Reinforcement Learning Fundamentals

Lecture 10: Markov Decision Process (MDP)

Dr Sandeep Manjanna
Assistant Professor, Plaksha University
sandeep.manjanna@plaksha.edu.in

In today’s class…

- Markov Process
- Markov Reward Process
- Markov Decision Process (MDP)
- Problem to formulation
- Examples

Schedule for Evaluation

Date	Evaluation	Description
2/16/2024	Finalizing Project	Finalize Team, and Project topic
2/23/2024	Project Proposal (5%)	2-page report for Project Proposal. This document is expected to include following but not limited to: • 1/2 page for introduction and related work, 1 page for the problem and the proposed work, 1/4 page for proposed evaluation, 1/4 page for references. Format will be shared with you.
2/28/2024	Quiz 2	
3/15/2024	In-class Exam 1 (10%)	
3/25/2024	Mid Term Progress Report (5%)	4-page report for Mid-term Progress Report. This document is expected to include following but not limited to: • The first two pages contain a copy of your project proposal. The remaining pages include: a status update, presenting what you have accomplished so far (include figures and results), and 1/4 page describing your next steps.
Mar 28th and 29th	Mid Term Presentation (5%)	Progress presentation
4/3/2024	Quiz 3	
4/24/2024	Quiz4	
5/3/2024	Final Project Submission (25%)	Include full code in git hub, 6 page report, multi-media with demo if required. The format for the report will be provided later
May 6th to 10th	Final Project Presentation (10%)	Final Presentation Details will be shared Later
May	In-class Exam 2 (20%)	

