Certainty Equivalence Control in Restless Bandits

Implications and Extensions

Chen Yan[1,3]
Joined work with Nicolas Gast[1], Bruno Gaujal[1], Alexandre Reiffer-Masson [2]

[1]Inria Grenbole [2]IMT Atlantique Brest [3]INRAE Avignon

November 21, 2023

Joint work with

Nicolas Gast
(Inria Grenoble)

Bruno Gaujal
(Inria Grenoble)

Alexandre
Reiffer-Masson
(IMT Atlantique)

Motivation

Background: Refined Mean Field Approximation

Mean field approximation: $\mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|^2\right] = \mathcal{O}(1/N)$

Background: Refined Mean Field Approximation

Mean field approximation: $\mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|^2\right] = \mathcal{O}(1/N)$

$\Rightarrow \mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|\right] = \mathcal{O}(1/\sqrt{N})$

Background: Refined Mean Field Approximation

Mean field approximation: $\mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|^2\right] = \mathcal{O}(1/N)$

$\Rightarrow \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$

However...

$$\left\|\mathbb{E}\left[\mathbf{X}^{(N)}\right] - \mathbf{x}^*\right\| \leq \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$$

Background: Refined Mean Field Approximation

Mean field approximation: $\mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|^2\right] = \mathcal{O}(1/N)$

$\Rightarrow \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$

However...

$$\left\|\mathbb{E}\left[\mathbf{X}^{(N)}\right] - \mathbf{x}^*\right\| \leq \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$$

Smoother Drift for More Precise Mean Field Approximations

[a] If the drift of the mean field system is smooth enough, then

$$\mathbb{E}\left[\mathbf{X}^{(N)}\right] = \mathbf{x}^* + \frac{C_1}{N} + \frac{C_2}{N^2} + \cdots + \frac{C_k}{N^k} + \ldots$$

[a] Gast, "Expected Values Estimated via Mean-Field Approximation are 1/N-Accurate"; Gast and Van Houdt, "A Refined Mean Field Approximation"

Background: Refined Mean Field Approximation

Mean field approximation: $\mathbb{E}\left[\|\mathbf{X}^{(N)} - \mathbf{x}^*\|^2\right] = \mathcal{O}(1/N)$

$\Rightarrow \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$

However...

$$\left\|\mathbb{E}\left[\mathbf{X}^{(N)}\right] - \mathbf{x}^*\right\| \leq \mathbb{E}\left[\left\|\mathbf{X}^{(N)} - \mathbf{x}^*\right\|\right] = \mathcal{O}(1/\sqrt{N})$$

Smoother Drift for More Precise Mean Field Approximations

[a] If the drift of the mean field system is smooth enough, then

$$\mathbb{E}\left[\mathbf{X}^{(N)}\right] = \mathbf{x}^* + \frac{C_1}{N} + \frac{C_2}{N^2} + \cdots + \frac{C_k}{N^k} + \dots$$

[a] Gast, “Expected Values Estimated via Mean-Field Approximation are 1/N-Accurate”; Gast and Van Houdt, “A Refined Mean Field Approximation”

♠ Can we incorporate *control* into this framework?

(Re)Formulate the RB and the WCMDP

Restless Bandit: A Single Arm

A single arm of the RB is a Markov decision process consists of:

- state $= \{1 \ldots S\}$ $\rightsquigarrow$ notation s
- action $= \{\text{pull}, \text{not pull}\}$ $\rightsquigarrow$ notation a
- two transition Probability matrices corresponding to the two actions $\rightsquigarrow$ notation P with entries $P^a_{ss'}$
- reward $\rightsquigarrow$ notation r with entries $r(s, a)$

Restless Bandit: A Single Arm

A single arm of the RB is a Markov decision process consists of:

- state $= \{1 \dots S\} \rightsquigarrow$ notation s
- action $= \{\text{pull}, \text{not pull}\} \rightsquigarrow$ notation a
- two transition Probability matrices corresponding to the two actions $\rightsquigarrow$ notation P with entries $P^a_{ss'}$
- reward $\rightsquigarrow$ notation r with entries $r(s, a)$

Maximize the total expected reward on the single arm over a finite horizon T

$$\max_{\Pi : s \to a} \quad \mathbb{E}_{\Pi}\left[\sum_{t=1}^{T} r(s(t), a(t))\right] \tag{1a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s(t+1) = s' \mid s(t) = s, a(t) = a\right) = P^a_{ss'}, \tag{1b}$$

$$a(t) \in \{0, 1\}, \tag{1c}$$

$$s(1) \text{ is given} \tag{1d}$$

Restless Bandit: N Arms

Consider a collection of N such arms, each evolves *independently*:

Restless Bandit: N Arms

Consider a collection of N such arms, each evolves *independently*:

Maximize the total expected reward over these N arms

Restless Bandit: N Arms

Consider a collection of N such arms, each evolves *independently*:

Maximize the total expected reward over these N arms

$$\max_{\Pi : \mathbf{s} \to \mathbf{a}} \quad \mathbb{E}_{\Pi}\left[\sum_{n=1}^{N}\sum_{t=1}^{T} r(s_n(t), a_n(t))\right] \tag{2a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s_n(t+1) = s'_n \mid s_n(t) = s, a_n(t) = a\right) = P^a_{ss'}, \tag{2b}$$

$$\mathbf{a}(t) \in \{0,1\}^N, \tag{2c}$$

$$\mathbf{s}(1) \text{ is given} \tag{2d}$$

Vector notation: $\mathbf{a}(t) = (a_1(t), \ldots, a_N(t))$ and $\mathbf{s}(t) = (s_1(t), \ldots, s_N(t))$

Restless Bandit: N Arms with Constraint

Restless Bandit Problem Formulation

$$\max_{\Pi : \mathbf{s} \to \mathbf{a}} \quad \mathbb{E}_{\Pi}\Big[\sum_{n=1}^{N}\sum_{t=1}^{T} r(s_n(t), a_n(t))\Big] \tag{3a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s_n(t+1) = s'_n \mid s_n(t) = s, a_n(t) = a\right) = P^a_{ss'}, \tag{3b}$$

$$, \ \mathbf{a}(t) \in \{0,1\}^N, \tag{3c}$$

$$\mathbf{s}(1) \text{ is given} \tag{3d}$$

[1] Papadimitriou and Tsitsiklis, "The Complexity Of Optimal Queuing Network Control"

Restless Bandit: N Arms with Constraint

Constraint: *exactly* αN arms be pulled at each time ($0 < \alpha < 1$)

Restless Bandit Problem Formulation

$$\max_{\Pi : \mathbf{s} \to \mathbf{a}} \quad \mathbb{E}_{\Pi}\Big[\sum_{n=1}^{N}\sum_{t=1}^{T} r(s_n(t), a_n(t))\Big] \tag{3a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s_n(t+1) = s_n' \mid s_n(t) = s, a_n(t) = a\right) = P_{ss'}^{a}, \tag{3b}$$

$$, \ \mathbf{a}(t) \in \{0, 1\}^N, \tag{3c}$$

$$\mathbf{s}(1) \text{ is given} \tag{3d}$$

[1] Papadimitriou and Tsitsiklis, "The Complexity Of Optimal Queuing Network Control"

