Aalto University
School of Science

Discounted Problems - Theory

Jessica Norrbäck
Presentation *11*
30.10.2020

MS-E2191 Graduate Seminar on Operations Research
Fall 2020

Content

- Recap – DP Algorithm
- Discounted Finite Horizon DP Algorithm
- Infinite Horizon Problem
- Shorthand Notation & Monotonicity
- Discounted Problems – Bounded Cost Per Stage
- Scheduling and Multiarmed Bandit Problem

Recap – DP Algorithm

The optimal cost $J^*(x_0)$ for state x_0 can be solved by starting with

$$J_N(x_N) = g_N(x_N)$$

and iterating backwards from $N - 1$ to 0, using the DP algorithm:

$$J_k(x_k) = \min_{u_k \in U(x_k)} \mathbb{E}_{w_k}[g_k(x_k, u_k, w_k) + J_{k+1}(f_k(x_k, u_k, w_k))], k = 0, \dots, N - 1$$

Discounted Finite Horizon DP Algorithm (1/2)

We introduce a discount factor $\alpha \epsilon (0,1)$ to account for the time value of money

Suppose we accumulate costs of the first N stages and add a terminal cost $\alpha^N J(x_N)$, where $J: X \to \mathbb{R}$. The total expected cost is

$$\mathbb{E}_{w_k, k=0,1,\dots}\left\{\alpha^N J(x_N) + \sum_{k=0}^{N-1} \alpha^k g(x_k, \mu_k(x_k), w_k)\right\}$$

discount factor $0 < \alpha < 1$

Discounted Finite Horizon DP Algorithm (2/2)

The minimum cost can be calculated by starting with $J_N(x) = \alpha^N J(x)$ and iterating backwards with the DP algorithm

$$J_{N-k}(x) = \min_{u \in U(x)} \mathbb{E}\{\alpha^{N-k} g(x,u,k) + J_{N-k+1}(F(x,u,w))\} \qquad (1)$$

Denoting $V_k = \frac{J_{N-k}(x)}{\alpha^{N-k}}$, we can rewrite (1) as

$$V_{k+1}(x) = \min_{u \in U(x)} \mathbb{E}\{g(x,u,w) + \alpha V_k\left(f(x,u,w)\right)\}$$

Infinite Horizon Problem (1/2)

Given a discrete time dynamic system

$$x_{k+1} = f(x_k, u_k, w_k), \qquad k = 0,1,\ldots$$

where $x_k \in X, u_k \in U$ and $w \sim P(\cdot \mid x_k, u_k)$, we want to find a policy $\pi = \{\mu_0, \mu_1, \ldots\}$ for all $x_k \in X, k = 0,1,\ldots$ that minimizes the cost function

$$J_\pi(x_0) = \lim_{N\to\infty} \mathbb{E}_{w_k, k=0,1,\ldots} \left\{ \sum_{k=0}^{N-1} \alpha^k g(x_k, \mu_k(x_k), w_k) \right\}$$

Infinite Horizon Problem (2/2)

The optimal cost function is defined by

$$J^*(x) = \min_{\pi \in \Pi} J_\pi(x), \qquad x \in X$$

where Π is the set of **admissible policies** π.

- For most problems, the optimal policy is independent of the initial state
- Very often such a policy is **stationary**

$$\pi = \{\mu, \mu, \dots\}$$

Shorthand Notation

Applying DP mapping to $J: X \to \mathbb{R}$, we obtain

$$(TJ)(x) = \min_{u \in U(x)} \mathbb{E}_w \{ g(x, u, w) + \alpha J\big(f(x, u, w)\big) \}, \qquad x \in X$$

$\Rightarrow$ TJ is the optimal cost function for the one-stage problem with cost g and terminal cost αJ.

For any stationary policy μ, we denote

$$\big(T_\mu J\big)(x) = \mathbb{E}\{ g(x, u, w) + \alpha J\big(f(x, u, w)\big) \}, \qquad x \in X$$

Monotonicity

Monotonicity Lemma.

For any functions $J: X \to \mathbb{R}$ and $J': X \to \mathbb{R}$, such that

$$J(x) \leq J'(x), \qquad \forall x \in X$$

and any stationary policy $\mu: X \to U$, it holds that

$$(T^k J)(x) \leq \left(T^k J'\right)(x) \text{ and } (T_\mu^k J)(x) \leq \left(T_\mu^k J'\right)(x), \quad \forall x \in X, k = 1,2,\ldots$$

Preview of Infitinite Horizon Results

We are aiming for the following type of results:

1. **Convergence of DP algorithm**

$$J^*(x) = \lim_{k \to \infty} \left(T^k J\right)(x), \qquad x \in X$$

2. **Bellman's Equation**

$$J^*(x) = \min_{u \in U(x)} E\left\{g(x, u, w) + \alpha J^*\left(f(x, u, w)\right)\right\}, \qquad x \in X$$

$$J^* = TJ^*$$

3. **Characterization of optimal stationary policies**

If $\mu(x)$ attains the minimum in the right-hand side of Bellman's equation, the stationary policy μ is optimal.

