Сложность доказательств

19 декабря 2018 г.

1 Abstract

Древовидная система доказательств-это система, в которой все предыдущие правила нужно выводить снова(если $(P_k, ...P_{k+s})$-набор строк, которые мы вывели на предыдущих шагах и P_{k+s+1}-строка выводимая из нашего набора, то для ее вывода нужно вывести каждый элемент набора с самой первой строчки)

Теорема 1 *Для древовидных аналогов LS существует нижняя экспоненциальная оценка на длину доказательства.*

Полуалгебраичные системы доказательств

Все системы доказательств оперируют с многочленами. Формула в кнф записыается так: Для каждого клоза, содержащего переменные x_i и отрицания y_i, берется уравнение $(1-x_1)*(1-x_2)*\cdots*(1-x_n)*y_1*y_2*\cdots*y_m=0$, так как переменные еа самом деле 0 или 1, fr;t добавляются уравнения $x*x=x$ для всех переменных. Мы будем не доказывать наличие выполняющего набора, а добавлять ее отрицание в систему и доказывать противоречивость системы.

Системы доказательств

Nullstellensatz(NS) - Изначально у нас есть система уравнений $f_i=0$, доказательство противоречивости -набор $g_i : \sum f_i * g_i = 1$, такие по тореме Гильберта о нулях существуют над алг. замкнутым полем.

Polynomial calculus (PC) - вывод док-ва в Nullstellensatz: из уравнения p=0,q=0 выводится p+q=0, из p=0 выводится pq=0. Доказательство противоречивости - вывод $1=0$.

Pozitivestellensatz(PSZ) - многочлены над R, есть система уравнений $f_i=0$, доказательство противоречивости - набор $g_i, h_i : g_i : \sum f_i * g_i = 1 + \sum h_i^2$.

Positivestellensatz calculus (PSZC) - вывод док-ва в Positivestellennsatz, правила вывода те же, что и в PC, а док-во противоречивости - набор многочленов h_i и вывод $0 = +\sum h_i^2$ из правил вывода.

Lovasz-Schreier calculus (LS) - манипулируем не с равенствами, а с неравенствами, из $f_1 \geq 0, f_2 \geq 0, \ldots, f_n \geq 0$ выводится $\sum a_i f_i \geq 0$, где $a_i \geq 0$ - константы. Также из $f \geq 0$следует $fx \geq 0, f(1-x) \geq 0$ для переменной x. Аксиомы - $x^2 - x \geq 0, x - x^2 \geq 0$ для всех переменных x. Вывод противоречия - вывод $-1 \geq 0$.

LS_+ - то же, но еще есть аксиома $l^2 \geq 0$ для любого полинома l.

Булева степень многочлена - степень многочлена, взятого по модулю $x_i^2 - x_i$ для всех переменных в нем. Булева степень доказательства - максимальная из булевых степеней многочлена в процессе.

Мы доказываем несуществование решения задачи о рюкзаке (доказываем несуществование x_i, таких что $f = x_1 + x_2 + \ldots + x_n = m$)) при нецелом m, то есть добавляем f в систему и доказываем противоречивость. Пусть $A(m)$ - ступенчатая функция, равная 0 вне $[0, n]$, $2k+4$ на $[k, k+1]$ и $[n-k-1, n-k]$.

Th: Булева степень любого док-ва неразрешимости задачи о рюкзаке в PC не менее $\frac{(n-1)}{2}$, в PSCZ - не менее $min(\frac{n-1}{2}, A(m))$.

Th: Размер любого доказательства неразрешимости задачи о рюкзаке при $m = \frac{2n+1}{4}$ в NS и PSZ экспоненциален.

Статические доказательства

NS - статическая версия PC, PSZ - статическая версия PSZC.

Static LS: - пусть есть система неравенств $s_i \geq 0$. Доказательство ее противоречивости - набор многочленов $w_{i,j}$, каждый из которых - произведение мономов $x_i, (1-x_i)$ и констант $a_{i,j} \geq 0$, такой что $\sum s_i \sum a_{i,j} w_{i,j} = -1$.

