Lots of NP-complete Problems

Recall (Cook-Levin): SAT is NP-complete, and this is also true for $CNFSAT$, the class of expressions in CNF form.

Starting from Cook-Levin, we want to show that other problems are NP-complete.

To start, we will consider CNF formulas where each clause has exactly 3 literals (3-CNF).

Note: While encodings are still very important (and we need to specify what our encodings will be,) we are going to stop using angle brackets, and just identify instances with their encodings. This will simplify the presentation.

$3SAT$ is NP-complete

$3SAT = \{\phi \mid \phi$ is in 3-CNF form and is satisfiable$\}$

We could encode instances using DIMACS format. For convenience, we will use a more readable encoding, e.g., a *list of variables* $X = x_1, x_2, \ldots, x_m$, followed by a *list of clauses* $C_1, C_2, \ldots, C_k$ (each clause is a list of 3 literals, e.g., $\{x_1, \overline{x}_5, x_8\}$)

Claim: $3SAT$ is in NP.

Proof: Since SAT is in NP so is $3SAT$, since it is a restriction of SAT.

Alternate Proof that $3SAT \in \text{NP}$

Alternatively, guess a satisfying truth assignment $t : X \to \{0, 1\}$, encoded as, e.g., $x_1 = 0, x_2 = 1, \ldots, x_m = 1$

To verify that the input is in $3SAT$ format, for each clause look up the values of the 3 literals in the clause to see if at least one of their values is 1. This takes $O(mk \log k)$ time on a TM.

The length of the input is $O((m + k) \log k)$ so $O(mk \log k)$ is polynomial in the length of the input. Thus, there is a polynomially bounded nondeterministic TM which accepts $3SAT$.

SAT Reduces to $3SAT$

$\underline{\phi}$	$\underline{f(\phi)}$
(a) $X = \{x_1, x_2, ...x_m\}$	$X \cup$ {extra variables}
(b) Clause with 3 literals	Same clause
(c) Clause with 2 literals, e.g., $C_i = \{a, b\}$	Add new variable p_i Create two clauses, What are they?
(d) Clause with 1 literal, e.g. $C_i = \{a\}$	Add two variables p_i, q_i Create 4 clauses, What are they?

SAT Reduces to $3SAT$

(e) Clause with $r > 3$ literals,
e.g., $C_i = \{z_1, z_2, ...z_r\}$

Add $r - 3$ new variables
and $r - 2$ new clauses,
$\{z_1, z_2, y_1\}$,
$\{z_3, \overline{y}_1, y_2\}$,
$\{z_4, \overline{y}_2, y_3\}$,
$...$,
$\{z_{r-2}, \overline{y}_{r-4}, y_{r-3}\}$
$\{z_{r-1}, z_r, \overline{y}_{r-3}\}$

Reduction (cont'd)

The 3-SAT instance has length $O((m^2 + k)\log k)$ and can be constructed in $(m+k)^{O(1)}$ time.

Claim: If $\phi \in SAT$ then $f(\phi) \in 3SAT$.

Proof: Suppose there is a truth assignment T for ϕ. We show there is a truth assignment T' which satisfies each clause of $f(\phi)$. Let $T' = T$ for $x \in X$.

Reduction (cont'd)

We much consider cases (b)-(e) in the definition.

(b) Since $T = T'$ for variables in these clauses, then these clauses are satisfied.

(c) Set $T'(p_i) = 1$.

(d) Set $T'(p_i), T'(q_i) = 1$.

(e) For some j, z_j is 1 under T. Set $T'(y_i) = 0$ for $i \geq j$; $T'(y_i) = 1$ for $i < j$. Then every clause has a true literal.

The Other Direction

Assuming $f(\phi) \in 3SAT$ with satisfying assignment T', we show that $T = T'$ restricted to X is a satisfying assigment for ϕ:

(b) trivial

(c) If $T'(p_i) = 1$, then $T'(\overline{p_i}) = 0$ and vice versa, so it must be the case that one of a or b is set to 1 by T', and $\{a, b\}$ is satisfied.

(d) For any T', one setting of the extra variables will make two corresponding literals false, so $T'(a) = 1$.

(e) Proof by contradiction: Suppose all literals in C_i are false under T'. Then y_1 is true. $T'(y_1) = 1 \rightarrow T'(y_2) = 1 \rightarrow ... \rightarrow T'(y_{r-3}) = 1$. But then z_{r-1} or z_r is true, giving a contradiction.

