ECE276B: Planning & Learning in Robotics
Lecture 3: Markov Decision Processes

Instructor:
Nikolay Atanasov: natanasov@ucsd.edu

Teaching Assistant:
Hanwen Cao: h1cao@ucsd.edu

JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

Notation and Terminology

$t \in \{0, \ldots, T\}$	discrete time
$\mathbf{x} \in \mathcal{X}$	discrete/continuous state
$\mathbf{u} \in \mathcal{U}$	discrete/continuous control
$p_0(\mathbf{x})$	prior probability density/mass function defined on $\mathcal{X}$
$p_f(\mathbf{x}' \mid \mathbf{x}, \mathbf{u})$	motion model
$\ell(\mathbf{x}, \mathbf{u})$	stage cost of choosing control $\mathbf{u}$ in state $\mathbf{x}$
$\mathfrak{q}(\mathbf{x})$	terminal cost at state $\mathbf{x}$
$\pi_t(\mathbf{x})$	control policy: **function** from state $\mathbf{x}$ at time t to control $\mathbf{u}$
$V_t^\pi(\mathbf{x})$	value function: **expected cumulative cost** of starting at state $\mathbf{x}$ at time t and acting according to π
$\pi_t^*(\mathbf{x})$, $V_t^*(\mathbf{x})$	optimal control policy and value function

Markov Chain

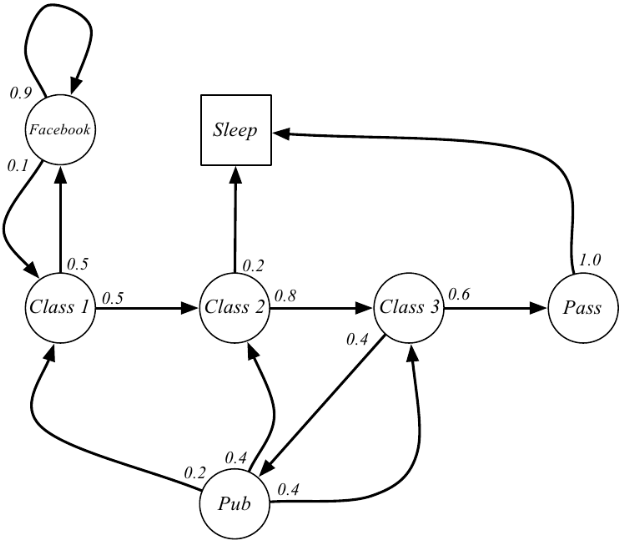
A **Markov Chain** is a stochastic process defined by a tuple $(\mathcal{X}, p_0, p_f, T)$:

- $\mathcal{X}$ is a discrete/continuous set of states
- p_0 is a prior pmf/pdf defined on $\mathcal{X}$
- $p_f(\cdot \mid \mathbf{x})$ is a conditional pmf/pdf defined on $\mathcal{X}$ for given $\mathbf{x} \in \mathcal{X}$ that specifies the stochastic process transitions
- T is a finite/infinite time horizon

- When there is a finite number of states, $\mathcal{X} := \{1, \ldots, N\}$, the motion model p_f is a probability mass function (pmf) and can be summarized by an $N \times N$ matrix with elements:

$$P_{ij} := \mathbb{P}(x_{t+1} = j \mid x_t = i) = p_f(j \mid x_t = i)$$

Example: Student Markov Chain

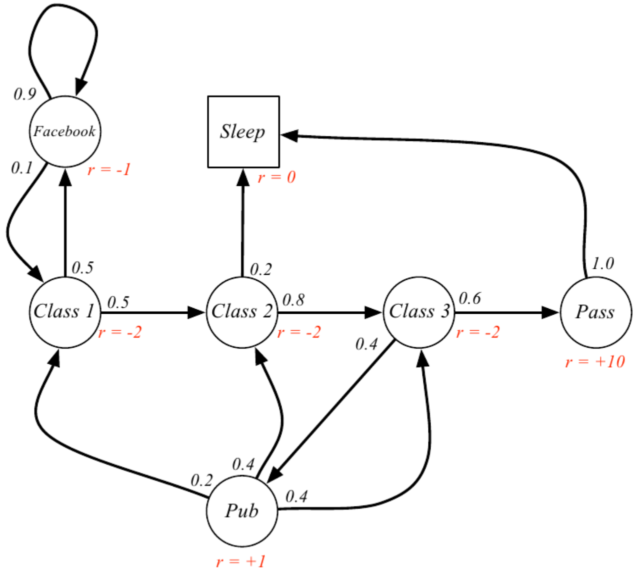


Markov Reward Process

A Markov Reward Process (MRP) is a Markov chain with costs defined by a tuple $(\mathcal{X}, p_0, p_f, T, \ell, \mathfrak{q}, \gamma)$:

