Московский государственный технический университет имени Н.Э. Баумана

На правах рукописи

Ключарёв Пётр Георгиевич

АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ МОДЕЛИРОВАНИЯ КВАНТОВОГО КОМПЬЮТЕРА

Специальность 05.13.11 — математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей.

Автореферат диссертации
на соискание ученой степени
кандидата технических наук

Москва
2009

Работа выполнена в Московском государственном техническом университете им. Н.Э. Баумана

Научный руководитель: кандидат технических наук, доцент
Медведев Николай Викторович

Официальные оппоненты: доктор технических наук, профессор
Шеремет Игорь Анатольевич

кандидат технических наук, доцент
Рудаков Игорь Владимирович

Ведущая организация: Учреждение Российской академии наук
Вычислительный центр им. А. А. Дородницына РАН.

Защита состоится 4 июня 2009 г. в 14 часов 30 минут на заседании диссертационного совета Д 212.141.10 в Московском государственном техническом университете им. Н.Э. Баумана по адресу: 105005, Москва, 2-ая Бауманская, д.5.

С диссертацией можно ознакомиться в библиотеке Московского государственного технического университета им. Н.Э. Баумана.

Автореферат разослан 27 апреля 2009 г.

Ученый секретарь
диссертационного совета  к.т.н., доцент Иванов С.Р.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. В настоящее время активно развивается теория квантовых вычислений. Несмотря на то, что идея квантового компьютера была высказана еще Р. Фейнманом в 1982 г. и с тех пор проводятся научные исследования по этой тематике, квантовые компьютеры еще не созданы. Однако, уже сейчас ясно, что теоретических ограничений для этого нет. Кроме того, имеются определенные достижения в области теории квантовых вычислений.

Несмотря на то, что квантовому компьютеру присущи некоторые особенности (например, вычисления на квантовом компьютере могут быть только обратимыми), любой алгоритм, предназначенный для классического компьютера, можно реализовать и для квантового. Кроме того, квантовый компьютер обладает серьезными преимуществами, по сравнению с классическим компьютером – квантовым параллелизмом, который позволяет одновременно производить вычисления с различными исходными данными. Существуют так называемые квантовые алгоритмы, то есть алгоритмы, использующие преимущества квантового компьютера. Квантовые алгоритмы позволяют решать некоторые задачи значительно более эффективно по сравнению с классическими алгоритмами.

Квантовых алгоритмов на настоящий момент известно мало. Наиболее важные – это квантовый алгоритм факторизации натуральных чисел и квантовой алгоритм дискретного логарифмирования, разработанные П. Шором (1994 г.), а также алгоритм поиска, разработанный Л. Гровером (1996 г.). Такое положение вещей, по-видимому, обусловлено в частности отсутствием возможности запуска и отладки сколь-нибудь сложных квантовых алгоритмов. В то же время существующие квантовые алгоритмы имеют значительно меньшую вычислительную сложность, по сравнению с классическими аналогами. Так, квантовые алгоритмы факторизации натуральных чисел и дискретного логарифмирования имеет полиномиальную сложность, в то время как классические аналоги имеют надполиномиальную сложность. Это позволяет надеяться на то, что возможно построение эффективных квантовых алгоритмов для решения каких-либо других задач.

В связи с наличием эффективных квантовых алгоритмов факторизации целых чисел и вычисления дискретного логарифма, можно утверждать, что разработка практических образцов квантовых компьютеров приведет к тому, что основные ассиметричные криптографические алгоритмы окажутся нестойкими. Такие криптоалгоритмы используются для шифрования данных, для электронных цифровых подписей, а также для распределения криптографических ключей. Нестойкость этих криптоалгоритмов повлечет за собой нарушение секретности больших объемов информации во всем мире, что приведет к серьезным и труднопредсказуемым последствиям для информационной безопасности государств, различных предприятий, в том числе банков, а также физических лиц.

В условиях отсутствия практических квантовых компьютеров, встает задача разработки комплекса программ для моделирования квантового компьютера. Следует заметить, что такой комплекс программ ни коем образом не мо-

жет заменить квантового компьютера. Его возможности сильно ограничены, однако он позволит запускать и отлаживать квантовые программы, что очень ценно как для разработки новых квантовых алгоритмов, так и для целей обучения студентов основам квантовых вычислений. Курсы, посвященные квантовым вычислениям, в настоящее время читаются во многих высших учебных заведениях России, Европы и США. В ряде ВУЗов, например в Калифорнийском Техническом Университете и в МГУ им. Ломоносова, открыты кафедры квантовых вычислений.

Вместе с тем, возникает задача разработки языка для описания алгоритмов для квантового компьютера. Несмотря на то, что в настоящее время разработано уже несколько таких языков, только для одного из них создан интерпретатор, позволяющий выполнять программы, написанные на этом языке.

Существующие программные средства для имитации квантового компьютера имеют ограничения. В частности, они могут имитировать квантовый компьютер, имеющий память размером не более 20 – 32 кубита. Кроме того, они обладают низкой скоростью имитации. Эти обстоятельства не позволяют производить имитацию сложных квантовых алгоритмов.

Все это позволяет утверждать, что тема настоящей диссертации является актуальной.

Объектом исследования является математическая модель квантового компьютера.

Предметом исследования выступают методы моделирования работы квантового компьютера.