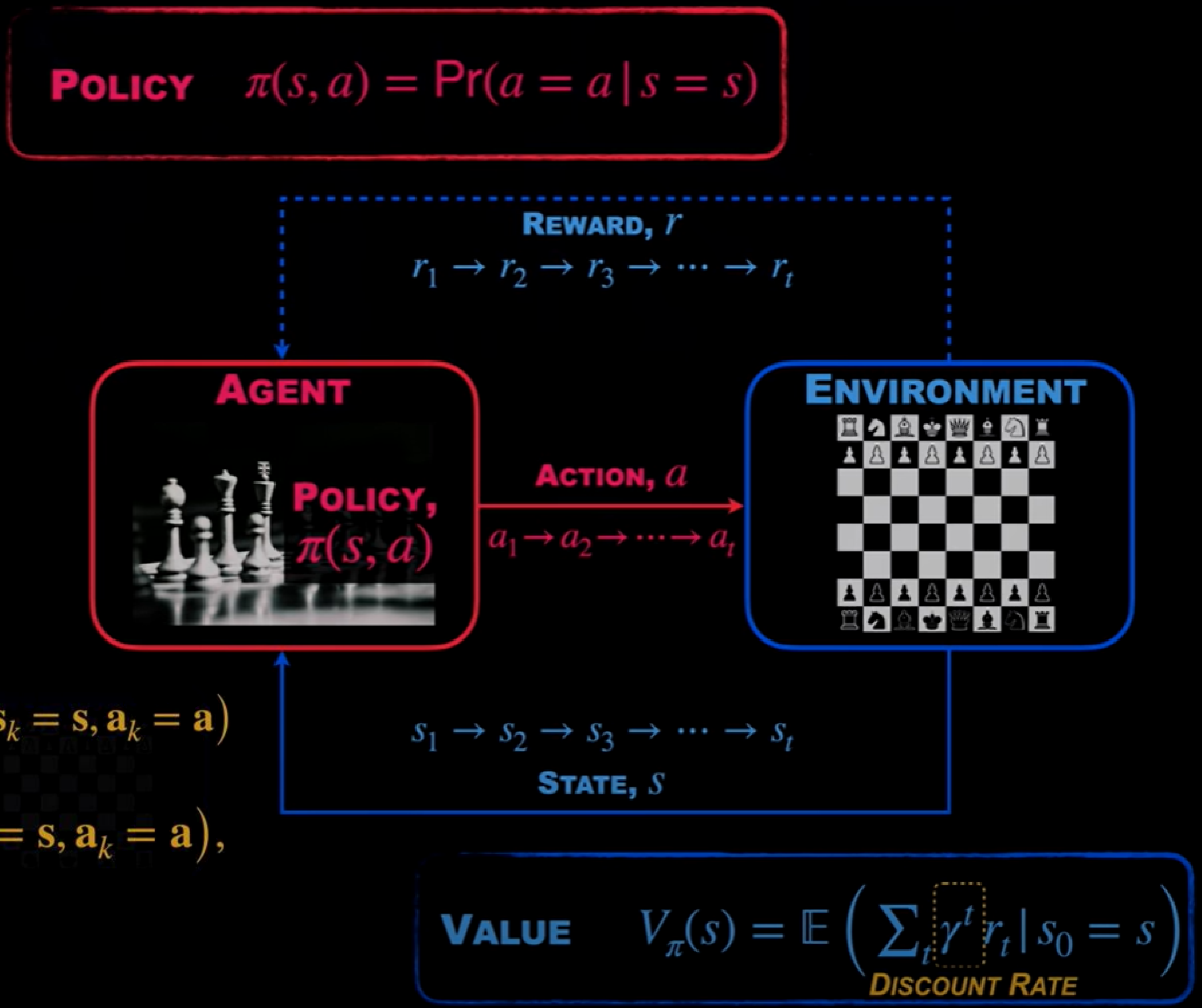
Inside an RL Agent
Model

- A model predicts what the environment will do next
- $\mathcal{P}$ predicts the next state
- $\mathcal{R}$ predicts the next (immediate) reward, e.g.

Transition Model → $\mathcal{P}_{ss'}^{a} = \mathbb{P}[S_{t+1} = s' \mid S_t = s, A_t = a]$

Reward Function / Return → $\mathcal{R}_{s}^{a} = \mathbb{E}\left[R_{t+1} \mid S_t = s, A_t = a\right]$

Inside an RL Agent



$$R(\mathbf{s}', \mathbf{s}, \mathbf{a}) = \mathbf{Pr}\left(\mathbf{r}_{k+1} \mid \mathbf{s}_{k+1} = \mathbf{s}', \mathbf{s}_k = \mathbf{s}, \mathbf{a}_k = \mathbf{a}\right)$$

$$P(\mathbf{s}', \mathbf{s}, \mathbf{a}) = \mathbf{Pr}\left(\mathbf{s}_{k+1} = \mathbf{s}' \mid \mathbf{s}_k = \mathbf{s}, \mathbf{a}_k = \mathbf{a}\right),$$

Dr Steve Brunton’s course : Reinforcement Learning.

Markov Property

- “the state” at time t, means whatever information about the environment that is available to the agent at time t.

- The state can include immediate observations, highly processed observations, and structures built over time from a sequence of observations.

- Ideally, a state should summarize past observations so as to retain all essential information.

- “The future is independent of the past given the present”

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, \ldots, S_t]$$

  - The state captures all relevant information from the history
  - Once the state is known, the history may be thrown away
  - i.e. The state is a sufficient statistic of the future

State Transition Matrix

For a Markov state s and successor state s', the *state transition probability* is defined by

$$\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$$

State transition matrix $\mathcal{P}$ defines transition probabilities from all states s to all successor states s',

$$\mathcal{P} = \textit{from} \quad \begin{array}{c} \textit{to} \\ \begin{bmatrix} \mathcal{P}_{11} & \ldots & \mathcal{P}_{1n} \\ \vdots & & \\ \mathcal{P}_{n1} & \ldots & \mathcal{P}_{nn} \end{bmatrix} \end{array}$$

where each row of the matrix sums to 1.

