Lab 4 - CSPs – Artificial Intelligence

Teacher: Stephan Schiffel

February 11, 2022

Note: For this lab, you can work together in teams of up to 3 students. However, this is not a necessity. The assignment is small enough to do it alone in which case you may get more experience.

You will need Python and Google's OR-Tools, a pretty efficient and easy to use library for solving various kinds of constraint satisfaction and constraint optimization problems.

Time Estimate

2 hours in addition to the time spend in the lab class assuming you attended the lectures on CSPs.

Problem Description

The following logic puzzle is known as Zebra Puzzle or Einstein Puzzle (source: Wikipedia). It was first published in the Life International magazine in 1962.

1. There are five houses.
2. The Englishman lives in the red house.
3. The Spaniard owns the dog.
4. Coffee is drunk in the green house.
5. The Ukrainian drinks tea.
6. The green house is immediately to the right of the ivory house.
7. The Old Gold smoker owns snails.
8. Kools are smoked in the yellow house.
9. Milk is drunk in the middle house.
10. The Norwegian lives in the first house.
11. The man who smokes Chesterfields lives in the house next to the man with the fox.
12. Kools are smoked in the house next to the house where the horse is kept.
13. The Lucky Strike smoker drinks orange juice.
14. The Japanese smokes Parliaments.
15. The Norwegian lives next to the blue house.

Now, who drinks water? Who owns the zebra?

To get to a unique solution, we assume that each of the five houses has a different color, each of the five inhabitants has a different nationality, prefers a different brand of cigarettes, a different drink, and owns a different pet. The five houses are arranged in a row (no house has more than 2 neighbors, two houses have just 1 neighbor), left and right are from your point of view, the first house is the left-most one.

Tasks

1. (20 points) Model the problem as a CSP, that is define variables, their domains and constraints between them. There are different ways of modelling this. Typically you want to have fewer variables and smaller domains (to reduce the size of the state space) and fewer or simpler constraints (to speed up constraint propagation). What are your variables and their domains?

   Note that your choice of variables and domains influences how you need to encode the constraints (in part 3). Thus it might be helpful to already think about how you could encode the constraints using your variables.

2. (20 points) How big is the state space? How big is the search tree? Shortly, explain your answer.

3. (60 points) Download the material for the lab and implement your model (implement all things marked with TODO). Your model should have a unique solution.

4. (10 points) Run the program and compare the results (run time and number of branches) to the expected complexity of the problem. How do you explain these different results?

Material

The Python project for the lab can be found on Canvas.

You need to implement the function `setup_csp()` in `einstein.py`. To run the code simply run `python einstein.py`.

Hints

- The assumptions you make for the problem may require adding some constraints as well or you may get multiple solutions. If done correctly, there should be only one solution to the problem.
- A variable in a CSP is not the same as a variable in Python. You need to create variables in the model using on of the `New...Var()` methods.
- Google Or Tools only supports integer variables. If the variables in your model are not numeric, you need to map the non-numeric values to numbers.
- Google Or Tools come with a fairly wide range of predefined constraints. The ones that are most useful for this problem are probably `x == y`, `x != y`, `x == y + 1`, and `|x - y| == 1`.
- To implement your model you need to create one variable object for each variable, set the right domains for the variables and create constraints for the given facts using the variables you defined (typically one constraint for each fact).
- To understand how to use OR-Tools, the guides are quite helpful.

Handing In

Hand in a PDF report with the answers to all the questions the python file with your code on Canvas.