# Intro to Markov Decision Processes

## + Assignment 2 handout

Jan Mrkos

PUI Tutorial
Week 9

# Outline

- Assignment 2
- Motivation
- MDP definition and examples
- MDP solution
- Value function calculation

# Motivation

Any problems with stochastic outcomes

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.
- Robotic navigation

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.
- Robotic navigation

In addition, MDPs form a basis of many techniques in

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.
- Robotic navigation

In addition, MDPs form a basis of many techniques in

- Reinforcement Learning

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.
- Robotic navigation

In addition, MDPs form a basis of many techniques in

- Reinforcement Learning
- Game theory (extensive form games)

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Motivation

Any problems with stochastic outcomes

- Dynamic pricing: deciding on prices for products based on demand, buying price, stock
- Maintenance and repair: when to replace/inspect based on age, condition, etc. [1]
- Agriculture: how much to plant based on weather and soil state.
- Purchase and production: how much to produce based on demand.
- Robotic navigation

In addition, MDPs form a basis of many techniques in

- Reinforcement Learning
- Game theory (extensive form games)

Important extension - Partial Observable MDPs

[1] https://stats.stackexchange.com/questions/145122/real-life-examples-of-markov-decision-processes

# Markov Decision Process

Markovian:

# Markov Decision Process

Markovian:

- Named after Andrey Markov (1856 - 1922)

# Markov Decision Process

Markovian:

- Named after Andrey Markov (1856 - 1922)
- Memoryless, the next evolution of the systems depends ONLY on the current state, NOT on the sequence of events that lead to the state.

# Markov Decision Process

Markovian:

- Named after Andrey Markov (1856 - 1922)
- Memoryless, the next evolution of the systems depends ONLY on the current state, NOT on the sequence of events that lead to the state.

Decision process:

# Markov Decision Process

Markovian:

- Named after Andrey Markov (1856 - 1922)
- Memoryless, the next evolution of the systems depends ONLY on the current state, NOT on the sequence of events that lead to the state.

Decision process:

- You are expected to make a sequence of decision as responses to the changes in the environment.

# Markov Decision Process

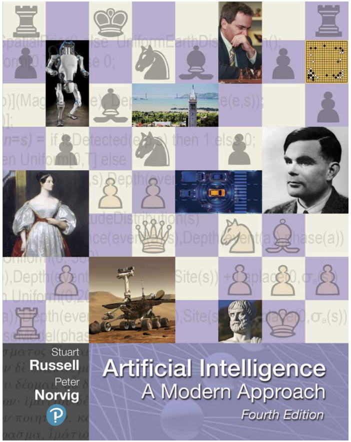
Markovian:

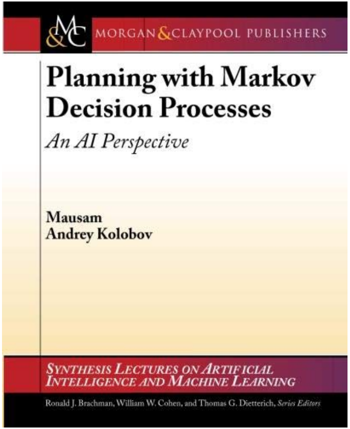
- Named after Andrey Markov (1856 - 1922)
- Memoryless, the next evolution of the systems depends ONLY on the current state, NOT on the sequence of events that lead to the state.

Decision process:

- You are expected to make a sequence of decision as responses to the changes in the environment.
- Plan vs. policy: "In planning, the problem is finding the plan. In MDP, the problem is executing the plan."

# Resources




Also, I have heard good things about the free `https://algorithmsbook.com/`.

