УЧЕНЫЕ ЗАПИСКИ КАЗАНСКОГО УНИВЕРСИТЕТА
Том 156, кн. 3 Физико-математические науки *2014*

УДК 519.7

СЛОЖНОСТЬ ВЕТВЯЩИХСЯ ПРОГРАММ ДЛЯ ЧАСТИЧНО ОПРЕДЕЛЕННЫХ ФУНКЦИЙ

А.Ф. Гайнутдинова

Аннотация

Упорядоченные ветвящиеся диаграммы решений (OBDD – Ordered Binary Decision Diagrams) – известная модель для вычисления булевых функций. В работе рассмотрены частично определенные булевы функции и сложность их вычисления в различных вариантах OBDD (детерминированных, недетерминированных, вероятностных и квантовых). В качестве исследуемой меры сложности взята ширина OBDD. Известно, что при рассмотрении всюду определенных булевых функций вероятностные OBDD могут быть эффективнее детерминированных и данный разрыв в сложности может быть не более чем экспоненциальным. Аналогичный результат верен и при сравнении квантовых OBDD с классическими. Показано, что для частично определенных функций преимущество в сложности квантовых моделей перед классическими может быть усилено. Предложена частично определенная булева функция, которая вычислима с нулевой ошибкой квантовой OBDD ширины 2. При этом ширина классических детерминированной, вероятностной OBDD экспоненциально зависит от параметра функции. Предложено также бесконечное семейство частично определенных булевых функций таких, что любая функция из данного семейства вычислима с ограниченной ошибкой вероятностной OBDD ширины 2. При этом существует бесконечное подмножество функций из данного семейства таких, что ширина классической детерминированной OBDD для вычисления каждой функции из данного подмножества возрастает неограниченно.

Ключевые слова: упорядоченные ветвящиеся диаграммы решений, частично определенные функции, квантовые вычисления, вероятностные OBDD, детерминированные OBDD, сложность.

Введение

Сравнительный анализ различных моделей по их вычислительной мощности – важная задача математической кибернетики. Вычислительные модели можно подразделять на модели без памяти (схемы), модели с конечной памятью (автоматы, ветвящиеся программы), модели с бесконечной памятью (машины Тьюринга) [1]. Особый интерес представляет сравнение вариантов одной и той же модели, различающихся способом функционирования, а именно детерминированного, недетерминированного, вероятностного и квантового. Задача сравнительного анализа вероятностного и детерминированного вариантов различных моделей исследовалась в ряде работ (см., например [2–4]). В конце 80-х годов XX в. стала активно развиваться теория квантовых вычислений [5–7]. Интерес к этой области усилился после того, как были построены эффективные квантовые алгоритмы для решения ряда задач, для которых на сегодняшний день неизвестны эффективные классические (детерминированные, вероятностные) алгоритмы. В числе таких эффективных квантовых алгоритмов можно отметить квантовый полиномиальный алгоритм Шора факторизации числа, алгоритм Гровера поиска в неупорядоченной базе данных [8, 9]. Для известных классических моделей (машин Тьюринга, автоматов,

схем, ветвящихся программ и т. д.) были определены их квантовые аналоги и стали исследоваться их вычислительные возможности. Так, было показано, что квантовые конечные автоматы, использующие одно измерение, способны распознавать собственное подмножество регулярных языков. При этом были найдены примеры языков, для которых квантовый автомат оказался экспоненциально эффективнее классических (детерминированного, вероятностного). Было также показано, что максимальная экономия сложности, которую можно достичь на квантовых автоматах по сравнению с детерминированными, не может превышать экспоненту.

В последние годы для конечных автоматов стала активно исследоваться задача, являющаяся обобщением задачи распознавания языка [10, 11], а именно задача отделимости, где вместо всего множества слов заданного алфавита рассматривается лишь некоторое подмножество слов, входами для модели являются слова только из заданного подмножества. Таким образом, объединение языка и его дополнения составляет данное подмножество (вместо множества всех слов алфавита). Оказалось, что при рассмотрении проблем отделимости квантовые автоматы могут быть эффективнее классических более чем экспоненциально.

Ветвящиеся программы (BP – Branching Program) – известная модель для вычисления булевых функций [12], которая хорошо моделирует компьютерные программы, записанные с использованием операторов *if ... then ... else, goto*. Известно, что данный набор операторов составляет полный базис (используя только их можно вычислить произвольную булеву функцию). Впервые модель ветвящихся программ была предложена Ли в 1959 г. [13] и позднее Акерсом [14] как структура данных для переключательных функций. Первыми работами в области ветвящихся программ были [13, 15, 16]. С тех пор были определены и интенсивно исследуются различные варианты базовой модели. Модель BP имеет различные приложения: верификация моделей и программ, базы данных и др. Кроме того, ветвящиеся программы являются удобной моделью представления функций, позволяющей получать высокие нижние оценки. О соотношении модели ветвящихся программ с другими моделями вычислений можно сказать следующее. Известно, что логарифм сложности ветвящейся программы соответствует объему памяти машины Тьюринга, а максимальная длина вычислительного пути – времени вычисления [17, 18]. Известно также, что для любой функции f справедливы неравенства $C(f) \leq 3BP(f)$, $BP(f) \leq L_\Omega(f) + 1$, где $BP(f)$ – минимальная сложность ветвящейся программы, вычисляющей функцию f, $C(f)$ и $L_\Omega(f)$ – соответственно минимальные сложность схемы из функциональных элементов и сложность формулы для функции f [12]. Наиболее высокая нижняя оценка для явно заданной булевой функции была получена Э.И. Нечипоруком в 1966 г. [30]. Для модели ветвящихся программ эту оценку переложил П. Пудлак в 1984 г. [19].

Ветвящиеся программы, в которых на каждом вычислительном пути переменные считываются не более одного раза в одном и том же порядке, называются упорядоченными ветвящимися диаграммами решений (OBDD – Ordered Binary Decision Diagrams). Если порядок считывания переменных в такой модели совпадает с естественным, то ее можно рассматривать как модель неоднородных автоматов с переменной структурой. Преобразования в такой модели могут различаться на каждом шаге. Поскольку длина OBDD не превосходит длины входа, естественной мерой сложности в этом случае является ширина OBDD, что является аналогом количества состояний для автоматов. Известно, что почти все функции имеют экспоненциальную сложность вычисления в модели OBDD. При этом сложность представления функций в OBDD зависит от используемого программой порядка считывания переменных. Так, существуют функции, вычислимые OBDD ширины 2, но при использовании наихудшего порядка считывания переменных ширина OBDD

для этих функций имеет экспоненциальное значение. Известны функции, которые имеют экспоненциальную сложность представления в модели OBDD независимо от используемого программой порядка считывания переменных.

Вероятностные ветвящиеся программы были впервые определены в [4]. В этой работе было показано, что вероятностные OBDD могут быть экспоненциально эффективнее детерминированных и недетерминированных OBDD. Известно, что экспоненциальное преимущество вероятностных OBDD перед детерминированными является максимально возможным. Экспоненциальная нижняя оценка сложности вероятностной OBDD для явно заданной функции была доказана в работе [20]. Экспоненциальная нижняя оценка на размер OBDD для функции умножения была получена в [21].

Квантовый аналог классической ветвящейся программы был впервые определен в [22], где модель определялась как последовательность унитарных эволюций квантовой системы с заключительным измерением как процедурой извлечения результата вычислений. Известная модель перестановочных ветвящихся программ, рассматриваемая в [23], является частным случаем такой модели. В работе [23] было показано, что класс функций, вычислимых перестановочными BP полиномиальной сложности, в точности совпадает с классом NC_1 функций, представимых схемами из функциональных элементов логарифмической глубины полиномиальной сложности [23]. В работах [24, 25] были определены несколько иные модели квантовой BP. Было доказано, что все эти модели эквивалентны. В области сравнительного анализа квантовых и классических OBDD известны следующие результаты. Было показано, что квантовые OBDD могут быть экономнее детерминированных не более чем экспоненциально [22]. Были приведены примеры функций, для которых данная эффективность достижима. Так, для симметрической функции MOD_p, принимающей значение 1 только на наборах, в которых число единиц кратно p, где p – простое число, было показано, что она вычислима с ограниченной ошибкой квантовой OBDD ширины $O(\log p)$. Детерминированная OBDD и вероятностная стабильная OBDD, вычисляющая с ограниченной ошибкой, требуют ширины p для вычисления функции MOD_p [26]. Стабильность для OBDD означает, что преобразования не зависят от номера шага, на котором они применяются. Все перечисленные выше результаты были сформулированы для всюду определенных функций.

