Multi-Objective Verification on MDPs

Tim Quatmann, Joost-Pieter Katoen

Lehrstuhl für Informatik 2

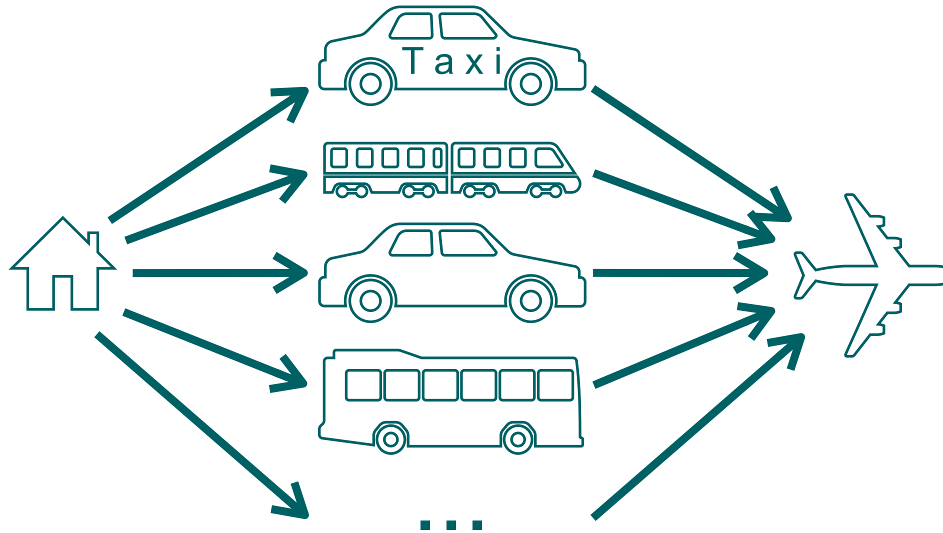
Software Modeling and Verification Chair

RWTH AACHEN UNIVERSITY

Motivation

Planning Under Uncertainty

- Scenario: Travel to the airport
- Travel time is **uncertain**

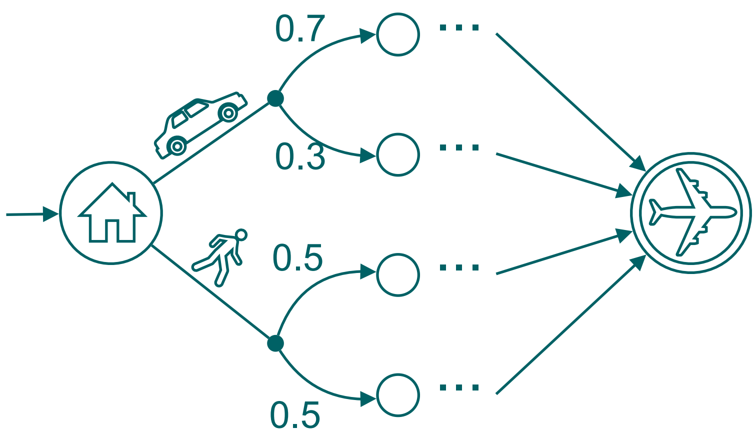


Goal: Arrive before the flight departs!

Motivation

Traveling with Computer Scientists

- Model trip to airport as a Markov decision process (MDP)
  - Controlable nondeterminism
  - Probabilistic branching

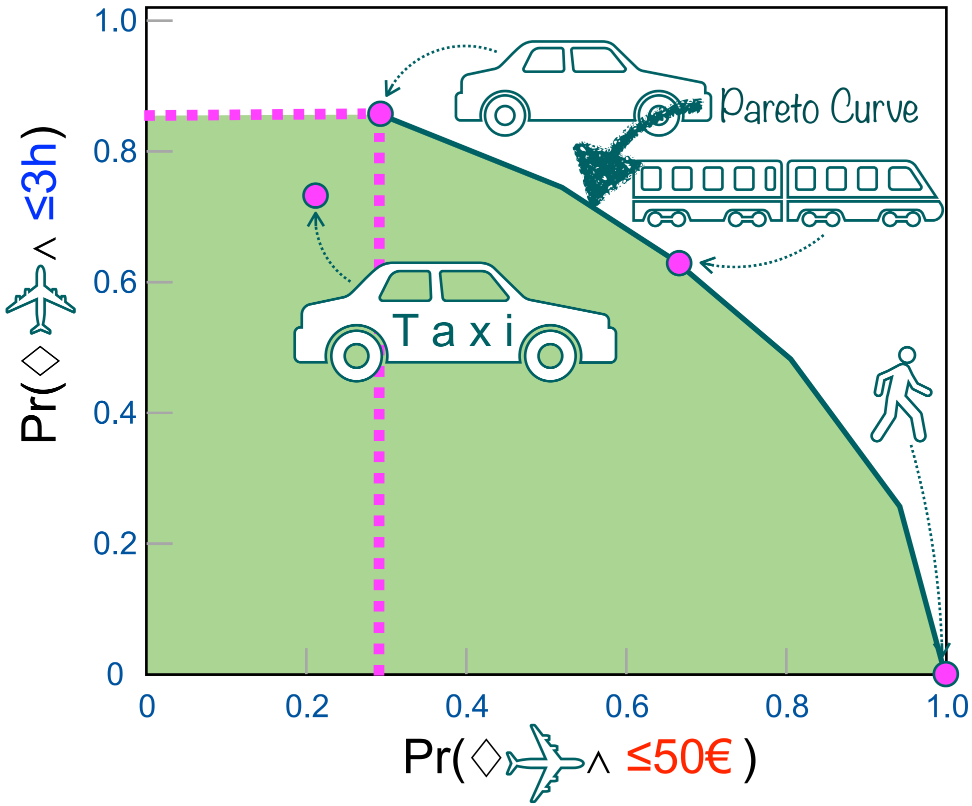


- Maximise probability to arrive before the flight departs
  - Pr(◇✈∧ ≤3h)
- Other types of cost play a role as well:
  - Maximize Pr(◇✈∧ ≤50€)
  - fuel, pollution, stress, waiting time, ...

