NP完全问题

Jun　Wu

wujun@yzu.edu.cn

May 16, 2018

决策问题vs. 优化问题

- **决策问题：** 问题的答案是"是"或"否"的一类问题。
  - 问题的提法一般是：给定$\cdots$，问是否$\cdots$？
  - 例如：给定布尔表达式F，问是否存在变量赋值使得F成真？
- **优化问题：** 问题有许多可行解，每个可行解关联一个值，优化问题要求找出具有最优关联值的可行解。
  - 问题的提法一般是：给定$\cdots$，求最$\cdots$？
  - 例如：给定$G = (V, E)$和边的权函数c，以及点$s, t \in V$，求s到t的最短路径？
- 复杂性理论一般只考虑决策问题。

为什么决策问题

- 任何优化问题均有相应的决策问题版本。
- 最短路问题：给定$G=(V,E)$和边的权函数c，以及点$s,t\in V$，问是否存在s到t的长度K以内的路径？
- 通常，相应的决策问题能有效求解，也能有效计算其优化问题。
- 若优化问题能被有效求解，相应的决策问题必然可以有效求解。
- 我们将一个具体问题的输入称作该问题的一个实例，如上述最短路问题中的G,c,s,t,K。
- 通常，我们将问题视作该问题所有可能的实例的集合。
- 一个决策问题自然地将所有实例分成了两类。
- 因此，可以形式化的定义：问题=所有答案为Yes的实例集合。

P类与NP类

- 容易的问题类：**P类**：可以在多项式时间内求解的问题类。
- 困难的问题类？
- 可能包含困难问题的类：**NP类**：可以在多项式时间内验证问题解的问题类。显然，$P \subseteq NP$。

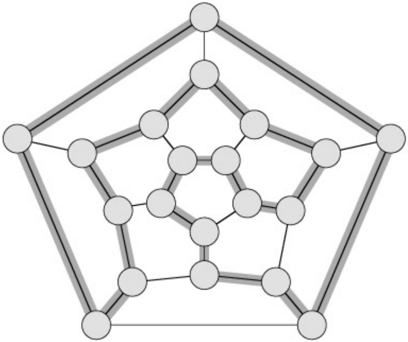


图: Hamiltonian cycles

最深刻的问题

$$\mathrm{P} \neq \mathrm{NP}?$$

Definition 1

决策问题L_1和L_2，存在算法P将任意L_1的实例转换为L_2的实例。若P满足：

1. P是多项式时间算法；
2. $\forall \alpha \in L_1 : L_1(\alpha) = \text{Yes} \Leftrightarrow L_2(P(\alpha)) = \text{Yes}$，

则称L_1可以多项式时间规约到L_2，记作$L_1 \leq_P L_2$。

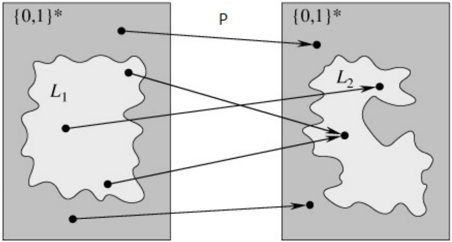
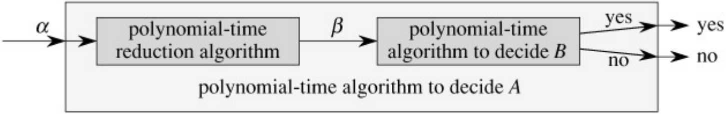


图: 多项式时间规约示意图

多项式时间归约的意义

- 多项式时间归约提供了比较问题难易的手段(从复杂性角度)；
- 若$L_1 \le_P L_2$，那么
- L_2存在多项式时间算法则L_1必存在多项式时间算法；
- L_1不存在多项式时间算法则L_2必不存在多项式时间算法；
- 即L_1比L_2容易。



NP完全问题

Definition 2 (NP-complete)