В настоящей работе рассматриваются частично определенные функции, то есть функции, которые определены не на всем множестве аргументов, а только на некотором его подмножестве A. При вычислении таких функций значения, которые получаются на входах, не принадлежащих множеству A, несущественны и могут быть произвольными. Рассмотрение частично определенных функций может быть связано с различными аспектами. Например, в реальном вычислительном устройстве некоторые наборы значений входов могут вообще не встречаться и поведение устройства на таких входах является несущественным. Функционирование этих устройств может быть описано частично определенными булевыми функциями [1, § 2.1]. Еще одной мотивацией для исследования частично определенных функций является то, что задача вычисления таких функций ветвящимися программами аналогична задаче отделимости для автоматов. Принимая во внимание результаты, имеющие место для конечных автоматов, распознающих проблемы отделимости, можно ожидать аналогичных результатов для ветвящихся программ, вычисляющих частично определенные функции, превосходящих соответствующие результаты для всюду определенных функций. При этом следует отметить, что техника доказательств, которая используется для моделей автоматов, в большинстве случаев не может быть напрямую применена для модели OBDD в силу различия

данных моделей. В отличие от автоматной модели, в модели OBDD переменные входного слова могут считываться в произвольном порядке, и преобразования могут быть различными на каждом шаге вычисления. Данные возможности дают больше вычислительной мощности ветвящимся программам по сравнению с конечными автоматами.

В работе представлены следующие результаты для OBDD, вычисляющих частично определенные функции. Показано, что квантовые OBDD, вычисляющие частично определенные функции, могут быть эффективнее классических детерминированных и вероятностных и отличие в сложности может быть более чем экспоненциальным.

В разд. 1 приведены определения и необходимые для изложения факты. В разд. 2 исследуются основные свойства частично определенных булевых функций. В разд. 3 приводятся известные результаты по сравнительной сложности квантовых и классических OBDD, вычисляющих всюду определенные функции. Показано, что при вычислении указанных функций квантовые OBDD могут быть экспоненциально эффективнее классических детерминированных и вероятностных OBDD; экспоненциальное преимущество квантовых моделей перед классическими является максимально возможным. В разд. 4 рассматриваются частично определенные булевы функции. Установлено, что для частично определенных функций преимущество в сложности квантовых OBDD может более чем экспоненциальным. Введена частично определенная булева функция $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$, зависящая от параметра k, заданная только на наборах, в которых число единиц кратно 2^k. Указанная функция определена на основе семейства унарных проблем отделимости (promise problems), рассмотренных в работе [11], где показано, что это семейство может распознаваться без ошибки квантовым автоматом с двумя состояниями. Классический детерминированный автомат требует не менее 2^{k+1} состояний.

Показано, что частично определенная функция $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ вычислима без ошибки квантовой OBDD ширины 2. Получены нижние оценки 2^{k+1} ширины детерминированной OBDD, вычисляющей $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$, и ширины стабильной вероятностной OBDD, вычисляющей $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ с ограниченной ошибкой.

В разд. 5 рассмотрена коммуникационная модель вычисления. Доказательство нижних оценок сложности вычисления всюду определенных функций в модели OBDD часто проводится с привлечением аппарата коммуникационных вычислений. Нами получена верхняя оценка вычисления частично определенной функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ в односторонней коммуникационной модели.

В разд. 6 предложена частично определенная булева функция $\mathtt{f}^{\mathtt{p},\varepsilon}_{\mathtt{n}}$, вычислимая с ограниченной ошибкой вероятностной OBDD ширины 2. Данная функция определена на основе унарного семейства проблем отделимости, рассмотренного в [27]. Доказано, что ширина детерминированной OBDD, вычисляющей функцию $\mathtt{f}^{\mathtt{p},\varepsilon}_{\mathtt{n}}$, растет неограниченно с уменьшением параметра функции.

1. Основные определения

Определение 1. Детерминированная ветвящаяся программа над множеством переменных $X = \{x_1, \ldots, x_n\}$ – это ориентированный ациклический граф с финальными вершинами, помеченными 0 и 1 (будем называть их, соответственно, отвергающими и принимающими). Каждая внутренняя вершина помечена булевой переменной $x \in X$ и имеет два исходящих ребра, помеченных 0 и 1 соответственно. ВР представляет булеву функцию $f : \{0,1\}^n \to \{0,1\}$ следующим образом. Вычисление значения $f(\sigma)$ для входного набора $\sigma \in \{0,1\}^n$ начинается из выделенной начальной вершины. Для каждой внутренней вершины, помеченной переменной x_j, осуществляется переход из этой вершины либо по 0-ребру, либо по 1-ребру

в соответствии со значением σ_j, которое принимает переменная x_j во входном наборе. Значение функции f для входа σ – это значение достигнутой финальной вершины.

Сложность $\mathrm{Size}(P)$ ветвящейся программы P – это количество ее внутренних вершин.

Длина $\mathrm{Length}(P)$ ветвящейся программы P – это количество ребер в самом длинном пути из начальной вершины в конечную.

Длина ВР очевидным образом оценивает время, требуемое для вычисления функции f в худшем случае. Логарифм сложности ВР оценивает память, затрачиваемую в процессе вычисления.

Ветвящаяся программа называется *один раз считывающей*, если на любом пути из начальной вершины в финальную каждая переменная считывается не более одного раза.

Ветвящаяся программа называется *уровневой*, если ее вершины могут быть разбиты на уровни $0, 1, \ldots$ таким образом, что ребра из вершин уровня i ведут только в вершины уровня $(i+1)$ для каждого i.

Ширина $\mathrm{Width}(P)$ уровневой ВР P – это максимум от числа вершин на уровне, взятый по всем уровням программы P.

Ясно, что $\mathrm{Size}(P) \leq \mathrm{Length}(P) \cdot \mathrm{Width}(P)$.

Уровневая ВР P называется *забывающей*, если во всех вершинах одного уровня P считывается одна и та же переменная.

OBDD (Ordered Binary Decision Diagram) – это уровневая, забывающая, один раз считывающая ветвящаяся программа.

Поскольку для модели OBDD ее длина не превосходит n, естественной мерой сложности в этом случае является ширина OBDD.

Вероятностная OBDD (POBDD – Probabilistic OBDD) является обобщением детерминированной модели. Для простоты можем считать, что все уровни содержат одинаковое число вершин, пронумерованных от 1 до d. POBDD над множеством переменных $X = \{x_1, \ldots, x_n\}$ ширины d определяется следующим образом:

$$P_n = (v_0, T, \mathit{Accept}).$$

Здесь v_0 – d-мерный стохастический вектор-столбец – вектор распределения вероятностей вершин на начальном нулевом уровне, $T = \{\langle j_i,\ A_i(0),\ A_i(1)\rangle\}_{i=1}^{n}$ – последовательность d-мерных стохастических преобразований, где $A_i(0)$, $A_i(1)$ – стохастические по столбцам $(d \times d)$-матрицы, $\mathit{Accept} \subseteq \{1, \ldots, d\}$ – множество принимающих вершин. На i-м шаге, $i = 1, \ldots, n$, программа P_n считывает значение входной переменной $x_{j_i} = \sigma_{j_i}$ и преобразует текущий вектор распределения вероятностей v_{i-1} в вектор $v_i = A_i(\sigma_{j_i})$. После считывания входного набора $\sigma = \sigma_1, \ldots, \sigma_n$ финальный вектор имеет вид $v_n(\sigma) = A(\sigma)v_0$, где $A(\sigma) = A(\sigma_n) \cdots A(\sigma_1)$. Пусть $v_n(\sigma) = (p_1, \ldots, p_d)$. Программа P_n завершает вычисление и принимает входной набор с вероятностью

$$Pr_{accept}^{P_n}(\sigma) = \sum_{i \in Accept} p_i.$$

Детерминированная OBDD может быть определена как частный случай POBDD, когда элементы начального вектора распределения вероятностей и матриц преобразований являются нулями или единицами.

Недетерминированная OBDD (NOBDD) позволяет на каждом шаге при считывании переменной переходить из вершины текущего уровне в более чем одну вершину последующего уровня. Таким образом, для входного набора σ могут существовать несколько вычислительных путей. Недетерминированная OBDD P_n

принимает входной набор σ, если существует вычислительный путь, соответствующий данному набору, который завершается в принимающей вершине. В противном случае P_n отвергает набор σ. Эквивалентное определение недетерминированной модели можно дать, основываясь на определении вероятностной OBDD. Недетерминированная OBDD, вычисляющая функцию f – это POBDD, которая принимает все наборы $\sigma : f(\sigma) = 1$ с вероятностью $Pr_{accept}(\sigma) > 0$ и наборы $\sigma : f(\sigma) = 0$ с вероятностью $Pr_{accept}(\sigma) = 0$.

Прежде чем перейти к определению квантовой OBDD, приведем основные понятия квантовых вычислений, необходимые для дальнейшего изложения. Более подробную информацию о квантовых вычислениях можно найти, например, в [7]. Квантовая система (QS) с d устойчивыми состояниями может быть описана при помощи d-мерного комплекснозначного гильбертова пространства $\mathcal{H}^d$ с базисом $\mathcal{B} = \{|q_1\rangle, \ldots, |q_d\rangle\}$, где $|q_j\rangle \in \mathcal{H}^d$ – вектор-столбец (кет-вектор), содержащий 1 в j-й позиции и нули во всех остальных. Состояния $q_1, \ldots, q_d$ называются устойчивыми состояниями и могут рассматриваться как классические состояния системы. В процессе вычислений квантовая система QS может находиться в суперпозиции своих устойчивых состояний. Чистое состояние QS (обозначается как $|\psi\rangle = (z_1, \ldots, z_d)$) – это вектор пространства $\mathcal{H}^d$ с нормой 1 (унитарный вектор): $\sqrt{\langle\psi|\psi\rangle} = 1$. В суперпозиции $|\psi\rangle$ каждое устойчивое состояние q_i представлено с амплитудой z_i, то есть квантовая система находится одновременно во всех своих устойчивых состояниях, в каждом с соответствующей амплитудой. Унитарная эволюция – это изменение состояния квантовой системы за определенный период времени, которое описывается d-мерной унитарной матрицей U. Матрица U называется унитарной, если $U \cdot U^{\dagger} = I$, где $U^{\dagger}$– транспонированная комплексно сопряженная к U матрица, I – единичная матрица. Измерение квантовой системы QS – это процедура извлечения результата вычисления. При измерении QS, находящейся в состоянии $|\psi\rangle = (z_1, \ldots, z_d)$, результатом измерения является состояние q_i с вероятностью $|z_i|^2$.