Slide from David Silver’s course: “Reinforcement Learning.”

Markov Process

A Markov process is a memoryless random process, i.e. a sequence of random states $S_1, S_2, \ldots$ with the Markov property.

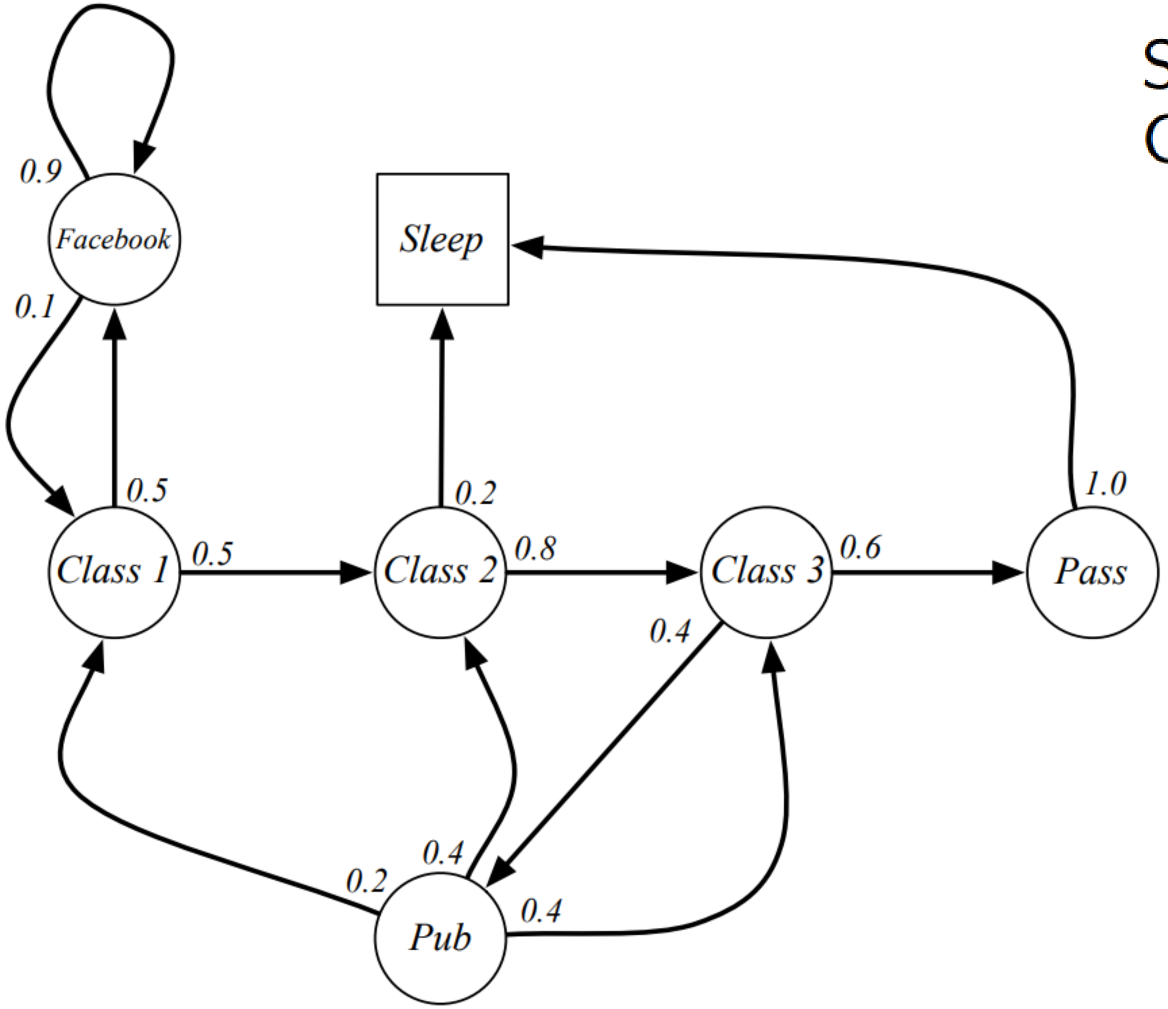
Definition

A *Markov Process* (or *Markov Chain*) is a tuple $\langle \mathcal{S}, \mathcal{P} \rangle$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{P}$ is a state transition probability matrix,
  $\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$

Slide from David Silver's course: "Reinforcement Learning."

