

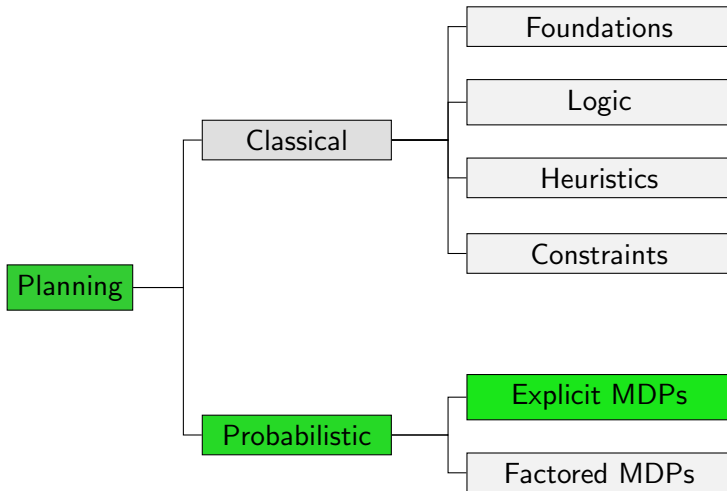
# Planning and Optimization

## F2. Bellman Equation & Linear Programming

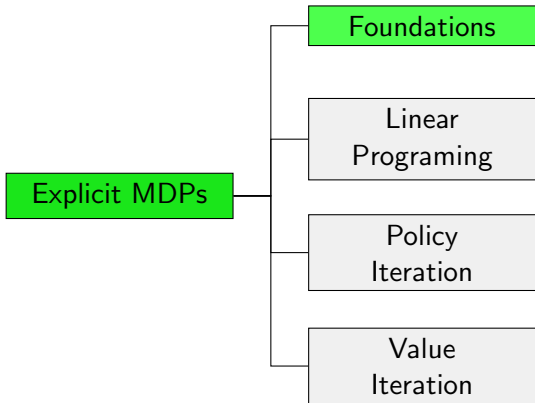
Malte Helmert and Gabriele Röger

Universität Basel

# Content of this Course



# Content of this Course: Explicit MDPs



# Introduction

# Quality of Solutions

- Solution in classical planning: plan
- Optimality criterion of a solution in classical planning:  
minimize plan cost

# Quality of Solutions

- Solution in classical planning: plan
- Optimality criterion of a solution in classical planning:  
minimize plan cost
- Solution in probabilistic planning: policy
- What is the optimality criterion of a solution  
in probabilistic planning?

## Example: Swiss Lotto

### Example (Swiss Lotto)

What is the **expected payoff** of placing one bet in Swiss Lotto for a cost of CHF 2.50 with (simplified) payouts and probabilities:

CHF 30.000.000 with prob.	$1/31474716$	(6 + 1)
CHF 1.000.000 with prob.	$1/5245786$	(6)
CHF 5.000 with prob.	$1/850668$	(5)
CHF 50 with prob.	$1/111930$	(4)
CHF 10 with prob.	$1/11480$	(3)

## Example: Swiss Lotto

### Example (Swiss Lotto)

What is the **expected payoff** of placing one bet in Swiss Lotto for a cost of CHF 2.50 with (simplified) payouts and probabilities:

CHF 30.000.000 with prob.  $1/31474716$  (6 + 1)

CHF 1.000.000 with prob.  $1/5245786$  (6)

CHF 5.000 with prob.  $1/850668$  (5)

CHF 50 with prob.  $1/111930$  (4)

CHF 10 with prob.  $1/11480$  (3)

**Solution:**

$$\frac{30000000}{31474716} + \frac{1000000}{5245786} + \frac{5000}{850668} + \frac{50}{111930} + \frac{10}{11480} - 2.5 \approx -1.35.$$



## Expected Values under Uncertainty

### Definition (Expected Value of a Random Variable)

Let  $X$  be a random variable with a finite number of **outcomes**  $d_1, \dots, d_n \in \mathbb{R}$ , and let  $d_i$  happen with probability  $p_i \in [0, 1]$  (for  $i = 1, \dots, n$ ) s.t.  $\sum_{i=1}^n p_i = 1$ .

