

CMSC 351 - Introduction to Algorithms  
Spring 2012  
Lecture 23

**Instructor:** MohammadTaghi Hajiaghayi  
**Scribe:** Rajesh Chitnis

## 1 Introduction

In this lecture we will continue to learn more about NP-completeness.

## 2 Finding a Clique is NP-complete

**Definition 1** A clique  $C$  of a graph  $G$  is a subgraph of  $G$  such that all vertices in  $C$  are connected to each other.

### Clique

Given an undirected graph  $G = (V, E)$  and an integer  $k$ , determine whether  $G$  contains a clique of size  $\geq k$ .

**Theorem 1** CLIQUE is NP-complete.

**Proof:** Clearly the problem is in NP, since given a solution  $C$  as a certificate, we can check whether it is a clique in  $O(|C|^2)$  time. Now we reduce 3-SAT to CLIQUE. Let the input for 3-SAT be  $E = E_1 E_2 \dots E_m$ . For each clause  $E_i = (x + y + z)$  of 3-SAT we associate a “column” of three variables (even if they appear in other clauses). For edges, vertices from the same column are not connected. Vertices from different columns are connected except when they correspond to the same variable appearing in complementary form. We set  $k = m$  (we are free to choose  $k$ ). For example, if  $S = (x+y+z)(\bar{x}+\bar{y}+z)(\bar{x}+y+\bar{z})$ , then the construction is as shown in Figure 1. Now we claim that  $G$  has a clique of size  $\geq m$  if and only if  $E$  is satisfiable. Note that since we can choose at most one vertex from each column, the clique size can not exceed  $m$ .

Assume  $E$  is satisfiable. Then there is a truth assignment in which each clause contains at least one true variable. Choose that variable, and we get a clique of size  $m$ .

Conversely, if we have a clique of size  $m$ , then since from each column we can pick only one vertex, if we assign the corresponding variable a value 1, and other variable arbitrary, we are done. Since all vertices in the clique are connected to one another, and we made sure that  $x$  and  $\bar{x}$  are never connected, this truth assignment is consistent. ■

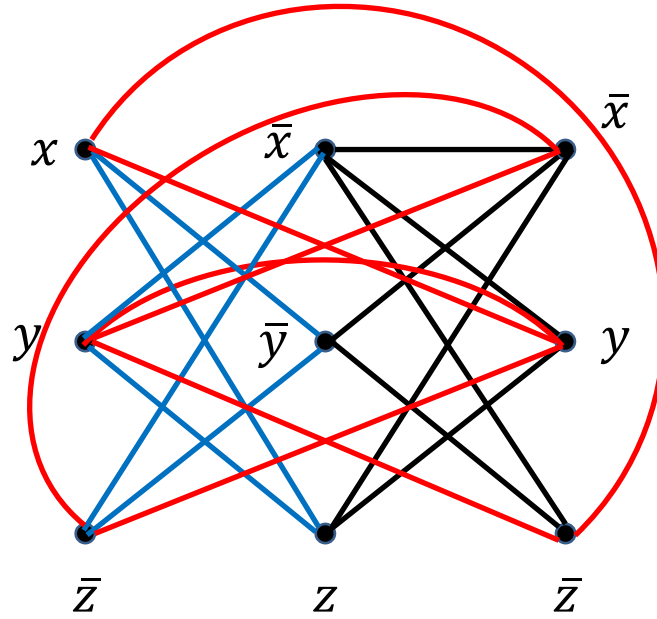


Figure 1:

### 3 Finding a Vertex Cover is NP-complete

**Definition 2** Let  $G = (V, E)$  be a given graph. A set  $C \subseteq V(G)$  is a set of vertices such that every edge in  $E(G)$  is incident to at least one vertex from  $C$ .

**Vertex Cover**

Given an undirected graph  $G = (V, E)$  and an integer  $k$ , determine whether  $G$  has a vertex cover of size  $\leq k$ .

**Theorem 2** VERTEX COVER is NP-complete.

**Proof:** Clearly the problem is in NP, since given a solution  $C$  as a certificate, we can check whether it is a vertex cover in  $O(|E| + |V|)$  time. Now we reduce

CLIQUE to VERTEX COVER. We have to transform an arbitrary instance of the CLIQUE into an instance of VERTEX COVER such that answer to CLIQUE is YES if and only if the answer to VERTEX COVER is YES. Let  $G = (V, E)$  be an instance of CLIQUE. Let  $\bar{G} = (V, \bar{E})$  be the complement graph of  $G$ ; namely  $\bar{G}$  has the same set of vertices and two vertices are connected in  $\bar{G}$  if and only if they are not connected in  $G$ . We claim that the CLIQUE problem is reduced to VERTEX COVER represented by the graph  $\bar{G}$  and  $n - k$ , where  $n = |V|$ .

Suppose  $C$  is a clique in  $G$ . Then the set of vertices  $V \setminus C$  covers all the edges of  $\bar{G}$ , because in  $\bar{G}$  there are no edges connecting vertices in  $C$  (they are all in  $G$ ). Thus  $V \setminus C$  is a vertex cover of size  $n - k$  in  $\bar{G}$ .

Conversely, let  $D$  be a vertex cover in  $\bar{G}$ . Then  $D$  covers all the edges in  $\bar{G}$ , so in  $\bar{G}$  there cannot be any edges connecting vertices in  $V \setminus D$ , i.e.,  $V \setminus D$  forms a clique in  $G$ .

Therefore there is a vertex cover of size  $n - k$  in  $\bar{G}$  if and only if there is a clique of size  $k$  in  $G$ . The reduction is clearly poly-time since it requires only the construction of  $\bar{G}$  from  $G$  which takes  $O(|V|^2)$  time. ■

## 4 More NP-complete Problems

Now we briefly describe some more well-known NP-complete problems:

### 1. Bin-Packing or Knapsack

We have seen these earlier in the course.

### 2. Independent Set

Given a graph  $G$  and an integer  $k$ , determine whether  $G$  contains an independent set of size  $\geq k$ . The NP-completeness follows from a straightforward reduction from Clique.

### 3. Dominating Set

Given a graph  $G$  and an integer  $k$ , determine whether  $G$  contains a dominating set of size  $\leq k$  (i.e. a set  $D$  of vertices such that every vertex of  $G \setminus D$  is adjacent to at least one vertex from  $D$ ). The NP-completeness follows from a straightforward reduction from Vertex Cover.

### 4. Hamiltonian Cycle

A Hamiltonian cycle in a graph is a simple cycle that contains each vertex exactly once. The problem is to determine whether a given graph contains a Hamiltonian cycle. The problem is NP-complete for both undirected and directed graphs by reduction from Vertex Cover.

### 5. Hamiltonian Path

A Hamiltonian path in a graph is a simple path that contains each vertex exactly once. The problem is to determine whether a given graph contains a Hamiltonian path. The problem is NP-complete for both undirected and directed graphs by reduction from Vertex Cover.

**6. Set Cover**

Given a set of elements  $\{1, 2, \dots, n\}$  (called the universe) and  $n$  sets whose union comprises the universe and an integer  $k$ , the Set Cover problem is to find at most  $k$  sets whose union still contains all elements of the universe. The NP-completeness follows from reduction from Vertex Cover. Set Cover is a very important problem in practical applications.

Note that in general for all NP-complete problems we can use backtracking and branch-and-bound techniques to obtain the optimum solution (in exponential time). Alternatively, we can use approximation algorithms to obtain an approximate solution in polynomial time.

**References**

- [1] Udi Manber, *Introduction to Algorithms - A Creative Approach*