Planning and Optimization

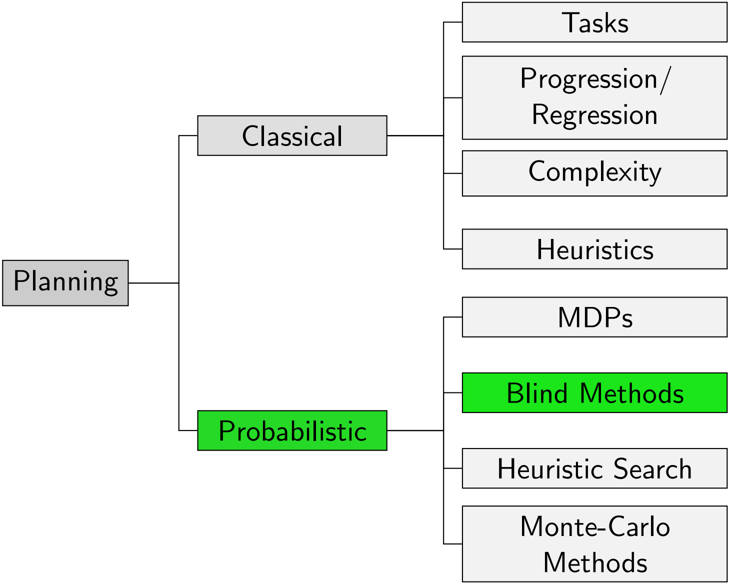
F5. Determinization

Gabriele Röger and Thomas Keller

Universität Basel

November 28, 2018

Content of this Course



Suboptimal Probabilistic Planning

Large SSPs and FH-MDPs

- Before: optimal policies and exact state-values for small SSPs, FH-MDPs and DR-MDPs.
- Now: focus on large SSPs and FH-MDPs (no more DR-MDPs)
- Further algorithms not necessarily optimal (may generate suboptimal policies)

Interleaved Planning & Execution

- Number of states of executable policy usually exponential in number of state variables. (Why?)
- For large SSPs and FH-MDPs, executable policy cannot be provided explicitly by enumeration.
- Solution: (possibly approximate) compact representation of executable policy required to describe solution
  $\Rightarrow$ not part of this lecture.
- Alternative solution: policies defined only on minimal set of states

Interleaved Planning & Execution for SSPs

Plan-execute-monitor cycle for SSP $\mathcal{T}$:

- plan action a for the current state s_0
- execute a
- observe new current state s'
- update $\mathcal{T}$ by setting $s_0 := s'$
- repeat until $s_0 \in S_\star$

Interleaved Planning & Execution for FH-MDPs

Plan-execute-monitor cycle for FH-MDP $\mathcal{T}$:

- plan action a for the current state s_0
- execute a
- observe new current state s'
- update $\mathcal{T}$ by setting $s_0 := s'$ and $H := H - 1$
- repeat until $H = 0$

Interleaved Planning & Execution in Practice

+ avoid loss of precision that often comes with compact description of executable policy
+ do not waste time with planning for states that are never reached during execution
- poor decisions can be avoided by spending more time with deliberation before execution
- in SSPs, this can even mean that computed policy is not proper and execution never reaches the goal

Estimated Policy Evaluation

Estimated Policy Evaluation

- The quality of a policy is described by its expected cost (SSP) or reward (FH-MDP) $V_\pi(s_0)$ in the initial state s_0
- VI, PI and Linear Programming compute optimal policy π and quality $V_\pi(s_0) = V_\star(s_0)$
- Quality of explicitly given policy π can be computed by acyclic or iterative policy evaluation (in small SSPs / FH-MDPs)
- Impossible with interleaved planning & execution

Estimated Policy Evaluation

- Estimate quality of policy π
- Execute π for $n \in \mathbb{N}$ times
- Let ρ_π^i denote the accumulated cost (SSP) or reward (FH-MDP) of the i-th run (execution) of π. Then use

$$\tilde{V}_\pi := \frac{1}{n} \cdot \sum_{i=1}^{n} \rho_\pi^i$$

  as quality estimate.
- With strong law of large numbers we have

$$\tilde{V}_\pi \to V_\pi(s_0) \text{ for } n \to \infty$$

- Good approximation if n sufficiently large

Determinization

What is a Determinization?