Restless Bandit: N Arms with Constraint

Constraint: *exactly* αN arms be pulled at each time ($0 < \alpha < 1$)

Restless Bandit Problem Formulation

$$\max_{\Pi : \mathbf{s} \to \mathbf{a}} \quad \mathbb{E}_{\Pi}\Big[\sum_{n=1}^{N}\sum_{t=1}^{T} r(s_n(t), a_n(t))\Big] \tag{3a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s_n(t+1) = s_n' \mid s_n(t) = s, a_n(t) = a\right) = P_{ss'}^{a}, \tag{3b}$$

$$\mathbf{a}(t) \cdot \mathbf{1}^\top = \alpha N, \ \mathbf{a}(t) \in \{0,1\}^N, \tag{3c}$$

$$\mathbf{s}(1) \text{ is given} \tag{3d}$$

[1] Papadimitriou and Tsitsiklis, “The Complexity Of Optimal Queuing Network Control”

Restless Bandit: N Arms with Constraint

Constraint: *exactly* αN arms be pulled at each time ($0 < \alpha < 1$)

Restless Bandit Problem Formulation

$$\max_{\Pi:\mathbf{s}\to\mathbf{a}} \quad \mathbb{E}_{\Pi}\Big[\sum_{n=1}^{N}\sum_{t=1}^{T} r(s_n(t), a_n(t))\Big] \tag{3a}$$

$$\text{s.t.} \quad \mathbb{P}\left(s_n(t+1) = s'_n \mid s_n(t) = s, a_n(t) = a\right) = P^{a}_{ss'}, \tag{3b}$$

$$\mathbf{a}(t)\cdot\mathbf{1}^{\top} = \alpha N, \ \mathbf{a}(t) \in \{0,1\}^{N}, \tag{3c}$$

$$\mathbf{s}(1) \text{ is given} \tag{3d}$$

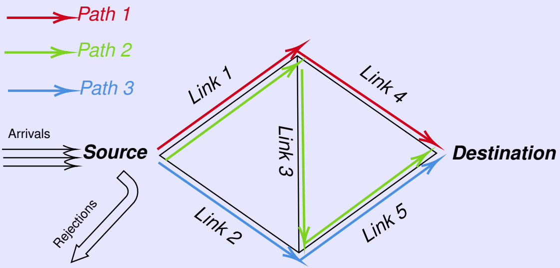
♣ Adding a single constraint renders the problem extremely hard to solve exactly [1]

[1] Papadimitriou and Tsitsiklis, “The Complexity Of Optimal Queuing Network Control”

Adding More Constraints and More Actions: WCMDP

Network Routing to Maximize Utility

- Routing three types of arrival flows from Source to Destination via the three paths
- A link may be occupied by multiple paths and has a maximal capacity ⇒ multiple constraints appear naturally



♣ Can be modeled into a *weakly coupled Markov decision process (WCMDP)* [2] [3]

[2] Adelman and Mersereau, "Relaxations of weakly coupled stochastic dynamic programs"

[3] See Yan and Reiffers-Masson, "Certainty Equivalence Control-Based Heuristics in Multi-Stage Convex Stochastic Optimization Problems" for a study of this example

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s
$\mathbf{U}^{(N)} \in \mathbb{R}^S$: ... $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s
$\mathbf{U}^{(N)} \in \mathbb{R}^S$: ... $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Occupation Measure Formulation

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s
$\mathbf{U}^{(N)} \in \mathbb{R}^S$: ... $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Occupation Measure Formulation

$$\max_{\Pi : \mathbf{X}^{(N)} \to \mathbf{U}^{(N)}} \quad \mathbb{E}_\Pi\left[\sum_{t=1}^{T} \mathbf{r} \cdot \left(\mathbf{X}^{(N)}(t) - \mathbf{U}^{(N)}(t), \mathbf{U}^{(N)}(t)\right)\right] \tag{4a}$$

$$\text{s.t.} \quad \text{Markov evolution of each arm given } \mathbf{U}^{(N)}(t), \tag{4b}$$

$$\mathbf{U}^{(N)}(t) \cdot \mathbf{1}^\top = \alpha, \; \mathbf{0} \le \mathbf{U}^{(N)}(t) \le \mathbf{X}^{(N)}(t), \tag{4c}$$

$$\mathbf{X}^{(N)}(1) \text{ is given} \tag{4d}$$

Π consists of T maps $\pi_t : \mathbf{X}^{(N)}(t) \mapsto \mathbf{U}^{(N)}(t)$ that are

- $\mathcal{F}_t$-measurable
- feasible: $\mathbf{U}^{(N)}(t) \in \mathcal{U}(\mathbf{X}^{(N)}(t)) := \{\mathbf{u} \mid \mathbf{u} \cdot \mathbf{1}^\top = \alpha, \; \mathbf{0} \le \mathbf{u} \le \mathbf{X}^{(N)}(t)\}$

Restless Bandit: Markov Evolution

Example ($N = 5$, $S = 2$)

At $t = 1$, we have 2 arms in state ① and 3 arms in state ②, so that $\mathbf{X}^{(N)}(1) = (\frac{2}{5}, \frac{3}{5})$.
Suppose that $\mathbf{U}^{(N)}(1) = (\frac{1}{5}, \frac{2}{5})$. And

$$\mathbf{P}^0 = \begin{pmatrix} .2 & .8 \\ .4 & .6 \end{pmatrix} \quad \mathbf{P}^1 = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}$$

Restless Bandit: Markov Evolution

Example ($N = 5$, $S = 2$)

At $t = 1$, we have 2 arms in state ① and 3 arms in state ②, so that $\mathbf{X}^{(N)}(1) = (\frac{2}{5}, \frac{3}{5})$.
Suppose that $\mathbf{U}^{(N)}(1) = (\frac{1}{5}, \frac{2}{5})$. And

$$\mathbf{P}^0 = \begin{pmatrix} .2 & .8 \\ .4 & .6 \end{pmatrix} \quad \mathbf{P}^1 = \begin{pmatrix} .5 & .5 \\ .7 & .3 \end{pmatrix}$$

The law of $\mathbf{X}^{(N)}(2)$, given $\mathbf{X}^{(N)}(1)$ and $\mathbf{U}^{(N)}(1)$, is a sum of 5 independent *categorical distributions*, divided by 5:

$$\mathbf{X}^{(N)}(2) \mid \mathbf{X}^{(N)}(1), \mathbf{U}^{(N)}(1) \sim \frac{1}{5}\Big(\text{Categorical}(.2, .8) + \text{Categorical}(.5, .5) + \underbrace{\text{Categorical}(.7, .3) + \text{Categorical}(.7, .3)}_{\text{Multinomial}(2;\ .7, .3)} + \text{Categorical}(.4, .6)\Big)$$

Restless Bandit: Markov Evolution

Markov Evolution of the Occupation Measure

[a] Given $\mathbf{X}^{(N)}(t)$ and $\mathbf{U}^{(N)}(t)$, we can write:

$$\mathbf{X}^{(N)}(t+1) = \phi(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)) + \mathcal{E}(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t))$$

where $\phi(\cdot,\cdot)$ is a deterministic affine function, and $\mathcal{E}(\cdot,\cdot)$ is a random vector satisfying

$$\mathbb{E}\left[\mathcal{E}(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)) \mid \mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)\right] = \mathbf{0}$$

$$\mathrm{var}\left[\mathcal{E}(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)) \mid \mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)\right] = \mathcal{O}(\frac{1}{N})$$