Static LS_+: - пусть есть система неравенств $s_i \geq 0$. Доказательство ее противоречивости - набор многочленов $w_{i,j}$, каждый из которых - произведение мономов $x_i, (1-x_i)$ или квадрат другого многочлена и констант $a_{i,j} \geq 0$, такой что $\sum s_i \sum a_{i,j} w_{i,j} = -1$.

Th: при $m = \frac{2*n+1}{4}$ размер любого док-ва рюкзака в Static LS, Static LS_+ экспоненциален от n.

В моих с Михаилом докладах были следующие системы доказательств:
1. Системы Фреге.
2. Игры Пудлака и Баса.
Системы Фреге определялись не нами.
Что такое игра Пудлака и Баса?

Есть два игрока: Павел и Сэм, у них есть тавтология ϕ. Сэм говорит, что знает набор значений переменных, при которм ϕ ложно. Павел пытается уличить Сэма и задаёт ему вопросы про значение произвольных формул от переменных формулы ϕ. Сэм отвечает. Павел уличает Сэма, если он получает непосредственное противоречие, это значит, например, он спрашивал ответы для формул $\phi \vee \psi$, ϕ, ψ, но ответы не сошлись. Деревом игры называется такое двоичное дерево, каждая внутреняя вершина которого помечена формулой, одно из рёбер которого помечено 0, другое 1. В каждом листе должно быть непосредственное противоречие (мы всегда считаем, что есть ответ 0 для исходной формулы ϕ).

Мы рассматриваем формулы, в которых используются только бинарные операции $\vee$ и $\wedge$ и унарная операция $\neg$.

Это две системы доказательств сводятся друг к другу:

Лемма 1. *Система Фреге моделирует исчисление секвенций. Древовидная система Фреге моделирует древовидное исчисление секвенций.*

Лемма 2. *По доказательству формулы ϕ в системе Фреге размера s можно построить дерево игры Пудлака-Баса высоты $O(\log s)$ и размера $poly(s)$, где константа зависит только от правил системы Фреге.*

Лемма 3. *По дереву игры Пудлака-Баса для формулы ϕ высоты h и размера s можно построить древовидный вывод секвенции $\vdash \phi$ высоты $h + O(1)$ и размера $poly(s)$.*

Нижняя оценка для систем Фреге ограниченной глубины:

Теорема 1. *Пусть F - система Фреге. Тогда для достаточно больших n для любой глубины d доказательство $\neg PHP_n^{n+1}$ в F имеет размер как минимум 2^{n^μ} для любого $\mu < \frac{1}{2}(\frac{1}{5})^{d+c}$, где c - это константа, которая зависит только от систем Фреге.*

Resolution proof system

Нам дали некоторую формулу Ф в КНФ, и мы хотим ее опровергнуть. Для этого нам понадобятся все клозы из этой формулы (их называют аксиомами) и два правила вывода:

1. The Resolution Rule: $\frac{E \vee x \quad F \vee \neg x}{E \vee F}$
2. The Weakening Rule: $\frac{E}{E \vee F}$

Здесь E, F - дизъюнкты, а x - переменная. Такими операциями мы хотим получить пустой клоз. Вообще говоря, второе правило излишне, но его добавляют для удобства.

Мы только что определили систему доказательств в резолюциях[1] (Resolution proof system[2]). У этой системы доказательств бывают разновидности, например, можно запретить переиспользовать выведенные клозы и получить систему доказательств tree-like Resolution (потому что вывод раньше был DAG, а теперь – дерево).

Определение 1. *Размером доказательства называется количество вершин в графе вывода. Обозначается $S(\pi)$ для general Resolution и $S_T(\pi)$ для tree-like Resolution.*

TL;DR. *На самом деле все затевается только ради следствий 1, 2 и 3. Можно читать только их, если не особо интересно что за формулы там внутри написаны.*

Определение 2. *Шириной клоза C называется количество литералов в нем. Шириной формулы называется максимальная ширина клоза в ней. Шириной резолюционного доказательства называется максимальная ширина клозов в нем. Обозначается $\omega(C)$, $\omega(\Phi)$ и $\omega(\pi)$ соответственно.*

Теорема 1. $\omega(\Phi \vdash 0) \leq \omega(\Phi) + \log S_T(\Phi)$

Теорема 2. $\omega(\Phi \vdash 0) \leq \omega(\Phi) + O(\sqrt{n \log S(\Phi)})$

Запоминать формулировки теорем выше не нужно, они призваны показать, что из того, что разность $\omega(\Phi \vdash 0) - \omega(\Phi)$ большая, следует экспоненциальная нижняя оценка на размер доказательства в резолюциях.

