Lec32

Saturday, April 18, 2015 10:05 AM

- Deterministic SSP: Principle of Optimality:

$$J_k(i) = \min_{j=1,2,\ldots,N} \{ a_{ij} + J_{k+1}(j) \}$$

- Finite horizon stochastic SP

$$J_k(x_k) = \min_{u_k \in U_k(x_k)} E_{w_k} \left[g_k(x_k, u_k, w_k) + J_{k+1}(f_k(x_k, u_k, w_k)) \right]$$

- Infinite-horizon SSP

$$J^*(i) = \min_u \quad g(i,u) + \sum_j P_{ij}(u) J^*(j)$$

- Discounted problems

- Using the SSP mapping

$$J^*(i) = \min_u \{ g(i,u) + \sum_j \underbrace{\alpha P_{ij}(u)}_{\substack{\text{transition} \\ \text{probability} \\ \text{in SSP}}} J^*(j) \}$$

$$\Leftrightarrow J^*(i) = \min_u \{ g(i,u) + \alpha \sum_j P_{ij}(u) J^*(j) \}$$

future cost is discounted

future cost from j

by α

- Another way to look at the Bellman Equation for discounted problem

$$J(i) = \min E\left[g(i,u_1) + \alpha g(j,u_2) + \alpha^2 g(j',u_3) \cdots\right]$$

$$= \min_{u_1} g(i,u_1) + \alpha \cdot \underbrace{\min E\left[g(j,u_2) + \alpha g(j',u_3) + \cdots\right]}_{J(j)}$$

Value iteration versus policy iteration

Sunday, March 29, 2015 9:46 AM

- Computationally, how to find the optimal policy μ?

① Directly solve the Bellman's Equation

- Usually hard for large problems

② Value Iteration

- Take any initial values $J_0(i)$
- Run the DP iteration

$$J_{k+1}(i) = \min_{\mu} \left[g(i, u) + \sum_{j=1}^{n} p_{ij}(u) J_k(j) \right]$$

- Generally requires an infinite number of iterations
  - Same as saying that finite-horizon payoff approaches the infinite-horizon payoff
  - At the speed of ρ^K.

③ Policy Iteration

- Start with any stationary policy μ^0.

- Given μ^k, compute its payoff by solving

$$J(i) = g(i, \mu^k(i)) + \sum_{j=1}^{n} p_{ij}(\mu^k(i)) J(j)$$

  - called "policy evaluation"
  - A linear program of n variables

- Perform "policy-improvement"

$$\mu^{k+1}(i) = \arg\min_{\mu} \left[g(i,u) + \sum_{j=1}^{n} p_{ij}(u) J_{\mu^k}(j) \right]$$

- Stop if $\mu^{k+1} = \mu^k$

- Can show that

$$J_{\mu^{k+1}}(i) \leq J_{\mu^k}(i) \qquad \text{for all } i \text{ \& } k$$

$\Rightarrow$ Policy values always improve

- For finite-state systems, the total number of possible policies is finite

  $\Rightarrow$ must terminate after a finite number of iterations.

  - $\mu^{k+1} = \mu^k$ satisfies Bellman's Equation

$\Rightarrow$ optimal.

Contraction mapping

Tuesday, April 11, 2023 10:06 AM

- For discounted-cost problems (or positive termination prob. in every step), value iteration converges geometrically fast because we can show that the Bellman operator is a contraction mapping

$$J(i),\, i=1,\dots,n$$

$$\mapsto \min_u \left\{ g(i,u) + \alpha \sum_j P_{ij}(u) J(j) \right\}$$

- Denote this mapping by B
  - Let $\vec{v} = [J(i)]_{i=1,\dots,n}$

$$\vec{v} \mapsto B(\vec{v})$$

- To see why B is a contraction, compare $B(\vec{v}_1)$ & $B(\vec{v}_2)$

- We can show that

$$\|B(\vec{v}_1) - B(\vec{v}_2)\|_\infty \le \alpha \|\vec{v}_1 - \vec{v}_2\|_\infty$$

  - Suppose $\|\vec{v}_1 - \vec{v}_2\|_\infty = \Delta$

$$\Rightarrow \quad |J_1(i) - J_2(i)| \le \Delta \quad \text{for all } i$$

  - Thus, for every u

$$\left| \left[g(i,u) + \alpha \sum_j P_{ij}(u) J_1(j) \right] \right.$$

$$\Big| \left[g(i,u) + \alpha \sum_j P_{ij}(u) J_1(j) \right] - \left[g(i,u) + \alpha \sum_j P_{ij}(u) J_2(j) \right] \Big|$$

$$= \alpha \Big| \sum_j P_{ij}(u) \left[J_1(j) - J_2(j) \right] \Big|$$

$$\leq \alpha \Delta$$

$$\Rightarrow \| B(\vec{v}_1) - B(\vec{v}_2) \|_\infty \leq \alpha \Delta$$

Linear programming

Sunday, March 29, 2015 9:57 AM

- Bertsekas P416

- If we start from any vector

$$J_0 = (J_0(1), J_0(2), \cdots J_0(n))$$

Such that

$$J_0(i) < \min_u g(i,u) + \sum_{j=1}^{n} P_{ij}(u) J_0(j), \quad \text{for all } i \qquad (*)$$

- Apply the DP iteration

$$J_{k+1}(i) = \min_u g(i,u) + \sum_{j=1}^{n} P_{ij}(u) J_k(j)$$

- We have seen that $J_k(i) \to J^*(i)$

- Further, we can show that $J_k(i) \le J_{k+1}(i)$ for all k & i.
    - Trivially hold for $k = 0$
    - Induction by k.

- This implies that

$$J_0(i) \le J^*(i)$$

for all $J_0(i)$ that satisfies $(*)$

- In other words, $[J^*(1), \cdots J^*(n)]$ is component-wise larger than any other vector $[J_0(1), \cdots J_0(n)]$ that satisfies $(*)$

- We can then conclude that $(J^*(1) \cdots J^*(n))$ must be the solution to the following linear program:

$$\max \sum_{i=1}^{n} \beta_i J(i) \qquad (\beta_i > 0)$$

$$\text{sub to} \quad J(i) \le g(i,u) + \sum_j P_{ij}(u) J(j)$$

for all u, i.

- # of variables = # of states n
  - Which is smaller compared to the LP using $y^k_{s,a}$.
- # of constraints : $n \times A$.