Exercises for the Lecture
Practical SAT Solving

Summer Semester 2019

Assignment 1

Institute for Theoretical Informatics, Karlsruhe Institute of Technology (KIT)

Tomas Balyo, Carsten Sinz

25.04.2019

Due 09.05.2019

Exercise 1 (Coloring by Incremental SAT Challenge) [7(+7) points]

Implement an incremental SAT based graph vertex coloring solver in C or C++ using the IPASIR interface. Your application should take a single command line argument – a DIMACS file with a graph and find the smallest number of colors needed to color the graph. The application should print the number of colors required and the color of each vertex.

For the technical details of implementing an incremental solver based application follow the benchmark submission instructions of the 2016 SAT Competition Incremental Track: `http://baldur.iti.kit.edu/sat-competition-2016/index.php?cat=incremental`.

Test instances available at `http://mat.gsia.cmu.edu/COLOR/instances.html` (use the *.col instances, the *.col.b are in binary format). The best (fastest) solver gets 7 bonus points.

Exercise 2 (Sudoku Challenge) [7(+7) points] Write an encoder (in a programming language of your choice) for the generalized Sudoku puzzle. The generalized Sudoku puzzle of order n is an $n^2 \times n^2$ grid, consisting of n^2 sub-blocks of size $n \times n$, to be filled with numbers $1, \ldots, n^2$, such that

- in each row each number occurs exactly once,
- in each column each number occurs exactly once, and
- in each sub-block each number occurs exactly once.

The well-known Sudoku problem (see `https://en.wikipedia.org/wiki/Sudoku`) is the same as the generalized Sudoku puzzle of order 3. Benchmarks and input format description can be found here `https://baldur.iti.kit.edu/sat/files/sudokus.zip`. The best encoding (solving the most instances and fastest) will get a bonus of 7 points.

Exercise 3 (van der Waerden Numbers) [3 points]

Calcucate how many variables and clauses are in the SAT encoding presented in the lecture for checking whether $W(2,k) > n$?

Exercise 4 (Pythagorean Triples) [6 points] Find a solution (a pythagorean coloring) for the first 1000 numbers (1, 2, ..., 1000). Estimate the number of variables and clauses in the Pythagorean triples encoding from the lecture (as a function of n).