[a] Gast, Gaujal, and Yan, “The LP-update policy for weakly coupled Markov decision processes”, Lemma 1

♢ For large N, the occupation measure’s evolution behaves almost like a deterministic system

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s
$\mathbf{U}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Occupation Measure Formulation

$$\max_{\Pi : \mathbf{X}^{(N)} \to \mathbf{U}^{(N)}} \quad \mathbb{E}_{\Pi}\left[\sum_{t=1}^{T} \mathbf{r} \cdot \left(\mathbf{X}^{(N)}(t) - \mathbf{U}^{(N)}(t), \mathbf{U}^{(N)}(t)\right)\right] \tag{5a}$$

$$\text{s.t.} \qquad , \tag{5b}$$

$$\mathbf{U}^{(N)}(t) \cdot \mathbf{1}^{\top} = \alpha, \; \mathbf{0} \leq \mathbf{U}^{(N)}(t) \leq \mathbf{X}^{(N)}(t), \tag{5c}$$

$$\mathbf{X}^{(N)}(1) \text{ is given} \tag{5d}$$

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s
$\mathbf{U}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Occupation Measure Formulation

$$\max_{\Pi : \mathbf{X}^{(N)} \to \mathbf{U}^{(N)}} \quad \mathbb{E}_\Pi \left[\sum_{t=1}^{T} \mathbf{r} \cdot \left(\mathbf{X}^{(N)}(t) - \mathbf{U}^{(N)}(t), \mathbf{U}^{(N)}(t) \right) \right] \tag{5a}$$

$$\text{s.t.} \quad \mathbf{X}^{(N)}(t+1) = \phi(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)) + \mathcal{E}(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)), \tag{5b}$$

$$\mathbf{U}^{(N)}(t) \cdot \mathbf{1}^\top = \alpha, \ \mathbf{0} \le \mathbf{U}^{(N)}(t) \le \mathbf{X}^{(N)}(t), \tag{5c}$$

$$\mathbf{X}^{(N)}(1) \text{ is given} \tag{5d}$$

$\phi(\cdot,\cdot)$: deterministic (affine) drift
$\mathcal{E}(\cdot,\cdot)$: density dependent noise

Restless Bandit: Occupation Measure Formulation

$\mathbf{X}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $X_s^{(N)}$ is the *fraction* of arms in state s

$\mathbf{U}^{(N)} \in \mathbb{R}^S$: the s-th coordinate $U_s^{(N)}$ is the *fraction* of arms in state s to be pulled

Occupation Measure Formulation

$$\max_{\Pi : \mathbf{X}^{(N)} \to \mathbf{U}^{(N)}} \quad \mathbb{E}_\Pi \left[\sum_{t=1}^{T} \mathbf{r} \cdot \left(\mathbf{X}^{(N)}(t) - \mathbf{U}^{(N)}(t), \mathbf{U}^{(N)}(t) \right) \right] \tag{5a}$$

$$\text{s.t.} \quad \mathbf{X}^{(N)}(t+1) = \phi(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)) + \mathcal{E}(\mathbf{X}^{(N)}(t), \mathbf{U}^{(N)}(t)), \tag{5b}$$

$$\mathbf{U}^{(N)}(t) \cdot \mathbf{1}^\top = \alpha, \; \mathbf{0} \le \mathbf{U}^{(N)}(t) \le \mathbf{X}^{(N)}(t), \tag{5c}$$

$$\mathbf{X}^{(N)}(1) \text{ is given} \tag{5d}$$

$\phi(\cdot,\cdot)$: deterministic (affine) drift

$\mathcal{E}(\cdot,\cdot)$: density dependent noise

♣ What if the $\mathcal{E}(\cdot,\cdot)$ terms were not there?

Framework to construct CEC

Multi-Stage Stochastic Optimization [5]

Let h, ϕ be affine, f be concave and g be convex $\mathcal{C}^2$-smooth functions of appropriate dimensions, and $\mathcal{E}$ be density dependent noise [4]

A Multi-Stage Stochastic Optimization Problem

$$V_{\text{opt}}(\mathbf{X}(1)) = \max_{\Pi : \mathbf{X} \to \mathbf{U}} \quad \mathbb{E}_{\Pi}\left[\sum_{t=1}^{T} f(\mathbf{X}(t), \mathbf{U}(t))\right] \tag{6a}$$

$$\text{s.t.} \quad \mathbf{X}(t+1) = \phi(\mathbf{X}(t), \mathbf{U}(t)) + \mathcal{E}(\mathbf{X}(t), \mathbf{U}(t)), \tag{6b}$$

$$g(\mathbf{X}(t), \mathbf{U}(t)) \leq \mathbf{0}, \; h(\mathbf{X}(t), \mathbf{U}(t)) = \mathbf{0}, \tag{6c}$$

$$\mathbf{X}(1) \text{ is given} \tag{6d}$$

where Π consists of T feasible and $\mathcal{F}_t$-measurable maps $\pi_t : \mathbf{X}(t) \mapsto \mathbf{U}(t)$

[4] We drop the dependence on N in the vectors.

[5] Shapiro, Dentcheva, and Ruszczynski, *Lectures on stochastic programming: modeling and theory*, Chapter 3

The Certainty Equivalence Problem

Certainty Equivalence Control (CEC) [6]: replace all the uncertainties by their nominal values

$$\max_{\Pi : \mathbf{X} \to \mathbf{U}} \mathbb{E}_{\Pi}\left[\sum_{t=1}^{T} f(\mathbf{X}(t), \mathbf{U}(t))\right] := V_{\text{opt}}(\mathbf{X}(1))$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{X}(t+1) = \phi(\mathbf{X}(t), \mathbf{U}(t)) + \mathcal{E}(\mathbf{X}(t), \mathbf{U}(t)), \\ & g(\mathbf{X}(t), \mathbf{U}(t)) \leq \mathbf{0},\ h(\mathbf{X}(t), \mathbf{U}(t)) = \mathbf{0}, \\ & \mathbf{X}(1) \text{ is given} \end{aligned}$$

$$\max_{\mathbf{u}[1,T]} \left[\sum_{t=1}^{T} f(\mathbf{x}(t), \mathbf{u}(t))\right] := V_{\text{rel}}(\mathbf{X}(1))$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{x}(t+1) = \phi(\mathbf{x}(t), \mathbf{u}(t)), \\ & g(\mathbf{x}(t), \mathbf{u}(t)) \leq \mathbf{0},\ h(\mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0}, \\ & \mathbf{x}(1) = \mathbf{X}(1) \text{ is given} \end{aligned}$$