- $\mathcal{X}$ is a discrete/continuous set of states
- p_0 is a prior pmf/pdf defined on $\mathcal{X}$
- $p_f(\cdot \mid \mathbf{x})$ is a conditional pmf/pdf defined on $\mathcal{X}$ for given $\mathbf{x} \in \mathcal{X}$ that specifies the stochastic process transitions
- T is a finite/infinite time horizon
- $\ell(\mathbf{x})$ is a function specifying the stage cost of state $\mathbf{x} \in \mathcal{X}$
- $\mathfrak{q}(\mathbf{x})$ is a terminal cost of being in state $\mathbf{x}$ at time T
- $\gamma \in [0, 1]$ is a discount factor

Example: Student Markov Reward Process



Value Function

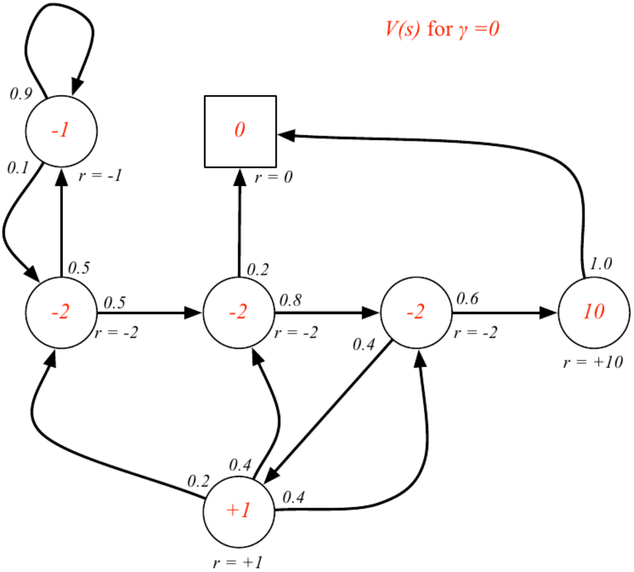
- **Value function**: the expected cumulative cost of an MRP starting from state $\mathbf{x} \in \mathcal{X}$ at time t

- **Finite-horizon**: trajectories terminate at fixed $T < \infty$

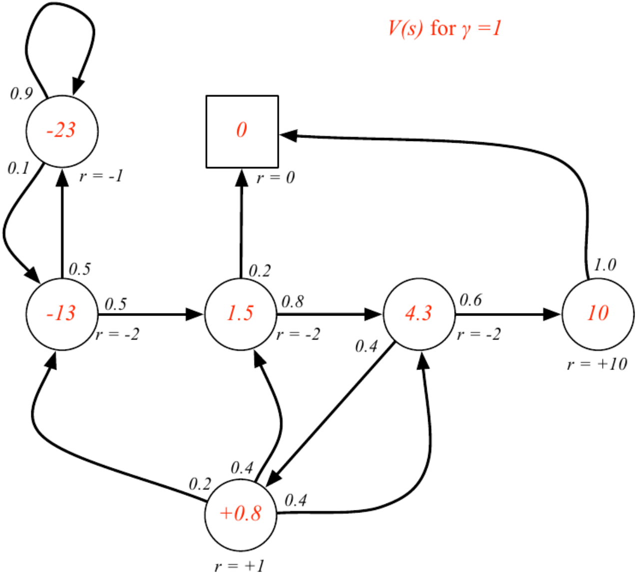
$$V_t(\mathbf{x}) := \mathbb{E}\left[\mathfrak{q}(\mathbf{x}_T) + \sum_{\tau=t}^{T-1} \ell(\mathbf{x}_\tau) \mid \mathbf{x}_t = \mathbf{x}\right]$$

- **Infinite-horizon**:
  - **First-exit**: trajectories terminate at the first passage time $T := \inf\{t \in \mathbb{N} | \mathbf{x}_t \in \mathcal{T}\}$ to a terminal state $\mathbf{x}_t \in \mathcal{T} \subseteq \mathcal{X}$
  - **Discounted**: trajectories continue forever but the costs are discounted by a **discount factor** factor $\gamma \in [0, 1)$ that specified the present value of future costs:
    - γ close to 0 leads to myopic/greedy evaluation
    - γ close to 1 leads to nonmyopic/far-sighted evaluation
    - Mathematically convenient since it avoids infinite costs as $T \to \infty$
  - **Average-cost**: trajectories continue forever and the value function is the expected average stage cost

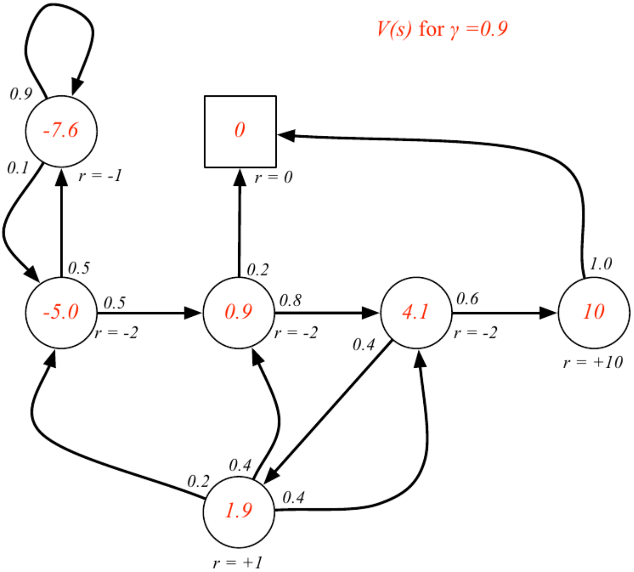
Example: Cumulative Reward of the Student MRP