若问题L满足：

1. $L \in NP$，并且
2. $\forall L' \in NP : L' \leq_P L$。

则问题L是*NPC*的，若L只满足条件*2*则称问题是*NP-hard*的。

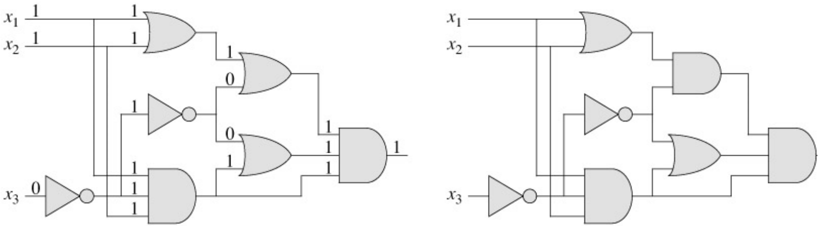
- NP完全问题，从复杂性角度，是所有NP类问题中最困难的问题；
- NP类并不涵盖所有可能的问题，不在NP类中的困难问题是NP-hard问题。
- NP完全问题可以不止一个。

Cook定理

Theorem 3 (Cook定理)

CIRCUIT-SAT问题是NPC问题。

- Cook定理给出第一个NPC问题，从而打开了计算复杂性理论的大门。



(a) (b)

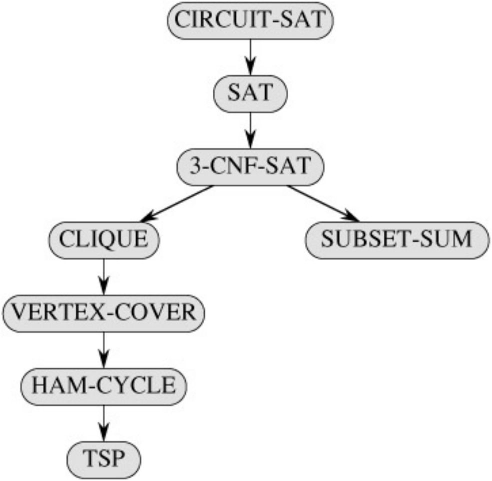
NP完全问题的证明方法

Lemma 4

$L' \leq_P L$，若L'是*NPC*，那么L是*NP-hard*问题；且若$L \in NP$，则L是*NPC*。

- 证明问题L是NPC的步骤：
  1. 证明$L \in NP$；
  2. 找一已知的NPC问题L'，设计归约算法P；
  3. 证明P是多项式时间的；
  4. 证明$\forall \alpha \in L_1 : L_1(\alpha) = \text{Yes} \Leftrightarrow L_2(P(\alpha)) = \text{Yes}$。

路线图


CIRCUIT-SAT
SAT
3-CNF-SAT
CLIQUE
SUBSET-SUM
VERTEX-COVER
HAM-CYCLE
TSP

SAT和3-SAT

- **SAT问题：**
- 实例：布尔表达式ϕ；
- 问题：是否存在变量的赋值使得ϕ的值为真？
- 例，$\phi = ((x_1 \rightarrow x_2) \vee \neg((\neg x_1 \leftrightarrow x_3) \vee x_4)) \wedge \neg x_2$
- SAT问题是NPC，Circuit-SAT$\leq_P$SAT。

- **3-SAT问题：**
- 实例：3-CNF表达式ϕ；
- 问题：是否存在变量的赋值使得ϕ的值为真？
- 例，$\phi = (x_1 \vee \bar{x}_3 \vee \bar{x}_2) \wedge (x_3 \vee x_2 \vee x_4) \wedge (\bar{x}_1 \vee \bar{x}_3 \vee \bar{x}_4)$

Theorem 5

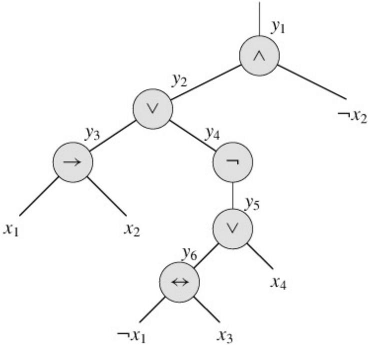
3-SAT是NPC问题。

证明思路:

