Сложность пропозициональных доказательств

Эдуард Алексеевич Гирш

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

ПОМИ РАН

30 сентября 2010 г.

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0, 1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0, 1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \texttt{False}$.

Моделирование секущих плоскостей в системах Фреге

Представление линейных неравенств

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0,1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \texttt{False}$.
- надо вычислить сумму $\sum_t c_t y_t$ для $y_t \in \{0,1\}$ и сравнить с c.

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0,1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \texttt{False}$.
- надо вычислить сумму $\sum_t c_t y_t$ для $y_t \in \{0,1\}$ и сравнить с c.
- $\mathrm{Add}((F_0, \ldots, F_k), (G_0, \ldots, G_k))_i =$
  $$F_i \oplus G_i \oplus \bigvee_{0 \leq j < i} (F_j \wedge G_j \wedge \bigwedge_{j < l < i} (F_l \oplus G_l)).$$

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0,1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \mathtt{False}$.
- надо вычислить сумму $\sum_t c_t y_t$ для $y_t \in \{0,1\}$ и сравнить с c.
- $\mathrm{Add}((F_0, \ldots, F_k), (G_0, \ldots, G_k))_i =$
  $$F_i \oplus G_i \oplus \bigvee_{0 \leq j < i} (F_j \wedge G_j \wedge \bigwedge_{j < l < i} (F_l \oplus G_l)).$$
- $\mathrm{SAdd}(\vec{F}, \vec{G}, \vec{H})_i = F_i \oplus G_i \oplus H_i$.
- $\mathrm{CAdd}(\vec{F}, \vec{G}, \vec{H})_{i+1} = (F_i \wedge G_i) \vee (F_i \wedge H_i) \vee (G_i \wedge H_i)$.

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0, 1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \texttt{False}$.
- надо вычислить сумму $\sum_t c_t y_t$ для $y_t \in \{0, 1\}$ и сравнить с c.
- $\mathrm{Add}((F_0, \ldots, F_k), (G_0, \ldots, G_k))_i =$
  $$F_i \oplus G_i \oplus \bigvee_{0 \leq j < i} (F_j \wedge G_j \wedge \bigwedge_{j < l < i} (F_l \oplus G_l)).$$
- $\mathrm{SAdd}(\vec{F}, \vec{G}, \vec{H})_i = F_i \oplus G_i \oplus H_i$.
- $\mathrm{CAdd}(\vec{F}, \vec{G}, \vec{H})_{i+1} = (F_i \wedge G_i) \vee (F_i \wedge H_i) \vee (G_i \wedge H_i)$.
- $\mathrm{SUM}(c_1 y_1, \ldots, c_n y_n)$: складываем SAdd, CAdd, последние — Add.

- побитное кодирование чисел формулами;
- неравенства вида $\sum_t c_t y_t \geq c$,
  где $c_t, c \geq 0$, $y_t \in \{0,1\}$ ($y_t = x_t$ или $y_t = \neg x_t$).
- $c_t \cdot y_t$ — это $(Y_0, \ldots, Y_k)$,
  где $Y_i = y_t$, если $(c_t)_i = 1$; иначе $Y_i = \texttt{False}$.
- надо вычислить сумму $\sum_t c_t y_t$ для $y_t \in \{0,1\}$ и сравнить с c.
- $\mathrm{Add}((F_0, \ldots, F_k), (G_0, \ldots, G_k))_i =$
  $$F_i \oplus G_i \oplus \bigvee_{0 \leq j < i} (F_j \wedge G_j \wedge \bigwedge_{j < l < i} (F_l \oplus G_l)).$$
- $\mathrm{SAdd}(\vec{F}, \vec{G}, \vec{H})_i = F_i \oplus G_i \oplus H_i$.
- $\mathrm{CAdd}(\vec{F}, \vec{G}, \vec{H})_{i+1} = (F_i \wedge G_i) \vee (F_i \wedge H_i) \vee (G_i \wedge H_i)$.
- $\mathrm{SUM}(c_1 y_1, \ldots, c_n y_n)$: складываем SAdd, CAdd, последние — Add.
- $\vec{F} > \vec{G}$ представляется как $\bigvee_i (F_i \wedge \neg G_i \wedge \bigwedge_{j>i} (F_j \equiv G_j))$.

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
    - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$
        - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.
  - докажем $\vec{F} \geq \vec{G} \quad \wedge \quad \vec{F'} \geq \vec{G'} \quad \supset \quad \mathrm{Add}(\vec{F}, \vec{F'}) \geq \mathrm{Add}(\vec{G}, \vec{G'})$.