[6] Bertsekas, *Dynamic programming and optimal control: Volume I*, Chapter 6

The Certainty Equivalence Problem

Certainty Equivalence Control (CEC) [6]: replace all the uncertainties by their nominal values

$$\max_{\Pi : \mathbf{X} \to \mathbf{U}} \mathbb{E}_{\Pi}\left[\sum_{t=1}^{T} f(\mathbf{X}(t), \mathbf{U}(t))\right] := V_{\text{opt}}(\mathbf{X}(1))$$

$$\text{s.t.} \quad \mathbf{X}(t+1) = \phi(\mathbf{X}(t), \mathbf{U}(t)) + \mathcal{E}(\mathbf{X}(t), \mathbf{U}(t)),$$
$$g(\mathbf{X}(t), \mathbf{U}(t)) \leq \mathbf{0}, \; h(\mathbf{X}(t), \mathbf{U}(t)) = \mathbf{0},$$
$$\mathbf{X}(1) \text{ is given}$$

$$\max_{\mathbf{u}[1,T]} \left[\sum_{t=1}^{T} f(\mathbf{x}(t), \mathbf{u}(t))\right] := V_{\text{rel}}(\mathbf{X}(1))$$

$$\text{s.t.} \quad \mathbf{x}(t+1) = \phi(\mathbf{x}(t), \mathbf{u}(t)),$$
$$g(\mathbf{x}(t), \mathbf{u}(t)) \leq \mathbf{0}, \; h(\mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0},$$
$$\mathbf{x}(1) = \mathbf{X}(1) \text{ is given}$$

Observations:

- The r.h.s. is simply a deterministic and convex mathematical program
- Were it be that $\mathcal{E}(\cdot,\cdot)$ are identically zero, the two problems are identical
- When $\mathcal{E}(\cdot,\cdot)$ are "small", the solutions to the two problems should be "close"
- $V_{\text{opt}}(\mathbf{X}(1)) \leq V_{\text{rel}}(\mathbf{X}(1))$ because of the convexity assumptions

[6] Bertsekas, *Dynamic programming and optimal control: Volume I*, Chapter 6

The CEC: Intuition of Why It Works

Let $\mathbf{u}^*(1), \ldots, \mathbf{u}^*(T)$ be an optimal solution to the deterministic problem
$\rightsquigarrow \mathbf{x}^*(1), \ldots, \mathbf{x}^*(T)$

[7] Recall $\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \mid g(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}, h(\mathbf{x}, \mathbf{u}) = \mathbf{0}\}$

The CEC: Intuition of Why It Works

Let $\mathbf{u}^*(1), \ldots, \mathbf{u}^*(T)$ be an optimal solution to the deterministic problem
$\rightsquigarrow \mathbf{x}^*(1), \ldots, \mathbf{x}^*(T)$

Suppose somehow we *have constructed* a feasible policy $\pi_t : \mathbf{x}(t) \mapsto \mathcal{U}(\mathbf{x}(t))$ for all $1 \leq t \leq T$ [7] such that

- $\pi_t(\mathbf{x}^*(t)) = \mathbf{u}^*(t)$
- $\pi_t(\cdot)$ are well-behaved in a neighbourhood of $\mathbf{x}^*(t)$ (i.e. smooth enough)

[7] Recall $\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \mid g(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}, h(\mathbf{x}, \mathbf{u}) = \mathbf{0}\}$

The CEC: Intuition of Why It Works

Let $\mathbf{u}^*(1), \ldots, \mathbf{u}^*(T)$ be an optimal solution to the deterministic problem
$\rightsquigarrow \mathbf{x}^*(1), \ldots, \mathbf{x}^*(T)$

Suppose somehow we *have constructed* a feasible policy $\pi_t : \mathbf{x}(t) \mapsto \mathcal{U}(\mathbf{x}(t))$ for all $1 \leq t \leq T$ [7] such that

- $\pi_t(\mathbf{x}^*(t)) = \mathbf{u}^*(t)$
- $\pi_t(\cdot)$ are well-behaved in a neighbourhood of $\mathbf{x}^*(t)$ (i.e. smooth enough)

Then:

$\mathbf{x}^*(1) = \mathbf{X}(1)$

$\mathbf{x}^*(2) = \phi(\mathbf{x}^*(1), \mathbf{u}^*(1)) = \phi(\mathbf{x}^*(1), \pi_1(\mathbf{x}^*(1)))$

$\mathbf{X}(2) = \phi(\mathbf{X}(1), \pi_1(\mathbf{X}(1))) + \mathcal{E}(\mathbf{X}(1), \pi_1(\mathbf{X}(1))) \approx \phi(\mathbf{x}^*(1), \pi_1(\mathbf{x}^*(1))) = \mathbf{x}^*(2)$

[7] Recall $\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \mid g(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}, h(\mathbf{x}, \mathbf{u}) = \mathbf{0}\}$

The CEC: Intuition of Why It Works

Let $\mathbf{u}^*(1), \ldots, \mathbf{u}^*(T)$ be an optimal solution to the deterministic problem $\rightsquigarrow \mathbf{x}^*(1), \ldots, \mathbf{x}^*(T)$

Suppose somehow we *have constructed* a feasible policy $\pi_t : \mathbf{x}(t) \mapsto \mathcal{U}(\mathbf{x}(t))$ for all $1 \leq t \leq T$ [7] such that

- $\pi_t(\mathbf{x}^*(t)) = \mathbf{u}^*(t)$
- $\pi_t(\cdot)$ are well-behaved in a neighbourhood of $\mathbf{x}^*(t)$ (i.e. smooth enough)

Then:

$\mathbf{x}^*(1) = \mathbf{X}(1)$

$\mathbf{x}^*(2) = \phi(\mathbf{x}^*(1), \mathbf{u}^*(1)) = \phi(\mathbf{x}^*(1), \pi_1(\mathbf{x}^*(1)))$

$\mathbf{X}(2) = \phi(\mathbf{X}(1), \pi_1(\mathbf{X}(1))) + \mathcal{E}(\mathbf{X}(1), \pi_1(\mathbf{X}(1))) \approx \phi(\mathbf{x}^*(1), \pi_1(\mathbf{x}^*(1))) = \mathbf{x}^*(2)$

Because $\mathcal{E}$ is small and $\pi_1(\cdot)$ is smooth

[7] Recall $\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \mid g(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}, h(\mathbf{x}, \mathbf{u}) = \mathbf{0}\}$

The CEC: Intuition of Why It Works

More generally, for time-step t:

$$\mathbf{x}^*(t) \approx \mathbf{X}(t)$$
$$\mathbf{x}^*(t+1) = \phi(\mathbf{x}^*(t), \mathbf{u}^*(t)) = \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t)))$$
$$\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \pi_t(\mathbf{X}(t))) + \mathcal{E}(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))$$

The CEC: Intuition of Why It Works

More generally, for time-step t:

$$\mathbf{x}^*(t) \approx \mathbf{X}(t)$$
$$\mathbf{x}^*(t+1) = \phi(\mathbf{x}^*(t), \mathbf{u}^*(t)) = \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t)))$$
$$\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \pi_t(\mathbf{X}(t))) + \mathcal{E}(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))$$
$$\Rightarrow \mathbf{X}(t+1) \approx \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t))) = \mathbf{x}^*(t+1)$$

