Abstraction heuristics

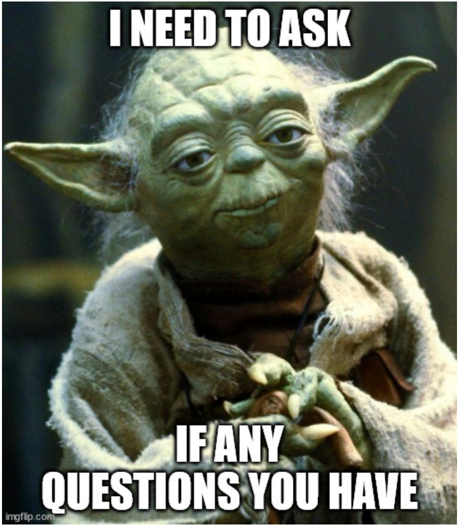
Merge & Shrink

Michaela Urbanovská

PUI Tutorial
Week 7

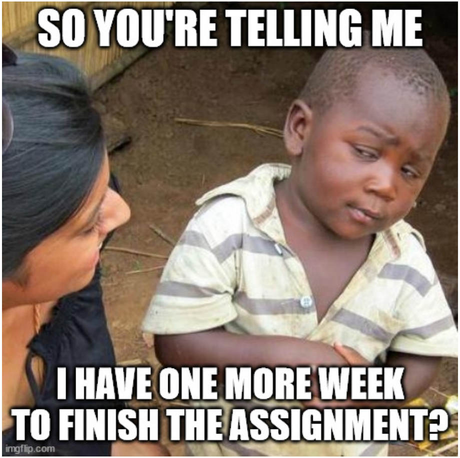
Lecture check

- Any questions regarding the lecture?



Organization stuff

- Lectures reorganized
- Tutorials reorganized in response
- Assignment 1 deadline moved by a week
- New deadline: **11.4.2022**



Obtaining heuristics

- Many different possible ways to obtain a heuristic in general
- One general idea: solve a **simplified** version of the problem
  - relaxation
  - abstraction

Obtaining heuristics

- Many different possible ways to obtain a heuristic in general
- One general idea: solve a **simplified** version of the problem
  - relaxation
  - abstraction
- This week: **abstraction**

Abstraction heuristics

- Simplification of the problem
- Making the problem smaller by dropping state distinctions

Transition system $\mathcal{T} = \langle S, L, T, I, G \rangle$

- S - finite set of states
- L - finite set of labels
- $T \subseteq S \times L \times S$ - transition relation
- $I \subseteq S$ - set of initial states
- $G \subseteq S$ - set of goal states
- $c(l) \in \mathrm{R}_0^+, \forall l \in L$ - cost function for each label

Abstraction heuristics

Transition system for problem P

Transition system $\mathcal{T}(P)$ is defined for FDR problem $P = \langle V, O, s_{init}, s_{goal}, c \rangle$.
The mapping goes as followed:

- S is set of states over V
- $L = O$
- $T = \{(s, o, t) | res(o, s) = t\}$
- $I = \{s_{init}\}$
- $G = \{s | s \in S, s \text{ is consistent with } s_{goal}\}$

Abstraction heuristics

Abstraction definition

- Let's have two transition systems $\mathcal{T}^1 = \langle S^1, L, T^1, I^1, G^1 \rangle$ and $\mathcal{T}^2 = \langle S^2, L, T^2, I^2, G^2 \rangle$ with the same set of labels L.
- Let's have an **abstraction function** $\alpha : S^1 \mapsto S^2$ which maps S^1 to S^2.
- S^2 is an **abstraction** of S^1 if
  - $\forall s \in I^1$ holds that $\alpha(s) \in I^2$
  - $\forall s \in G^1$ holds that $\alpha(s) \in G^2$
  - $\forall (s, l, t) \in T^1$ holds that $(\alpha(s), l, \alpha(t)) \in T^2$

