Е. Я. Данцин

ВЕРОЯТНОСТНАЯ ПРОВЕРКА ДОКАЗАТЕЛЬСТВ В ИСЧИСЛЕНИЯХ

1. Введение

Говоря неформально, *вероятностно проверяемое доказательство* – это доказательство, которое можно проверить, не читая его полностью, а прочтя только небольшое число его битов. Проверка является вероятностной: верное доказательство всегда принимается, а ошибочный кандидат в доказательства отвергается с большой вероятностью. Понятие вероятностно проверяемого доказательства[1] было формализовано, в основном, в [4, 2] и изучалось во многих других работах, относящихся к интерактивным доказательствам и проверке правильности программ (историю создания этого понятия можно найти в обзорах [3, 8]).

Основные результаты о вероятностно проверяемых доказательствах показывают, что если у теоремы t есть доказательство экспоненциального размера $2^{|t|^{O(1)}}$, то можно эффективно преобразовать это доказательство в вероятностно проверяемое доказательство, которое проверяется за полиномиальное время $|t|^{O(1)}$, причем читается только $O(1)$ его битов ([5, 4, 2]). Хотя исходная мотивация была связана с проверкой вычислений, некоторые результаты о вероятностно проверяемых доказательствах имели неожиданные последствия для приближенного решения задач дискретной оптимизации (см. обзор [1]).

Известные примеры вероятностно проверяемых доказательств выглядят очень непохожими на обычные доказательства в исчислениях, т.е. на доказательства, формализуемые как выводы из аксиом. Типичное вероятностно проверяемое доказательство – это набор таблиц (массивов целых чисел), которые предста-

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований.

[1]Термины "прозрачное (transparent) доказательство" и "голографическое доказательство" иногда используются как синонимы термина "вероятностно проверяемое доказательство".

вляют функции многих переменных над конечными областями. Вероятностная проверка таких доказательств включает нетривиальные процедуры (например, вероятностный тест на то, что функция, представленная данной таблицей, близка к полиному низкой степени). Возникает вопрос: могут ли обычные доказательства в исчислениях быть вероятностно проверяемыми, т.е. возможно ли проверять доказательства в исчислении, рассматривая только небольшое число членов вывода?

Исчисления, обладающие таким свойством, определяются в разделе 2. Они называются *исчислениями с вероятностно проверяемыми доказательствами* и допускают очень простой способ проверки доказательств. А именно, дерево P, которое представляет кандидата в доказательства, принимается или отвергается следующим “наивным” алгоритмом:

> Выбрать путь в P, идущий от корня к листьям, случайным образом. Проверить применение правил вывода в каждой из его вершин. Если все применения правильны, и последняя вершина является аксиомой, то принять P. В противном случае – отвергнуть.

Кроме того, эта проверка предполагается ограниченной по времени полиномом $|t|^{O(1)}$, где t – слово, приписанное корню P, т.е. кандидат в теоремы. Если выбор пути и проверка не закончились за это время, то P отвергается. Таким образом, время проверки полиномиально от $|t|$, даже если размер кандидата в доказательства экспоненциален от $|t|$.

Этот алгоритм проверки является осмысленным, так как для исчислений с вероятностно проверяемыми доказательствами выполняются условия полноты и корректности:

1. *Полнота.* Для каждой теоремы есть доказательство, которое всегда принимается, т.е. принимается с вероятностью 1.
2. *Корректность.* Если кандидат в теоремы недоказуем, то любой кандидат в доказательства для него отвергается с вероятностью, не меньшей $1/2$.

Эти условия означают, что алгоритм проверки – это алгоритм типа Монте-Карло с односторонней ошибкой ([11]). Отрицательный ответ (P отвергается) гарантирует, что P не является доказательством. Однако, положительный ответ (P принимается и объявляется доказательством) может быть ошибочным с вероятностью меньшей $1/2$. Конечно, вероятность ошибки можно уменьшить повторными применениями алгоритма. Заметим, что подобные условия полноты и корректности выполняются также и для систем интерактивных доказательств, и для систем вероятностно проверяемых доказательств общего вида.

Две теоремы в разделах 3 и 4 характеризуют дедуктивную силу исчислений с вероятностно проверяемыми доказательствами, т.е. сложность распознавания теорем в таких исчислениях. Первая устанавливает верхнюю оценку, а вторая – нижнюю:

1. Для любого исчисления с вероятностно проверяемыми доказательствами множество его теорем принадлежит PSPACE.
2. Существует исчисление с вероятностно проверяемыми доказательствами, у которого множество теорем является PSPACE-трудным.

Доказательство PSPACE-трудности использует технику, разработанную для того, чтобы установить один из основных результатов о системах интерактивных доказательств, а именно, соотношение IP=PSPACE ([9, 13]). Более точно, это доказательство использует конструкцию, предложенную А. Шенем в его упрощенном доказательстве IP=PSPACE ([14]).

Исчисления с вероятностно проверяемыми доказательствами имеют особенность, которая отличает их правила вывода. Именно эта особенность позоляет нам проверять доказательство, рассматривая только один случайно выбранный путь. Рассмотрим обычное свойство правил вывода в любом исчислении: если все посылки доказуемы, то и заключение доказуемо, или, эквивалентно, если заключение недоказуемо, то хотя бы одна посылка недоказуема. Многопосылочные правила вывода в исчислениях с вероятностно проверяемыми доказательствами обладают более сильным свойством:

> Если заключение недоказуемо, то “почти” все посылки недоказуемы, т.е. все посылки, возможно за исключением малого числа, недоказуемы.

Посылки такого правила *избыточны*, так как нет необходимости доказывать все посылки, а достаточно доказать только несколько из них. Исчисление, построенное в доказательстве теоремы о PSPACE-трудности, дает пример правил со свойством избыточности. Эти правила, по существу, формализуют очевидное свойство полиномов низкой степени над конечным полем: если два полинома степени d различны, то они различны везде, кроме не более d точек. Это же свойство полиномов является ключевым элементом и в упомянутой технике для доказательства IP=PSPACE.