Discounted Problems – Bounded Cost Per Stage

The cost per stage g satisfies

$$|g(x,u,w)| \le M, \qquad \forall (x,u,w) \in X \times U \times W,$$

where M is scalar and $\alpha \in]0,1[$.

Convergence of the DP Algorithm. For any bounded function $J: X \to \mathbb{R}$, we have

$$J^*(x) = \lim_{N\to\infty} (T^N J)(x), \qquad \forall x \in X$$

Markov Chain Notation

Transition probabilities are given by

$$p_{ij}(u) = P(x_{k+1} = j | x_k = i, u_k = u), \qquad i, j \in X, u \in U(i)$$

The mapping T in terms of the transition probabilities

$$(TJ)(i) = \min_{u \in U(i)} \sum_{j \in X} p_{ij}(u)\big(g(i,u,j) + \alpha J(j)\big), \qquad i \in X$$

Bellman's equation takes the form

$$J^*(i) = \min_{u \in U(i)} \sum_{j \in X} p_{ij}(u)\big(g(i,u,j) + \alpha J^*(j)\big), \qquad i \in X$$

Application: Machine Replacement

A machine can be in any of n states (1 = perfect condition, …, n = not working). The transition probabilities p_{ij} are given. For operating in state i, there is a cost g(i). In each period, we can either

1) operate the machine one more period in its current state
2) replace the machine with a new machine (state 1 at cost R)

The machine is guaranteed to stay one period in state 1 when repaired, after which it deteriorates to states j with probabilities p_{1j}.

We assume infinite horizon and discount factor $\alpha \in]0,1[$.

Scheduling and Multiarmed Bandit Problem

Suppose we have n projects, of which one can be worked at a time. The state of all other projects remains fixed. If project l is worked on at time k, we receive an expected reward $\alpha^k R^l(x_k^l)$, where $\alpha \in (0,1)$.

The state x_k is worked on at time k, its state evolves according to

$$x_{k+1}^l = f^l(x_k^l, w_k^l)$$

Further, we assume that there is a possibility to retire permanently from all projects at any time k, of which we receive a final reward $\alpha^k M$.

Index Rule

For each project l, there is a function $m^l(x^l)$, such that the optimal policy at time k is to

- Retire, if $M > \max\{m^{\bar{l}}(x^{\bar{l}})\}$
- Work on project l, if $m^l(x_k^l) = \max_{\bar{l}}\{m^{\bar{l}}(x^{\bar{l}})\} \geq M$.

The index rule is an optimal stationary policy.

Index Function

The function

$$m^l(x^l) = \min\{M \mid J^l(x^l, M) = M\}$$

Is called the **index function.**

- Provides indifference threshold at each state

Project-by-Project Retirement Policies

Retirement set:

$$X^l = \{x^l \mid m^l(x^l) < M\}$$

There exists an optimal **project-by project retirement policy** that permanently retires projects in the same way as if they were the only projects available.

- Retire project l, if $x^l \in X^l$
- Work on some project, if $x^j \notin X^j$ for some j.

Deteriorating and Improving Cases

Improving cases:

$$m^l(x^l) \leq m^l(f^l(x^l, w^l))$$

➔ Retire at first period or select project with maximal index at first period and continue working on that project.

Deteriorating cases:

$$m^l(x^l) \geq m^l(f^l(x^l, w^l))$$

➔ Retire if $M > \max_l \frac{R^l(x^l)}{1-\alpha}$, else work on project l with maximal one-step reward $R^l(x^l)$.

Aalto University
School of Science

Thank you!

References

D.P. Bertsekas (2012), Dynamic Programming and Optimal Control, Vol. II, 4th Edition: Approximate Dynamic Programming.

Athena Scientific, Belmont, MA. pp. 3-32

Homework

Recall the proposition from slide 11:

Convergence of the DP Algorithm. For any bounded function $J: X \to \mathbb{R}$, we have

$$J^*(x) = \lim_{N\to\infty} (T^N J)(x), \qquad \forall x \in X$$

The main parts of the proof is given in the Word template. Your task is to fill in the missing parts of the proof.

DL: 6.11.2020

Submission: jessica.norrback@aalto.fi