Algorithm Homework 4

龙肖灵
Xiaoling Long
Student ID.:81943968
email:longxl@shanghaitech.edu.cn

November 25, 2017

Note: When proving problem A is NP-complete, please clearly divide your answer into three steps:

(1) Prove that problem A is in NP.

(2) Choose an NP-complete problem B and for any B instance, construct an instance of problem A.

(3) Prove that the yes/no answers to the two instances are the same.

Problem 1

Given a conjunctive normal form formula and an integer k, can this formula be satisfied by an assignment in which at most k variables are true? Prove this problem is NP-complete.

Solution.

(1) Certification: Give an assignment in which less than or equal k variable are true. We can certify this problem whether this formula is satisfied. So, this problem is NP.

(2) Reduction 1: First of all, $3-SAT$ is a special case of this problem when $k = n$. If we can solve this problem. We can solve all $3-SAT$ problem. So 3-SAT $\leq_p CNF_k$

(3) Reduction 2: And Similarly, we can construct an instance of CNF_k from any circuit instance.

Step 1: Create a CNF_k variable x_i fpr each circuit element i.

Step 2: Convert all logical predicates into $\vee$:

- $x = \neg y \quad \Rightarrow \quad x \vee y,\ \bar{x} \vee \bar{y}$
- $x = y \vee z \quad \Rightarrow \quad x \vee \bar{y},\ x \vee \bar{z},\ \bar{x} \vee y \vee z$
- $x = y \wedge z \quad \Rightarrow \quad \bar{x} \vee y,\ \bar{x} \vee z,\ x \vee \bar{y} \vee \bar{z}$

Step 3: Set output value and input value. If it is 1, then x, or not $\neg x$

Step 4: Then CNF_k consist of all of clauses.

(4) Proof(2): So if this formula can be satisfied by an assignment when $k = n$, the CIRCUIT-SAT can be satisfied, since output value equals to 1. Or not, the CIRCUIT-SAT cnnot be satisfied. And if CIRCUIT-SAT can be satisfied, our construction ensures that all the nodes in CIRTUIT-SAT are correctly coputed an the output is 1, the CNF_k can be satisfied as well.

(5) Conclusion: CIRCUIT-SAT$\leq_p$ CNF_k. This problem CNF_k is NP-complete.

Done. ■

Problem 2

Textbook Chapter 8 Exercises 17 (Note: You must use the Directed-Hamiltonian-Cycle problem in your reduction.) You are given a directed graph $G = (V, E)$ with weights we on its edges $e \in E$. The weights can be negative or positive. The Zero-Weight-Cycle Problem is to decide if there is a simple cycle in G so that the sum of the edge weights on this cycle is exactly 0. Prove that this problem is NP-complete.

Solution.

(1) Certification: Given a cycle, we can add all weights to check whether this circle is a ZERO-Weight-Cycle or not. So Zero-weight-Cycle problem is NP.

(2) Construction: Split each directed edge into 2 edges from any Directed-Hamiltonian-Cycle instance G and the weights of the edges are 1 and -1 respectively. Such as there exists an edge $(u, v) \in E$.

$$\text{u} \longrightarrow \text{v}$$

After splitting, we have sub-graph in G' as following,

$$\text{u} \xrightarrow{\ 1\ } v' \xrightarrow{\ \text{-1}\ } v''$$

And all edges come into v come into v' now, and all edges leave out leave from v'' now.

(3) Proof: If there is a cycle in G, then there must exist a cycle along with all node v_i' and v_i'', where v_i are the nodes in the cycle in graph G.
If there exists a Zero-Weight-Cycle in G', then we can also find a cycle in G in same order.

(4) Conclusion: Directed-Hamiltonian-Cycle$\leq_p$ Zero-Weight-Cycle. So Zero-Weight-Cycle problem is NP-complete.

Done. ■

Problem 3

Given a graph $G = (V, E)$, select a subset of the nodes such that for every node $v \in V$, v is selected or at least one of its neighbors is selected. We would like to know if we can find such a subset of at most K nodes. Show that this problem is NP-complete.

Solution.

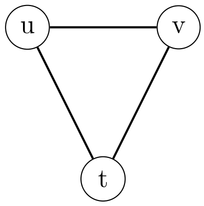
(1) Certification: Given a subset, we can check whether all nodes can be satisfied. So this problme is NP.

(2) We can reduction from **Vertex Cover** .

(3) Construction: Construct a triangle based on each edge from **Vertex Cover (**G**)**. If there is an edge in G

$$\text{u} \text{ ——— } \text{v}$$

Then we add a nodes and two edges into G' as following,



Then we can get the new graph G'.

(4) Proof: If **Vertex Cover** can be satisfiede, then we can choose the same nodes in subset to satisfy the problem.
If The graph G' can be satisfied, then in one trangle we at least selected a node, if we selected node t, we can selected any one of u and V in the problem **Vertex Cover** (G) . The **Vertex Cover** can be satisfied .

