COM S 511 Homework 10

Mahdi Banisharifdehkordi
Student ID: 994741802

April 19, 2024

Problem 1

Membership in NP

A problem is in NP if there exists a certificate for a "yes" instance of the problem that can be verified in polynomial time. For the Star Problem, a subgraph $H \subseteq G$ claimed to be a shooting star with $2r$ vertices. To confirm that H contains exactly $2r$ vertices, we need to check that exactly r of these vertices form a clique and verify that the remaining r vertices form a path, which is connected to one of the clique's vertices.

These verification steps are computationally basic and can be executed in polynomial time, confirming that STAR is in NP.

NP-Hardness

To establish that the Star Problem is NP-hard, we perform a polynomial-time reduction from the CLIQUE problem, which is known to be NP-complete.

Reduction Process

1. **Given Instance:** Take a graph $G = (V, E)$ with a clique number k from the CLIQUE problem.
2. **Constructing an Instance of Star Problem:**
   - Utilize the vertices V and edges E from G.
   - Add k new vertices $\{u_1, u_2, \ldots, u_k\}$, to form a path.
   - Connect u_k to each vertex in V, enabling potential attachment to any clique in G.
   - Set $r = k$ in the Star Problem.
3. **Reduction Validity:**
   - **If CLIQUE has a solution:** A clique of size k in G combines with the path of new vertices to form a shooting star of size $2k$ in the new graph.
   - **If Star Problem has a solution:** The clique portion of the shooting star must correspond to a clique of size k in G, verifying the clique solution.

The polynomial-time reduction from CLIQUE to the Star Problem indicates that solving the Star Problem is at least as difficult as solving the CLIQUE problem. Since CLIQUE is NP-complete, this implies that the Star Problem is also NP-hard. Combining its NP membership and NP-hardness confirms that the Star Problem is NP-complete.

Problem 2

We propose a brute-force algorithm, Algorithm A, to determine if at least k rectangles overlap. This algorithm checks all pairs of rectangles to identify overlapping pairs, a task that involves $O(n^2)$ operations since each pair comparison requires $O(1)$ time. Thus, the overall time complexity is polynomial, and we conclude that RECTANGLE $\in$ P.

To prove RECTANGLE is polynomial-time reducible to CLIQUE (RECTANGLE $\leq_p$ CLIQUE), we note that any problem in P, including RECTANGLE, can be reduced to any NP-complete problem such as CLIQUE. This is because solving RECTANGLE directly in polynomial time allows us to construct an instance of CLIQUE that is trivially solvable if RECTANGLE is solvable.

Specific Reduction

We can create a more explicit reduction to demonstrate how RECTANGLE $\leq_p$ CLIQUE:

- Construct a graph where each vertex represents a rectangle.
- Add an edge between two vertices if their corresponding rectangles overlap.

In this graph, a clique of size k represents a set of k rectangles where each overlaps with all others in the set. This mapping directly translates an instance of RECTANGLE into CLIQUE:

1. Construct vertices for each rectangle in R.
2. Connect vertices with edges if their rectangles overlap, requiring $O(n^2)$ pairwise comparisons.

This transformation is polynomial in time, ensuring that solving the CLIQUE problem with this graph also solves the RECTANGLE problem.

Given RECTANGLE $\in$ P, it is also in NP since any problem in P is also in NP. As RECTANGLE is in P and we assume $P \neq NP$, RECTANGLE cannot be NP-complete unless $P = NP$. From the reduction RECTANGLE $\leq_p$ CLIQUE and knowing RECTANGLE $\in$ P:

- CLIQUE is at least as hard as RECTANGLE.
- Since anything in P can be as easy or easier than RECTANGLE, this does not imply CLIQUE $\in$ NPC based on this reduction alone.

Note: While we know CLIQUE $\in$ NPC, this analysis by itself does not lead us to that conclusion independently of known results.