[1] Осторожно, все русские названия являются не особо интеллектуальной собственностью воспаленного сознания человека, писавшего этот файл.

[2] Но копипастить я умею.

Определение 3 (PHP_n^m). *PHP_n^m - конъюнкция следущих клозов:*

- $P_i := \bigvee_{1 \le j \le n} x_{ij}, 1 \le i \le m$
- $H_{i,i'}^j := \neg x_{ij} \vee \neg x_{i'j}, 1 \le i, i' \le m, 1 \le j \le n$

Это обычный принцип Дирихле. Про него ничего доказать не получится, потому что он имеет широкие клозы P_i, а $\omega(PHP_n^m \vdash 0) \le n$, поэтому мы модифицировали формулу двумя разными способами и для них все доказывали.

Определение 4 ($EPHP_n^m$). *Заменим все P_i на EP_i*

$$EP_i = \neg y_{i0} \wedge \bigwedge_{j=1}^{n} (y_{ij-1} \vee x_{ij} \vee \neg y_{ij}) \wedge y_{in}$$

Теорема 3. *Если $m > n$, то $\omega(EPHP_n^m \vdash 0) \ge n/3$.*

Следствие 1. *Если $m > n$, то $S_T(EPHP_n^m \vdash 0) = 2^{\Omega(n)}$*

Это один из двух основных результатов.

Еще мы рассматривали другой способ модифицировать формулу. Можно брать не все условия, а только те, которые есть в некотором двудольном графе (Если взять $K_{m,n}$, получится обычный PHP_n^m). Это называется $G - PHP$, причем, здесь G - граф. Конечно, G будет экспандером.

Лемма 1. *Пусть G, G' - графы на одном и том же множестве вершин, причем $E(G) \subset E(G')$, где E - множество ребер графа. Тогда $S(G - PHP) \le S(G' - PHP)$.*

Теорема 4. $\omega(G - PHP \vdash 0) \ge \frac{re}{2}$

Где r, e - параметры экспандера (если все еще не понятно, забудьте про эту теорему).

Следствие 2. $S(PHP_n^{n+1}) = 2^{\Omega(n)}$

Следствие 3. $S(PHP_n^m) = 2^{\Omega(\frac{n^2}{m \log m})}$

А это второй основной результат.

1 Polynomial calculus proofs

Опр: Дано поле K и множество переменных. **Polynomial calculus refutation** множества аксиом P, это последовательность полиномов такие что последняя строчка это 1 и каждая строчка это либо аксиома, либо получается из предедущих строчек использованием правил вывода: $\frac{f \quad g}{\alpha f+\beta g}$ и $\frac{f}{x\cdot f}$. Где α, β из K это скаляры а x любая переменная.

Опр: Степень опровержения равна d если степени все полиномов в опровержении степени не больше d.

Мы считаем, что полиномы x^2-x входят в аксиомы для всех переменных x. Это означает, что аксиомы $f_1, ..., f_k$ опровержимы тогда и только тогда когда $f_1 = f_2 = ... = f_k = 0$ не имеет $0-1$ решений.

Обозн: Для полинома f пусть $\bar{f}$ это единственный полилинейный полином равный f по модулю идеала порожденного всеми полиномами $x^2 - x$.

Обозн: Множество из чисел от 1 до i будем обозначать как $[i]$

Обозн: $x_{ij} = 1$ означает, что голубь i стоит в клетке j.

Пусть $Q_i = 1 - \sum_{j\in[n]} x_{ij}$. Тогда $\neg PHP_n^m$

это следующие полиномы:

1) Q_i для $i \in [m]$ 2) $x_{ij}x_{ij'}$ для $i \in [m], j, j' \in [n], j \neq j'$ 3) $x_{ij}x_{i'j}$ для $i, i' \in [m], j \in [n], i \neq i'$

Опр: Пусть T это множество всех мономов $x_{i_1j_1}...x_{i_kj_k}$ таких что все i_l различны и все j_l различны и пусть T_d будет множество всех мономов степени не более d.