The CEC: Intuition of Why It Works

More generally, for time-step t:

$$\mathbf{x}^*(t) \approx \mathbf{X}(t)$$
$$\mathbf{x}^*(t+1) = \phi(\mathbf{x}^*(t), \mathbf{u}^*(t)) = \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t)))$$
$$\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \pi_t(\mathbf{X}(t))) + \mathcal{E}(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))$$
$$\Rightarrow \mathbf{X}(t+1) \approx \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t))) = \mathbf{x}^*(t+1)$$

Because $\mathcal{E}$ is small and $\pi_t(\cdot)$ is smooth

The CEC: Intuition of Why It Works

More generally, for time-step t:

$$\mathbf{x}^*(t) \approx \mathbf{X}(t)$$
$$\mathbf{x}^*(t+1) = \phi(\mathbf{x}^*(t), \mathbf{u}^*(t)) = \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t)))$$
$$\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \pi_t(\mathbf{X}(t))) + \mathcal{E}(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))$$
$$\Rightarrow \mathbf{X}(t+1) \approx \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t))) = \mathbf{x}^*(t+1)$$

Because $\mathcal{E}$ is small and $\pi_t(\cdot)$ is smooth

Since

$$V_\Pi(\mathbf{X}(1)) = \mathbb{E}_\Pi\left[\sum_{t=1}^{T} f(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))\right]$$
$$V_{\text{rel}}(\mathbf{X}(1)) = \sum_{t=1}^{T} f(x^*(t), \pi_t(\mathbf{x}^*(t)))$$

We deduce that $V_\Pi(\mathbf{X}(1)) \approx V_{\text{rel}}(\mathbf{X}(1))$

The CEC: Intuition of Why It Works

More generally, for time-step t:

$$\mathbf{x}^*(t) \approx \mathbf{X}(t)$$
$$\mathbf{x}^*(t+1) = \phi(\mathbf{x}^*(t), \mathbf{u}^*(t)) = \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t)))$$
$$\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \pi_t(\mathbf{X}(t))) + \mathcal{E}(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))$$
$$\Rightarrow \mathbf{X}(t+1) \approx \phi(\mathbf{x}^*(t), \pi_t(\mathbf{x}^*(t))) = \mathbf{x}^*(t+1)$$

Because $\mathcal{E}$ is small and $\pi_t(\cdot)$ is smooth

Since

$$V_\Pi(\mathbf{X}(1)) = \mathbb{E}_\Pi\left[\sum_{t=1}^{T} f(\mathbf{X}(t), \pi_t(\mathbf{X}(t)))\right]$$
$$V_{\text{rel}}(\mathbf{X}(1)) = \sum_{t=1}^{T} f(x^*(t), \pi_t(\mathbf{x}^*(t)))$$

We deduce that $V_\Pi(\mathbf{X}(1)) \approx V_{\text{rel}}(\mathbf{X}(1))$

As $V_\Pi(\mathbf{X}(1)) \leq \underbrace{V_{\text{opt}}(\mathbf{X}(1)) \leq V_{\text{rel}}(\mathbf{X}(1))}_{\text{because of convexity}} \Rightarrow V_\Pi(\mathbf{X}(1)) \approx V_{\text{opt}}(\mathbf{X}(1))$

Recapitulation (I)

$$\max_{\Pi : \mathbf{X} \to \mathbf{U}} \mathbb{E}_\Pi \left[\sum_{t=1}^{T} f(\mathbf{X}(t), \mathbf{U}(t)) \right] := V_{\text{opt}}(\mathbf{X}(1)) \qquad \max_{\mathbf{u}[1,T]} \left[\sum_{t=1}^{T} f(\mathbf{x}(t), \mathbf{u}(t)) \right] := V_{\text{rel}}(\mathbf{X}(1))$$

s.t. $\mathbf{X}(t+1) = \phi(\mathbf{X}(t), \mathbf{U}(t)) + \mathcal{E}(\mathbf{X}(t), \mathbf{U}(t)),$
$g(\mathbf{X}(t), \mathbf{U}(t)) \leq \mathbf{0}, \; h(\mathbf{X}(t), \mathbf{U}(t)) = \mathbf{0},$
$\mathbf{X}(1)$ is given

s.t. $\mathbf{x}(t+1) = \phi(\mathbf{x}(t), \mathbf{u}(t)),$
$g(\mathbf{x}(t), \mathbf{u}(t)) \leq \mathbf{0}, \; h(\mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0},$
$\mathbf{x}(1) = \mathbf{X}(1)$ is given

Meta Theorem: Local Regularity determines Convergence Rate

[a] Suppose the density dependent noise $\mathcal{E}$ is such that $\mathrm{var}[\mathcal{E}] \leq \varepsilon$ [b], with $\varepsilon > 0$ *sufficiently small*. Let $\mathbf{u}^*(t), \mathbf{x}^*(t)$, $1 \leq t \leq T$ be an optimal solution to the r.h.s. above. For a *feasible* [c] policy Π and all t, if π_t in a neighbourhood of $\mathbf{x}^*(t)$:

1. is Lipschitz-continuous $\Rightarrow V_{\text{opt}} - V_\Pi \leq C_1 \cdot \sqrt{\varepsilon}$
2. is $\mathcal{C}^2$-smooth $\Rightarrow V_{\text{opt}} - V_\Pi \leq C_2 \cdot \varepsilon$
3. is affine $\Rightarrow V_{\text{opt}} - V_\Pi \leq C_3 \cdot e^{-C_4/\varepsilon}$

where the C's are positive constants depend on f, g, h, ϕ and T, but independent of ε.

---

[a] Yan and Reiffers-Masson, "Certainty Equivalence Control-Based Heuristics in Multi-Stage Convex Stochastic Optimization Problems"

[b] $\mathrm{var}[\mathcal{E}(\mathbf{x}, \mathbf{u}) \mid \mathbf{x}, \mathbf{u}] \leq \varepsilon$ holds uniformly for all $(\mathbf{x}, \mathbf{u})$

[c] feasibility means that $\pi_t(\mathbf{x}) \in \mathcal{U}(\mathbf{x}) = \{\mathbf{u} \mid g(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}, h(\mathbf{x}, \mathbf{u}) = \mathbf{0}\}$

Recapitulation (II)

$$\max_{\Pi : \mathbf{X} \to \mathbf{U}} \mathbb{E}_\Pi \left[\sum_{t=1}^{T} \mathbf{r} \cdot (\mathbf{X}(t) - \mathbf{U}(t), \mathbf{U}(t)) \right] := V_{\text{opt}}(\mathbf{X}(1))$$

$$\text{s.t.} \quad \mathbf{X}(t+1) = \phi(\mathbf{X}(t), \mathbf{U}(t)) + \mathcal{E}(\mathbf{X}(t), \mathbf{U}(t)),$$
$$\mathbf{U}(t) \cdot \mathbf{1}^\top = \alpha, \ \mathbf{0} \leq \mathbf{U}(t) \leq \mathbf{X}(t),$$
$$\mathbf{X}(1) \text{ is given}$$

$$\max_{\mathbf{u}[1,T]} \left[\sum_{t=1}^{T} \mathbf{r} \cdot (\mathbf{x}(t) - \mathbf{u}(t), \mathbf{u}(t)) \right] := V_{\text{rel}}(\mathbf{X}(1))$$

$$\text{s.t.} \quad \mathbf{x}(t+1) = \phi(\mathbf{x}(t), \mathbf{u}(t)),$$
$$\mathbf{u}(t) \cdot \mathbf{1}^\top = \alpha, \ \mathbf{0} \leq \mathbf{u}(t) \leq \mathbf{x}(t),$$
$$\mathbf{x}(1) = \mathbf{X}(1) \text{ is given}$$