2. Исчисления с вероятностно проверяемыми доказательствами

Системы доказательств и исчисления. Наиболее общее понятие *системы доказательств* определяется следующим образом. Пусть Σ^* – множество всех слов в конечном алфавите Σ. *Система доказательств* – это предикат $\mathcal{S}(t, P)$, где $t, P \in \Sigma^*$. Слова t и P называются соответственно *кандидатом в теоремы* и *кандидатом в доказательства*. Слово P называется *доказательством* слова t, если значение $\mathcal{S}(t, P)$ истинно. Слово t называется *теоремой*, если t имеет доказательство.

Приведем простой пример системы доказательств: Σ – это $\{0, 1\}$, слова понимаются как числа в двоичной записи, значение $\mathcal{S}(t, P)$ истинно тогда и только тогда, когда P является нетривиальным делителем t. Множество теорем этой системы совпадает с можеством составных чисел.

Любое исчисление также является примером системы доказательств. В этом случае $\mathcal{S}(t, P)$ считается истинным, если P кодирует дерево вывода t из аксиом при помощи правил вывода. Более точно, исчисления определяются следующим образом.

Правилом вывода (для краткости, *правилом*) называется кортеж $(s_1, \ldots, s_m, s)$, где $s_1, \ldots, s_m, s \in \Sigma^*$. Слова $s_1, \ldots, s_m$ называются *посылками*, s называется *заключением*. Правило обычно записывается в виде

$$\frac{s_1 \quad \ldots \quad s_m}{s}$$

и читается “s непосредственно следует из $s_1, \ldots, s_m$”. *Исчислением* называется пара (A, R), где A – это подмножество слов из Σ^*, называемых *аксиомами*, а R – это множество правил. Оба этих множества обычно предполагаются разрешимыми. Более того, в этой статье мы ограничиваемся исчислениями, в которых и множества аксиом, и множества правил разрешимы в полиномиальное время. Поэтому мы отождествляем исчисления с парами $(\mathcal{A}, \mathcal{R})$, где $\mathcal{A}$ и $\mathcal{R}$ – полиномиально вычислимые разрешающие алгоритмы для аксиом и правил соответственно.

Доказательство в исчислении – это дерево, вершинам которого приписаны слова, удовлетворяющие условиям:

1. Если вершина v – лист, то слово, приписанное v, является аксиомой.
2. Если вершина v не является листом, то слово, приписанное v, непосредственно следует из слов, приписанных ее сыновьям.

Слово t называется *теоремой*, если существует доказательство, корню которого приписано слово t.

Под *кандидатом в доказательства* для t мы подразумеваем любое дерево, вершинам которого приписаны слова и корню которого приписано слово t. Вершина в таком дереве называется *правильной*, если она удовлетворяет условиям 1 и 2 выше. Ясно, что кандидат в доказательства является доказательством тогда и только тогда, когда все его вершины являются правильными.

Пути и их вероятности. Пусть P – кандидат в доказательства. *Путь* в P – это последовательность вершин $v_1, \dots, v_l$ таких, что v_1 является корнем, v_l является листом, а v_{i+1} является одним из сыновей v_i для каждого $1 \leq i < l$. Число l называется *длиной* пути. Путь называется *правильным*, если все его вершины правильные. *Степень* вершины в дереве – это число ее сыновей. Под *максимальной степенью* пути понимается максимум степеней его вершин. Аналогично, под *максимальной длиной слова* понимается максимум длин слов, приписанных вершинам пути.

Будем рассматривать длину пути, его максимальную степень и максимальную длину слова как сложностные характеристики пути. Пусть t – слово, приписанное корню P, и пусть $|t|$ обозначает его длину. Будем говорить, что путь *ограничен* полиномом q, если его длина, максимальная степень и максимальная длина слова не превосходят $q(|t|)$.

Допустим, мы выбираем путь в дереве случайным образом, т.е. мы начинаем с корня v_1, выбираем следующую вершину v_2 случайно среди сыновей v_1, и т.д. Ясно, что путь $v_1, v_2, v_3, \dots$ выбирается с вероятностью $(1/d_1) \cdot (1/d_2) \cdot \dots$, где d_i обозначает степень вершины v_i. Таким образом, под *вероятностью пути* $v_1, \dots, v_l$ понимается произведение $(1/d_1) \cdot \dots \cdot (1/d_{l-1})$.

Основное определение. Исчисление $(\mathcal{A}, \mathcal{R})$ называется *исчислением с вероятностно проверяемыми доказательствами*, если существует полином q такой, что выполнены следующие условия:

1. *Полнота.* Каждая теорема $(\mathcal{A}, \mathcal{R})$ имеет доказательство, в котором все пути правильны и ограничены полиномом q.

2. *Корректность.* Если t не является теоремой $(\mathcal{A}, \mathcal{R})$, то любой кандидат в доказательства для t имеет свойство: суммарная вероятность путей, которые правильны и ограничены полиномом q, меньше $1/2$.

Напомним "наивный" алгоритм проверки, описанный в разделе 1. Этот алгоритм принимает или отвергает кандидата в доказательства P, выбирая случайный путь в P и проверяя правильность его вершин. Если все вершины правильные, то P принимается, в противном случае – отвергается. Кроме того, P отвергается и в случае, когда выбор пути и его проверка требуют времени больше, чем $q(|t|)$, где q – полином. Пусть $(\mathcal{A}, \mathcal{R})$ – исчисление с вероятностно проверяемыми доказательствами. Тогда, как легко видеть, (1) каждая теорема имеет доказательство, которое алгоритм всегда принимает; (2) если t не является теоремой, то алгоритм отвергает любой кандидат в доказательства для t с вероятностью, не меньшей $1/2$.

3. Сложность распознавания теорем: верхняя оценка

Следующая теорема показывает, что если $(\mathcal{A}, \mathcal{R})$ – исчисление с вероятностно проверяемыми доказательствами, то существует алгоритм с полиномиально ограниченной памятью, который распознает теоремы $(\mathcal{A}, \mathcal{R})$.

Теорема 1. *Для любого исчисления с вероятностно проверяемыми доказательствами множество его теорем принадлежит PSPACE.*

Доказательство. Пусть $(\mathcal{A}, \mathcal{R})$ – исчисление с вероятностно проверяемыми доказательствами. В силу условия полноты, $(\mathcal{A}, \mathcal{R})$ обладает следующим свойством: существует полином q такой, что каждая теорема t имеет доказательство, в котором высота, максимальная степень и максимальная длина формул не превосходят $q(|t|)$. Докажем, что для любого исчисления с этим свойством, множество теорем принадлежит PSPACE. Так как PSPACE совпадает с классом языков, которые распознаваются альтернированными машинами Тьюринга с полиномиально ограниченным временем работы (см., например, [12]), будем строить такую машину M для множества теорем $(\mathcal{A}, \mathcal{R})$.