Example: Cumulative Reward of the Student MRP



Example: Cumulative Reward of the Student MRP



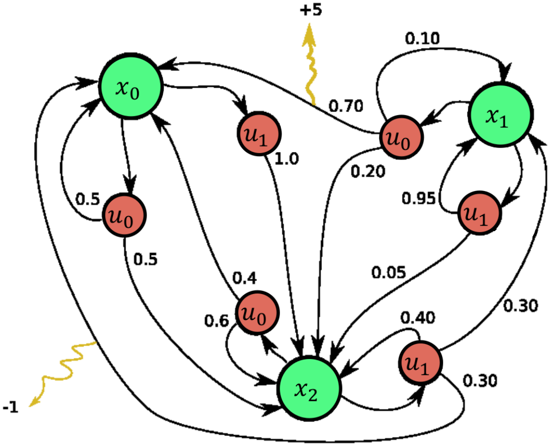
Markov Decision Process

A Markov Decision Process (MDP) is a Markov Reward Process with controlled transitions defined by a tuple $(\mathcal{X}, \mathcal{U}, p_0, p_f, T, \ell, \mathfrak{q}, \gamma)$

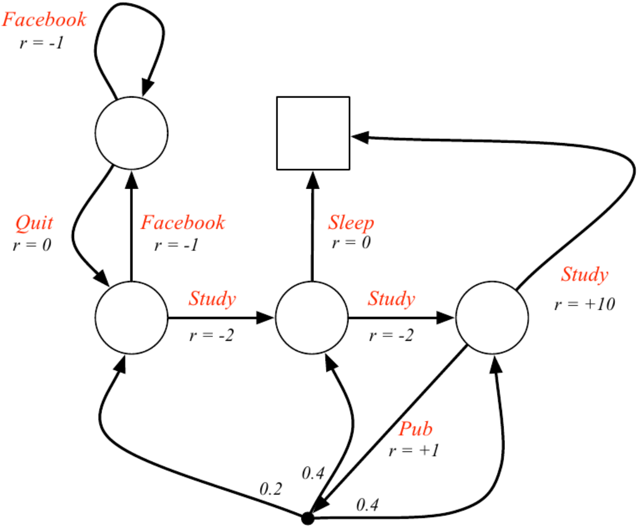
- $\mathcal{X}$ is a discrete/continuous set of states
- $\mathcal{U}$ is a discrete/continuous set of controls
- p_0 is a prior pmf/pdf defined on $\mathcal{X}$
- $p_f(\cdot \mid \mathbf{x}_t, \mathbf{u}_t)$ is a conditional pmf/pdf defined on $\mathcal{X}$ for given $\mathbf{x}_t \in \mathcal{X}$ and $\mathbf{u}_t \in \mathcal{U}$ and summarized as matrices $P_{ij}^u := p_f(j \mid x_t = i, u_t = u)$ in the finite-dimensional case
- T is a finite/infinite time horizon
- $\ell(\mathbf{x}, \mathbf{u})$ is a function specifying the cost of applying control $\mathbf{u} \in \mathcal{U}$ in state $\mathbf{x} \in \mathcal{X}$
- $\mathfrak{q}(\mathbf{x})$ is a terminal cost of being in state $\mathbf{x}$ at time T
- $\gamma \in [0, 1]$ is a discount factor

Example: Markov Decision Process

- A control u_t applied in state x_t determines the next state x_{t+1} and the obtained cost $\ell(x_t, u_t)$



Example: Student Markov Decision Process



Control Policy and Value Function

- **Control policy**: a function π that maps a time step $t \in \mathbb{N}$ and a state $\mathbf{x} \in \mathcal{X}$ to a feasible control input $\mathbf{u} \in \mathcal{U}$

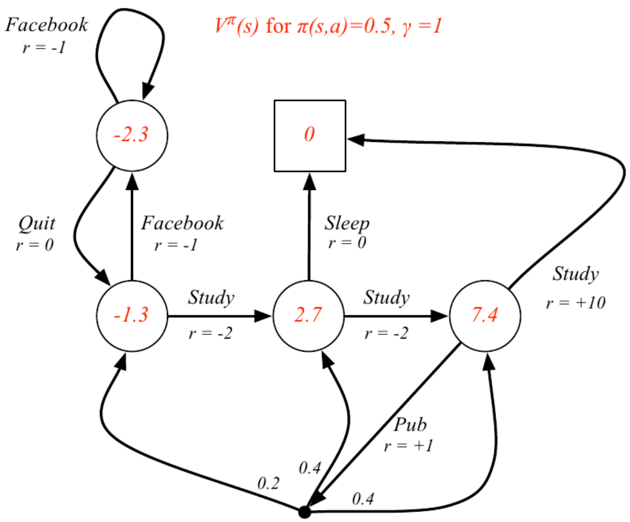
- **Value function**: expected cumulative cost of a policy π applied to an MDP with initial state $\mathbf{x} \in \mathcal{X}$ at time t:

