第十四周作业-solution

LECTURER: 杨启哲 LAST MODIFIED: 2023 年 12 月 26 日

1. (教材习题 9.6) 设计一个非确定算法来求解旅行商问题。

解答.

注意到旅行商问题的叙述如下:

给定 n 个城市和一个 $n\times n$ 的距离矩阵 D, 其中 $D[i,j]$ 表示从城市 i 到城市 j 的距离, 旅行商要从某个城市出发, 经过每个城市恰好一次, 最后回到出发的城市, 问旅行商的最短路线是什么?

我们考虑其判定问题形式:

给定 n 个城市和一个 $n\times n$ 的距离矩阵 D 以及一个正整数 K, 其中 $D[i,j]$ 表示从城市 i 到城市 j 的距离, 旅行商要从某个城市出发, 经过每个城市恰好一次, 最后回到出发的城市, 问是否存在一条长度不超过 K 的路线?

一个求解旅行商问题判定版本的非确定算法如下:

判定版本的旅行商问题 DTSP(G,K)

- **猜测阶段** 猜测长度为 n 一个顶点序列.
- **验证阶段** 验证该序列是否为一个满足要求的路径, 即:
  - 该序列是否是一条路径。
  - 该序列是否恰好包含 n 个顶点。
  - 如果是一条路径, 长度是否不超过 K。

  如果满足上述条件, 则接受, 否则拒绝。

对于最优化版本, 我们可以使用通用的策略, 即令 $K=\sum_{i,j}D[i,j]$, 不停的二分调用判定版本的非确定性算法, 直到找到最小的 K' 使得该算法接受。具体如下:

求解旅行商问题 TSP(G)

- 调用 DTSP(G,K), 其中 $K=\sum_{i,j}D[i,j]$, 如果返回 $False$, 则拒绝。
- 否则令 K' $=\frac{K}{2}$, 继续调用 DTSP(G,K')。
- 如果返回 $True$, 则继续调用 DTSP(G,$\frac{K'}{2}$); 否则调用 DTSP(G,$\frac{K'+K}{2}$), 直到找到最小的 K 使得该算法接受。

□

2. (教材习题 9.7) 设 Π_1 和 Π_2 是两个问题, 且 $\Pi_1\propto_{poly}\Pi_2$。假设问题 Π_2 能在 $O(n^k)$ 时间内解出, 并且归约可以在 $O(n^j)$ 时间内完成。证明问题 Π_1 可以在 $O(n^{jk})$ 时间内求解出。

解答. 令 I_1 是 Π_1 的一个实例，其规模为 n。由于 $\Pi_1 \propto_{polu} \Pi_2$，并且归约可以在 $O(n^j)$ 的时间内完成，从而 I_1 可以通过归约转化为 Π_2 的一个实例 I_2，满足：

- I_2 的规模为 $O(n^j)$。
- I_1 可以被求解当且仅当 I_2 可以被求解。

由于 Π_2 可以在 $O(n^k)$ 内解出，因此解决 I_2 需要 $O((n^j)^k) = O(n^{jk})$ 的时间内解出，从而解决 I_1 可以在 $O(n^{jk}) + O(n^j) = O(n^{jk})$ 的时间内解出，即问题 Π_1 可以在 $O(n^{jk})$ 时间内求解出。 □

3. (2$-SAT$ 问题)2$-SAT$ 问题是在 SAT 问题基础上增加每个子句至多包含两个文字的限制，即每个子句形如 $(c \vee c')$，这里的 c 为 x_i 或者 $\neg x_i$。我们可以发现，增加了这样的限制后，该问题便是可以高效解决的了。请给出一个多项式时间的算法赖求解 2$-SAT$ 问题。