Пусть t – входное слово, которое должно быть принято машиной M в случае, если t является теоремой $(\mathcal{A}, \mathcal{R})$, и отвергнуто в противном случае. Мы говорим, что кортеж $(s_1, \ldots, s_m)$ слов является *возможным кортежем посылок* для слова s, если исчисление $(\mathcal{A}, \mathcal{R})$ содержит правило

$$\frac{s_1 \quad \ldots \quad s_m}{s},$$

где $m \leq q(|t|)$, и длина каждого из слов $s_1, \ldots, s_m$ также не превосходит $q(|t|)$.

Опишем И/ИЛИ-дерево вычисления альтернированной машины M. Вычисление начинается с проверки входного слова t. Если это слово является аксиомой, то M останавливается в принимающем состоянии. В противном случае M переходит в ИЛИ-состояние и разветвляет свое вычисление, порождая все возможные кортежи посылок для t. Каждый из них можно вычислить за полиномиальное время следующим образом. Машина M порождает все кортежи, состоящие из не более $q(|t|)$ слов, длины которых не превосходят $q(|t|)$. Применяя разрешающий алгоритм для правил вывода, M проверяет, является ли данный кортеж возможным кортежем посылок для t. Если да, то M продолжит работу с этим кортежем, в противном случае M останавливается в отвергающем состоянии. Если не существует ни одного возможного кортежа посылок, то M также останавливается в отвергающем состоянии.

Пусть $(s_1, \ldots, s_m)$ – возможный кортеж посылок для t. Работа машины M на нем состоит в том, что M переходит в И-состояние и разветвляет свое вычисление на обработку слов $s_1, \ldots, s_m$ отдельно. Каждое слово s_i обрабатывается аналогично слову t, а именно: (1) M проверяет, не является ли s_i аксиомой; (2) если нет, то M переходит в ИЛИ-состояние и разветвляет вычисление для обработки всех возможных кортежей посылок для s_i; (3) M расщепляет каждый возможный кортеж на отдельные слова; и т.д. Ясно, что M принимает t тогда и только тогда, когда t – теорема.

По предположению, каждая теорема t имеет доказательство, высота которого не превосходит $q(|t|)$. Поэтому можно ограничить высоту И/ИЛИ-дерева некоторым полиномом q', т.е. если вход t не был принят за $q'(|t|)$ шагов, то M отвергает его. Таким образом, M принимает или отвергает вход за полиномиальное время.

□

4. Сложность распознавания теорем: нижняя оценка

Покажем, что любая задача из PSPACE может быть сведена к распознаванию теорем некоторого исчисления с вероятностно проверяемыми доказательствами. Тем самым, учитывая теорему 1, мы получаем точную характеризацию дедуктивной силы таких исчислений.

Теорема 2. *Существует исчисление с вероятностно проверяемыми доказательствами, у которого множество теорем является PSPACE-трудным.*

Доказательство. Доказательство будет состоять из двух частей (разделы 4.1 и 4.2 соответственно). В первой части доказательства мы описываем полиномиальное сведение PSPACE-полного языка, а именно, языка, состоящего из истинных булевских формул с кванторами, к некоторому множеству арифметических формул. Булевская формула с кванторами Φ переводится за полиномиальное время в арифметическую формулу $\alpha(\Phi)$ с ограниченными кванторами, которая истинна тогда и только тогда, когда Φ истинна. Вторая часть доказательства – это построение исчисления $(\mathcal{A}, \mathcal{R})$, которое является исчислением с вероятностно проверяемыми доказательствами и обладает следующим свойством: формула вида $\alpha(\Phi)$ является теоремой $(\mathcal{A}, \mathcal{R})$ тогда и только тогда, когда она истинна. Тем самым будет показано, что $(\mathcal{A}, \mathcal{R})$ – требуемое исчисление.

4.1. Перевод булевских формул с кванторами в арифметические формулы

Перевод булевских формул с кванторами в арифметические формулы делается в два шага. Первый шаг – это перевод в последовательность полиномов над конечным полем. Истинная булевская формула с кванторами переводится в последовательность, у которой последний полином тождественно равен 1. В случае ложной формулы последний полином последовательности тождественно равен 0. Этот шаг заимствован из упрощенного доказательства теоремы IP=PSPACE в статье А. Шеня [14]. Второй шаг – представление таких полиномов арифметическими формулами. Этот способ представления был предложен автором в [7], близкий способ был использован в [6]. Используя это представление, мы формализуем утверждение о том, что последний полином последовательности тождественно равен 1, в виде арифметической формулы. Такая формула доказуема в исчислении $(\mathcal{A}, \mathcal{R})$ тогда и только тогда, когда утверждение действительно истинно.

Булевские формулы с кванторами. Под *булевской формулой с кванторами* мы подразумеваем формулу

$$Q_1 x_1 \ldots Q_n x_n \phi(x_1, \ldots, x_n),$$

где $Q_1, \ldots, Q_n$ – это кванторы $\forall$ или $\exists$, а $\phi(x_1, \ldots, x_n)$ – это булевская формула, содержащая только переменные $x_1, \ldots, x_n$. Заметим, что такая формула не содержит свободных переменных по определению. Пусть QBF обозначает язык, состоящий из истинных булевских формул с кванторами (известно, что этот язык является PSPACE-полным). Пусть QBF^* обозначает подмножество QBF, определяемое следующим ограничением на бескванторную часть: каждая переменная имеет не более 3 вхождений в ϕ.