Corollary: Special Case of Restless Bandit Problem with N arms

[a] Let $\mathbf{u}^*(t), \mathbf{x}^*(t)$, $1 \leq t \leq T$ be an optimal solution to the r.h.s. above. For a feasible policy Π and all t, if π_t in a neighbourhood of $\mathbf{x}^*(t)$:

1. is Lipschitz-continuous $\Rightarrow V_{\text{opt}} - V_\Pi \leq C_1/\sqrt{N}$
2. is affine $\Rightarrow V_{\text{opt}} - V_\Pi \leq C_3 \cdot e^{-C_4 N}$

[a] Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality"

Recapitulation (II)

$$\max_{\Pi\,:\,\mathbf{X}\to\mathbf{U}} \mathbb{E}_\Pi\left[\sum_{t=1}^{T}\mathbf{r}\cdot(\mathbf{X}(t)-\mathbf{U}(t),\mathbf{U}(t))\right] := V_{\text{opt}}(\mathbf{X}(1))$$

$$\text{s.t.}\quad \mathbf{X}(t+1)=\phi(\mathbf{X}(t),\mathbf{U}(t))+\mathcal{E}(\mathbf{X}(t),\mathbf{U}(t)),$$

$$\mathbf{U}(t)\cdot\mathbf{1}^\top=\alpha,\ \mathbf{0}\le\mathbf{U}(t)\le\mathbf{X}(t),$$

$\mathbf{X}(1)$ is given

$$\max_{\mathbf{u}[1,T]} \left[\sum_{t=1}^{T}\mathbf{r}\cdot(\mathbf{x}(t)-\mathbf{u}(t),\mathbf{u}(t))\right] := V_{\text{rel}}(\mathbf{X}(1))$$

$$\text{s.t.}\quad \mathbf{x}(t+1)=\phi(\mathbf{x}(t),\mathbf{u}(t)),$$

$$\mathbf{u}(t)\cdot\mathbf{1}^\top=\alpha,\ \mathbf{0}\le\mathbf{u}(t)\le\mathbf{x}(t),$$

$\mathbf{x}(1)=\mathbf{X}(1)$ is given

Corollary: Special Case of Restless Bandit Problem with N arms

[a] Let $\mathbf{u}^*(t),\mathbf{x}^*(t)$, $1\le t\le T$ be an optimal solution to the r.h.s. above. For a feasible policy Π and all t, if π_t in a neighbourhood of $\mathbf{x}^*(t)$:

1. is Lipschitz-continuous $\Rightarrow V_{\text{opt}}-V_\Pi\le C_1/\sqrt{N}$
2. is affine $\Rightarrow V_{\text{opt}}-V_\Pi\le C_3\cdot e^{-C_4N}$

---

[a] Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality"

✠ These results tell us nothing about how to construct a such policy π!

Policy Construction and Regularity

Back on RB and WCMDP (finite horizon)

Restless Bandits (finite horizon):

- The Lagrangian policy with optimal tiebreaking [8]: $\mathcal{O}(1/\sqrt{N})$
- The fluid-priority policy [9]: $\mathcal{O}(1/N)$ if *non-degenerate*
- The water-filling policy; the LP-update policy [10]: $e^{-\mathcal{O}(N)}$ if *non-degenerate* + taking care of the *rounding error*

Weakly Coupled MDPs (finite horizon):

- The fluid-priority policy [11]: $\mathcal{O}(1/N)$ if *non-degenerate* (weaker)
- The LP-update policy [12]: $\mathcal{O}(1/N)$ if *non-degenerate*
- The occupation measure sampling policy [13]: $\mathcal{O}(1/\sqrt{N})$ overall

[8] Brown and Smith, “Index Policies and Performance Bounds for Dynamic Selection Problems”

[9] Zhang and Frazier, “Restless Bandits with Many Arms: Beating the Central Limit Theorem”

[10] Gast, Gaujal, and Yan, “LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality”

[11] Zhang and Frazier, “Near-optimality for infinite-horizon restless bandits with many arms”

[12] Gast, Gaujal, and Yan, “The LP-update policy for weakly coupled Markov decision processes”

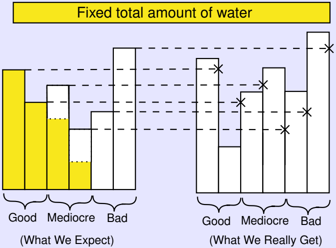
[13] Zayas-Cabán, Jasin, and Wang, “An Asymptotically Optimal Heuristic for General Non-Stationary Finite-Horizon Restless Multi-Armed Multi-Action Bandits”

What We Expect vs. What We Get in Reality

A Problem of Water Filling

- We fill a fixed amount of water into a collection of buckets to gain a utility. The buckets are classified into good (fully filled), mediocre (partially filled) and bad (no filled) via our estimation
- To maximize the utility, the *proportions* to partially fill the mediocre buckets have been carefully estimated, see the dotted lines in Mediocre buckets

The challenge of best matching the reality with our expectation



How to fill in reality?

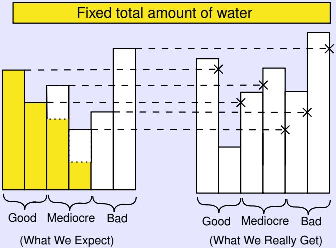
[a] See Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality", Section 4.2 for an illustration of how this problem is related to the RB

What We Expect vs. What We Get in Reality

A Problem of Water Filling

- We fill a fixed amount of water into a collection of buckets to gain a utility. The buckets are classified into good (fully filled), mediocre (partially filled) and bad (no filled) via our estimation
- To maximize the utility, the *proportions* to partially fill the mediocre buckets have been carefully estimated, see the dotted lines in Mediocre buckets

The challenge of best matching the reality with our expectation



How to fill in reality?

♣ The challenge is that the size of the buckets are *random variables* and our estimation are based on their *mean values*, before knowing their *true values* [a]

[a] See Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality", Section 4.2 for an illustration of how this problem is related to the RB

Policy Construction: Projection

For each time-step t, the feasible control set is

$$\mathcal{U}(\mathbf{X}(t)) = \{\mathbf{u} \mid g(\mathbf{X}(t), \mathbf{u}) \leq \mathbf{0},\ h(\mathbf{X}(t), \mathbf{u}) = \mathbf{0}\} \rightsquigarrow \text{ a set parameterized by } \mathbf{X}(t)$$

Idea: We *project* the vector $\mathbf{x}^*(t)$ onto $\mathcal{U}(\mathbf{X}(t))$.