# Markov Decision Process

Tuple $\langle S, A, D, T, R \rangle$:

- $S$: finite set of states agent can find itself in
- $A$: finite set of action agent can perform
- $D$: finite set of timesteps
- $T$: transition function - transitions between states
- $R$: reward function - rewards obtained from transitions

# Markov Decision Process

Tuple $\langle S, A, D, T, R \rangle$:

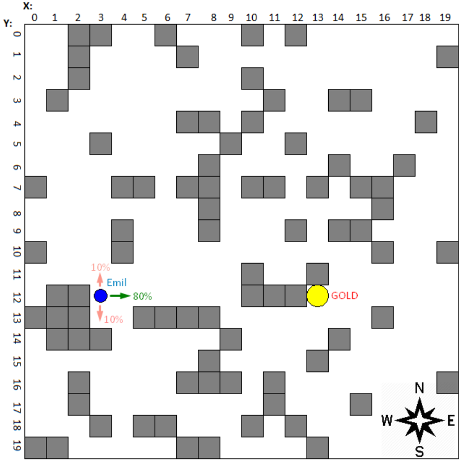
- $S$: finite set of states agent can find itself in
- $A$: finite set of action agent can perform
- $D$: finite set of timesteps
- $T$: transition function - transitions between states
- $R$: reward function - rewards obtained from transitions

---

⚠Only one of many possible definitions!

# Example: Emil in the gridworld

- $S$: Possible Emils positions
- $A$: Move directions
- $D$: Emil has e.g. 200 steps to find gold
- $T$: stochastic movement, e.g. 10% to move to the side of selected action
- $R$: e.g. +100 for finding gold, -1 for each move



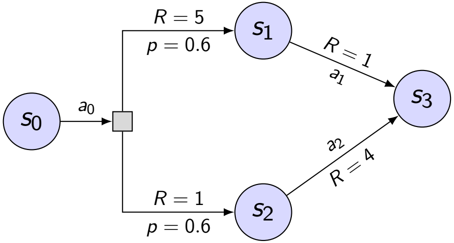
# MDP example - blackjack

Blackjack

- $S$: Possible player hands and played cards
- $A$: Hit, Stand, ...
- $T$: Possible drawn cards,
- $R$: Win/loose at the end

# Example: Abstract example

- $S$: $S_0, S_1, S_2, S_3$
- $A$: $a_0, a_1, a_2$
- $T$ : $$\begin{aligned} T(S_0, a_0, S_1) &= 0.6 \\ T(S_0, a_0, S_2) &= 0.4 \\ T(S_1, a_1, S_3) &= 1 \\ T(S_2, a_2, S_3) &= 1 \end{aligned}$$
- $R$ : $$\begin{aligned} R(S_0, a_0, S_1) &= 5 \\ R(S_0, a_0, S_2) &= 2 \\ R(S_1, a_1, S_3) &= 1 \\ R(S_2, a_2, S_3) &= 4 \end{aligned}$$




[1] Example: [Mausam, Kobolov: Planning With Markov Decision Processes]

# When MDP might be a good model?

- *Domain with uncertainty* - uncertain outcomes of actions
- *Sequential decision making* - for *sequences* of decisions
- *Fair Nature* - no one is actively playing against us
- *Full observability, perfect sensors* - we know where agent is
- *Cyclic domain structures* - when states can be revisited

# Policy

**Def: Policy**

Assignment of action to state, $\pi : S \rightarrow A$

- *Partial policy* - e.g. output of robust replanning
- *Complete policy* - domain of $\pi$ is whole state space $S$.
- *Stationary policy* - independent of timestep (e.g. emil)
- *Markovian policy* - dependent only on last state

---

⚠In general, policy can be history dependent and stochastic!