Утв: Любой полином это линейная комбинация термов из T без увеличения степени.

Мы хотим построить базис B_d от векторного пространтсва порожденного T_d так чтобы элементами базиса были произведения неких переменных и неких аксиом Q, например $x_{3,1}x_{5,3}Q_2Q_4$. Если мы справимся все строчки доказательства выписать в этом базисе то 1 нельзя будет вывести из аксиом. Определение B_d использует понятие "pigeon dance"которое мы сейчас определим

Опр: Пусть $A = \{a|a$ функция из $[m]$ в

$$\{0, 1, ..., n\}$$

такая что

$$\forall i, a(i) = a(i')$$

означает что $i = i'\}$

Опр:$A_d = \{a \in A||a| \leq d\}$

Для $a = \{(i_1, j_1), ..., (i_k, j_k), (i'_1, 0), ..., (i'_l, 0)$ где j_m ненулевые, определяем $x_a = x_{i_1,j_1}...x_{i_k,j_k}Q_{i'_1}..Q_{i'_l}$ и

$$\hat{a} = \{(i_1, j_1), ..., (i_k, j_k)\}$$

Опр: Интуитивно, pigeon dance это когда у нас есть расстановка голубей по клеткам (те которые не в какой то клетке считаем в клетке 0), и мы передвигаем первого голубя до какой то незабитой клетки большей по номеру чем в той который он сидит. И так с каждем голубем.

Опр: Минимальный pigeon dance это когда передвигаем голубей до **минимальной** незабитой клетки большей по номеру.

Теорема 1. *Если существует pigeon dance то существует минимальный pigeon dance.*

Опр $B_d = \{x_a | a \in A_d$ *и существует pigeon dance на тех голубях которые не в клетке 0}*

Утв *Если* $d \leq \lceil n/2 \rceil$ *и* $a \in A_d$ *то существует pigeon dance на* a *тогда и только тогда когда существует на* $\hat{a}$

Утв *Минимальный pigeon dance это биекция*

Утв B_d *это базис*

Теорема 2. PHP^n_m *не имеет polynomial calculus refutation степени* $\leq \lceil n/2 \rceil$

Секущие плоскости

Используем пропозициональные переменные $\bar{p}$ с интерпретацией $0 = false$ и $1 = true$. Строка доказательства - это

$$\sum_k c_k p_k \geq C,$$

где c_k и C - целые.

Аксиомы: $p_k \geq 0$ и $-p_k \geq -1$ (т.е. $0 \leq p_k \leq 1$) для каждой пропозициональной переменной p_k.

Парвила:

1. Сложение. Из $\sum_k c_k p_k \geq C$ и $\sum_k d_k p_k \geq D$ получаем $\sum_k (c_k + d_k) p_k \geq C + D$;

2. Деление. Из $\sum_k c_k p_k \geq C$ получаем $\sum_k \frac{c_k}{d} p_k \geq \left\lceil \frac{C}{d} \right\rceil$, $d > 0$ - целое, которое делит каждое c_k;

3. Умножение. Из $\sum_k c_k p_k \geq C$ получаем $\sum_k d c_k p_k \geq dC$, где d - произвольное положительное целое.

Для опровержения множества неравенств надо получить противоречие $0 \geq 1$.

Statement. *По невыполнимой формуле в КНФ можно построить доказательство в секущих плоскостях.*

Statement. *Секущие плоскости моделируют резолюцию.*

Нижняя оценка

Идея: Извлечь из доказательства монотонную булеву схему и применить оценку (теорему Разборова) на монотонную схемную сложность.

Theorem 0.1. (Пудлак) *Если формула $A(x, y)$ такая, что все вхождения x положительны (т.е. без отрицания), или формула $B(x, y)$ такая, что все вхождения x отрицательны, то по доказательству $A(x, y) \wedge B(x, y)$ в секущих плоскостях размера s можно построить вещественную монотонную схему C размера $\leq s$ такую, что $C(x) = 1 \quad \forall x \in U$ и $C(x) = 0 \quad \forall x \in V$, где $U = \{x \mid \exists y : A(x, y) = 1\}$, $V = \{x \mid \exists z : A(x, z) = 1\}$.*