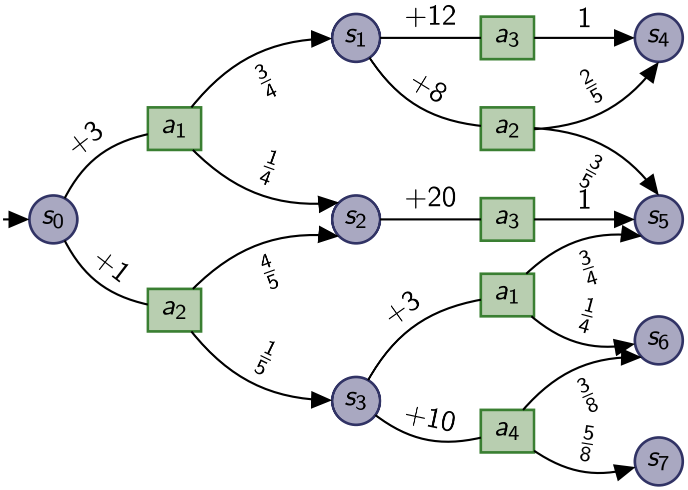
- Replace probabilistic actions with deterministic ones
- Results in deterministic planning problem
- For SSPs, this is a classical planning task
- SSP/FH-MDP and its determinization are related but not equivalent
- Determinization often solvable in practice even if MDP is not

How do we Come up with a Determinization?

Two main types of determinization

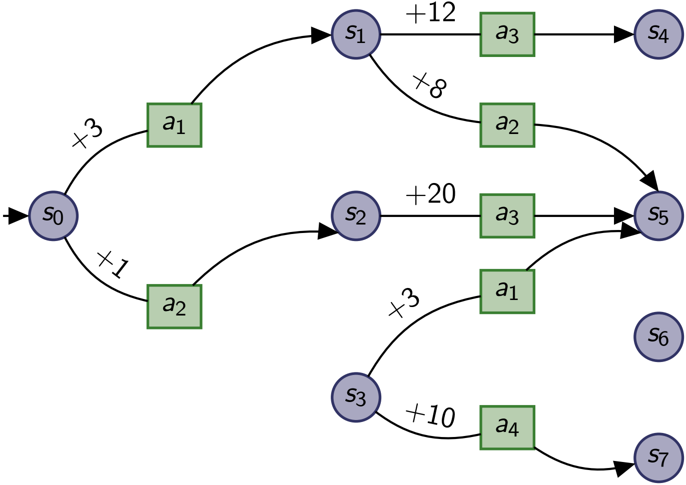
- Single-outcome determinization
  - Remove all outcomes from actions except for one
  - In most practical applications, the most likely outcome is preserved
- All-outcomes determinization
  - Create one action for each outcome
  - Corresponds to most desired single-outcome determinization (the planner is allowed to choose the outcome)

