NP-трудные задачи

Задача поиска

Пространство поиска $\sim 2^n$

$\equiv$ Задача поиска (search problem)

$C(I, S)$ – предикат от двух параметров

I – (instance) условие

S – (solution) решение

Вход: I $\qquad$ $C(I,S) = true \iff$

Выход: S $\qquad$ S – решение для I

Мы будем рассматривать задачи поиска, где C – вычисляется за $O(poly(|I| + |S|))$

$\equiv$ Задача разрешения (decision problem)

$C(I, S)$ – —"—

Вход: I

Выход: $\exists ?\, S : C(I, S) = true$

$\equiv$ Задача выполнимости (Satisfiability) (SAT)

Вход: формула в КНФ

$$\varphi = (x_1 \vee \overline{x_2} \vee x_3) \wedge (x_4 \vee x_5 \vee x_6 \vee \overline{x_1}) \wedge \ldots \wedge (x_2 \vee \overline{x_4})$$

переменная $\qquad$ литерал $\qquad$ дизъюнкт / клоз (clause)

(search): найти значения $\underline{x_1 \ldots x_n}$:
выполняющий набор

$\varphi(x_1 \ldots x_n) = true$

(decision): $\exists ?\ x_1 \ldots x_n : \varphi(x_1 \ldots x_n) = true$

Лемма: Можно решить decision problem для SAT за полиномиальное время $\Leftrightarrow$ можно решить и search problem за полиномиальное время.

$\Leftarrow$ очевидно

$\Rightarrow$ $\exists A: A(\varphi) = 1 \Leftrightarrow \exists x_1 \ldots x_n : \varphi(x_1 \ldots x_n) = true$

(A - полиномиальный алгоритм)

B: $A(\varphi) \to 1$

$\searrow 0 \Rightarrow B(\varphi) =$ нет решения

$A(\varphi[x_1 = true]) = 1 \to A(\varphi[x_1 = true, x_2 = false])$

$A(\varphi[x_1 = false])$

$A(\varphi[x_1 = true, x_2 = true])$

$\downarrow$

...

Время работы B:

$time(B(\varphi)) = time(A(\varphi)) \cdot n$

NB: алгоритм - эффективный $\Leftrightarrow$ работает за $O(poly(n)) = n^{O(1)}$.

Что известно про SAT?

- Не умеют решать быстрее $O(2^n poly(n))$
- Умеют решать K-SAT быстрее: $O(2^{\varepsilon n})$ $\varepsilon < 1$ при $K \geq 3$
- При $K = 2$ - решается за $O(|\varphi|)$
- Хорновские формулы:

в ∀ дизъюнкте ≤ 1 переменной без отрицания

за $O(|\varphi|)$

TSP (travelling salesman problem, коммивояжёра)

Вход: полный взвешенный граф

Выход: обход всех всех вершин графа минимальной стоимости

// Аналог: MST

Hamiltonian Cycle (Ham, Rudrata cycle)

Вход: граф

Выход: циклический обход всех вершин графа (∀ вершина только 1 раз)

~ Hamiltonian Path

// Euler cycle

Balanced Cut // Min Cut

Вход: взвешенный граф

Выход: Минимальный сбалансированный разрез:

$$V = C \sqcup \overline{C}$$

$$\frac{1}{3}|V| \leq |C| \leq \frac{2}{3}|V|$$

Integer Linear Programming // LP

Вход: A, b, c

Выход: x : $Ax \leq b$, $c^T x \to \max$

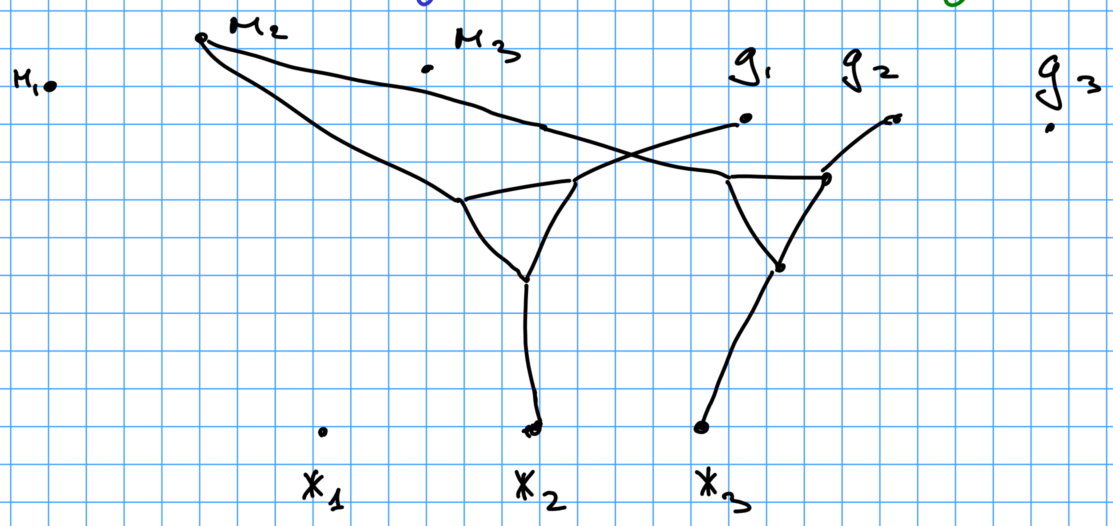
$x_i \in \mathbb{Z}$

Zero-one equation (ZOE)

Вход: $A \in \{0,1\}^{n \times n}$

Выход: x : $Ax = \bar{1}$, $x \in \{0,1\}^n$

3D-matching // Matching



Вход: мультиграф с 3-рёбрами
Выход: выдать n непересек. рёбер.

