S©MR

СИБИРСКИЕ ЭЛЕКТРОННЫЕ
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ
Siberian Electronic Mathematical Reports
http://semr.math.nsc.ru
ISSN 1813-3304

Том 20, № 2, стр. 144–145 (2023) УДК 519.71
https://doi.org/10.33048/semi.2023.20.??? MSC 68Q12,03D15

НЕДЕТЕРМИНИРОВАННЫЕ КВАНТОВЫЕ OBDD БОЛЬШОЙ ШИРИНЫ

А.Ф. Гайнутдинова

Abstract: In the paper we investigate Ordered Binary Decision Diagrams (OBDDs) — a model for computing boolean functions. We provide a comparative complexity analysis of quantum and classical nondeterministic OBDDs of large width. We explore the complexity of calculating the well-known boolean function 'Equality' in non-deterministics quantum OBDDs (NQOBDDs) for different orders of reading variables. We show that, using an order of reading variables, for which the width of classical nondeterministic OBDDs is constant, the width of NQOBDD is linear. We define a boolean function for which the width of NQOBDDs is exponential, regardless of the variables order. For this function, we present a quantum algorithm that computes it with zero error. We introduce a new hierarchy based on quantum nondeterministic OBDDs of large width.

Keywords: branching program, ordered binary decision diagram, nondeterminism, quantum algorithm, complexity, complexity class, computational model, hierarchy of complexity classes, lower bound, upper bound.

Gainutdinova, A.F., Nondeterministic quantum OBDDs of large width.

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета ("ПРИОРИТЕТ-2030").

1 Введение

Исследование вычислительных возможностей квантовых моделей в сравнении с их классическими аналогами, и построение эффективных квантовых алгоритмов для моделей с различными ограничениями является актуальным направлением исследований. Ветвящиеся программы (BP – Branching Programs) – известная модель для вычисления булевых функций, основанная на применении операций “if”, “then”, “else” и “goto”. Эта модель имеет разнообразные приложения, в частности, в области верификации моделей и программ, в базах данных и т.д. [1, с. 5–11]. Установлено, что логарифм сложности ветвящейся программы соответствует объёму памяти машины Тьюринга, а максимальная длина вычислительного пути – времени вычисления [2, 3].

Модель квантовых BP, как последовательность унитарных эволюций квантовой системы с заключительным измерением для извлечения результата вычислений, была впервые определена в работе [4]. В работах [5, 6] были определены несколько иные модели квантовых ветвящихся программ. Была доказана эквивалентность всех этих моделей. С тех пор модель квантовых BP активно исследовалась различными авторами. В частности, было показано, что квантовые BP могут быть эффективнее классических аналогов.

Упорядоченные ветвящиеся диаграммы решений (OBDD – Ordered Binary Decision Diagrams) представляют собой модель BP с ограничениями: на каждом вычислительном пути переменные считываются в одном и том же порядке не более одного раза. Поскольку длина OBDD не превосходит длины n входа, естественной мерой сложности для этой модели является её ширина. OBDD фиксированной ширины, использующие одинаковые преобразования на каждом шаге (стабильные OBDD), можно рассматривать как конечные автоматы, обрабатывающие слова фиксированной длины. Различные варианты OBDD: детерминированные, недетерминированные, вероятностные, квантовые, исследовались разными авторами [7, 5, 8, 6, 9, 10]. Было показано, что вероятностные OBDD могут быть экспоненциально более эффективными, чем детерминированные и недетерминированные [7], а квантовые OBDD – экспоненциально более эффективными, чем детерминированные и стабильные вероятностные и что данная оценка точна [8]. Было продемонстрировано превосходство квантового недетерминизма над классическим для моделей OBDD: была представлена функция, вычислимая недетерминированными квантовыми OBDD (NQOBDD) константной ширины, в то время как классические недетерминированные OBDD (NOBDD) для этой функции имеют неконстантную ширину [9]. Более детальное исследование квантовых недетерминированных OBDD и их сравнение с классическими моделями было представлено в работе [10], где рассматривались NQOBDD линейной и сублинейной ширины, для которых, в частности, было показано, что квантовые и классические недетерминированные модели несравнимы между собой.

Во всех упомянутых выше работах рассматриваемые NQOBDD имели линейную и сублинейную ширину. В данной работе мы продолжаем исследование недетерминированных OBDD, сосредотачиваясь преимущественно на программах сверхлинейной ширины. Одной из особенностей модели OBDD является то, что она может использовать произвольный порядок считывания переменных. Упомянутые выше результаты о сравнительной сложности квантовых и классических OBDD основывались на симметрических булевых функциях, для которых порядок считывания переменных входа не важен, поскольку значение функции на конкретном наборе зависит только от числа нулей и единиц в этом наборе, а не от их расположения. При этом ширина OBDD для таких функций не более чем линейна. Для получения высоких нижних оценок необходимо исследование несимметрических булевых функций. Для таких функций сложность OBDD может существенным образом зависеть от того, в каком порядке программа считывает переменные. Известны примеры функций, для которых разница в сложности в зависимости от используемого порядка считывания экспоненциальна. При этом задача нахождения наилучшего порядка считывания для заданной функции является NP-полной [11, с. 135]. Для устранения зависимости сложности от порядка считывания переменных входа используют различные приемы при конструировании функций, для которых не удаётся подобрать оптимальный порядок считывания.

В данной работе мы исследуем NQOBDD линейной и сверхлинейной ширины. Мы рассматриваем известную булеву функцию “Равенство” и сложность её вычисления в квантовых недетерминированных OBDD при использовании различного порядка считывания переменных. С использованием метода доказательства нижней оценки сложности NQOBDD, впервые представленного в материалах конференции [10], мы показываем, что при использовании “наихудшего порядка” квантовая недетерминированная сложность функции экспоненциальна, а при использовании “наилучшего” порядка, при котором классическая сложность для этой функции равна 3, квантовая сложность линейна. Мы показываем, что доказанные нижние оценки точны. На основе функции “Равенство” мы конструируем функцию, не чувствительную к используемому порядку считывания переменных входа, для которой доказываем нижнюю оценку сложности вычисления в квантовых недетерминированных OBDD и предлагаем квантовый алгоритм её вычисления. На основе полученных результатов мы представляем результат о иерархии классов сложности, основанных на модели недетерминированных квантовых OBDD сверхлинейной ширины.