- 3-SAT$\in NP$。
- SAT$\leq_P$3-SAT。
  - 给定SAT实例ϕ，构造ϕ的语法树;
  - 根据语法树，将ϕ等价地变换为若干子句的合取连接ϕ';
  - 利用真值表，去除ϕ'中不符合CNF范式的运算符，得ϕ'';
  - 将ϕ''不足3个字的子句变换为每个子句3个字的形式ϕ''';
  - 证明ϕ可满足当且仅当ϕ'''可满足。

ϕ的语法树和ϕ'

$$\phi = ((x_1 \to x_2) \vee \neg((\neg x_1 \leftrightarrow x_3) \vee x_4)) \wedge \neg x_2$$



$$\begin{aligned}
\phi' \quad \equiv \quad y_1 \quad & \wedge \quad (y_1 \leftrightarrow (y_2 \wedge \bar{x}_2)) \\
& \wedge \quad (y_2 \leftrightarrow (y_3 \vee y_4)) \\
& \wedge \quad (y_3 \leftrightarrow (x_1 \to x_2)) \\
& \wedge \quad (y_4 \leftrightarrow \bar{y}_5) \\
& \wedge \quad (y_5 \leftrightarrow (y_6 \vee x_4)) \\
& \wedge \quad (y_6 \leftrightarrow (\bar{x}_1 \leftrightarrow x_3))
\end{aligned}$$

真值表和ϕ''

设$\phi_1' = (y_1 \leftrightarrow (y_2 \wedge \bar{x}_2))$。

y_1	y_2	x_2	$(y_1 \leftrightarrow (y_2 \wedge \neg x_2))$
1	1	1	0
1	1	0	1
1	0	1	0
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	1
0	0	0	1

$$\begin{aligned}\neg\phi_1' \quad \equiv \quad & (y_1 \wedge y_2 \wedge x_2) \\ \vee \quad & (y_1 \wedge \bar{y}_2 \wedge x_2) \\ \vee \quad & (y_1 \wedge \bar{y}_2 \wedge \bar{x}_2) \\ \vee \quad & (\bar{y}_1 \wedge y_2 \wedge \bar{x}_2)\end{aligned}$$

根据德摩根律，有

$$\begin{aligned}\phi \quad \equiv \quad & (\bar{y}_1 \vee \bar{y}_2 \vee \bar{x}_2) \\ \wedge \quad & (\bar{y}_1 \vee y_2 \vee \bar{x}_2 \\ \wedge \quad & (\bar{y}_1 \vee y_2 \vee x_2) \\ \wedge \quad & (y_1 \vee \bar{y}_2 \vee x_2)\end{aligned}$$

- ϕ''已基本是3-CNF范式公司了，唯一区别
- 有部分子句不足3个字。
- 如y_1。通过增加变量p, q，将其等价变换为

$$\begin{aligned} y_1 &\equiv (y_1 \vee p \vee q) \\ &\wedge (y_1 \vee p \vee \bar{q}) \\ &\wedge (y_1 \vee \bar{p} \vee q) \\ &\wedge (y_1 \vee \bar{p} \vee \bar{q}) \end{aligned}$$

