SAARLAND UNIVERSITY ARTIFICIAL INTELLIGENCE
Dr. Álvaro Torralba, Prof. Wolfgang Wahlster
Dr. Cosmina Croitoru, Daniel Gnad, Marcel Steinmetz
Yannick Körber, Michael Barz
Christian Bohnenberger, Sören Bund-Becker, Sophian Guidara,
Alexander Rath, Khansa Rekik, Julia Wichlacz, Anna Wilhelm

Exercise Sheet 7.

Solutions due Tuesday, **June 19**, 16:00 – 16:15, in the lecture hall.[1]

Exercise 25. (2.5 Points)

Consider the following problem. Today, Johannes is traveling by car from Saarbrücken to Frankfurt. As he is a friend of car sharing, he is taking two passengers with him for the trip, "PassengerA" and "PassengerB". Both passengers need to arrive to Frankfurt Central Station. Your task is to model this problem as a STRIPS planning task, assuming the following framework.

PassengerA is to be picked up from Sarrbrücken Central Station, whereas PassengerB wants to be picked up from University Bus Terminal.

PassengerA lives in the city, so he needs to walk to the central station. PassengerB needs to walk from MPI to University Bus Terminal where Johannes will pick him up.

To ride the car, passengers have to give a secret code to Johannes, which is written in a booking form. PassengerA knows the secret code whereas PassengerB does not, but he has the bookinf form so he can read it.

Johannes can drive the car with an action called Drive. As he lives in Dudweiler, he needs to drive to the university and Saarbrücken Central Station then continue to Frankfurt.

(a) Write a STRIPS formalization of the initial state and the goal.

(b) Write a STRIPS formalization of the five actions: drive, readCode, walk, pickupPassenger, dropPassenger. In doing so, please do make use of "object variables", i.e., write

[1] Solutions in paper form only, and solution submission only at the stated time at the stated place. At most 3 authors per solution. All authors must be in the same tutorial group. All sheets of your solution must be stapled together. At the top of the first sheet, you must write the names of the authors and the name of your tutor. Your solution must be placed into the correct box for your tutorial group. Also, you should write the solutions of the exercises in order, in particular, do not interleave parts of different exercises otherwise we may oversee part of your solution. Please, don't use red ink, preferably use a black or blue pen instead. If you don't comply with these rules, 3 points will be subtracted from your score for this sheet.

the actions up in a parameterized way and indicate, for each parameter, by which objects it can be instantiated (e.g. drive (x, y) for all $x, y \in \{SBCentral, Frankfurt, UniBusTerminal, Dudweiler\}$).

In both (a) and (b), make use of the following predicates: (**do not use any other predicates**)

- $At(x, y)$: To indicate that object $x \in \{PassengerA, PassengerB, Car\}$ is at position $y \in \{SBCentral, Frankfurt, UniBusTerminal, Dudweiler, MPI\}$.
- $HasBF(x)$: To indicate that passenger $x \in \{PassengerA, PassengerB\}$ has the booking form.
- $InCar(x)$: To indicate that passenger $x \in \{PassengerA, PassengerB\}$ is currently in the car.
- $KnowsCode(x)$: To indicate that passenger $x \in \{PassengerA, PassengerB\}$ knows the secret code.
- $Arrived(x)$: To indicate that passenger $x \in \{PassengerA, PassengerB\}$ arrived at Frankfurt Central Station.

Clarifications:

- The total set of locations are: $\{SBCity, SBCentral, UniBusTerminal, Dudweiler, MPI, Frankfurt\}$.
- PickupPassenger action at a location: it requires the car and the passenger to be at the same location. It requires the passenger to know the code. The set of locations where a passenger can be picked-up is $\{SBCentral, UniBusTerminal\}$.
- DropPassenger action at a location: it requires the car to be at that location and the passenger should have been picked up. The set of locations where a passenger can be dropped is $\{SBCentral, UniBusTerminal, Frankfurt\}$. Dropping a passenger means that he has not been picked up anymore (we have renamed $PickedUp(x)$ by $InCar(x)$ to clarify this).
- $walk(x, y)$ is only legal for $x, y \in \{SBCity, SBCentral, UniBusTerminal, Dudweiler, MPI\}$). In particular, one cannot walk from/to Frankfurt.
- The actions should be written in a parametrized way. However, you can also specify special cases separately.

Exercise 26. (2.5 Points)

Consider the following planning task whose initial state and goal are illustrated in Figure 1. A truck T has to transport a package P_1 to its goal location. To do so, it can drive between the two connected locations and (un)load the package at its current position. In the initial state, the truck is at location A and the package is loaded in the truck; In the end, the truck should be at its starting location A and the package at location B.

Figure 1: Initial State and Goal of a Planning Task

1. Complete the following STRIPS formalization of the planning task depicted in Figure 1:

$$\begin{aligned} I &= \{\text{atP}(1, T), \text{atT}(A)\} \\ G &= \{?\} \\ A &= \{drive(x, y) \mid (x, y) \in \{(A, B), (B, A)\}\} \\ &\cup \{load(i, x) \mid i \in \{1\}, x \in \{?\}\} \\ &\cup \{unload(i, x) \mid i \in \{?\}, x \in \{?\}\} \end{aligned}$$

$$\begin{aligned} drive(x, y) &= pre : \{?\} \\ & \quad add : \{?\} \\ & \quad del : \{?\} \\ load(i, x) &= pre : \{?\} \\ & \quad add : \{?\} \\ & \quad del : \{?\} \\ unload(i, x) &= pre : \{?\} \\ & \quad add : \{?\} \\ & \quad del : \{?\} \end{aligned}$$

2. Draw the **reachable part** of the state space of the planning task. Denote every state by the facts that are currently true, i.e., the current positions of the truck and the package (for example, the initial state can be denoted by AT).

