error

$\varepsilon \leftarrow \tau + 1$

$k \leftarrow 0$

while $\varepsilon > \tau$ **do**

    $v_{k+1} \leftarrow T v_k$

    $\varepsilon \leftarrow \|v_k - v_{k+1}\|_\infty$

    $k \leftarrow k + 1$

end

Compute a v_k-greedy policy σ

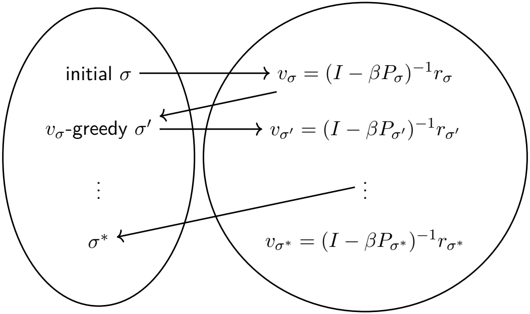
return σ

VFI is

- easy to understand
- easy to implement
- globally convergent
- relatively robust

But convergence can be slow...

Howard Policy Iteration



Iterates between computing the value of a given policy and computing the greedy policy associated with that value

Algorithm 2: Howard policy iteration for MDPs

input $\sigma \in \Sigma$

$v_0 \leftarrow v_\sigma$ and $k \leftarrow 0$

repeat

  $\sigma_k \leftarrow$ a v_k-greedy policy

  $v_{k+1} \leftarrow (I - \beta P_{\sigma_k})^{-1} r_{\sigma_k}$

  if $v_{k+1} = v_k$ **then** break

  $k \leftarrow k + 1$

return σ_k

Proposition. HPI returns an exact optimal policy in a finite number of steps

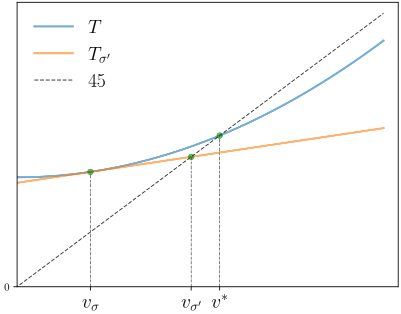
Also, rate of convergence is faster than VFI

In fact HPI is analogous to gradient-based Newton iteration on T

- Details are in the text

In general, for a given fixed point problem,

1. Newton iteration yields a quadratic rate of convergence
2. Successive approximation yields a linear rate of convergence



- σ' is v_σ-greedy if $T_{\sigma'} v_\sigma = T v_\sigma$
- $v_{\sigma'}$ is the fixed point of $T_{\sigma'}$

Optimistic Policy Iteration

OPI is a "convex combination" of VFI and HPI

Similar to HPI except that

- HPI takes current σ and obtains v_σ
- OPI takes current σ and iterates m times with T_σ

Recall that, for any $v \in \mathbb{R}^n$, we have $T_\sigma^m v \to v_\sigma$ as $m \to \infty$

Hence OPI replaces v_σ with an approximation

Algorithm 3: Optimistic policy iteration for MDPs

input $v_0 \in \mathbb{R}^n$, an initial guess of v^*
input τ, a tolerance level for error
input $m \in \mathbb{N}$, a step size
$k \leftarrow 0$
$\varepsilon \leftarrow \tau + 1$
while $\varepsilon > \tau$ **do**
    $\sigma_k \leftarrow$ a v_k-greedy policy
    $v_{k+1} \leftarrow T_{\sigma_k}^m v_k$
    $\varepsilon \leftarrow \|v_k - v_{k+1}\|_\infty$
    $k \leftarrow k + 1$
end
return σ_k

Fact. OPI $=$ VFI when $m = 1$

Proposition. For all values of m we have $v_k \to v^*$

If m is large, OPI is similar to HPI

- because $\lim_{m \to \infty} T_{\sigma_k}^m v_k = v_{\sigma_k}$

Rules of thumb:

- parallelization tends to favor HPI
- OFI is simple and dominates VFI for many values of m
- VFI works well when β is small and optimization is cheap