CMSC 858F: Algorithmic Lower Bounds: Fun with Hardness Proofs
Spring 2022
Chapter 5: NP-Hardness Using a Miscellany of Graph Problems

Instructor: Mohammad T. Hajiaghayi
Scribe: Joel Rajakumar

April 22, 2022

1 Summary

This chapter involves proving the NP-completeness of different problems using vertex covers, coloring, and ordering. The significant problems discussed are listed below:

1. Problems in P
   (a) Exact vertex cover
   (b) Edge cover
   (c) 2-coloring
2. Problems that Vertex Cover can be reduced to
   (a) Shortest Common Subsequence
      i. Restricted Shortest Common Subsequence $\leq_p$ Flood-it on grid
   (b) Induced Subgraph Vertex Cover
      i. Rectilinear Steiner tree
3. Problems that k-Coloring can be reduced to
   (a) Planar 3-coloring
      i. Push 1-X
      ii. Push 1-G

4. Problems that graph orientation can be reduced to
   (a) Packing trominoes into a polygon
5. Problems that MinLA can be reduced to
   (a) Crossing number
   (b) Bipartite crossing number
   (c) Other Linear Layout problems

2 Comments and Suggestions

- I thought that each section within the chapter seemed a little disconnected from the previous. This is probably unavoidable as each section deals with very different problems. However, I think there should be a brief sentence or two at the end of each section, introducing the next section so that the reader can follow the structure of the chapter.
- I think there should be a hint for Exercise 5.1.3 to inform the reader on which graph problem to reduce from. I had difficulty trying to reduce from 3-colorability. In the homework, I solved the problem by reducing from Exact Cover, but Exact Cover is not mentioned in this chapter. Adding to that, I think it would be useful to mention Exact Cover briefly along with vertex cover, or give it as another exercise problem.
- As opposed to the previous chapter, this chapter involves several differently structured graph problems that are all NP-complete. Personally, I often find myself wasting a lot of time (in Homeworks) trying to reduce directly from the wrong graph problem, which can get difficult. I think it would be useful to include a very brief discussion/list of the sort of problems that each of Hamiltonian Cycle, Vertex Cover, k-coloring, and ordering can be easily reduced to (e.g. perhaps some of Garey and Johnson's graph theory problems). This way it improves the reader's understanding of how to use them as tools in hardness proofs.
- In the beginning of section 5.3, the steps to reduce the Flood-it game are described, but the flood-it game itself is not defined till 5.3.2. Without an idea of what the flood-it game is, it was hard to see why we cared about the RSCS problem until much later. I think there are two ways to structure this that could be better:
  1. Move the definition of flood-it in 5.3.2 to the beginning of section 5.3, then outline the steps to reduce the Flood-it game (i.e. currently page 149), then define the SCS and RSCS problem (5.3.1), and finally complete the reduction proof.
  2. Create a separate section before 5.3 (before mentioning the flood-it game) for the SCS and RSCS problems alone. Reference these problems in the Flood-it section that comes after.