3. Modify the STRIPS task such that there is a second package P_2 that is initially at A and should be transported to location B, and the truck has a maximal capacity and can only load *up to* one package.

4. How many states does the state space of the modified task have? Are all states reachable from the initial state? Are all states solvable? Give a short explanation (you do not have to draw the whole state space of the modified task, a few sentences of explanation suffice).

Exercise 27. (3 points)

Dieter Schlau has invited his two best friends Rainer Theoretiker and Claus Clever to a house party and therefore wants to bake cookies. As he loves chocolate cookies, Rainer loves vanilla cookies and Claus prefers nut cookies, he decided to bake three different kinds of cookies. In order to distinguish the different cookies at the party, Dieter wants to buy differently shaped baking dishes: a circle, a star and a rectangle. While moving along a road map that takes the form of a directed binary tree of depth 3 (see Figures 2 and 3), Dieter tries to get the three different baking dishes. Each tree node might contain objects having any of the three different shapes; moving to that node immediately collects all the objects present. Note that the edges are directed, i.e., you can pass them only in one direction. The corresponding STRIPS planning task $\Pi = (P, A, I, G)$ looks as follows:

$$\begin{aligned}
P &= \{at(x) \mid x \in \{A, \dots, O\}\} \cup \{got(c) \mid c \in \{circle, rectangle, star\}\} \\
A &= \{move(x, y) \mid (x, y) \text{ is an edge in the tree}\} \\
I &= \{at(A)\} \\
G &= \{got(circle), got(rectangle), got(star)\}
\end{aligned}$$

The action $move(x, y)$ is specified as follows:

- In case the destination y does not contain any objects:

$$move(x, y) : \quad \begin{aligned} &pre : \{at(x)\} \\ &add : \{at(y)\} \\ &del : \{at(x)\} \end{aligned}$$

  for all x, y in the tree with an edge from x to y.

- In case the destination y does contain objects (shown here for circle and star; other cases are defined accordingly):

$$move(x, y) : \quad \begin{aligned} &pre : \{at(x)\} \\ &add : \{at(y), got(circle), got(star)\} \\ &del : \{at(x)\} \end{aligned}$$

  for all x, y in the tree with an edge from x to y, where y contains both a circle and a star.

Perform Greedy Best-First Search as specified in Chapter 14 of the lecture. As heuristic function, use the following goal counting heuristic: For a state s, $h(s)$ is the number of

goal facts $p \in G$ that are not true in s. (In other words: $h(s) = |G \setminus s|$.) Whenever you need to break ties, i.e., when $h(s) = h(s')$ is minimal for at least two states s and s' in the open list, do so alphabetically. That is, choose the state whose name of the graph node (the node x where the state contains the fact $at(x)$) comes first in the alphabet.
To perform the algorithm, draw the search space, showing all states generated during the search, including an edge from each expanded state to the child nodes generated when expanding it. Identify each state s by the name of the respective graph node (the node x for which "$at(x)$" is true in s), as well as the subset of baking dishes already collected in the state (those shapes x for which "got(x)" is true in s). Further, annotate each state with the pair containing its heuristic value and the number indicating the order in which the node is *selected* for expansion (For generated states that are not selected for expansion, use "–" as the expansion order number.)

(a) Consider the road map in Figure 2.

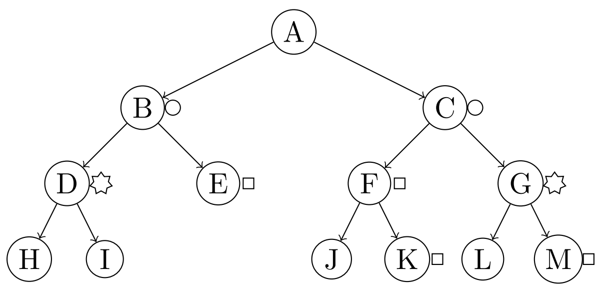

Figure 2: Tree for Exercise 27(a).

(b) Consider the road map in Figure 3.

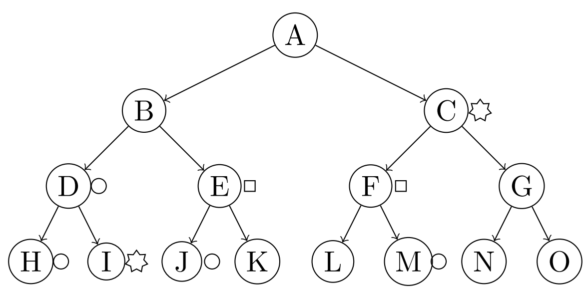

Figure 3: Tree for Exercise 27(b).

(c) What can you say about the behavior of search on the two different road maps of (a) and (b)? What properties of the road maps and of the heuristic function h used are responsible for that behavior? Give a short explanation of 2 - 3 sentences.

Exercise 28. (2 Points)

As specified in the lecture, PolyPlanLen is the problem of deciding, given a STRIPS planning task Π and an integer B bounded by a polynomial in the size of Π, whether or not there exists a plan for Π of length at most B.

Prove that PolyPlanLen is **NP**-hard. Tip: Use a polynomial reduction from SAT.