Hint: 可以考虑将其转换成一张图

解答. 我们将其转换成一张图，给定一个具有 n 个变量的 2$-SAT$ 公式 f，其有 m 个子句 C_i，我们构造如下的图 $G = (V, E)$：

- $V = \{v_x, v_{\neg x} \mid x$ 是 f 的一个变量$\}$.
- E 中包含这样的边：
  - 对于每个子句 C_i，如果 $C_i = (c \vee c')$，则 E 中包含边 $(v_{\neg c}, v_{c'})$ 和 $(v_{\neg c'}, v_c)$。

图 G 具有如下的性质：如果存在一条 v_x 到 v_y 的路径，则必然存在一条 $v_{\neg y}$ 到 $v_{\neg x}$ 的路径。我们断言，如果 G 中存在某个强连通分量 C，使得 C 中同时包含了 v_x 和 $v_{\neg x}$，则 f 不可满足；否则 f 可满足。

事实上，如果 C 中同时包含了 v_x 和 $v_{\neg x}$，则存在一条从 v_x 到 $v_{\neg x}$ 的环。反设 f 是可满足的，考虑任何一个使其可满足的赋值，其一定将 x 赋值为 $True$ 或者 $False$，不妨设为 $True$，令 v_x 到 $v_{\neg x}$ 的环上的顶点依次如下：

$$v_x, v_{x_1}, \ldots v_{x_k}, v_{\neg x}, v_{y_1}, \ldots v_{y_l}, v_x$$

注意到 $(v_x, v_{x_1}) \in E$，由定义：$\neg x \vee x_1$ 是 f 的一个子句，由 f 是可满足的，可得 $\neg x \vee x_1$ 为真，从而 x_1 必须为真。同理可得 $x_2, \ldots, x_k$ 都为真，从而 $v_{\neg x}$ 到 v_x 的环上的顶点都为真，从而 $\neg x$ 为真，与 x 为真矛盾，因此 f 不可满足。

另一方面，如果 G 中不存在一个强连通分量同时包含了 v_x 和 $v_{\neg x}$，则我们可以通过如下的方式构造一个相应的使 f 满足的赋值：

- 对于每个变元 x，如果存在 v_x 到 $v_{\neg x}$ 的路径，则令 x 为 $False$；如果存在 $v_{\neg x}$ 到 v_x 否则令 x 为 $True$；我们记这一部分为真的文字为 $Var = \{x_1, \ldots, x_k\}$(也就是说如果 x 赋值为 0，我们就添加 $\neg x$ 进去)
- 对于剩余的变元 x，如果存在 Var 中对应的点到 v_x 的路径，则赋值为 $True$; 如果存在 Var 中对应的点到 $v_{\neg x}$ 的路径，则赋值为 $False$。
- 若还剩余变元，则对其随意赋值。

我们证明上述赋值是使 f 满足的。事实上我们只需要证明上述赋值时不会矛盾的即可，因为其满足了所有边对应的析取子句。这由下述事实保证：

- 如果在第一步中，x 和 y 被赋值，不妨设为 $False$ 和 $True$。如果此时存在一条 v_y 到 v_x 的路径，则必然存在一条 $v_{\neg x}$ 到 $v_{\neg y}$ 的路径，从而存在一个 $v_x \to *v_{\neg x} \to^* v_{\neg y} \to^* v_y \to^* v_x$ 的环，矛盾。

- 如果在第二步中，对于某个变元 z；其从第一步赋值的变元 x,y 出发会得到不同的赋值结果，即令 x 和 y 分别被赋为 $False$ 和 $True$，则存在如下的两条路径：

  $$v_x \to^* v_{\neg x} \to^* v_{\neg z},\ v_{\neg y} \to^* v_y \to^* v_z$$

  则由上述的性质，我们存在如下的圈：

  $$v_x \to^* v_{\neg x} \to^* v_{\neg z} \to^* \ v_{\neg y} \to^* v_y \to^* v_z \to^* v_x$$

注意到构造 G 的过程是多项式的，而求解强连通分量的过程也是多项式的，因此我们可以在多项式时间内求解 $2\text{−}SAT$ 问题。 □