Stochastic Dynamic Programming

V. Leclère (CERMICS, ENPC)

May 4, 2018

Contents

Contents

Controlled Dynamic System

A controlled dynamic system is defined by its *dynamic*

$$x_{t+1} = f_t(x_t, u_t)$$

and initial state x_0.
The variables

- x_t is the *state* of the system,
- u_t is the *control* applied to the system at time t.

Example :

- x_t is the position and speed of a satellite, u_t the acceleration due to the engine (at time t).
- x_t is the stock of products available, u_t the consumption at time t
- ...

Optimization Problem

We want to solve the following optimization problem

$$\min_{u_0,\dots,u_{T-1}} \quad \sum_{t=0}^{T-1} L_t\big(x_t, u_t\big) + K\big(x_T\big) \tag{1a}$$

$$s.t. \quad x_{t+1} = f_t(x_t, u_t), \qquad x_0 \text{ given} \tag{1b}$$

$$u_t \in U_t(x_t) \tag{1c}$$

Where

- $L_t(x,u)$ is the cost incurred between t and $t+1$ for a starting state x with control u;
- $K(x)$ is the final cost incurred for the final state x;
- f_t is the dynamic of the dynamical system;
- $U_t(x)$ is the set of admissible controls at time t with starting state x.

Note : this is a Shortest Path Problem on an acircuitic directed graph.

Problem decomposition

The problem can be written

$$\min_{u_0} \quad \Big\{ L_0(x_0, u_0) + \min_{u_1,\dots,u_{T-1}} \quad \sum_{t=1}^{T-1} L_t(x_t, u_t) + K(x_T) \Big\}$$

$$\begin{aligned} s.t. \quad & x_{t+1} = f_t(x_t, u_t) \\ & x_1 = f_0(x_0, u_0) \\ & u_t \in U_t(x_t) \end{aligned}$$

Or, more simply,

$$\min_{u_0} L_0(x_0, u_0) + V_1\big(f_0(x_0, u_0)\big)$$

where $V_1(x)$ is the value of the problem starting at time $t = 1$ with state $x_1 = x$.

Bellman value function

More generically, we denote $V_{t_0}(x)$ the optimal value of the problem starting at time t with state x:

$$
\begin{aligned}
V_{t_0}(x) = \min_{u_{t_0},\dots,u_{T-1}} \quad & \sum_{t=t_0}^{T-1} L_t\big(x_t, u_t\big) + K\big(x_T\big) && \text{(2a)} \\
s.t. \quad & x_{t+1} = f_t(x_t, u_t), \qquad x_{t_0} = x && \text{(2b)} \\
& u_t \in U_t(x_t) && \text{(2c)}
\end{aligned}
$$

Bellman Equation

Theorem

We have the Bellman equation (we assume existence of minimizers)

$$
\begin{aligned}
V_T(x) &= K(x) && \forall x \in \mathbb{X}_T \\
V_t(x) &= \min_{u_t \in U_t(x)} L_t(x, u_t) + V_{t+1} \circ \underbrace{f_t(x, u_t)}_{x_{t+1}} && \forall x \in \mathbb{X}_t.
\end{aligned}
$$

And the optimal policy is given by

$$
\pi_t^\sharp(x) \in \operatorname*{arg\,min}_{u_t \in U_t(x)} \Big\{ L_t(x, u_t) + V_{t+1} \circ \underbrace{f_t(x, u_t)}_{x_{t+1}} \Big\} \qquad \forall x \in \mathbb{X}_t.
$$

Policy

Definition

An admissible policy for problem (1) is a sequence of function π_t mapping the set $\mathbb{X}_t$ of possible state at time t into the set $\mathbb{U}_t$ of possible controls and such that

$$\forall t \in [\![0, T-1]\!], \quad \forall x \in \mathbb{X}_t, \qquad \pi_t(x) \in U_t(x).$$

Contents

Stochastic Controlled Dynamic System

A stochastic controlled dynamic system is defined by its dynamic

$$\boldsymbol{x}_{t+1} = f_t(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_{t+1})$$

and initial state

$$\boldsymbol{x}_0 = x_0$$

The variables

- $\boldsymbol{x}_t$ is the state of the system,
- $\boldsymbol{u}_t$ is the control applied to the system at time t,
- $\boldsymbol{\xi}_t$ is an exogeneous noise.