Работа организована следующим образом В разделе 2 представлены определения моделей. Раздел 3 содержит вспомогательные сведения, используемые в доказательствах. Раздел 4 посвящен методу доказательства нижней оценки квантовых недетерминированных OBDD. Как было упомянуто выше, данный метод впервые был представлен в материалах конференции [10]. Раздел 5 посвящен исследованию сложности вычисления функции "Равенство" в квантовых недетерминированных OBDD при использовании различных порядков считывания переменных. Доказываются верхние и нижние оценки сложности. В разделе 6 мы определяем и исследуем функцию "XOR-перемешанное равенство". Доказывается экспоненциальная нижняя оценка сложности вычисления этой функции в квантовых недетерминированных OBDD при использовании любого порядка считывания. Приводится квантовый алгоритм для недетерминированного вычисления данных функции в модели OBDD. В разделе 7 приводится результат о иерархии классов сложности, основанных на квантовых недетерминированных OBDD большой ширины.

2 Определения моделей

В работе мы используем верхний индекс для нумерации векторов и наборов, нижний индекс – для нумерации элементов векторов и наборов.

Детерминированная ветвящаяся программа (BP – Branching Program) над множеством переменных $X = \{x_1, \ldots, x_n\}$ – это ориентированный ациклический граф с финальными вершинами, помеченными 0 и 1 (будем называть их отвергающими и принимающими вершинами, соответственно). Каждая внутренняя вершина помечена булевской переменной $x \in X$, и имеет два исходящих ребра, помеченных 0 и 1, соответственно. Ветвящаяся программа обрабатывает входной набор $\sigma \in \{0,1\}^n$ следующим образом. Вычисление 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. BP представляет булеву функцию $f : \{0,1\}^n \to \{0,1\}$, если для любого $\sigma \in \{0,1\}^n$ значение достигнутой финальной вершины совпадает со значением $f(\sigma)$.

Сложность $Size(P)$ ветвящейся программы P – это количество её внутренних вершин. *Длина* $Length(P)$ ветвящейся программы P – это максимальная длина пути из начальной вершины в конечную. Длина BP очевидным образом оценивает время, требуемое для вычисления функции f в худшем случае. Сложность BP оценивает память, затрачиваемую в процессе вычисления.

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

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

Ширина $Width(P)$ уровневой BP P – это максимальное число вершин на уровне. Очевидно, что $Size(P) \leq Length(P) \cdot Width(P)$.

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

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

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

Модель OBDD константной ширины, в которой используется естественный порядок считывания, а преобразования, соответствующие одному и тому же значению входа, одинаковы для всех уровней, эквивалентна модели конечных автоматов [12].

Недетерминированная OBDD (NOBDD) допускает переходы из вершины текущего уровня в более чем одну вершину последующего уровня при считывании одной и той же переменной. В этом случае для входного набора σ могут существовать несколько вычислительных путей. NOBDD P принимает входной набор σ, если существует вычислительный путь, соответствующий данному набору, который завершается в принимающей вершине. В противном случае, P отвергает набор σ.

Для определения квантовой OBDD нам понадобятся некоторые сведения из теории квантовых вычислений. Для большей информации см., например [13]. Квантовая система (QS) с d устойчивыми состояниями (использующая $\log d$ квантовых битов) может быть описана при помощи d-мерного комплекснозначного Гильбертова пространства $\mathcal{H}^d$. Чистое состояние QS – это элемент пространства $\mathcal{H}^d$, вектор-столбец $|\psi\rangle = (z_0, \ldots, z_{d-1})$ с единичной нормой (унитарный вектор): $|| \, |\psi\rangle \, || = \sqrt{\langle\psi|\psi\rangle} = 1$ ($\langle\psi|$ – сопряженный к $|\psi\rangle$ вектор-строка). Комплексное число z_i ($i = 0, \ldots, d-1$) называется амплитудой устойчивого состояния $|i\rangle$, где $|i\rangle$ обозначает унитарный вектор со значением 1 в позиции i (нумерация элементов вектора осуществляется с 0). Таким образом, чистое состояние – это суперпозиция устойчивых состояний QS с комплекснозначными амплитудами. Унитарная эволюция – это изменение состояния квантовой системы за определённый период времени, описывается d-мерной унитарной матрицей U. Матрица U называется унитарной, если выполняется $UU^\dagger = I$, где $U^\dagger$– транспонированная комплексносопряжённая к U матрица, I – единичная матрица. Унитарная эволюция является обратимым преобразованием.

Квантовое измерение – это процедура извлечения классической информации из квантового состояния. Ортогональное измерение QS описывается системой операторов $\mathcal{O} = \{P_1, \ldots, P_t\}$, действующих в $\mathcal{H}^d$ таких, что $P_i = P_i^\dagger$, $P_i^2 = P_i$, $P_iP_j = \mathbf{0}$, $\sum_{i=1}^t P_i = I$ ($i, j = 1, \ldots, t$, $i \neq j$,

$t \leq d$). Если $|\psi\rangle$ – состояние QS до измерения, то результатом измерения является одно из значений из множества $\{1, \ldots, t\}$. При этом:

(1) $p_k = ||P_k|\psi\rangle||^2$ – вероятность того, что исходом измерения является значение k;

(2) $|\psi'\rangle = \frac{P_k|\psi\rangle}{||P_k|\psi\rangle||}$ – состояние квантовой системы после измерения, результатом которого является значение k.

Квантовая OBDD Q ширины d и длины l ((d, l)-QOBDD) определяется как

$$Q = \left(|\psi^0\rangle, R, \mathcal{O}_{final}\right),$$

где $|\psi^0\rangle$ – начальная суперпозиция; R – последовательность (длины l) инструкций, содержащих d-мерные унитарные преобразования квантовой системы QS с d устойчивыми состояниями, определенная следующим образом:

$$R = \{\langle j_i, U_i(0), U_i(1)\rangle\}_{i=1}^{l},$$

где $U_i(0)$ и $U_i(1)$ – унитарные $(d \times d)$-матрицы, описывающие преобразования, применяемые на i-ом шаге, $\mathcal{O}_{final} = \{P_{accept}, P_{reject}\}$ – система операторов, задающих финальное измерение с исходами *accept* и *reject*, соответственно.