The Projection Policy

Let $\mathbf{u}^*(t), \mathbf{x}^*(t)$, $1 \leq t \leq T$ be an optimal solution to the deterministic problem. The *projection policy* consists of taking for each time-step t

$$\pi_t^{(\text{proj})} : \mathbf{X}(t) \mapsto \text{Proj}_{\mathcal{U}(\mathbf{X}(t))}(\mathbf{x}^*(t))$$

Policy Construction: Projection

For each time-step t, the feasible control set is

$$\mathcal{U}(\mathbf{X}(t)) = \{\mathbf{u} \mid g(\mathbf{X}(t), \mathbf{u}) \leq \mathbf{0},\ h(\mathbf{X}(t), \mathbf{u}) = \mathbf{0}\} \rightsquigarrow \text{ a set parameterized by } \mathbf{X}(t)$$

Idea: We *project* the vector $\mathbf{x}^*(t)$ onto $\mathcal{U}(\mathbf{X}(t))$.

The Projection Policy

Let $\mathbf{u}^*(t), \mathbf{x}^*(t)$, $1 \leq t \leq T$ be an optimal solution to the deterministic problem. The *projection policy* consists of taking for each time-step t

$$\pi_t^{(\mathrm{proj})} : \mathbf{X}(t) \mapsto \mathrm{Proj}_{\mathcal{U}(\mathbf{X}(t))}(\mathbf{x}^*(t))$$

Advantages:

1. $\pi_t^{(\mathrm{proj})}(\cdot)$ is feasible by construction
2. $\pi_t^{(\mathrm{proj})}(\mathbf{x}^*(t)) = \mathbf{u}^*(t)$, and we expect that $\pi_t^{(\mathrm{proj})}(\mathbf{X}(t)) \approx \mathbf{u}^*(t)$, provided that $\mathbf{X}(t) \approx \mathbf{x}^*(t)$
3. A projection is relatively easy to compute (compared to solving a multi-stage mathematical program each time for the update policy)

Policy Construction: Projection

For each time-step t, the feasible control set is

$$\mathcal{U}(\mathbf{X}(t)) = \{\mathbf{u} \mid g(\mathbf{X}(t), \mathbf{u}) \leq \mathbf{0},\ h(\mathbf{X}(t), \mathbf{u}) = \mathbf{0}\} \rightsquigarrow \text{ a set parameterized by } \mathbf{X}(t)$$

Idea: We *project* the vector $\mathbf{x}^*(t)$ onto $\mathcal{U}(\mathbf{X}(t))$.

The Projection Policy

Let $\mathbf{u}^*(t), \mathbf{x}^*(t)$, $1 \leq t \leq T$ be an optimal solution to the deterministic problem. The *projection policy* consists of taking for each time-step t

$$\pi_t^{(\text{proj})} : \mathbf{X}(t) \mapsto \text{Proj}_{\mathcal{U}(\mathbf{X}(t))}(\mathbf{x}^*(t))$$

Advantages:

1. $\pi_t^{(\text{proj})}(\cdot)$ is feasible by construction
2. $\pi_t^{(\text{proj})}(\mathbf{x}^*(t)) = \mathbf{u}^*(t)$, and we expect that $\pi_t^{(\text{proj})}(\mathbf{X}(t)) \approx \mathbf{u}^*(t)$, provided that $\mathbf{X}(t) \approx \mathbf{x}^*(t)$
3. A projection is relatively easy to compute (compared to solving a multi-stage mathematical program each time for the update policy)

♣: The analysis for the regularity of the mapping $\pi_t^{(\text{proj})}(\cdot)$ is non-trivial. See Yan and Reiffers-Masson, “Certainty Equivalence Control-Based Heuristics in Multi-Stage Convex Stochastic Optimization Problems”, Appendix B

Policy Construction: Update

For each time-step t, given the current state vector $\mathbf{X}(t)$, we solve a new program

$$\begin{aligned} V_{\text{rel}}(\mathbf{X}(t)) := \max_{\mathbf{u}[t,T]} \quad & \Big[\sum_{t=t'}^{T} f(\mathbf{x}(t'), \mathbf{u}(t')) \Big] \\ \text{s.t.} \quad & \mathbf{x}(t'+1) = \phi(\mathbf{x}(t'), \mathbf{u}(t')), \\ & g(\mathbf{x}(t'), \mathbf{u}(t')) \le \mathbf{0}, \ \ h(\mathbf{x}(t'), \mathbf{u}(t')) = \mathbf{0}, \\ & \mathbf{x}(t) = \mathbf{X}(t) \text{ is given} \end{aligned}$$

Idea: Denote by $\hat{\mathbf{u}}[t, T]$ an optimal solution. We pick the first (the t-th for real) control $\hat{\mathbf{u}}(t)$

Policy Construction: Update

For each time-step t, given the current state vector $\mathbf{X}(t)$, we solve a new program

$$V_{\mathrm{rel}}(\mathbf{X}(t)) := \max_{\mathbf{u}[t,T]} \left[\sum_{t=t'}^{T} f(\mathbf{x}(t'), \mathbf{u}(t')) \right]$$

$$\text{s.t.} \quad \mathbf{x}(t'+1) = \phi(\mathbf{x}(t'), \mathbf{u}(t')),$$

$$g(\mathbf{x}(t'), \mathbf{u}(t')) \leq \mathbf{0}, \; h(\mathbf{x}(t'), \mathbf{u}(t')) = \mathbf{0},$$

$$\mathbf{x}(t) = \mathbf{X}(t) \text{ is given}$$

Idea: Denote by $\hat{\mathbf{u}}[t, T]$ an optimal solution. We pick the first (the t-th for real) control $\hat{\mathbf{u}}(t)$

The Update Policy

For each time-step t, upon observing the state vector $\mathbf{X}(t)$, solve the program $V_{\mathrm{rel}}(\mathbf{X}(t))$ for $\hat{\mathbf{u}}[t, T]$, and use the control

$$\pi_t^{(\text{update})} : \mathbf{X}(t) \mapsto \hat{\mathbf{u}}(t) \in \text{ the first control of } \operatorname*{arg\,max}_{\mathbf{u}[t,T]} V_{\mathrm{rel}}(\mathbf{X}(t))$$

Policy Construction: Update

For each time-step t, given the current state vector $\mathbf{X}(t)$, we solve a new program

$$V_{\text{rel}}(\mathbf{X}(t)) := \max_{\mathbf{u}[t,T]} \left[\sum_{t=t'}^{T} f(\mathbf{x}(t'), \mathbf{u}(t'))\right]$$
$$\text{s.t.} \quad \mathbf{x}(t'+1) = \phi(\mathbf{x}(t'), \mathbf{u}(t')),$$
$$g(\mathbf{x}(t'), \mathbf{u}(t')) \leq \mathbf{0}, \; h(\mathbf{x}(t'), \mathbf{u}(t')) = \mathbf{0},$$
$$\mathbf{x}(t) = \mathbf{X}(t) \text{ is given}$$

Idea: Denote by $\hat{\mathbf{u}}[t, T]$ an optimal solution. We pick the first (the t-th for real) control $\hat{\mathbf{u}}(t)$

The Update Policy

For each time-step t, upon observing the state vector $\mathbf{X}(t)$, solve the program $V_{\text{rel}}(\mathbf{X}(t))$ for $\hat{\mathbf{u}}[t, T]$, and use the control