- **Finite-horizon**: trajectories terminate at fixed $T < \infty$:

$$V_t^{\pi}(\mathbf{x}) := \mathbb{E}\left[\mathfrak{q}(\mathbf{x}_T) + \sum_{\tau=t}^{T-1} \ell(\mathbf{x}_\tau, \pi_\tau(\mathbf{x}_\tau)) \mid \mathbf{x}_t = \mathbf{x}\right]$$

- **Infinite-horizon**: as $T \to \infty$, optimal policies become stationary, i.e., $\pi := \pi_0 \equiv \pi_1 \equiv \cdots$
  - **First-exit**: trajectories terminate at the first passage time $T := \inf\{t \in \mathbb{N} | \mathbf{x}_t \in \mathcal{T}\}$ to a terminal state $\mathbf{x}_t \in \mathcal{T} \subseteq \mathcal{X}$
  - **Discounted**: trajectories continue forever but the costs are discounted by a factor $\gamma \in [0, 1)$
  - **Average-cost**: trajectories continue forever and the value function is the expected average stage cost

Example: Value Function of Student MDP



Alternative Cost Formulations

- **Noise-dependent costs**: allow stage costs ℓ' to depend on the motion noise $\mathbf{w}_t$:

$$V_0^\pi(\mathbf{x}) := \mathbb{E}_{\mathbf{w}_{0:T},\mathbf{x}_{1:T}}\left[\mathfrak{q}(\mathbf{x}_T) + \sum_{t=0}^{T-1} \ell'(\mathbf{x}_t, \pi_t(\mathbf{x}_t), \mathbf{w}_t) \mid \mathbf{x}_0 = \mathbf{x}\right]$$

- Using the pdf $p_w(\cdot \mid \mathbf{x}_t, \mathbf{u}_t)$ of $\mathbf{w}_t$, this is equivalent to our formulation:

$$\ell(\mathbf{x}_t, \mathbf{u}_t) := \mathbb{E}_{\mathbf{w}_t \mid \mathbf{x}_t, \mathbf{u}_t}\left[\ell'(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t)\right] = \int \ell(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) p_w(\mathbf{w}_t \mid \mathbf{x}_t, \mathbf{u}_t) d\mathbf{w}_t$$

  The expectation can be computed if p_w is known or approximated.

- **Joint cost-state pdf**: allow random costs ℓ' with joint pdf $p(\mathbf{x}', \ell' \mid \mathbf{x}, \mathbf{u})$. This is equivalent to our formulation as follows:

$$p_f(\mathbf{x}' \mid \mathbf{x}, \mathbf{u}) := \int p(\mathbf{x}', \ell' \mid \mathbf{x}, \mathbf{u}) d\ell'$$

$$\ell(\mathbf{x}, \mathbf{u}) := \mathbb{E}\left[\ell' \mid \mathbf{x}, \mathbf{u}\right] = \int\int \ell' p(\mathbf{x}', \ell' \mid, \mathbf{x}, \mathbf{u}) d\mathbf{x}' d\ell'$$

Alternative Motion-Model Formulations

- **Time-lag motion model**: $\mathbf{x}_{t+1} = f_t(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{u}_{t-1}, \mathbf{w}_t)$
- Can be converted to the standard form via **state augmentation**
- Let $\mathbf{y}_t := \mathbf{x}_{t-1}$ and $\mathbf{s}_t := \mathbf{u}_{t-1}$ and define the augmented dynamics:

$$\tilde{\mathbf{x}}_{t+1} := \begin{bmatrix} \mathbf{x}_{t+1} \\ \mathbf{y}_{t+1} \\ \mathbf{s}_{t+1} \end{bmatrix} = \begin{bmatrix} f_t(\mathbf{x}_t, \mathbf{y}_t, \mathbf{u}_t, \mathbf{s}_t, \mathbf{w}_t) \\ \mathbf{x}_t \\ \mathbf{u}_t \end{bmatrix} =: \tilde{f}_t(\tilde{\mathbf{x}}_t, \mathbf{u}_t, \mathbf{w}_t)$$

- Note that this procedure works for an arbitrary number of time lags but the dimension of the state space grows and increases the computational burden exponentially ("curse of dimensionality")