- 由于三步变换皆是恒等变换，因此ϕ可满足当且仅当ϕ'''可满足；
- 三步均是线性时间的变换。

$\square$

Definition 6

CLIQUE=$\{< G, k > | G$中存在规模为k的团$\}$。

Definition 7

CLIQUE$=\{< G, k > |G$中存在规模为k的团$\}$。

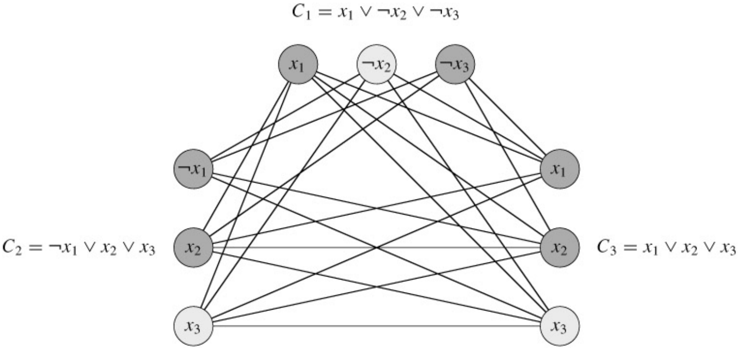


图: 3-SAT$\leq_P$CLIQUE

最小顶点覆盖

Definition 8

VERTEX-COVER=$\{< G, k > |G$中具有规模为k的顶点覆盖$\}$。

Definition 9

VERTEX-COVER=$\{< G, k > | G$中具有规模为k的顶点覆盖$\}$。

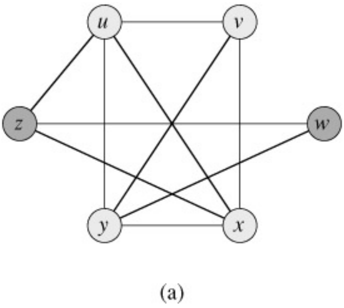


图: CLIQUE$\leq_P$VERTEX-COVER

汉密尔顿回路与旅行商问题

Definition 10

HAM-CYCLE$=\{G|G$是汉密尔顿的$\}$。

- VERTEX-COVER$\leq_P$HAM-CYCLE。

Definition 11

TSP$=\{< G, c, k > |G$是完全图，c是边的权值函数且G中存在长度为k的汉密尔顿回路$\}$.

- HAM-CYCLE$\leq_P$TSP。

子集和问题

Definition 12

SUBSET-SUM$=\{< S, t > |\exists S' \subseteq S : t = \sum_{s\in S'} s\}$。

Definition 13

SUBSET-SUM$=\{< S,t > | \exists S' \subseteq S : t = \sum_{s\in S'} s\}$。

		x_1	x_2	x_3	C_1	C_2	C_3	C_4
v_1	=	1	0	0	1	0	0	1
v_1'	=	1	0	0	0	1	1	0
v_2	=	0	1	0	0	0	0	1
v_2'	=	0	1	0	1	1	1	0
v_3	=	0	0	1	0	0	1	1
v_3'	=	0	0	1	1	1	0	0
s_1	=	0	0	0	1	0	0	0
s_1'	=	0	0	0	2	0	0	0
s_2	=	0	0	0	0	1	0	0
s_2'	=	0	0	0	0	2	0	0
s_3	=	0	0	0	0	0	1	0
s_3'	=	0	0	0	0	0	2	0
s_4	=	0	0	0	0	0	0	1
s_4'	=	0	0	0	0	0	0	2
t	=	1	1	1	4	4	4	4

3-SAT$\leq_P$SUBSET-SUM

$$C_1 = (x_1, \bar{x}_2, \bar{x}_3)$$

$$C_2 = (\bar{x}_1, \bar{x}_2, \bar{x}_3)$$

$$C_3 = (\bar{x}_1, \bar{x}_2, x_3)$$

$$C_4 = (x_1, x_2, x_3)$$

关于P和NP的看法

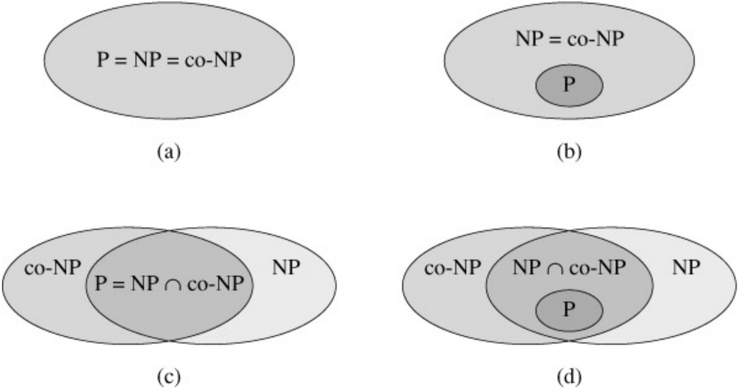


图: 四种可能的关系