Нетрудно убедиться, что QBF^* также является PSPACE-полным языком. Действительно, пусть Φ обозначает формулу $Q_1 x_1 \ldots Q_n x_n \phi$. Построим полиномиальный по времени алгоритм, который преобразует формулу Φ в другую булевскую формулу с кванторами Φ' такую, что (1) Φ и Φ' имеют одно и то же истинностное значение, и (2) каждая переменная Φ' входит в бескванторную часть не более 3 раз. Этот алгоритм преобразует формулу Φ за n шагов. Допустим, что переменная x_1 имеет m вхождений в бескванторную часть Φ, и пусть $m > 3$. Тогда алгоритм заменяет эти вхождения на m новых вхождений переменных $x_{11}, \ldots, x_{1m}$ соответственно и строит формулу Φ_1:

$$Q_1 x_{11} \ldots Q_1 x_{1m}\ Q_2 x_2 \ldots \quad (\ \phi(x_{11}, \ldots, x_{1m},\ x_2, \ldots)\ \&\ \phi_1\ ),$$

где ϕ_1 – это формула $(x_{11} \to x_{12})\ \& \ldots \&\ (x_{1m} \to x_{11})$. Очевидно, что ϕ_1 истинна тогда и только тогда, когда новые переменные имеют одинаковые истинностные значения. Поэтому формулы Φ и Φ_1 эквивалентны, и каждая из переменных $x_{11}, \ldots, x_{1m}$ входит в бескванторную часть Φ_1 ровно 3 раза. На втором шаге алгоритм строит Φ_2, заменяя переменную x_2 аналогичным образом, и т.д. Последняя формула Φ_n – это требуемая формула Φ'. Таким образом, множество QBF^* является PSPACE-полным.

Перевод в последовательности полиномов ([14]). Пусть $\mathbf{F}$ – конечное поле. Мы говорим, что булевская формула

$\phi(x_1, \ldots, x_n)$ и полином $f(x_1, \ldots, x_n)$ над $\mathbf{F}$ *согласованы*, если они принимают одни и те же значения на аргументах 0 и 1 (логические константы "ложь" и "истина" отождествляются соответственно с элементами 0 и 1 поля $\mathbf{F}$). Любую булевскую формулу ϕ можно преобразовать в полином f такой, что f и ϕ согласованы. Мы строим f, преобразуя подформулы ϕ следующим образом:

1. Булевские переменные становятся переменными для элементов $\mathbf{F}$.
2. Каждая подформула $\neg\psi$ заменяется на $1 - \psi$.
3. Каждая подформула $\psi_1 \& \psi_2$ заменяется на $\psi_1 \cdot \psi_2$.
4. Каждая подформула $\psi_1 \vee \psi_2$ заменяется на $\psi_1 + \psi_2 - \psi_1 \cdot \psi_2$.

Легко видеть, что значения f и ϕ совпадают, если аргументы $x_1, \ldots, x_n$ принимают значения 0 и 1. Очевидно также, что если каждая переменная входит в ϕ не более 3 раз, то степень полинома f по любой из переменных не превосходит 3.

Для построения полиномов, соответствующих булевским формулам с кванторами, мы определяем три операции над полиномами. Пусть q – полином, который содержит переменную x. Пусть $q(0, \ldots)$ и $q(1, \ldots)$ обозначают полиномы, которые получаются из q, если мы заменяем x на 0 и 1 соответственно. Полиномы $\mathcal{A}xq$, $\mathcal{E}xq$ и $\mathcal{L}xq$ определяются следующим образом:

$$\begin{aligned}
&\mathcal{A}xq = q(0, \ldots) \cdot q(1, \ldots);\\
&\mathcal{E}xq = q(0, \ldots) + q(1, \ldots) - q(0, \ldots) \cdot q(1, \ldots);\\
&\mathcal{L}xq = q \pmod{x^2 - x}.
\end{aligned}$$

Последняя операция "линеаризует" полином q относительно переменной x, т.е. мы заменяем все вхождения x^d $(d > 1)$ в q на x. Таким образом, полиномы $\mathcal{L}xq$ и q имеют одно и то же множество переменных. Полиномы $\mathcal{A}xq$ и $\mathcal{E}xq$ имеют то же множество переменных, что и q, за исключением x. Заметим также, что полиномы q и $\mathcal{L}xq$ согласованы на аргументах 0 и 1.

Пусть Φ – булевская формула с кванторами $Q_1 x_1 \ldots Q_n x_n$ $\phi(x_1, \ldots, x_n)$ такая, что каждая переменная входит в ϕ не более 3 раз. Операции $\mathcal{A}$, $\mathcal{E}$ и $\mathcal{L}$ позволяют построить последовательность полиномов $f_0, \ldots, f_t$, которые удовлетворяют следующим условиям:

1. Полином f_0 согласован с бескванторной подформулой ϕ.

2. Полином f_t согласован с Φ, т.е. f_t тождественно равен 1, если Φ истинна, и f_t тождественно равен 0 в противном случае.
3. Каждый полином f_{i+1} является результатом применения одной из операций $\mathcal{A}$, $\mathcal{E}$ или $\mathcal{L}$ к полиному f_i (при $0 \le i < t$).
4. Каждый полином f_i имеет степень не более 3 по любой из его переменных.

В качестве полинома f_0 нашей последовательности возьмем полином f, соответствующий ϕ. Здесь мы фиксируем поле $\mathbf{F}$, которое до сих пор могло быть любым конечным полем. А именно, мы берем в качестве $\mathbf{F}$ поле Z/pZ, где p – наименьшее простое число, большее $6n^2$, а n – число переменных в ϕ. Выбор такой границы для p станет ясным в разделе 4.2. Таким образом, f_0 – полином над Z/pZ, и, очевидно, f_0 удовлетворяет условиям 1 и 4.

Затем мы линеаризуем f_0 относительно всех переменных, т.е. мы строим $f_1, \ldots, f_n$, последовательно применяя $\mathcal{L}$:

$$f_1 = \mathcal{L}x_1 f_0\,, \quad \ldots, \quad f_n = \mathcal{L}x_n f_{n-1}\,.$$

Последний полином f_n является мультилинейным. Построение следующего полинома f_{n+1} зависит от квантора Q_n в Φ: если Q_n – это $\forall$, то

$$f_{n+1}(x_1, \ldots, x_{n-1}) = \mathcal{A}x_n f_n(x_1, \ldots, x_n)\,,$$

а если Q_n – это $\exists$, то

$$f_{n+1}(x_1, \ldots, x_{n-1}) = \mathcal{E}x_n f_n(x_1, \ldots, x_n)\,.$$

В обоих этих случаях f_{n+1} согласован с подформулой $Q_n x_n \phi$ и удовлетворяет условию 4 на степень (каждая переменная входит со степенью не больше 2). Далее мы снова применяем операцию $\mathcal{L}$, чтобы получить мультилинейный полином, т.е. строим $n-1$ полиномов:

$$f_{n+2} = \mathcal{L}x_1 f_{n+1}\,, \quad \ldots, \quad f_{2n} = \mathcal{L}x_{n-1} f_{2n-1}\,.$$

Если f_{n+1} линеен по некоторым переменным, соответствующие применения могут быть опущены.