Квантовая OBDD (QOBDD) на множестве переменных $X = \{x_1, \ldots, x_n\}$, определенная на QS с d устойчивыми состояниями, имеет вид

$$Q_n = \big(|\psi_0\rangle, T, Accept\big).$$

Здесь

- $|\psi_0\rangle$ – начальный унитарный вектор;
- $T = \{\langle j_i, U_i(0), U_i(1)\rangle\}_{i=1}^{n}$ – последовательность унитарных преобразований, $U_i(0)$ и $U_i(1)$– $(d \times d)$-унитарные матрицы;
- $Accept \subseteq \{1, \ldots, d\}$ – множество принимающих состояний.

Процесс вычисления программы Q_n на входном наборе $\sigma = \sigma_1, \ldots, \sigma_n$ аналогичен вычислению в вероятностном случае. Другим образом определяется вероятность принятия входного набора. Пусть $|\psi(\sigma)\rangle = (z_1, \ldots, z_n)$ – финальный вектор распределения амплитуд состояний программы после считывания σ. Программа Q_n принимает входной набор с вероятностью

$$Pr_{accept}^{Q_n}(\sigma) = \sum_{i \in Accept} |z_i|^2.$$

OBDD называется *стабильной*, если преобразования, применяемые на каждом шаге, не зависят от номера шага.

Пусть P – вероятностная (квантовая) OBDD над множеством переменных $X = \{x_1, \ldots, x_n\}$. Будем говорить, что P вычисляет булеву функцию $f(x_1, \ldots, x_n)$ *с ограниченной ошибкой*, если существует константа $\varepsilon \in (0, 1/2]$ такая, что $\Pr_{accept}^{P}(\sigma) \geq 1/2 + \varepsilon$, если $f(\sigma) = 1$, и $\Pr_{accept}^{P}(\sigma) \leq 1/2 - \varepsilon$, если $f(\sigma) = 0$.

Пусть P – квантовая OBDD над множеством переменных $X = \{x_1, \ldots, x_n\}$. Будем говорить, что P вычисляет булеву функцию f *без ошибки*, если $\Pr^P_{accept}(\sigma) = 1$ при $f(\sigma) = 1$, и $\Pr^P_{accept}(\sigma) = 0$ при $f(\sigma) = 0$. Будем говорить, что P *недетерминированно* вычисляет функцию f, если $\Pr^P_{accept}(\sigma) > 0$ при $f(\sigma) = 1$, и $\Pr^P_{accept}(\sigma) = 0$ при $f(\sigma) = 0$.

Обозначим через $\mathsf{OBDD}^{\mathsf{d}}$, $\mathsf{POBDD}^{\mathsf{d}}_{1/2+\varepsilon}$, $\mathsf{QOBDD}^{\mathsf{d}}_{\mathsf{exact}}$ множество булевых функций, вычислимых соответственно детерминированными, вероятностными с ограниченной ошибкой, квантовыми с нулевой ошибкой OBDD ширины d.

Определение 2. Булева функция $f_n : \{0,1\}^n \to \{0,1\}$ называется частично определенной, если $f_n^{-1}(1) \cup f_n^{-1}(0) \neq \{0,1\}^n$.

2. Всюду определенные и частично определенные функции

Пусть $f : \{0,1\}^n \to \{0,1\}$ – всюду определенная функция, $A \subset \{0,1\}^n$, $g : A \to \{0,1\}$ – частично определенная функция.

Будем называть функцию f доопределением функции g с множества A до множества $\{0,1\}^n$, если функция f совпадает с функцией g на множестве A. Соответственно, функцию g будем называть сужением функции f до множества A и обозначать $g = \mathrm{Part}_{\mathrm{A}}(f)$, то есть

$$g = \mathrm{Part}_{\mathrm{A}}(f) \Leftrightarrow g(\sigma) = f(\sigma) \quad \text{для любых} \quad \sigma \in A.$$

Теорема 1. *Пусть $f : \{0,1\}^n \to \{0,1\}$ – произвольная всюду определенная функция, вычислимая ветвящейся программой сложности S. Тогда для любого $A \subseteq \{0,1\}^n$ любая частично определенная функция $g : A \to \{0,1\}$ такая, что $g = \mathrm{Part}_{\mathrm{A}}(f)$, вычислима ветвящейся программой сложности не более чем S.*

Доказательство. Пусть P_n – ветвящаяся программа, вычисляющая функцию f. Поскольку g – частично определенная функция, входы множества $\{0,1\}^n \setminus A$ являются несущественными для g и при вычислении значения, выдаваемые для данных входов, могут быть произвольными. Поскольку $g = \mathrm{Part}_{\mathrm{A}}(f)$, то значения функции g на входах из множества A совпадают со значениями функции f для соответствующих входов. Следовательно, программа P_n, вычисляющая функцию f, является также программой, вычисляющей функцию g. □

Теорема 2. *Пусть $f : \{0,1\}^n \to \{0,1\}$ – всюду определенная функция, $A \subseteq \{0,1\}^n$, $g : A \to \{0,1\}$ – частично определенная функция такая, что $g = \mathrm{Part}_{\mathrm{A}}(f)$. Пусть функция g вычислима ветвящейся программой сложности не менее S. Тогда f вычислима ветвящейся программой сложности не менее S.*

Доказательство. Докажем от противного. Предположим, что любая ветвящаяся программа, вычисляющая функцию g, имеет сложность не менее чем S. Пусть $g = \mathrm{Part}_{\mathrm{A}}(f)$, и при этом функция f вычислима ветвящейся программой сложности строго меньше S. Это противоречит теореме 1. Теорема доказана. □

Отношение эквивалентности. При доказательстве нижних оценок для всюду определенных функций на множестве слов одинаковой длины можно рассматривать отношение эквивалентности. Пусть f_n – всюду определенная функция. Два слова $\sigma, \sigma' \in \{0,1\}^k$ $(k \leq n)$ называются k-эквивалентными относительно функции f_n $(\sigma \equiv^k_{f_n} \sigma')$, если для любых слов $\gamma \in \{0,1\}^{n-k}$ выполняется равенство $f_n(\sigma\gamma) = f_n(\sigma'\gamma)$. Нетрудно убедиться, что отношение $\equiv^k_{f_n}$ является отношением эквивалентности. Оно разбивает множество всех слов длины k на классы эквивалентности, и любая пара слов из множества $\{0,1\}^k$ является либо эквивалентной, либо неэквивалентной друг другу. Обозначим через $w_k(f_n)$ число попарно

k-неэквивалентных относительно функции f_n слов из множества $\{0,1\}^k$ $(k \le n)$. Положим $w(f_n) = \max_k w_k(f_n)$.

Предложение 1. *Пусть f_n – всюду определенная функция, P_n – детерминированная ветвящаяся программа, вычисляющая функцию f_n. Тогда выполняется соотношение* $\mathrm{Width}(P_n) \ge w(f_n)$.

Доказательство данного утверждения является «фольклорным» и следует из принципа Дирихле.

Определим аналогичное отношение для частично определенных функций. Пусть f_n – частично определенная булева функция, $\sigma \in \{0,1\}^k$, $k \le n$. Будем называть слово $\gamma \in \{0,1\}^{n-k}$ подходящим для слова σ, если слово $\sigma\gamma \in \in (f_n)^{-1}(0) \cup (f_n)^{-1}(1)$. Будем называть два слова σ и σ' $(\sigma, \sigma' \in \{0,1\}^k,\ k \le n)$ сравнимыми, если любое слово γ является подходящим для σ тогда и только тогда, когда γ является подходящим для σ'.

Пусть $\sigma, \sigma' \in \{0,1\}^k$ $(k \le n)$ сравнимы. Тогда σ и σ' называются k-эквивалентными $(\sigma \equiv^k_{f_n} \sigma')$, если для любых подходящих $\gamma \in \{0,1\}^{n-k}$ выполняется равенство $f_n(\sigma\gamma) = f_n(\sigma'\gamma)$. Соответственно, будем называть слова σ и σ' k-неэквивалентными, если существует подходящее слово γ такое, что $f_n(\sigma\gamma) \ne \neq f_n(\sigma'\gamma)$. Отметим что, отношение $\equiv^k_{f_n}$, так же как и для всюду определенных функций, является отношением эквивалентности. Однако, в отличие от случая всюду определенных функций, любые два слова одинаковой длины являются либо эквивалентными, либо неэквивалентными друг другу, либо несравнимыми друг с другом.