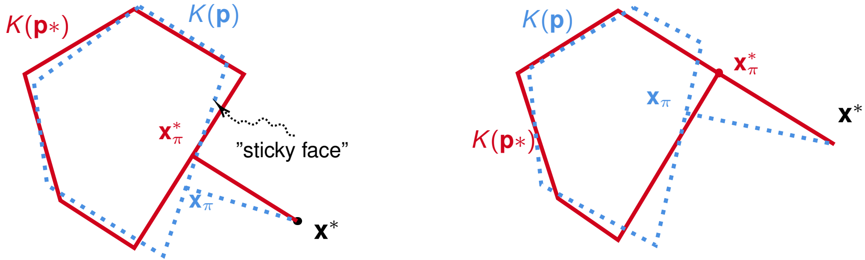
$$\pi_t^{(\text{update})} : \mathbf{X}(t) \mapsto \hat{\mathbf{u}}(t) \in \text{ the first control of } \arg\max_{\mathbf{u}[t,T]} V_{\text{rel}}(\mathbf{X}(t))$$

♣: The analysis for the regularity of the mapping $\pi_t^{(\text{update})}(\cdot)$ relies on the same set of tools for analysing $\pi_t^{(\text{proj})}(\cdot)$.

Illustration of the Regularity

Projection onto a polygon:

$$\mathbf{x}_\pi^* = \Pi_{K(\mathbf{p}^*)}(\mathbf{x}^*) \xrightarrow[\text{on } \mathbf{p}^*]{\text{small perturbation}} \mathbf{x}_\pi = \Pi_{K(\mathbf{p})}(\mathbf{x}^*)$$

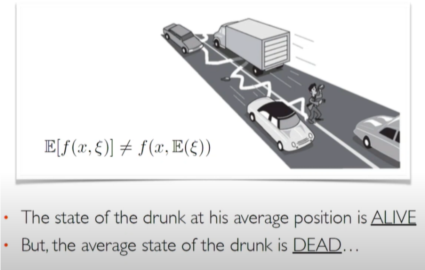


On the left, $\mathbf{x}_\pi^*$ is *non-degenerate*. On the right, $\mathbf{x}_\pi^*$ is *degenerate* [14]

[14] The terminology "sticky face" is coined in the survey article Robinson, "Variational conditions with smooth constraints: structure and analysis"

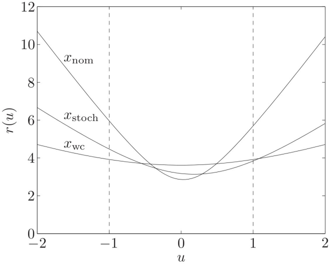
Conclusion

Is CEC always a good idea?



In some cases completely ignore uncertainty can lead to severe consequences. [a]

[a] Taken from a slide in this video of Phebe Vayanos: *Robust Optimization and Sequential Decision-Making*



[a] Minimize over vector x:
$\|(A_0 + uA_1)\cdot x - b\|^2$
where $u \sim$ uniform$(-2, 2)$, A_0, A_1 and b are known matrices and vector.
x_{nom}: Use the nominal value (CEC)
x_{stoch}: Stochastic optimization
x_{wc}: Worst case optimization (RO)

[a] Taken from Boyd and Vandenberghe, *Convex optimization*, *Example 6.5, page 320*

Link with Robust and Distributional Robust Optimization

✠ : $\pi(\mathbf{x}^*) = \mathbf{u}^*$ is unnecessary if we are not interested in the asymptotic limit where the variances are zero

♠ : What we really need is less demanding:

$$\pi(\mathbf{X}) \approx \mathbf{u}^* \text{ for } \textit{any } \mathbf{X} \approx \mathbf{x}^* \text{ and } \pi(\cdot) \text{ are } \textit{smooth} \text{ there}$$

[15] Boyd and Vandenberghe, *Convex optimization*, Section 11.3

[16] The link with DRO may be much deeper, see e.g. Blanchet et al., “Unifying Distributionally Robust Optimization via Optimal Transport Theory”

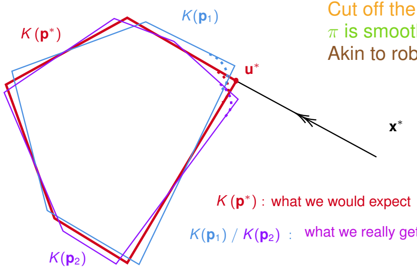
Link with Robust and Distributional Robust Optimization

✠ : $\pi(\mathbf{x}^*) = \mathbf{u}^*$ is unnecessary if we are not interested in the asymptotic limit where the variances are zero

♠ : What we really need is less demanding:

$$\pi(\mathbf{X}) \approx \mathbf{u}^* \text{ for } \textit{any } \mathbf{X} \approx \mathbf{x}^* \text{ and } \pi(\cdot) \text{ are } \textit{smooth} \text{ there}$$

Use barrier functions [15] to smooth out the degenerate corners (dotted curves) [16]



Cut off the corner: we lose optimality

π is smoother: we gain regularity

Akin to robust optimization

---

[15] Boyd and Vandenberghe, *Convex optimization*, Section 11.3

[16] The link with DRO may be much deeper, see e.g. Blanchet et al., "Unifying Distributionally Robust Optimization via Optimal Transport Theory"

What we did not cover in this talk

1. How to dealt with *discrete* action space?
   - Take convex hull: this leads to *two* layers of relaxation
   - Require tools from geometric algorithm and combinatorial optimization
   - Efficiently compute the projection onto the convex hull of a (large) collection of points; algorithmic version of Caratheodory's theorem to apply *randomized rounding*
2. How to *scale* in the convex case?
   - When all the convex functions are *homogenous*
   - Scale with the horizon T: *fluid limit* vs. *mean field limit*
   - Formulate the infinite horizon time-averaged reward problem?
3. Interesting applications?
   - Network utility maximization problem from telecommunication
   - Network inventory management from inventory control
   - ... Your turn to discover!

Based on

1. Infinite horizon RB: Gast, Gaujal, and Yan, "Exponential asymptotic optimality of Whittle index policy" *Queueing Systems*
2. Finite horizon RB: Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality" *Mathematics of Operations Research*
3. Finite horizon WCMDPs: Gast, Gaujal, and Yan, "The LP-update policy for weakly coupled Markov decision processes" *arXiv*
4. Finite horizon Convex Case: Yan and Reiffers-Masson, "Certainty Equivalence Control-Based Heuristics in Multi-Stage Convex Stochastic Optimization Problems" *arXiv*

Based on

1. Infinite horizon RB: Gast, Gaujal, and Yan, "Exponential asymptotic optimality of Whittle index policy" *Queueing Systems*
2. Finite horizon RB: Gast, Gaujal, and Yan, "LP-based policies for restless bandits: necessary and sufficient conditions for (exponentially fast) asymptotic optimality" *Mathematics of Operations Research*
3. Finite horizon WCMDPs: Gast, Gaujal, and Yan, "The LP-update policy for weakly coupled Markov decision processes" *arXiv*
4. Finite horizon Convex Case: Yan and Reiffers-Masson, "Certainty Equivalence Control-Based Heuristics in Multi-Stage Convex Stochastic Optimization Problems" *arXiv*