GRUNDLAGEN DER KÜNSTLICHEN INTELLIGENZ
Exercise 3: CSP – Problems 16th and 23th November 2018
Markus Koschi

1 Presented Problems

Problem 3.1: Turning n-ary constraints into binary constraints

(from *Russell & Norvig 3ed.* q. 7.6) Suppose that we have $CSP = (X, D, E$[1]$)$ with

$$\begin{aligned} X &= \{A, B, C\}, \\ D &= \{\text{dom}(A), \text{dom}(B), \text{dom}(C)\}, \\ E &= \big\{\langle (A, B, C),\ A + B = C \rangle\big\}, \end{aligned}$$

where $\text{dom}(A)$, $\text{dom}(B)$, and $\text{dom}(C)$ denote the domain of variable A, B, and C, respectively, and each domain can be $\{0, 1, \dots, 9\}$ for example.

Problem 3.1.1: Draw the constraint hypergraph for the CSP. In this case, a hypergraph is a graph with two types of nodes. The first type of node represents the *variables*, depicted by ○, and the second type of node represents the constraint, depicted by □. Based on the number of variables involved, what is the type of the constraint?

Problem 3.1.2: We can eliminate the higher-order constraint in E by replacing the constraint node □ with a new variable node Z. (We denote this new CSP as CSP'.) What is the domain for variable Z? (Hint: The domain for variable Z can be ordered pairs of other values.) What is the new constraint set E' after introducing the new variable Z?

Problem 3.1.3: Modify CSP' such that it only contains binary constraints and formally express the new $CSP'' = (X'', D'', E'')$.

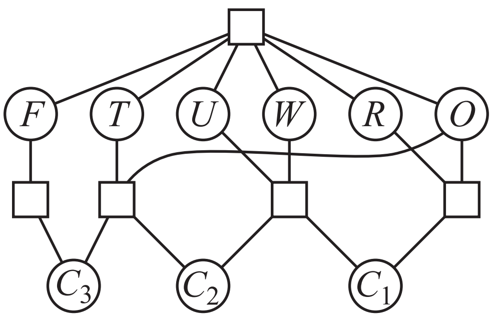
Problem 3.1.4: Taking inspiration from previous solutions, how can you generally turn a n-ary constraint into binary constraints?

Problem 3.2: Solving a CPS by hand performing backtracking search with minimum-remaining-values (MRV) and degree heuristics, least-constraining-value heuristics, and forward checking

(from *Russell & Norvig 3ed.* q. 7.5) Suppose that we have the cryptarithmetic problem as shown in Fig. 1.

$$\begin{array}{r} T\ W\ O \\ +\ T\ W\ O \\ \hline F\ O\ U\ R \end{array}$$

(a)



(b)

Figure 1: (a) A cryptarithmetic problem. Each letter stands for a distinct digit; the aim is to find a substitution of digits for letters such that the resulting sum is arithmetically correct. (b) The constraint hypergraph for the cryptarithmetic problem, showing the `Alldiff` constraint (square box at the top) as well as the column addition constraints (four square boxes in the middle). The variables $C1$, $C2$, and $C3$ represent the carry digits for the three columns.

[1] the symbol E is taken from German word *Einschränkung*.

We model the cryptarithmetic problem as $CSP = (X, D, C)$ with

$$\begin{aligned}
X &= \{F, T, U, W, R, O, C_1, C_2, C_3\} \\
D &= \{\text{numbers}, \dots, \text{numbers}, \text{binary}, \text{binary}, \text{binary}\} \\
C &= \{\langle (O, R, C_1), \quad O + O = R + 10 \cdot C_1 \rangle, \\
& \langle (U, W, C_1, C_2), \quad C_1 + W + W = U + 10 \cdot C_2 \rangle, \\
& \langle (O, T, C_2, C_3), \quad C_2 + T + T = O + 10 \cdot C_3 \rangle, \\
& \langle (C_3, F), \quad C_3 = F \rangle, \\
& \langle (F, T, U, W, R, O), \quad \texttt{Alldiff}(F, T, U, W, R, O) \rangle \},
\end{aligned}$$

where numbers $= \{0, 1, 2, \dots, 9\}$ and binary $= \{0, 1\}$.

Problem 3.2.1: Replace all boxes which correspond to higher-order constraints by binary constraints. Use the approach of Problem 3.1 and introduce variables such as X_1, X_2, and X_3, etc.

Problem 3.2.2: Sort the variables once by their domain size (i.e. number of remaining values) and once by their degree (i.e. number of constraints on other unassigned variables).

Problem 3.2.3: Perform backtracking search to solve the cryptarithmetic problem: Determine which variable to expand next by applying the minimum-remaining-values (MRV) heuristic; if there is a tie, use degree heuristics; if there is a tie again, choose one randomly. Use least-constraining-value heuristics to decide which value to assign. After each assignment, perform forward checking. Backtrack if you find an inconsistency.

2 Additional Problems

Problem 3.3: Modeling a CSP and solving it with SAVILEROW/MINION

(from *Russell & Norvig 3ed.* q. 7.7) Consider the following (Zebra) puzzle:

> In a row of five houses, each with different color, live five persons of different nationalities, each of whom prefers a different brand of candy, a different drink, and a different pet. Given the following facts, the questions to answer are "Where does the zebra live, and in which house do they drink Irn-Bru?"
>
> 1. The Englishman lives in the red house.
> 2. The Spaniard owns the dog.
> 3. The Norwegian lives in the first house on the left.
> 4. The green house is immediately to the right of the ivory house.
> 5. The man who eats Hershey bars lives in the house next to the man with the fox.
> 6. Kit Kats are eaten in the yellow house.
> 7. The Norwegian lives next to the blue house.
> 8. The Smarties eater owns a snail.
> 9. The Snickers eater drinks orange juice.
> 10. The Ukrainian drinks tea.
> 11. The Japanese person eats Milky Ways.
> 12. Kit Kats are eaten in a house next to the house where the horse is kept.
> 13. Coffee is drunk in the green house.
> 14. Milk is drunk in the middle house.

Problem 3.3.1: Model the zebra puzzle above as CSP. Discuss different representations this CSP could have.

Problem 3.3.2: Why would one prefer one representation over another?

Problem 3.3.3: Model the CSP with the SAVILEROW modeling language and obtain the solution using the MINION solver. Instruction for installing SAVILEROW and MINION are provided in Moodle. (Note that SAVILEROW and MINION are not relevant for the exam.)