Предложение 2. *Пусть f_n – частично определенная функция, P_n – детерминированная ветвящаяся программа, вычисляющая функцию f_n. Тогда для любой пары k-неэквивалентных слов σ, σ' вычислительные пути, соответствующие этим словам, не могут вести в одну и ту же вершину.*

Доказательство. Пусть $\sigma, \sigma' \in \{0,1\}^k$, $k < n$ и $\sigma \not\equiv^k_{f_n} \sigma'$. Если вычислительные пути, соответствующие данным словам, приводят в одну и ту же вершину v, то для любых $\gamma \in \{0,1\}^{n-k}$ вычисления на словах $\sigma\gamma$ и $\sigma'\gamma$ будут приводить в одну и ту же финальную вершину. Поскольку σ, σ' неэквивалентны, существует слово γ такое, что $f_n(\sigma\gamma) \ne f(\sigma'\gamma)$. Вычисления на таких входах будут давать неверный результат. Теорема доказана. □

3. Сложность по ширине для квантовых и классических OBDD, вычисляющих всюду определенные функции

Известно, что квантовые OBDD, вычисляющие с ограниченной ошибкой, могут быть эффективнее детерминированных и вероятностных OBDD [26]. Рассмотрим известную симметрическую булеву функцию MOD^p_n от n переменных, принимающую значение 1 только на тех входных наборах, в которых число единиц кратно p, где p – простое число:

$$MOD^p_n(\sigma) = \begin{cases} 1, & \text{если } \#_1(\sigma) = 0 \pmod p, \\ 0 & \text{в противном случае,} \end{cases}$$

$\#_1(\sigma)$ – число единиц в наборе σ.

В работе [26] установлены следующие результаты.

Теорема 3. *Функция MOD^p_n вычислима с ограниченной ошибкой квантовой стабильной OBDD ширины $O(\log p)$.*

Теорема 4. *Любая детерминированная OBDD, вычисляющая MOD_n^p, имеет ширину не менее p. Любая стабильная вероятностная OBDD, вычисляющая MOD_n^p с ограниченной ошибкой, имеет ширину не менее p.*

Следующая нижняя оценка ширины квантовой OBDD показывает, что достигнутое преимущество является максимально возможным для всюду определенных функций (см. [28]).

Теорема 5. *Пусть функция f вычислима один раз читающей квантовой ветвящейся программой Q. Тогда* $\mathrm{Width}(Q) = \Omega(\log \mathrm{Width}(P))$, *где P – детерминированная OBDD минимальной ширины, вычисляющая f.*

Таким образом, экспоненциальное преимущество квантовых OBDD перед классическими является максимально возможным. Ниже мы показываем, что при рассмотрении частично определенных функций преимущество квантовых OBDD перед классическими может быть более чем экспоненциальным.

4. Сравнительная сложность по ширине для квантовых и классических OBDD, вычисляющих частично определенные функции

В работе [11] рассмотрено следующее семейство унарных проблем отделимости (promise problems): $A^k = (A^k_{yes}, A^k_{no})$, где $A^k_{yes} = \{a^{(2i)2^k} \mid i \geq 0\}$, $A^k_{no} = \{a^{(2i+1)2^k} \mid i \geq 0\}$, k – произвольное положительное целое число. Показано, что семейство A^k может распознаваться без ошибки квантовым автоматом с 2 состояниями. При этом классический детерминированный автомат требует не менее 2^{k+1} состояний.

На основе данного языка определим семейство частично определенных булевых функций $\mathtt{PartialMOD} = \{\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}} : k \geq 0, n \geq 2^{k+1}\}$, где

$$\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}(\sigma) = \begin{cases} 1, & \text{если } \#_1(\sigma) = 0 \pmod{2^{k+1}}, \\ 0, & \text{если } \#_1(\sigma) = 2^k \pmod{2^{k+1}}, \\ * & \text{в противном случае,} \end{cases}$$

Здесь $\sigma \in \{0,1\}^n$, и данная функция не определена на наборах, отображающихся в *.

Теорема 6. *Для любого $k \geq 0$ существует стабильная квантовая OBDD ширины 2, вычисляющая частично определенную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки.*

Доказательство. Квантовая стабильная OBDD Q_n, вычисляющая функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки, устроена следующим образом:

$$Q_n = \big(|\psi_0\rangle, T, \mathit{Accept}\big).$$

Здесь $|\psi_0\rangle = (1,0)$, $T = \{\langle j, U(0), U(1)\rangle\}_{j=1}^n$, $\mathit{Accept} = \{1\}$. Программа Q_n считывает входные переменные в естественном порядке $x_1, \ldots, x_n$. Если для входного набора σ на очередном шаге значение считанной переменной σ_j равно 1, то применяется преобразование $U(1) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$, где $\theta = \dfrac{\pi}{2 \cdot 2^k}$. Если значение считанной переменной σ_j равно 0, то применяется тождественное преобразование $U(0) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Если во входном наборе σ число единиц $\#_1(\sigma) = 0 \pmod{2^{k+1}}$, то программа Q_n завершит работу в состоянии $|\psi_n(\sigma)\rangle = (\pm 1, 0)$ и примет набор σ

с вероятностью 1. На входных наборах σ с числом единиц $\#_1(\sigma) = 2^k \pmod{2^{k+1}}$ финальное состояние $|\psi_n(\sigma)\rangle = (0, \pm 1)$. Вероятность принятия таких наборов программой Q_n равна 0. Следовательно, Q_n вычисляет функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки. Теорема доказана. □

Следствие 1. $\mathtt{PartialMOD} \subseteq \mathsf{QOBDD}^2_{\mathsf{exact}}$.

Ниже показывается, что ширина классической OBDD (детерминированной и стабильной вероятностной, вычисляющей с ограниченной ошибкой) для функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ не может быть меньше 2^{k+1}. Приведенная нижняя оценка для детерминированной модели верна для общего нестабильного случая. Доказательство аналогичного результата в [11] для автоматной модели существенным образом использует свойство стабильности модели. Доказательство для модели OBDD требует использования иной техники, поскольку потенциально нестабильность может дать преимущества. Отметим также невозможность напрямую использовать методы, применяемые для всюду определенных функций из-за наличия несравнимых путей в программе, вычисляющей частично определенную функцию. Несложно построить детерминированную стабильную OBDD ширины 2^{k+1}, вычисляющую $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$.

Теорема 7. *Для любых $k \geq 0$ и $n \geq 2^{2k+1} - 2^k$ любая детерминированная OBDD, вычисляющая частично определенную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$, имеет ширину не меньше 2^{k+1}.*

Доказательство. Отношение эквивалентности слов относительно функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ определим следующим образом. Для любых двух слов σ, $\sigma' \in \{0,1\}^k$ $(k \leq n)$ положим

- $\sigma \equiv^k_{\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}} \sigma' \Leftrightarrow \#_1(\sigma) = \#_1(\sigma') \pmod{2^{k+1}}$;
- $\sigma \not\equiv^k_{\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}} \sigma' \Leftrightarrow \#_1(\sigma) \neq \#_1(\sigma') \pmod{2^{k+1}}$ и $\#_1(\sigma) = \#_1(\sigma') \pmod{2^k}$;
- в противном случае слова σ, σ' несравнимы между собой.

Пусть P_n – детерминированная OBDD, вычисляющая частично определенную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$. Обозначим $N = 2^k$. Сразу заметим, что согласно предложению 2 вычисления на неэквивалентных словах не должны приводить в одну и ту же вершину.

Рассмотрим множество $\Gamma = \{\gamma^j : \gamma^j \in \{0,1\}^{2N-1}, \gamma^j = \underbrace{0\ldots0}_{2N-1-j}\,\underbrace{1\ldots1}_{j}, j = 0, \ldots 2N-1\}$. Будем естественным образом отождествлять произвольное слово ν с элементом $a = \#_1(\nu) \pmod{2N}$ из аддитивной группы $\mathbb{Z}_{2N}$. Будем называть два слова одинаковой длины различными, если число единиц по модулю $2N$ в них различно. Обозначим $\rho(\gamma^1, \gamma^2) = |\gamma^1 - \gamma^2|$.

Предположим, что $\mathrm{Width}(P_n) = t < 2N$. На i-м шаге доказательства, $i = 1, 2, \ldots$, будем оценивать количество слов с различным числом единиц по модулю $2N$, которые приводят в одну и ту же вершину (обозначим ее v_i). На i-м шаге будем рассматривать уровень с номером $(2N-1)i$.

На шаге $i = 1$ в соответствии с принципом Дирихле существуют два различных слова σ^1, σ^2 из Γ, приводящие в одну и ту же вершину v_1 уровня $2N-1$. Отметим, что $\rho(\sigma^1, \sigma^2) \neq N$, поскольку в противном случае σ^1, σ^2 неэквивалентны и ведут в одну и ту же вершину.

По индукции покажем, что на каждом шаге доказательства количество слов с различным числом единиц, приводящих в одну и ту же вершину, будет увеличиваться.

Шаг $i = 2$. В соответствии с принципом Дирихле, существуют два различные слова γ^1, $\gamma^2 \in \Gamma$ такие, что слова $\sigma^1\gamma^1$, $\sigma^1\gamma^2$, $\sigma^2\gamma^1$, $\sigma^2\gamma^2$ приводят в одну и ту же

вершину v_2 уровня $(2N-1)2$. Отметим, что $\rho(\gamma^1,\gamma^2)\neq N$, так как в противном случае слова $\sigma^1\gamma^1,\ \sigma^1\gamma^2$ неэквивалентны и ведут в одну и ту же вершину.