Theorem 0.2. (Разборов) *Пусть C - монотонная вещественная схема, принимающая на вход векторы из 0 и 1 длины $\binom{n}{2}$, кодирующие граф на n вершинах. Пусть C выдает 1, если граф содержит клику размера m, и выдает 0, если вершины графа можно раскрасить в $m - 1$ цвет, где $m = \left\lfloor \frac{1}{8} \left(\frac{n}{\log n} \right)^{\frac{2}{3}} \right\rfloor$. Тогда размер схемы хотя бы $2^{\Omega((\frac{n}{\log n})^{\frac{1}{3}})}$.*

Запишем формулой, что граф одновременно имеет клику размера m и правильным образом красится в $m-1$ цвет. Причем сделаем это так, чтобы попасть в условие теоремы Пудлака (это можно сделать). Тогда из двух предыдущих теорем получим нажнюю оценку.

Theorem 0.3. *При $m = \left\lfloor \frac{1}{8}\left(\frac{n}{\log n}\right)^{\frac{2}{3}} \right\rfloor$ размер доказательства в секущих плоскостях формулы $Clique \wedge Coloring$ есть $2^{\Omega((\frac{n}{\log n})^{\frac{1}{3}})}$.*

Мы смотрим на доказательства корректности метода резолюций Res. Корректность любой системы доказательств можно записать в виде формулы (для фиксированных r, n, m), так называемой Reflection Principle:

$$SAT_r^n(x, z) \wedge REF_{r,m}^n(x, y),$$

где r — количество клозов в формуле, n — количество переменных в ней, m — длина опровержения y для x. x здесь длины $r \times n \times 2$ кодирует саму формулу — $x_{i,v,b} = 1$ тогда и только тогда, когда переменная v (если $b = 1$, то с отрицанием, иначе положительно) входит в i-й клоз формулы. z кодирует выполняющий набор, но он не длины n, а длины $n + 2rn$ — там еще для каждого клоза хранится, какой литерал его выполняет. $SAT_r^n(x, z)$ просто проверяет, что z правда кодирует выполняющий набор формулы, которую, в свою очередь, кодирует x.

y же кодирует доказательство невыполнимости формулы в резолюциях — там так же кодируются клозы, плюс информация, из резолюции каких клозов очередной клоз был получен, и по какой переменной резолюция происходила. $REF_{r,m}^n(x, y)$ проверяет, что y кодирует опровержение x в Res. Получается, что если Res корректна, то Reflection Principle невыполнима — ведь если фиксировать формулу, то есть x, то нельзя одновременно выполнить и SAT, и REF, то есть подобрать и выполняющий набор z, и опровержение y.

Получается, чтобы доказать, что в Res нельзя опровергнуть выполнимые формулы, можно доказать невыполнимость Reflection Principle. Это и будет наш способ доказательство корректности. Вообще говоря, есть всякие теоремы о том, что из существования короткого опровержения Reflection Principle для системы доказательств A в системе доказательств B следуют всякие связи автоматизируемости этих систем, но это не то, о чем речь шла на семинаре.

Сначала мы доказали, что Reflection Principle для Res имеет короткое доказательство в Res[2]. Res[k] — это те же Res, только есть дополнительные переменные, которые соответствуют конъюнкциям не более, чем k исходных литералов. Другими словами, в Res[k] можно клозы делать не просто дизъюнкцией литералов (то есть формулами в 1-CNF), а формулами в k-CNF. Правила там самые естественные — если мы вывели $l_i \vee C$ и $l_j \vee C$, то выводим $(l_i \wedge l_j) \vee C$, и наоборот, из последнего можно вывести $l_i \vee C$ и $l_j \vee C$. Все правила для Res сохраняются. Это и есть, на пальцах, определение Res[2]. Как проходило доказательство писать не буду, там просто много-много техники.

Потом мы доказали, что Reflection Principle для Res в Res коротко не опровергается. Для этого мы пользуемся тем, что графы, содержащие клики размера $2k$ не отделяются короткими монотонными схемами от k-раскрашиваемых графов. Кроме того, по опровержению в резолюциях формулы вида $A(x, y) \wedge B(x, z)$ можно построить монотонную схему, которая по данному x говорит, что невыполнимо - $A(x, y)$ или $B(x, z)$. Теперь мы от противного строим схему для отделения графов с кликами от k-раскрашиваемых графов, подставив в Reflection Principle формулу $COL_k(G, q)$, которая проверяет, является ли q правильной раскраской G в k цветов.

