AI Planning

YY. Exam Preparation

Ready, Steady, Go!

Álvaro Torralba, Cosmina Croitoru

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Winter Term 2018/2019

Thanks to Prof. Jörg Hoffmann for slide sources

Agenda

1. Basic Facts
2. Exam-Relevant Contents
3. Example Exercises

Where and When: Exam

Building E1 3, HS003
Monday, 11th February 2019
14:00 – 16:30

Where and When: Exam Inspection

Building E1 1, room 3.06
Thursday, 14th February 2019
10:00 – 12:00

Fact Sheet

- The exam will be about applying the algorithms/results from the course to example planning tasks.
- Ultimate reference are the post-handouts.
- ANY slides/books/papers allowed.
- NO laptops or mobile phones in exam! Pocket calculators won't be needed.
- The exam will be 120 minutes, starting from the moment we allow you to look at the exercises.
- If we start a bit late, we end a bit late.
- You must bring your own paper to write on.

Exam-Relevant Contents

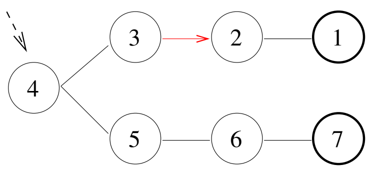
Here's what is NOT relevant to the exam:

- Chapters 1, 3, 4, 21, and the Christmas Surprise Lecture.
- Chapter 2 Section "Extended Planning Frameworks".
- Chapter 8 Section "Graphplan Representation".
- Chapter 13 Section "Concrete Merge-and-Shrink Strategies", "M&S Abstraction Mappings".
- Chapter 15 Section "ps. Landmarks and Hitting Sets".
- Chapter 16 Section "A Walk Through the Zoo".

→ Everything else is relevant.

Causal Graphs and Domain Transition Graphs

Consider this planning task:



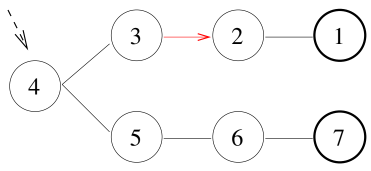
- Variables: $at : \{1, \dots, 7\}$; $v(x) : \{T, F\}$ for $x \in \{1, \dots, 7\}$.
- Actions: $drive(x, y)$ where $|x - y| = 1$ and $(x, y) \neq (2, 3)$; precondition: $\{at = x\}$, effect: $\{at = y, v(y) = T\}$.
- Costs: Unit.
- Initial state: $at = 4, v(4) = T, v(x) = F$ for $x \neq 4$.
- Goal: $v(1) = T$, $v(7) = T$.

(i) Draw the causal graph of this task.

(ii) Draw the domain transition graphs of variables at and $v(2)$ in this task. Annotate the arcs in these graphs with their outside conditions.

$h^{\max}$ and h^{FF}

Consider this planning task:



- Variables: $at : \{1, \dots, 7\}$; $v(x) : \{T, F\}$ for $x \in \{1, \dots, 7\}$.
- Actions: $drive(x, y)$ where $|x - y| = 1$ and $(x, y) \neq (2, 3)$; precondition: $\{at = x\}$, effect: $\{at = y, v(y) = T\}$.
- Costs: Unit.
- Initial state: $at = 4, v(4) = T, v(x) = F$ for $x \neq 4$.
- Goal: $v(1) = T$, $v(7) = T$.

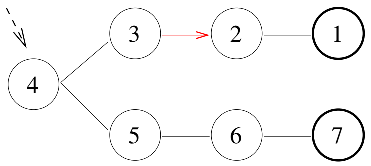
(i) Compute $h^{\max}(I)$ using the dynamic programming algorithm from the lecture. Write down the table constructed by this algorithm (you may leave out the columns for facts true in the initial state, as these will remain 0 throughout anyway).

(ii) What is the $h^{\max}$ best-supporter function $bs_I^{\max}$? Give your answer in the form of a table mapping facts to actions.

(iii) Execute the relaxed plan extraction algorithm from the lecture, using $bs_I^{\max}$. Indicate the facts opened and closed during the execution of the algorithm.

h^+ and Search

Consider this planning task:



- Variables: $at : \{1, \dots, 7\}$; $v(x) : \{T, F\}$ for $x \in \{1, \dots, 7\}$.
- Actions: $drive(x, y)$ where $|x - y| = 1$ and $(x, y) \neq (2, 3)$; precondition: $\{at = x\}$, effect: $\{at = y, v(y) = T\}$.
- Costs: Unit.
- Initial state: $at = 4, v(4) = T, v(x) = F$ for $x \neq 4$.
- Goal: $v(1) = T$, $v(7) = T$.