Examples

- Stock of water in a dam:
  - $\boldsymbol{x}_t$ is the amount of water in the dam at time t,
  - $\boldsymbol{u}_t$ is the amount of water turbined at time t,
  - $\boldsymbol{\xi}_t$ is the inflow of water at time t.
- Boat in the ocean:
  - $\boldsymbol{x}_t$ is the position of the boat at time t,
  - $\boldsymbol{u}_t$ is the direction and speed chosen at time t,
  - $\boldsymbol{\xi}_t$ is the wind and current at time t.
- Subway network:
  - $\boldsymbol{x}_t$ is the position and speed of each train at time t,
  - $\boldsymbol{u}_t$ is the acceleration chosen at time t,
  - $\boldsymbol{\xi}_t$ is the delay due to passengers and incident on the network at time t.

Optimization Problem

We want to solve the following optimization problem

$$\min \quad \mathbb{E}\Big[\sum_{t=0}^{T-1} L_t\big(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_{t+1}\big) + K\big(\boldsymbol{x}_T\big)\Big] \tag{3a}$$

$$s.t. \quad \boldsymbol{x}_{t+1} = f_t(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_{t+1}), \qquad \boldsymbol{x}_0 = x_0 \tag{3b}$$

$$\boldsymbol{u}_t \in U_t(\boldsymbol{x}_t) \tag{3c}$$

$$\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t := \sigma\big(\boldsymbol{\xi}_0, \cdots, \boldsymbol{\xi}_t\big) \tag{3d}$$

Where

- constraint (3b) is the dynamic of the system ;
- constraint (3c) refer to the constraint on the controls;
- constraint (3d) is the information constraint : $\boldsymbol{u}_t$ is choosen knowing the realisation of the noises $\boldsymbol{\xi}_0, \dots, \boldsymbol{\xi}_t$ but without knowing the realisation of the noises $\boldsymbol{\xi}_{t+1}, \dots, \boldsymbol{\xi}_{T-1}$.

Dynamic Programming Principle

Theorem

Assume that the noises $\boldsymbol{\xi}_t$ *are independent and exogeneous. Then, there exists (under technical assumptions satisfied in the discrete case) an optimal solution, called a strategy, of the form* $\boldsymbol{u}_t = \pi_t(\boldsymbol{x}_t)$.

We have

$$\pi_t(x) \in \arg\min_{u \in U_t(x)} \mathbb{E}\Big[\underbrace{L_t(x, u, \boldsymbol{\xi}_{t+1})}_{\text{current cost}} + \underbrace{V_{t+1} \circ f_t(x, u, \boldsymbol{\xi}_{t+1})}_{\text{future costs}}\Big] ,$$

where (Dynamic Programming Equation)

$$\begin{cases} V_T(x) &= K(x) \\ V_t(x) &= \min_{u \in U_t(x)} \mathbb{E}\Big[L_t(x, u, \boldsymbol{\xi}_{t+1}) + V_{t+1} \circ \underbrace{f_t(x, u, \boldsymbol{\xi}_{t+1})}_{``\boldsymbol{x}_{t+1}"}\Big] \end{cases}$$

Dynamic Programming Principle

Theorem

Assume that the noises $\boldsymbol{\xi}_t$ are independent and exogeneous. Then, there exists (under technical assumptions satisfied in the discrete case) an optimal solution, called a strategy, of the form $\boldsymbol{u}_t = \pi_t(\boldsymbol{x}_t)$.
We have

$$\pi_t(x) \in \arg\min_{u \in U_t(x)} \mathbb{E}\Big[\underbrace{L_t(x, u, \boldsymbol{\xi}_{t+1})}_{\text{current cost}} + \underbrace{V_{t+1} \circ f_t\big(x, u, \boldsymbol{\xi}_{t+1}\big)}_{\text{future costs}} \Big] ,$$

where (Dynamic Programming Equation)

$$\begin{cases} V_T(x) & = K(x) \\ V_t(x) & = \min_{u \in U_t(x)} \mathbb{E}\Big[L_t(x, u, \boldsymbol{\xi}_{t+1}) + V_{t+1} \circ \underbrace{f_t\big(x, u, \boldsymbol{\xi}_{t+1}\big)}_{\text{"}\boldsymbol{X}_{t+1}\text{"}} \Big] \end{cases}$$