Multi-objective Model Checking

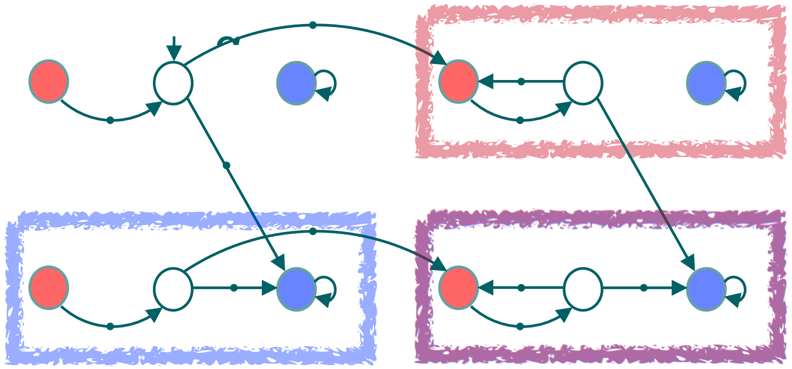
Analyse Tradeoffs Between Objectives

Arrive within 3 hours
vs.
Invest less than 50 €



Overview

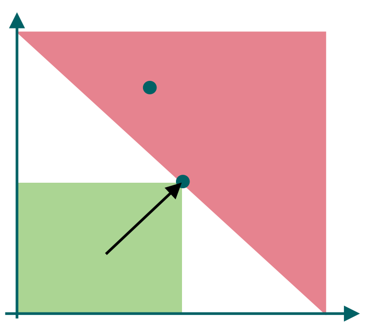
- **MDPs with Multiple Objectives**



- **Linear Programming Approach**

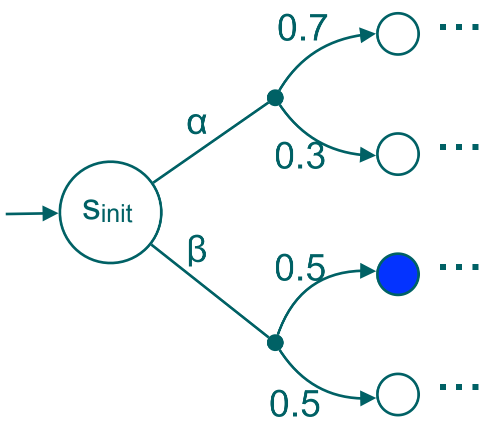
$$f_{in}(s) = f_{out}(s) \quad \text{for all } s \in S / S_{\perp}$$
$$y_{s,\alpha,t} \geq 0 \quad \text{for all } s,t \in S,\ \alpha \in Act(s)$$
$$\Sigma_{s\in S-i}\ \Sigma_{\alpha\in Act(s)}\ \Sigma_{t\in S+i}\ y_{s,\alpha,t} \geq p_i \quad \text{for all } i \in \{1, \ldots, m\}$$

- **Weighted Sum Approach**



Markov Decision Processes (MDPs)

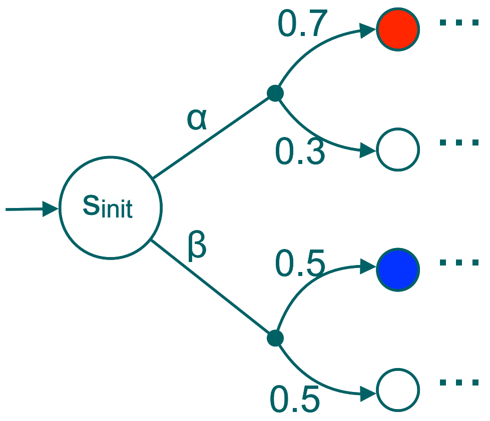
- Markov decision process $M = (S, Act, \mathbf{P}, s_{init})$
  - Nondeterminism
  - Probabilistic branching
- Randomised Policy $\mathfrak{S}$ resolves nondeterminism:
  - $\mathfrak{S}(\pi)(\alpha)$ = "Probability to choose action α when observing finite path π"
- Probability measure $Pr^{\mathfrak{S}}$:
  - $Pr^{\mathfrak{S}}(\Diamond G)$ = "Probability to reach G under $\mathfrak{S}$"



MDPs with Multiple Objectives

- Single-objective: maximal probability
  - $Pr_{max}(\lozenge G) \coloneqq \max_{\mathfrak{S}} Pr^{\mathfrak{S}}(\lozenge G)$

- Multi-objective: tradeoff
  - $Pr_{max}(\lozenge G_1)$ vs. $Pr_{max}(\lozenge G_2)$ vs. ...
  - There is not a single policy that maximises all probabilities



Geometry

- $\mathbf{p} = (p_1, \ldots, p_m) \in \mathbb{R}^m$ is a point in $m \in \mathbb{N}$ dimensional Euclidean space
- $\mathbf{p}[i]$ refers to $p_i \in \mathbb{R}$
- Let $\mathbf{p},\mathbf{q} \in \mathbb{R}^m$ and $\lambda \in \mathbb{R}$
    - $\mathbf{p} \leq \mathbf{q}$ holds iff $\mathbf{p}[i] \leq \mathbf{q}[i]$ for all $i \in \{1, \ldots, m\}$
    - $\mathbf{p} < \mathbf{q}$ holds iff $\mathbf{p} \leq \mathbf{q}$ and $\mathbf{p} \neq \mathbf{q}$
    - $\lambda \cdot \mathbf{p} \coloneqq (\lambda \cdot \mathbf{p}[1], \ldots, \lambda \cdot \mathbf{p}[m]) \in \mathbb{R}^m$ (scalar multiplication)
    - $\mathbf{p} \cdot \mathbf{q} \coloneqq \Sigma_{1 \leq i \leq m}\, \mathbf{p}[i] \cdot \mathbf{q}[i] \in \mathbb{R}$ (dot product)

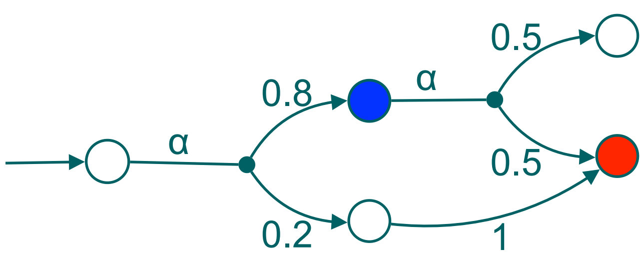
Definition:
A set $B \subseteq \mathbb{R}^m$ is convex iff $\mathbf{p},\mathbf{q} \in B$ implies $\lambda \cdot \mathbf{p} + (1-\lambda) \cdot \mathbf{q} \in B$ for all $0 \leq \lambda \leq 1$.

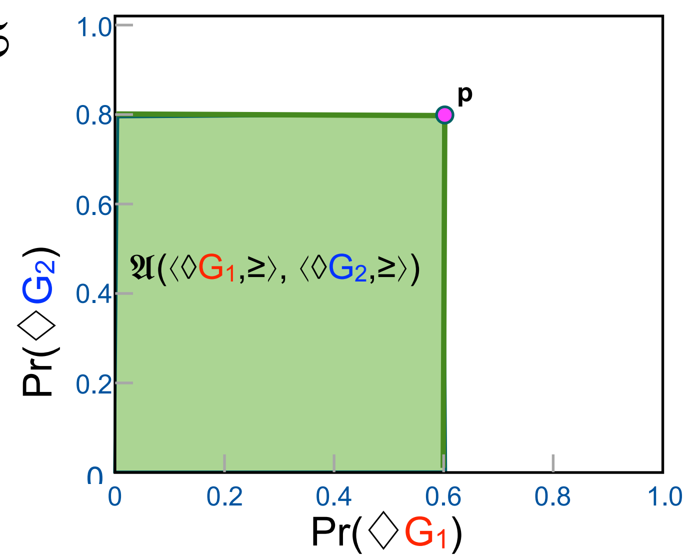
Achievable Points

Definition:
For MDP $M = (S, Act, \mathbf{P}, s_{init})$ let
- $\Pi_1, \Pi_2, \dots, \Pi_m \subseteq Paths(M)$ be m measurable sets of paths, and
- $\sim_1, \sim_2, \dots, \sim_m \subseteq \{<, \leq, \geq, >\}$.

A point $\mathbf{p} \in [0,1]^m \subseteq \mathbb{R}^m$ is called achievable iff
there is a (randomised) policy $\mathfrak{S}$ s.t. $Pr^{\mathfrak{S}}(\Pi_i) \sim_i \mathbf{p}[i]$ for all $i \in \{1, \dots, m\}$.