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\dots, c_t y_t, \dots), \mathrm{SUM}(\dots, d_t y_t, \dots)) \equiv \mathrm{SUM}(\dots, (c_t + d_t) y_t, \dots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.
  - докажем $\vec{F} \geq \vec{G} \quad \wedge \quad \vec{F'} \geq \vec{G'} \quad \supset \quad \mathrm{Add}(\vec{F}, \vec{F'}) \geq \mathrm{Add}(\vec{G}, \vec{G'})$.
- умножение (деление) на константу...

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.
  - докажем $\vec{F} \geq \vec{G} \quad \wedge \quad \vec{F'} \geq \vec{G'} \quad \supset \quad \mathrm{Add}(\vec{F}, \vec{F'}) \geq \mathrm{Add}(\vec{G}, \vec{G'})$.
- умножение (деление) на константу...
- округление

$$\frac{\sum (ac_t) y_t \geq ac + r}{\sum c_t y_t \geq c + 1} \qquad (r < a)$$

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\dots, c_t y_t, \dots), \mathrm{SUM}(\dots, d_t y_t, \dots)) \equiv \mathrm{SUM}(\dots, (c_t + d_t) y_t, \dots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.
  - докажем $\vec{F} \geq \vec{G} \quad \wedge \quad \vec{F'} \geq \vec{G'} \quad \supset \quad \mathrm{Add}(\vec{F}, \vec{F'}) \geq \mathrm{Add}(\vec{G}, \vec{G'})$.
- умножение (деление) на константу...
- округление

$$\frac{\sum (a c_t) y_t \geq ac + r}{\sum c_t y_t \geq c + 1} \qquad (r < a)$$

— разбор случаев (т.е. док-во от противного):

$$\mathrm{SUM}(\dots, c_t y_t, \dots) \geq c + 1 \quad \vee \quad \neg(\mathrm{SUM}(\dots, c_t y_t, \dots) \geq c + 1),$$

из второго следует $\leq c$, умножим обратно на a. ...

- просуммируем $\sum c_t y_t \geq c$ и $\sum d_t y_t \geq d$:
  - докажем по индукции, что $\mathrm{Add}(\mathrm{SUM}(\ldots, c_t y_t, \ldots), \mathrm{SUM}(\ldots, d_t y_t, \ldots)) \equiv \mathrm{SUM}(\ldots, (c_t + d_t) y_t, \ldots$
    - равенство $\mathrm{Add}(c_t y_t, d_t y_t)_i \equiv (c_t + d_t)_i y_t$ – разбор случаев.
    - $y_t + \neg y_t$ — аналогично.
  - докажем $\vec{F} \geq \vec{G} \quad \wedge \quad \vec{F'} \geq \vec{G'} \quad \supset \quad \mathrm{Add}(\vec{F}, \vec{F'}) \geq \mathrm{Add}(\vec{G}, \vec{G'})$.
- умножение (деление) на константу...
- округление

$$\frac{\sum (ac_t) y_t \geq ac + r}{\sum c_t y_t \geq c + 1} \qquad (r < a)$$

  — разбор случаев (т.е. док-во от противного):

$$\mathrm{SUM}(\ldots, c_t y_t, \ldots) \geq c + 1 \quad \vee \quad \neg(\mathrm{SUM}(\ldots, c_t y_t, \ldots) \geq c + 1),$$

  из второго следует $\leq c$, умножим обратно на a. . . .
- свойства нуля: $\mathrm{Add}(\vec{F}, 0)_i \equiv F_i$ и $0 < 1$.
- $\mathrm{SUM}(0y_1, \ldots, 0y_n) \geq 1$, очевидно, ложно.

Оптимальные полуалгоритмы

Определение

A — оптимальный полуалгоритм для $L \iff$
для всякого A' имеется полином p, т.ч. $\forall x \in L$

$$\mathrm{time}_A(x) \leq p(\mathrm{time}_{A'}(x) + |x|).$$

Оптимальные полуалгоритмы

Определение

A — оптимальный полуалгоритм для $L \iff$
для всякого A' имеется полином p, т.ч. $\forall x \in L$

$$\mathrm{time}_A(x) \le p(\mathrm{time}_{A'}(x) + |x|).$$

Левинский оптимальный алгоритм для решения задачи поиска `SAT`: запустить "параллельно" все возможные алгоритмы, проверить выданный "выполняющий" набор, если верен — выдать.