Остальные полиномы последовательности строятся аналогично: f_{2n+1} определяется квантором Q_{n-1}, затем следуют примемения $\mathcal{L}$ и т.д. Легко видеть, что $f_0, \ldots, f_t$ удовлетворяют условиям 1–4, и $t < n^2$. Заметим также, что операции $\mathcal{A}$ и $\mathcal{E}$ применяются только к мультилинейным полиномам.

Представление полиномов арифметическими формулами. Под арифметическими формулами мы понимаем формулы языка формальной арифметики, включающей символы 0, $'$, $+$ и $\cdot$ (см., например, [10]). Термы $0, 0', 0'', \ldots$, представляющие натуральные числа, называются *нумералами*. Формула $t_1 < t_2$ является сокращением для формулы $\exists x(x' + t_1 = t_2)$, где переменная x не входит в термы t_1 и t_2. Ограниченные кванторы $\forall x_{<t} A$ и $\exists x_{<t} A$ определяются как сокращения для формул $\forall x(x < t \ \rightarrow \ A)$ и $\exists x(x < t \ \& \ A)$. Выражения вида a^2 и a^3 обозначают соответственно термы $a \cdot a$ и $a \cdot a \cdot a$. Сравнение $a \equiv b \pmod p$ обозначает арифметическую формулу

$$\exists x_{<a'} \ (a = b + xp) \ \vee \ \exists y_{<b'} \ (b = a + yp),$$

где a и b – термы, p – нумерал. Мы иногда будем писать $a \equiv b$, опуская модуль p, если он будет ясен из контекста. Обозначение $a \not\equiv b$ используется для отрицания этой формулы. Все кванторы, явно выписанные в формулах ниже, подразумеваются ограниченными нумералом p. Однако, чтобы избежать слишком громоздких выражений, мы будем опускать границу, т.е. писать $\forall x$ и $\exists x$ вместо соответственно $\forall x_{<p}$ и $\exists x_{<p}$. Буквы с чертой наверху, например, $\bar{x}_1, \bar{x}_2, \ldots$, будут использоваться для обозначения нумералов.

Пусть $f(x_1, \ldots, x_n)$ – полином над Z/pZ. Говорят, что арифметическая формула $A(x_1, \ldots, x_n, y)$ *представляет* полином f, если для всех натуральных чисел $\bar{x}_1, \ldots, \bar{x}_n, \bar{y}$ выполнены следующие условия:

1. Если $f(\bar{x}_1, ..., \bar{x}_n) \equiv \bar{y} \pmod p$, то формула $A(\bar{x}_1, ..., \bar{x}_n, \bar{y})$ доказуема в арифметике.
2. Если $f(\bar{x}_1, ..., \bar{x}_n) \not\equiv \bar{y} \pmod p$, то формула $\neg A(\bar{x}_1, ..., \bar{x}_n, \bar{y})$ доказуема в арифметике.

Построим арифметические формулы $A_0, \ldots, A_t$, представляющие полиномы $f_0, \ldots, f_t$. Они строятся по индукции: A_0 определяется явно, а каждая следующая формула A_{i+1} определяется в терминах A_i.

Явное выражение для полинома $f_0(x_1, \ldots, x_n)$ не является термом формальной арифметики, так как содержит знак $-$. Однако, оно может быть преобразовано в арифметический терм заменой числа -1 на нумерал, представляющий $p-1$. Пусть $f_0^+(x_1,\ldots,x_n)$ обозначает арифметический терм, получающийся в результате такой замены. Определим A_0 как формулу

$$f_0^+(x_1,\ldots,x_n) \equiv y\,.$$

Очевидно, $A_0(x_1,\ldots,x_n,y)$ представляет полином $f_0(x_1,\ldots,x_n)$. Для того, чтобы определить A_{i+1} при $i \geq 0$, рассмотрим три случая, которые соответствуют операциям $\mathcal{A}$, $\mathcal{E}$ и $\mathcal{L}$.

1. Допустим, что

$$f_{i+1}(x_1,\ldots,x_{m-1}) = \mathcal{A}x_m f_i(x_1,\ldots,x_{m-1},x_m)\,,$$

и $A_i(x_1,\ldots,x_m,y_i)$ представляет f_i. Напомним, что полином f_i мультилинеен. В частности, f_i линеен по x_m, т.е. для любых чисел $\bar{x}_1,\ldots,\bar{x}_{m-1}$ существуют константы $\bar{c}_0$ и $\bar{c}_1$ такие, что

$$f_i(\bar{x}_1,\ldots,\bar{x}_{m-1},x_m) = \bar{c}_1 x_m + \bar{c}_0\,.$$

Опираясь на этот факт, определим формулу $A_{i+1}(x_1,\ldots,x_{m-1},y_{i+1})$ как

$$\exists c_0\ \exists c_1\ (M_1\ \&\ M_2)\,,$$

где M_1 и M_2 обозначают соответственно формулы

$$\forall x_m\ \forall y_i\ (A_i \leftrightarrow (y_i \equiv c_1 x_m + c_0)),$$

$$y_{i+1} \equiv c_0(c_1+c_0).$$

Первый член конъюнкции $M_1 \& M_2$ утверждает, что функция, представленная формулой A_i, согласована с линейной функцией $h(x_m) = \bar{c}_1 x_m + \bar{c}_0$. Второй член определяет значения функции, представленной A_{i+1}, а именно, на аргументах $\bar{x}_1,\ldots,\bar{x}_{m-1}$ значение равно $h(0)\cdot h(1)$.

2. В случае $f_{i+1} = \mathcal{E}x_m f_i$ формула A_{i+1} определяется аналогично:

$$\exists c_0\ \exists c_1\ (M_1\ \&\ M_2)\,,$$

где M_1 обозначает ту же формулу, что и в случае 1, т.е.

$$\forall x_m\ \forall y_i\ (A_i \leftrightarrow (y_i \equiv c_1 x_m + c_0))\,,$$

а M_2 обозначает формулу

$$y_{i+1} + c_0(c_1+c_0) \equiv c_0 + (c_1+c_0)\,.$$

Здесь, в соответствии с определением $\mathcal{E}$, значение y_{i+1} определяется равным $h(0)+h(1)-h(0)h(1)$.