Abstraction heuristics

Abstraction heuristic

Let P denote an FDR planning task and $\mathcal{A}$ denote an **abstraction** of its transition system $\mathcal{T}(P)$.
Abstraction heuristic induced by $\mathcal{A}$ and α is the function

$$h^{\mathcal{A},\alpha} = h^*_{\mathcal{A}}(\alpha(s)), \forall s \in S$$

Abstraction heuristics

Abstraction heuristic

Let P denote an FDR planning task and $\mathcal{A}$ denote an **abstraction** of its transition system $\mathcal{T}(P)$.
Abstraction heuristic induced by $\mathcal{A}$ and α is the function

$$h^{\mathcal{A},\alpha} = h^*_{\mathcal{A}}(\alpha(s)), \forall s \in S$$

Synchronized product

Given two transition systems $\mathcal{T}^1 = \langle S^1, L, T^1, I^1, G^1 \rangle$ and $\mathcal{T}^2 = \langle S^2, L, T^2, I^2, G^2 \rangle$ with the same labels, their **synchronized product** $\mathcal{T}^1 \otimes \mathcal{T}^2 = \mathcal{T}$ is a transition system $\mathcal{T} = \langle S, L, T, I, G \rangle$, where

- $S = S^1 \times S^2$
- $T = \{((s_1, s_2), l, (t_1, t_2)) | (s_1, l, s_2) \in T^1, (s_2, l, t_2) \in T^2\}$
- $I = I^1 \times I^2$
- $G = G^1 \times G^2$

Merge & Shrink heuristic

- Different types of abstraction heuristics
- How to select α?
- In this tutorial: **merge & shrink**
- Consists of
  - merging = computing synchronized products of the abstractions
  - shrinking = abstracting the abstractions further
- There are many strategies...so we will just focus on the main thought behind it

Merge & Shrink heuristic

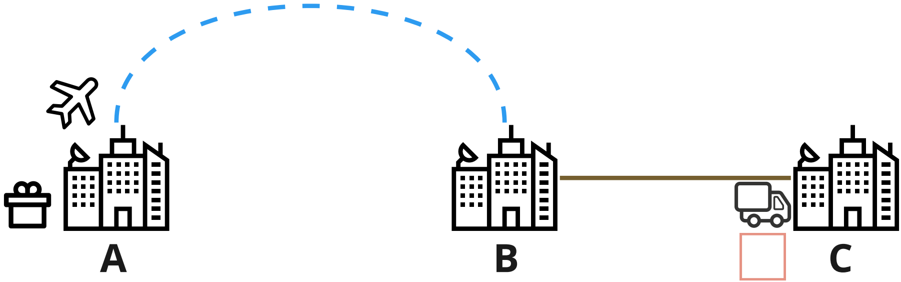
Transition systems $\mathcal{T}^1$ and $\mathcal{T}^2$

- $L^1 = L^2 = \{a, b, c, d, e\}$
- $S^1 = \{A, B, C, D\}$
- $T^1 = \{(A, a, B), (B, b, C), (C, c, A), (A, d, A), (A, e, D)\}$
- $I^1 = \{A\}$
- $G^1 = \{A, C\}$
- $S^2 = \{X, Y, Z\}$
- $T^2 = \{(X, a, Y), (X, a, Z), (Y, b, Z), (Z, c, Y), (Z, d, Y), (Z, e, Z)\}$
- $I^2 = \{X\}$
- $G^2 = \{X\}$

Let's compute synchronized product!

Logistics example

Let's try an example we know well...



