Project Proposal:
Parallel SAT Solver

Team

Wonhyuk Choi (uni: **wc2737**) & Andrew de Soler (uni: **apd2149**)

Abstract

Our final project is to create a **parallel SAT solver**, which will solve boolean satisfiability problems through static partitioning. The final product will be an executable that spawns parallel threads to solve different boolean satisfiability problems in a SIMD manner.

SAT Solver

We will first implement a sequential SAT solver in Haskell. We will first try to implement a simple brute-force backtracking algorithm, and if time permits, optimize it to use the Davis–Putnam–Logemann–Loveland (DPLL) algorithm for SAT solvers. The SATLIB library has a large amount of boolean examples that we can test our code on.

Parallelization

We will use static partitioning to parallelize our computations, much like Simon Marlow's sudoku problems presented in class. We will explore different strategies, experiment with the number of partitions, consider empirical results based on the number of cores, and other modifications that try to speed up solving of many boolean satisfiability problems.