QOBDD Q обрабатывает вход $\sigma = \sigma_1 \ldots \sigma_n \in \{0, 1\}^n$ следующим образом. Q начинает работу в суперпозиции $|\psi^0\rangle$. Пусть после текущего шага Q находится в состоянии $|\psi\rangle$. Тогда на следующем i-ом шаге ($i = 1, \ldots, n$) работы Q считывает очередной символ σ_{j_i} входного слова $\sigma \in \Sigma^n$, определяемый последовательностью R инструкций программы, и преобразует текущую суперпозицию $|\psi\rangle$ в суперпозицию $|\psi'\rangle = U_i(\sigma_{j_i})|\psi\rangle$. После считывания входного набора производится измерение финальной суперпозиции $|\psi_{final}\rangle = U_n(\sigma_{i_n}) \cdots U_1(\sigma_{i_1})|\psi^0\rangle$. Если исход измерения *accept*, вход принимается, в противном случае – отвергается. Вероятность принятия слова σ определяется как

$$Pr^{Q}_{accept}(\sigma) = ||P_{accept}|\psi_{final}\rangle||^2.$$

Q недетерминированно вычисляет функцию f, если Q принимает вход σ с вероятностью > 0 тогда и только тогда, когда $f(\sigma) = 1$. Такую OBDD будем называть недетерминированной квантовой OBDD (NQOBDD). Q вычисляет функцию f без ошибки, если Q принимает с вероятностью 1 входы σ, для которых $f(\sigma) = 1$ и принимает с вероятностью 0 входы σ, для которых $f(\sigma) = 0$.

3 Сведения из линейной алгебры

Пусть V – векторное пространство над полем комплексных чисел с нормой $||\cdot||_2$. Через $\mathbf{0}$ обозначим нулевой элемент V.

Система векторов $\psi_1, \psi_2, \ldots, \psi_d \in V$ называется линейно зависимой, если существуют числа $\alpha_1, \alpha_2, \ldots, \alpha_d \in \mathbb{C}$, одновременно не равные нулю такие, что $\alpha_1\psi_1 + \cdots + \alpha_d\psi_d = \mathbf{0}$. Если это равенство выполняется только при $\alpha_1 = \alpha_2 = \cdots = \alpha_d = 0$, то система векторов называется линейно независимой. Два вектора ψ_1 и ψ_2 линейно зависимы тогда и только тогда, когда $\psi_1 = \alpha\psi_2$ или $\psi_2 = \beta\psi_1$ при некоторых α, β, т.е. когда векторы ψ_1 и ψ_2 коллинеарны.

Лемма 1. *Пусть $\psi_1, \psi_2, \ldots, \psi_d \in V$ – линейно независимая система векторов, U – унитарное преобразование в пространстве V. Тогда вектора $U\psi_1, U\psi_2, \ldots, U\psi_d$ линейно независимы.*

Доказательство. Унитарное преобразование является взаимнооднозначным преобразованием, которое можно рассматривать как переход к другому базису. Взаимнооднозначное линейное преобразование сохраняет свойство линейной независимости преобразуемых векторов. □

Свойства нормы:

(1) $||\psi|| = 0 \Rightarrow \psi = \mathbf{0}$;
(2) $\forall\psi, \phi \in V, ||\psi + \phi|| \le ||\psi|| + ||\phi||$ (неравенство треугольника);
(3) $\forall\alpha \in \mathbb{C}, \forall\psi \in V, ||\alpha\,\psi|| = |\alpha| \cdot ||\psi||$.

Лемма 2. *Пусть вектора $\psi_1, \ldots, \psi_m, \psi \in V$, $\psi_1, \ldots, \psi_m$ – линейно-независимы. Пусть U – линейное преобразование пространства V такое, что*

$$||U\psi_i|| = 0, i = 1, \ldots, m,$$
$$||U\psi|| \neq 0.$$

Тогда система векторов $\{\psi_1, \ldots, \psi_m, \psi\}$ линейно независима.

Доказательство. Докажем от противного. Предположим, вектор ψ линейно зависим с системой $\psi_1, \ldots, \psi_m$. Тогда существуют $\alpha_1, \alpha_2, \ldots, \alpha_m \in \mathbb{C}$, одновременно не равные нулю такие, что $\psi = \alpha_1\psi_1 + \cdots + \alpha_m\psi_m$.

В силу линейности имеем $U\psi = U(\alpha_1\psi_1 + \cdots + \alpha_m\psi_m) = \alpha_1 U\psi_1 + \cdots + \alpha_m U\psi_m$.

По свойству нормы имеем $||U\psi|| \le |\alpha_1| \cdot ||U\psi_1|| + \cdots + |\alpha_m| \cdot ||U\psi_m||$.

По условию леммы $||U\psi_1|| = \cdots = ||U\psi_m|| = 0$. Значит $||U\psi|| = 0$. Получили противоречие. □

4 Нижняя оценка ширины NQOBDD

Пусть $f : \{0,1\}^n \to \{0,1\}$ – произвольная булева функция, $\pi = (i_1, \ldots, i_n)$ – произвольная перестановка индексов $\{1, \ldots, n\}$. Для $X = \{x_1, \ldots, x_n\}$, целого k $(0 < k < n)$ обозначим $X_k^\pi = \{x_{i_1}, \ldots, x_{i_k}\}$. Набор значений $\sigma \in \{0,1\}^k$, сопоставленный переменным из множества X_k^π определяет подфункцию $f_{\pi,k}^\sigma : \{0,1\}^{n-k} \to \{0,1\}$.

Множество пар $S_k^\pi = \{(\sigma, \gamma) : \sigma \in \{0,1\}^k, \gamma \in \{0,1\}^{n-k}\}$ назовем строгим 1-полным множеством (*strong 1-fooling set*) для функции f, если выполняются следующие условия:

(1) $f_{\pi,k}^\sigma(\gamma) = 1$ для любой пары $(\sigma, \gamma) \in S_k^\pi$,

(2) для любых двух пар $(\sigma,\gamma),(\sigma',\gamma') \in S^\pi_k$ выполняется $f^{\sigma}_{\pi,k}(\gamma')=0$ и $f^{\sigma'}_{\pi,k}(\gamma)=0$.

Для двух наборов $\sigma,\sigma' \in \{0,1\}^k$ будем говорить что набор $\gamma \in \{0,1\}^{n-k}$ отличает набор σ от набора σ' если выполняется $f^{\sigma}_{\pi,k}(\gamma)>0$ и $f^{\sigma'}_{\pi,k}(\gamma)=0$. Отметим, что данное свойство не является симметричным.

Теорема 1. [10] *Для любой* QOBDD Q*, недетерминированно вычисляющей функцию* $f:\{0,1\}^n \to \{0,1\}$ *и использующей порядок* $\pi = (i_1,\dots,i_n)$ *считывания переменных, выполняется*

$$Width(Q) \geq \max_k |S^\pi_k|.$$

Доказательство. Пусть $d=\max_k |S^\pi_k|$, где $S^\pi_k = \{(\sigma^1,\gamma^1),\dots,(\sigma^d,\gamma^d)\}$ – строгое 1-полное множество для функции f. Предположим, что существует QOBDD Q, недетерминированно вычисляющая функцию f с использованием порядка π считывания переменных, ширина которой $< d$. Рассмотрим уровень k. Пусть $\Psi = \{|\psi(\sigma^j)\rangle \,|\, j=1,\dots,d\}$ – множество квантовых состояний, в которых находится программа после обработки входов $\sigma^1,\dots,\sigma^d$, т.е. $|\psi(\sigma^j)\rangle = U(\sigma^j)|\psi^0\rangle$.