- We also say that point **p** is achieved by policy $\mathfrak{S}$
- $\mathfrak{A}(\langle \Pi_i, \sim_i \rangle_m)$ denotes the set of achievable points
- Example:





Achievable Points

Lemma: The set of achievable points $\mathfrak{A}(\langle \Pi_i, \sim_i \rangle_m)$ is convex.

Proof (sketch):

- Let **p**, **q** in $\mathfrak{A}(\langle \Pi_i, \sim_i \rangle_m)$, i.e., **p** and **q** are achieved by some policies $\mathfrak{S}_{\mathbf{p}}$ and $\mathfrak{S}_{\mathbf{q}}$.
- For any $\lambda \in [0,1]$, the point $\lambda \cdot \mathbf{p} + (1-\lambda) \cdot \mathbf{q}$ is achieved by the randomised policy $\mathfrak{S}$ which initially flips a coin:
  - With probability λ, it mimics policy $\mathfrak{S}_{\mathbf{p}}$
  - With probability $1-\lambda$, it mimics policy $\mathfrak{S}_{\mathbf{q}}$

Multi-Objective Reachability

- For simplicity, we only consider maximising reachability probabilities, i.e.,
  - $\Pi_i = \Diamond G_i$ for goal-states $G_i \subseteq S$
  - $\sim_i \; = \; \geq$

- We simply write $\mathfrak{A}(\langle \Diamond G_i \rangle_m)$ instead of $\mathfrak{A}(\langle \Diamond G_i, \geq \rangle_m)$

- $\mathfrak{A}(\langle \Diamond G_i \rangle_m)$ is downward closed, i.e.,

  $\mathbf{p} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ and $\mathbf{p} \geq \mathbf{q}$ implies $\mathbf{q} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$

Pareto Curve

Definition:

Let $\mathfrak{A}(\langle \Diamond G_i \rangle_m)$ be a set of achievable points.

- $\mathbf{q} \in \mathbb{R}^m$ dominates $\mathbf{p} \in \mathbb{R}^m$, iff $\mathbf{q} > \mathbf{p}$.
- $\mathbf{p} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ is Pareto optimal, iff no $\mathbf{q} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ dominates $\mathbf{p}$, i.e., $\mathbf{q} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ implies $\mathbf{q} \not> \mathbf{p}$.
- $\mathfrak{P}(\langle \Diamond G_i \rangle_m) \coloneqq \{ \mathbf{p} \mid \mathbf{p} \text{ is Pareto optimal} \}$ is the Pareto curve.

Multi-Objective Verification Queries

Achievability Query:

Given: MDP M, goal state sets $G_1, \ldots, G_m$, $\mathbf{p} \in \mathbb{R}^m$

Output: True iff $\mathbf{p} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$

Quantitative Query:

Given: MDP M, goal state sets $G_1, \ldots, G_m$, $p_2, \ldots, p_m \in \mathbb{R}$

Output: $\max \{ p_1 \in \mathbb{R} \mid (p_1, p_2, \ldots, p_m) \in \mathfrak{A}(\langle \Diamond G_i \rangle_m) \}$

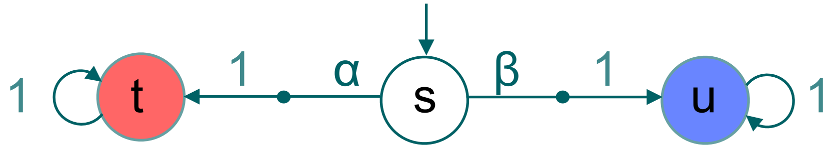
Pareto Query:

Given: MDP M, goal state sets $G_1, \ldots, G_m$

Output: Pareto Curve $\mathfrak{P}(\langle \Diamond G_i \rangle_m)$

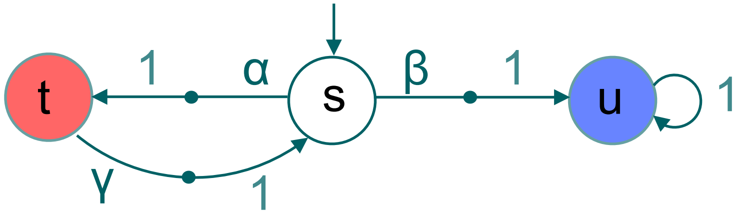
Policy Requirements

In general, we need policies with randomisation and finite memory, e.g.:



$(0.5, 0.5) \in \mathfrak{A}(\Diamond\{t\}, \Diamond\{u\})$?

Only with randomised policy $\mathfrak{S}(s)(\alpha) = \mathfrak{S}(s)(\beta) = 0.5$

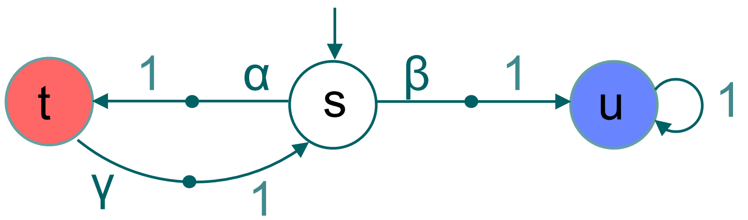


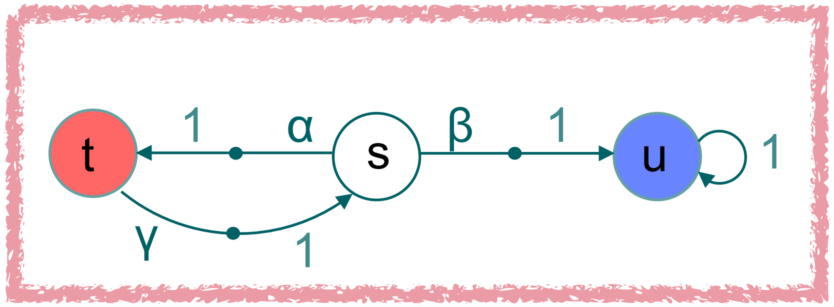
$(1, 1) \in \mathfrak{A}(\Diamond\{t\}, \Diamond\{u\})$?

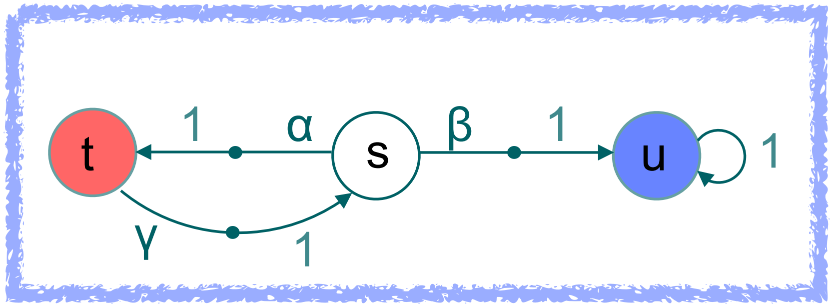
Only with finite-memory policy $\mathfrak{S}(\pi)(\alpha) = \begin{cases} 1, & \text{if } \pi \text{ has not visited } \{t\}\text{, yet} \\ 0, & \text{otherwise} \end{cases}$

