Discrete Optimisation
Exercise Session 7: Constraint Programming

4th November 2016

After this session, you should be able to solve all exercises of Chapter 8 in the exercises book.

Exercise 1 (sudoku)**.** Sudoku is a kind of puzzle game. It involves an $N^2 \times N^2$ grid, some of them being filled. The goal is to fill all spaces with numbers between 1 and N^2 while meeting a series of constraints: "the same single integer may not appear twice in the same row, column or in any of the N^2 $N \times N$ subregions of the $N^2 \times N^2$ playing board" [Wikipedia].

1. Explain how a human could solve any sudoku puzzle using the same principles as constraint programming. Apply it on the given grid.
2. Write a mixed-integer programming model to solve sudoku puzzles.
3. * The solver performs many operations on the model before actually solving it (this is called *presolving*), and for the sudoku it might very well solve the problem (without needing any branch-and-bound or cutting plane). How does these operations relate to the way a human would solve the problem (1)?
4. Write a constraint programming model to solve sudoku puzzles. Use only the constraint `alldifferent`.

Exercise 2 (magic square)**.** "A magic square is an arrangement of distinct integers in an $N \times N$ grid, where the numbers in each row, and in each column, and the numbers in the main and secondary diagonals, all add up to the same number" [Wikipedia]. This number is given by the following formula, based on the size of the grid N:

$$\frac{N\left(N^2+1\right)}{2}.$$

Thus, for a 3×3 grid, the three digits in each row, column, and diagonal must sum up to 15; for a 4×4 grid, they must sum up to 34; etc.

Write a constraint programming model to solve magic squares. Use only the constraint `alldifferent` and linear equalities.

Exercise 3 (N queens)**.** The N-queens problem is about placing N queens in an $N \times N$ grid such that no two queens can threaten each other. The rules for queens are those of chess: a queen can attack another one if they are on the same row, column, or diagonal.

1. Write a constraint programming model to solve the n-queens problem. Use only the constraint `alldifferent`.
2. Solve the problem for $N = 4$ using reasoning (domain filtering based on the constraints) and enumeration. How many solutions exist?