Efficient Computations

$\mathbf{P} = \{L \mid$ some poly time bounded deterministic Turing machine accepts $L\}$.

$\mathbf{NP} = \{L \mid$ some poly time bounded nondeterministic Turing machine accepts $L\}$.

coNP=$\{L \mid \overline{L} \in \mathbf{NP}\}$.

P=NP?

NP

Proposition: Suppose $L \in \mathbf{NP}$.
Then there exists a (deterministic) polynomial time computable predicate $P(x, y)$, and a polynomial $q(\cdot)$ such that
$x \in L$ iff $(\exists y \mid |y| \leq q(|x|))[P(x, y)]$.

Proof: Suppose N is a $q(n)$ time bounded NDTM accepting L.
Without loss of generality assume that N has exactly two choices in each state.

$P(x, y)$ is defined as follows.

Let $y = y_1 y_2 \cdots y_m$.

If $m > q(|x|)$ then reject.

Otherwise simulate N, where at step i, choose the next state based on whether y_i is 0 or 1.

$P(x, y)$ is 1 iff N accepts in the above simulation.

Now, $(\exists y \mid |y| \leq q(|x|))[P(x, y)]$ iff $N(x)$ has an accepting path.

In the proposition one often calls y such that $P(x, y) = 1$ as a "certificate" or "proof" that $x \in L$.
Thus one can consider $\mathbf{NP}$ as class of languages for which "proofs" can be easily (in polynomial time) verified.

Reducibility

$L_1 \leq_m^p L_2$ (read: L_1 is poly time, many-one, reducible to L_2): there exists poly time computable function f such that $x \in L_1 \Leftrightarrow f(x) \in L_2$.

$L_1 \leq_T^p L_2$ (read: L_1 is poly time, Turing, reducible to L_2): there exists a polynomial time oracle Turing machine M, such that the M^{L_2} accepts L_1.

$L_1 \leq_m^{\log \text{space}} L_2$ (read: L_1 is log-space many-one reducible to L_2):
there exists a function f, which is computable by a $\log$ space bounded Turing machine, such that $x \in L_1 \Leftrightarrow f(x) \in L_2$.

NP-completeness

A set L is said to be **NP**-complete iff
(1) $L \in$ **NP**, and
(2) $(\forall L' \in$**NP**$)[L' \leq_m^p L]$.

If (2) is satisfied, then the problem is said to be **NP**-hard. The interest in **NP**-complete problems arises from the fact that many of the interesting combinatorial problems are **NP**-complete.

Proposition: $\leq_m^p$ is reflexive and transitive.
Proof:
Reflexive: Any L can be reduced to itself by identity function $f(x) = x$.

Transitive: Suppose $L_1 \leq_m^p L_2$ and $L_2 \leq_m^p L_3$.
Suppose f, g are polynomial time computable functions such that
$x \in L_1 \Leftrightarrow f(x) \in L_2$ and $x \in L_2 \Leftrightarrow g(x) \in L_3$.
Let $h(x) = g(f(x))$. Clearly h is polynomial time computable.
Now $x \in L_1 \Leftrightarrow f(x) \in L_2 \Leftrightarrow g(f(x)) \in L_3$.
Thus $x \in L_1 \Leftrightarrow h(x) \in L_3$.
Thus $L_1 \leq_m^p L_3$. This shows that $\leq_m^p$ is transitive.

Corollary: If L is **NP**-complete, $L' \in \mathbf{NP}$ and $L \leq_m^p L'$ then L' is **NP**-complete.
The above corollary allows us to prove that a problem $L' \in \mathbf{NP}$ is **NP**-complete by just showing that $L' \in \mathbf{NP}$ and some KNOWN **NP**-complete problem is polynomial time, many one reducible to L'.

Graph: $G = (V, E)$. V is a set of vertices/nodes. $E \subseteq V \times V$ is a set of edges.

Directed graph: Edge $(u, v) \in E$, is directed from u to v.
Undirected graph: Edge $(u, v) \in E$ is undirected. That is, if $(u, v) \in E$, then $(v, u) \in E$. The set of edges is symmetric.

