Homework 9

Name: 方嘉聪 ID: 2200017849

Problem 1.(Search and Decision Problems).

As discussed in class, NP is a class of decision problems, i.e., the answer is either "yes" or "no". This choice is convenient and often also captures the difficulty of searching for a solution. In our class, we have shown the reduction from search to decision for some problems. Here, you will show the search-to-decision reduction for two more examples.

(1) **3-SAT**: A 3-SAT formula consists of m clauses, each of which is a disjunction of three literals (where a literal is one of n variables or its negation). We say a 3-SAT formula F is satisfiable if there is an assignment of the n variables such that F evaluates to true. The 3-SAT problem asks us to decide whether a given F is satisfiable.

Suppose you are given a black box for a function 3SAT that determines whether a given 3-SAT formula is satisfiable with timing cost T_{3SAT}. Design an algorithm that finds a satisfying assignment for F (assuming it is satisfiable) using $O(n)$ calls to 3SAT and polynomially many other steps. Prove that your algorithm is correct and analyze its running time.

(2) **3-Coloring**: We say an undirected graph is 3-colorable if there is an assignment of the colors $\{r, g, b\}$ to the vertices (a coloring) such that no two adjacent vertices have the same color. The 3-Coloring problem asks us to decide whether a given graph is 3-colorable.

Suppose you are given a black box for a function 3Color that determines whether a given graph is 3-colorable with timing cost T_{3Color}. Design an algorithm that finds a coloring of a given graph using $O(n)$ calls to 3Color and polynomially many other steps, where n is the number of vertices in the input graph. Prove that your algorithm is correct and analyze its running time. ◀

Answer. (1) 只需要对 n 个变量依次赋值, 考虑如下的算法:

Algorithm 1 Find a satisfying assignment for 3-SAT

Input: a 3-SAT formula F with n variables, m clauses.

Output: a satisfying assignment S for F.

1: 把 F 中的一个变量 x_1 赋值为 1.
2: 将 F 中包含 x_1 的子句删去, 对于包含 $\neg x_1$ 的子句 $(\neg x_1 \vee y \vee z)$, 替换为 $(y \vee y \vee z)$ 其中 $(y \neq \neg x_1)$.
3: 设新得到 formula 记为 F'. 调用 `3SAT`(F'), 记结果为 R.
4: **if** $R = 1$ **then**
5: `S.append`$(x_1 = 1)$, 对 F' 递归调用第 1 行 (若为空直接返回 S).
6: **else**
7: `S.append`$(x_1 = 0)$
8: 将 F 中包含 $\neg x_1$ 的子句删去, 对于包含 x_1 的子句 $(x_1 \vee y \vee z)$, 替换为 $(y \vee y \vee z)$ 其中 $(y \neq x_1)$.
9: 记这样得到的 formula 为 F'', 对 F'' 递归调用第 1 行 (若为空直接返回 S).
10: **return** S

算法正确性: 第 2 行的算法时由于若 $x_1 = 1$ 则包含 x_1 的子句赋值为 1, 包含 $\neg x_1$ 的子句赋值取决于子句内的其他变量. 第 8 行同理. 每一次调用 F 的变量数减 1, 递归最多需要 n 次.

时间复杂度: 总共调用 $O(n)$ 次 3SAT, 每一步只需遍历一遍 formula 用时 $O(m)$, 是多项式时间的. 故总的时间复杂度为 $O(nT_{3SAT} + mn)$

(2) 设图 $G = (V, E)$, $|V| = n$. 首先调用 `3Color`(G), 若返回否, 说明 G 不是 3-colorable 的. 若返回是, 向 G 中添加点 x_r, x_g, x_b 和边 $(x_r, x_g), (x_r, x_b), (x_g, x_b)$, 记得到的图为 G_0, 那么 G_0 也是 3-colorable 的. 下面我们通过递归的方式找到 G 的一个 3-coloring. 按照 $v_1, v_2, \cdots, v_n$ 的顺序操作每个点, 得到一个满足 3-colorable 的图序列: $G_0, G_1, G_2, \cdots, G_n$, 其中 $G_i \to G_{i+1}$ 通过如下的步骤得到:

- 构造图 $G_i^1 = G_i \cup \{(r, v_i), (b, v_i)\}$, $G_i^2 = G_i \cup \{(r, v_i), (g, v_i)\}$, $G_i^3 = G_i \cup \{(g, v_i), (b, v_i)\}$.
- 依次调用 `3Color`$(G_i^j), (j = 1, 2, 3)$.
- 若 `3Color`$(G_i^1) = 1$, 那么 v_i 颜色可以为 g, 令 $G_{i+1} = G_i^1$.
- 若 `3Color`$(G_i^2) = 1$, 那么 v_i 颜色可以为 b, 令 $G_{i+1} = G_i^2$.
- 若上述均不为 1, 那么 v_i 颜色可以为 r, 令 $G_{i+1} = G_i^3$.

