3SAT

Andreas Klappenecker [partially based on slides by Jennifer Welch]



3SAT

Given a boolean function in conjunctive normal form such that every clause contains exactly three literals, decide whether the formula is satisfiable.

[This a special case of SAT]

Proving NP-Completeness

How do you prove that a decision problem L is NP-complete? (1) Show that L is in NP. (2.a) Choose an appropriate known NP-complete language L'. (2.b) Show L' ≤_p L

Proof Strategy

(1) 3SAT is in NP, since we can check in polynomial time whether a given truth assignment evaluates to true. (2.a) Choose SAT as a known NP-complete problem. (2.b) Describe a reduction from SAT inputs to 3SAT inputs computable in polynomial time SAT input is satisfiable iff constructed 3SAT input is satisfiable

General Idea of the Reduction

We're given an arbitrary CNF formula $C = c_1 \wedge c_2 \wedge ... \wedge c_m$ over set of variables, where each c_i is a clause (a disjunction of literals).

We will replace each clause c_i with a conjunction of clauses c_i' , and may use some extra variables. Each clause in c_i' will have exactly 3 literals. The transformed input will be conjunction of all the clauses in all the c_i' .

Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 1: k = 1. Use extra variables y_i^1 and y_i^2 . Replace c_i with 4 clauses: $(z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^2) \wedge (z_1 \vee y_i^2)$

Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 2: k = 2. Use extra variable y_i^1 . Replace c_i with 2 clauses: $(z_1 \vee z_2 \vee \neg y_i^1) \wedge (z_1 \vee z_2 \vee y_i^1).$



Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 3: k = 3. No extra variables are needed. Keep c_i : $(z_1 \vee z_2 \vee z_3)$



Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 4: k > 3. Use extra variables y_i^1 , ..., y_i^{k-3} . Replace c_i with k-2 clauses: $(\mathbf{Z}_1 \vee \mathbf{Z}_2 \vee \mathbf{y}_1)$ Text $\wedge (\neg y_i^1 \vee z_3 \vee y_i^2) \wedge (\neg y_i^2 \vee z_4 \vee y_i^3) \wedge \dots$ $\wedge (\neg y_i^{k-5} \vee z_{k-3} \vee y_i^{k-4}) \wedge (\neg y_i^{k-4} \vee z_{k-2} \vee y_i^{k-3})$ $\wedge (\neg y_i^{k-3} \vee z_{k-1} \vee z_k)$



Polynomial Time Reduction

Each new formula is at most a constant times larger than the original formula, and the translation is straightforward. Therefore, the reduction is polynomial time.

Correctness of the Reduction

Show that CNF formula C is satisfiable iff the 3-CNF formula C' constructed is satisfiable.

=>: Suppose that C is satisfiable. We need to construct a satisfying truth assignment for C'.

For variables in C' that are already in C, we use same truth assignments as for C.

How should we assign T/F to the new variables?

Truth Assignment for New Variables

Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 1: k = 1. Use extra variables y_i^1 and y_i^2 . Replace c_i with 4 clauses: $(z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^2) \wedge (z_1 \vee y_i^2)$

Assign y_i 's with arbitrary values, as z_1 is true

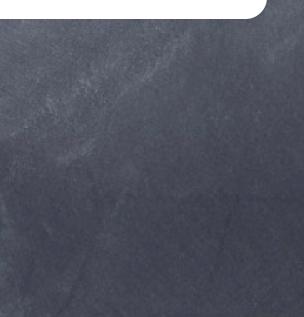
Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 2: k = 2. Use extra variable y_i^1 . Replace c_i with 2 clauses:

 $(z_1 \vee z_2 \vee \neg y_i^1) \wedge (z_1 \vee z_2 \vee y_i^1).$

Assign y_i 's with arbitrary values, as $z_1 \vee z_2$ is true





Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 3: k = 3. No extra variables are needed. Keep c_i : $(z_1 \vee z_2 \vee z_3)$

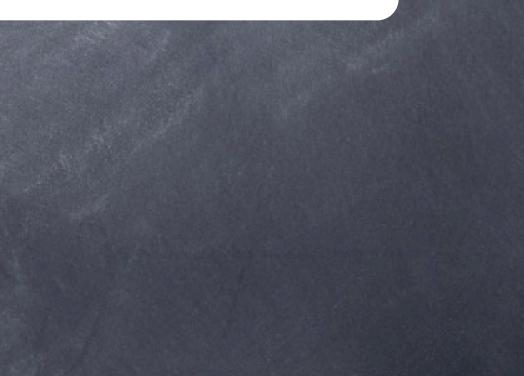


Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 4: k > 3. Use extra variables y_i^1 , ..., y_i^{k-3} . Replace c_i with k-2 clauses:

 $(\mathbf{Z}_1 \vee \mathbf{Z}_2 \vee \mathbf{y}_1)$ $\wedge (\neg y_i^1 \vee Z_3 \vee y_i^2) \wedge (\neg y_i^2 \vee Z_4 \vee y_i^3) \wedge \dots$ $\wedge (\neg y_i^{k-5} \vee z_{k-3} \vee y_i^{k-4}) \wedge (\neg y_i^{k-4} \vee z_{k-2} \vee y_i^{k-3})$ $\wedge (\neg y_i^{k-3} \vee z_{k-1} \vee z_k)$

If z_1 or z_2 is true, set all y_i 's to false, so all later clauses have a true literal.

