Assignment 1

题目: Chapter04最后一页

1、1) VC实例: 无向图 $G=(V,E)$, 找最小顶点子集 $C\subseteq V$ 能覆盖所有边.
SC实例: 全集 U, $S=\{S_1,\cdots,S_m\}$, $S_i\subseteq U$, 求最小子集 $C\subseteq S$ 覆盖 U.
可如下构造 SC:
$U=E$, $S=\{S_v \mid v\in V\}$, $S_v=\{e\in E \mid e$ 连在 v 上$\}$
该映射的时间复杂度为 $O(|V|+|E|)$.

2) 已知 VC 是 NP-hard, 假设 SC 存在多项式时间精确算法 A,
可如下构造解 VC 的算法 A':
1. 输入 $G=(V,E)$
2. 用1)构造 SC 实例 (U,S)
3. 调用 A 解出最小集合覆盖 C
4. 输出 $C=\{v \mid S_v\in C\}$

Step 1、2、4 均为多项式时间, Step 3 调用一次 Oracle.
因此 VC 可 Cook 归约到 SC, 因此 SC 是 NP-hard.

2、使用优先队列优化

Algorithm:
1. Initialization:
a. 计算所有顶点 degree deg[v]
b. 所有顶点加入最大堆
c. initalize covered[e] = false
2. loop 到无边覆盖
a. 取出堆顶元素 v, 将 v 加入覆盖集 C
b. 遍历 v 的邻边 e = (u, v)
if covered[e] = false:
covered[e] = true, deg[v] -= 1, 更新 u 在堆中的位置
c. deg[v] = 0 并从堆中移除

No.
Date

每个顶点和边都处理1次，堆操作每次均为 $O(\log|V|)$

因此复杂度为 $O((|V|+|E|)\log|V|)$

3、举出以下反例：

$G=(V,E)$，$V=\{s, a_1, a_2, \cdots, a_m, b_m, b_{m-1}, \cdots, b_1, t\}$

边权：$W_{s,a_1} = W_{a_i,a_{i+1}} = W_{a_m,b_m} = W_{b_i,b_{i-1}} = W_{b_1,t} = W_{a_i,b_i} = 1$

$W_{s,b_1} = 1+\varepsilon$ ，$W_{s,t} = L$（任意大）

NN算法：$s \to a_1 \to a_2 \to \cdots \to a_m \to b_m \to b_{m-1} \to \cdots \to b_1 \to t \to s$

总代价 $\approx 2m+L$

最优：用 $a_i \to b_i$ 交替走，总代价 $\approx m+$ 常数

m 很大时，$\frac{2m+L}{m}$ 可任意大.

AI使用：deepseek

Q1. 2) 提示思路

Q3. 给出反例