ECE 558

Lecture 1 August 25th

Focus:

1. Centralised Stochastic Control
   - Stochastic dynamical System (!)
   - One Controller (!)
   - Controller has perfect recall (!)

2. Structural and Foundational aspects

Why?

Algorithmic ideas will be developed but explored only to some extent.

Algorithms Focus of Reinforcement Learning (RL)

CSE 598 Satinder Baveja FA2025

ECE 602 RL LEI YING WN 2026

Motivation: lots of applications Controls, Communications, Networks, Signal Processing, OR, Math, Finance, Economics, Statistics, Manufacturing, Robotics, AI, . . . , Quantum

Examples:

1. Scheduling in Cloud Systems/ Data Centers

Servers

Dispatcher

Jobs

Many servers millions

Service times are stochastic/ random

When to send jobs?

Server

queue

Criteria:

Minimize delay of processing.

Perspective A: Dispatcher sees full state, knows ability of servers

Based on job type, send the job to a specific server!

$\boxed{\text{MARKOV DECISION PROCESS (MDP)}}$

Perspective B: SENDING INFORMATION IS COSTLY

Dispatcher polls for information

⇒ Poll a fixed number of servers

Partial observability - imperfect information

$\boxed{\text{PARTIALLY OBSERVED MDPs (POMDPs)}}$

Poll a fixed number of servers

(i) Soft constraint - On the average meet this

(ii) Hard constraint - Never allowed to exceed this.

$\boxed{\text{CONSTRAINED MDPs + POMDPs}}$ (More advanced, Could be a Project topic)

2. Controlling the trajectory of a spacecraft, autonomous car, robot

System state known - MDP (State "feedback")

If sensors used - POMDP (Output "feedback")

3. Portfolio management

Have some money on day t - x_t

Have K instruments to choose from.

$K+1$ options - Choosing distribution π_t on $K+1$ options

Option k gets $\pi_t^k x_t$ amount of money

Y_{t+1}^k - a stochastic outcome

Function of $\pi_t^k x_t$ & Market behaviour (Stochastic)

$$x_{t+1} = \sum_{k=0}^{K} Y_{t+1}$$

Goal: T periods, π_t's each day to maximize wealth at the end of the period, but minimizing the chance of going bankrupt.

---

4. Active Hypothesis Testing

Compare 2 images to find differences (small differences)

C	A
B	D

A	C
B	D

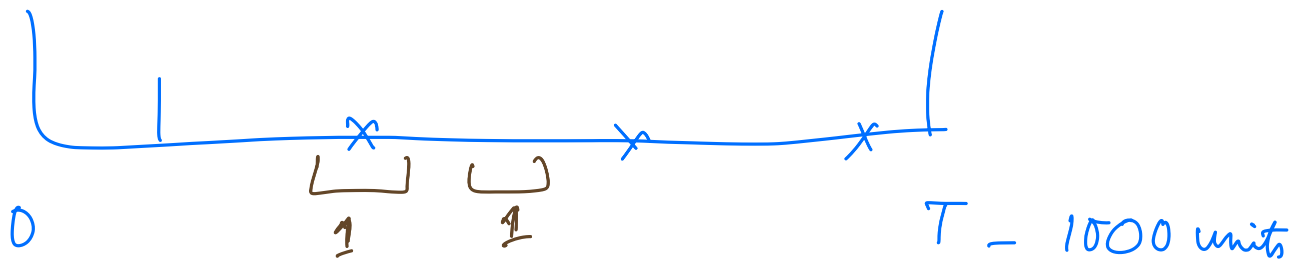
(KNOWN MODEL)