Independent Set // IS on trees

Вход: граф, число k
Выход: независимое мн-во размера k
(нет рёбер м/у верш. этого мн-ва)

Vertex Cover // VC on trees

Вход: граф, число k
Выход: вершинное покрытие размера k
($\forall$ребро имеет один из концов в этом множестве)

Set Cover

Вход: $\{S_i\} : \bigcup S_i = U$, k
Выход: k S_i : $\bigcup\limits_{j=1}^{k} S_{i_j} = U$

Clique:

Вход: граф, число k
Выход: клика размера k

Longest Path // Shortest Path

Вход: граф, вершины s, t, число k
Выход: путь $s \to t$ длины k

Задача о рюкзаке // Упакующий рюкзак

- . . -

SubSet Sum

Вход: $U \subset \mathbb{Z}$, k

Выход: $M \subseteq U$: $\sum_{a \in M} a = k$

Graph Coloring

Вход: граф, число k

Выход: раскраска графа

P vs NP

≡ P – класс задач поиска, решаемых за $n^{O(1)}$ ($\exists$ эффективный алгоритм)

≡ NP – класс задач поиска, у которых предикат вычисляется за $n^{O(1)}$

Основной вопрос: $P \stackrel{?}{=} NP$

Известно: $P \subset NP$

Схожий вопрос: $coNP \stackrel{?}{=} NP$

$NP \neq coNP \Rightarrow P \neq NP$

Утв: Все упомянут. задачи из класса NP

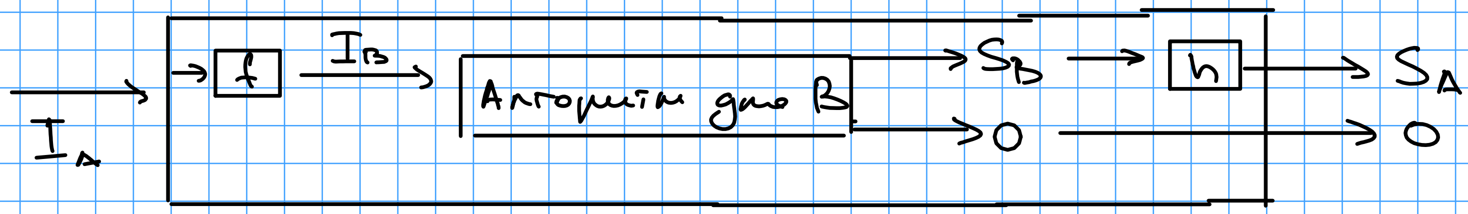
Сведе́ние

$A \to B$ // сведение по Карпу, many-one сведение

$\exists$ полин. вычисл. функция:

$f: \{I_A\} \to \{I_B\}$

$h: \{S_B\} \to \{S_A\}$



1. $\forall I_A \quad \exists S_A: C_A(I_A, S_A) \Leftrightarrow \exists S_B: C_B(f(I_A), S_B)$
2. $\forall I_A \ \forall S_B: C_B(f(I_A), S_B) \Rightarrow C_A(I_A, h(S_B))$

Св-ва:

1. B решается за $n^{O(1)}$ $\Rightarrow$ A решается за $n^{O(1)}$
2. $A \to B, \ B \to C \Rightarrow A \to C$

≡ NP-трудная задача

Задача к которой сводится $\forall$ задача из класса NP

≡ NP-полная задача

NP-трудная задача из класса NP (класс NPC)

Бывают сложные задачи в NP, про которые не известно, что они NPC

- Factoring
  Вход: n
  Выход: $p, q : p \cdot q = n, \quad p, q \geq 1$
- Graph Isomorphism
  Вход: G_1, G_2
  Выход: π – перест. вершин: $\pi(G_1) = G_2$

Сведение:

SAT $\to$ 3SAT

SAT: $(x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5 \dots \vee x_k) \wedge \dots$

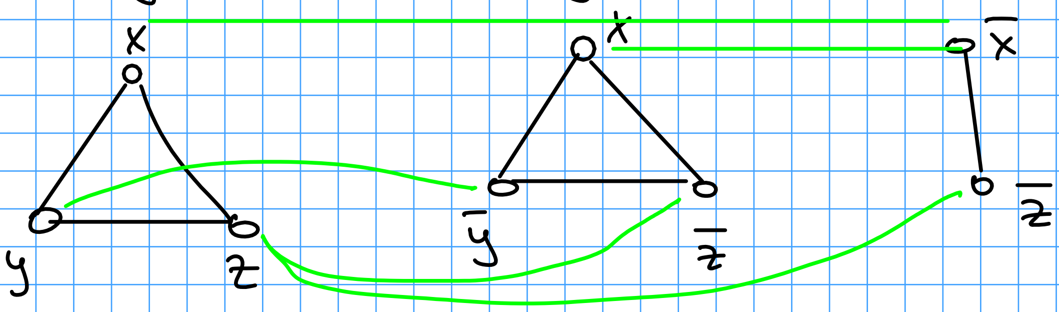
3SAT: $(x_1 \vee x_2 \vee y_1) \wedge (\bar{y}_1 \vee x_3 \vee y_2) \wedge (\bar{y}_2 \vee x_4 \vee y_3) \wedge \dots$
$\wedge \dots (\bar{y}_{k-4} \vee x_{k-2} \vee y_{k-3}) \wedge (\bar{y}_{k-3} \vee x_{k-1} \vee x_k)$

$\sim k$ клозов

описание f

3SAT $\to$ IS

$(x \vee y \vee z) \wedge (x \vee \bar{y} \vee \bar{z}) \wedge (\bar{x} \vee \bar{z})$



Ищем IS размера # клозов