Cycles: $v_1, v_2, \ldots, v_k, v_1$ such that (v_i, v_{i+1}), for $1 \leq i < k$ and (v_k, v_1) are (directed) edges in the graph. Here we assume that the edges used, (v_i, v_{i+1}), for $1 \leq i < k$ and (v_k, v_1) are all pairwise distinct.
Acyclic: There are no sequence of vertices $v_1, v_2, \ldots, v_k$ such that (v_i, v_{i+1}), for $1 \leq i < k$, and (v_k, v_1) are (directed) edges in the graph (where the edges used, (v_i, v_{i+1}), for $1 \leq i < k$ and (v_k, v_1), are all pairwise distinct).
Child, Parent: For directed graph, $(u, v) \in E$, then v is child of u, and u is parent of v.

Some famous NP complete problems

1. Satisfiability:
INSTANCE: A set U of variables and a collection C of clauses over U.
QUESTION: Is there a satisfying truth assignment for C?
Here, a clause is of the form $(A \vee B \vee \neg C)$.
Thus, satisfiability problem is of the form
$(A \vee B \vee \neg C) \wedge (E \vee F \vee \neg A) \wedge (F \vee B \vee \neg C) \ldots$.
$A, \neg A, B, \neg B \ldots$ are called literals.

3-SAT: Each clause has at most (exactly) 3 literals.

2. 3-Dimensional Matching:
INSTANCE: Three disjoint finite sets X, Y, Z, each of cardinality n, and a set $S \subseteq X \times Y \times Z$.
QUESTION: Does S contain a matching? i.e. is there a subset $S' \subseteq S$ such that $|S'| = n$ and no two elements of S' agree in any coordinate?

3. Vertex Cover:
INSTANCE: A graph $G = (V, E)$ and a positive integer $K \leq |V|$.
QUESTION: Is there a vertex cover of size K or less for G?
i.e. is there a subset $V' \subseteq V$ such that, $|V'| \leq K$ and for each edge $(u, v) \in E$, at least one of u, v belongs to V'?

4. MAX-CUT:
INSTANCE: An undirected graph $G = (V, E)$, and a positive integer $K \leq |E|$.
QUESTION: Is there a cut of G with size $> K$? Here (X, Y) is said to be a cut of G, if (X, Y) is a partition of V. That is, $X \cap Y = \emptyset$ and $X \cup Y = V$. Size of a cut (X, Y) of G, is $|\{(v, w) \mid v \in X$ and $w \in Y$ and $(v, w) \in E\}|$. That is, size of a cut (X, Y) is the number of edges in G which connect X and Y.

5. Clique:
INSTANCE: A graph $G = (V, E)$ and a positive integer $K \leq |V|$.
QUESTION: Does G contain a clique of size K or more? i.e. is there a subset $V' \subseteq V$, such that $|V'| \geq K$, and for all distinct $u, v \in V'$, $(u, v) \in E$?

6. Independent Set:
INSTANCE: A graph $G = (V, E)$ and a positive integer $K \leq |V|$.
QUESTION: Does G contain an independent set of size K or more? i.e. is there a subset $V' \subseteq V$, such that $|V'| \geq K$, and for all distinct $u, v \in V'$, $(u, v) \notin E$?

7. Hamiltonian Circuit:
INSTANCE: A graph $G = (V, E)$
QUESTION: Does G contain a Hamiltonian circuit? i.e. is there a simple circuit which goes through all the vertices of G?

8. Partition:
INSTANCE: A finite set A and a size $s(a) > 0$, for each $a \in A$.
QUESTION: Is there a subset A' of A such that $\sum_{a \in A'} s(a) = \sum_{a \in A - A'} s(a)$?
Note: Here $s(a)$ is given in binary for each $a \in A$. So the length of the input is proportional to $|A| + \sum_{a \in A} \log s(a)$.

9. Set Cover:
INSTANCE: A finite set A, a collection $\{S_1, S_2, \ldots, S_m\}$ of subsets of A, and a number k.
QUESTION: Is there a subset Y of $\{1, \ldots, m\}$, of size at most k, such that $A \subseteq \bigcup_{i \in Y} S_i$.

10. Traveling Salesman Problem:
INSTANCE: A complete weighted graph $G = (V, E)$, and a bound B.
QUESTION: Is there a Hamiltonian circuit of weight $\leq B$?
Note: Here weights of the edges and B are given in binary. So the length of the input is proportional to $|V| + |E| + \log B + \sum_{e \in E} \log wt(e)$.

Theorem: (a) If one of the **NP**-complete problems is solvable in polynomial time, then all the problems in **NP** are solvable in polynomial time. In other words, $\mathbf{P} = \mathbf{NP}$.

