Solving 2SAT

A 2SAT instance is an instance of Satisfiability in which each clause contains two literals. We present a polynomial time algorithm which reduces a 2SAT instance to the problem of finding the strong components of a digraph (see page 1082 of Cormen for a definition).

If the 2SAT instance I is over variables $x_1, .., x_n$ then the graph G_I we consider has vertex set $\{x_1, .., x_n, \overline{x_1}, ..., \overline{x_n}\}$. Each clause A *or* B corresponds to two edges of G_I, one from $\overline{A}$ to B,the other from $\overline{B}$ to A. Thus if I has m clauses there are exactly $2m$ edges in G_I.

Note that if AB is an edge then in any satisfying truth assignment for which A is true, B must also be true (since the clause corresponding to the edge must be satisfied). It follows by induction that if there is a (directed) path P from A to B then in any satisfying truth assignment if A is true then so is B. Now, we cannot set both x_i and $\overline{x_i}$ to true, in a satisfying truth assignment. Thus, if there is a directed path from x_i to $\overline{x_i}$, x_i must be false in any satisfying truth assignment. Similarily, if there is a directed path from $\overline{x_i}$ to x_i then x_i must be true in any satisfying truth assignment. Thus, if for some i, x_i and $\overline{x_i}$ are in the same strong component then there is no satisfying truth assignment.

We now show that if there is no such i, then there is a satisfying truth assignment. To begin, we note that if AB is an edge of G_I then so is $\overline{BA}$. Applying this fact to all the edges of a path, we see that there is a path from A to B in G_I precisely if there is a path from $\overline{B}$ to $\overline{A}$ in G_I. Symmetrically, there is a path from B to A in G_I precisely if there is a path from $\overline{A}$ to $\overline{B}$ in G_I. Thus, A and B are in the same strong component of G_I, precisely if $\overline{A}$ and $\overline{B}$ are.

So, assuming there is no strong component containing both a variable and its negation, we can pair the strong components as $U, \overline{U}$ such that the literals appearing in $\overline{U}$ are precisely the negations of the literals appearing in U.

We take a topological sort of the acyclical component graph of G (see Figure 22.9 of Cormen), and set the variables in U to be true if U appears after $\overline{U}$ in this order and false otherwise. To show this assigment is satisfying it is enough to show that there is no edge AB with A true and B false (since the edges corresponding to an unsatisfied clause have this form). Assume the contrary and let U be the first component of the strong component graph which contains an endpoint of such an edge. Let vw be an edge with v true, w false and one of v or w in U. Note that one of v or w is not in U, because all of the literals in U received the same truth value. By our choice of U, the vertex in the edge which is not in U is in a component appearing after U in the topological sort and hence $v \in U$. Since v is true, this means that $\overline{U}$ appears before U in our topological sort. But $\overline{wu}$ is also an edge from a true vertex to a false vertex. This edge contains a vertex of $\overline{U}$, contradicting our choice of U.

So, to solve 2SAT, we need only find the strong components of a digraph. In the next class, we present a linear time algorithm for this problem.