Goal: Tell apart the images (if different) & identify differences. FAST

How? Take different views at different granularities

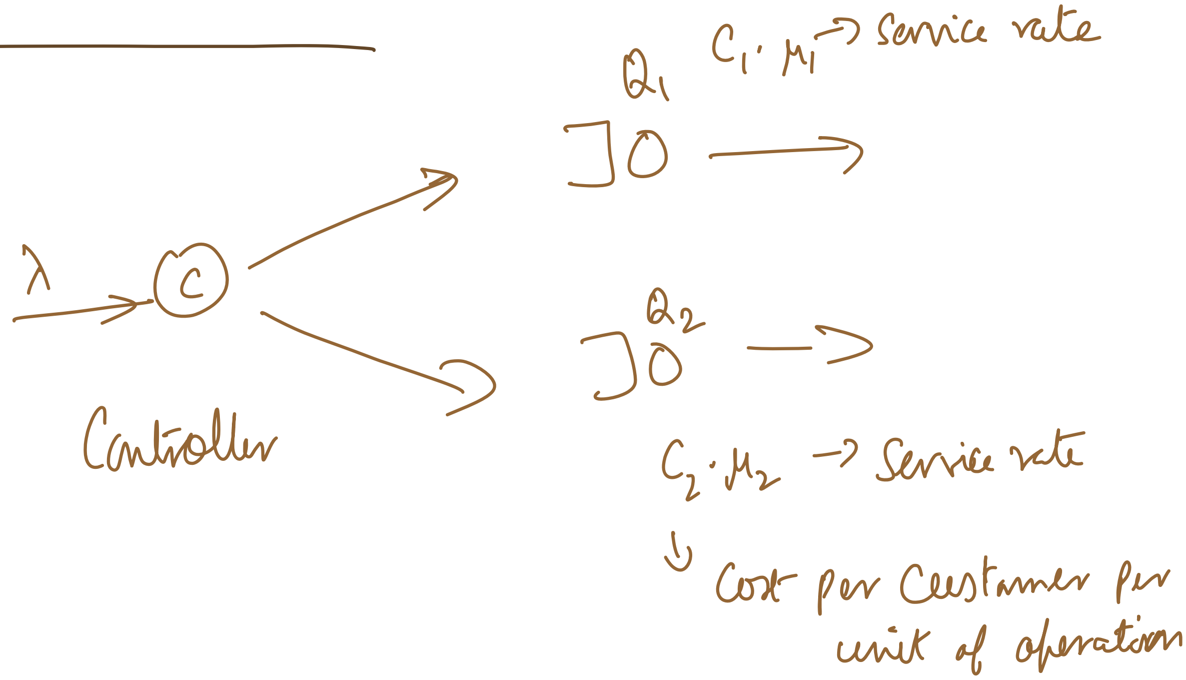
Choice {
- High resolution over a small region
- low resolution over a bigger region

Choosing next resolution & region based on past choices & results.



1 unit at a time

(1)



Network operates until time T

Occupancy of each queue at time $t \in \{1, 2, \ldots, T\} \triangleq [T]$

X_t^1 in Q_1

X_t^2 in Q_2

$$A_t = \begin{cases} \text{Route to } Q_1 & Q_1 \\ \text{Route to } Q_2 & Q_2 \end{cases} \qquad A_t \in \{Q_1, Q_2\}$$

Goal: Find a routing policy

$$g = (g_1, g_2, g_3, \dots, g_T)$$

that minimizes

$$\mathbb{E}^g \left[\sum_{t=1}^{T} \left(c_1 x_t^1 + c_2 x_t^2 \right) \right]$$

g_t — $(x_1^1, x_1^2), (x_2^1, x_2^2), \dots, (x_t^1, x_t^2)$

(Past states values including the present)

$A_1, A_2, \dots, A_{t-1}$

(Past action values)

Controller using all the past information

PERFECT RECALL

Assmpt: Queues are finite — max B

$$x_t^1 \in \{0, 1, 2, \dots, B\} \quad (B+1)$$

$g_t : H_t \to A_t$ $\qquad$ $A_t = \{Q_1, Q_2\}$

H_t takes values in a space that has size

$$((B+1)^2)^t \times 2^{t-1}$$

Number of functions $= 2^{((B+1)^2)^t \times 2^{t-1}}$

- Paring down to simpler functions to choose from while ensuring good performance.

- DP - breaks the time dependence

---

② Spacecraft - Vector state

$t \in \{0, 1, \dots\}$

$$x_{t+1} = \underset{\text{State}}{x_t} + \underset{\text{Control}}{u_t} + \underset{\text{noise}}{w_t}$$

Observation $\quad y_t = C x_t + v_t$ → noise in sensors.

Define $\quad y_{0:t} = (y_0, y_1, \dots, y_t)$

$\quad\quad\quad\; u_{0:t} = (u_0, u_1, \dots, u_t)$

$u_t = g_t(y_{0:t}, u_{0:t-1})$ and at time t

"Cost" is $\quad x_t^* Q x_t + u_t^* R u_t \quad t < T$

$\quad\quad\quad\quad x_T^* Q x_T \quad$ & if $t = T$

x^* - Transpose of x (Hermitian transpose)

$$x = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}, \text{ then } x^* = \begin{bmatrix} 0 & 1 & 2 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix}, \quad A^* = \begin{bmatrix} 0 & 2 \\ 1 & 3 \end{bmatrix}$$

Goal: Choose g to minimize

$$\mathbb{E}^g \left[\sum_{t=0}^{T-1} \left(x_t^* Q x_t + u_t^* R u_t \right) + x_T^* Q x_T \right]$$

$Q = I \qquad R = I$

$$x_t^* Q x_t = \sum_{k=1}^{K} (x_t^k)^2$$

Covered - Single controller with perfect recall.

Discrete-time, but state discrete or continuous.

Not covered (1) Continuous time problems (Technically more demanding, conceptually same ideas apply)

(2) Multi-controller problems (Conceptual leap)

(a) Team

(b) Games (c) Games of teams.