Artificial Intelligence

2023/2024 Prof: Sara Bernardini

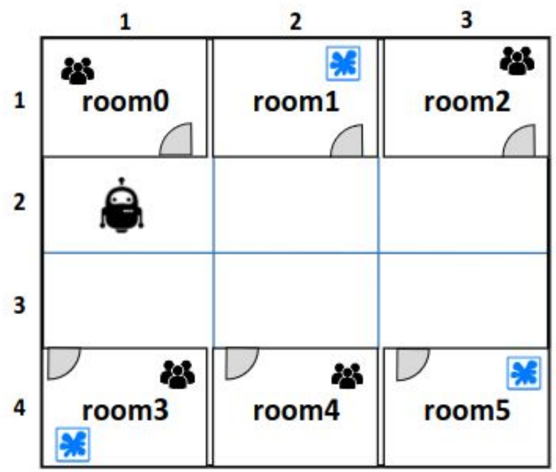
Exercises

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.

MARRtino works in hotel

Our robot MARRtino works in a small hotel, whose map is represented as a grid world (see figure). MARRtino has the task of cleaning the rooms that are marked dirty in the map. MARRtino starts in the cell in front of room0 and can navigate the environment by moving inside the map in any of the 8 adjacent cells, that are traversable; it can also enter the hotel rooms by activating a specialized behavior, when is in the cell in front of the door. Once in a room MARRtino can clean it and then exit. MARRtino should not enter the rooms where it knows there are guests.



(a) Describe the domain in PDDL;

(b) Describe the problem in PDDL;

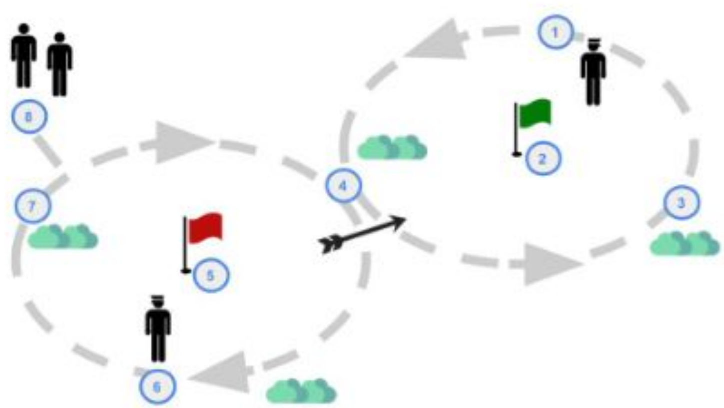
(c) Discuss the forward planning process to reach the goal, using a *perfect* heuristic that gives for each state the number of steps to reach the goal; for each step, show the current state, the applicable actions and the state resulting from the application of the chosen action.

Steal the flag: Track A

The goal of a team of 2 players is to steal 2 flags surveilled by 2 guards. The figure depicts the environment, the initial position of the players, the flags and the arrow. The two guards patrol around the flags in the direction of the grey arrows. You can assume that at any time, each agent (and object) is at one of the node highlighted in the figure.
The players move independently in the environment and need to cooperate in order to steal the flags. To achieve their goal, the players can use different strategies with different actions. In particular, each player can:

- Move between two nodes even though they are not near (starting from the initial situation, for example, agent P0 can move to node 3 instantly, move-P0(3)). While moving, the players do not have to worry about being seen by the guards. In fact, there are enough bushes for them to hide.
- Take the arrow when the two guards are not near. **Two nodes are *near* if the difference of their node value is equal to 1** (e.g. node 2 and node 1 are considered near since |2 - 1| is equal to 1).
- Shoot the arrow if one of the two guard is near and the other is not near. Once shot, the arrow remains at the target node (e.g. if the guard to shoot is at node 1, after the action, the new position of the arrow is node 1).
- Capture the guard if both players and one guard is at a given node, and the other guard is not near that node.
- Steal a flag *iff* the guards have been taken care of.