# Value function (of a policy)

**Def: Value function**

Assignment of value to state, $V : S \rightarrow < -\infty, \infty >$

# Value function (of a policy)

**Def: Value function**

Assignment of value to state, $V : S \rightarrow < -\infty, \infty >$

**Def: Value function of a policy**

Assignment of value to state based on utility of rewards obtained by following policy $\pi$ from a state, $V^\pi : S \rightarrow < -\infty, \infty >$, $V^\pi(s) = u(R_1^{\pi_s}, R_2^{\pi_s}, \dots)$

# Value function (of a policy)

**Def: Value function**

Assignment of value to state, $V : S \rightarrow < -\infty, \infty >$

**Def: Value function of a policy**

Assignment of value to state based on utility of rewards obtained by following policy $\pi$ from a state, $V^{\pi} : S \rightarrow < -\infty, \infty >$, $V^{\pi}(s) = u(R_1^{\pi_s}, R_2^{\pi_s}, \ldots)$

**Def: Optimal MDP solution**

Optimal MDP solution is a policy $\pi^*$ such that value function $V^{\pi^*}$ called optimal value function dominates all other value functions in all states, $\forall s V^{\pi^*}(s) \geq V^{\pi}(s)$.

# Value function (of a policy)

**Def: Value function of a policy**

Assignment of value to state based on utility of rewards obtained by following policy $\pi$ from a state, $V^\pi : S \to < -\infty, \infty >$, $V^\pi(s) = u(R_1^{\pi_s}, R_2^{\pi_s}, \dots)$

**Def: Optimal MDP solution**

Optimal MDP solution is a policy $\pi^*$ such that value function $V^{\pi^*}$ called optimal value function dominates all other value functions in all states, $\forall s V^{\pi^*}(s) \geq V^\pi(s)$.

---

Questions:

- How can we pick $u$? Can we choose $u(R_1, R_2, \dots) = \sum_i R_i$ ?

# Expected linear aditive utility

**Def: Expected linear aditive utility**

Function $u(R_t, R_{t+1}, \ldots) = \mathbb{E}\left[\sum_{t'=t}^{|D|} \gamma^{t'} R_{t'}\right]$ is expected linear additive utility

Sounds convoluted, but it gives

**Bellman equation**

$V^\pi(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^\pi(s')]]$

# Expected linear aditive utility

## Def: Expected linear aditive utility

Function $u(R_t, R_{t+1}, \ldots) = \mathbb{E}\left[\sum_{t'=t}^{|D|} \gamma^{t'} R_{t'}\right]$ is expected linear additive utility

Sounds convoluted, but it gives

## Bellman equation

$V^\pi(s) = [\sum_{s' \in S} T(s, a, s')[R(s, a, s') + \gamma V^\pi(s')]]$

- $\gamma \in (0, 1]$ is a discount factor, makes agent prefer earlier rewards.

# Expected linear aditive utility

**Def: Expected linear aditive utility**

Function $u(R_t, R_{t+1}, \ldots) = \mathbb{E}\left[\sum_{t'=t}^{|D|} \gamma^{t'} R_{t'}\right]$ is expected linear additive utility

Sounds convoluted, but it gives

**Bellman equation**

$V^\pi(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^\pi(s')]]$

- $\gamma \in (0,1]$ is a discount factor, makes agent prefer earlier rewards.
- Risk-neutral

# Expected linear aditive utility

## Def: Expected linear aditive utility

Function $u(R_t, R_{t+1}, \ldots) = \mathbb{E}\left[\sum_{t'=t}^{|D|} \gamma^{t'} R_{t'}\right]$ is expected linear additive utility

Sounds convoluted, but it gives

## Bellman equation

$V^\pi(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^\pi(s')]]$

- $\gamma \in (0,1]$ is a discount factor, makes agent prefer earlier rewards.
- Risk-neutral
- For infinite $D$ and bounded rewards, $\gamma < 1$ gives convergence (why?)

# Expected linear aditive utility

**Def: Expected linear aditive utility**

Function $u(R_t, R_{t+1}, \ldots) = \mathbb{E}\left[\sum_{t'=t}^{|D|} \gamma^{t'} R_{t'}\right]$ is expected linear additive utility

Sounds convoluted, but it gives

**Bellman equation**

$V^\pi(s) = [\sum_{s' \in S} T(s, a, s')[R(s, a, s') + \gamma V^\pi(s')]]$

- $\gamma \in (0, 1]$ is a discount factor, makes agent prefer earlier rewards.
- Risk-neutral
- For infinite $D$ and bounded rewards, $\gamma < 1$ gives convergence (why?)
- Under certain conditions, implies existence of optimal solution(s)