Определение 1. *Система Фреге содержит (1) правильно сформулированные формулы над каким-то пропозициональным языком L; (2) конечное множество аксиом; (3) конечное множество правил вывода.*

Теорема 1. *Пусть F_1 и F_2 две системы Фреге над каким-то языком L. Тогда существует константа $c > 0$ такая, что для любой формулы φ и любого n, если φ выводится в F_1 за n шагов, то φ выводится в F_2 за cn шагов.*

Определение 2. *Размер доказательства — символьная длина доказательства, т.е. $\sum_{i=1}^{m} |\psi_i|$, где ψ_i —шаги доказательства.*

Теорема 2. *Формула PHP_n имеет полиномиальный размер доказательства Фреге.*

NP-pairs & Co.

A **disjoint NP-pair** is a pair (A, B) of nonempty sets A and B such that A, B ∈ NP and A∩B = ∅.
A **separator** is a set S such that $A \subseteq S$ and $B \subseteq \bar{S}$.

(A, B) is **many-one reducible** in polynomial-time to (C, D) ($(A,B) \leq_m^{pp} (C,D)$), if there exists a polynomial-time computable function f such that f(A) ⊆ C and f(B) ⊆ D.

(A, B) is **Turing reducible** in polynomial-time to (C, D) ($(A,B) \leq_T^{pp} (C,D)$), if there exists a polynomial-time oracle Turing machine M such that for every separator S of (C, D), L(M,S) is a separator of (A, B).

The **canonical pair** of f is the disjoint NP-pair (SAT^*, REF_f):

$$\begin{aligned} \mathrm{SAT}^* &= \{(x,0^n) \mid x \in \mathrm{SAT}\} \\ \mathrm{REF}_f &= \{(x,0^n) \mid \neg x \in \mathrm{TAUT} \text{ and } \exists y[|y| \leq n \text{ and } f(y) = \neg x]\}. \end{aligned}$$

Theorem 1. For every disjoint NP-pair (A, B) there exists a proof system f such that $(SAT^*, REF_f) \equiv_m^{pp} (A,B)$.

One proof system Π_w is **simulated** by another one Π_s if the shortest proof for every tautology in Π_s is at most polynomially longer than its shortest proof in Π_w.
The notion of **p-simulation** is similar, but requires also a polynomial-time computable function for translating the proofs from Π_w to Π_s.
A **(p-)optimal propositional proof system** is one that (p-)simulates all other propositional proof systems.

An **acceptor** for a language L is an algorithm that answers 1 for $x \in L$ and does not stop otherwise.
An **acceptor O is optimal** if for any other (correct) acceptor A, for every x ∈ L, the acceptor O stops on x in time bounded by a polynomial in |x| and the time taken by A(x).

Theorem 1. Optimal acceptors for TAUT exist <=> p-optimal proof systems for TAUT exist.
(TAUT - language of all propositional tautologies)

Семинар по сложности доказательств
Ограниченная арифметика: основные определения

Золотов Б.

1.

Язык ограниченной арифметики — $=,\ \leq;\quad 0,\ S,\ +,\ \cdot,\ \lfloor \frac{x}{2} \rfloor,\ |x|,\ \#,\ \leq.\ \ x\#y = 2^{|x|\cdot|y|}$.

Ограниченный квантор — вида $(Qx \leq t)$. Остро ограниченный — вида $(Qx \leq |t|)$. Ограниченная формула — логическая формула только с такими кванторами.

Иерархия ограниченных формул Σ^b_k, Π^b_k — определяется чередованием ограниченных кванторов, на строго ограниченные забиваем. Предикат лежит в классе Σ^p_k полиномиальной иерархии, если и только если определяется Σ^b_k–формулой.

2.

T^i_2 — первопорядковая теория в языке ограниченной арифметики, задающаяся аксиомами: (а) BASIC, описывающими свойства арифметических операций (б) аксиомой индукции для каждой формулы из Σ^b_i с одной свободной переменной.