Лемма 3. *Множество* Ψ *линейно независимо.*

Доказательство. Доказательство от противного. Предположим, что множество векторов Ψ является линейно зависимым множеством. Тогда существует состояние $|\psi\rangle = |\psi(\sigma^i)\rangle \in \Psi$, которое может быть представлено в виде линейной комбинации остальных векторов из множества Ψ:

$$|\psi(\sigma^i)\rangle = \sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_j |\psi(\sigma^j)\rangle,$$

при этом неверно, что $\alpha_j=0$ для всех j.

Пусть γ^i такое, что $(\sigma^i,\gamma^i)\in S^\pi_k$. Согласно определению строгого 1-полного множества, для всех входов σ^j $(j\neq i)$, выполняется $f^{\sigma^j}_{\pi,k}(\gamma^i)=0$, следовательно, программа Q принимает входы $\sigma^j\gamma^i$ с нулевой вероятностью:

$$Pr^Q_{accept}(\sigma^j\gamma^i) = ||P_{accept}U(\gamma^i)|\psi(\sigma^j)\rangle||^2 = 0.$$

После обработки входа $\sigma^i\gamma^i$ финальное состояние программы

$$|\psi(\sigma^i\gamma^i)\rangle = U(\gamma^i)|\psi(\sigma^i)\rangle = U(\gamma^i)\sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_j |\psi(\sigma^j)\rangle$$

и по свойству линейности преобразований

$$|\psi(\sigma^i\gamma^i)\rangle = \sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_j U(\gamma^i)|\psi(\sigma^j)\rangle = \sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_j |\psi(\sigma^j\gamma^i)\rangle.$$

Тогда

$$Pr^{Q}_{accept}(\sigma^i\gamma^i) = ||P_{accept}|\psi(\sigma^i\gamma^i)\rangle||^2 = ||\sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_j P_{accept}|\psi(\sigma^j\gamma^i)\rangle||^2 \leq$$

$$(\sum_{\substack{j=1\\ j\neq i}}^{d} |\alpha_j|\, ||P_{accept}\psi(\sigma^j\gamma^i)||)^2 = 0.$$

Следовательно, программа Q неверно вычисляет функцию f, поскольку принимает набор $\sigma^i\gamma^i$, для которого $f^{\sigma^i}_{\pi,k}(\gamma^i) > 0$, с нулевой вероятностью. Получили противоречие. Следовательно, наше предположение о том, что множество Ψ линейно зависимо, неверно. $\square$

Так как множество Ψ состояний программы, достижимых на уровне k линейно независимо, размерность пространства состояний программы Q не может быть меньше d. Следовательно, $Width(Q) \geq d$. $\square$

5 Функция "Равенство"

Функция "Равенство" $\mathtt{EQ_{2n}} : \{0,1\}^{2n} \to \{0,1\}$ определяется следующим образом:

$$\mathtt{EQ_{2n}}(\sigma) = \begin{cases} 1, & \text{если } \sigma_1 \dots \sigma_n = \sigma_{n+1} \dots \sigma_{2n}, \\ 0, & \text{иначе.} \end{cases}$$

Известно, что сложность вычисления данной функции критическим образом зависит от порядка считывания переменных. Так, ширина классической OBDD (детерминированной и недетерминированной), вычисляющей EQ_{2n} равна 3, если переменные считываются в порядке x_1, x_{n+1}, x_2, x_{n+2} $\dots$, x_n, x_{2n} и равна $\Omega(2^n)$, если сначала считываются переменные первой половины набора и только потом – переменные второй половины набора.

Покажем, что сложность недетерминированной QOBDD для этой функции при использовании "наилучшего" порядка считывания линейна.

Теорема 2. *Любая* QOBDD*, недетерминированно вычисляющая функцию* $\mathtt{EQ_{2n}}$ *и использующая порядок считывания переменных* $\pi = (1, n+1, 2,\ n+2, \dots,\ n, 2n)$*, имеет ширину не менее* $n+1$.

Доказательство. Пусть Q – NQOBDD, вычисляющая $\mathtt{EQ_{2n}}$ и считывающая переменные в порядке $\pi = (1, n+1, 2,\ n+2, \dots,\ n, 2n)$.

Вычисление на входе $\sigma = \sigma_1 \cdots \sigma_n$ начинается из начальной суперпозиции $|\psi^0\rangle$. На шаге l программа считывает переменную $x_{i_l} = \sigma_{i_l}$ и преобразует суперпозицию $|\psi(\sigma_{i_1} \cdots \sigma_{i_{l-1}})\rangle$ в суперпозицию $|\psi(\sigma_{i_1} \cdots \sigma_{i_l})\rangle$. После считывания входного слова Q производит финальное измерение финальной суперпозиции $|\psi(\sigma_{i_1} \cdots \sigma_{i_n})\rangle$ и принимает входной набор с вероятностью $Pr^{Q}_{accept}(\sigma) = ||P_{accept}|\psi(\sigma_{i_1} \cdots \sigma_{i_n})\rangle||^2$.

На каждом уровне l программы Q будем рассматривать множество Ψ_l квантовых состояний, достижимых программой на этом уровне, а также подмножество $\Phi_l \subseteq \Psi_l$ состояний, которые являются линейно независимыми векторами.

Лемма 4. *Пусть $|\psi^1\rangle, \ldots, |\psi^m\rangle, |\psi\rangle \in \Psi_l$ ($m \geq 1$) и $|\psi^1\rangle, \ldots, |\psi^m\rangle$ – линейно независимы, где $|\psi^i\rangle = |\psi(\sigma^i)\rangle$ для $i = 1, \ldots, m$ и $|\psi\rangle = |\psi(\sigma)\rangle$. Если существует строка $\gamma \in \{0,1\}^{n-l}$, отличающая строку σ от каждой из строк $\sigma^1, \ldots, \sigma^m$, то множество $\{|\psi^1\rangle, \ldots, |\psi^m\rangle, |\psi\rangle\}$ линейно независимо.*

Доказательство. Пусть $U = U_n(\gamma_{n-l}) \cdots U_{l+1}(\gamma_1)$. Тогда выполняется $||P_{accept}U|\psi^i\rangle|| = 0$ для всех $i = 1, \ldots, m$, и $||P_{accept}U|\psi\rangle|| > 0$. Согласно Лемме 2, множество $\{|\psi^1\rangle, \ldots, |\psi^m\rangle, |\psi\rangle\}$ является линейно независимым. □

Индукцией по i ($i = 1, \ldots, n$) покажем, что после считывания i-й пары: значения переменной x_i и значения переменной x_{n+i} (на уровне $l = 2i$) $|\Phi_{2i}| \geq i + 1$.