Note: it is enough to write the action for only one player



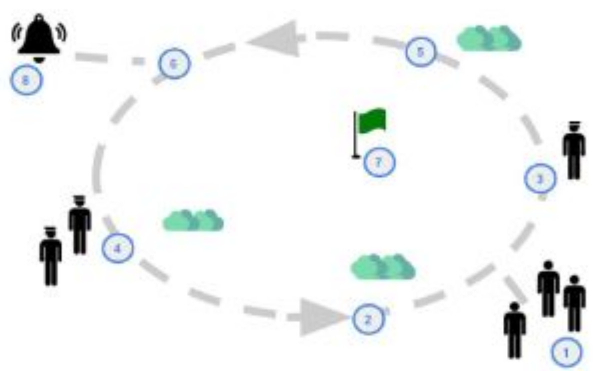
1. Characterize the state-space, the initial and the goal state
2. Describe the operators needed to solve the problem
3. Show one possible solution
4. Draw the first **3 steps** of the tree generated by forward search assuming a perfect heuristic. Show all the actions applicable at the traversed states, and attach to each node the current state

Steal the gold: Track B

The goal of a team of 3 players is to steal the gold guarded by 3 policemen. The figure depicts the environment, the initial position of the players, the policemen, the gold and the bell. The policemen M0 and M1 do not move and remain at their initial location, while M2 patrols the environment in the direction of the grey arrows. You can assume that at any time, each player (and object) is at one of the node highlighted in the figure.
The players move independently in the environment and need to cooperate in order to steal the gold. To achieve their goal, the players can use different strategies with different actions. In particular, each player can:

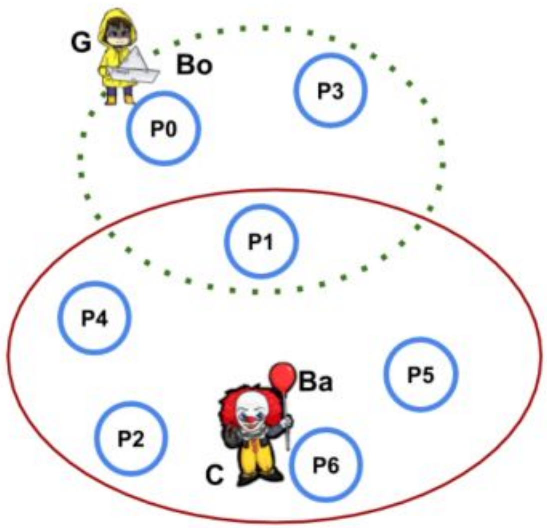
- Move between two nodes even though they are not near (starting from the initial situation, for example, agent P0 can move to node 3 instantly, move-P0(3)). While moving, the players do not have to worry about being seen by the police. In fact, there are enough bushes for them to hide.
- Ring the bell if the policeman is not near their location. **Two nodes are *near* if the difference of their node value is less or equal than 2** (e.g. node 3 and node 1 are considered near since |3 - 1| is less or equal than 2). The policemen will hear the bell ringing and get distracted by it.
- Capture the policemen if they are distracted and the remaining policeman is not near their location.
- Assault the policemen at a particular node, if the number of policemen is less than the number of players (e.g. to assault 2 guards, the 3 players have to be at the same node). To assault does not require the policemen to be distracted but the remaining policeman has to be not near that location.
- Steal the gold *iff* the guards have been taken care of.

Note: it is enough to write the action for only one player



1. Characterize the state-space, the initial and the goal state
2. Describe the operators needed to solve the problem
3. Show one possible solution

Group A. A clown C and Georgie G are tired of playing with their toys and they want to exchange them. G wants to play with the balloon Ba while, C wants to play with the boat Bo. Unfortunately, they are both very shy and they never want to be at the same place Pi. Hence, they have to find a common place (for example $P1$) where to drop and collect objects (they can drop an object in a place and move into another place). G and C cannot hold two objects at the same time. The environment is depicted in the figure, G can only move in places within the dotted line; while C can only move within the continuous line. Places within the same set are all connected (i.e. $\{P0, P1, P3\}$ for the dotted set and $\{P1, P2, P4, P5, P6\}$ for the other). The figure shows the initial state where G holds Bo and he is at $P0$, and C holds Ba and he is at $P6$. The goal state is represented by G at $P3$ holding Ba and C at $P5$ holding Bo.



(a) Define the problem and the domain file in PDDL

(b) Show one possible sequence of actions to a goal state including all the states in the sequence

(c) Draw the first 3 steps of the tree generated by forward search assuming a perfect heuristic (a heuristic choosing the move in the plan given above). Show all the actions applicable at each of the traversed states, and the state reached.

AI experts are well payed unless they are unemployed.
Every AI expert in the company AIco gets an AI bonus.
Everyone getting an AI bonus is employed.
Gino gets an AI bonus.

