Структурная теория сложности

Эдуард Алексеевич Гирш

http://logic.pdmi.ras.ru/~hirsch

ПОМИ РАН

16 ноября 2008 г.

IP = PSPACE

- Простое включение **IP** $\subseteq$ **PSPACE**:

  если перебирать все возможные ответы provera и все возможные случайные строки verifiera, то просто выяснить “$\exists$ док-во Pr{verifier примет} такая-то”.

- Трудное включение **PSPACE** $\subseteq$ **IP**:

  достаточно построить интерактивный протокол для **PSPACE**-полной задачи `QBF`.

Интерактивный протокол для QBF

Арифметизация

формула	$\mapsto$	многочлен
`False`	$\mapsto$	0
`True`	$\mapsto$	1
$\alpha \wedge \beta$	$\mapsto$	$\alpha \cdot \beta$
$\neg\alpha$	$\mapsto$	$1 - \alpha$
$\alpha \vee \beta$	$\mapsto$	$1 - (1-\alpha)(1-\beta) =: \alpha \odot \beta$

Операторы, соответствующие кванторам, преобразуют многочлены:

$$\begin{aligned}(A_x P)(\ldots) &= P(0,\ldots) \cdot P(1,\ldots),\\ (E_x P)(\ldots) &= P(0,\ldots) \odot P(1,\ldots);\end{aligned}$$

дополнительный оператор линеаризации:

$$(L_x P)(x,\ldots) = P \mod (x^2 - x).$$

Итак, надо доказать

$$q_{x_1}^{(1)} \quad L_{x_1} \quad q_{x_2}^{(2)} \quad L_{x_1} L_{x_2} \quad q^{(x_3)} \quad \ldots \quad q_{x_n}^{(n)} \quad L_{x_1} \ldots L_{x_n} \quad P(x_1 \ldots x_n) = 1.$$

Интерактивный протокол для QBF

Протокол

Пусть d — размер исходной формулы.
Рекурсивный протокол для доказательства

$$q_{x_i} q'_{\dots} \dots q''_{\dots} P(x_1, \dots, x_{i-1}, x_i \dots x_n)|_{x_1=r_1,\dots,x_{i-1}=r_{i-1}} = c :$$

Обозначим

$$R(x_1, \dots, x_i) = q'_{\dots} \dots q''_{\dots} P(x_1, \dots, x_i).$$

Prover даёт коэффициенты этого многочлена $s(x_i)$ степени не выше d.

$q = A$. Если $s(0)s(1) \neq c$, verifier отвергает.
В противном случае рекурсивно проверяет
$R(r_1 \dots r_{i-1}, r_i) = s(r_i)$ для случайного r_i.

$q = E$. Аналогично, с проверкой $s(0) \odot s(1) = c$.

Вер. на одном шаге $\leq \frac{d}{\text{размер поля}}$.
Итого $\leq \frac{1}{d}$ для поля размера $\geq d^4$.

Интерактивный протокол для QBF

Протокол

Пусть d — размер исходной формулы.
Рекурсивный протокол для доказательства

$$q_{x_i} q'_{\dots} \dots q''_{\dots} P(x_1, \dots, x_{i-1}, x_i \dots x_n)|_{x_1=r_1,\dots,x_{i-1}=r_{i-1}} = c :$$

Обозначим

$$R(x_1, \dots, x_i) = q'_{\dots} \dots q''_{\dots} P(x_1, \dots, x_i).$$

Prover даёт коэффициенты этого многочлена $s(x_i)$ степени не выше d.

$q = L$. Аналогично, но r_i уже есть!
Verifier проверяет $s(0) + (s(1) - s(0))r_i = c$ и рекурсивно проверяет, что $R(r_1, \dots, r_{i-1}, r'_i) = s(r'_i)$ для нового случайного r'_i.

Вер. на одном шаге $\leq \frac{d}{\text{размер поля}}$.
Итого $\leq \frac{1}{d}$ для поля размера $\geq d^4$.

Лемма

За полин. время $F \mapsto F_1, \ldots, F_m$, т.ч.

- F выполнима $\Rightarrow$ какая-то F_i одновыполнима с вер. $\geq 1/2$;
- F невыполнима $\Rightarrow$ все F_i невыполнимы.