FDR representation

FDR problem $P = \langle V, O, s_I, s_G, c \rangle$

$V = \{a, t, p\}$

$D_a = \{A, B\}$ $D_t = \{B, C\}$ $D_p = \{A, B, C, a, t\}$

$s_I = \{a = A, t = C, p = A\}$

$s_G = \{p = C\}$

$O =$

	pre	eff	c
fAB	a=A	a=B	1
fBA	a=B	a=A	1
dBC	t=B	t=C	1
dCB	t=C	t=B	1
laA	a=A, p=A	p=a	1
laB	a=B, p=B	p=a	1
ltB	t=B, p=B	p=t	1
ltC	t=C, p=C	p=t	1
uaA	p=a, a=A	p=A	1
uaB	p=a, a=B	p=B	1
utB	p=t, t=B	p=B	1
utC	p=t, t=C	p=C	1

Atomic projections

- One possible representation is by **atomic projections**
- One transition system for one variable from $V = \{a, t, p\}$

$\mathcal{T}^{\mathbf{a}}$
$S^a = \{aA, aB\}$
$I^a = \{aA\}$
$G^a = \{aA, aB\}$

$\mathcal{T}^{\mathbf{t}}$
$S^t = \{tB, tC\}$
$I^t = \{tC\}$
$G^t = \{tB, tC\}$

$\mathcal{T}^{\mathbf{p}}$
$S^p =$
$\{pA, pB, pC, pa, pt\}$
$I^p = \{pA\}$
$G^p = \{pC\}$

Transitions in atomic projections

$L = \{fAB, fBA, dBC, dCB, laA, laB, uaA, uaB, ltB, ltC, utB, utC\}$

Transitions in atomic projections

$L = \{fAB, fBA, dBC, dCB, laA, laB, uaA, uaB, ltB, ltC, utB, utC\}$
$T^a = \{(aA, fAB, aB), (aB, fBA, aA)\}$

Transitions in atomic projections

$L = \{fAB, fBA, dBC, dCB, laA, laB, uaA, uaB, ltB, ltC, utB, utC\}$
$T^a = \{(aA, fAB, aB), (aB, fBA, aA)\}$
$T^t = \{(tB, dBC, tC), (tC, dCB, tB)\}$

Transitions in atomic projections

$L = \{fAB, fBA, dBC, dCB, laA, laB, uaA, uaB, ltB, ltC, utB, utC\}$
$T^a = \{(aA, fAB, aB), (aB, fBA, aA)\}$
$T^t = \{(tB, dBC, tC), (tC, dCB, tB)\}$
$T^p = \{(pA, laA, pa), (pa, uaA, pA), (pa, uaB, pB), (pB, laB, pa),$
$(pB, ltB, pt), (pt, utB, pB), (pt, utC, pC), (pC, ltC, pt)\}$

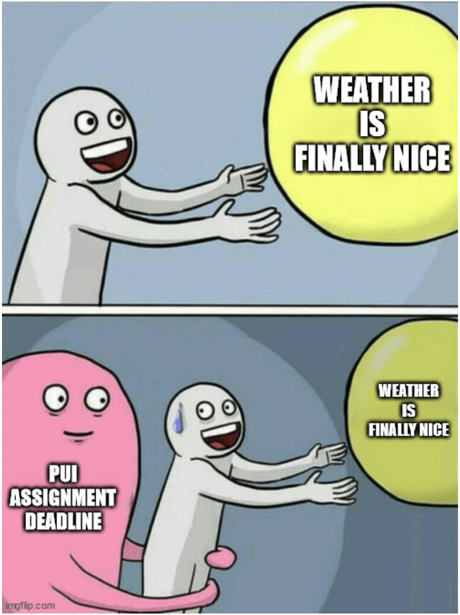
Merge & Shrink

1. Create atomic projections (one per variable)
2. Merge two arbitrary transition systems (synchronized product)
3. Shrink the new transition graph (merge states together to create smaller abstraction)
4. Repeat 2 and 3 until you're left with one abstraction in which you can find the solution

Recap

- How to create synchronized product
- Atomic projections
- Merge & Shrink main principle
  - **merging** = creating synchronized products of two transition systems
  - **shrinking** = creating smaller abstraction

The End



Feedback form link