ECE374 SP23 HW9

Contributors

Zhirong Chen (zhirong4)

Ziyuan Chen (ziyuanc3)

Problem 3

Are the following problems in P, NP, co-NP, NP-Hard, NP-complete? Either way, prove it.

(a) A kite is a graph on an even number of vertices, say $2n$, in which n of the vertices form a clique and the remaining n vertices are connected in a tail that consists of a path joined to one of the vertices of the clique. Given a graph and a goal g, the *max kite problem* asks for a subgraph that is a kite and contains $2g$ nodes. What complexity classes does kite belong in?

(b) A $4\mathrm{kite}$ is exactly the same problem, but this time $g = 4$. What complexity classes does $4\mathrm{kite}$ belong in?

Solution

(a)

The *max kite problem* is in **NP**, because given a certificate (a subgraph of G with $2g$ nodes that is a kite), we can verify in polynomial time whether the subgraph is indeed a kite by looking for a g-sized clique and a g-sized path.

We then show the *max kite problem* is NP-hard by a reduction from 3-SAT.

Given an instance of 3-SAT with variables $x_1, x_2, \ldots, x_n$ and clauses $C_1, C_2, \ldots, C_m$, we construct a graph G as follows:

- For each variable x_i, we create a clique K_i on three vertices, corresponding to the three possible truth values of x_i.
- For each clause C_j with literals l_1, l_2, l_3, we create a tail consisting of a path of length two connected to one of the vertices of the clique corresponding to the literal's variable. Specifically, if l_k is the literal x_i or $\neg x_i$, we connect the first vertex of the path to the vertex in K_i corresponding to l_k.
- Finally, we add edges between all pairs of vertices in different cliques.

It can be shown that G has a kite with $2n$ nodes if and only if the 3-SAT instance is satisfiable. Specifically, we can map each vertex in the kite to a variable or its negation, and the clique vertices to the possible truth values of the variable. Then, the path connected to a clique vertex represents the truth value assigned to the variable in the satisfying assignment.

Therefore, the kite problem is NP-complete.

ANOTHER SOLUTION

Checking whether a subgraph G_sub is a kite can be done in polynomial time: Assume G_sub has 2n vertices. To verify that G_sub is a kite, we can follow these steps:

- Select the n vertices with the highest degree.
- Check whether these n vertices form a clique.
- If it is a clique, then check whether the other n vertices form a path attached to this clique.

This procedure can be done in polynomial time. Because a YES instance can be verified in polynomial time, the max kite problem is an NP problem.

We can prove the max kite problem is NP-hard by using a reduction from the NP-complete clique problem. Given an arbitrary graph G and integer k, we obtain H by adding a path with length k to each vertex in G. Specifically, for any vertex u ∈ G, add k vertices v_u,i (i = 1, 2, ...k), and add k edges (v_u,i, v_u,i+1) for i = 1, 2, ...k-1 and (u, v_u,1).

Now, we can show that:

- If there's a clique in G with size k, then let g=k; there must be a kite in H with size 2g. This is because the clique must be in H, and there is always a path with length g connected to a vertex in the clique.
- If there's a kite with size 2g in H, let k=g; there must be a clique in G with size k. This is because the kite in H contains a clique with size k, and this clique must be in G since the added vertices v_u,i (u ∈ G, i = 1, 2, ...k) cannot be in a clique when k > 2. If k <= 2, then G must have a clique with size k.

By the reduction from an NP-complete problem, the max kite problem is proven to be NP-complete because it is also an NP problem.

(b)

$4\mathrm{kite}$ is in **P**.

Given that g=4, we can employ an $O(n^8)$ algorithm to examine all possible sets of 4 vertices constituting the clique and 4 vertices composing the path. Subsequently, we can verify in polynomial time if these chosen 8 vertices create a valid 4-kite. As the entire process of discovering a solution operates within polynomial time and requires polynomial space, the 4-kite problem is classified as a P problem.