Analysis of Algorithms – HW6

Li-Hen Chen

Department of Computer Science & Engineering, Texas A&M University

1.

A vertex cover of an undirected graph is a subset of its vertices such that for every edge (u, v) of the graph, either 'u' or 'v' is in vertex cover.

VC-D: Given a graph G and an integer k, decide whether G contains a vertex cover of at most k vertices.

VC-O: Given a graph G, construct a minimum vertex (k') cover for G

Assume **VC-O is solvable in polynomial time**. We can tell from the above description that if k'>k, we can solve the both VC-D and VC-O. If k'<k, we cannot solve the VC-D problem. Because we know k' is the minimum vertex number, the k cannot be solved if k'<k. As a result, we can prove that VC-D is solvable in polynomial time only if VC-O is solvable in polynomial time.

2.

If any problem is in NP, then, given a solution to the problem and an instance of the problem (a graph G and a positive integer k in this case), we will be able to verify the certificate in polynomial time.

The certificate for the vertex cover problem is a subset V' of V, which contains the vertices in the vertex cover. We can check whether the set V' is a vertex cover of size k using the following strategy (for a graph G(V, E)):

```
let count be an integer
set count to 0
for each vertex v in V'
    remove all edges adjacent to v from set E
    increment count by 1
    if count = k and E is empty
    then
        the given solution is correct
    else
        the given solution is wrong
```

It is plain to see that this can be done in polynomial time. Thus, the vertex cover problem is in the class NP.

3.

If we would the Clique problem is NP-complete by knowing that the Independent Set problem is NP-complete, we have to construct an algorithm to solve Clique given an algorithm to solve IS. Show that an instance of Clique can be solved using a polynomial number of operations, and a polynomial number of calls to a black box that can solve IS. As a result, I perform the reduction of Clique to Independent Set.
Let's define IS and Clique problem.
IS: Given an undirected graph G and k, is there a set I of k vertices in G' in which no two are adjacent?

Clique: Given a graph G and an integer k, is there a set C of k vertices in G such that for every pair v and w of vertices in C, v and w are adjacent in G?

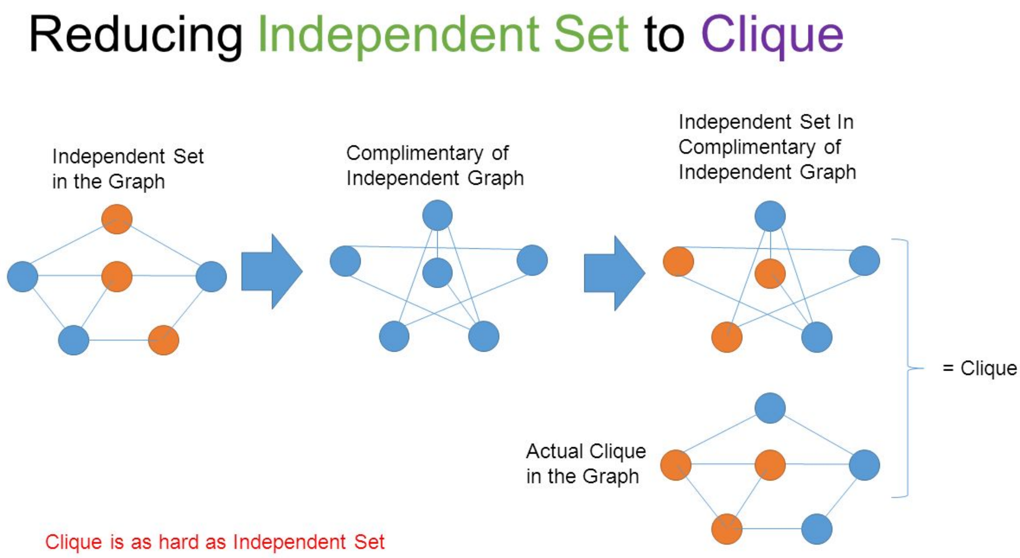
How to prove a NP-complete problem.
In order to prove that a problem Clique is NP-complete, we need to do the following steps:

1. Prove your problem L belongs to NP (that is that given a solution you can verify it in polynomial time)
2. Select a known NP-complete problem (Independent Set)
3. Describe an algorithm f that transforms Independent Set into Clique
4. Prove that the algorithm is correct (formally: $x \in L'$ if and only if $f(x) \in L$)
5. Prove that algorithm f runs in polynomial time

As a result

To reduce IS problem to a Clique problem for a given graph G=(V,E), we have to construct a complimentary graph G'=(V',E'). Therefore, an independent set in a graph G is a clique in the complement of G and vice-versa.
Given this, a simple transformation would be given G and k to produce G^c (the

complement of G) and k. Then, G has an independent set of size k if and only if G^c has a clique of size k. It can be shown in following figure.



As all problems in NP can be reduced to the clique problem (the definition of clique being in NPC) all problems in NP can be reduced to independent set too and so it is in NPC. And since we reduced Independent Set to Clique in polynomial time it cannot be harder than the clique problem either.