# Example

## Bellman equation

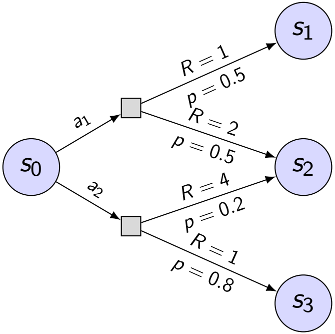
$V^\pi(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^\pi(s')]]$

# Example

## Bellman equation

$V^\pi(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^\pi(s')]]$

Look at the following small MDP. Which action would you take?

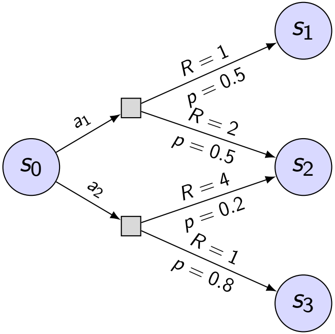


# Example

## Bellman equation

$V^{\pi}(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^{\pi}(s')]]$

Calculate value of a policy $\pi(S_1) = a_1$

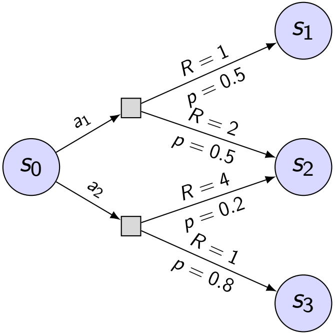


# Example

**Bellman equation**

$V^{\pi}(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^{\pi}(s')]]$

Calculate value of a policy $\pi(S_1) = a_2$

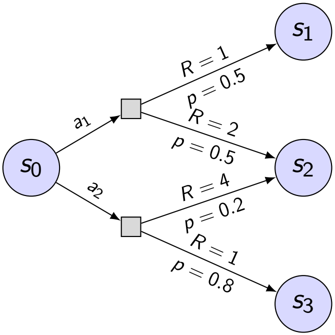


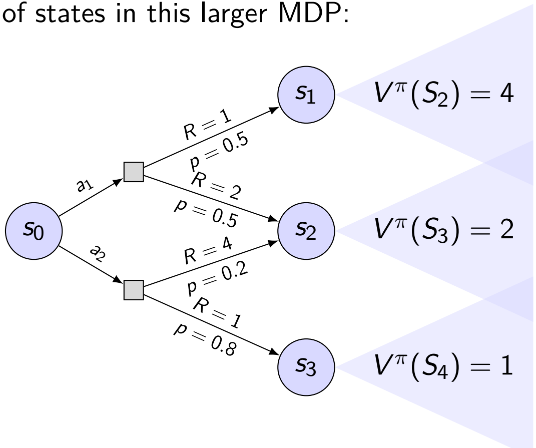
# Example

## Bellman equation

$V^{\pi}(s) = [\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^{\pi}(s')]]$

Calculate value of both policies given the value of states in this larger MDP:



# Optimality principle

When using expected linear additive utility, "MDP" has an optimal deterministic Markovian policy $\pi^*$.

## Thm: The optimality principle for infinite-horizon MDPs

Infinite horizon MDP with $V^\pi(s_t) = \mathbb{E}\left[\sum_{t'=0}^{\infty} \gamma^{t'} R^\pi_{t+t'}\right]$ and $\gamma \in [0,1)$. Then there exists optimal value function $V^*$, is stationary, Markovian, and satisfies for all $s$:

$$V^*(s) = \max_\pi V^\pi(s)$$

$$V^*(s) = \max_{a \in A}\left[\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]\right]$$

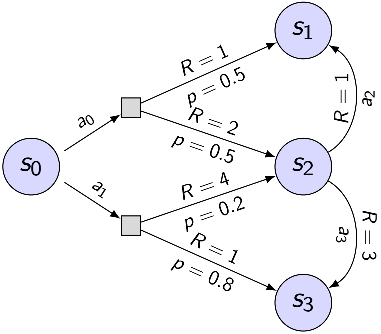
$$\pi^*(s) = \arg\max_{a \in A}\left[\sum_{s' \in S} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]\right]$$

# Examples

In the examples, we will use $\gamma = 1$ since we are in domains with finite horizon (and have guaranteed convergence).