3 Problems

1. A BICLIQUE EDGE COVER is a set of bicliques (that is complete bipartite subgraphs), that covers all edges in E. Prove the following problem is NP-complete
   - INSTANCE: A graph $G(V, E)$ and a positive integer k
   - QUESTION: Does G admit a biclique edge cover containing at most k bicliques?
   - BONUS: Prove it remains NP-complete for bipartite-graphs
2. A CLIQUE COVER is a partition of V into cliques (that is complete subgraphs). Prove the following problem is NP-complete
   - INSTANCE: A graph $G(V, E)$ and a positive integer k
   - QUESTION: Does G admit a clique cover containing at most k cliques?
   - BONUS: Prove it remains NP-complete for planar graphs
3. A COMPLETE COLORING is the opposite of the coloring that you may be used to in the sense that it is a vertex coloring in which every pair of colors appears on at least one pair of adjacent vertices. Prove the following problem is NP-complete
   - INSTANCE: A graph $G(V, E)$ and a positive integer k
   - QUESTION: Does there exist a partition of V into k or more disjoint sets $V_1, V_2, \dots, V_k$ such that each V_i is an independent set for G and such that for each pair of distinct sets $V_i, V_j, V_i \cup V_j$ is not an independent set.
4. A DOMATIC PARTITION is a partition of V into disjoint sets $V_1, V_2, ..., V_k$ such that each V_i is a dominating set for G. Prove the following problem is NP-complete
   - INSTANCE: A graph $G(V, E)$ and a positive integer k
   - QUESTION: Does there exist a domatic partition of V into at least k disjoint sets?
5. Two graphs $G(V, E), H(V', E')$ are isomorphic (i.e. $G \cong H$) if there exist a bijection $f \colon V \to V'$ such that $\{v_1, v_2\} \in E \iff \{f(v_1), f(v_2)\} \in E'$. Prove the following SUBGRAPH ISOMORPHISM problem is NP-complete
   - INSTANCE: A graph $G(V, E)$ and a graph $H(V, E)$
   - QUESTION: Is there a subgraph $G_0 = (V_0, E_0) \mid V_0 \subseteq V, E_0 \subseteq E \cap (V_0 \times V_0)$ such that $G_0 \cong H$
   - BONUS: Prove that it remains NP-complete for planar graphs

6. A FEEDBACK VERTEX SET of a graph is a set of vertices whose removal leaves a graph without cycles. Prove the following problem is NP-complete
   - INSTANCE: An (undirected or directed) graph $G(V, E)$ and a positive integer k.
   - QUESTION: Is there a subset $X \subseteq V$ with $|X| \leq k$ such that, when all vertices of X and their adjacent edges are deleted from G, the remainder is cycle-free?

7. Prove the MONOCHROMATIC TRIANGLE problem is NP-complete
   - INSTANCE: A graph $G(V, E)$
   - QUESTION: Is there a partition of E into two disjoint sets E_1, E_2 such that neither $G(V, E_1)$ nor $G(V, E_2)$ contains a triangle?

8. Most quantum computers have limited "connectivity," which means you can only do quantum logic gates (which are similar to classical AND/OR gates) between certain pairs of qubits but not all. Thus, we can represent the physical layout of a quantum computer by a graph $G(V, E)$, where the vertices represent physical qubits, and the edges represent the pairs of physical qubits on which logic gates can be performed. When a scientist writes a program for this quantum computer, he wants to perform a sequence of logic gates on pairs of "logical" qubits V', i.e. qubits for which he has set a logical value of 0 or 1 in the beginning of his program. The question is, is there a way to assign the scientist's logical qubits onto physical qubits during compilation so that the limited connectivity matches up with his desired gates? Thus, prove that the QUBIT ASSIGNMENT problem is NP-complete
   - INPUT: A graph of the physical qubits $G(V, E)$, a set of logical qubits V', and a set of pairs representing the logical qubits used in each gate operation $P = \{(v_1', v_2'), ...(v_i', v_j')\}$ where $v_i', v_j' \in V'$
   - QUESTION: Is there is a mapping $f : V' \to V$, such that $(f(v_i'), f(v_j')) \in E \; \forall (v_i', v_j') \in P$?
   - HINT: This is basically the subgraph isomorphism problem in disguise
   - SOURCE: https://dl.acm.org/doi/10.1145/3168822

9. Consider the following scheduling problem. You are given a list of final exams $F_1, ...F_n$ to be scheduled, and a list of students $S_1, ...S_m$. Each student is taking some specified subset of these exams. You must schedule each exam into exactly one slot such that no student is required to take two exams in the same slot. The problem is to determine if such a schedule exists that uses only k slots. Prove this problem is NP-complete.

10. Prove the EDGE DOMINATING SET problem is NP-complete.

- INPUT: A graph $G(V, E)$ and a positive integer k
- QUESTION: Does G contain a set of at most k edges that dominate all other edges in G?
- BONUS: Prove that it remains NP-complete for (i) Bipartite graphs with maximum degree 3 (ii) Planar graphs with maximum degree 3