The **expected value** of  $X$  is  $\mathbb{E}[X] = \sum_{i=1}^n (p_i \cdot d_i)$ .

# Bellman Equation

# Value Functions for MDPs

## Definition (Value Functions for MDPs)

Let  $\pi$  be a policy for MDP  $\mathcal{T} = \langle S, A, R, T, s_0, \gamma \rangle$ .

The **state-value**  $V_\pi(s)$  of  $s \in S_\pi(s_0)$  under  $\pi$  is defined as

$$V_\pi(s) := Q_\pi(s, \pi(s))$$

where the **action-value**  $Q_\pi(s, a)$  of  $s$  and  $a$  under  $\pi$  is defined as

$$Q_\pi(s, a) := R(s, a) + \gamma \cdot \sum_{s' \in \text{succ}(s, a)} T(s, a, s') \cdot V_\pi(s').$$

The state-value  $V_\pi(s)$  describes the **expected reward** of applying  $\pi$  in MDP  $\mathcal{T}$ , starting from  $s$ .

# Bellman Equation in MDPs

## Definition (Bellman Equation in MDPs)

Let  $\mathcal{T} = \langle S, A, R, T, s_0, \gamma \rangle$  be an MDP.

The **Bellman equation** for a state  $s$  of  $\mathcal{T}$  is the set of equations that describes  $V_*(s)$ , where

$$V_*(s) := \max_{a \in A(s)} Q_*(s, a)$$

$$Q_*(s, a) := R(s, a) + \gamma \cdot \sum_{s' \in \text{succ}(s, a)} T(s, a, s') \cdot V_*(s').$$

The solution  $V_*(s)$  of the Bellman equation describes the **maximal expected reward** that can be achieved from state  $s$  in MDP  $\mathcal{T}$ .

# Optimal Policy in MDPs

What is the policy that achieves the maximal expected reward?

## Definition (Optimal Policy in MDPs)

Let  $\mathcal{T} = \langle S, A, R, T, s_0, \gamma \rangle$  be an MDP.

A policy  $\pi$  is an **optimal policy** if  $\pi(s) \in \arg \max_{a \in A(s)} Q_*(s, a)$  for all  $s \in S_\pi(s_0)$  and the **expected reward** of  $\pi$  in  $\mathcal{T}$  is  $V_*(s_0)$ .

# Value Functions for SSPs

## Definition (Value Functions for SSPs)

Let  $\mathcal{T} = \langle S, A, c, T, s_0, S_\star \rangle$  be an SSP and  $\pi$  be a policy for  $\mathcal{T}$ . The **state-value**  $V_\pi(s)$  of  $s$  under  $\pi$  is defined as

$$V_\pi(s) := \begin{cases} 0 & \text{if } s \in S_\star \\ Q_\pi(s, \pi(s)) & \text{otherwise,} \end{cases}$$

where the **action-value**  $Q_\pi(s, a)$  of  $s$  and  $a$  under  $\pi$  is defined as

$$Q_\pi(s, a) := c(a) + \sum_{s' \in \text{succ}(s, a)} T(s, a, s') \cdot V_\pi(s').$$

The state-value  $V_\pi(s)$  describes the **expected cost** of applying  $\pi$  in SSP  $\mathcal{T}$ , starting from  $s$ .

# Bellman Equation in SSPs

## Definition (Bellman Equation in SSPs)

Let  $\mathcal{T} = \langle S, A, c, T, s_0, S_\star \rangle$  be an SSP.

The **Bellman equation** for a state  $s$  of  $\mathcal{T}$  is the set of equations that describes  $V_\star(s)$ , where

$$V_\star(s) := \begin{cases} 0 & \text{if } s \in S_\star \\ \min_{a \in A(s)} Q_\star(s, a) & \text{otherwise,} \end{cases}$$

$$Q_\star(s, a) := c(a) + \sum_{s' \in \text{succ}(s, a)} T(s, a, s') \cdot V_\star(s').$$

The solution  $V_\star(s)$  of the Bellman equation describes the **minimal expected cost** that can be achieved from state  $s$  in SSP  $\mathcal{T}$ .

# Optimal Policy in SSPs

What is the policy that achieves the minimal expected cost?