Leslie Valiant, Vijay Vazirani

Лемма

За полин. время $F \mapsto F_1, \ldots, F_m$, т.ч.

- F выполнима $\Rightarrow$ какая-то F_i одновыполнима с вер. $\geq 1/2$;
- F невыполнима $\Rightarrow$ все F_i невыполнимы.

- Пусть $a^{(1)}, \ldots, a^{(D)}$ — все вып. наборы F. Оставим один!
- $x \in \{0,1\}^n \sim x \in [0\,..\,2^n - 1]$.
- $F(x) \mapsto F(x) \wedge (x \mod p = r)$.
- $i \in [0\,..\,n]$ — угадаем $i = \lceil \log_2 D \rceil$ с вер. $1/(n+1)$.
- Случайные $p \in [1..b]$ и $r \in [0..b-1]$, где $b = 4 \cdot 2^i n^2$.
- p — хорошее для $a^{(k)}$, если оно простое и $\forall j \neq k\ a^{(k)} \not\equiv a^{(j)} \pmod p$.
- Для данного $a^{(k)}$ плохих простых $\leq n \cdot D$, а всего простых $0.92129 \cdot b/\ln b > b/\log_2 b \geq 2^{i+1} n$. Итого $\geq 2^i n$ хороших.
- Хорошая пара (p, r) отличает хоть какой-то набор от других. Хороших пар $\geq D \cdot 2^i n$.
- А всего пар b^2. Вероятность выбрать хорошую $\geq 1/(32n^4)$ (с учётом i).

Probabilistically Checkable Proofs (PCP)

- Доступ к доказательству π по адресу (номер бита $\mapsto$ бит).
- Язык L вероятностно проверяем, если $\exists$ вер. полин. A:

$$\begin{aligned} x \in L &\Rightarrow \exists \pi \ \Pr\{A^{\pi}(x) = 1\} = 1, \\ x \notin L &\Rightarrow \forall \pi' \ \Pr\{A^{\pi'}(x) = 1\} < \frac{1}{2}. \end{aligned}$$

- $L \in \mathbf{PCP}(r(n), q(n))$, если он вероятностно проверяем с $r(n)$ случ. битами и $q(n)$ запросами к док-ву.

Теорема (PCP Theorem)

$\mathbf{NP} = \mathbf{PCP}(O(\log n), O(1))$.

Замечание

На самом деле достаточно трёх битов для вер. $1 - \delta$ vs $\frac{1}{2} + \delta$.

Напоминание: оптимизационные задачи

Оптимизационная задача = массовая задача M + целевая функция f.

Точный алгоритм для задачи максимизации находит по x решение s_*:

- $(x, s_*) \in M$;
- $f(x, s_*) = \max\limits_{s:(x,s)\in M} f(x, s)$.

ρ-приближённый алгоритм находит решение $s_\approx$:

- $(x, s_\approx) \in M$;
- $f(x, s_\approx) \geq \rho \cdot f(x, s_*)$.

Неаппроксимируемость

Теорема (PCP Theorem)

$\mathbf{NP} = \mathbf{PCP}(O(\log n), O(1))$.

Теорема (Переформулировка PCP Theorem)

$\exists \rho < 1$ т.ч. $\forall L \in \mathbf{NP}$ имеется полин. $f : \{0,1\}^* \mapsto \{3\text{–КНФ}\}$ т.ч.

$$\begin{aligned} x \in L &\Rightarrow f(x) \text{ выполнима,} \\ x \notin L &\Rightarrow \text{нельзя выполнить даже долю } \rho \text{ клозов } f(x). \end{aligned}$$

Следствие

$\mathbf{P} \neq \mathbf{NP} \Rightarrow \nexists$ полин. ρ-приближённого алгоритма для MAX–3–SAT.

Замечание

На самом деле если все клозы длины три, то $7/8$-приближённый есть, а $(7/8 + \varepsilon)$-приближённого нет.

Walsh-Hadamard code

- Everything is done modulo 2.
- $\mathtt{WH}(x) = (\langle x, y \rangle)_{y \in \{0,1\}^n}$ (2^n bits instead of n).