Оптимальные полуалгоритмы

Определение

A — оптимальный полуалгоритм для $L \iff$
для всякого A' имеется полином p, т.ч. $\forall x \in L$

$$\mathrm{time}_A(x) \leq p(\mathrm{time}_{A'}(x) + |x|).$$

Левинский оптимальный алгоритм для решения задачи поиска `SAT`: запустить “параллельно” все возможные алгоритмы, проверить выданный “выполняющий” набор, если верен — выдать.

Замечание

Левинский алгоритм не для языка `TAUT`.

Оптимальные полуалгоритмы

Определение

A — оптимальный полуалгоритм для $L \iff$
для всякого A' имеется полином p, т.ч. $\forall x \in L$

$$\mathrm{time}_A(x) \le p(\mathrm{time}_{A'}(x) + |x|).$$

Левинский оптимальный алгоритм для решения задачи поиска SAT: запустить "параллельно" все возможные алгоритмы, проверить выданный "выполняющий" набор, если верен — выдать.

Замечание

Левинский алгоритм не для языка TAUT.
…и не для языка SAT.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$
$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longleftarrow$:

- Оптимальное док-во формулы F размера n:
  - Номер системы Π;
  - Π-доказательство формулы F.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longleftarrow$:

- Оптимальный полуалгоритм O работает полиномиальное время на любом подмножестве тавтологий из $\mathbf{P}$.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\Longleftrightarrow$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longleftarrow$:

- Оптимальный полуалгоритм O работает полиномиальное время на любом подмножестве тавтологий из **P**.
- Для любой системы доказательств Π, легко (за полиномиальное время) записать тавтологию $\mathrm{Con}_{\Pi,n}$, означающую "Π корректна для формул размера n".

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longleftarrow$:

- Оптимальный полуалгоритм O работает полиномиальное время на любом подмножестве тавтологий из **P**.
- Для любой системы доказательств Π, легко (за полиномиальное время) записать тавтологию $\mathrm{Con}_{\Pi,n}$, означающую "Π корректна для формул размера n".
- Значит, O полиномиален на $C_\Pi = \{\mathrm{Con}_{\Pi,n}\}_{n\in\mathbb{N}}$.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longleftarrow$:

- Оптимальный полуалгоритм O работает полиномиальное время на любом подмножестве тавтологий из **P**.
- Для любой системы доказательств П, легко (за полиномиальное время) записать тавтологию $\mathrm{Con}_{\Pi,n}$, означающую "П корректна для формул размера n".
- Значит, O полиномиален на $C_\Pi = \{\mathrm{Con}_{\Pi,n}\}_{n\in\mathbb{N}}$.
- Оптимальное док-во формулы F размера n:
    - Номер системы П;
    - Протокол работы O на $\mathrm{Con}_{\Pi,n}$;
    - П-доказательство формулы F.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ *p*-оптимальная система док-в $\iff$ $\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longrightarrow$:

- Пусть Π — *p*-оптимальная.

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$
$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longrightarrow$:

- Пусть Π — p-оптимальная.
- Оптимальный полуалгоритм: "параллельный" запуск всех O_i, претендующих на выдачу Π-доказательств.
- Выданное O_i "док-во" проверяется Π;
  если правильное — вернуть "1".

Оптимальные полуалгоритмы vs системы доказательств

Теорема (Krajíček, Pudlák, 1989)

$\exists$ p-оптимальная система док-в $\iff$

$\exists$ оптимальный полуалгоритм для $\mathtt{TAUT}$.

$\Longrightarrow$:

- Пусть Π — p-оптимальная.
- Оптимальный полуалгоритм: "параллельный" запуск всех O_i, претендующих на выдачу Π-доказательств.
- Выданное O_i "док-во" проверяется Π; если правильное — вернуть "1".

- По p-оптимальности Π для любого алгоритма A его протокол может быть за полиномиальное время преобразован в Π-док-во некоторым f. Композиция A и f имеется в $\{O_i\}_i$.

p-Optimal proof system from optimal acceptor for any paddable language [Messner, 99]

Definition

L is paddable if there is an injective non-length-decreasing polynomial-time padding function $\mathrm{pad}_L \colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ that is polynomial-time invertible on its image and such that $\forall x, w \ (x \in L \iff \mathrm{pad}_L(x, w) \in L)$.

Optimal proof:

- description of proof system Π;
- Π-proof π of F;
- 1^t (for how long can we work?).

Verification:

- run optimal acceptor on $\mathrm{pad}_L(x, \pi)$;
- for a correct proof, it accepts in a polynomial time because for a correct system Π, the set $\{\mathrm{pad}_L(x, \pi) \mid x \in L, \ \Pi(x, \pi) = 1\} \subseteq L$ can be accepted in a polynomial time.