Alternative Motion-Model Formulations

- System dynamics: $\mathbf{x}_{t+1} = f_t(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t)$
- **Correlated Disturbance**: $\mathbf{w}_t$ correlated across time (colored noise):

$$\mathbf{y}_{t+1} = A_t \mathbf{y}_t + \boldsymbol{\xi}_t$$
$$\mathbf{w}_t = C_t \mathbf{y}_{t+1}$$

  where A_t, C_t are known and $\boldsymbol{\xi}_t$ are independent random variables

- **Augmented state**: $\tilde{\mathbf{x}}_t := (\mathbf{x}_t, \mathbf{y}_t)$ with dynamics:

$$\tilde{\mathbf{x}}_{t+1} = \begin{bmatrix} \mathbf{x}_{t+1} \\ \mathbf{y}_{t+1} \end{bmatrix} = \begin{bmatrix} f_t(\mathbf{x}_t, \mathbf{u}_t, C_t(A_t \mathbf{y}_t + \boldsymbol{\xi}_t)) \\ A_t \mathbf{y}_t + \boldsymbol{\xi}_t \end{bmatrix} =: \tilde{f}_t(\tilde{\mathbf{x}}_t, \mathbf{u}_t, \boldsymbol{\xi}_t)$$

- **State estimator**: note that $\mathbf{y}_t$ must be observed at time t, which can be done using a state estimator

Hidden Markov Model

A Hidden Markov Model (HMM) is a Markov Chain with partially observable states defined by a tuple $(\mathcal{X}, \mathcal{Z}, p_0, p_f, p_h, T)$

- $\mathcal{X}$ is a discrete/continuous set of states
- $\mathcal{Z}$ is a discrete/continuous set of observations
- p_0 is a prior pmf/pdf defined on $\mathcal{X}$
- $p_f(\cdot \mid \mathbf{x}_t)$ is a conditional pmf/pdf defined on $\mathcal{X}$ for given $\mathbf{x}_t \in \mathcal{X}$ and summarized by a matrix $P_{ij} := p_f(j \mid x_t = i)$ in the finite-dim case
- $p_h(\cdot \mid \mathbf{x}_t)$ is a conditional pmf/pdf defined on $\mathcal{Z}$ for given $\mathbf{x}_t \in \mathcal{X}$ and summarized by a matrix $O_{ij} := p_h(j \mid x_t = i)$ in the finite-dim case
- T is a finite/infinite time horizon

Partially Observable Markov Decision Process

A Partially Observable Markov Decision Process (POMDP) is an MDP with partially observable states defined by a tuple $(\mathcal{X}, \mathcal{U}, \mathcal{Z}, p_0, p_f, p_h, T, \ell, \mathfrak{q}, \gamma)$

- $\mathcal{X}$ is a discrete/continuous set of states
- $\mathcal{U}$ is a discrete/continuous set of controls
- $\mathcal{Z}$ is a discrete/continuous set of observations
- p_0 is a prior pmf/pdf defined on $\mathcal{X}$
- $p_f(\cdot \mid \mathbf{x}_t, \mathbf{u}_t)$ is a conditional pmf/pdf defined on $\mathcal{X}$ for given $\mathbf{x}_t \in \mathcal{X}$ and $\mathbf{u}_t \in \mathcal{U}$ and summarized by a matrix $P^u_{ij} := p_f(j \mid x_t = i, u_t = u)$ in the finite-dim case
- $p_h(\cdot \mid \mathbf{x}_t)$ is a conditional pmf/pdf defined on $\mathcal{Z}$ for given $\mathbf{x}_t \in \mathcal{X}$ and summarized by a matrix $O_{ij} := p_h(j \mid x_t = i)$ in the finite-dim case
- T is a finite/infinite time horizon
- $\ell(\mathbf{x}, \mathbf{u})$ is a function specifying the cost of applying control $\mathbf{u} \in \mathcal{U}$ in state $\mathbf{x} \in \mathcal{X}$
- $\mathfrak{q}(\mathbf{x})$ is a terminal cost of being in state $\mathbf{x}$ at time T
- $\gamma \in [0, 1]$ is a discount factor

Comparison of Markov Models

	observed	partially observed
uncontrolled	**Markov Chain/MRP**	**HMM**
controlled	**MDP**	**POMDP**

- Markov Chain + Partial Observability = HMM
- Markov Chain + Control = MDP
- Markov Chain + Partial Observability + Control = HMM + Control = MDP + Partial Observability = POMDP

Bayes Filter

- A probabilistic inference technique for summarizing information $\mathbf{i}_t := (\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1}) \in \mathcal{I}$ about a partially observable state $\mathbf{x}_t$

- The Bayes filter keeps track of:
$$p_{t|t}(\mathbf{x}_t) := p(\mathbf{x}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1})$$
$$p_{t+1|t}(\mathbf{x}_{t+1}) := p(\mathbf{x}_{t+1} \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t})$$

- Derived using total probability, conditional probability, and Bayes rule based on the motion and observation models of the system