3. Допустим, что A_i представляет f_i и

$$f_{i+1}(x_1,\dots,x_m)=\mathcal{L}x_j f_i(x_1,\dots,x_m),$$

где $1\le j\le m$ (чтобы упростить выражения, рассматриваемые ниже, разберем только случай $j=m$). В отличие от операций $\mathcal{A}$ и $\mathcal{E}$, которые применяются только к мультилинейным полиномам f_i, операция $\mathcal{L}$ может применяться к f_i, содержащим переменные степени 3. Для любых чисел $\bar{x}_1,\dots,\bar{x}_{m-1}$ существуют константы $\bar{c}_0$, $\bar{c}_1$, $\bar{c}_2$ и $\bar{c}_3$ такие, что $f_i(\bar{x}_1,\dots,\bar{x}_{m-1},x_m)$ является кубическим полиномом от переменной x_m:

$$f_i(\bar{x}_1,\dots,\bar{x}_{m-1},x_m)=\bar{c}_3x_m^3+\bar{c}_2x_m^2+\bar{c}_1x_m+\bar{c}_0\,.$$

Поэтому мы определяем $A_{i+1}(x_1,\dots,x_m,y_{i+1})$ как

$$\exists c_0\exists c_1\exists c_2\exists c_3\ (M_1\ \&\ M_2)\,,$$

где M_1 и M_2 обозначают соответственно формулы

$$\forall x_m\ \forall y_i\ (A_i\leftrightarrow(y_i\equiv c_3x_m^3+c_2x_m^2+c_1x_m+c_0)),$$

$$y_{i+1}\equiv c_3x_m+c_2x_m+c_1x_m+c_0\,.$$

Таким образом, последняя формула последовательности A_t содержит только одну свободную переменную y_t и включает все предшествующие формулы $A_0,\dots,A_{t-1}$ в качестве подформул. В [7] показано, что формулы $A_0,\dots,A_t$ представляют полиномы $f_0,\dots,f_t$ (лемма 1).

Арифметические формулы $\alpha(\Phi)$. Используя конструкции определенные выше, можно перевести булевскую формулу с кванторами Φ в арифметическую формулу $\alpha(\Phi)$ следующим образом. Мы переводим Φ в последовательность полиномов $f_0,\dots,f_t$, а затем строим формулы $A_0,\dots,A_t$, представляющие эти полиномы. Формула $\alpha(\Phi)$ определяется как $A_t(1)$, она выражает утверждение о том, что полином f_t тождественно равен 1. Более точно: если Φ истинна, то $A_t(1)$ доказуема в арифметике, а если Φ ложна, то доказуема формула $\neg A_t(1)$. Это следует из того факта, что A_t представляет f_t. Ясно, что построение $\alpha(\Phi)$ по Φ может быть сделано за полиномиальное время.

4.2. Построение исчисления с требуемыми свойствами

Теперь мы определим исчисление $(\mathcal{A}, \mathcal{R})$, которое будет обладать следующими свойствами:

1. Арифметическая формула вида $\alpha(\Phi)$ является теоремой $(\mathcal{A}, \mathcal{R})$ тогда и только тогда, когда она истинна;
2. $(\mathcal{A}, \mathcal{R})$ – исчисление с вероятностно проверяемыми доказательствами.

Аксиомы и правила вывода. В качестве аксиом нашего исчисления возьмем все истинные арифметические формулы вида $a \equiv b \pmod p$, где a и b – термы, не содержащие переменных, p – нумерал. Очевидно, что имеется полиномиальный по времени алгоритм $\mathcal{A}$, который определяет, является ли формула такого вида истинной.

Правила вывода определяются тремя схемами. Первая схема определяет однопосылочные правила, которые позволяют вводить квантор существования в заключение:

$$\frac{F(\bar{x})}{\exists x_{<p} F(x)}, \tag{1}$$

где F – произвольная арифметическая формула, а $\bar{x}$ и p – числа, удовлетворяющие неравенству $\bar{x} < p$ и представленные нумералами. Вторая схема задает правила

$$\frac{F_1}{F_1 \ \& \ F_2}, \tag{2}$$

где F_1 – произвольная арифметическая формула, а F_2 – аксиома нашего исчисления, т.е. истинная формула вида $a \equiv b \pmod p$.

Правила, определяемые третьей схемой, играют ключевую роль в нашем исчислении. Они оперируют формулами вида A_i, определенными выше. Напомним, что A_i имеет свободные переменные $x_1, \ldots, x_m, y_i$ (список $x_1, \ldots, x_m$ пуст в случае последней формулы последовательности). Рассмотрим формулу A_i и подстановку σ, которая приписывает некоторые числа всем переменным $x_1, \ldots, x_m$ кроме x_j. Подставляемые числа предполагаются меньшими, чем соответствующий модуль p. Результат применения такой подстановки к формуле A_i будет обозначаться через $A_i^\sigma(x_j, y_i)$. Третья схема имеет вид:

$$\frac{A_i^\sigma(0, h(0)) \qquad A_i^\sigma(1, h(1)) \qquad \ldots \qquad A_i^\sigma(p-1, h(p-1))}{\forall x_{<p} \ \forall y_{<p} \ (A_i^\sigma(x, y) \ \leftrightarrow \ y \equiv h(x))}, \tag{3}$$

где $h(x)$ в заключении обозначает любой полином вида $\bar{c}_1 x + \bar{c}_0$ или $\bar{c}_3 x^3 + \bar{c}_2 x^2 + \bar{c}_1 x + \bar{c}_0$ с числовыми коэффициентами, а $h(0), \dots, h(p-1)$ в посылках обозначают значения этого полинома, меньшие p и представленные нумералами. Эти правила будут использованы для удаления квантора всеобщности перед вхождением A_i в формулу A_{i+1}.

Ясно, что имеется разрешающий алгоритм для правил всех трех типов, полиномиально ограниченный по времени. Каждое правило в $(\mathcal{A}, \mathcal{R})$ удовлетворяет условию корректности: если посылки истинны, то и заключение истинно. Так как аксиомы исчисления истинны, то все теоремы $(\mathcal{A}, \mathcal{R})$ являются истинными арифметическими формулами.