База индукции. $\Phi_0 = \{|\psi^0\rangle\}$. На первом шаге после считывания $x_1 = \sigma_1$ ($\sigma_1 \in \{0,1\}$), согласно Лемме 4, множество $\Phi_1 = \{|\psi(0)\rangle, |\psi(1)\rangle\}$ является линейно независимым множеством, так как набор 1^{2n-1} отличает строку 1 от строки 0. После считывания $x_{n+1} = \sigma_{n+1}$ ($\sigma_{n+1} \in \{0,1\}$), множество состояний $\Phi_2 = \{|00\rangle, |10\rangle\}$ является линейно независимым по Лемме 1. Следовательно, $|\Phi_2| \geq 2$.

Индукционный шаг (для $i = 2, \ldots, n$). По предположению индукции $\Psi_{2(i-1)}$ содержит не менее i векторов. Обозначим их $|\psi^{j_0}\rangle, \ldots, |\psi^{j_{i-1}}\rangle$, а соответствующие им частичные входы $\sigma^{j_0}, \ldots, \sigma^{j_{i-1}}$. После считывания $x_i = \sigma_i$, по Лемме 1 множество $\Phi^0_{2(i-1)} = \{U(0)|\psi^{j_0}\rangle, \ldots, U(0)|\psi^{j_{i-1}}\rangle\}$ является линейно независимым, вектор $|\psi(1^{2i-1})\rangle = U_{2i-1}(1) \cdots U_1(1)|\psi^0\rangle$ не входит в $\Phi^0_{2(i-1)}$, при этом строка $1^{2n-2i+1}$ отличает строку 1^{2i-1} от любой из строк $\sigma^{j_0}0, \ldots, \sigma^{j_{i-1}}0$. Значит множество $\Phi_{2i-1} = \Phi^0_{2(i-1)} \cup \{|\psi(1^{2i-1})\rangle\}$ является линейно независимым. После считывания $x_{n+i} = \sigma_{n+i}$, по Лемме 1 множество $\Psi_{2i} = \Phi^0_{2i-1} = \{U(0)|\psi\rangle \,|\, |\psi\rangle \in \Phi_{2i-1}\}$ является линейно независимым. Следовательно, $|\Phi_{2i}| \geq i + 1$.

После считывания n-ой пары x_n, x_{2n} на уровне $l = 2n$ получаем $|\Phi_{2n}| \geq n + 1$. Таким образом, размерность пространства состояний программы Q не менее $n + 1$, что завершает доказательство теоремы. □

Покажем, что доказанная нижняя оценка точна.

Теорема 3. *Существует* QOBDD Q *ширины* $n+1$, *вычисляющая функцию* $\mathtt{EQ_{2n}}$ *без ошибки и использующая порядок считывания переменных* $\pi = (1,\, n+1,\, 2,\, n+2,\, \ldots,\, n,\, 2n)$.

Доказательство. Программа Q использует регистр из $n + 1$ состояния $s_0, s_1, \ldots, s_n$, где s_0 – начальное и принимающее состояние.

При считывании пары значений переменных x_i, x_{n+i} $(i = 1, \dots, n)$ программа применяет преобразования:

- $|s_0\rangle \to |s_i\rangle$, $|s_i\rangle \to |s_0\rangle$, если считанное значение 1;
- $|s_0\rangle \to |s_0\rangle$, $|s_i\rangle \to |s_i\rangle$, если считанное значение 0;
- $|s_j\rangle \to |s_j\rangle$, для всех $j = 1, \dots, n, j \neq i$.

Докажем корректность работы программы. Пусть вход σ такой, что $\mathtt{EQ_{2n}}(\sigma) = 1$. Это означает, что $\sigma_i = \sigma_{n+i}$ $\forall i = 1, \dots, n$. В этом случае программа будет завершать обработку каждой пары в состоянии $|s_0\rangle$ и примет такой вход с вероятностью 1.

Пусть вход σ такой, что $\mathtt{EQ_{2n}}(\sigma) = 0$. Это означает, что $\exists i \in \{1, \dots, n\}$ для которого $\sigma_i \neq \sigma_{n+i}$. После обработки этой пары программа останется в состоянии s_i и, таким образом, амплитуда состояния s_0 до конца обработки будет равна 0. Вероятность принятия такого набора равна 0. □

Теорема 4. *Любая* QOBDD*, недетерминированно вычисляющая функцию* $\mathtt{EQ_{2n}}$ *и считывающая переменные одной из половин набора после считывания всех переменных второй половины набора, имеет ширину не менее* 2^n.

Доказательство. Пусть Q – QOBDD, недетерминированно вычисляющая $\mathtt{EQ_{2n}}$, использующая порядок считывания $\pi = (i_1, \dots, i_{2n})$, при котором любая переменная одной из половин набора считывается после того, как считаны все переменные другой половины набора.

Множество $S_n^\pi = \{(\sigma, \gamma) : \sigma \in \{0,1\}^n, \gamma \in \{0,1\}^n\}$ является строгим 1-полным множеством для функции $\mathtt{EQ_{2n}}$. Согласно Теореме 1, $Width(Q) \geq |S_n^\pi|$. Заметим, что $|S_n^\pi| = 2^n$, что завершает доказательство теоремы. □

Теорема 5. *Пусть* $\pi = (i_1, \dots, i_{2n})$ *– произвольная перестановка индексов* $\{1, \dots, 2n\}$ *такая, что* $\pi(i) \leq n$ $\forall i \leq n$ *и* $\pi(i) > n$ $\forall i > n$ *либо* $\pi(i) > n$ $\forall i \leq n$ *и* $\pi(i) \leq n$ $\forall i > n$. *Существует* QOBDD *ширины* 2^n, *недетерминированно вычисляющая функцию* $\mathtt{EQ_{2n}}$ *и считывающая переменные в порядке* π.