Прибавление одного и того же числа не изменяет расстояния между числами, поэтому

$$\rho(\sigma^1+\gamma^1,\sigma^2+\gamma^1)=\rho(\sigma^1,\sigma^2),\quad \rho(\sigma^1+\gamma^2,\sigma^2+\gamma^2)=\rho(\sigma^1,\sigma^2).$$

Пусть $\gamma^2>\gamma^1$. Обозначим $\Delta=\gamma^2-\gamma^1$. Оценим количество различных чисел среди чисел $\sigma^1+\gamma^1,\ \sigma^2+\gamma^1,\ \sigma^1+\gamma^1+\Delta,\ \sigma^2+\gamma^1+\Delta$. Так как числа σ^1,σ^2 различны и $\rho(\sigma^1,\sigma^2)\neq N$, то числа из пары $\sigma^1+\gamma^1,\ \sigma^2+\gamma^1$ могут совпадать с соответствующими числами из пары $\sigma^1+\gamma^1+\Delta,\ \sigma^2+\gamma^1+\Delta$, только если $\Delta=0 \pmod{2N}$, а это не так, поскольку γ^1,γ^2 различны и $\gamma^1,\gamma^2<2N$.

В этом случае по крайне мере три числа из $\sigma^1+\gamma^1,\ \sigma^2+\gamma^1,\ \sigma^1+\gamma^2,\ \sigma^2+\gamma^2$ различны.

Индукционный шаг. Пусть на шаге $i-1$ числа $\sigma^1,\dots,\sigma^i$ все различны и пути, соответствующие этим числам, ведут в одну и ту же вершину v_{i-1} уровня $(2N-1)\times\times(i-1)$.

По принципу Дирихле существуют различные слова $\gamma^1,\gamma^2\in\Gamma$ такие, что соответствующие пути ведут из вершины v_{i-1} в одну и ту же вершину v_i уровня $(2N-1)i$. В этом случае слова $\sigma^1\gamma^1,\dots,\sigma^i\gamma^1,\ \sigma^1\gamma^2,\dots,\sigma^i\gamma^2$ ведут в одну и ту же вершину v_i. Оценим количество различных слов из этого набора. Отметим, что $\rho(\gamma^1,\gamma^2)\neq N$, так как в противном случае слова $\sigma^1\gamma^1,\ \sigma^1\gamma^2$ неэквивалентны и ведут в одну и ту же вершину.

Числа $\sigma^1,\dots,\sigma^i$ различны и $\rho(\sigma^l,\sigma^j)\neq N$ для любых $l,\ j,\ l\neq j$. Пусть $\sigma^1<\dots<\sigma^i$. Последовательность чисел $\sigma^1+\gamma^1,\dots,\sigma^i+\gamma^1$ может в точности совпадать с последовательностью чисел $\sigma^1+\gamma^1+\Delta,\dots,\sigma^i+\gamma^1+\Delta$, только если $\Delta=0 \pmod{2N}$, а это не так, поскольку γ^1,γ^2 различны, и $\gamma^1,\gamma^2<2N$.

Предположим, что последовательность $\sigma^1+\gamma^1+\Delta,\dots,\sigma^i+\gamma^1+\Delta$ является перестановкой чисел последовательности $\sigma^1+\gamma^1,\dots,\ \sigma^i+\gamma^1$. Тогда существуют числа $a_0,\dots,a_r$, принадлежащие $\mathbb{Z}_{2N}$, такие, что все a_j принадлежат последовательности $\sigma^1+\gamma^1,\dots,\ \sigma^i+\gamma^1$, при этом $a_0=a_r=\sigma^1+\gamma^1,\ a_j=a_{j-1}+\Delta$, $j=1,\dots,r$. Тогда $r\Delta=2Nm$. Так как $N=2^k,\ \Delta<2N$ и $\Delta\neq N$, то r четно. Для $z=r/2$ имеем $z\Delta=Nm$. Поскольку все числа последовательности $\sigma^1+\gamma^1,\dots,\sigma^i+\gamma^1$ различны, то $\rho(a_0,a_z)=N$. Таким образом, получили, что a_0,a_z неэквивалентны и ведут в одну и ту же вершину v_i. В этом случае программа неверно вычисляет функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$.

Итак, после шага i доказательства получили, что не менее $i+1$ различных слов ведут в одну и ту же вершину $(2N-1)i$-го уровня.

Рассмотрим шаг N. На этом шаге получим, что $N+1$ различных слова ведут в одну и ту же вершину v_N. Среди этих слов обязательно существуют по крайне мере два неэквивалентных. Это доказывает, что P_n неверно вычисляет функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$. Теорема доказана. □

Следствие 2. *Для любого $k\geq 0$ существует частично определенная функция $f\in\mathtt{PartialMOD}$ такая, что $f\in\mathsf{OBDD}^{2^{k+1}}$ и $f\notin\mathsf{OBDD}^{2^k}$.*

Теорема 8. *Для любого k существует бесконечно много n таких, что любая стабильная вероятностная OBDD, вычисляющая функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$, имеет ширину не меньше 2^{k+1}.*

Доказательство. Предположим противное, то есть допустим, что существует стабильная вероятностная OBDD P_n ширины $d<2^{k+1}$, вычисляющая $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ с вероятностью $1/2+\varepsilon$ для фиксированного $\varepsilon\in(0,1/2]$. Обозначим

через $v_j = (v_j[1], \ldots, v_j[d])$ вектор распределения вероятностей вершин программы P_n на j-м уровне, где $v_j[i]$ – вероятность нахождения программы P_n в i-й вершине j-го уровня. Вычислительный процесс программы P_n на входе $\sigma = \sigma^1, \ldots, \sigma_n$ может быть описан следующим образом:

• программа P_n начинает вычисление из начального распределения вероятностей – вектора v_0;

• на j-м шаге, $1 \le j \le n$, P_n считывает значение входной переменной σ_{i_j} и преобразует вектор v_{j-1} в вектор $v_j = Av_{j-1}$, где A – стохастическая матрица размера $(d \times d)$, $A = A(0)$, если $\sigma_{i_j} = 0$, и $A = A(1)$, если $\sigma_{i_j} = 1$;

• После последнего n-го шага программа P_n принимает входной набор σ с вероятностью $P_{accept}(\sigma) = \sum\limits_{i \in Accept} v_n[i]$, при этом $P_{accept}(\sigma) \ge 1/2 + \varepsilon$, если $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}(\sigma) = 1$, и $P_{accept}(\sigma) \le 1/2 - \varepsilon$, если $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}(\sigma) = 0$.

Без ограничения общности полагаем, что P_n считывает переменные в естественном порядке $x_1, \ldots, x_n$. Будем рассматривать входные наборы $\widetilde{\sigma}_n, \ldots, \widetilde{\sigma}_1$ такие, что $\widetilde{\sigma}_i = \widetilde{\sigma}_i^0 \widetilde{\sigma}_i^1$, где $\widetilde{\sigma}_i^0 = \underbrace{0 \ldots 0}_{n-i}$, $\widetilde{\sigma}_i^1 = \underbrace{1 \ldots 1}_{i}$.

Для $i \in \{1, \ldots, n\}$ обозначим через α^i распределение вероятностей после считывания $\widetilde{\sigma}_i^0$, то есть $\alpha^i = A^{n-i}(0)v_0$. В наборе $\widetilde{\sigma}_i^1$ содержатся только единицы, поэтому вычислительный процесс после считывания $\widetilde{\sigma}_i^0$ может быть описан цепью Маркова. В этом случае α^i – начальное распределение вероятностей для процесса Маркова, $A(1)$ – переходная матрица.

Состояния цепи Маркова разделяются на эргодические и невозвратные (см., например, [29]). Эргодическое множество состояний – это множество, которое процесс никогда не сможет покинуть, если в него однажды попадет. Невозвратное множество – это множество, в которое процесс не может вернуться, если его покидает. Эргодическое состояние – это элемент эргодического множества. Невозвратное состояние – это элемент невозвратного множества.

Любая цепь Маркова обязательно содержит хотя бы одно эргодическое множество. Наличие невозвратных множеств необязательно. Если цепь Маркова содержит более чем одно эргодическое множество, то между этими множествами нет абсолютно никакого взаимодействия. Следовательно, мы имеем две или более изолированные цепи Маркова, объединенные вместе, которые мы можем изучать по отдельности. Если цепь Маркова состоит из единственного эргодического множества, то она называется эргодической цепью. Каждая эргодическая цепь либо регулярна, либо циклична.

Если эргодическая цепь регулярна, то достаточно высокая степень матрицы переходных вероятностей содержит только положительные элементы. Это означает, что из какого бы состояния процесс ни начался, по прошествии достаточно большого числа шагов он может оказаться в любом состоянии. Кроме того, существует предельный вектор вероятностей, не зависящий от выбора вектора начального распределения вероятностей (см., например, теорему 4.1.6 [29]).