Interpretation of Bellman Value

The Bellman's value function $V_{t_0}(x)$ can be interpreted as the value of the problem starting at time t_0 from the state x. More precisely we have

$$
\begin{aligned}
V_{t_0}(x) = \min \quad & \mathbb{E}\Big[\sum_{t=t_0}^{T-1} L_t\big(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_{t+1}\big) + K\big(\boldsymbol{x}_T\big)\Big] \\
s.t. \quad & \boldsymbol{x}_{t+1} = f_t(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_{t+1}), \qquad \boldsymbol{x}_{t_0} = x \\
& \boldsymbol{u}_t \in U_t(\boldsymbol{x}_t) \\
& \sigma(\boldsymbol{u}_t) \subset \sigma\big(\boldsymbol{\xi}_0, \cdots, \boldsymbol{\xi}_t\big)
\end{aligned}
$$

Introducing Bellman's operator

We rewrite Bellman's equation as

$$\begin{cases} V_T &= K, \\ V_t &= \mathcal{T}_t(V_{t+1}) \end{cases}$$

where

$$\mathcal{T}_t(A) : x \mapsto \min_{u \in U_t(x)} \mathbb{E}\Big[\big\{L_t(x, u, \xi_{t+1}) + A \circ f_t(x, u, \xi_{t+1})\big\}\Big]$$

Indeed, an optimal policy is given by

$$\pi_t(x) \in \arg\min \mathcal{T}_t(V_{t+1})(x) .$$

Introducing Bellman's operator

We rewrite Bellman's equation as

$$\begin{cases} V_T &= K, \\ V_t &= \mathcal{T}_t(V_{t+1}) \end{cases}$$

where

$$\mathcal{T}_t(A) : x \mapsto \min_{u \in U_t(x)} \mathbb{E}\Big[\big\{L_t(x, u, \xi_{t+1}) + A \circ f_t(x, u, \xi_{t+1})\big\}\Big]$$

Indeed, an optimal policy is given by

$$\pi_t(x) \in \arg\min \mathcal{T}_t(V_{t+1})(x) \ .$$

Induced policy

- If you have an estimation of future costs $\left\{\tilde{V}_t\right\}_{t\in[\![1,T]\!]}$, you can deduce an admissible policy given by

$$\pi_t^{\tilde{V}}(x) \in \arg\min \mathcal{T}_t(\tilde{V}_{t+1})(x) \ .$$

- If $\tilde{V}_{t+1} = V_{t+1}$ then the induced policy is optimal.
- The induced policy, like any admissible policy, produce admissible trajectories

$$\boldsymbol{X}_{t+1}^{\tilde{V}} = f_t\Big(\boldsymbol{X}_t^{\tilde{V}}, \pi_t^{\tilde{V}}(\boldsymbol{X}_t^{\tilde{V}}), \boldsymbol{\xi}_{t+1}\Big) \ .$$

- Any admissible strategy yields an upper bound of the problem:

$$V_0(x_0) \leq \mathbb{E}\Big[\sum_{t=0}^{T-1} L_t\Big(\boldsymbol{X}_t^{\tilde{V}}, \pi_t^{\tilde{V}}(\boldsymbol{X}_t^{\tilde{V}}), \boldsymbol{\xi}_{t+1}\Big) + K(\boldsymbol{X}_T^{\tilde{V}})\Big]$$

Monte-Carlo

- Let $\{\boldsymbol{C}_i\}_{i\in\mathbb{N}}$ be a sequence of identically distributed random variables with finite variance.
- Then the Central Limit Theorem ensure that

$$\sqrt{n}\Big(\frac{\sum_{i=1}^n \boldsymbol{C}_i}{n} - \mathbb{E}[\boldsymbol{C}_1]\Big) \Longrightarrow G \sim \mathcal{N}(0, Var[\boldsymbol{C}_1])\ ,$$

  where the convergence is in law.
- In practice it is often used in the following way. Asymptotically,

$$\mathbb{P}\Big(\mathbb{E}\big[C_1\big] \in \Big[\bar{\boldsymbol{C}}_n - \frac{1.96\boldsymbol{\sigma}_n}{\sqrt{n}}, \bar{\boldsymbol{C}}_n + \frac{1.96\boldsymbol{\sigma}_n}{\sqrt{n}}\Big]\Big) \simeq 95\%\ ,$$

  where $\bar{\boldsymbol{C}}_n = \frac{\sum_{i=1}^n \boldsymbol{C}_i}{n}$ is the empirical mean and $\boldsymbol{\sigma}_n = \sqrt{\frac{\sum_{i=1}^n (\boldsymbol{C}_i - \bar{\boldsymbol{C}}_n)^2}{n-1}}$ the empirical standard semi-deviation.