Example: Finite Horizon MDP

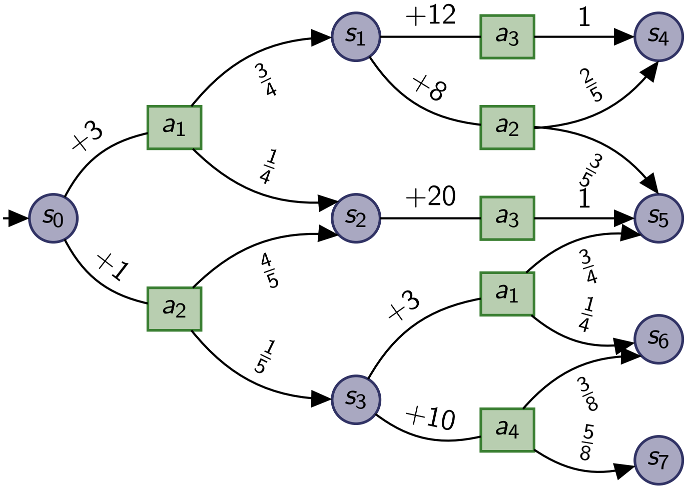


Example: Single-outcome Determinization

Remove all outcomes but one

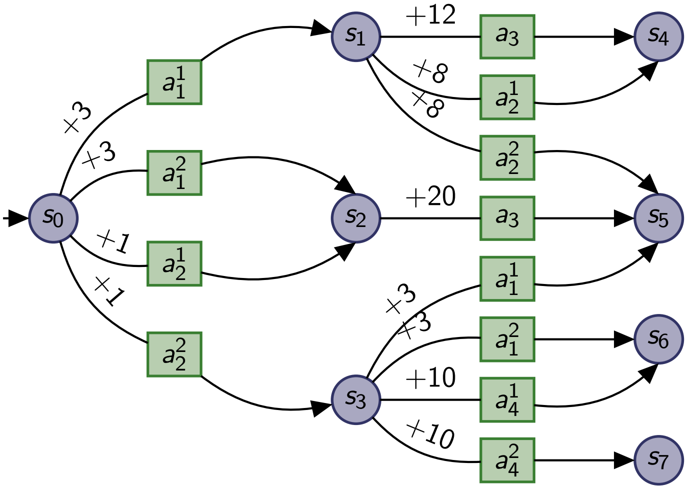


Example: All-outcomes Determinization



Example: All-outcomes Determinization

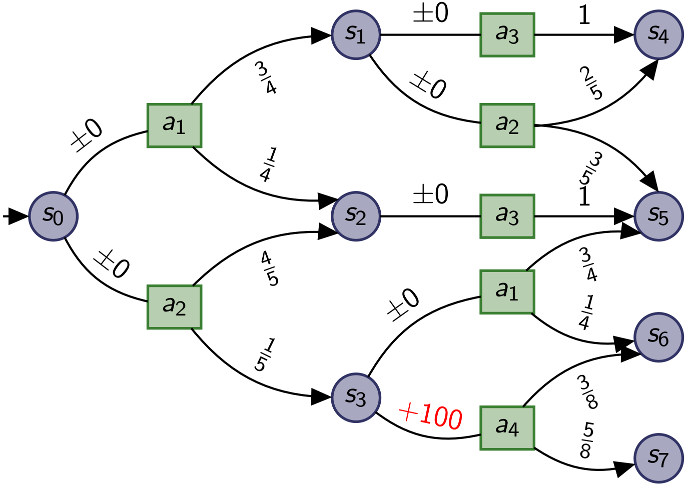
Generate one action per outcome



Determinizations: Weaknesses

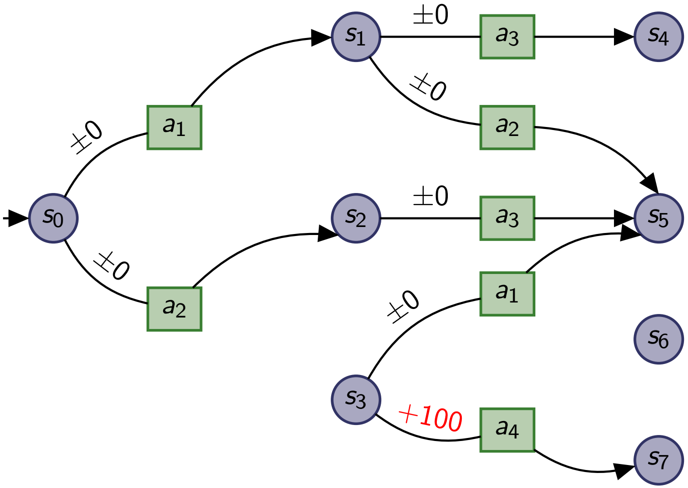
- Single-outcome determinizations: important parts of state space can become unreachable $\Rightarrow$ poor policy or unsolvable