Условие корректности правил можно переформулировать таким образом: если заключение ложно, то хотя бы одна из посылок ложна. Важно отметить, что правила (3) обладают более сильным свойством, а именно: если заключение ложно, то все посылки, возможно за исключением не более трех, тоже ложны. Действительно, так как формулы A_i представляют полиномы степени три или меньше, заключение эквивалентно утверждению о тождественном равенстве двух кубических полиномов над Z/pZ. Каждая посылка $A_i^\sigma(k, h(k))$ эквивалентна утверждению о равенстве этих двух полиномов в точке k. Если два кубических полинома различны, то они могут совпадать не более, чем в трех точках, т.е. только три посылки могут быть истинны, а остальные должны быть ложны.

Это свойство правил третьрго типа показывает, что их посылки *избыточны*: для того, чтобы доказать истинность заключения, нет необходимости доказывать все посылки – достаточно доказать только четыре из них. Именно это свойство избыточности и позволяет нам проверять доказательства в $(\mathcal{A}, \mathcal{R})$ без чтения всего доказательства целиком.

Доказательства для истинных формул вида $\alpha(\Phi)$**.** Пусть формула $\alpha(\Phi)$ истинна. Пусть $f_0, \dots, f_t$ – соответствующая последовательность полиномов над Z/pZ, где p – наименьшее простое число, большее $6n^2$. Эти полиномы представляются формулами $A_0, \dots, A_t$. Покажем, что $A_t(1)$ доказуема в $(\mathcal{A}, \mathcal{R})$. По существу ее доказательство будет формализацией интерактивного доказательства для $f_t = 1$, предложенного А. Шенем в [14].

Доказательство строится “снизу вверх”, т.е. от теоремы к аксиомам. Так как последний полином f_t строится из f_{t-1} посредством операций $\mathcal{A}$ или $\mathcal{E}$, формула $A_t(1)$ имеет вид $\exists c_0 \exists c_1 F(c_0, c_1)$. Поэтому она может быть выведена по правилам вида (1) из формулы $F(\bar{c}_0, \bar{c}_1)$, где $\bar{c}_0$ и $\bar{c}_1$ – некоторые числа. А именно, мы выбираем числа $\bar{c}_0$ и $\bar{c}_1$ равными коэффициентам полинома f_{t-1}. Формула $F(\bar{c}_0, \bar{c}_1)$ имеет вид $F_1 \& F_2$, где F_1 – это формула вида

$$\forall x_m \ \forall y_{t-1} \ (A_{t-1} \leftrightarrow (y_t \equiv c_1 x_m + c_0)),$$

а F_2 – это либо сравнение $1 \equiv \bar{c}_0(\bar{c}_1 + \bar{c}_0)$ (в случае операции $\mathcal{A}$), либо сравнение $1 + \bar{c}_0(\bar{c}_1 + \bar{c}_0) \equiv \bar{c}_0 + (\bar{c}_1 + \bar{c}_0)$ (в случае операции $\mathcal{E}$). В силу выбора чисел $\bar{c}_0, \bar{c}_1$ и предположения $f_t = 1$ это сравнение должно быть истинным. Поэтому конъюнкция $F_1 \& F_2$ выводима по правилу (2) из F_1. Так как формула F_1 имеет тот же вид, что и заключение правил вида (3), то F_1 может быть выведена из посылок

$$A_{t-1}(0, h_0) \qquad A_{t_1}(1, h_1) \qquad \ldots \qquad A_{t_1}(p-1, h_{p-1}),$$

где $h_0, \ldots, h_{p-1}$ обозначают значения полинома $\bar{c}_1 x_m + \bar{c}_0$ на $0, \ldots, p-1$ соответственно.

Подобно формуле $A_t(1)$, каждая посылка $A_{t-1}(k, h_k)$ формализует утверждение $f_{t-1}(k) = h_k$. Также, подобно $A_t(1)$, эти формулы не содержат переменных и имеют вид $\exists c_0 \exists c_1 F(c_0, c_1)$ или, если f_{t-1} строится линеаризацией, вид $\exists c_0 \ldots \exists c_3 F(c_0, \ldots, c_3)$. Поэтому мы можем выводить их аналогичным образом. Вначале мы удаляем внешние кванторы существования в $A_{t-1}(k, h_k)$, используя правила (1) и подставляя в качестве констант коэффициенты полинома f_{t-2}. Удаление кванторов и подстановка приводят к конъюнкциям, в которых второй член является истинным сравнением и может быть удален. Первый член начинается с кванторов всеобщности и может быть выведен по правилу (3) из p посылок.

Такой вывод продолжается до тех пор, пока все кванторы не будут удалены, и мы не получим формулы $A_0^\sigma(k, h_k)$, не содержащие свободных переменных. Эти формулы являются аксиомами, так как они представляют истинные сравнения вида $f_0(\bar{x}_1, \ldots, \bar{x}_n) \equiv \bar{y}$). Таким образом, мы получили доказательство $A_t(1)$.

Замечание. Те же рассуждения показывают, что все истинные формулы вида

$$\forall x_{<p}\, \forall y_{<p}\ (A_i^\sigma(x,y)\ \leftrightarrow\ y \equiv h(x)).$$

также являются теоремами $(\mathcal{A},\mathcal{R})$. Этот факт будет использован ниже.

Условия полноты и корректности для $(\mathcal{A},\mathcal{R})$. 1. *Полнота.* Нетрудно видеть, что для любого доказательства в $(\mathcal{A},\mathcal{R})$ максимальная длина пути, максимальная степень и максимальная длина слова ограничены полиномом от длины теоремы. Действительно, все правила уменьшают длину посылок по сравнению с длиной заключения, и, следовательно, длины путей и слов не превосходят длину теоремы. Для того, чтобы убедиться, что максимальная степень ограничена полиномом от длины теоремы, достаточно установить полиномиальную границу от n, т.е. от числа переменных в исходной булевской формуле. Такая граница обеспечивается нашим выбором p как наименьшего простого числа, большего $6n^2$. Число p не превосходит $12n^2$, так как для любого $k > 1$ имеется простое число в интервале $[k, 2k]$. Таким образом, $(\mathcal{A},\mathcal{R})$ обладает свойством полноты.