Markov Process Example



Sample episodes for Student Markov Chain starting from $S_1 = C1$

$$S_1, S_2, ..., S_T$$

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C2 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep
- C1 FB FB C1 C2 C3 Pub C1 FB FB FB C1 C2 C3 Pub C2 Sleep

Slide from David Silver's course: "Reinforcement Learning."

Markov Reward Process Example

A Markov reward process is a Markov chain with values.

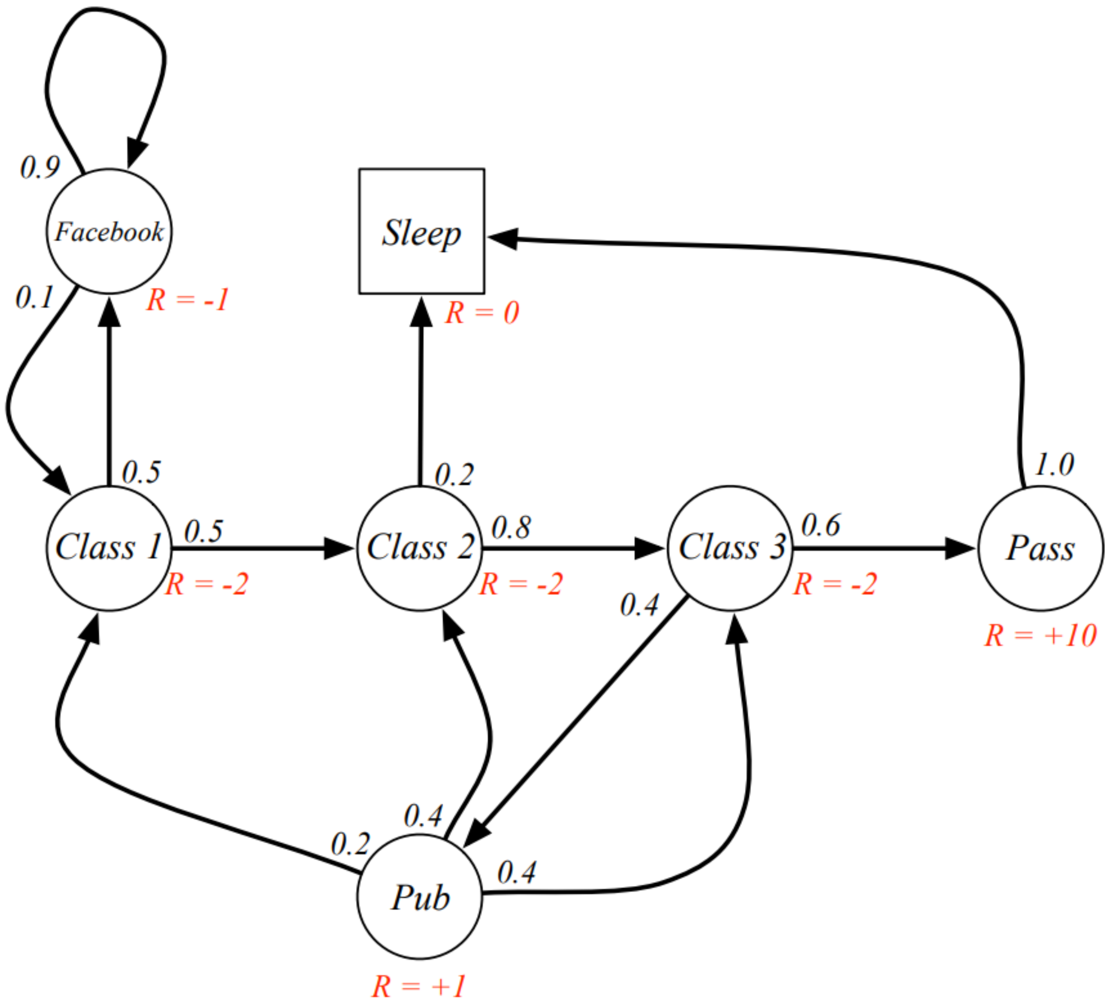
Definition

A *Markov Reward Process* is a tuple $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$

- $\mathcal{S}$ is a finite set of states
- $\mathcal{P}$ is a state transition probability matrix,
  $\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$
- $\mathcal{R}$ is a reward function, $\mathcal{R}_s = \mathbb{E}\left[R_{t+1} \mid S_t = s\right]$
- γ is a discount factor, $\gamma \in [0, 1]$

Slide from David Silver's course: "Reinforcement Learning."

Markov Reward Process Example



What will be the return for each of these samples?

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C2 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep
- C1 FB FB C1 C2 C3 Pub C1 FB FB FB C1 C2 C3 Pub C2 Sleep

Slide from David Silver's course: "Reinforcement Learning."

