Vertex Cover

RJ, JaeSang, Eric

April 26, 2020

Chapter 1

What is Vertex Cover

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. Although the name is Vertex Cover, the set covers all edges of the given graph.

1.1 What Does that mean

It is a set that contains vertices that contain every edge.

1.1.1 What is the Vertex Cover Problem

Given an undirected graph, the vertex cover problem is to find minimum size vertex cover.

1.1.2 Real Life Example

Let us say that a vertex of a graph is a crossroads and the edges of the graph represent roads. We are given the task to put up security cameras. The city demands us to use the least number possible. By using vector cover, we are able to accomplish that.

Chapter 2

Is the vertex cover problem in NP?

2.1 What Does it mean to be NP

To show that the VC problem is in NP, we think about the definition of NP-problem. The problem that can be solved in polynomial time with non-deterministic algorithm is in NP. If I got a candidate of solution of problem in NP, then I can check whether it is valid solution in polynomial time.

2.1.1 The Proof

So, give me a set S of vertices of G(V,E). 1. Choose one edge without any marking from E. 2. Check it is incident of some vertex in S. 3. If it is, then mark on the edge and go to step 1 and repeat. If not, then the set S is not vertex cover so end the process. 4. If every edge in E is incident of some vertex in S, then the S is vertex cover of G(V,E). The process from 1 to 3 or 4 will take $|E|$ time, which is definitely polynomial time. In conclusion, the vertex cover problem is in NP.

Chapter 3

Is Vertex Cover NP Hard

3.1 How to Prove NP Hard

To show vertex cover is in NP-Hard, we think about the Karp's reduction.

Assume that we have a solver for a problem and that we want to show it is NP-Hard. If we can convert any input for a known NP-complete problem into input for our solver with polynomial time work.

And if the answer for the problem is true if and only if the answer for NP-complete problem is true, then the problem is in NP-Hard.

3.1.1 The reduction Problem

We can use the fact that CLIQUE is in NP-Complete. If we can convert input to CLIQUE into VC with polynomial time work,then we can show VC is in NP-Hard.

So, what is CLIQUE?

The CLIQUE problem is the computational problem of finding cliques (subsets of vertices, all adjacent to each other, also called complete subgraphs) in a graph.

3.1.2 step 1

To find a clique in graph G(V,E).

Given a graph G(V, E) and a positive integer k, if there exists a set of vertices S $\subseteq$ V such that $|S| \geq$ k and for all u, v $\in$ S (u, v) $\in$ E then it is a clique.

To make it easy, CLIQUE is finding a clique that is a complete subgraph of G(V,E).

3.1.3 step 2

The input to CLIQUE is G(V,E).

If we convert the input G(V,E) for CLIQUE into G'(V,E') ,then it is input for VC.

Also it takes $|V2|$ time to convert the input.

But we have to show that in G(V,E) , a set S of vertices is the clique if and only if $V - S$ is the vertex cover in G'(V,E').

3.1.4 step 3

If S is a clique in G(V,E), then $V - S$ is a vertex cover G'(V,E'). Since S is a clique, for any edge (u,v)$\in$ E', one of u or v is not incident to any vertex in S. So one of u or v or both are incident to some vertex in $V - S$. That means $V - S$ covers every edge in E',which means $V - S$ is a vertex cover of G'(V,E')

3.1.5 step 4

If $V - S$ is the vertex cover in G'(V,E'), then S is the clique in G(V,E). Since $V - S$ is the vertex cover in G'(V,E'), if (u,v) $\in$ E' then u or v or both are in $V - S$. So if u $\notin V - S$ and v $\notin V - S$ then (u,v) $\notin$ E' by contrapositive. i.e If u $\in$ S and v $\in$ S then (u,v) $\in$ E. That means S is a clique in G(V,E).

Vertex Cover is NP Hard and therefore NP complete