The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual
Chapter 8, Problem 8.1

We prove a slightly stronger result, namely that we can solve in polynomial time any instance of *Satisfiability* in which every variable appears at most twice. If a variable appears only once or appears twice as the same literal, then we simply set the literal to true, then remove all satisfied clauses and repeat the process. Thus the only problem of interest arises when every variable appears once uncomplemented and once complemented. But we have seen (in our proof that *3SAT* is NP-complete) that two clauses that share a variable, with one clause including the uncomplemented literal and the other the complemented literal, are logically linked into a single clause. Thus we can process each variable in turn, linking the two clauses in which it appears and eliminating that variable. If the resulting clause contains both a literal and its complement, it is trivially satisfiable—we can eliminate it and process the clauses left over in the fashion just described; otherwise we just replace the two clauses just processed by the new joined clause and repeat our step. The process stops when all clauses are eliminated (success) or when we have created two single-literal clauses for the same variable, one complemented and the other uncomplemented (failure). Clearly, the whole process takes at most quadratic time (a linear number of variables to be processed, each taking up to linear time).