## Definition (Optimal Policy in SSPs)

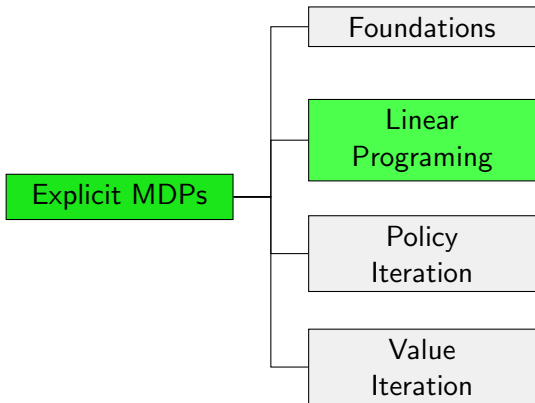
Let  $\mathcal{T} = \langle S, A, c, T, s_0, S_\star \rangle$  be an SSP.

A policy  $\pi$  is an **optimal policy** if  $\pi(s) \in \arg \min_{a \in A(s)} Q_\star(s, a)$  for all  $s \in S_\pi(s_0) \setminus S_\star$  and the **expected cost** of  $\pi$  in  $\mathcal{T}$  is  $V_\star(s_0)$ .



# Linear Programming

# Content of this Course: Explicit MDPs



# Linear Programming for SSPs

- Bellman equation: set of equations that describes the **expected cost** for each state.
  - there are  $|S|$  variables and  $|S|$  equations (replacing  $Q_*$  in  $V_*$  with the corresponding equation)
  - If we solve these equations, we can determine an optimal policy for the SSP from the state-values.
  - **Problem**: how can we deal with the **minimization**?
- ⇒ We have solved the “same” problem before with the help of an LP solver

# Reminder: LP for Shortest Path in State Space

## Variables

Non-negative variable  $\text{Distance}_s$  for each state  $s$

## Objective

Maximize  $\text{Distance}_{s_0}$

## Subject to

$\text{Distance}_{s_*} = 0$  for all goal states  $s_*$

$\text{Distance}_s \leq \text{Distance}_{s'} + c(\ell)$  for all transitions  $s \xrightarrow{\ell} s'$

# LP for Expected Cost in SSP

## Variables

Non-negative variable  $\text{ExpCost}_s$  for each state  $s$

## Objective

Maximize  $\text{ExpCost}_{s_0}$

## Subject to

$\text{ExpCost}_{s_*} = 0$  for all goal states  $s_*$

$$\text{ExpCost}_s \leq \left( \sum_{s' \in S} T(s, a, s') \cdot \text{ExpCost}_{s'} \right) + c(a)$$

for all  $s \in S$  and  $a \in A(s)$

# LP for Expected Reward in MDP

## Variables

Non-negative variable  $\text{ExpReward}_s$  for each state  $s$

## Objective

Minimize  $\text{ExpReward}_{s_0}$

## Subject to

$$\text{ExpReward}_s \geq \left( \gamma \cdot \sum_{s' \in S} T(s, a, s') \text{ExpReward}_{s'} \right) + R(s, a)$$

for all  $s \in S$  and  $A \in A(s)$

# Complexity of Probabilistic Planning

- an **optimal solution** for MDPs or SSPs can be computed with an **LP solver**
- requires  $|S|$  variables and  $|S| \cdot |A|$  constraints
- we know that LPs can be solved in **polynomial time**
- $\Rightarrow$  solving MDPs or SSPs is a **polynomial time** problem

How does this relate to the complexity result for classical planning?

# Complexity of Probabilistic Planning

- an **optimal solution** for MDPs or SSPs can be computed with an **LP solver**
- requires  $|S|$  variables and  $|S| \cdot |A|$  constraints
- we know that LPs can be solved in **polynomial time**
- $\Rightarrow$  solving MDPs or SSPs is a **polynomial time** problem

How does this relate to the complexity result for classical planning?

Solving MDPs or SSPs is polynomial in  $|S| \cdot |A|$ .



# Summary

# Summary

- The state-values of a policy specify the **expected reward (cost)** of following that policy.
- The **Bellman equation** describes the state-values of an optimal policy.
- **Linear Programming** can be used to solve MDPs and SSPs in time **polynomial** in the size of the MDP/SSP.