Если цепь Маркова циклична, то она имеет период t, и все ее состояния разбиваются на t циклических множеств, $t > 1$. При выбранном начальном состоянии процесс движется по циклическим множествам в определенном порядке, возвращаясь в множество, содержащее начальное состояние, через каждые t шагов. По прошествии достаточного длительного времени процесс может находиться в любом состоянии циклического множества, соответствующего данному моменту. Следовательно, для каждого из циклических множеств степень t матрицы переходных вероятностей описывает регулярную цепь Маркова. К тому же, если эргодическая цепь циклична с периодом t, то она имеет не менее t состояний.

Пусть $C_1, \ldots, C_l$ – циклические множества состояний марковской цепи с периодами $t_1, \ldots, t_l$ соответственно. Обозначим через D наименьшее общее кратное чисел $t_1, \ldots, t_l$.

Лемма 1. *Число D кратно 2^{k+1}.*

Доказательство. Предположим противное, то есть допустим, что D не кратно 2^{k+1}. После каждых D шагов вычислительный процесс может находиться в любом состоянии множества, содержащего принимающее состояние, и степень D матрицы $A(1)$ описывает регулярную цепь Маркова для этого множества. Из теории марковских цепей известно, что существует α_{acc} такое, что $\lim\limits_{r\to\infty} \alpha_{acc}^{rD} = \alpha_{acc}$, где α_{acc}^{i} – вероятность нахождения в принимающем состоянии после i-го шага. Следовательно, для любого $\varepsilon > 0$ верно, что

$$|\alpha_{acc}^{r\cdot D} - \alpha_{acc}^{r'\cdot D}| < 2\varepsilon \tag{1}$$

для достаточно больших $r,\ r'$.

Так как число D не кратно 2^{k+1}, то оно может быть представлено в виде $D = m \cdot 2^l$ ($l \le k$, m нечетно). Для любого нечетного s число $s \cdot D$ не кратно 2^{k+1}. Поскольку по предположению P_n вычисляет функцию $\mathtt{PartialMOD}_{\mathtt{n}}^{\mathtt{k}}$ с вероятностью $1/2 + \varepsilon$, то $\alpha_{acc}^{s\cdot m\cdot 2^l 2^{k-l+1}} \ge 1/2 + \varepsilon$, $\alpha_{acc}^{s\cdot m\cdot 2^l 2^{k-l}} \le 1/2 - \varepsilon$. Это противоречит неравенству 1 для достаточно большого s. Лемма доказана. □

Лемма 2. *Существует цикл с периодом t, кратным 2^{k+1}.*

Доказательство. Поскольку D кратно 2^{k+1}, среди чисел $t_1, \ldots, t_l$, являющихся степенями двойки, должно быть по крайней мере одно, кратное 2^{k+1}.

Лемма доказана. □

Поскольку существует цикл с периодом $t \ge 2^{k+1}$, то ширина программы $P_n \ge 2^{k+1}$.

Теорема доказана. □

Следствие 3. *Для любого $k \ge 0$ существует частично определенная функция $f \in \mathtt{PartialMOD}$ такая, что $f \in \mathsf{POBDD}_{1/2+\varepsilon}^{2^{k+1}}$ и $f \notin \mathsf{POBDD}_{1/2+\varepsilon}^{2^{k}}$.*

5. Коммуникационная сложность частично определенной функции $\mathtt{PartialMOD}_{\mathtt{n}}^{\mathtt{k}}$

OBDD является вычислительной моделью с хорошими математическими свойствами, позволяющими получать высокие нижние оценки для конкретных функций. Для доказательства нижних оценок для всюду определенных функций разработаны различные методы.

Метод Нечипорука [30] основан на том, что функция, имеющая большое количество различных подфункций, не может быть представима ветвящейся программой малого размера. Для ветвящихся программ этот метод переложил П. Пудлак в 1984 г. [19].

Пусть $f : \{0,1\}^n \to \{0,1\}$ – функция, зависящая от переменных множества $X = \{x_1, \ldots, x_n\}$, $S \subset X$. Фиксируя переменные из множества $X \setminus S$ константами, получаем подфункции f, которые будем называть подфункциями функции f на множестве S.

Теорема 9 [19]. *Пусть f – булева функция, множество X переменных которой разбито на m непересекающихся групп $S_1, \ldots, S_m$, $s_i(f)$ – количество*

различных подфункций функции f на множестве S_i. Тогда сложность ветвящейся программы, вычисляющей функцию f, не может быть меньше чем

$$\Omega\left(\sum_{i=1}^{m}\frac{\log s_i(f)}{\log\log s_i(f)}\right).$$

При доказательстве нижних оценок для ширины OBDD, вычисляющих всюду определенные функции, также используется так называемый коммуникационный подход, основанный на применении аппарата коммуникационных вычислений, с помощью которого разработаны эффективные методы получения нижних оценок сложности.

Коммуникационная модель вычислений была предложена Яо в 1979 г. [31]. В этой модели имеются два вычислителя A и B, которые совместно хотят вычислить значение функции $f : \{0,1\}^n \to \{0,1\}$ на входном наборе $\sigma \in \{0,1\}^n$. Пусть $X = \{x_1, \ldots, x_n\}$ – множество переменных функции f, $X = \{X_A, X_B\}$ – разбиение множества X на две части X_A и $X_B = X \setminus X_A$. Переменные из множества X_A поступают на вход вычислителя A, переменные из множества X_B – на вход вычислителя B. Для того чтобы найти значение функции на входе σ, вычислителям необходимо обмениваться информацией друг с другом. Алгоритм вычисления функции на данной модели называется протоколом. Мерой сложности протокола является количество битов, которые вычислителям необходимо переслать друг другу до того момента, когда вычислитель B сможет выдать результат – значение $f(\sigma)$. Односторонний коммуникационный протокол – это протокол, организованный следующим образом. Вычислитель A начинает вычисление, передает необходимую информацию вычислителю B, после чего вычислитель B выдает ответ. Сложность одностороннего протокола – это максимальное по всем входным наборам число битов, переданных от A к B. Односторонней коммуникационной сложностью $CC_1(f)$ функции f называется сложность наилучшего одностороннего коммуникационного протокола, вычисляющего функцию f.

Теорема 10 [32]. *Пусть f – всюду определенная булева функция, D – детерминированная OBDD, вычисляющая функцию f. Тогда* $\mathrm{Width}(P) \geq 2^{CC_1(f)}$.

Доказательство теоремы основано на том, что вычисление ветвящейся программой P значения функции f на входном наборе σ можно рассматривать как односторонний коммуникационный протокол. Разобъем OBDD горизонтально на две части P_1 и P_2. Считаем, что вычисления в верхней части программы производится вычислителем A, в нижней части – вычислителем B. Для входного набора σ вычислитель A начинает вычисление из начальной вершины. После того, как A произвел вычисление на своей части P_1 программы, вычислитель B продолжает вычисление из той вершины, в которой завершил работу вычислитель A. Таким образом, номер вершины, в которой завершил работу вычислитель A, является сообщением, которое A должен передать B, чтобы тот смог продолжить работу.

Перечисленные выше методы получения нижних оценок хорошо работают для всюду определенных функций. Однако при рассмотрении частично определенных функций они в большинстве своем неприменимы. Ниже мы показываем, что односторонняя коммуникационная сложность функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ равна единице для любого k. При этом, как доказано в предыдущем разделе, ширина OBDD для этой функции растет с увеличением параметра k. Таким образом, доказательство нижних оценок для частично определенных функций требует привлечения другой техники.

Теорема 11. *Для любого разбиения переменных* $X = \{X_A, X_B\}$ *выполняется равенство*

$$CC_1(\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}) = 1.$$

Доказательство. Опишем односторонний коммуникационный протокол для вычисления функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$. Пусть $X = \{X_A, X_B\}$ – произвольное разбиение множества переменных, на вход вычислителя A поступают значения переменных из множества X_A, на вход вычислителя B – из множества X_B. Вычислитель A вычисляет значение a (число единиц в своем наборе) и передает вычислителю B значение 1, если $0 < a \leq 2^k \pmod{2^{k+1}}$, и значение 0, если $a > 2^k \pmod{2^{k+1}}$ или $a = 0 \pmod{2^{k+1}}$.

Вычислитель B считает значение b (количество единиц в своем входном наборе по модулю 2^{k+1}) и выдает ответ $\mathrm{res} = 1$ тогда и только тогда, когда $a \leq 2^k \pmod{2^{k+1}}$ и $b \geq 2^k \pmod{2^{k+1}}$ либо $a > 2^k \pmod{2^{k+1}}$ и $b < 2^k \pmod{2^{k+1}}$. Сложность построенного протокола равна 1. Теорема доказана. □

6. Сравнительная сложность по ширине для вероятностных и детерминированных OBDD, вычисляющих частично определенные функции

В этом разделе исследуется сравнительная сложность детерминированных OBDD и вероятностных OBDD, вычисляющих с ограниченной ошибкой. В [27] рассмотрено семейство унарных проблем отделимости (promise problems) $UP(p)$. Авторы показывают, что любая проблема из данного семейства может распознаваться с ограниченной ошибкой вероятностным автоматом с двумя состояниями. При этом для классического детерминированного, а также для недетерминированного автомата, распознающего данную проблему, количество состояний увеличивается в ростом параметра p. На основе указанного семейства унарных проблем отделимости, определим следующую булеву функцию. Пусть $\varepsilon \in (0, \sqrt{5}/2 - 1)$, $p \in (1/2 + \varepsilon, 1)$. Тогда положим

$$\mathtt{f}^{\mathtt{p},\varepsilon}_{\mathtt{n}}(\sigma) = \begin{cases} 1, & \text{если } \ p^{\#_1(\sigma)} \geq 1/2 + \varepsilon, \\ 0, & \text{если } \ p^{\#_1(\sigma)} \leq 1/2 - \varepsilon, \\ * & \text{в противном случае,} \end{cases}$$

где $\sigma \in \{0,1\}^n$, и функция не определена на наборах, отображающихся в *.

