

Knowledge Representation and Reasoning 2022

Homework Assignment #3

Due: March 21, 2022, 23:59.

Submit your solution as an .lp file—completing the template.

For more detailed instructions, see:

<https://canvas.uva.nl/courses/28686/assignments/307965>

The assignment

In this homework assignment, you will write an ASP program to solve the following (toy) planning problem. This problem is about finding a way to deliver some packages from a warehouse to specified delivery locations, using several (electric) trucks.

Brief overview of the setting The problem works in time steps. At each time step, each truck can perform exactly one action—move to an adjacent node, wait, (un)load a package, and so on (more details below). The problem operates on a map that consists of a graph with nodes and edges (see Figure 1). All nodes and edges are single capacity, meaning that only a single truck can be on a node or travel on an edge at each time step. The task is to find a plan (a sequence of actions, one per truck per time step) after which the goal state is achieved: all packages are delivered, the trucks are empty and on the parking spots, and the trucks have enough fuel to reach a charging station.

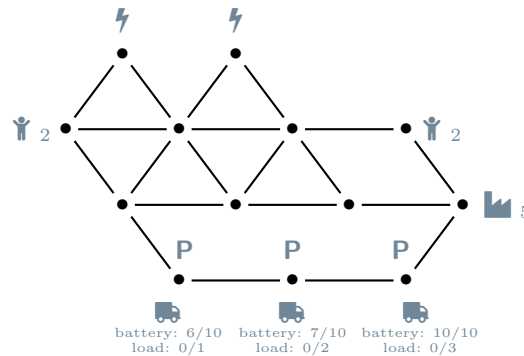


Figure 1: Road map with locations and trucks.

The initial state The initial state of the world is given by the road map depicted in Figure 1. This figure features the following information:

- ⚡: charging station for the trucks (on the corresponding node).
- 🧑_n: delivery location (on the corresponding node) where n packages need to be delivered.
- 🏭_m: warehouse that contains m packages (on the corresponding node).
- P: parking spot (on the corresponding node).
- 🚚: delivery truck (located on the corresponding node).
- battery: ℓ/c : battery level (ℓ) and capacity (c) of a truck.
- load: ℓ/c : package load (ℓ) and capacity (c) of a truck.

Actions and their conditions and effects At each time step, each truck performs exactly one of the following actions:

- It can **wait**—the truck stays in the current location, the battery level stays the same.
- It can **move** to an adjacent node—the battery level goes down by 1 unit.
- If it is at the warehouse, the warehouse is not empty, and the truck still has capacity, it can **load** one single package from the warehouse into the truck—the battery level stays the same.
- If it is at a delivery location that still needs some packages, and the truck has at least one package, it can **unload** one single package at the delivery location—the battery level stays the same.
- If it is at a charging station, and the battery is not at full capacity, it can **charge**—the battery level increases by 1 unit

The following additional restrictions have to be taken into account:

- If the battery level of a truck is 0, it cannot move to another node.
- At each location, there can be at most one truck at the same time.
- Trucks cannot pass each other on a piece of road—e.g., trucks at adjacent locations cannot swap places in a single time step.

The goal The goal is to achieve a state that has all of the following properties:

- All required deliveries must have been performed.
- Each truck is located on a parking spot (which parking spot exactly does not matter).
- Each truck must have no packages aboard.
- Each truck must have a battery level that allows it to reach some charging station—e.g., if the shortest path from the truck to a charging station has length ℓ , then the truck's battery level must be at least ℓ .

The task Encode this problem (for the map depicted in Figure 1) into ASP. That is, write an answer set program whose answer sets correspond to the (shortest) sequences of actions that achieve the goal from the initial state of the world.

Hints and notes:

- There exists a sequence of actions to achieve the goal that requires at most 25 time steps. You may use this upper bound in your solution.
- The example solution for this assignment that we made runs in about 15 seconds on a laptop (using the above upper bound of 25 time steps). Depending on the exact way you encode the problem, it might take a bit more (or less) time to run, but you should expect it to be in the same order of magnitude (e.g., within minutes rather than requiring hours).
- Start simple, and slowly introduce the various elements of the problem. For example, begin with only the move and wait actions—and check if your encoding of this works correctly—before you add other elements like packages, battery level, etc. A useful order in which to implement the different elements is the order in which things are mentioned under “Evaluation Criteria” (below).

Submission instructions

Your submission should be an answer set program (in plain text, as a file with extension `.lp`) and should be submitted via Canvas. In particular, use the provided template, and complete it.

- In the template, the overall code is already split into different sections—e.g., “Input representation”, “Goal conditions”, etc. Please stick to this division (as much as possible).
- For some sections, there is not very much to explain—for example for the section “Input representation”—and for these sections your explanation should take the form of short inline comments between the code.
- For some other sections, a bit more elaborate explanation of your approach is needed. This is indicated in the section header in the template—and please write your explanation there. (For these sections, you may additionally use inline comments.)

Evaluation criteria

Your solution will be graded on correctness as well as on how well you explain what you did. You can roughly achieve the following amounts of points for the following criteria. There are 10 points in total, and your grade will be the number of points that you obtain.

- **6 points:**

- your solution represents the input in an appropriate way (i.e., the road map with its various special locations, the trucks with their location, their initial load and battery level, etc);
- your solution represents the possible actions that can be taken by trucks;
- the answer sets of your solution correspond to appropriate sequences of actions;
- your solution has a (correct) representation of the state of the world at the various time steps, based on the actions that are chosen;
- the goal condition of ‘packages delivered’ is expressed correctly;

- **1 point:**

- your solution correctly takes into account the single capacity property of nodes and edges, i.e., that trucks may not use the same node/edge at the same time;

- **1 point:**

- your solution correctly takes into account the load and battery restrictions into account;

- **0.5 point:**

- the goal condition of ‘trucks at parking spots’ is expressed correctly;
- the goal condition of ‘trucks empty’ is expressed correctly;

- **1 point:**

- the goal condition of ‘trucks within reach of charging station’ is expressed correctly;

- **0.5 point:**

- your solution contains a suitable optimization statement that leads to finding a solution that achieves the goal in a minimal number of time steps.