Lecture 10

→ Don't forget to work on your presentation topic & results.

Student Discussion

cleaning up your code?
↳ DDP.

discussing the iLQR

Worth of policies.

iLQR gives you a set of control gains with respect to
↳ Base policy
↳ Final gains @ each timestep.

timestep by timestep

Quiz

Model predictive control
↳ popular term
↳ Most popular next to PD control.
↳ Definition varies

What is an issue with DP?
↳ Curse of dimensionality
↳ Takes a long time.

Good thing about DP
↳ Incorporates feedback
↳ can handle mistakes in dynamics.

What if you have a nonlinear set of dynamics
Computational space is burdensome, &
time is important and feedback is
important due to model mismatch
noise or perturbations.

Suggested : Sliding mode optimal
↳ Not great.

Sen Singer : DDP (only invented in 2004)
So MPC was the way to go

(iLQR was invented in 60's)

Model predictive Control

↳ Suboptimal Control Strategy

↳ Nonlinear

↳ Feedback

↳ Computationally tractable

1) Calculate the deterministic optimal path over a limited horizon

↳ Greedy

↳ Brute Force

↳ DP

How discritized

How limited

2) only execute 1st action

a whole sequence of actions.

3) Get feedback information

Loop back

4 signature traits that make MPC - MPC

1) Certainty equivalence principle

the optimal action of the expected value is unaffected by noise or perturbations

↳ LQR if there was additive noise it would shift the actions but does not change them

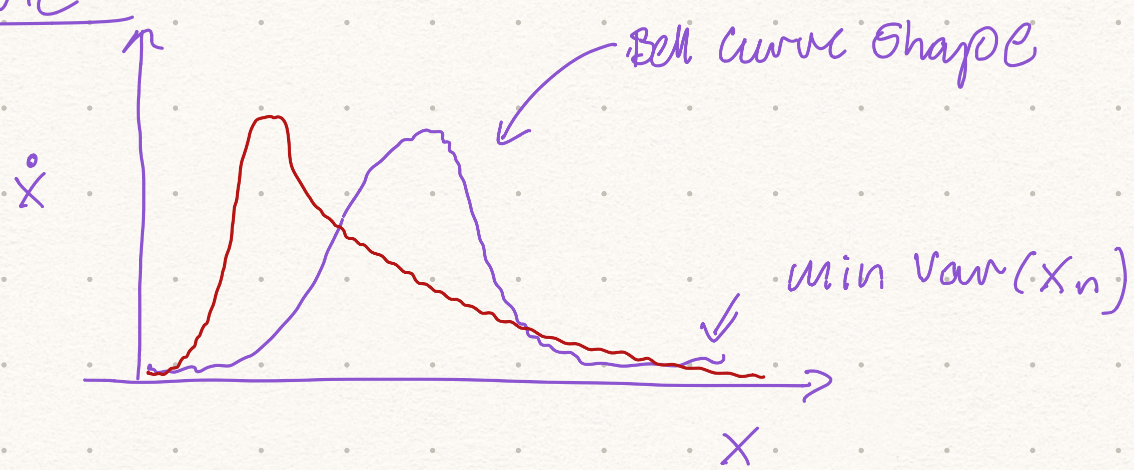
↳ Therefore ignore additive noise

$$\epsilon \sim N(0, \sigma^2)$$

↳ Does not hold for Multiplicitive noise

$$\epsilon \sim N(0, v^2)$$

Example



2) Multi step lookahead

Limited lookahead, the terminal cost @ the end of the window is suboptimal approximation

Multi step lookahead is a subset of limited lookahead in which the window is sufficiently large that the approx is decent.

$$J_{K+1} \equiv 0$$

$$J_{K+i} \equiv J_{[N]}$$ penalty after time ends for not being at the optimum.

3) Rollout Algorithm

the terminal cost at the end of the window is a known Sub optimal policy

4) Stability

↳ MPC is a feedback controller

↳ If designed properly is inherintly Stable.

↳ Needs proper Rollout

Looking at the Rollout with respect to Quiz.

Quiz with 100 questions. only answer one.

Max expected reward

$i = \arg\max [v]$ deterministic

$\arg\max E[v] = \arg\max [p \cdot v]$

Quiz with 100 questions.
You can answer any question until you get one wrong.

From Quiz is it similar to previous.

Index of preference (IOP)

$$\boxed{IOP = \frac{P_i V_i}{1 - P_i}}$$ optimal policy here

is this sorted?

Quiz with 100 questions.
You can only answer 10 questions Stop when you get one wrong.

$$\boxed{IOP = \frac{P_i V_i}{1 - P_i}}$$ Not optimal here (sub optimal) but a good policy

100q

Excluding $E_r[V_1 P_1]$

$E_r[V_1 P_1]$ + IOP ranking for the other questions.