Теорема 12. *Для любых* $p \in (1/2, 1)$ *функция* $\mathtt{f}^{\mathtt{p},\varepsilon}_{\mathtt{n}}$ *вычислима с ограниченной ошибкой вероятностной OBDD ширины* 2.

Доказательство. Определим вероятностную OBDD P_n следующим образом:

$$P_n = (v_0, T, \mathit{Accept}),$$

где $v_0 = (1,0)$, $\mathit{Accept} = \{s_0\}$, $T = \{\langle i, A(0), A(1)\rangle\}_{i=1}^n$, где $A(0) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $A(1) = \begin{pmatrix} p & 0 \\ 1-p & 1 \end{pmatrix}$. Программа P_n начинает работу в состоянии s_0. При считывании нуля программа выполняет тождественное преобразование, при считывании единицы программа остается в состоянии s_0 с вероятностью p и переходит в состояние s_1 с вероятностью $1-p$. По построению очевидно, что P_n вычисляет функцию $\mathtt{f}^{\mathtt{p},\varepsilon}_{\mathtt{n}}$ с ограниченной ошибкой. Теорема доказана. □

Теорема 13. *Для любых* $\varepsilon \in (0, \sqrt{5}/2 - 1)$ *существует бесконечно много* $p \in (1/2 + \varepsilon, 1)$ *таких, что детерминированная OBDD, вычисляющая функцию* $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}$, *имеет ширину, большую чем*

$$\log\left(\frac{1}{2}+\varepsilon\right)\Big/\log\left(\frac{1-2\varepsilon}{1+2\varepsilon}\right)+2.$$

Доказательство. Предположим противное. Обозначим

$$w=\log\left(\frac{1}{2}+\varepsilon\right)\Big/\log\left(\frac{1-2\varepsilon}{1+2\varepsilon}\right)+2$$

и допустим, что существует детерминированная OBDD P_n, вычисляющая функцию $\mathtt{f}_{\mathtt{n}}^{\mathtt{p}}$ и имеющая ширину, меньшую или равную w. Обозначим $A_p = \lfloor \log_p(1/2+\varepsilon) \rfloor$, $R_p = \lceil \log_p(1/2-\varepsilon) \rceil$. Для любых $j \le A_p$ выполняется неравенство $p^j \ge 1/2+\varepsilon$, и для любых $j \ge R_p$ выполняется неравенство $p^j \le 1/2-\varepsilon$. Обозначим $d = R_p - A_p$.

Пусть $\sigma, \sigma' \in \{0,1\}^k$ $(k < n)$ – произвольные слова. Докажем следующую лемму.

Лемма 3. *Для того чтобы два различных слова* $\sigma, \sigma' \in \{0,1\}^k$ *были* k*-неэквивалентными относительно функции* $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}$, *необходимо и достаточно, чтобы выполнялись условия*

- $|\#_1(\sigma) - \#_1(\sigma')| \ge d$;
- $\min(\#_1(\sigma), \#_1(\sigma')) \le A_p$.

Доказательство. Без ограничения общности полагаем $\#_1(\sigma') > \#_1(\sigma)$. Установим достаточность. Предположим, что выполняются условия леммы. Покажем, что тогда слова σ, σ' являются k-неэквивалентными. Прежде всего заметим, что если выполняются $\#_1(\sigma') \ge R_p$ и $\#_1(\sigma) \le A_p$, то строки σ, σ' неэквивалентны, так как в этом случае для слова $\gamma = \underbrace{0\ldots0}_{n-k}$ выполняются равенства $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}(\sigma\gamma) = 1$ и $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}(\sigma'\gamma) = 0$.

Пусть $\#_1(\sigma') < R_p$. Обозначим $\Delta = R_p - \#_1(\sigma')$ и рассмотрим слово $\gamma = \underbrace{0\ldots0}_{n-\#_1(\sigma')-\Delta}\underbrace{1\ldots1}_{\Delta}$. Для слова $\sigma'\gamma$ выполняется соотношение $\#_1(\sigma'\gamma) = \#_1(\sigma') + \#_1(\gamma) = R_p$, а значит, $\mathtt{f}_{\mathtt{n}}^{\mathtt{p}}(\sigma'\gamma) = 0$.

Рассмотрим слово $\sigma\gamma$. Поскольку $\#_1(\sigma') - \#_1(\sigma) \ge d$ и $\#_1(\gamma) = \Delta = R_p - \#_1(\sigma')$, то $\#_1(\sigma\gamma) = \#_1(\sigma) + \#_1(\gamma) \le \#_1(\sigma') - d + \#_1(\gamma) = \#_1(\sigma') - d + R_p - \#_1(\sigma') = R_p - d \le A_p$. Следовательно, $\mathtt{f}_{\mathtt{n}}^{\mathtt{p}}(\sigma\gamma) = 1$.

Докажем необходимость. Предположим, слова σ, σ' являются k-неэквивалентными, но при этом $\#_1(\sigma') - \#_1(\sigma) < R_p - A_p$ либо $\#_1(\sigma) > A_p$. Согласно определению неэквивалентности слов существует слово $\gamma \in \{0,1\}^{n-k}$, которое различает слова σ, σ'. По определению, функция $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}$ принимает значение 1 на наборах, в которых число единиц не больше A_p, и принимает значение 0 на наборах, в которых число единиц не меньше R_p. Поэтому $\mathtt{f}_{\mathtt{n}}^{\mathtt{p}}(\sigma\gamma) = 1$, $\mathtt{f}_{\mathtt{n}}^{\mathtt{p},\varepsilon}(\sigma'\gamma) = 0$, что эквивалентно $\#_1(\sigma) + \#_1(\gamma) \le A_p$ и $\#_1(\sigma') + \#_1(\gamma) \ge R_p$. Отсюда получаем $(\#_1(\sigma') + \#_1(\gamma)) - (\#_1(\sigma) + \#_1(\gamma)) \ge R_p - A_p$. Следовательно, $\#_1(\sigma') - \#_1(\sigma) \ge R_p - A_p$, а это противоречит нашему предположению. Второе условие леммы очевидно должно выполняться для неэквивалентных слов. Лемма доказана. □

Пусть $\Gamma = \{\gamma^j : \gamma^j \in \{0,1\}^{w-1}, \gamma^j = \underbrace{0\ldots0}_{w-1-j}\underbrace{1\ldots1}_{j}, j = 0,\ldots,w-1\}$. Будем проводить доказательство по шагам. На шаге i, $i = 1, 2, \ldots$, будем рассматривать

уровень с номером $(w-1)i$. Покажем, что если ширина каждого уровня меньше или равна w, то программа P_n неверно вычисляет функцию $\mathtt{f}_\mathtt{n}^{\mathtt{p},\varepsilon}$. Будем проводить доказательство индукцией по i.

Базис индукции. На шаге $i=1$ после считывания $(w-1)$ символа входного слова согласно принципу Дирихле найдутся два различных слова σ_1, $\sigma_2 \in \Gamma$ такие, что вычислительные пути, соответствующие этим словам, ведут в одну и ту же вершину $(w-1)$-го уровня. Обозначим эту вершину $(w-1)$-го уровня через v_1^{w-1}. Положим $d_1 = |\#_1(\sigma_1) - \#_1(\sigma_2)|$, $l_1 = \min(\#_1(\sigma_1), \#_1(\sigma_2))$. Очевидно, что $d_1 \geq 1$, $l_1 \leq w-2$.

На шаге $i=2$ на уровне $2(w-1)$ согласно принципу Дирихле существует вершина, в которую ведут вычислительные пути, исходящие из вершины v_1^{w-1}, которые соответствуют различным словам γ_1, $\gamma_2 \in \Gamma$. Обозначим эту вершину $2(w-1)$-го уровня через $v_2^{2(w-1)}$. Таким образом, в вершину $v_2^{2(w-1)}$ ведут вычислительные пути, соответствующие различным словам $\sigma_1\gamma_1$, $\sigma_1\gamma_2$, $\sigma_2\gamma_1$, $\sigma_2\gamma_2$. Обозначим $d_2 = \max_{i,j,i',j' \in \{1,2\}}(|\#_1(\sigma_i\gamma_j) - \#_1(\sigma_{i'}\gamma_{j'})|)$, $l_2 = \min_{i,j\in\{1,2\}}(\#_1(\sigma_i\gamma_j))$. Поскольку γ_1 и γ_2 различны, очевидно, что $d_2 \geq 2$, $l \leq 2(w-2)$.