(5) Conclusion: Vertex Cover $\leq_p$ Subset selection. This problem is NP-complete.

Done. ■

Problem 4

Suppose you are going to schedule courses for the SIST and try to make the number of conflicts no more than K. You are given 3 sets of inputs: $C = \{\cdots\}, S = \{\cdots\}, R = \{\{\cdots\}, \{\cdots\}, \cdots\}$. C is the set of distinct courses. S is the set of available time slots for all the courses. R is the set of requests from students, consisting of a number of subsets, each of which specifies the courses a student wants to take. A conflict occurs when two courses are scheduled at the same slot even though a student requests both of them. Prove this schedule problem is NP-complete. Example:

$$K = 1; C = \{a, b, c, d\}, S = \{1, 2, 3\}, R = \{\{a, b, c\}, \{a, c\}, \{b, c, d\}\}$$

An acceptable schedule is:

$$a \to 1; b \to 2; c, d \to 3;$$

Here only one confilct occurs. An unacceptalbe schedule is:

$$a \to 1; b, c \to 2; d \to 3;$$

Here two $(> K)$ conflicts occur.

Solution.

(1) Certification: Given a schedule, we can count the number of the confilct. So , this schedule problem is NP.

(2) We can reduction from 3-Colorability problem.

(3) Construction: C is the set of all nodes. S is the set of 3 colors. If there exists an edge $(u, v) \in E$, it means that there is a student wants to take these two courses u and v. And $K = 0$.

(4) Proof: If the schedule problem can be satisfied, then there is no course a student wants to take in same slot. This ensure no adjacent nodes have the same color.
If the 3-Colorability can be satified, then all red nodes can be scheduled into slot 1, blue into slot 2,

and green into slot 3. And all students' courses at the different slots. The schedule problem can be satified as well.

(5) Conclusion: 3-Colorability $\leq_p$ Schedule Problem. So this schedule problme is NP-complete.

Done. ■

Problem 5

A company has two trucks and must deliver a number of parcels to a number of addresses. They want both drivers to be home at the end of the day. This gives the following decision problem:
Instance: Set V of locations; for each pair of locations $v, w \in V$, an integer distance $d(v, w)$; a starting location $s \in V$; and an integer K.
Question: Are there two cycles that both start at s, such that every location in V is on at least one of the two cycles, and both cycles have length at most K?
Show that this problem is NP-complete.
Note: A cycle is a path $v_1, v_2, \cdots, v_{k-1}, v_k$ in which $v_1 = v_k$, $k > 2$, and the first $k-1$nodes are all distinct.

Solution.

(1) Certification: Given to cycles, we can chech whether this planning problem can be satisfied. So this planning problem is NP.

(2) We can reduction from HAMILTONIAN-CYCLE Problem. A Hamiltonian cycle can be splitted into 2 isometric cycle after adding 2 edges.

(3) Construction: Select any node in the V as start node. Then add some edges which nodes isn't adjacent to s and the weight is 2. And the weight of all $(u, v) \in E$ is 1. And $K = \lceil \frac{n}{2} \rceil + 2$

(4) Proof: If HAMILTONIAN-CYCLE Problem can be satisfied, then we can start at s and go throught same length or one is greater 1 than another, then come back to s.
If the planning problem can be satisfied, then delete the additional edges, connect the last nodes of two cycle. We can get the Hamiltonian cycle.

(5) Conclusion: HAMILTONIAN-CYCLE $\leq_p$ Planning Problem. So this problem is NP-complete.

Done. ■

Problem 6

Consider the Knapsack problem. We have n items, each with weight a_j and value c_j $(j = 1, \cdots, n)$. All a_j and c_j are positive integers. The question is to find a subset of the items with total weight at most b such that the corresponding profit is at least k (b and k are also integers). Show that Knapsack is NP-complete by a reduction from **Subset Sum** .

Solution.

(1) Certification: Given a subset, we can certify whether this problem can be satisfied. So this knapsack is NP.

(2) Construcation: We can do some assignment as following

$$c_i = 1, a_i = w_i, i = 1, \cdots, n \quad c_{n+1} = n, a_{n+1} = \sum_i w_i - W$$

We also have $b = \sum_i w_i$ and $k = n+1$. If we can find a subset with total weight b and the correspoding profit is at least $n + 1$, then we can solve this **Subset Sum** .

(3) Proof: If This **Knapsack Problem** can be satisfied, then the subset of Knapsack without a_{n+1} will be the solusiton of **Subse Sum** .
If **subsection Sum** can be satisfied, then the construction ensure the knapsack can be satisfied.

(4) Conclusion: Subset Sum $\leq_p$ Knapsack Problem. So this Knapsack Problem is NP-complete.

Done. $\blacksquare$