

There are two kinds of problems: problems marked “T” are theoretical problems, and problems marked “E” involve coding and experimentation.

Problem T1.1. (a) Modify the enumerative reachability algorithm discussed in class so that it produces a trajectory from the initial state to a target state in case the target set is reachable.

(b) How will you modify a reachability algorithm using depth first search that uses hashing to store states to return a counterexample trace? Would your technique work if you use breadth-first search?

Problem T1.2. Let C_1 and C_2 be two combinational circuits (i.e., circuits that use AND, OR, and NOT gates) on n inputs. You are interested in checking if these circuits are equivalent, that is, they produce the same output on all inputs. Show how you can reduce the equivalence checking problem to a query to SAT. Remember that the SAT solver accepts formulas in conjunctive normal form.

Problem T1.3. Show how you can produce a resolution proof of unsatisfiability when a SAT solver stops with failure (that is, the input formula is unsatisfiable). (This is discussed in the lecture notes by Sharad Malik on the course home page.)

Problem E1.4. This assignment will give you some experience using SAT solvers. Download the MiniSAT satisfiability solver and look at the code. Write a Sudoku solver that reduces Sudoku to a SAT problem, and solves the SAT problem using MiniSAT. You can get Sudoku instances from <http://www.websudoku.com/>. The problems are classified as “Easy”, “Medium”, or “Hard”. When you run MiniSAT on these problems, is there a trend based on number of conflicts or backtracks that support this classification?

Can you use a SAT solver to *generate* a Sudoku puzzle? Remember that a Sudoku puzzle should have a unique solution.

[Paper Reading] Read the following paper on enumerative model checking of C code.

Model Checking for Programming Languages using Verisoft. Patrice Godefroid. POPL 1997: 174-186.

(A Google search should find it.) Write a one-paragraph summary for the paper, pointing out the main idea and at least one direction of future work not explicitly mentioned in the paper.

[Paper Reading] Read the notes on SAT solving by Sharad Malik on the course page.