Индукционный шаг. Пусть на уровне с номером $(i-1)(w-1)$ существует вершина $v_{i-1}^{(i-1)(w-1)}$, в которую ведут пути, соответствующие словам $\sigma_1, \ldots, \sigma_{j_{i-1}}$. При этом d_{i-1} – максимальная разница в числе единиц между этими словами, l_{i-1} – минимальное число единиц в этих словах. На шаге i рассматриваем уровень с номером $i(w-1)$. По принципу Дирихле существуют два различные слова γ, $\gamma' \in \Gamma$ такие, что вычислительные пути, выходящие из вершины $v_{i-1}^{(i-1)(w-1)}$, которые соответствуют этим словам, ведут в одну и ту же вершину $(i(w-1))$-го уровня. Обозначим эту вершину через $v_i^{i(w-1)}$. Рассмотрим слова $\sigma_1\gamma, \ldots, \sigma_{j_{i-1}}\gamma$, $\sigma_1\gamma', \ldots, \sigma_{j_{i-1}}\gamma'$.

Поскольку $0 \leq \#_1(\gamma), \#_1(\gamma') \leq w-1$ и $\#_1(\gamma) \neq \#_1(\gamma')$, то $d_i \geq d_{i-1}+1$, $l_i \leq i(w-2)$.

Таким образом, после k шагов имеем, что $d_k \geq k$, $l_k \leq k(w-2)$. Применяя лемму 3, получаем, что если $w \leq A_p/d+2$, то в одну и ту же вершину приведут вычислительные пути, соответствующие неэквивалентным словам, а следовательно, программа P_n неправильно вычисляет функцию $\mathtt{f}_\mathtt{n}^{\mathtt{p},\varepsilon}$. Получили противоречие.

Имеем, что

$$\log(1/2+\varepsilon)/\log\left(\frac{1-2\varepsilon}{1+2\varepsilon}\right) > 1$$

при $\varepsilon \in (0, \sqrt{5}/2-1)$, и значение этого выражения растет с уменьшением значения ε. Теорема доказана. □

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект № 14-07-00878).

Summary

A.F. Gainutdinova. Complexity of Branching Programs for Partial Functions.

In this paper we investigate a model of computation of Boolean functions – Ordered Binary Decision Diagrams (OBDD). We consider partial functions and complexity of their computation in different variants of OBDD (deterministic, probabilistic, and quantum). The interested complexity measure is a width of OBDD. It is known that for total functions, bounded-error probabilistic OBDDs can be more effective than the deterministic ones, and this gap cannot be more than exponential. The similar result holds when we compare quantum and deterministic models. In this paper it is shown that for partial functions the gap between

quantum and classical, and between probabilistic and deterministic OBDDs can be more than exponential. A partial function is presented which is computed without an error by a quantum OBDD of width 2. Deterministic and bounded-error probabilistic OBDDs for this function must have widths exponentially depending on the parameter of the function. An infinite family of partial functions is also presented such that each function from this family is computed by a bounded-error probabilistic OBDD of width 2. There exists an infinite subset of functions from this family such that a width of a deterministic OBDD for each function from this subset increases indefinitely.

Keywords: Ordered Binary Decision Diagrams, partial functions, quantum computation, probabilistic OBDDs, deterministic OBDDs, complexity.

Литература

1. *Шоломов Л.А.* Основы теории дискретных логических и вычислительных устройств. – М.: Наука, 1980. – 400 с.
2. *Рабин М.* Вероятностные автоматы // Кибернетический сб. – М.: Мир, 1964. – Вып. 9. – С. 123–141.
3. *Фрейвалд Р.В.* Об увеличении числа состояний при детерминизации конечных вероятностных автоматов // Автоматика и вычисл. техника. – 1982. – № 3. – С. 39–42.
4. *Ablayev F., Karpinski M.* On the power of randomized branching programs // Proc. ICALP'96, LNCS 1099. – Springer, 1996. – P. 348–356.
5. *Манин Ю.И.* Вычислимое и невычислимое. – М.: Сов. радио, 1980. – 128 с.
6. *Feynman R.* Simulating physics with computers // Int. J. Theor. Phys. – 1982. – V. 21, No 6, 7. – P. 467–488.
7. *Nielsen M.A., Chuang I.L.* Quantum Computation and Quantum Information. – Cambridge: Cambridge Univ. Press, 2000. – 700 p.
8. *Shor P.* Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer // SIAM J. Sci. Statist. Comput. – 1997. – V. 26, No 5. – P. 1484–1509.
9. *Grover L.* A fast quantum mechanical algorithm for database search // Proc. STOC'96. – N. Y: ACM New York, 1996. – P. 212–219.
10. *Goldreich O.* On promise problem: A survey // Goldreich O., Rosenberg A.L., Selman A.L. (eds.) Shimon Even Festschrift. LNCS 3895. – 2006. – – P. 254–290.
11. *Ambainis A., Yakaryılmaz A.* Superiority of exact quantum automata for promise problems // Inf. Process. Lett. – 2012. – V. 112, No 7. – P. 289–291.
12. *Wegener I.* Branching Programs and Binary Decision Diagrams. – SIAM, 2000. – 411 p.
13. *Lee C.Y.* Representation of switching circuits by binary-decision programs // Bell Syst. Tech. J. – 1959. – V. 38, No 4. – P. 985–999.
14. *Akers S.B.* Binary decision diagrams // IEEE Trans. Comput. – 1978. – V. C-27, No 6. – P. 509–516.
15. *Кузьмин В.А.* Оценка сложности реализации функций алгебры логики простейшими видами бинарных программ // Методы дискретного анализа в теории кодов и схем: Сб. науч. тр. – Новосибирск: Ин-т математики СО АН СССР, 1976. – Вып. 29. – С. 11–39.
16. *Mazek W.* A fast algorithm for the string editing problem and decision graph complexity: Master's Thesis. – Massachusetts Institute of Techonology, 1976.
17. *Cobham A.* The recognition problem for the set of perfect squares // Proc. 7th Symposium on Switching and Automata Theory (SWAT). – 1996. – P. 78–87.

18. *Pudlàk P., Zak S.* Space complexity of computations: Tech. Report. – Prague: Math. Inst., CSAV, 1983. – 30 p.

19. *Pudlàk P.A.* A Lower Bound on Complexity of Branching Program // Proc. of the Mathematical Foundations of Computer Science. – Springer-Verlag, 1984. – P. 480–489.

20. *Ablayev F.* Randomization and nondeterminism are incomparable for polynomial ordered binary decision diagrams // Proc. of the 24th Int. Coll. on Automata, Languages, and Programming (ICALP). LNCS 1256. – 1997. – P. 1965–202.

21. *Ablayev F., Karpinski M.* A lower bound for integer multiplication on randomized read-once branching programs // Electronic Colloquium on Computational Complexity, Report No 11. – 1998. – URL: http://eccc.hpi-web.de/report/1998/011/, свободный.

22. *Ablayev F., Gainutdinova A., Karpinski M.* On Computational Power of Quantum Branching Programs // Proc. 13th Int. Symposium "Fundamentals of computation theory" FCT 2001, Riga, Latvia, LNCS 2138. – 2001. – P. 59–70.

23. *Баррингтон Д.* Ветвящиеся программы ограниченной ширины, имеющие полиномиальную сложность, распознают в точности языки из NC^1 // Кибернетический сб. – М.: Мир. 1991. – Вып. 28. – С. 94–113.

24. *Nakanishi M., Hamaguchi K., Kashiwabara T.* Ordered quantum branching programs are more powerful than ordered probabilistic branching programs under a bounded-width restriction // Proc. 6th Annual Int. Conf. on Computing and Combinatorics, COCOON'2000. LNCS 1858. – Springer-Verlag, 2000. – P. 467–476.

25. *Sauerhoff M., Sieling D.* Quantum branching programs and space-bounded nonuniform quantum complexity // Theor. Comp. Sci. – 2005. – V. 334, No 1–3. – P. 177–225.

26. *Гайнутдинова А.Ф.* О сравнительной сложности квантовых и классических бинарных программ // Дискретная матем. – 2002. – Вып. 14, № 3. – С. 109–121.

27. *Geffert V., Yakarylmaz A.* Classical automata on promise problems // Descriptional Complexity of Formal Systems. 16th Int. Workshop, DCFS 2014 Proc. LNCS 8614. – 2014. – P. 126–137.

28. *Ablayev F., Gainutdinova A., Karpinski M., Moore C., Pollette C.* On the computational power of probabilistic and quantum branching program // Information and Computation. – 2005. – V. 203, No 2. – P. 145–162.

29. *Kemeny J.G., Snell J.L.* Finite Markov Chains. – D. Van Nostrand Comp. Inc., 1960. – 210 p.

30. *Нечипорук Э.И.* Об одной булевской функции // Докл. АН СССР. – 1966. – Т. 169, № 4. – С. 765–766.

31. *Yao A.C.* Some Complexity Questions Related to Distributed Computing // Proc. 11th STOC. – V. 14. – P. 209–213.

32. *Wegener I.* Communication Complexity and BDD Lower Bound Techniques // Numbers, Information and Complexity. – Springer-Verlag, 2000. – P. 615–628.

Поступила в редакцию
25.07.14

Гайнутдинова Аида Фаритовна – кандидат физико-математических наук, доцент кафедры теоретической кибернетики, Казанский (Приволжский) федеральный университет, г. Казань, Россия.

E-mail: *aida.ksu@gmail.com*