Доказательство. Пусть Q – NQOBDD, вычисляющая $\mathtt{EQ_{2n}}$, использующая порядок $\pi = (i_1, \dots, i_{2n})$ считывания переменных, удовлетворяющий условию теоремы. Для определенности считаем, что Q сначала считывает переменные первой половины набора, потом – переменные второй половины набора (противоположный случай доказывается аналогично). Q имеет 2^n состояний $s_0, \dots s_{2^n-1}$, где s_0 – начальное и принимающее состояние. При считывании переменной $x_i = \sigma_i$ $(i = 1, \dots, 2n,\ \sigma_i \in \{0,1\})$ Q применяет преобразования:

- $|s_j\rangle \to |s_{j+2^{i-1}\sigma_i \pmod{2^n}}\rangle$, $j = 0, \dots, 2^n - 1$, если $i \in \{1, \dots, n\}$,
- $|s_j\rangle \to |s_{j-2^{i-n-1}\sigma_i+2^n \pmod{2^n}}\rangle$, $j = 0, \dots, 2^n - 1$, если $i \in \{n + 1, \dots, 2n\}$.

После считывания первой половины входа состояние программы будет равно $|s_{m(\sigma_1\ldots\sigma_n)}\rangle$, где $m(\sigma_1\ldots\sigma_n)$ – целое число, двоичное представление которого равно $\sigma_1\ldots\sigma_n$. После считывания второй половины набора состояние программы равно $|s_{m(\sigma_1\ldots\sigma_n)+2^n-m(\sigma_{n+1}\ldots\sigma_{2n}) \pmod{2^n}}\rangle$. Если $\sigma_1\ldots\sigma_{n/2}=\sigma_{n+1}\ldots\sigma_{2n}$, финальное состояние программы будет $|s_0\rangle$ и программа примет такой набор с вероятностью 1. В противном случае программа завершит обработку входа в состоянии, отличном от $|s_0\rangle$ и примет такой набор с вероятностью 0. □

6 Функция "XOR-перемешанное равенство"

Как было показано в предыдущем разделе, сложность вычисления функции $\mathtt{EQ_{2n}}$ зависит от того, в каком порядке считываются переменные. Для устранения зависимости сложности вычисления от используемого порядка считывания, используют различные приемы для определения функций, при которых порядок следования битов входа определятся самим входным набором (или его частью). Для таких функций не удаётся подобрать оптимальный порядок считывания битов входа. В частности, в работе [9] рассматривалась функция "Перемешанное равенство" $\mathtt{EQS_n}$, определённая на основе функции "Равенство" и сложность её вычисления в классических детерминированных и недетерминированных OBDD.

В данной работе мы определим и исследуем функцию $\mathtt{EQXS_{2n}}$ (Equality-Xor-Shuffled), которая задаётся на основе функции $\mathtt{EQ_{2n}}$ с использованием приема, аналогичного описанному в работе [15]. Формально, функция $\mathtt{EQXS_{2n}}$ определяется следующим образом.

Пусть $x_1, x_2, x_3, \ldots, x_n, y_1, y_2, y_3, \ldots, y_n$ – последовательность переменных, от которых зависит функция (n кратно 2). Назовём переменные $x_1, \ldots, x_n$ – "переменными принадлежности", переменные $y_1, \ldots, y_n$ – "переменными значения". Переменную значения y_i назовём соответствующей переменной принадлежности x_i. По входной последовательности $\sigma\gamma \in \{0,1\}^n \times \{0,1\}^n$ формируются новые битовые последовательности τ, α, β:

- последовательность τ имеет длину n и формируется по значениям $\sigma_1, \ldots, \sigma_n$ переменных принадлежности $x_1, \ldots, x_n$ следующим образом: $\tau_1 = \sigma_1$, $\tau_i = \tau_{i-1} \oplus \sigma_i$ $(i = 2, \ldots, n)$;
- последовательности α, β формируются по переменным значения, начиная с пустых последовательностей: для $i = 1, \ldots, n$ бит значения γ_i дописывается к последовательности α, если $\tau_i = 0$, в противном случае γ_i дописывается к последовательности β;

Например, для набора $\sigma = 1011001101$ получим последовательности $\tau = 10011$, $\alpha = 10$, $\beta = 011$. Функция $\mathtt{EQXS_{2n}} = 1$ тогда и только тогда, когда $\alpha = \beta$.

Теорема 6. *Существует квантовая* OBDD, *вычисляющая функцию* $\mathtt{EQXS_{2n}}$ *с нулевой ошибкой, и имеющая ширину* $2^{n/2}(2n+4)$.

Доказательство. Программа Q считывает переменные в порядке x_1, y_1, x_2, y_2, …, x_n, y_n. Состояние программы хранится в регистрах: $|num_\alpha\rangle$, $|num_\beta\rangle$, $|\phi\rangle$, $|b\rangle$:

- однокубитный "регистр активности" $|b\rangle$ хранит значение бита τ_i, соответствующего последнему считанному биту принадлежности;
- регистр $|num\rangle$ является ортогональной суммой двух регистров $|num_\alpha\rangle$ и $|num_\beta\rangle$, которые устроены одинаково и хранят номер последнего считанного бита последовательности α и β, соответственно;
- регистр $|\phi\rangle$ хранит информацию о значении $(\alpha-\beta) \bmod 2^{n/2}$ (будем трактовать последовательности α, β как двоичные записи целых чисел);

Опишем подробнее каждый из регистров.

Регистр $|b\rangle$ имеет два состояния $|0\rangle$ и $|1\rangle$, изменяет свое состояние при считывании бита принадлежности $\sigma \in \{0,1\}$ и не меняется при считывании бита значения. Преобразования:

$$U(\sigma) = \begin{cases} NOT, & \text{если } \sigma = 1, \\ I, & \text{если } \sigma = 0, \end{cases}$$

где NOT – однокубитное преобразование инвертирования квантового бита, I – тождественное преобразование.

Состояние данного регистра хранит значение текущего бита последовательности τ и управляет преобразованиями на следующем шаге. Зона активности регистра $|num\rangle$ ($|num_\alpha\rangle$ или $|num_\beta\rangle$) меняется, если считанный бит принадлежности равен 1, и сохраняется, если считанный бит принадлежности равен 0. Таким образом, на следующем шаге при считывании бита значения преобразования будут производиться в регистре $|num_\alpha\rangle$, если очередной бит значения принадлежит последовательности α и в регистре $|num_\beta\rangle$, если бит значения принадлежит β.

Каждый из регистров $|num_\alpha\rangle$ и $|num_\beta\rangle$ имеет $n/2+1$ состояние

$$s_0, s_1, \ldots s_{n/2}$$

(состоит из $\log(n/2+1)$ кубитов). Начальное состояние $|s_0\rangle$. При считывании очередного бита принадлежности происходит преобразование $|s_i\rangle \to |s_{i+1 \bmod (n/2+1)}\rangle$ в регистре $|num_\alpha\rangle$, если значение регистра активности $|b\rangle = |0\rangle$, и в регистре $|num_\beta\rangle$, если $|b\rangle = |1\rangle$.