(b) If $\mathbf{P} \neq \mathbf{NP}$, then none of the $\mathbf{NP}$-complete problems are solvable in polynomial time.

Part (b) follows from (a). So we prove part (a).

Proof:

- Suppose L is $\mathbf{NP}$-complete, and $L \in \mathbf{P}$.
- Thus, for some polynomial h, there is a $h(|x|)$ time bounded TM $A(\cdot)$ which accepts L.
- Consider any problem $L' \in \mathbf{NP}$.
- Suppose $x \in L'$ iff $f(x) \in L$, where f is computed by TM M which is $q(|x|)$-time bounded, for some polynomial q.
- Consider $A'(x) = A(M(x))$. Note that A' accepts L'.
- A' is $q(|x|) + h(q(|x|))$-time bounded, which is polynomial in $|x|$.
- Thus, $L' \in \mathbf{P}$.

Vertex Cover

To see that Vertex Cover is in NP, given a graph (V, E), guess a $V' \subseteq V$, and verify that
(i) $|V'| \leq k$, and
(ii) for all $(v, w) \in E$, at least one of v, w is in V'. If the verification is successful, then accept; otherwise reject.

To show that Vertex Cover is NP-hard, consider the following reduction from 3SAT.
Suppose $U = \{x_1, x_2, \ldots, x_n\}$ is the set of variables and $C = \{c_1, c_2, \ldots, c_m\}$ is the set of clauses, where $c_i = (l_{i,1} \vee l_{i,2} \vee l_{i,3})$.
Then form the vertex cover instance $G = (V, E)$, where $V = \{u_i, w_i : 1 \leq i \leq n\} \cup \{z_{j,1}, z_{j,2}, z_{j,3} : 1 \leq j \leq m\}$.
Let
$E = \{(u_i, w_i) : 1 \leq i \leq n\} \cup \{(z_{j,1}, z_{j,2}), (z_{j,2}, z_{j,3}), (z_{j,1}, z_{j,3}) : 1 \leq j \leq m\} \cup \{(z_{j,r}, u_i) : l_{j,r} = x_i\} \cup \{(z_{j,r}, w_i) : l_{j,r} = \neg x_i\}$.
Let $k = 2m + n$

Intuitively, u_i represents x_i and w_i represents $\neg x_i$. $z_{j,r}$ represents the literal $l_{j,r}$. Clearly the above reduction can be done in polynomial time.
It is easy to verify that in any vertex cover, one must have (i) at least one of u_i, w_i for each i, $1 \leq i \leq n$ and (ii) at least two of $z_{j,1}, z_{j,2}, z_{j,3}$, for each j, $1 \leq j \leq m$. Thus, any vertex cover for G of size at most $2m + n$ must contain exactly one of u_i, w_i for each i, $1 \leq i \leq n$ and exactly two of $z_{j,1}, z_{j,2}, z_{j,3}$, for each j, $1 \leq j \leq m$.

If the 3SAT problem (U, C) has a satisfying assignment, then by correspondingly choosing u_i in V' iff x_i is true, w_i in V' iff x_i is false, and choosing two of $z_{j,1}, z_{j,2}, z_{j,3}$ to be in V' such that if $z_{j,r}$ is left out of V' then the literal $l_{j,r}$ is true, we can easily verify that V' is a vertex cover of G.

If the Vertex Cover problem (V, E) has a vertex cover, then consider the truth assignment: x_i is true iff u_i is in the vertex cover. It can now be shown that if $z_{j,r}$ is not in the vertex cover then, $l_{j,r}$ must be true (otherwise, both the vertices of the edge $(z_{j,r}, s_i)$ are not in the vertex cover, where s_i is u_i, if $l_{j,r} = x_i$, and s_i is w_i, if $l_{j,r} = \neg x_i$.)

Clique/Independent Set

It is easy to verify that clique is in NP: guess a subset $V' \subseteq V$ of size k, and verify that V' is a complete graph. Similarly for Independent Set

Suppose $G = (V, E)$ is a graph. Then, one can show that $G = (V, E)$ has a vertex cover of size k iff $G = (V, E)$ has an independent set of size $|V| - k$ iff $G' = (V, E^c)$ has a clique of size $|V| - k$. Here $E^c = \{(u, v) : u, v \in V, u \neq v\} - E$.

To see this note that V' is a vertex cover of G iff $V - V'$ is an independent set of G iff $V - V'$ is a clique of G'.
This proves that Clique and independent set are NP-complete.