Monte-Carlo II

- Computing the upper-bound, given a policy π (or an estimate of the value function) is numerically challenging.
- We can estimate the upper-bound through Monte-Carlo:
  - Draw a large number $N_{MC} \geq 1000$ of scenario $(\xi_1^k, \dots, \xi_T^k)$
  - For each scenario, iteratively compute a trajectory of state and control (X_t^k, U_t^k) along this scenario with $X_0^k = x_0$ and

$$\begin{cases} U_t^k & = \pi_t(X_t^k) \\ X_{t+1}^k & = f_t(X_t^k, U_t^k, \xi_{t+1}^k) \end{cases}$$

  - Compute the cost of the strategy along this scenario

$$C_k^\pi = \sum_{t=0}^{T-1} L_t(X_t^k, U_t^k, \xi_{t+1}^k) + K(X_T^k)$$

  - Estimate the upper-bound with $\mathbb{E}\left[\boldsymbol{C}^\pi\right] \approx \frac{1}{N_{MC}} \sum_{k=1}^{N_{MC}} C_k$

Information structure I

In Problem (3), constraint (3d) is the information constraint. There are different possible information structure.

- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_0$, the problem is open-loop, as the controls are choosen without knowledge of the realisation of any noise.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t$, the problem is said to be in decision-hazard structure as decision $\boldsymbol{u}_t$ is chosen without knowing $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{t+1}$, the problem is said to be in hazard-decision structure as decision $\boldsymbol{u}_t$ is chosen with knowledge of $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{T-1}$, the problem is said to be anticipative as decision $\boldsymbol{u}_t$ is chosen with knowledge of all the noises.

Information structure I

In Problem (3), constraint (3d) is the information constraint. There are different possible information structure.

- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_0$, the problem is open-loop, as the controls are choosen without knowledge of the realisation of any noise.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t$, the problem is said to be in decision-hazard structure as decision $\boldsymbol{u}_t$ is chosen without knowing $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{t+1}$, the problem is said to be in hazard-decision structure as decision $\boldsymbol{u}_t$ is chosen with knowledge of $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{T-1}$, the problem is said to be anticipative as decision $\boldsymbol{u}_t$ is chosen with knowledge of all the noises.

Information structure I

In Problem (3), constraint (3d) is the information constraint. There are different possible information structure.

- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_0$, the problem is open-loop, as the controls are choosen without knowledge of the realisation of any noise.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t$, the problem is said to be in decision-hazard structure as decision $\boldsymbol{u}_t$ is chosen without knowing ξ_{t+1}.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{t+1}$, the problem is said to be in hazard-decision structure as decision $\boldsymbol{u}_t$ is chosen with knowledge of ξ_{t+1}.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{T-1}$, the problem is said to be anticipative as decision $\boldsymbol{u}_t$ is chosen with knowledge of all the noises.

Information structure I

In Problem (3), constraint (3d) is the information constraint. There are different possible information structure.

- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_0$, the problem is open-loop, as the controls are choosen without knowledge of the realisation of any noise.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t$, the problem is said to be in decision-hazard structure as decision $\boldsymbol{u}_t$ is chosen without knowing $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{t+1}$, the problem is said to be in hazard-decision structure as decision $\boldsymbol{u}_t$ is chosen with knowledge of $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{T-1}$, the problem is said to be anticipative as decision $\boldsymbol{u}_t$ is chosen with knowledge of all the noises.

Information structure I

In Problem (3), constraint (3d) is the information constraint. There are different possible information structure.

- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_0$, the problem is open-loop, as the controls are choosen without knowledge of the realisation of any noise.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_t$, the problem is said to be in decision-hazard structure as decision $\boldsymbol{u}_t$ is chosen without knowing $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{t+1}$, the problem is said to be in hazard-decision structure as decision $\boldsymbol{u}_t$ is chosen with knowledge of $\boldsymbol{\xi}_{t+1}$.
- If constraint (3d) reads $\sigma(\boldsymbol{u}_t) \subset \mathcal{F}_{T-1}$, the problem is said to be anticipative as decision $\boldsymbol{u}_t$ is chosen with knowledge of all the noises.

Information structure II

Be careful when modeling your information structure:

- Open-loop information structure might happen in practice (you have to decide on a planning and stick to it). If the problem does not require an open-loop solution then it might be largely suboptimal (imagine driving a car eyes closed...). In any case it yields an upper-bound of the problem.
- In some cases decision-hazard and hazard-decision are both approximations of the reality. Hazard-decision yield a lower value then decision-hazard.
- Anticipative structure is never an accurate modelization of the reality. However it can yield a lower-bound of your optimization problem relying on deterministic optimization and Monte-Carlo.

