Satisfiability Checking - WS 2016/2017

Series 7

gereon.kremer@cs.rwth-aachen.de
https://ths.rwth-aachen.de/teaching/

Exercise 1

Consider the propositional logical formula with equalities:

$$\begin{array}{rcrclclcl} \varphi^{EQ} & := & & x_3 = x_5 & \wedge & (\neg x_1 = x_4 & \vee & \neg x_4 = x_5) \\ & & \wedge & x_4 = x_6 & \wedge & (x_4 = x_5 & \vee & x_3 = x_4) \\ & & \wedge & x_1 = x_2 & \wedge & (x_4 = x_5 & \vee & x_3 = x_6) \end{array}$$

The Boolean abstraction of this formula is

$$a_1 \wedge (\neg a_2 \vee \neg a_3) \wedge a_4 \wedge (a_3 \vee a_5) \wedge a_6 \wedge (a_3 \vee a_7).$$

Simulate how a less-lazy SMT solver solves φ^{EQ} for satisfiability as presented in the lecture. Show the progress in the SAT solver and the theory solver implementing an incremental and infeasible subset generating procedure for solving a conjunction of equalities for satisfiability. If the SAT solver makes a decision, it chooses the unassigned variable a_i with the lowest index and assigns it to false. Show how the theory solver benefits from its incrementality support, both when adding and removing constraints, and show how the infeasible subset(s) are computed.