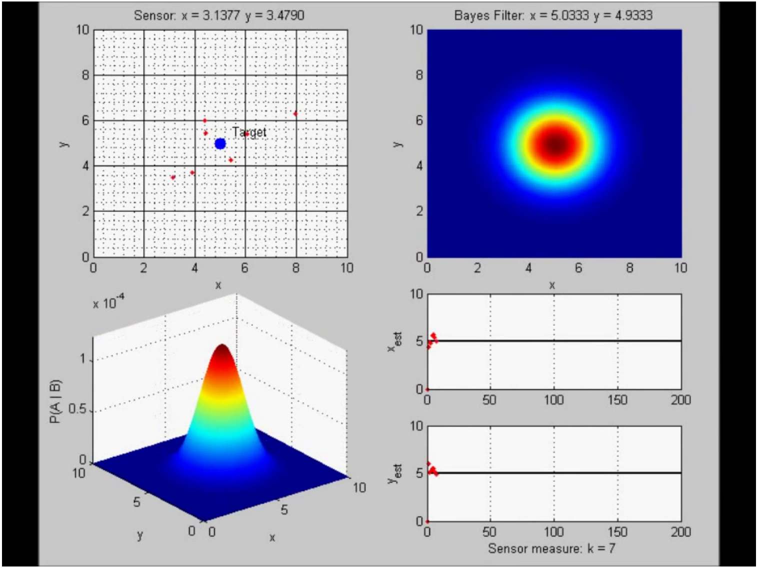
- **Motion model**: $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \sim p_f(\cdot \mid \mathbf{x}_t, \mathbf{u}_t)$

- **Observation model**: $\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{v}_t) \sim p_h(\cdot \mid \mathbf{x}_t)$

- **Bayes filter**: consists of **predict** and **update** steps:

$$p_{t+1|t+1}(\mathbf{x}_{t+1}) = \underbrace{\overbrace{\frac{1}{p(\mathbf{z}_{t+1}|\mathbf{z}_{0:t}, \mathbf{u}_{0:t})}}^{\frac{1}{\eta_{t+1}}} p_h(\mathbf{z}_{t+1} \mid \mathbf{x}_{t+1}) \overbrace{\int p_f(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_t) p_{t|t}(\mathbf{x}_t) d\mathbf{x}_t}^{\textbf{Predict: } p_{t+1|t}(\mathbf{x}_{t+1})}}_{\textbf{Update}}$$

Bayes Filter Example



Equivalence of POMDPs and MDPs

- A POMDP $(\mathcal{X}, \mathcal{U}, \mathcal{Z}, p_0, p_f, p_h, T, \ell, \mathfrak{q}, \gamma)$ is equivalent to an MDP $(\mathcal{P}(\mathcal{X}), \mathcal{U}, p_0, p_\psi, T, \bar{\ell}, \bar{\mathfrak{q}}, \gamma)$ such that:
  - **State space**: $\mathcal{P}(\mathcal{X})$ is the **continuous** space of pdfs/pmfs over $\mathcal{X}$
    - If $\mathcal{X}$ is continuous, then $\mathcal{P}(\mathcal{X}) := \left\{p : \mathcal{X} \to \mathbb{R}_{\geq 0} \mid \int p(\mathbf{x})d\mathbf{x} = 1\right\}$
    - If $|\mathcal{X}| = N$, then $\mathcal{P}(\mathcal{X}) := \{\mathbf{p} \in [0,1]^N \mid \mathbf{1}^\top \mathbf{p} = 1\}$
  - **Initial state**: $p_0 \in \mathcal{P}(\mathcal{X})$
  - **Motion model**: the Bayes filter $p_{t+1|t+1} = \psi(p_{t|t}, \mathbf{u}_t, \mathbf{z}_{t+1})$ plays the role of a motion model with the observations $\mathbf{z}_{t+1}$ acting as noise with density:

$$\eta(\mathbf{z} \mid p_{t|t}, \mathbf{u}_t) := \int \int p_h(\mathbf{z} \mid \mathbf{x}_{t+1}) p_f(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_t) p_{t|t}(\mathbf{x}_t) d\mathbf{x}_t d\mathbf{x}_{t+1}$$

  - **Cost**: the transformed stage and terminal cost functions are the expected values of the original ones:

$$\bar{\ell}(p, \mathbf{u}) := \int \ell(\mathbf{x}, \mathbf{u}) p(\mathbf{x}) d\mathbf{x} \qquad \bar{\mathfrak{q}}(p) := \int \mathfrak{q}(\mathbf{x}) p(\mathbf{x}) d\mathbf{x}$$