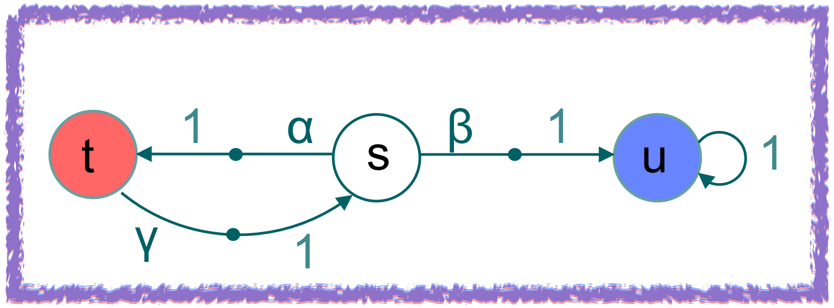
Goal-unfolding

- A policy might need to memorise which set G_i has been reached already
- Idea: Encode this information into the state-space.
- Then, positional (randomised) policies suffice



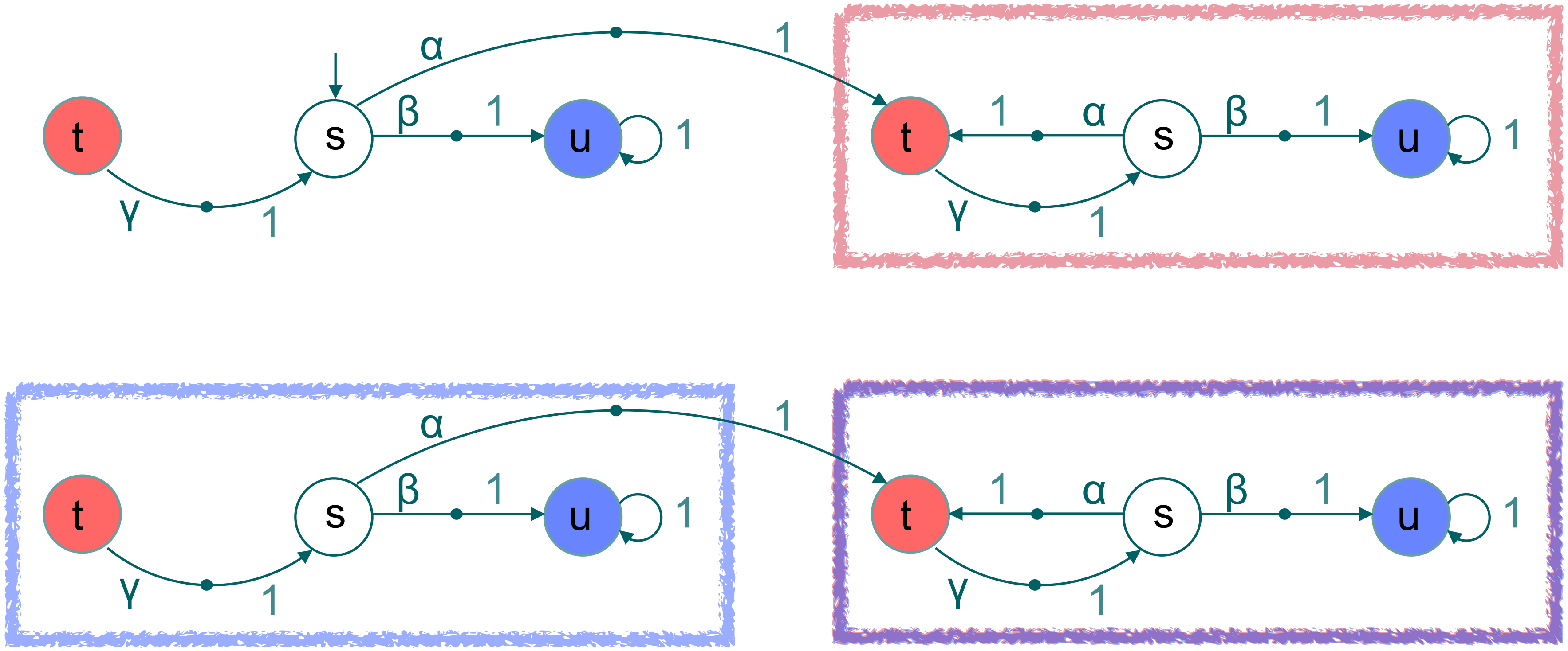






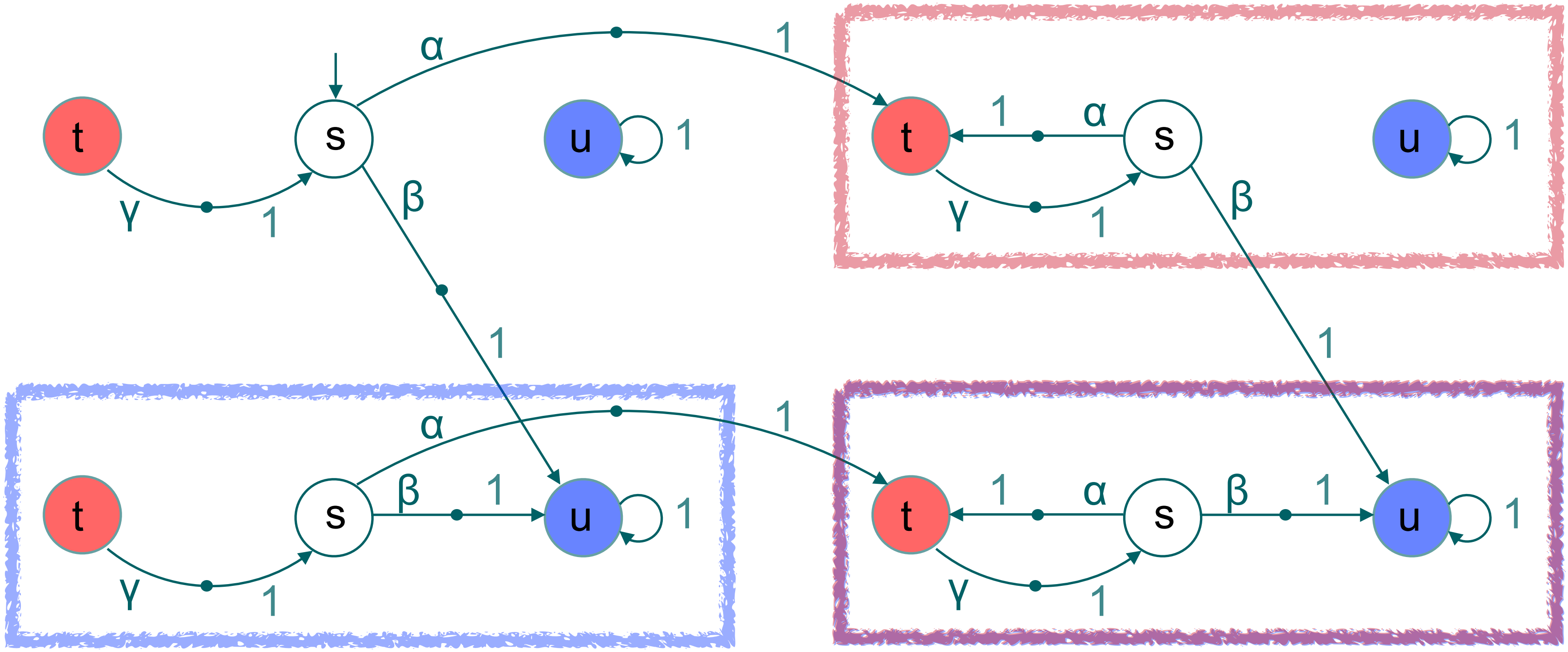
Goal-unfolding

- A policy might need to memorise which set G_i has been reached already
- Idea: Encode this information into the state-space.
- Then, positional (randomised) policies suffice