CLIQUE is NP-complete

$CLIQUE = \{(G, J) \mid G \text{ is a graph with a clique of size at least } J\}$

Let $G = (V, E)$. A *clique* is a subset V' of V such that every two nodes in V' are joined by an edge in E. The *size* of the clique is the number of nodes it contains.

Encoding for $CLIQUE$: nodes V, list of edges E, integer J

Clique is in NP

Let the TM guess the set of nodes in a clique It can verify that every edge between them is in E: For each pair of nodes, check list of edges. And we can check that the size of the clique is at least J. There are less than $|E|$ edges between the nodes in the clique. The cost of checking each is no more than $O(|E|)$, for a total of $O(|E|^2)$. The cost of counting the size of the clique is no more than $O(|V|)$. Hence the cost of verifying the guess is polynomial in the length of the instance.

$3SAT$ Reduces to $CLIQUE$

$3SAT$	$CLIQUE$
	J is the number of clauses.
For each literal in each clause e.g. $C_i = \{a_{i1}, a_{i2}, a_{i3}\}$	Create nodes a_{i1}, a_{i2}, a_{i3}
For each a_{ij} and $a_{i'j'}$ such that $i \neq i'$ and $a_{ij} \neq \overline{a}_{i'j'}$	Add edge $\{a_{ij}, a_{i'j'}\}$

Correctness of the Reduction

1. The function f from $3SAT$ to $CLIQUE$ is poly time computable.

2. If $\phi \in 3SAT$ then $f(\phi) \in CLIQUE.$
   Proof: Given a satisfying truth assignment T, select one node corresponding to a true literal from each clause. There is an edge between every pair of these, which implies a clique of size equal to the number of clauses.

3. If $f(\phi) \in CLIQUE$ then $\phi \in 3SAT$.
   Proof: Suppose there is a clique of size J. Then set those literals to TRUE. Since there is an edge between each pair, they are mutually satisfiable and they come from different triples. Hence there is a truth assignment with one true literal in each clause. Extend it to cover the remaining variables in any consistent way.

Vertex Cover is NP-complete

$VC = \{(G, K) \mid G \text{ has a vertex cover of size } \leq K\}$.

A *vertex cover* is a set of nodes such that every edge has at least one endpoint in the set.

A reasonable encoding of an instance:
list of nodes
list of edges
integer K

Vertex Cover is in NP

- Guess for VC: a list of vertices in a cover.

- For each edge, see if one of its endpoints is on the list. This takes $O(|E||V|\log|V|)$ time – polynomial in size of the input

$CLIQUE$ Reduces to VC

$CLIQUE$	VC
$G = (V, E)$	$G' = (V,\ \text{complement of } E)$
J	$K = \lvert V \rvert - J$

Correctness of the Reduction

1. An instance (G, J) of $CLIQUE$ has length $O((|V| + |E| + 1) \log |V|) = O(|V|^3)$. The instance of VC has length $O((|V| + |V|^2 - |E| + 1) \log |V|) = O((|V|^2 \log |V|) = O(|V|^3)$ and can be constructed in time $O(|V|^3)$.

2. If $(G, J) \in CLIQUE$ then $f((G, J)) \in VC$:

   Let C be the set of nodes in a clique of size J. There are no edges in G' between any pair of nodes in C, so every edge in G' is incident to some node in $V - C$, a set of size $|V| - J$. Hence there is a vertex cover of size $|V| - J$.

Correctness (continued)

3. If $f((G, J)) \in VC$, then $(G, J) \in CLIQUE$:

   If there's a vertex cover C' in G' then let a, b be any pair of nodes in $V - C'$. Since neither a or b are in C', there's no edge between them in G', which implies that there is an edge between them in G. I.e., the nodes in $V - C'$ form a clique in G. Hence there is a clique of size $|V| - J = |V| - (|V| - K) = K$ in G.

INDEPENDENT SET is NP-complete

$INDEPENDENT\ SET = \{(G, K) \mid G$ has an independent set of size $\geq K\}$.

An *independent set* is e a subset C with $|C| \geq K$ such that for all $v_i, v_j \in C$, there is no edge between v_i and v_j.

- Describe an encoding scheme for INDEPENDENT SET.
- Prove INDEPENDENT SET is in NP.
- Prove INDEPENDENT SET is NP-complete.