Optimal Control in a POMDP

- An infinite-dimensional stochastic optimization problem for a POMDP $(\mathcal{X}, \mathcal{U}, \mathcal{Z}, p_0, p_f, p_h, T, \ell, \mathfrak{q}, \gamma)$:

$$\begin{aligned}
\min_{\pi_{0:T-1}} \quad & \mathbb{E}\left[\gamma^T \mathfrak{q}(\mathbf{x}_T) + \sum_{t=0}^{T-1} \gamma^t \ell(\mathbf{x}_t, \mathbf{u}_t)\right] \\
\text{s.t.} \quad & \mathbf{x}_{t+1} \sim p_f(\cdot \mid \mathbf{x}_t, \mathbf{u}_t), && t = 0, \ldots, T-1 \\
& \mathbf{z}_{t+1} \sim p_h(\cdot \mid \mathbf{x}_t), && t = 0, \ldots, T-1 \\
& \mathbf{u}_t \sim \pi_t(\cdot \mid \mathbf{i}_t), && t = 0, \ldots, T-1 \\
& \mathbf{x}_0 \sim p_0(\cdot)
\end{aligned}$$

- The equivalent MDP $(\mathcal{P}(\mathcal{X}), \mathcal{U}, p_0, p_\psi, T, \bar{\ell}, \bar{\mathfrak{q}}, \gamma)$ with state $p_{t|t}$ leads to the problem:

$$\begin{aligned}
\min_{\pi_{0:T-1}} \quad & V_0^\pi(p_0) = \mathbb{E}\left[\gamma^T \bar{\mathfrak{q}}(p_{T|T}) + \sum_{t=0}^{T-1} \gamma^t \bar{\ell}(p_{t|t}, \mathbf{u}_t)\right] \\
\text{s.t.} \quad & p_{t+1|t+1} = \psi(p_{t|t}, \mathbf{u}_t, \mathbf{z}_{t+1}), && t = 0, \ldots, T-1 \\
& \mathbf{z}_{t+1} \sim \eta(\cdot \mid p_{t|t}, \mathbf{u}_t), && t = 0, \ldots, T-1 \\
& u_t \sim \pi_t(\cdot \mid p_{t|t}), && t = 0, \ldots, T-1
\end{aligned}$$

Finite-horizon Optimal Control in an MDP

Finite-horizon Optimal Control

The finite-horizon optimal control problem in an MDP $(\mathcal{X}, \mathcal{U}, p_0, p_f, T, \ell, \mathfrak{q}, \gamma)$ with initial state $\mathbf{x}$ at time t is:

$$\begin{aligned}
\min_{\pi_{t:T-1}} \; & V_t^{\pi}(\mathbf{x}) := \mathbb{E}_{\mathbf{x}_{t+1:T}} \left[\gamma^{T-t} \mathfrak{q}(\mathbf{x}_T) + \sum_{\tau=t}^{T-1} \gamma^{\tau-t} \ell(\mathbf{x}_\tau, \pi_\tau(\mathbf{x}_\tau)) \;\middle|\; \mathbf{x}_t = \mathbf{x} \right] \\
\text{s.t.} \; & \mathbf{x}_{\tau+1} \sim p_f(\cdot \mid \mathbf{x}_\tau, \pi_\tau(\mathbf{x}_\tau)), \qquad \tau = t, \ldots, T-1 \\
& \mathbf{x}_\tau \in \mathcal{X}, \;\; \pi_\tau(\mathbf{x}_\tau) \in \mathcal{U}
\end{aligned}$$

- Due to the equivalence between POMDPs and MDPs, we will focus exclusively on MDPs

Open-Loop vs Closed-Loop Control

- There are two different control methodologies:
  - **Open loop**: control inputs $\mathbf{u}_{0:T-1}$ are determined at once at time 0 as a function of $\mathbf{x}_0$ and do not change online depending on $\mathbf{x}_t$
  - **Closed loop**: control inputs are determined "just-in-time" as a function π_t of the current state $\mathbf{x}_t$
- Open-loop control is a special case of closed-loop control that disregards the states $\mathbf{x}_t$ and, hence, can never give better performance.
- In the absence of disturbances (and in a special linear quadratic Gaussian case), open-loop and closed-loop control give theoretically the same performance

Open-Loop vs Closed-Loop Control

- **Open-loop feedback control** (OLFC) recomputes a new open-loop sequence $\mathbf{u}_{t:T-1}$ online, whenever a new state $\mathbf{x}_t$ is available. OLFC is guaranteed to perform better than open-loop control and is computationally more efficient to obtain than closed-loop control.
- Open-loop control is computationally much cheaper than closed-loop control
- Consider a discrete-space example with $|\mathcal{X}| = 10$ states, $|\mathcal{U}| = 10$ control inputs, planning horizon $T = 4$, and given x_0:
  - There are $|\mathcal{U}|^T = 10^4$ different open-loop strategies
  - There are $|\mathcal{U}|(|\mathcal{U}|^{|\mathcal{X}|})^{T-1} = |\mathcal{U}|^{|\mathcal{X}|(T-1)+1} = 10^{31}$ different closed-loop strategies

Example: Chess Strategy Optimization

- **Objective**: come up with a strategy that maximizes the chances of winning a 2 game chess match.
- Possible outcomes:
  - Win/Lose: 1 point for the winner, 0 for the loser
  - Draw: 0.5 points for each player
  - If the score is equal after 2 games, the players continue playing until one wins (sudden death)
- Playing styles:
  - **Timid**: draw with probability p_d and lose with probability $(1 - p_d)$
  - **Bold**: win with probability p_w and lose with probability $(1 - p_w)$
  - **Assumption**: $p_d > p_w$