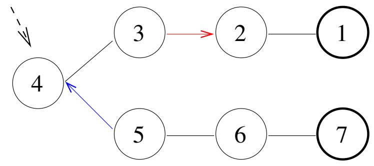
Run forward greedy best-first search, with the modifications explained below. Draw the search space, notating the states by their variable values; annotate each state with its heuristic value, and draw an edge from each state s to each of its successor states s' inserted into the open list when expanding s.

As the heuristic function, use h^+. In the open list, if two states s_1 and s_2 have $h^+(s_1) = h^+(s_2)$, then order s_1 before s_2 if $s_1(at) < 4$ and $s_2(at) > 4$, and vice versa order s_2 before s_1 if $s_2(at) < 4$ and $s_1(at) > 4$; otherwise, choose any order you like.

When generating a child node n', check whether n'.State has been generated beforehand already, and if so, skip it.

Cost Partitioning

Consider this planning task:



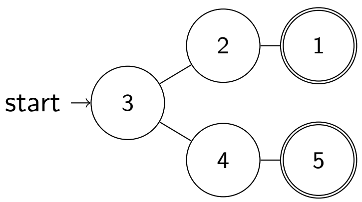
- Variables: $at : \{1, \dots, 7\}$; $v(x) : \{T, F\}$ for $x \in \{1, \dots, 7\}$.
- Actions: $drive(x, y)$ where $|x - y| = 1$ and $(x, y) \neq (2, 3)$; precondition: $\{at = x\}$, effect: $\{at = y, v(y) = T\}$.
- Costs: Unit 1 except $c(drive(5, 4)) = 2$.
- Initial state: $at = 4, v(4) = T, v(x) = F$ for $x \neq 4$.
- Goal: $v(1) = T$, $v(7) = T$.

Assume somebody gives you the following collection of action sub-sets: $\mathcal{C} = \{L_1, L_2\}$ where $L_1 = \{drive(6, 5), drive(5, 4)\}$ and $L_2 = \{drive(5, 4), drive(4, 3)\}$.

(i) What is the value of $h^{LM}_{L_1}(I)$ and $h^{LM}_{L_2}(I)$? Justify your answer.
(ii) What is $h^{\mathcal{C}}(I)$? Justify your answer.
(iii) Write down an LP encoding whose optimal solutions correspond to the optimal cost partitionings for I and $h^{LM}_{L_1}(I), h^{LM}_{L_2}(I)$.
(iv) What is an optimal cost partitioning for I and $h^{LM}_{L_1}(I), h^{LM}_{L_2}(I)$? Does this improve on $h^{\mathcal{C}}(I)$? Justify your answer.

Symmetries

Consider this planning task:



- Variables: $at : \{1, \dots, 5\}$; $v(x) : \{T, F\}$ for $x \in \{1, \dots, 7\}$.
- Actions: $drive(x, y)$ where $|x - y| = 1$; precondition: $\{at = x\}$, effect: $\{at = y, v(y) = T\}$.
- Initial state: $at = 3, v(4) = T, v(x) = F$ for $x \neq 4$.
- Goal: $v(1) = T$, $v(5) = T$.

- Draw the Problem Description Graph (PDG) as specified in the lecture. Indicate different type of nodes with different symbols (e.g., ○, □, ◇, △).
- Are the following permutations of facts automorphisms of the PDG? All omitted facts are assumed to be mapped onto themselves. Justify your answer.
  $\sigma_1 : \{v(1) \leftrightarrow v(5)\}$, $\sigma_2 : \{v(1) \leftrightarrow v(5), v(2) \leftrightarrow v(4)\}$,
  $\sigma_3 : \{at = 1 \leftrightarrow at = 5, v(1) \leftrightarrow v(7)\}$,
  $\sigma_4 : \{at = 1 \leftrightarrow at = 5, at = 2 \leftrightarrow at = 4, v(1) \leftrightarrow v(5)\}$,
  $\sigma_5 : \{at = 1 \leftrightarrow at = 5, at = 2 \leftrightarrow at = 4, v(1) \leftrightarrow v(5), v(2) \leftrightarrow v(4)\}$.
- Considering the permutations above that form part of an automorphism of the PDG, list the resulting orbits in the state space.