Markov Decision Process

- MDP, *M*, is the tuple: $M = \langle S, \boxed{A}, p, r \rangle$
  - S : set of states.
  - A : set of actions.
  - $p : S \times A \times S \rightarrow [0,1]$: probability of transition.
  - $r : S \times A \times S \rightarrow \mathbb{R}$: expected reward.

$$\mathcal{P}_{ss'}^{a} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s, A_t = a\right]$$

$$\mathcal{R}_{s}^{a} = \mathbb{E}\left[R_{t+1} \mid S_t = s, A_t = a\right]$$

- Policy: $\pi : S \times A \rightarrow [0,1]$ (can be deterministic)
- Maximize total expected reward
- Learn an *optimal* policy

How to compute the expected reward?

1. Discrete distribution over r:
2. R is from Real numbers:

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Markov Decision Process

- MDP, *M*, is the tuple: $M = \langle S, \boxed{A}, p, r \rangle$
  - S : set of states.
  - A : set of actions.
  - $p : S \times A \times S \to [0, 1]$: probability of transition.
  - $r : S \times A \times S \to \mathbb{R}$: expected reward.
- Policy: $\pi : S \times A \to [0,1]$ (can be deterministic)
- Maximize total expected reward
- Learn an *optimal* policy

***π (a|s)* or *π (s,a)* = ?**

What does it mean for policy to be deterministic?

Slide from Prof. Ravindran's course: "Reinforcement Learning."