Finite-state Model of the Chess Match

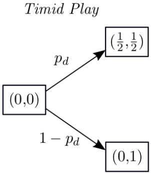
- The **state** $\mathbf{x}_t$ is a 2-D vector with our and the opponent's score after the t-th game
- The **control** u_t is the play style: $\mathcal{U} = \{\text{timid, bold}\}$
- The **noise** w_t is the score of the next game
- Since timid play does not make sense during the sudden death stage, the planning horizon is $T = 2$
- We can construct a **time-dependent motion model** P^u_{ijt} for $t \in \{0, 1\}$ (shown on the next slide)
- **Cost**: minimize loss probability: $-P_{win} = \mathbb{E}_{\mathbf{x}_{1:2}}\left[\mathfrak{q}(\mathbf{x}_2) + \sum_{t=0}^{1} \ell(\mathbf{x}_t, u_t)\right]$,

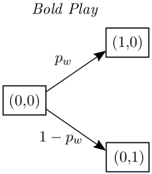
  where

$$\ell(\mathbf{x}, u) = 0 \quad \text{and} \quad \mathfrak{q}(\mathbf{x}) = \begin{cases} -1 & \text{if } \mathbf{x} = \left(\frac{3}{2}, \frac{1}{2}\right) \text{ or } (2, 0) \\ -p_w & \text{if } \mathbf{x} = (1, 1) \\ 0 & \text{if } \mathbf{x} = \left(\frac{1}{2}, \frac{3}{2}\right) \text{ or } (0, 2) \end{cases}$$

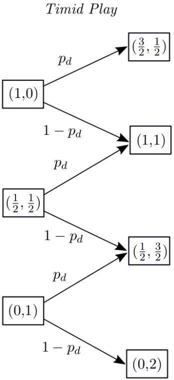
Chess Transition Probabilities

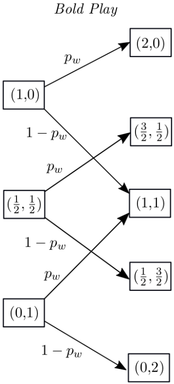
Game 1:





Game 2:





Open-Loop Chess Strategy

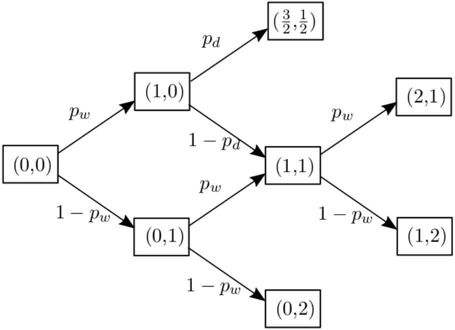
- There are 4 admissible open-loop policies:
  1. timid-timid: $P_{win} = p_d^2 p_w$
  2. bold-bold: $P_{win} = p_w^2 + p_w(1-p_w)p_w + (1-p_w)p_w p_w = p_w^2(3-2p_w)$
  3. bold-timid: $P_{win} = p_w p_d + p_w(1-p_d)p_w$
  4. timid-bold: $P_{win} = p_d p_w + (1-p_d)p_w^2$
- Since $p_d^2 p_w \le p_d p_w \le p_d p_w + (1-p_d)p_w^2$, timid-timid is not optimal
- The best achievable winning probability is:

$$\begin{aligned} P^*_{win} &= \max\{\overbrace{p_w^2(3-2p_w)}^{\text{bold-bold}}, \overbrace{p_d p_w + (1-p_d)p_w^2}^{\text{3. or 4.}}\} \\ &= p_w^2 + p_w(1-p_w)\max\{2p_w, p_d\} \end{aligned}$$

- In the open-loop case, if $p_w \le 0.5$, then $P^*_{win} \le 0.5$
  - For $p_w = 0.45$ and $p_d = 0.9$, $P^*_{win} = 0.43$
  - For $p_w = 0.5$ and $p_d = 1.0$, $P^*_{win} = 0.5$
- If $p_d > 2p_w$, bold-timid and timid-bold are optimal open-loop policies; otherwise bold-bold is optimal

Closed-Loop Chess Strategy

- There are 16 admissible policies
- Consider one option: play timid if and only if ahead (it will turn out that this is optimal)



- The probability of winning is:
  $P_{win} = p_d p_w + p_w((1-p_d)p_w + p_w(1-p_w)) = p_w^2(2-p_w) + p_w(1-p_w)p_d$
- Note that in the closed-loop case we can achieve P_{win} larger than 0.5 even when p_w is less than 0.5:
  - For $p_w = 0.45$ and $p_d = 0.9$, $P_{win} = 0.5$
  - For $p_w = 0.5$ and $p_d = 1.0$, $P_{win} = 0.625$