SAPIENZA
UNIVERSITÀ DI ROMA

Artificial Intelligence

2023/2024 Prof: Sara Bernardini

Lab 9:
STRIPS, FDR,
PDDL

Francesco Argenziano
email: argenziano@diag.uniroma1.it

*The slides have been prepared using the textbook material available on the web, and the slides of the previous editions of the course by Prof. Luigia Carlucci Aiello, Prof. Daniele Nardi and Dott. Fabio Previtali.

Exercises: STRIPS

Consider the following problem: XinYue wants to bake moon cakes for the moon festival. She has three different types of stamps: Luck, Wealth, and Health. She bakes 3 different types of cakes: veggie, chicken, rose. In order to stamp on a cake she has to be focusing it. Initially, XinYue does not focus anything and she can only ever focus a single cake at a time. She can hold a stamp in either hand: left, right. She wants to **only** stamp Health on the veggie cake, Wealth on the chicken cake, and Luck and Health on the rose cake.

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

(b) Give a STRIPS formalization of the actions: LookAt, PickUp, Drop, Stamp. Please use "object variables", i.e., write the actions up in a parameterized way and indicate, for each parameter, by which objects it can be instantiated (e.g. $move(x, y)$ for all $x, y \in \{loc1, loc2, loc3\}$).

Exercises: STRIPS

- In both (a) and (b), you should use the following predicates, but you are free to add new predicates if necessary:
  - $Focusing(c)$: To indicate that XinYue is focusing on cake $c \in \{veggie, chicken, rose, none\}$.
  - $Available(s)$: To indicate that stamp $s \in \{Luck, Wealth, Health\}$ is available, so can be picked up and is not currently held in a hand.
  - $Empty(h)$: To indicate that hand $h \in \{left, right\}$ is empty.
  - $Holding(h, s)$: To indicate that XinYue is holding stamp $s \in \{Luck, Wealth, Health\}$ in hand $h \in \{left, right\}$.
  - $Stamped(c, s)$: To indicate that stamp $s \in \{Luck, Wealth, Health\}$ has been stamped on cake $c \in \{veggie, chicken, rose\}$.

Exercises: FDR

Let $\Pi = \{V, A, I, G\}$ be an FDR planning task describing the job of a modern pigeon, that has to go to the place where the message is (L2), take it, then send it over via sms to the receiver. Note that, due to low mobile signal, the pigeon has to go to L3 to send the message (this is next to a tree - only place that the mobile is working). Define the FDR task formally (V, A, I, G).
Notes:

- Possible locations are L1, L2 and L3. L1 is adjacent to L2, L2 is adjacent to L3. The pigeon can only fly between adjacent locations.
- Predicate names: at-pigeon, mes-found, mes-sent.
- Action names: move,take-mes,send-mes.

FDR 2 STRIPS

Definition (FDR-2-STRIPS). *Let* $\Pi = (V, A, c, I, G)$ *be an FDR planning task. The STRIPS conversion of* Π *is the STRIPS task* $\Pi^{\mathsf{STR}} = (P_V, A^{\mathsf{STR}}, c, I, G)$ *where:*

- $P_V = \{v = d \mid v \in V, d \in D_v\}$ *is the set of (STRIPS) facts.*
- $A^{\mathsf{STR}} = \{a^{\mathsf{STR}} \mid a \in A\}$ *where* $pre_{a^{\mathsf{STR}}} = pre_a$, $add_{a^{\mathsf{STR}}} = eff_a$, *and*

$$del_{a^{\mathsf{STR}}} = \bigcup_{(v=d) \in eff_a} \begin{cases} \{v = pre_a(v)\} & \text{if } pre_a(v) \text{ is defined} \\ \{v = d' \mid d' \in D_v \setminus \{d\}\} & \text{otherwise} \end{cases}$$

- *The cost function* c *is defined by* $c(a^{\mathsf{STR}}) := c(a)$ *for all* $a^{\mathsf{STR}} \in A^{\mathsf{STR}}$.
- I *and* G *are identical to those of* Π.

STRIPS 2 FDR (Naive)

Definition (STRIPS-2-FDR). *Let $\Pi = (P, A, c, I, G)$ be a STRIPS planning task. The FDR conversion of Π is the FDR task $\Pi^{\mathsf{FDR}} = (V_P, A^{\mathsf{FDR}}, c, I^{\mathsf{FDR}}, G^{\mathsf{FDR}})$ where:*

- $V_P = \{v_p \mid p \in P\}$ *is the set of variables, all Boolean.*
- $A^{\mathsf{FDR}} = \{a^{\mathsf{FDR}} \mid a \in A\}$ *where* $pre_{a^{\mathsf{FDR}}} = \{v_p = T \mid p \in pre_a\}$ *and* $eff_{a^{\mathsf{FDR}}} = \{v_p = T \mid p \in add_a\} \cup \{v_p = F \mid p \in del_a\}$.
- *The cost function c is defined by $c(a^{\mathsf{FDR}}) := c(a)$ for all $a^{\mathsf{FDR}} \in A^{\mathsf{STR}}$.*
- $I = \{v_p = T \mid p \in I\}$; *and* $G = \{v_p = T \mid p \in G\}$.

PDDL

- PDDL is the de-facto standard input language in the planning area.
- It allows us to describe to the computer planning domains and problems and let a planner solve them for us
- We define *predicates* instead of STRIPS propositions and *action schemas* instead of STRIPS actions.
- To facilitate everyone, we are going to use an online planner https://editor.planning.domains/#
  - but get Ubuntu asap please