The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual
Chapter 7, Problem 7.11

The version of *Minimal Unsatisfiability* defined for this problem is thus a version of *Satisfiability* where we ask whether or not the instance can be made satisfiable by removing any one of its clauses (not knowing whether or not the original instance is itself satisfiable).

The interesting part of the problem here is membership in NP. We need to exhibit a certificate for every "yes" instance. If an instance has n variables and m clauses, we can devise a certificate that will have m (possibly distinct) truth assignments, one for each clause that can be removed; such a certificate has length $\Theta(mn)$ and can be checked in polynomial time by checking each truth assignment in turn, ignoring for each its corresponding clause. Thus the problem has concise certificates that can be checked in polynomial time and is in NP.

To show that it is NP-complete, we use a very simple transformation from *Satisfiability*: we simply double the number of variables and write every clause "twice"—once with the first set of variables and once with the second set. If the original instance was satisfiable, the constructed instance is satisfiable as well—there is no need to remove any clause to satisfy it, but it never hurts to remove one. If the original instance is unsatisfiable, then no single clause can be removed to make the constructed instance satisfiable—by removing a single clause we might make one copy of the original instance satisfiable, but the other copy remains unchanged and thus unsatisfiable. That the transformation runs in polynomial time is obvious.