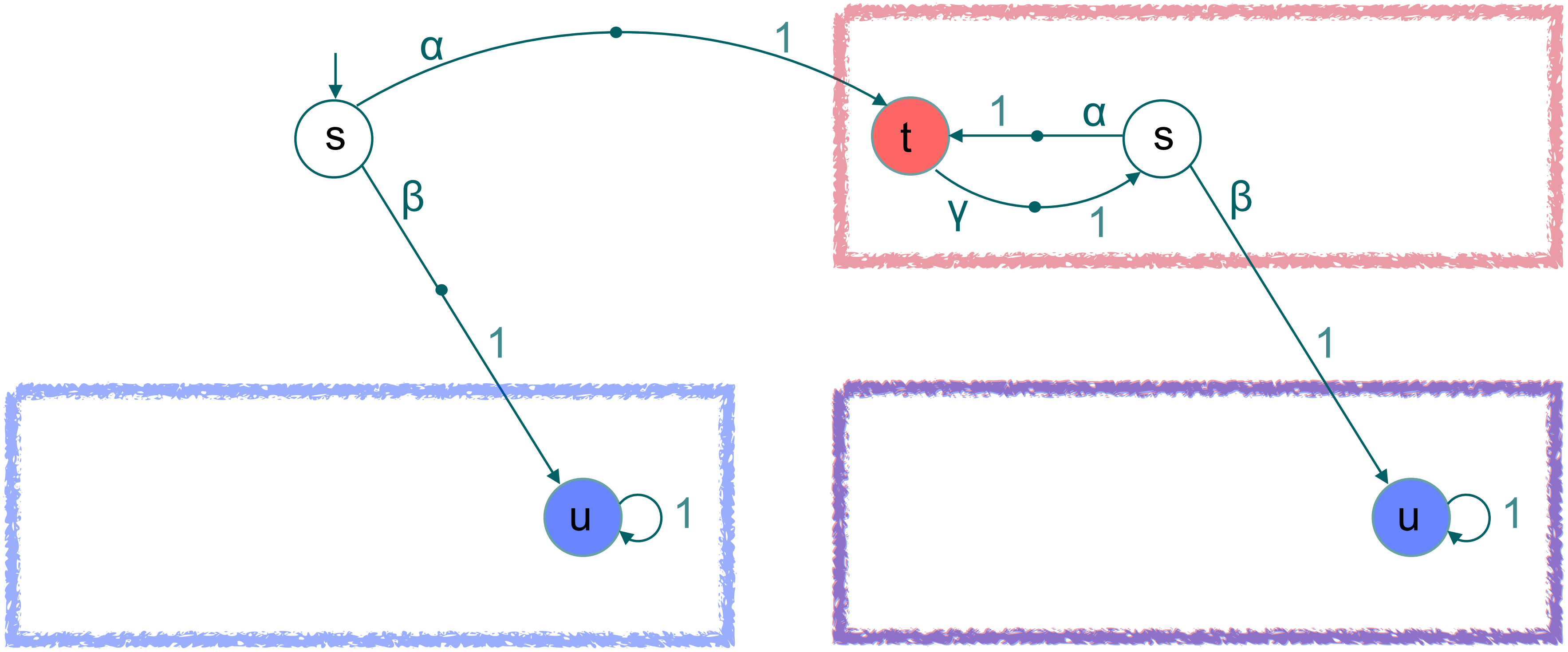
Goal-unfolding

- A policy might need to memorise which set G_i has been reached already
- Idea: Encode this information into the state-space.
- Then, positional (randomised) policies suffice



Goal-unfolding

- A policy might need to memorise which set G_i has been reached already
- Idea: Encode this information into the state-space.
- Then, positional (randomised) policies suffice



Goal-unfolding

Definition:
The goal-unfolding of MDP $M = (S, Act, \mathbf{P}, s_{init})$ and $G_1, \ldots, G_m \subseteq S \setminus \{s_{init}\}$ is the MDP $M_U = (S \times \{0,1\}^m, Act, \mathbf{P}_U, \langle s_{init}, (0, \ldots, 0)\rangle)$, where

$$\mathbf{P}_U(\langle s,\mathbf{b}\rangle, \alpha, \langle t,\mathbf{c}\rangle) = \begin{cases} \mathbf{P}(s, \alpha, t) & \text{, if } \mathbf{c} = succ(\mathbf{b},t) \\ 0 & \text{, otherwise} \end{cases}$$

and for $i \in \{1, \ldots, m\}$:

$$succ(\mathbf{b},t)[i] = \begin{cases} 1 & \text{, if } t \in G_i \\ \mathbf{b}[i] & \text{, otherwise} \end{cases}$$

- The size of M_U is polynomial in the size of M and exponential in m

Goal-unfolding

Lemma: **p** is achievable in M iff **p** is achievable in M_U.

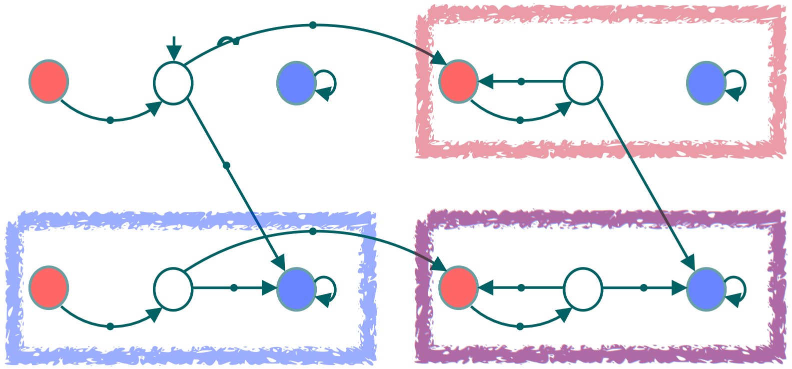
- To answer a multi-objective query for M, we can analyse M_U instead

Lemma: **p** is achievable in M_U iff **p** is achieved by a **positional** policy.