2. *Корректность.* Теперь мы доказываем, что $(\mathcal{A},\mathcal{R})$ удовлетворяет и второму условию из определения исчислений с вероятностно проверяемыми доказательствами. Рассмотрим слово, которое не является теоремой $(\mathcal{A},\mathcal{R})$. Рассмотрим произвольный кандидат в доказательства для этого слова и выберем случайный путь $v_1,\dots,v_l$. Какова вероятность, что этот путь правильный? Достаточно говорить только о правильных путях, так как рассуждения в предыдущем абзаце показывают, что любой правильный путь в $(\mathcal{A},\mathcal{R})$ ограничен некоторым полиномом.

Пусть $s_1,\dots,s_l$ – слова, приписанные вершинам $v_1,\dots,v_l$. По предположению s_1 не является теоремой, а путь $v_1 \dots, v_l$ правильный. Тогда путь содержит хотя бы одно применение многопосылочного правила (если правильный путь состоит только из однопосылочных правил, то s_1 должно быть теоремой). Пусть k – наименьший индекс, для которого s_k является заключением многопосылочного правила вида (3). Слово s_k является арифметической формулой вида

$$\forall x_{<p}\, \forall y_{<p}\ (A_i^\sigma(x,y)\ \leftrightarrow\ y \equiv h(x)).$$

Здесь числа i и p удовлетворяют условию $i < p/6$, потому что по определению формул $A_0,\dots,A_t$ имеют место неравенства $i <$

n^2 и $p > 6n^2$. Из правильности пути следует также, что каждое последующее применение многопосылочных правил содержит p посылок. Число таких применений не превосходит i.

Формула s_1 получается из s_k применениями однопосылочных правил. Так как s_1 не является теоремой, то s_k также недоказуема в $(\mathcal{A}, \mathcal{R})$. Более того, как следует из замечания выше, s_k является ложной. Таким образом, формула s_k ложна, в то время как заключительная формула s_l истинна.

Мы говорим, что переход от s_j к s_{j+1} *приводит к ошибке*, если выполняются следующие условия:

1. Слова s_j и s_{j+1} являются соответственно заключением и одной из посылок правила вида (3).
2. Формула s_j ложна, а s_{j+1} истинна.

Нетрудно видеть, что путь $s_1, \dots, s_l$ может быть правильным, только если в нем имеется хотя бы один переход, приводящий к ошибке. Найдем верхнюю границу для вероятности того, что путь содержит такой переход. Напомним, что правила (3) обладают свойством: если заключение ложно, то все посылки, возможно за исключением не более трех, также ложны. Следовательно, данный переход от s_j к s_{j+1} приводит к ошибке с вероятностью, не большей $3/p$. Так как выполняется неравенство $i < p/6$, то число всех переходов меньше $p/6$. Поэтому вероятность меньше, чем $(3/p)(p/6) = 1/2$. Таким образом, путь $s_1, \dots, s_l$ является правильным с вероятностью меньшей $1/2$.
$\square$

Литература

1. S. Arora and C. Lund, *Hardness of approximation.* In D. Hochbaum, editor, *Approximation algorithms for NP-hard problems*, chapter 10. PWS Publishing Company, Boston (1997) (to appear).
2. S. Arora and S. Safra, *Probabilistic checking of proofs: A new characterization of NP.* In *Proceedings of the 33rd Annual IEEE Symposium on Foundations of Computer Science*, (1992) 2–13.
3. L. Babai, *Transparent proofs and limits to approximation.* In *Proceedings of the First European Congress of Mathematicians, Paris, 1992, vol. 1*, 31–92. Birkhäuser Verlag, 1994.
4. L. Babai, L. Fortnow, L.A. Levin, and M. Szegedy, *Checking computations in polylogarithmic time.* In *Proceedings of the 23rd Annual ACM Symposium on the Theory of Computing*, (1991) 21–31.
5. L. Babai, L. Fortnow, and C. Lund, *Non-deterministic exponential time has two-prover interactive protocols. Computational Complexity*, 1(1): (1991) 3–40;

(An early version appeared in *Proc. 31st FOCS*, (1990), 16–25..

6. Е. Данцин, *Доказательства в арифметике, использующие случайные числа.* — Зап. научн. семин. ПОМИ **220** (1995), 49–71.
7. E. Dantsin, *A logic-style version of interactive proofs.* Technical Report UPMAIL Technical Report No.124, Computing Science Department, Uppsala University, March 1996.
8. O. Goldreich, *A taxonomy of proof systems.* In L. A. Hemaspaandra and A. Selman, editors, *Complexity theory retrospective II.* 1996.
9. C. Lund, L. Fortnow, H. Karloff, and N. Nisan, *Algebraic methods for interactive proof systems.* — J. of the ACM **39(4)** (1992), 859–868 (An early version appeared in *Proc. 31st FOCS*, 2–10, 1990).
10. E. Mendelson, *Introduction to mathematical logic.* Van Nostrand, 1964; Русский перевод: Э. Мендельсон, Введение в математическую логику, М., Наука, 1971.
11. R. Motwani and P. Raghavan, *Randomized algorithms.* Cambridge University Press, 1995.
12. C. Papadimitriou, *Computational Complexity.* Addison-Wesley, 1994.
13. A. Shamir, *IP=PSPACE.* — J. of the ACM **39(4)** (1992), 869–877 (An early version appeared in *Proc. 31st FOCS*, 11-15, 1990).
14. A. Shen, *IP=PSPACE: Simplified Proof.* — J. of the ACM **39(4)** (1992), 878–880.

Dantsin E. Probabilistic verification of proofs in calculuses.

The paper defines a special kind of probabilistically checkable proof system, namely, probabilistically checkable proof calculuses (PCP calculuses). A proof in such a calculus can be verified with enough confidence by examining only one random path in the proof tree, without reading the whole proof. The verification procedure just checks all applications of inference rules along the path; its running time is supposed to be polynomial in the theorem length. It is shown that the deductive power of PCP calculuses is characterized as follows: (i) the decision problem for theorems is in PSPACE for all PCP calculuses, and (ii) this problem is PSPACE-hard for some of them.

С.-Петербургское отделение
Математического института
им. В. А. Стеклова РАН

Поступило 19 августа 1997 г.