最终得到了 G 得一个合法的 3-coloring. 总共需要调用 $O(2n+1) = O(n)$ 次 `3Color`, 符合题意. ◁

Problem 2. (3-Dimensional Matching).
In our class, we introduced how to solve the bipartite matching problem in polynomial time. In particular, the perfect bipartite matching problem can be written in the following way: Suppose we are given two disjoint sets A and B, each of size n, and a set P of pairs drawn from $A \times B$. The goal is to determine whether there exists a set of n pairs in P such that each element in $A \cup B$ is contained in exactly one of these pairs.

As a generalization, consider the following 3-dimensional matching problem: Suppose we are given three disjoint sets A, B, and C, each of size n, and a set T of triples drawn from $A \times B \times C$. The goal is to determine whether there exists a set of n triples in T such that each element in $A \cup B \cup C$ is contained in exactly one of these triples.

(1) Prove that 3-dimensional-matching $\leq_P$ Set-Cover. This implies that 3-dimensional-matching $\in$ NP.

(2) Prove that 3-SAT $\leq_P$ 3-dimensional-matching. Together with (1), this implies that the 3-dimensional-matching problem is NP-complete. ◀

Answer. (1) 如果 $\bigcup T \neq A \cup B \cup C$, 那么显然不存在 3 维匹配. 考虑如下的多项式规约 f:

$$f : \langle S \rangle \to \langle \mathcal{U}, T, n \rangle\,,\ \text{其中} S = \{S_1, \cdots, S_n\} \subseteq T,\ \mathcal{U} = A \cup B \cup C$$

注意到若 $S \in$ 3-dimensional-matching, 那么 S 对应的集合族 $\mathcal{F} = \{S_1, \cdots, S_n\}$ 恰好是 $\mathcal{U}$ 的一个大小为 n 的 Set-Cover. 若 $S \notin$ 3-dimensional-matching, 显然 $\mathcal{U}$ 不存在一个大小至少为 n 的 Set-Cover. 那么

$$f(S) \in \text{3-dimensional-matching} \iff f(S) \in \text{Set-Cover}$$

故 3-dimensional-matching $\leq_P$ Set-Cover, 证毕. 进而 3-dimensional-matching $\in$ NP.

注: 直接从 NP 的定义出发, 考虑证书为 T 的一个 n 元子集 S, 直接验证 S 是否是 3-dimensional-matching 即可. 明显验证时间是多项式的, 且 $|S|$ 是多项式的. 进而有 3-dimensional-matching $\in$ NP.

(2) 考虑如下的规约 g, 对于任意一个 3CNF 公式 φ(n 个变量, m 个子句), 按照如下步骤构造 $g(\varphi)$:

- 对于任意一个变量 x_i, 添加点集 A_i, B_i 和三元组 t_{ij}(令 $k = m$):

$$A_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,2k}\},\ B_i = \{b_{i,1}, b_{i,2}, \cdots, b_{i,2k}\},\ t_{ij} = (a_{i,j}, a_{i,j+1}, b_{i,j})$$

  总共有 $4nk$ 个点, $2nk$ 个三元组.
- 对于任意赋值, 若 $x_i = 1$, 那么将 t_{ij}(j 为奇数) 添加到匹配中. 若 $\neg x_i = 1$, 则将 t_{ij}(j 为偶数) 添加到匹配中. 此时 $\{A_i\}$ 都被添加进了匹配中, $\{B_i\}$ 还剩下 nk 个点未被添加进匹配中.
- 对任意一个子句 $C_j = (x_{j_1} \vee x_{j_2} \vee x_{j_3})$, 添加点集 $N_j = \{c_j, \hat{c}_j\}$ 与三元组 $r_{j_1}, r_{j_2}, r_{j_3}$:

$$\forall i = \{1,2,3\},\ r_{j_i} = (c_j, \hat{c}_j, b_{j_i,2j}),\ \text{若 } x_{j_i} \text{ 为 } x_{j_i} \text{ 的形式.}$$
$$r_{j_i} = (c_j, \hat{c}_j, b_{j_i,2j-1}),\ \text{若 } x_{j_i} \text{ 为 } \neg x_{j_i} \text{ 的形式.}$$