- For the analysis of M_U we only need to consider positional policies

Overview

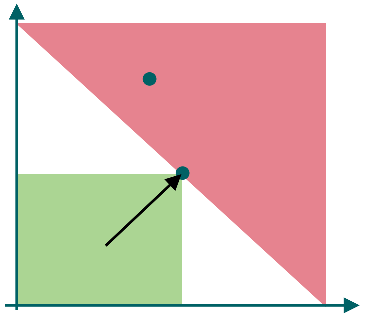
- **MDPs with Multiple Objectives**



- **Linear Programming Approach**

$f_{in}(s) =$	$f_{out}(s)$	for all $s \in S / S_{\perp}$
$y_{s,\alpha,t} \geq$	0	for all $s,t \in S$, $\alpha \in Act(s)$
$\Sigma_{s \in S\text{-}i}\, \Sigma_{\alpha \in Act(s)}\, \Sigma_{t \in S+i}\, y_{s,\alpha,t} \geq$	p_i	for all $i \in \{1, \ldots, m\}$

- **Weighted Sum Approach**



Linear Programming Approach

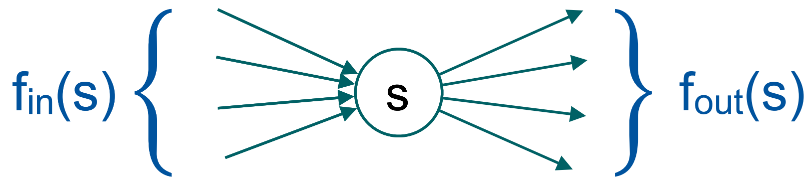
- Idea: Use variables $y_{s,\alpha}$ to encode the expected number of times we leave state s via action $\alpha \in Act(s)$

Exp. times s is entered

- $f_{in}(s) \coloneqq \Sigma_{t \in S} \Sigma_{\alpha \in Act(t)} y_{t,\alpha} \cdot \mathbf{P}_U(t, \alpha, s) + \begin{cases} 1 & \text{, if s is the initial state} \\ 0 & \text{, otherwise} \end{cases}$

- $f_{out}(s) \coloneqq \Sigma_{\alpha \in Act(s)} y_{s,\alpha}$

Exp. times s is left



- We assert $f_{in}(s) = f_{out}(s)$

Solving Quantitative Queries

Quantitative Query:
Given: MDP M, goal state sets $G_1, \ldots, G_m$, $p_2, \ldots, p_m \in \mathbb{R}$
Output: $\max \{ p_1 \in \mathbb{R} \mid (p_1, p_2, \ldots, p_m) \in \mathfrak{A}(\langle \Diamond G_i \rangle_{\mathbf{m}}) \}$

- Consider the goal-unfolding M_U
- For $i \in \{1, \ldots, m\}$ let $S_{-i} = \{ \langle s,\mathbf{b} \rangle \mid \mathbf{b}[i] = 0 \}$ and $S_{+i} = \{ \langle s,\mathbf{b} \rangle \mid \mathbf{b}[i] = 1 \}$
- Let $S_\perp = \{ \langle s,\mathbf{b} \rangle \mid \langle t,\mathbf{c} \rangle \in Post^*(\langle s,\mathbf{b} \rangle) \text{ implies } \mathbf{c} = \mathbf{b} \}$
- Return the optimum of the following LP:

Exp. times G_1 is entered

$$\max \sum_{s\in S_{-1}} \sum_{\alpha\in Act(s)} \sum_{t\in S_{+1}} y_{s,\alpha} \cdot \mathbf{P}_U(s, \alpha, t) \quad \text{such that:}$$

$$f_{in}(s) = f_{out}(s) \quad \text{for all } s \in S / S_\perp$$

$$y_{s,\alpha} \geq 0 \quad \text{for all } s \in S, \alpha \in Act(s)$$

$$\sum_{s\in S_{-i}} \sum_{\alpha\in Act(s)} \sum_{t\in S_{+i}} y_{s,\alpha} \cdot \mathbf{P}_U(s, \alpha, t) \geq p_i \quad \text{for all } i \in \{2, \ldots, m\}$$

Exp. times G_i is entered

Solving Achievability Queries

Achievability Query:
Given: MDP M, goal state sets $G_1, \ldots, G_m$, $\mathbf{p} \in \mathbb{R}^m$
Output: True iff $\mathbf{p} \in \mathfrak{A}(\langle \lozenge G_i \rangle_m)$

- Consider the goal-unfolding M_U
- For $i \in \{1, \ldots, m\}$ let $S_{-i} = \{ \langle s,\mathbf{b} \rangle \mid \mathbf{b}[i] = 0 \}$ and $S_{+i} = \{ \langle s,\mathbf{b} \rangle \mid \mathbf{b}[i] = 1 \}$
- Let $S_\perp = \{ \langle s,\mathbf{b} \rangle \mid \langle t,\mathbf{c} \rangle \in Post^*(\langle s,\mathbf{b} \rangle) \text{ implies } \mathbf{c} = \mathbf{b} \}$
- Return True iff the following LP has a feasible solution

max 0 such that:

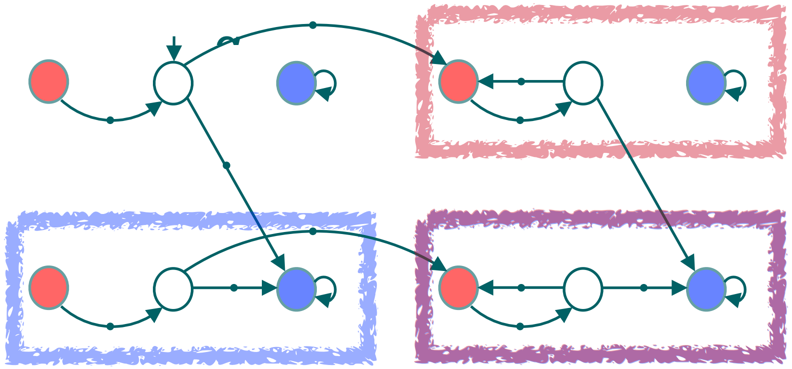
$$f_{in}(s) = f_{out}(s) \quad \text{for all } s \in S / S_\perp$$

$$y_{s,\alpha} \geq 0 \quad \text{for all } s \in S,\ \alpha \in Act(s)$$

$$\Sigma_{s \in S_{-i}} \Sigma_{\alpha \in Act(s)} \Sigma_{t \in S_{+i}}\, y_{s,\alpha} \cdot \mathbf{P}_U(s, \alpha, t) \geq p_i \quad \text{for all } i \in \{1, \ldots, m\}$$

Overview

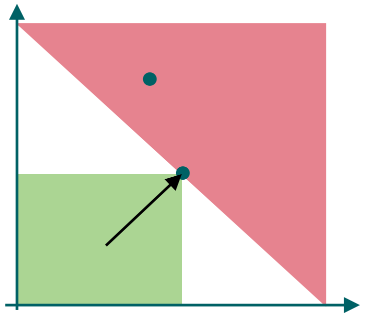
- **MDPs with Multiple Objectives**



- **Linear Programming Approach**

$f_{in}(s) =$	$f_{out}(s)$	for all $s \in S / S_{\perp}$
$y_{s,\alpha,t} \geq$	0	for all $s,t \in S$, $\alpha \in Act(s)$
$\Sigma_{s\in S\text{-}i}\ \Sigma_{\alpha\in Act(s)}\ \Sigma_{t\in S\text{+}i}\ y_{s,\alpha,t} \geq$	p_i	for all $i \in \{1, \ldots, m\}$