Single-outcome Determinization: Weaknesses



Single-outcome Determinization: Weaknesses

Important parts of the MDP can become unreachable

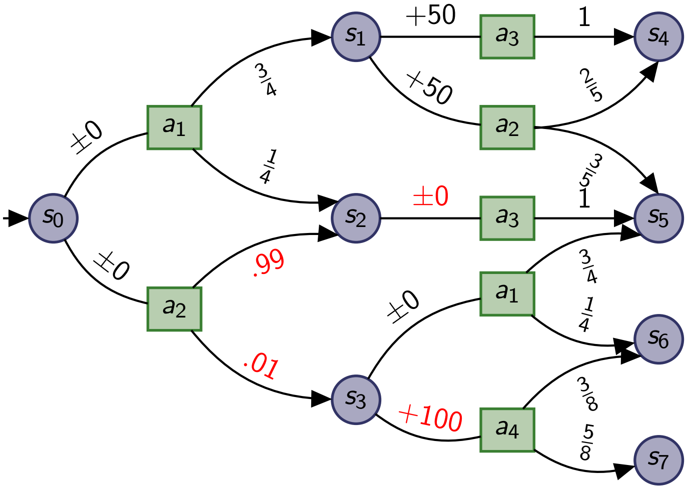


Determinizations: Weaknesses

- Single-outcome determinizations: important parts of state space can become unreachable $\Rightarrow$ poor policy or unsolvable
- All-outcomes determinization: utterly optimistic

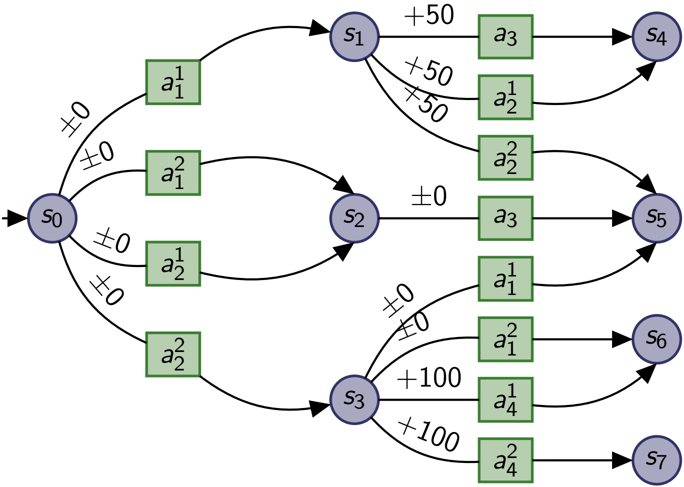
All-outcomes Determinization: Weaknesses

All-outcomes determinization is too optimistic



All-outcomes Determinization: Weaknesses

All-outcomes determinization is too optimistic



Determinizations: Weaknesses

- Single-outcome determinizations: important parts of state space can become unreachable ⇒ poor policy or unsolvable
- All-outcomes determinization: utterly optimistic
- All-outcomes determinization: number of outcomes can be exponential in the number of parallel probabilistic effects
  Note: Unlike the previous, this is a problem on the syntactic level

All-outcomes Determinizations: Weaknesses

Example

Consider the operator

$$o = \langle \top, (p_{11} : v_1 \mid p_{12} : \neg v_1) \wedge \cdots \wedge (p_{n1} : v_n \mid p_{n2} : \neg v_n), 1 \rangle$$

of a planning task with set of variables $V = \{v_1, \ldots, v_n\}$.
All states in the set of states S over V are possible outcomes of o, and the number of deterministic operators in the all-outcomes determinization is hence 2^n.

Determinizations in Practice

Despite the inherent weaknesses, determinizations have been used successfully in practice. Consider the winners of all probabilistic tracks of the International Planning Competition:

- 2004: FF-Replan (Yoon, Fern & Givan) interleaves planning & execution of plan in determinization
- 2006: FPG (Buffet & Aberdeen) learns a policy utilizing FF-Replan
- 2008: RFF (Teichteil-Königsbuch, Infantes & Kuter) extends determinization-based plan to policy
- 2011 and 2014: PROST-2011 (Keller & Eyerich) and PROST-2014 (Keller & Geißer) use determinization-based lookahead heuristic
- 2018: PROST-DD (Geißer & Speck) use BDD representation of determinization as heuristic

Summary

Summary

- The plan-execute-monitor cycle allows compact policies that can be executed
- Policies can be evaluated by repeated execution
- A single-outcome determinization removes all outcomes from a transition except for one
- An all-outcomes determinization creates a deterministic transition for each outcome
- Both types of determinization have inherent weaknesses
- Determinizations have been applied successfully in practice