

Notes on SAT Solving

William Schultz

April 11, 2022

1 DPLL

The original DPLL algorithm views satisfiability as a search problem over the tree of possible boolean assignments, and combines this with *conditioning* and *unit propagation* techniques. In its most basic form, given a CNF formula Δ , we start by assigning some variable a value of *true* or *false*, and then simplify the overall formula. We say that Δ is *conditioned* on a literal L , notated as $\Delta|L$, if we replace the value of literal L with *true*, and the value of literal $\neg L$ with *false*.

After assigning a literal with a concrete value, we simplify the overall formula. If we simplify to something that is satisfied, then we're done, and can return a satisfying assignment. If we ever reach a state where all clauses cannot be satisfied, then we return unsatisfiable.