- **Weighted Sum Approach**



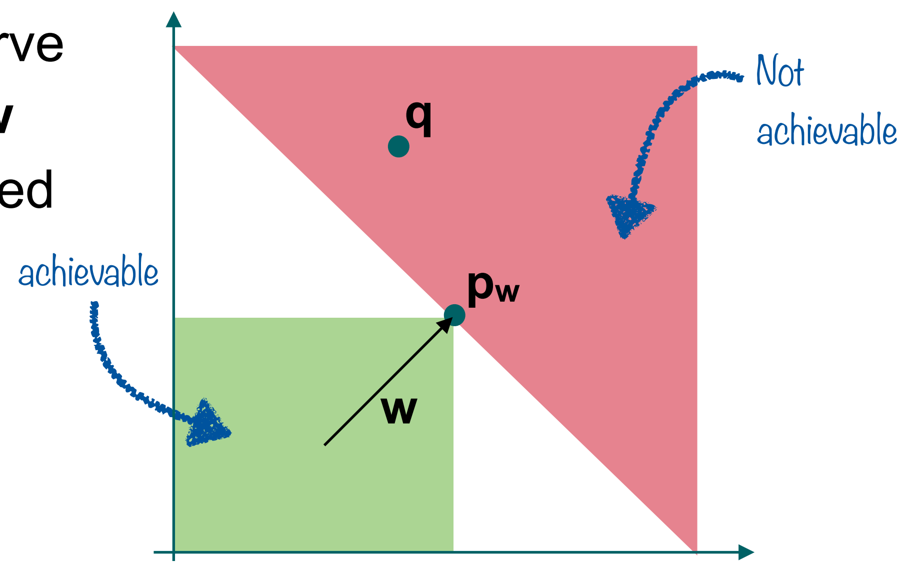
Weighted Sum Approach

Theorem:
For $\mathbf{w} \in [0,1]^m$ let

- $\mathfrak{S}_\mathbf{w} \in \arg\max_\mathfrak{S} \left(\Sigma_{1 \le i \le m} \, \mathbf{w}[i] \cdot Pr^\mathfrak{S}(\lozenge G_i) \right)$ and
- $\mathbf{p}_\mathbf{w} = \left(Pr^{\mathfrak{S}_\mathbf{w}}(\lozenge G_1), \ldots, Pr^{\mathfrak{S}_\mathbf{w}}(\lozenge G_m) \right)$.

Then, $\mathbf{p}_\mathbf{w} \in \mathfrak{A}(\langle \lozenge G_i \rangle_m)$ and for all $\mathbf{q} \in \mathbb{R}^m$: $\mathbf{w} \cdot \mathbf{q} > \mathbf{w} \cdot \mathbf{p}_\mathbf{w}$ implies $\mathbf{q} \notin \mathfrak{A}(\langle \lozenge G_i \rangle_m)$

- In particular, $\mathbf{p}_\mathbf{w}$ lies on the Pareto curve
- Approach: Compute $\mathbf{p}_\mathbf{w}$ for different $\mathbf{w}$
- Stop when the Pareto curve is explored



Software Modeling and Verification Chair

RWTH AACHEN UNIVERSITY

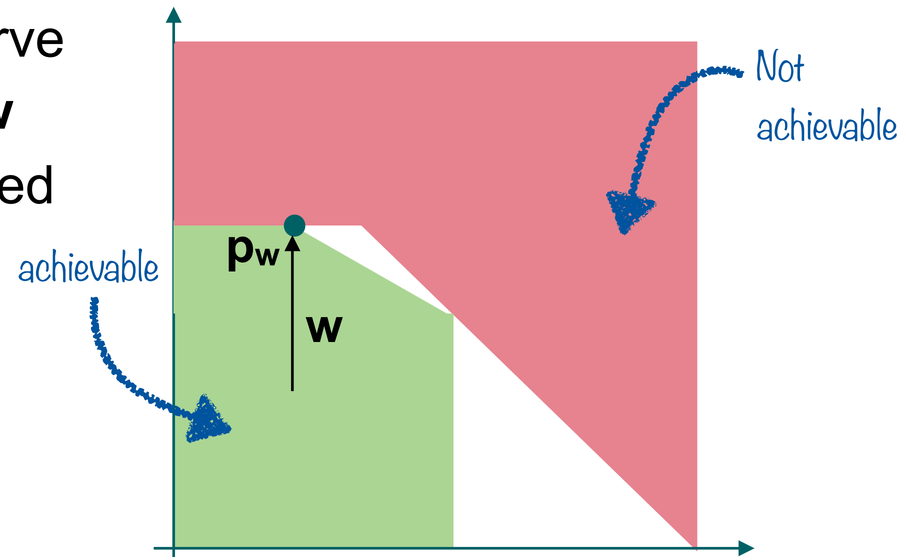
Weighted Sum Approach

Theorem:
For $\mathbf{w} \in [0,1]^m$ let

- $\mathfrak{S}_\mathbf{w} \in \arg\max_{\mathfrak{S}} \left(\Sigma_{1 \le i \le m}\, \mathbf{w}[i] \cdot Pr^{\mathfrak{S}}(\Diamond G_i) \right)$ and
- $\mathbf{p_w} = \left(Pr^{\mathfrak{S}_w}(\Diamond G_1), \ldots, Pr^{\mathfrak{S}_w}(\Diamond G_m) \right)$.

Then, $\mathbf{p_w} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ and for all $\mathbf{q} \in \mathbb{R}^m$: $\mathbf{w} \cdot \mathbf{q} > \mathbf{w} \cdot \mathbf{p_w}$ implies $\mathbf{q} \notin \mathfrak{A}(\langle \Diamond G_i \rangle_m)$

- In particular, $\mathbf{p_w}$ lies on the Pareto curve
- Approach: Compute $\mathbf{p_w}$ for different $\mathbf{w}$
- Stop when the Pareto curve is explored
- Recall: $\mathfrak{A}(\langle \Diamond G_i \rangle_m)$ is convex



Tim Quatmann, Joost-Pieter Katoen
Software Modeling and Verification Chair
RWTH AACHEN UNIVERSITY

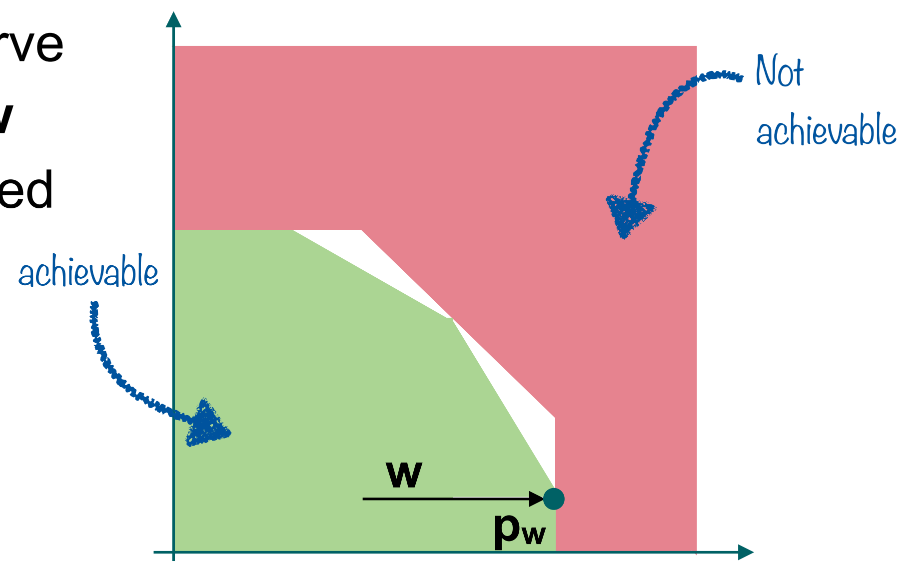
Weighted Sum Approach

Theorem:
For $\mathbf{w} \in [0,1]^m$ let

- $\mathfrak{S}_\mathbf{w} \in \arg\max_{\mathfrak{S}} \left(\Sigma_{1 \leq i \leq m} \mathbf{w}[i] \cdot Pr^{\mathfrak{S}}(\Diamond G_i) \right)$ and
- $\mathbf{p}_\mathbf{w} = \left(Pr^{\mathfrak{S}_w}(\Diamond G_1), \ldots, Pr^{\mathfrak{S}_w}(\Diamond G_m) \right)$.