S^i_2 — то же самое, но вместо аксиомы индукции для каждой формулы A включаем аксиому PIND, где переход от $\lfloor \frac{x}{2} \rfloor$ к x. $S^1_2 \subseteq T^1_2 \subseteq S^2_2 \subseteq T^2_2 \subseteq \ldots$.

$f\colon\ \mathbb{N} \longrightarrow \mathbb{N}$ — Σ^b_i–определяется теорией R, если существует формула $A(\vec{x}, y) \in \Sigma^b_i$ такая, что

(1) Всегда верно $A(\vec{n}, f(\vec{n}))$;

(2) Из R можно вывести, что для любого $\vec{n}$ существует *ровно один* y, такой что $A(\vec{n}, y)$.

Предикат $P \subseteq \mathbb{N}$ — Δ^b_i–определяется теорией R, если существуют Σ^b_i–формула A и Π^b_i формула B такие, что они обе задают P, и в R можно доказать их эквивалентность.

3.

Теорема: Пусть $A \in \Sigma^b_i$ — тогда существуют $B \in \Sigma^b_i, f \in \square^p_i$ и терм t такие, что:

(1) $S^i_2 \vdash B$ верна только если верна A;

(2) Для всякого $\vec{x}$ существует единственный y, т. ч. $B(\vec{x}, y)$,

(3) И этот y не превосходит t;

(4) Для всякого $\vec{n}$ верно $\mathbb{N} \models B(\vec{n}, f(\vec{n}))$, то есть, формула B задаёт функцию f.

Теорема: Если $f \in \square^p_i$, то существует задающая её формула B, такая что (2)–(4).

Теорема: Функции, Σ^b_i–определяющиеся теорией S^i_2, — в точности $\square^p_i$.

Теорема: Предикаты, Δ^b_i–определяющиеся теорией S^i_2, — в точности Δ^p_i из полиномиальной иерархии.

4.

Cut:

$$\frac{\Gamma \longrightarrow \Delta, A \qquad A, \Pi \longrightarrow X}{\Gamma,\ \Pi \quad \longrightarrow \quad \Delta, X}$$

Обычно стараемся от них избавиться (хотя бы от некоторых), чтобы было *subformula property*.

5.

Definition Fix $i \geq 1$. Let $B(\vec{a})$ be a Σ^b_i-formula with all free variables indicated. Then $Witness^{i,\vec{a}}_B(w,\vec{a})$ is a formula defined inductively by:

(1) If $B \in \Sigma^b_{i-1} \cup \Pi^b_{i-1}$ then $Witness^{i,\vec{a}}_B(w,\vec{a}) \Leftrightarrow B(\vec{a})$.

(2) If $B = C \vee D$ then

$$Witness^{i,\vec{a}}_B(w,\vec{a}) \Leftrightarrow Witness^{i,\vec{a}}_C(\beta(1,w),\vec{a}) \vee Witness^{i,\vec{a}}_D(\beta(2,w),\vec{a}).$$

(3) If $B = C \wedge D$ then

$$Witness^{i,\vec{a}}_B(w,\vec{a}) \Leftrightarrow Witness^{i,\vec{a}}_C(\beta(1,w),\vec{a}) \wedge Witness^{i,\vec{a}}_D(\beta(2,w),\vec{a}).$$

(4) If $B = (\exists x \leq t) C(\vec{a},x)$ then

$$Witness^{i,\vec{a}}_B(w,\vec{a}) \Leftrightarrow \beta(1,w) \leq t \wedge Witness^{i,\vec{a},b}_{C(\vec{a},b)}(\beta(2,w),\vec{a},\beta(1,w)).$$

(5) If $B = (\forall x \leq |t|) C(\vec{a},x)$ then

$$Witness^{i,\vec{a}}_B(w,\vec{a}) \Leftrightarrow (\forall x \leq |t|)\, Witness^{i,\vec{a},b}_{C(\vec{a},b)}(\beta(x+1,w),\vec{a},x).$$

(6) If $B = \neg C$ use prenex operations to push the negation sign inside.

6.

1. По доказательству в теории S^1_2 можно построить extended Frege–доказательство полиномиального размера.
2. По доказательству в теории $S_2 = \bigcup S^i_2 = T_2$ можно построить Frege–доказательство полиномиального размера и фиксированной глубины.