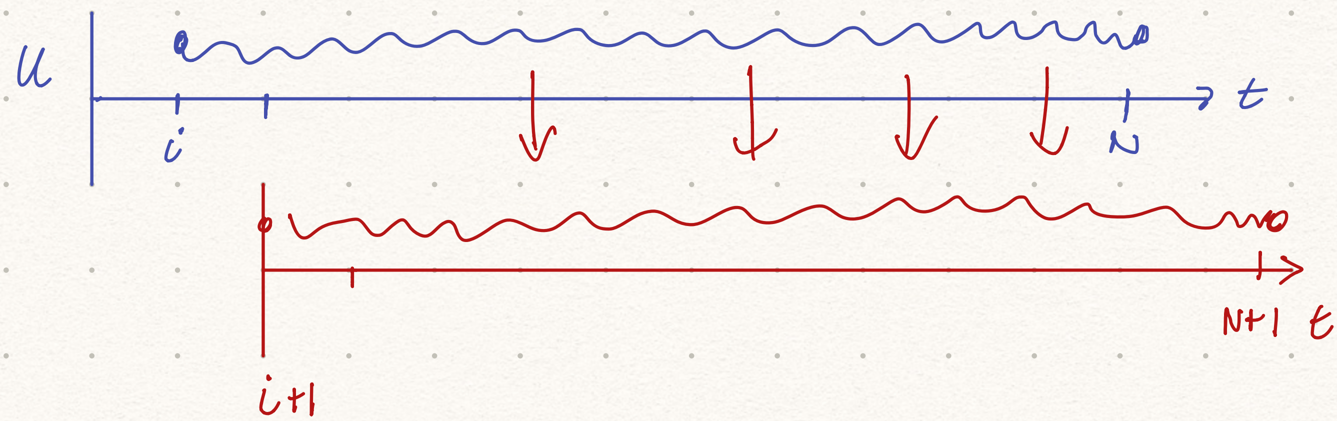
$E_r[V_2 P_2]$ + IOP ranking

$(err)^2$ - good rollout algorithm

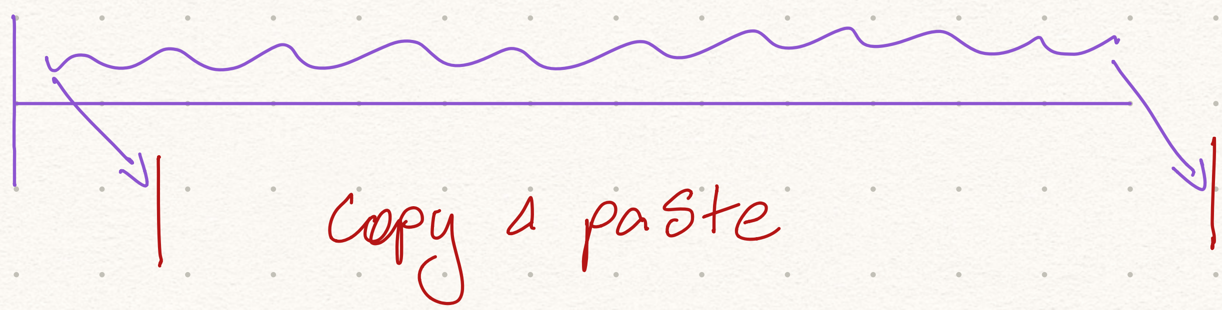
1st order system - at least 10 time points in the MPC window.

Start close to the final solution
↳ Online initialization.

Shift initialization



No shift initialization



When to use each one

Short window : No Shift
Medium - Long : Either

Non Autonomous
↳ periodic tracking
↳ dynamics depend on time
↳ cost function depends on time

Shift + Initialization

Categories of MPC

Robust MPC
↳ Min-Max
↳ optimizes system for worst case results

$$f(x, u)$$

$$f(x, u, w)$$

$$\min_{u} \max_{w} J(f(x, u, w))$$

Closed loop Min-Max
↳ optimazation over feedback policies.

Tube based

Multistaged MPC
↳ Incorporates different Scenarios

Adaptive Robust MPC
Incorporates parameterized estimation.

$u \Rightarrow x$
$f(x, u)$

In adaptive & Robust
↳ choice of u is not influenced by the fact that you could adapt.
↳ They are only influenced when they are changed.

Dual Control

↳ reach the goal.

↳ Learn the Model.

↳ Rock the boat to try & determine the dynamics.

1) Explicit

↳ Define the value of learning

$$J = J_p + 10 J_L$$

↳ Cost of learning

2) Implicit Methods.

↳ only care about performance at the end.

↳ Tapping on the breaks to test icy roads.

↳ Thangavel 2018

Similar to DP
↳ take a limited DP window.
↳ choosing Strategic heuristics.

Not after a Stationary policy.

Try coding up Brute Force.

Try showing plots of functions to see different behaviour.
↳ Vary one value for each graph.