- (a) Represent the above sentences in first order logic.
- (b) Transform them in CNF and tell whether there are any non-Horn clauses.
- (c) Prove that Gino is well payed by resolution, adding knowledge if necessary.
- (d) Prove by resolution that every AI expert in AIco is well payed.

Illustrate the procedure for generating a clausal (conjunctive) normal form of a generic formula in first order logic.

- (a) provide the CNF of the formula $A \wedge B \iff C \vee D$
- (b) provide the CNF of the formula $\forall x \, \exists y \, f(x) \wedge g(x, y) \Rightarrow \exists z \, h(x, z)$

Children are happy if (and only if) someone makes them happy. Parents make their children happy. Anna is the mother of Paola.

- (a) Represent the above sentences in first order logic.
- (b) Transform them in CNF, and tell whether there are any non-Horn clauses.
- (c) Is Paola a happy child? If she is, prove it using resolution. Otherwise, consider adding some knowledge in order to prove it (except for the trivial addition of $happyChild(Paola)$) and show the resolution proof.

Tell whether the following FOL sentences correctly represent the corresponding English ones. Explain the answer and provide the correct formula if the one given is incorrect.

Consider the state space depicted in Figure 3, where A is the initial state, and F and I are the goal states. The transitions are annotated by their costs.

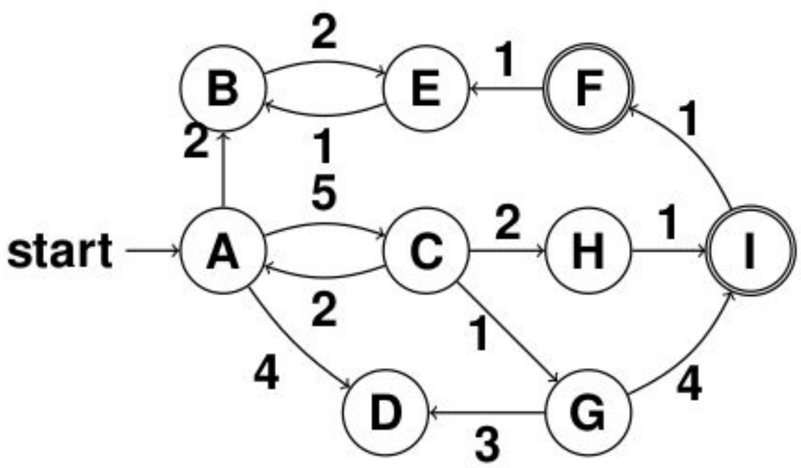


Figure 3: State space

i. Run the uniform-cost search algorithm on this problem. Draw the search graph and annotate **each node** with its g **value and the order in which states are selected for expansion**. Draw duplicate nodes, and mark them accordingly by crossing them out. If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first (e.g., a before d). Give the solution found by uniform-cost search. Is this solution guaranteed to be optimal? Justify your answer.

ii. Run the iterative-deepening search algorithm on this problem until it finds a solution. For each depth, depict the corresponding search tree. Annotate each state with the **order in which states are selected for expansion**. If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first. Give the solution found by iterative-deepening search. Is this solution guaranteed to be optimal? Justify your answer.

(a) Consider the STRIPS planning task with propositions $P = \{a, b, c, d, e\}$, initial state $I = \{\neg a, b, \neg c, d, e\}$, goal $G = \{a, d\}$, and actions $A = \{a_1, a_2, a_3\}$ (expressed as triples of preconditions, add effects and delete effects) where:

- $a_1 = \langle (b, d), (c, e), (d) \rangle$
- $a_2 = \langle (b), (a), (c, d) \rangle$
- $a_3 = \langle (a), (d), \emptyset \rangle$

Solve this problem with a breadth-first search (BFS) using the STRIPS regression method. Draw the search tree that you obtain and record the solution plan. Do not expand a node further if the formula at that node is unsatisfiable or represents a set of states that is a (strict or non-strict) subset of the set of states represented by the formula at a previously expanded node. Specify the result of regression for each node of the BFS tree.

Transform the following formula to CNF, specifying which steps you are applying and giving the intermediate results.

$$\big(\neg A \to (B \leftrightarrow C)\big) \wedge \big(D \to \neg(\neg C \to A)\big)$$

For the following formula, use resolution to determine if the formula is satisfiable or unsatisfiable. Write the resolution process in the form of a tree for easier readability.

