

Logical reasoning and programming, lab session 2

(September 26, 2022)

- 2.1** Use <http://fmv.jku.at/limboole/> on $\varphi = (a \rightarrow (c \wedge d)) \vee (b \rightarrow (c \wedge e))$.
- 2.2** Derive the empty clause from $\{\{\bar{a}, b\}, \{\bar{b}, c\}, \{a, \bar{c}\}, \{a, b, c\}, \{\bar{a}, \bar{b}, \bar{c}\}\}$ using resolution.
- 2.3** A clause $c_1 = \{l, l_1, \dots, l_n\}$ is blocked in φ by l if for every clause $c_2 \in \varphi$ such that $\bar{l} \in c_2$ the resolvent of c_1 and c_2 is a tautology. Prove that if a clause d is blocked (by a literal) in ψ , then $\psi \in \text{SAT}$ iff $\psi \setminus d \in \text{SAT}$.
- 2.4** Formulate graph coloring (a vertex coloring) as a SAT problem. Namely, given a graph G , does G admit a proper vertex coloring with k colors?
Discuss various possibilities how to formulate the problem. Moreover, are really all the constraints necessary?
- 2.5** Check this video, where zChaff colors the McGregor graph.
- 2.6** Let φ be a formula in CNF such that it contains only Horn clauses; they contain at most one positive literal. Show that SAT for φ is decidable in polynomial time.
Hint: Perform all the unit propagations first.
- 2.7** In fact, it is possible to improve the previous algorithm in such a way that it works in linear time.
- 2.8** Express the pigeonhole principle in propositional logic. Namely, define a propositional formula PHP_n^{n+1} , which says that you have $n + 1$ pigeons, n holes, every pigeon has a hole, and no two pigeons sit in the same hole.
- 2.9** Try PicoSAT/pycosat on PHP_n^{n+1} for small values of n . What is the maximal value of n for which you can solve PHP_n^{n+1} in one minute?
- 2.10** Install PySAT.
- 2.11** If you want to play with SAT solving a bit, then a standard exercise is to formalize Sudoku as a SAT problem and hence produce a Sudoku solver. Write a program that generates a problem specification in the DIMACS format in such a way that it is possible to specify an input (a partially completed grid) by appending¹ clauses saying which variables are true. You can use MiniSat or pycosat and some input is available from here, where each line has format XYZ meaning there is Z in cell (X, Y) .
By the way, is it possible to obtain also a generator of Sudoku puzzles this way?
- 2.12** Check the details of two watched literals, for example, these and these slides illustrate the data structure.

¹Note that this changes the number of clauses, a parameter specified in the DIMACS format.