Walsh-Hadamard code

- Everything is done modulo 2.
- $\mathtt{WH}(x) = (\langle x, y \rangle)_{y \in \{0,1\}^n}$ (2^n bits instead of n).
- Error correcting code:

$$\forall x, x' \quad \Pr_r\{\mathtt{WH}(x)(r) \neq \mathtt{WH}(x')(r)\} \geq \frac{1}{2}.$$

Walsh-Hadamard code

- Everything is done modulo 2.
- $\mathtt{WH}(x) = (\langle x, y \rangle)_{y \in \{0,1\}^n}$ (2^n bits instead of n).
- Error correcting code:

$$\forall x, x' \ \Pr_r\{\mathtt{WH}(x)(r) \neq \mathtt{WH}(x')(r)\} \geq \frac{1}{2}.$$

- Testing for δ-closeness:
  - f, g are δ-close (for $\delta < 1/2$) if $\Pr_y\{f(y) \neq g(y)\} < \delta$.
  - For $\delta < 1/3$, f is not δ-close to a linear function $\Rightarrow$

$$\Pr_{y,z}\{f(y+z) \neq f(y) + f(z)\} > \delta/2. \quad \text{[PROOF DELAYED]}$$

  - Repeat $O(1)$ times, get 0.001-closeness with any small error.

Walsh-Hadamard code

- Everything is done modulo 2.
- $\mathtt{WH}(x) = (\langle x, y \rangle)_{y \in \{0,1\}^n}$ (2^n bits instead of n).
- Error correcting code:

$$\forall x, x' \ \Pr_r\{\mathtt{WH}(x)(r) \neq \mathtt{WH}(x')(r)\} \geq \frac{1}{2}.$$

- Testing for δ-closeness:
  - f, g are δ-close (for $\delta < 1/2$) if $\Pr_y\{f(y) \neq g(y)\} < \delta$.
  - For $\delta < 1/3$, f is not δ-close to a linear function $\Rightarrow$

$$\Pr_{y,z}\{f(y+z) \neq f(y) + f(z)\} > \delta/2. \quad \text{[PROOF DELAYED]}$$

  - Repeat $O(1)$ times, get 0.001-closeness with any small error.
- Self-correction: if $\tilde{f}$ is δ-close to linear f, then

$$\forall y \ \Pr_{y'}\{f(y) = \tilde{f}(y') + \tilde{f}(y + y')\} \geq 1 - 2\delta$$

for $\delta < 1/4$.

$\mathbf{NP} \subseteq \mathbf{PCP}(n^{O(1)}, O(1))$.

- **NP**-complete language: satisfiable systems of quadratic equations mod 2.
- Input: a system presumably satisfied by $u = (u_1, \ldots, u_n) \in \{0,1\}^n$.

$\mathbf{NP} \subseteq \mathbf{PCP}(n^{O(1)}, O(1))$.

- **NP**-complete language: satisfiable systems of quadratic equations mod 2.
- Input: a system presumably satisfied by $u = (u_1, \ldots, u_n) \in \{0,1\}^n$.
- Proof: $\mathtt{WH}(u)$ and $\mathtt{WH}(u \otimes u)$, where $(y \otimes z)_{ij} = y_i z_j$.

$\mathbf{NP} \subseteq \mathbf{PCP}(n^{O(1)}, O(1))$.

- **NP**-complete language: satisfiable systems of quadratic equations mod 2.
- Input: a system presumably satisfied by $u = (u_1, \ldots, u_n) \in \{0,1\}^n$.
- Proof: $\mathtt{WH}(u)$ and $\mathtt{WH}(u \otimes u)$, where $(y \otimes z)_{ij} = y_i z_j$.
- Verifier:
  1. Test the proof for 0.001-closeness to linear,
     then use linear functions f and g by self-correction.
  2. Check that f and g are related as presumed.
     No $\Rightarrow \Pr\{f(r)f(r') \neq g(r \otimes r')\} > 0.250$.
  3. Verify that f satisfies the system:
     take a random sum of equations: coefficients a, rhs c;
     f does not satisfy $\Rightarrow \Pr\{g(a) \neq c\} > 0.5$.