Then, $\mathbf{p}_\mathbf{w} \in \mathfrak{A}(\langle \Diamond G_i \rangle_m)$ and for all $\mathbf{q} \in \mathbb{R}^m$: $\mathbf{w} \cdot \mathbf{q} > \mathbf{w} \cdot \mathbf{p}_\mathbf{w}$ implies $\mathbf{q} \notin \mathfrak{A}(\langle \Diamond G_i \rangle_m)$

- In particular, $\mathbf{p}_\mathbf{w}$ lies on the Pareto curve
- Approach: Compute $\mathbf{p}_\mathbf{w}$ for different $\mathbf{w}$
- Stop when the Pareto curve is explored
- Recall: $\mathfrak{A}(\langle \Diamond G_i \rangle_m)$ is convex



Weighted Sum Approach

Theorem:
For $\mathbf{w} \in [0,1]^m$ let

- $\mathfrak{S}_\mathbf{w} \in \arg\max_\mathfrak{S} \left(\Sigma_{1 \le i \le m}\, \mathbf{w}[i] \cdot Pr^\mathfrak{S}(\lozenge G_i) \right)$ and
- $\mathbf{p}_\mathbf{w} = \left(Pr^{\mathfrak{S}_w}(\lozenge G_1), \ldots, Pr^{\mathfrak{S}_w}(\lozenge G_m) \right)$.

Then, $\mathbf{p}_\mathbf{w} \in \mathfrak{A}(\langle \lozenge G_i \rangle_m)$ and for all $\mathbf{q} \in \mathbb{R}^m$: $\mathbf{w} \cdot \mathbf{q} > \mathbf{w} \cdot \mathbf{p}_\mathbf{w}$ implies $\mathbf{q} \notin \mathfrak{A}(\langle \lozenge G_i \rangle_m)$

- Proof (sketch):
  - $\mathbf{p}_\mathbf{w} \in \mathfrak{A}(\langle \lozenge G_i \rangle_m)$ follows by definition
  - Assume there is $\mathbf{q} \in \mathfrak{A}(\langle \lozenge G_i \rangle_m)$ with $\mathbf{w} \cdot \mathbf{q} > \mathbf{w} \cdot \mathbf{p}_\mathbf{w}$
  - Let $\mathbf{q}$ be achieved by policy $\mathfrak{S}$, i.e., $Pr^\mathfrak{S}(\lozenge G_i) \ge \mathbf{q}[i]$ for all $i \in \{1, \ldots, m\}$
  - $\Sigma_{1 \le i \le m}\, \mathbf{w}[i] \cdot Pr^{\mathfrak{S}_w}(\lozenge G_i) = \mathbf{w} \cdot \mathbf{p}_\mathbf{w} < \mathbf{w} \cdot \mathbf{q} \le \Sigma_{1 \le i \le m}\, \mathbf{w}[i] \cdot Pr^\mathfrak{S}(\lozenge G_i)$
  - Contradiction to definition of $\mathfrak{S}_w$

Computation of Points p_w for given w

Weighted Value Iteration:
Given: MDP M, goal state sets $G_1, \dots, G_m$, $\mathbf{w} \in [0,1]^m$, precision ε
Output: Point $\mathbf{p_w}$

- Consider the goal-unfolding $M_U = (S_U, Act, \mathbf{P}_U, \langle s_{init}, (0, \dots, 0)\rangle)$, where $S_U = S \times \{0,1\}^m$
- Let $\mathbf{g}(\mathbf{b},\mathbf{c}) \in \{0,1\}^m$ with $\mathbf{g}(\mathbf{b},\mathbf{c})[i] = 1$ iff $\mathbf{b}[i] = 0$ and $\mathbf{c}[i] = 1$ for $i \in \{1, \dots, m\}$
- For $\langle s,\mathbf{b}\rangle \in S_U$ and $i \in \{1, \dots, m\}$:
  - $x^0(\langle s,\mathbf{b}\rangle) \leftarrow 0$, $y^{0,i}(\langle s,\mathbf{b}\rangle) \leftarrow 0$, and $\mathfrak{S}_w(\langle s,\mathbf{b}\rangle) \leftarrow \alpha$ for some arbitrary $\alpha \in Act(\langle s,\mathbf{b}\rangle)$
- For $j \in \{1, 2, \dots\}$:
  - $x^j(\langle s,\mathbf{b}\rangle) \leftarrow \max_{\alpha \in Act(\langle s,\mathbf{b}\rangle)} \left(\Sigma_{\langle t,\mathbf{c}\rangle \in S_U} \mathbf{w} \cdot \mathbf{g}(\mathbf{b},\mathbf{c}) + \mathbf{P}_U(\langle s,\mathbf{b}\rangle, \alpha, \langle t,\mathbf{c}\rangle) \cdot x^{j-1}(\langle t,\mathbf{c}\rangle) \right)$
  - $A_{opt} \leftarrow \arg\max_{\alpha \in Act(\langle s,\mathbf{b}\rangle)} \left(\Sigma_{\langle t,\mathbf{c}\rangle \in S_U} \mathbf{w} \cdot \mathbf{g}(\mathbf{b},\mathbf{c}) + \mathbf{P}_U(\langle s,\mathbf{b}\rangle, \alpha, \langle t,\mathbf{c}\rangle) \cdot x^{j-1}(\langle t,\mathbf{c}\rangle) \right)$
  - if $\mathfrak{S}_w(\langle s,\mathbf{b}\rangle) \notin A_{opt}$ then $\mathfrak{S}_w(\langle s,\mathbf{b}\rangle) \leftarrow \alpha$ for some $\alpha \in A_{opt}$
  - For $i \in \{1, \dots, m\}$: $y^{j,i}(\langle s,\mathbf{b}\rangle) \leftarrow \Sigma_{\langle t,\mathbf{c}\rangle \in S_U} \mathbf{g}(\mathbf{b},\mathbf{c})[i] + \mathbf{P}_U(\langle s,\mathbf{b}\rangle, \mathfrak{S}_w(\langle s,\mathbf{b}\rangle), \langle t,\mathbf{c}\rangle) \cdot y^{j-1,i}(\langle t,\mathbf{c}\rangle)$
- Stop when $\max_{\langle s,\mathbf{b}\rangle \in S_U} (x^j(\langle s,\mathbf{b}\rangle) - x^{j-1}(\langle s,\mathbf{b}\rangle)) \le \varepsilon$
- Return point $\mathbf{p_w}$ with $\mathbf{p_w}[i] = y^{j,i}(\langle s_{init}, (0, \dots, 0)\rangle)$

Conclusion

- Multi-objective MDPs
  - Satisfy multiple properties at once
  - Need randomised, finite-memory policies
- Two Approaches
  - Linear programming approach
  - Weighted sum approach
- Active Field of Research
  - Ask us for Bachelor / Master thesis topics

Thank you for your attention