$2^{n/2}$-кубитный регистр $|\phi\rangle$ хранит неотрицательное значение $c = (\alpha - \beta) \bmod 2^{n/2}$. Устойчивые состояния регистра соответствуют значениям $0, 1, \ldots, 2^{n/2}-1$. При считывании бита из последовательности α (β) выполняется преобразование $|c\rangle \to |c + \alpha_j 2^{j-1} \pmod{2^{n/2}}\rangle$ ($|c\rangle \to |c - \beta_j 2^{j-1} + 2^{n/2} \pmod{2^{n/2}}\rangle$), если последний считанный бит значения является j-м битом последовательности α (β). Управляют изменением регистра $|\phi\rangle$ регистры $|num\rangle$ и $|b\rangle$.

Состояние программы Q описывается в виде

$$|\psi\rangle = |b\rangle \otimes \frac{1}{\sqrt{2}}(|num_\alpha\rangle \oplus |num_\beta\rangle) \otimes |\phi\rangle.$$

Начальное состояние

$$|\psi^0\rangle = |0\rangle \otimes \frac{1}{\sqrt{2}}(|s_0\rangle \oplus |s_0\rangle) \otimes |0\rangle.$$

Подпространство принимающих состояний – это подпространство, в котором состояние регистров $|num\rangle \otimes |\phi\rangle = \frac{1}{\sqrt{2}}(|s_{n/2}\rangle \oplus |s_{n/2}\rangle) \otimes |0\rangle$.

Опишем работу программы. Q попеременно считывает биты принадлежности и соответствующие им биты значения. На очередном шаге при считывании бита принадлежности σ_i $(i = 1, \dots, n)$ программа применяет преобразование $U(\sigma_i)$ к регистру $|b\rangle$. К регистрам $|num\rangle$ и $|\phi\rangle$ применяется тождественное преобразование.

При считывании бита значения γ_i $(i = 1, \dots, n)$ программа применяет преобразование U_ϕ, управляемое регистрами $|num\rangle$ и $|b\rangle$ и воздействующее на регистр $|\phi\rangle$. Затем применяет преобразование U_{num}, управляемое регистром $|b\rangle$ и воздействующее на регистр $|num\rangle$. При этом к регистру $|\phi\rangle$ применяется тождественное преобразование. Матрицы данных преобразований описываются следующим образом:

$$U_\phi(0) = \mathbf{I}, \qquad \mathbf{U}_\phi(\mathbf{1}) = \begin{pmatrix} U_\phi^\alpha & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & U_\phi^\beta \end{pmatrix},$$

где

$$U_\phi^\alpha = \begin{pmatrix} M_0^\alpha & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & M_1^\alpha & & \\ \vdots & & \ddots & \vdots \\ & & & \\ \mathbf{0} & \mathbf{0} & \dots & M_{n/2-1}^\alpha \end{pmatrix}, \quad U_\phi^\beta = \begin{pmatrix} M_0^\beta & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & M_1^\beta & & \\ \vdots & & \ddots & \vdots \\ & & & \\ \mathbf{0} & \mathbf{0} & \dots & M_{n/2-1}^\beta \end{pmatrix}.$$

Здесь M_j^α $(j = 0, \dots n/2 - 1)$– унитарные (перестановочные) матрицы, реализующие преобразования:

$$|c\rangle \to |(c + 2^j) \bmod 2^{n/2}\rangle.$$

Соответственно, M_j^β $(j = 0, \dots n/2 - 1)$– унитарные матрицы, реализующие преобразования:

$$|c\rangle \to |(c - 2^j + 2^{n/2}) \bmod 2^{n/2}\rangle,$$

$c \in \{0, \ldots, 2^{n/2} - 1\}$.

$$U_{num} = \begin{pmatrix} S & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & S \end{pmatrix} \otimes \mathbf{I},$$

где $S = \begin{pmatrix} 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \ldots & 1 & 0 \end{pmatrix}$ – матрица циклического сдвига, применяемая к регистру $|num_\alpha\rangle$ и регистру $|num_\beta\rangle$.

Докажем корректность работы программы. Пусть вход $\sigma\gamma$ такой, что $\mathtt{EQXS_{2n}}(\sigma\gamma) = 1$. Это означает одновременное выполнение двух условий:

(1) $|\alpha| = |\beta|$,
(2) $\forall i : \alpha_i = \beta_i$.

Выполнение условия 1 означает, что $|\alpha| = |\beta| = n/2$ и после обработки входа состояние регистров $|num_\alpha\rangle$ и $|num_\beta\rangle$ равно $|s_{n/2}\rangle$. Выполнение условия 2 означает, что числа, двоичными представлениями которых которых являются последовательности α и β, равны, откуда следует, что финальное состояние регистра $|\phi\rangle$ равно $|0\rangle$. Поэтому после обработки входа финальное состояние программы является одним из двух:

$$|\psi_{final}\rangle = |0\rangle \otimes 1/\sqrt{2}(|s_{n/2}\rangle \oplus |s_{n/2}\rangle) \otimes |0\rangle,$$

$$|\psi_{final}\rangle = |1\rangle \otimes 1/\sqrt{2}(|s_{n/2}\rangle \oplus |s_{n/2}\rangle) \otimes |0\rangle.$$

Вероятность принятия такого набора равна 1.

Пусть $\sigma\gamma$: $\mathtt{EQXS_{2n}}(\sigma\gamma) = 0$. В этом случае выполняется по крайней мере одно из двух условий:

(1) $|\alpha| \neq |\beta|$,
(2) $\exists i : \alpha_i \neq \beta_i$.

Если произошел случай 1, то $|\alpha| \neq n/2$ и $|\beta| \neq n/2$, следовательно, после завершения работы и в регистре $|num_\alpha\rangle$ и в регистре $|num_\beta\rangle$ амплитуда состояния $|s_{n/2}\rangle$ равна 0. Следовательно, вероятность принятия таких наборов равна 0.

Если $|\alpha| = |\beta|$, это означает, что произошел случай 2 и $\alpha \neq \beta$. Покажем, что в этом случае состояние регистра $|\phi\rangle$ не равно $|0\rangle$. Действительно, предположим, это это не так, и $c = \alpha - \beta + 2^{n/2} = 0 \mod 2^{n/2}$. Это означает, что $\alpha = \beta \mod 2^{n/2}$, но так как $\alpha < 2^{n/2}$ и $\beta < 2^{n/2}$, последнее равенство возможно только если $\alpha = \beta$. Следовательно, вероятность принятия в данном случае также равна 0.