Information structure II

Be careful when modeling your information structure:

- Open-loop information structure might happen in practice (you have to decide on a planning and stick to it). If the problem does not require an open-loop solution then it might be largely suboptimal (imagine driving a car eyes closed...). In any case it yields an upper-bound of the problem.
- In some cases decision-hazard and hazard-decision are both approximations of the reality. Hazard-decision yield a lower value then decision-hazard.
- Anticipative structure is never an accurate modelization of the reality. However it can yield a lower-bound of your optimization problem relying on deterministic optimization and Monte-Carlo.

Information structure II

Be careful when modeling your information structure:

- Open-loop information structure might happen in practice (you have to decide on a planning and stick to it). If the problem does not require an open-loop solution then it might be largely suboptimal (imagine driving a car eyes closed...). In any case it yields an upper-bound of the problem.
- In some cases decision-hazard and hazard-decision are both approximations of the reality. Hazard-decision yield a lower value then decision-hazard.
- Anticipative structure is never an accurate modelization of the reality. However it can yield a lower-bound of your optimization problem relying on deterministic optimization and Monte-Carlo.

Non-independence of noise in DP

- The Dynamic Programming equation requires only the time-independence of noises.
- This can be relaxed if we consider an extended state.
- Consider a dynamic system driven by an equation

$$\boldsymbol{y}_{t+1} = f_t(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\varepsilon}_{t+1})$$

  where the random noise $\boldsymbol{\varepsilon}_t$ is an AR1 process :

$$\boldsymbol{\varepsilon}_t = \alpha_t \boldsymbol{\varepsilon}_{t-1} + \beta_t + \boldsymbol{\xi}_t,$$

  $\{\boldsymbol{\xi}_t\}_{t\in\mathbb{Z}}$ being independent.
- Then $\boldsymbol{y}_t$ is called the physical state of the system and DP can be used with the information state $\boldsymbol{x}_t = (\boldsymbol{y}_t, \boldsymbol{\varepsilon}_{t-1})$.
- Generically speaking, if the noise $\boldsymbol{\xi}_t$ is exogeneous (not affected by decisions $\boldsymbol{u}_t$), then we can always apply Dynamic Programming with the state

$$(\boldsymbol{x}_t, \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_t).$$

Contents

Dynamic Programming Algorithm

Data: Problem parameters
Result: optimal control and value;
$V_T \equiv K$;
for $t : T-1 \to 0$ **do**
  for $x \in \mathbb{X}_t$ **do**
    $V_t(x) = \infty$;
    for $u \in U_t(x)$ **do**
      $v_u = \mathbb{E}\Big[L_t(x, u, \xi_{t+1}) + V_{t+1} \circ f_t\big(x, u, \xi_{t+1}\big)\Big]$;
      if $v_u < V_t(x)$ **then**
        $V_t(x) = v_u$;
        $\pi_t(x) = u$;

Algorithm 1: Dynamic Programming Algorithm (discrete case)

Number of flops: $O(T \times |\mathbb{X}_t| \times |\mathbb{U}_t| \times |\Xi_t|)$.

3 curses of dimensionality

1. State. If we consider 3 independent states each taking 10 values, then $|\mathbb{X}_t| = 10^3 = 1000$. In practice DP is not applicable for states of dimension more than 5.
2. Decision. The decision are often vector decisions, that is a number of independent decision, hence leading to huge $|U_t(x)|$.
3. Expectation. In practice random information came from large data set. Without a proper statistical treatment computing an expectation is costly. Monte-Carlo approach are costly too, and unprecise.

Numerical considerations

- The DP equation holds in (almost) any case.
- The algorithm shown before compute a look-up table of control for every possible state *offline*. It is impossible to do if the state is (partly) continuous.
- Alternatively, we can focus on computing *offline* an approximation $\tilde{V}_t$ of the value function V_t and derive the optimal control *online* by solving a one-step problem, solved only at the current state :

$$\pi_t(x) \in \operatorname*{arg\,min}_{u \in U_t(x)} \mathbb{E}\Big[L_t(x, u, \xi_{t+1}) + V_{t+1} \circ f_t\big(x, u, \xi_{t+1}\big)\Big]$$

- The field of Approximate DP gives methods for computing those approximate value function (decomposed on a base of functions).
- The simpler one consisting in discretizing the state, and then interpolating the value function.