  总共添加了 $2m$ 个点, $3m$ 个三元组.
- 对子句 C_j 的任意赋值, 若 $x_{j_i} = 1$ 则将 r_{j_i} 添加到匹配中 (若有多个为 1, 则任选一个添加). $\{C_j\}$ 都被添加进了匹配中, 另外添加了 m 个 $\{B_i\}$ 中的点.
- 注意到如上操作后, $\{B_i\}$ 还有 $nk-m$ 个点未被添加进匹配中. 为了变成 3-dimensional-matching, 我们额外添加 $nk - m$ 个二元组 $(e_i, \hat{e}_i)$, 每个这样的二元组与 $\{B_i\}$ 中未添加进匹配中的点一一对应. 这样就构成了一个 3-dimensional-matching, 其中:

$$A = \{a_{i,j} \mid j \text{ 为偶数}\} \cup \{c_j\} \cup \{e_j\},\ B = \{a_{i,j} \mid j \text{ 为奇数}\} \cup \{\hat{c}_j\} \cup \{\hat{e}_j\},\ C = \{b_{i,j}\}$$

规约显然是多项式时间的, 下面简要证明规约的正确性:

1.) 若 φ 是可满足的, 那么存在一个赋值使得 φ 中的每一个子句都为真. 对于每个子句, 这里 $|A_i| = |B_i| = 2k$ 保证了不会有 b_{ij} 被重复添加进匹配中. 最后的添加的 $nk - m$ 个二元组保证了 B 中的点都被添加进了匹配中. 故 $g(\varphi)$ 是 3-dimensional-matching 的.

2.) 若 A, B, C 是 3-dimensional-matching 的, 那么 $x_i = 0/1$ 取决于 $(c_j, \hat{c}_j, b_{i,j})$ 中 j 的奇偶性 (偶数为 1, 奇数为 0), 那么由构造可以得到一组赋值使得 φ 为真. 故 $g(\varphi)$ 是可满足的.

综上所述, 3-SAT $\leq_P$ 3-dimensional-matching, 证毕. $\triangleleft$

Problem 3. (Restricted Monotone Satisfiability).
Consider an instance of the Satisfiability problem, specified by clauses $C_1, \ldots, C_k$ over a set of Boolean variables $x_1, \ldots, x_n$. **We say that the instance is *monotone* if each literal is a non-negated variable (i.e., x_i can appear as a literal but $\neg x_i$ cannot)**. Monotone instances are always satisfiable since we can simply set each variable to 1.

For example, suppose we have the three clauses

$$(x_1 \vee x_2), (x_1 \vee x_3), (x_2 \vee x_3)$$

This is monotone, and indeed the assignment that sets all three variables to 1 satisfies all the clauses. But we can observe that this is not the only satisfying assignment; we could also have set x_1 and x_2 to 1, and x_3 to 0. Indeed, for any monotone instance, it is natural to ask how few variables we need to set to 1 in order to satisfy it.

Given a monotone instance of Satisfiability, together with a number k, the problem of *Restricted Monotone Satisfiability* asks: is there a satisfying assignment for the instance in which at most k variables are set to 1? Prove this problem is NP-complete. ◀

Answer. 不妨简记 *Restricted Monotone Satisfiability* 为 RMS.

(1) 首先证明 RMS $\in$ NP. 输入为一个 boolean formula φ 和 k, 考虑证书为一组下标 (不超过 k 个), 长度显然是 $\text{Poly}(|\langle\varphi, k\rangle|)$ 的, 验证机只需先检验 φ 是否为 monotone instance 以及 $k \leq n$, 在检验令证书中的下标对应的变量赋 1, φ 是否可满足即可.

(2) 再证明 Set-Cover $\leq_P$ RMS. 给定全集 $\mathcal{U} = \{c_1, c_2, \cdots, c_t\}$, 子集族 $\mathcal{S} = \{X_1, X_2, \cdots, X_n\}$, 判断是否存在大小不超过 k 的子集族 $\mathcal{S}' \subseteq \mathcal{S}$ 使得 $\bigcup \mathcal{S}' = \mathcal{U}$. 考虑如下规约 $f : \langle\mathcal{U}, \mathcal{S}, k\rangle \to \langle\varphi, k\rangle$.