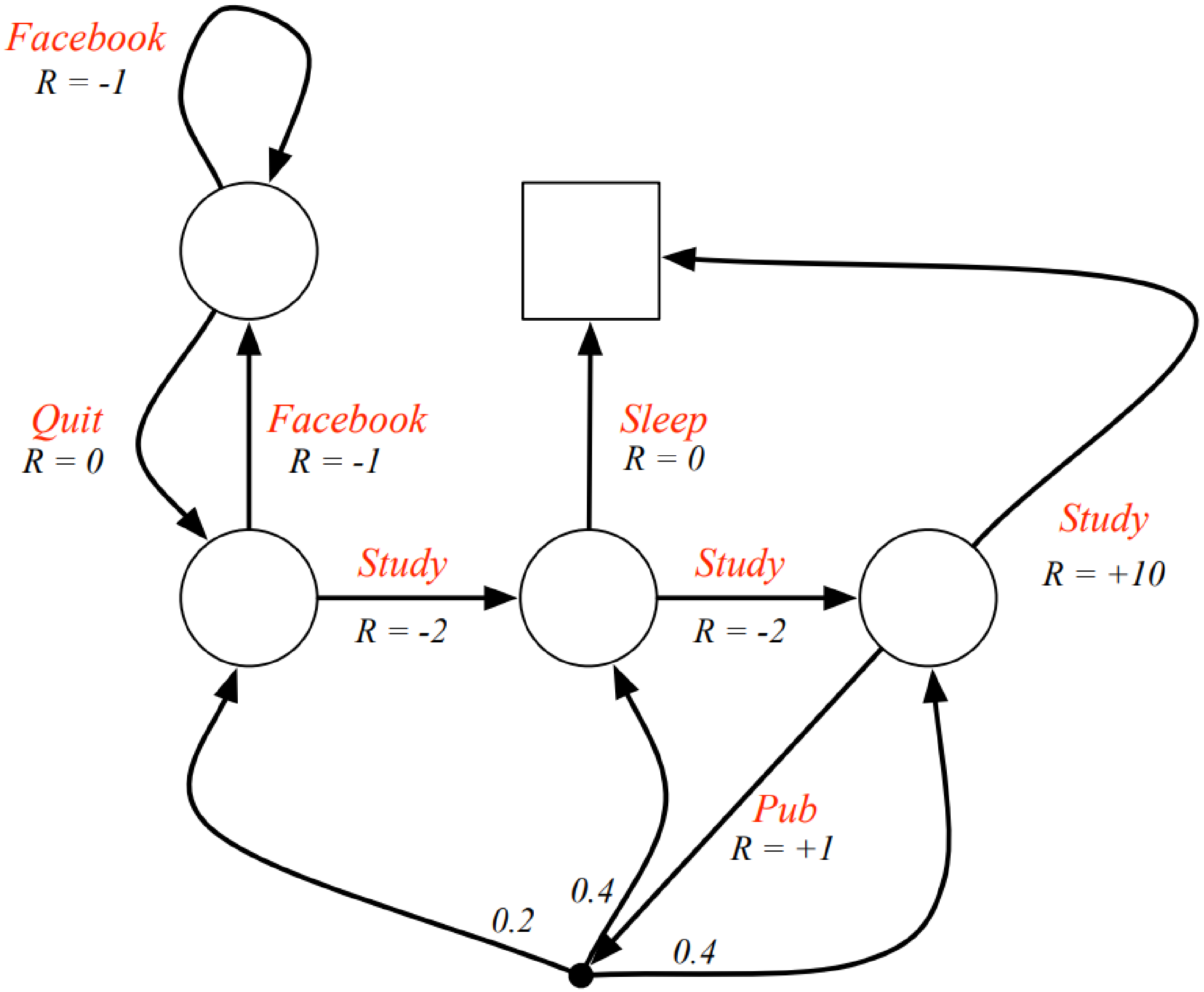
Markov Decision Process

- MDP, *M*, is the tuple: $M = \langle S, \boxed{A}, p, r \rangle$
  - S : set of states.
  - A : set of actions.
  - $p : S \times A \times S \rightarrow [0, 1]$: probability of transition.
  - $r : S \times A \times S \rightarrow \mathbb{R}$: expected reward.
- Policy: $\pi : S \times A \rightarrow [0,1]$ (can be deterministic)
- Maximize total expected reward
- Learn an *optimal* policy

The policy that achieves the maximum total expected reward is called Optimal Policy.

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Markov Reward Process Example



Slide from David Silver's course: "Reinforcement Learning."

Formulating an RL Problem

- States
  - –Enough information to take decisions
  - –Raw inputs often not sufficient
- Actions
  - –The control variables
  - –Discrete – items to recommend, moves in a game
  - –Continuous – torque to a motor
- Rewards
  - –Define the *goal* of the problem

States must follow Markov Property

Different levels of controls in learning to drive example.

Slide from Prof. Ravindran's course: "Reinforcement Learning."