$(\neg A \vee \neg D) \wedge (B \vee C \vee \neg D) \wedge (A \vee \neg B \vee \neg D) \wedge (A \vee B) \wedge (\neg A \vee D) \wedge (A \vee \neg B \vee D)$
$\wedge (\neg C \vee \neg D)$

Perform DPLL on the following formula to look for a satisfiable assignment. Assume that DPLL selects variables in alphabetical order (i.e., A,

B, C, D, E, . . .), and that the splitting rule first attempts the value False (F).

$$\{\{C\}, \{\neg A, B\}, \{A, \neg B, C\}, \{\neg B, \neg D\}, \{A, \neg C, D\}, \{B, D\}\}$$

Perform DPLL with clause learning. Assume that DPLL selects variables in alphabetical order (i.e., A, B, C, D, E, ...), and that the splitting rule first attempts the value False (F). Do this until the clause set is proven to be satisfiable or unsatisfiable.

$$\Delta_2 = \{\{A, B, \neg C, D\}, \{B, \neg C, D\}, \{A, \neg D\}, \{\neg A, \neg D\}, \{\neg B, \neg C\}, \{\neg B, C\}, \{A, \neg B, C, D\}, \{C, \neg D, E\}, \{C, \neg D, \neg E\}\}$$

Consider the following constraint network: $\gamma = (V, D, C)$:

- Variables: $V = \{a, b, c, d\}$
- Domains: For all $v \in V$: $D_v = \{5, 10, 15, 20\}$
- Constraints: $a > b + 10; b + d \leq 15; 20 < d + c; a + c > 30;$

1. Draw the constraint graph.
2. Can you solve the problem using the AC-3 algorithm? Please explain why and if you can, proceed to solve the problem using the algorithm.
3. Can you solve the problem using the AcyclicCG algorithm? Please explain why and if you can, proceed to solve the problem using the algorithm.

Consider the following constraint network: $\gamma = (V, D, C)$:

- Variables: $V = \{a, b, c, d, e, f, g\}$
- Domains: for all $u \in V, D_u = \{1, 2, 3, 4, 5, 6\}$
- Constraints: $a = 2d,\ g = 3d,\ d = b,\ e = a - 3,\ c = b + 3,\ f = e + 2$

Run the AC-3 algorithm, as specified in the lecture slides (Chapter 6, Section 4), on the given constraint network. For each iteration of the while-loop, give the content of the data structure M at the start of the iteration, the pair (u, v) removed from M, the domain D_u of u after the call to Revise (γ, u, v), and the pairs (w, u) added into M.

Note: Initialize M as a lexicographically ordered list (i.e., (a, b) would be before (a, c), both before (b, a) etc., if any of those exist). After initialization, use M as a FIFO queue, i.e., always remove the first (oldest) element from the queue and add new elements to the end of the queue.

Consider the following constraint network: $\gamma = (V, D, C)$:

- Variables: $V = \{a, b, c, d, e, f\}$
- Domains: For all $v \in V : D_v = \{1, 2, 3, 4, 5\}$
- Constraints: $c = d,\ b = d - 2,\ e = c - 1,\ f = 2e,\ a = e + 3$

Run the AcyclicCG algorithm on the constraint network by executing its 4 steps.

(a) Draw the constraint graph of γ. Pick a as the root and draw the directed tree obtained by performing step 1.(1 point)

(b) Give the resulting variable ordering obtained by performing step 2. If the ordering of the variables is not unique, break ties using alphabetical order.(1 point)

(c) List the calls to $Revise(\gamma, v_{parent(i)}, v_i)$ in the order executed by running step 3, and, for each of them, give the resulting domain of $v_{parent(i)}(2points)$.

(d) For each recursive call to BacktrackingWithInference in step 4, give the domain D'_{v_i} of the selected variable v_i after Forward Checking and the value $d \in D'_{v_i}$ assigned to v_i (2 points).

For exercises on:
-minimax, alpha beta pruning
-translation from natural language to FOL
there is plenty of material on Internet and tools to use, you shouldn't not have any problem in finding them.

For exercises on STRIPS/FDR: just repeat these exercises on PDDL by adapting them to the other formalism. The transformation between them is always possible.

There won't be any solutions available for these exercises.
Collaborate, discuss and study together. If you are really struggling, only in that case contact me for suggestions/office hours (but still, I won't give you any solution on them, we'll reason together)