- 每个 $X_i \in \mathcal{S}$ 对应着 φ 中一个变量 x_i. 每个 $c_i \in \mathcal{U}$ 对应着一个子句 C_i.
- 若 $X_i = \{c_{i,1}, \cdots, c_{i,s}\}$, 则在子句 $\{C_{i,1}, \cdots, C_{i,s}\}$ 中添加变量 x_i.
- 设 $\mathcal{S}' = \{X_{i_1}, \cdots, X_{i_{k'}}\}$, 其中 $k' \leq k$. 则令对应的变量 $x_{i_1}, \cdots, x_{i_{k'}}$ 赋值为 1, 其他为 0.

f 显然是多项式时间的. 下面证明规约的正确性:

1.) 若 $\langle\mathcal{U}, \mathcal{S}, k\rangle \in$ Set-Cover. 那么由于 $\mathcal{S}'$ 不交且 $\bigcup \mathcal{S}' = \mathcal{U}$, 说明赋值 $\{x_{i_1}, \cdots, x_{i_{k'}}\} = \{1\}$ 覆盖了所有的子句, 即 φ 可满足.

2.) 若 $f(\langle\mathcal{U}, \mathcal{S}, k\rangle) = \langle\varphi, k\rangle \in$ RMS. 不妨设一个可满足的赋值为 $\{x_{j_1}, \cdots, x_{j_{k''}}\} = \{1\}$ 且 $k'' \leq k$. 那么由构造对应的 $\mathcal{S}'' = \{X_{j_1}, \cdots, X_{j_{k''}}\}$ 为大小 $k'' \leq k$ 的覆盖, 即 $\langle\mathcal{U}, \mathcal{S}, k\rangle \in$ Set-Cover.

综上所述, RMS $\in$ NP, Set-Cover $\leq_P$ RMS, 故 RMS 是 NP-complete 的. ◁

Problem 4. (Do some Calculus!).
For functions $g_1, \ldots, g_\ell$, we define the function $\max(g_1, \ldots, g_\ell)$ via

$$[\max(g_1, \ldots, g_\ell)](x) = \max(g_1(x), \ldots, g_\ell(x))$$

Consider the following problem. You are given n *piecewise linear, continuous* functions $f_1, \ldots, f_n$ defined over the interval $[0, t]$ for some integer t. You are also given an integer B. You want to decide: do there exist k of the functions $f_{i_1}, \ldots, f_{i_k}$ so that

$$\int_0^t [\max(f_{i_1}, \ldots, f_{i_k})](x)\ dx \geq B?$$

Prove that this problem is NP-complete.

Note: A piecewise continuous function is a function that is continuous except at a finite number of points in its domain. ◀

Answer. 记这个问题为 Calc.

(1) 首先证明 Calc $\in$ NP. 输入为 n 个函数 $f_1, \cdots, f_n$ 和整数 B, k, 考虑证书为一组下标 (不超过 k 个). 注意到连续的分段线性函数的 max 还是一个连续的分段线性函数, 故只需要验证:

$$\int_0^t \left[\max\left(f_{i_1}, \ldots, f_{i_k}\right)\right](x)\ \mathrm{d}x \geq B$$

其中 $i_1, \cdots, i_k$ 为证书中的下标. 验证机只需计算上述积分, 验证是否大于等于 B, 是多项式时间的.

(2) 下面证明上一道题中的 RMS $\leq_P$ Calc. 设 $\forall \varphi \in$ RMS, 设 φ 有 n 的变量 $x_1, \cdots, x_n$ 和 m 个子句 $C_1, \cdots, C_m$. 考虑如下的多项式规约:

$$f_i(x) = \begin{cases} 1, & \forall j \leq x < j+1, \text{ if } x_i \in C_j, \forall j \in \mathbb{N}^*, \\ 0, & \text{Otherwise}. \end{cases}$$

那么若 φ 的一个可满足的赋值为 $\{x_{i_1}, \cdots, x_{i_k}\}$, 那么:

$$\left[\max(f_{i_1}, \cdots, f_{i_k})\right](x) = \begin{cases} 1, & \forall j \leq x < j+1, \text{ if } x_{i_1}, \cdots, x_{i_k} \in C_j, \forall j \in \mathbb{N}^*, \\ 0, & \text{Otherwise}. \end{cases}$$

故令 $B = m$, 那么 $\varphi \in$ RMS $\iff f_1, \cdots, f_n \in$ Calc.

综上所述, Calc 是 NP-complete 的. 证毕. $\lhd$