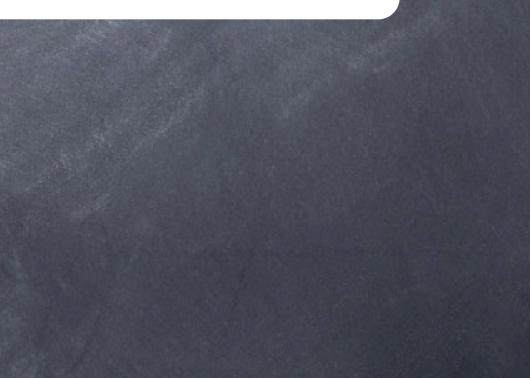


Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 4: k > 3. Use extra variables y_i^1 , ..., y_i^{k-3} . Replace c_i with k-2 clauses:

 $(\mathbf{Z}_1 \vee \mathbf{Z}_2 \vee \mathbf{y}_1)$ $\wedge (\neg y_i^1 \vee Z_3 \vee y_i^2) \wedge (\neg y_i^2 \vee Z_4 \vee y_i^3) \wedge \dots$ $\wedge (\neg y_i^{k-5} \vee z_{k-3} \vee y_i^{k-4}) \wedge (\neg y_i^{k-4} \vee z_{k-2} \vee y_i^{k-3})$ $\wedge (\neg y_i^{k-3} \vee z_{k-1} \vee z_k)$

If z_{k-1} or z_k is the first true literal of c_i , set all y_i 's to true, so all earlier clauses have a true literal.

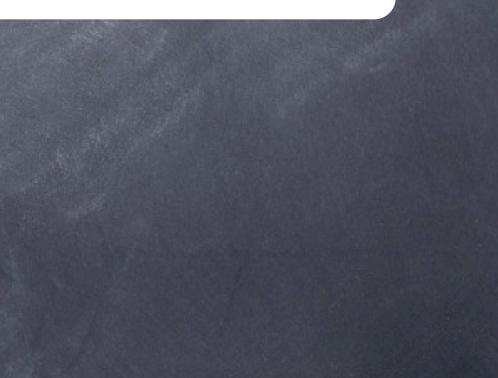


Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 4: k > 3. Use extra variables y_i^1 , ..., y_i^{k-3} . Replace c_i with k-2 clauses:

 $(Z_1 \vee Z_2 \vee Y_1)$ later y_i 's to false. $\wedge (\neg y_i^1 \vee z_3 \vee y_i^2) \wedge (\neg y_i^2 \vee z_4 \vee y_i^3) \wedge \dots$ $\wedge (\neg y_i^{k-5} \vee Z_{k-3} \vee y_i^{k-4}) \wedge (\neg y_i^{k-4} \vee Z_{k-2} \vee y_i^{k-3})$ $\wedge (\neg y_i^{k-3} \vee z_{k-1} \vee z_k)$

If first true literal is in between, set all earlier y_i 's to true and all



Correctness of Reduction

<=: Suppose the newly constructed 3SAT formula C' is satisfiable. We must show that the original SAT formula C is also satisfiable.

Use the same satisfying truth assignment for C as for C' (ignoring new variables).

Show each original clause has at least one true literal in it.

Original Clause is True

Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 1: k = 1. Use extra variables y_i^1 and y_i^2 . Replace c_i with 4 clauses: $c_i = (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^1 \vee y_i^2) \wedge (z_1 \vee y_i^2) \wedge (z_1 \vee y_i^2)$

> If c_i is true, then $c_i = z_1$ must be true, since one pair of literals in y_i^1 and y_i^2 must be true

Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 2: k = 2. Use extra variable y_i^1 . Replace c_i with 2 clauses: $C_i = (z_1 \vee z_2 \vee \neg y_i^1) \wedge (z_1 \vee z_2 \vee y_i^1).$

If c_i' is true, then $c_i = z_1 \vee z_2$ must be true





Let $c_i = z_1 \vee z_2 \vee \ldots \vee z_k$

Case 3: k = 3. No extra variables are needed. Keep c_i : $(z_1 \vee z_2 \vee z_3)$



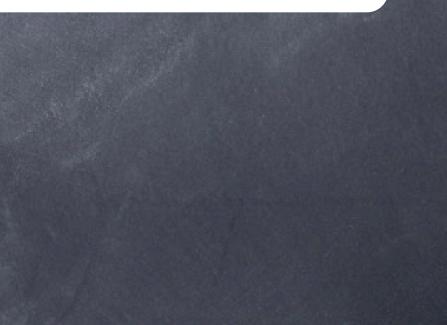
Let $C_i = Z_1 \vee Z_2 \vee ... \vee Z_k$

Case 4: k > 3. Use extra variables y_i^1 , ..., y_i^{k-3} . Replace c_i with k-2 clauses:

 $(\mathbf{Z}_1 \vee \mathbf{Z}_2 \vee \mathbf{y}^1)$ $\wedge (\neg y_i^1 \vee z_3 \vee y_i^2) \wedge (\neg y_i^2 \vee z_4 \vee y_i^3) \wedge \dots$ last clause in c_i must be false, $\wedge (\neg y_i^{k-5} \vee z_{k-3} \vee y_i^{k-4}) \wedge (\neg y_i^{k-4} \vee z_{k-2} \vee y_i^{k-3})$ $\wedge (\neg y_i^{k-3} \vee z_{k-1} \vee z_k)$

contradiction.

Suppose that there is a valuation such that c_i is true and c_i is false. Then y_i^k must be true for all k, so the



Conclusions

We have shown that SAT is in NP There exists a polynomial time reduction from SAT to 3SAT. Therefore, 3SAT is NP-complete.