Ширина итоговой программы равна $2 \cdot 2 \cdot (n/2 + 1) \cdot 2^{n/2} = 2^{n/2}(2n + 4)$. $\square$

Следствие 1. *Функция* $\mathtt{EQXS_{2n}}$ *вычислима квантовой недетерминированной* OBDD *ширины* $2^{n/2}(2n+4)$.

Доказательство. Квантовая OBDD, построенная в доказательстве Теоремы 6, вычисляет функцию $\mathtt{EQXS_{2n}}$ с нулевой ошибкой, а следовательно недетерминированно. □

Теорема 7. *Для любого порядка считывания переменных* QOBDD, *недетерминированно вычисляющая функцию* $\mathtt{EQXS_{2n}}$, *имеет ширину* $\Omega(2^{n/2})$.

Доказательство. Пусть Q – произвольная QOBDD, недетерминированно вычисляющая функцию $\mathtt{EQXS_{2n}}$ и использующая порядок π считывания переменных. Зафиксируем значения переменных принадлежности таким образом, что в соответствии с порядком π ровно $n/2$ первых считанных переменных значения принадлежали последовательности α, $n/2$ последних считанных переменных значения принадлежали последовательности β. Работу программы Q на получившихся наборах можно рассматривать как вычисление функции $\mathtt{EQ_n}$, при использовании порядка, когда одна половина набора считывается строго после другой половины. Согласно Теореме 4, ширина программы в этом случае $\Omega(2^{n/2})$. □

7 Иерархия для NQOBDD

Обозначим через $\mathsf{NOBDD_n^d}$, and $\mathsf{NQOBDD_n^d}$ классы булевых функций, зависящих от n переменных, вычислимых недетерминированными и квантовыми недетерминированными OBDD ширины не более d, соответственно.

В работе [10] представлена иерархия классов сложности для квантовых и классических недетерминированных OBDD линейной и сублинейной ширины. В частности, было показано следующее:

(1) Для любых $n > 1$ и $1 < d \le n$ выполняется

$$\mathsf{NQOBDD_n^{d-1}} \subsetneq \mathsf{NQOBDD_n^{d}}.$$

(2) $\mathsf{NQOBDD_n^{d_1}}$ и $\mathsf{NOBDD_n^{d_2}}$ несравнимы для любых пар (d_1, d_2), удовлетворяющих условию $1 < d_1, d_2 \le n/2$.

В данной работе мы представляем иерархию для квантовых недетерминированных OBDD сверхлинейной ширины.

Теорема 8. *Для любых* $n > 3$, $2 \le d \le 2^{n/4}$ *выполняется*

$$\mathsf{NQOBDD_n^{d-1}} \subsetneq \mathsf{NQOBDD_n^{4d(\log d+1)}}.$$

Доказательство. Включение $\mathsf{NQOBDD_n^{d-1}} \subseteq \mathsf{NQOBDD_n^{4d(\log d+1)}}$ очевидно. Покажем, что эти классы не совпадают. На основе функции $EQXS_n$ определим функцию $EQXS_n^k$, где $k \le n$, k кратно 4, при этом $EQXS_n^k$ существенным образом зависит от первых k переменных и $EQXS_n^k \equiv EQXS_k$. Согласно Теореме 6, $EQXS_n^{4\log d} \in \mathsf{NQOBDD_n^{4d(\log d+1)}}$. Согласно Теореме 7, $EQXS_n^{4\log d} \notin \mathsf{NQOBDD_n^{d-1}}$. □

References

[1] I. Wegener, *Branching programs and binary decision diagrams: theory and applications*, Society for Industrial and Applied Mathematics, 2000.

[2] A. Cobham, *The recognition problem for the set of perfect squares*, Proc. of the 7th Symposium on Switching an Automata Theory (SWAT), (1996), 78–87.

[3] P.Pudlak, S.Zak, *Space complexity of computations*, Technical report, Univ. Prague, 1983.

[4] F. Ablayev, A. Gainutdinova, M. Karpinski, *On Computational Power of Quantum Branching Programs*, Proc. of the 13th Intern. Symposium, Fundamentals of Computation Theory (FCT 2001, Riga, Latvia), LNCS, **2138**, (2001), 59–70.

[5] M. Nakanishi, K. Hamaguchi, T. Kashiwabara, *Ordered quantum branching programs are more powerful than ordered probabilistic branching programs under a bounded-width restriction*, Computing and Combinatorics: 6th Annual Intern. Conference, COCOON 2000, Proc., **2138**, (2000), 467–476.

[6] M. Sauerhoff, D. Sieling, *Quantum branching programs and space-bounded nonuniform quantum complexity*, Theoretical Computer Science, **334**:1–3, 177–225.

[7] F. Ablayev, M. Karpinski, *On the power of randomized branching programs*, Proc. ICALP, LNCS., **1099**, (1996), 348–356.

[8] F. Ablayev, A. Gainutdinova, M. Karpinski, C. Moore, C. Pollette, *On the computational power of probabilistic and quantum branching program*, Information and Computation, **203**:2, (2005), 145–162.

[9] F. Ablayev, A. Gainutdinova, K. Khadiev, A. Yakaryılmaz, *Very narrow quantum OBDDs and width hierarchies for classical OBDDs*, Lobachevskii Journal of Mathematics, **37**, (2016), 670–682.

[10] A. Gainutdinova, A. Yakaryılmaz, *Nondeterministic unitary OBDDs*, Computer Science–Theory and Applications: 12th Intern. Computer Science Symposium in Russia, Proc., **10304**, (2017), 126–140.

[11] C. Meinel, T. Theobald, *Algorithms and Data Structures in VLSI Design: OBDD-foundations and applications*, Springer Science & Business Media, 1998.

[12] F. Ablayev, A. Gainutdinova, *Complexity of quantum uniform and nonuniform automata*, In Developments in Language Theory, LNCS, **3572**, (2005), 78–87.

[13] M.Nielsen, I. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, 2000.

[14] F. Ablayev, A. Gainutdinova, K. Khadiev, A. Yakaryılmaz, *Very narrow quantum OBDDs and width hierarchies for classical OBDDs*, Proc. DCFS 2014, LNCS, **8614**, (2014), 53–64.

[15] K. Khadiev, A.Khadieva A., *Reordering method and hierarchies for quantum and classical ordered binary decision diagrams*, Computer Science–Theory and Applications: 12th Intern. Computer Science Symposium in Russia, Proc., **10304**, (2017), 162–175.

Aida Faritovna Gainutdinova
Kazan (Volga region) Federal University,
18, Kremlyovskaya street,
420008, Kazan, Russia
Email address: aida.ksu@gmail,com