# Calculate the *optimal* value function in acyclic MDP

- $S$: $\{S_0, S_1, S_2, S_3\}$
- $A$: $\{a_0, a_1, a_2, a_3\}$
- $T$ :
$$T(S_0, a_0, S_1) = 0.5$$
$$T(S_0, a_0, S_2) = 0.5$$
$$T(S_1, a_1, S_2) = 0.2$$
$$T(S_2, a_1, S_3) = 0.8$$
$$T(S_2, a_2, S_1) = 1$$
$$T(S_2, a_3, S_3) = 1$$
- $R$ :
$$R(S_0, a_1, S_1) = 1$$
$$R(S_0, a_1, S_2) = 2$$
$$R(S_0, a_2, S_2) = 4$$
$$R(S_0, a_2, S_3) = 1$$
$$R(S_2, a_2, S_1) = 1$$



# Calculate the value of a given policy $\pi$ in *cyclic* MDP

- $S$: $\{S_0, S_1, S_2, S_3\}$
- $A$: $\{a_0, a_1, a_2\} = \pi$ - only the policy actions are shown

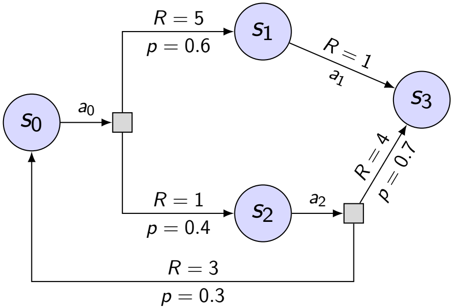
$$T(S_0, a_0, S_1) = 0.6$$
$$T(S_0, a_0, S_2) = 0.4$$

- $T$ : $T(S_1, a_1, S_3) = 1$

$$T(S_2, a_2, S_3) = 0.7$$
$$T(S_2, a_2, S_0) = 0.3$$
$$R(S_0, a_0, S_1) = 5$$
$$R(S_0, a_0, S_2) = 2$$

- $R$ : $R(S_1, a_1, S_3) = 1$

$$R(S_2, a_2, S_3) = 4$$
$$R(S_2, a_2, S_0) = 3$$



# Calculation notes

Looking at the calculations, what can you say about the calculations of value of *optimal* function?

## Calculation notes

Looking at the calculations, what can you say about the calculations of value of *optimal* function?

- In acyclic MDP, it can be straightforward to calculate the optimal value of states by taking the states in an appropriate order (which is?).

# Calculation notes

Looking at the calculations, what can you say about the calculations of value of *optimal* function?

- In acyclic MDP, it can be straightforward to calculate the optimal value of states by taking the states in an appropriate order (which is?).
- In a cyclic MDP, *for a given policy*, writing the Bellman equations for all states gives a set of linear equations. These can be solved using standard techniques from linear algebra (e.g. substitution :-), do you know other methods or solvers?).

# Calculation notes

Looking at the calculations, what can you say about the calculations of value of *optimal* function?

- In acyclic MDP, it can be straightforward to calculate the optimal value of states by taking the states in an appropriate order (which is?).
- In a cyclic MDP, *for a given policy*, writing the Bellman equations for all states gives a set of linear equations. These can be solved using standard techniques from linear algebra (e.g. substitution :-), do you know other methods or solvers?).
- In a cyclic MDP, calculating is complicated by the *max* term - non-linear set of equations.

**Thank you for participating in the tutorials :-)**

Please fill the feedback form →

`https://forms.gle/BimaGk1wUzb1rXba7`