Целью диссертационной работы является разработка алгоритмов для моделирования квантового компьютера, разработка программного обеспечения на основе этих алгоритмов и создание языка описания квантовых алгоритмов.
Для достижения этой цели в диссертации необходимо решить ***следующие задачи***:

1. Разработать методы представления состояний квантовых регистров.
2. Разработать алгоритмы обработки информации о состоянии квантового регистра.
3. Разработать способ моделирования квантовых вычислений.
4. Разработать библиотеку функций, для моделирования операций над квантовыми регистрами.
5. Разработать язык описания квантовых алгоритмов.
6. Разработать интерпретатор языка описания квантовых алгоритмов.

Методы исследования. Для решения поставленных задач в работе использовались методы дискретной математики, теории алгоритмов, функционального программирования.

Научная новизна.

- Разработан новый способ представления матриц и векторов. Особенностью способа является то, что матрицы и вектора представляются в виде графов специального вида – алгебраических решающих диаграмм. Разработаны

новые алгоритмы для действий над матрицами и векторами, представленными в виде алгебраических решающих диаграмм.

- Разработан новый способ имитации квантовых вычислений. Особенностью этого способа является использование представления состояния квантового регистра в виде алгебраической решающей диаграммы. Этот способ позволяет получить большую производительность по сравнению с существующими способами в ряде задач.
- Разработан новый язык для описания квантовых алгоритмов и интерпретатор для него.

На защиту выносится:

- Алгоритмы для действий над матрицами и векторами, представленными в виде алгебраических решающих диаграмм.
- Способ имитации квантовых вычислений, основанный на использовании алгебраических решающих диаграмм.
- Язык для описания квантовых алгоритмов.
- Программное обеспечение для имитации квантового компьютера.

Практическая значимость. Разработана библиотека функций для имитации квантового компьютера. На основе этой библиотеки разработан интерпретатор языка описания квантовых алгоритмов. Разработанное программное обеспечение может быть использовано как в качестве инструмента для отладки квантовых алгоритмов, так и в качестве средства для обучения студентов основам квантовых вычислений.

Апробация и внедрение результатов работы. Результаты диссертационной работы внедрены в Ракетно-космической корпорации «Энергия» им. С.П. Королева, а также в учебный процесс кафедры информационной безопасности МГТУ им. Н.Э. Баумана.

Основные результаты работы докладывались и обсуждались на ряде научных конференций, проводимых в МГТУ им. Н.Э. Баумана, а также на научных семинарах кафедры информационной безопасности МГТУ им. Н.Э. Баумана.

Публикации по теме диссертации. Основные результаты данной работы опубликованы в восьми научных трудах. В том числе две статьи опубликованы в изданиях, рекомендованных ВАК.

Структура и объем диссертации. Диссертация состоит из введения, четырех глав, заключения, библиографического списка и приложения. Основная часть диссертации изложена на 124 страницах текста. Библиографический список состоит из 98 наименований (из них 76 – на иностранных языках).

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснована актуальность проблемы моделирования квантового компьютера, сформулированы цель и задачи исследования.

В первой главе диссертации выполнен обзор литературы по теме диссертации. Рассмотрен математический аппарат теории квантовых вычислений. Рассмотрены основные понятия теории квантовых вычислений, такие, как квантовый бит (кубит) и квантовый регистр. В частности, состояние квантового регистра длины *n* представляет собой вектор в 2^n-мерном унитарном пространстве и записывается в виде линейной комбинации базисных состояний: $|\psi\rangle = \sum_{x=0}^{2^n-1} a_x |x\rangle$, где $a_x \in \mathbb{C}$. При этом, выполняется условие: $\sum_{i=0}^{2^n-1} |a_i|^2 = 1$.

Для получения информации о состоянии квантового регистра, его необходимо измерить. Важными свойствами квантовых систем является влияние процесса измерения на состояние квантовой системы и вероятностный характер процесса измерения. В случае если проводится измерение состояния всего регистра, результатом измерения будет являться один из базисных векторов *x* с вероятностью $|a_x|^2$ и после измерения регистр перейдет (часто используется термин сколлапсирует) в состояние $|x\rangle$, соответствующее результату измерения. В случае измерения *i*-го разряда, результат *u* будет выбираться из множества $\{0;1\}$ с вероятностью $p_u = \sum_{x:x_i=u} |a_x|^2$. После измерения, регистр перейдет в состояние $\frac{1}{\sqrt{\sum_{x:x_i=u} |a_x|^2}} \sum_{x:x_i=u} a_x |x\rangle$. Здесь запись x_i означает *i*-ый разряд квантового регистра. Измерение набора из *n* разрядов соответствует последовательному измерению разрядов из этого набора, при этом результат вычисления является элементом множества $\{0;1\}^n$.

Квантовый компьютер может осуществлять преобразования над квантовым регистром: $|\psi_2\rangle = U|\psi_1\rangle$. При этом оператор *U* является унитарным. С квантовыми системами можно производить только линейные унитарные преобразования, причем любое линейное унитарное преобразование допустимо. В силу линейности, квантовые преобразования полностью определяются их действием на базисные векторы. Существуют различные базисы квантовых преобразований, с помощью которых можно представить любое квантовое преобразование. Квантовый алгоритм представляет собой конечную последовательность квантовых преобразований над квантовым регистром и измерений квантового регистра.

Рассмотрен ряд квантовых алгоритмов. Так, рассмотрен алгоритм Гровера, который позволяет найти число, для которого выполняется заданный предикат. Кроме того, рассмотрены алгоритмы Шора для факторизации натуральных

чисел и для вычисления дискретного логарифма в мультипликативной группе вычетов по простому модулю. Оба алгоритма Шора выполняются за полиномиальное время, в отличие от лучших классических алгоритмов, известных на настоящее время, которые имеют сверхполиномиальную сложность.

Рассматриваются способы описания квантовых алгоритмов, такие как: квантовые схемы, квантовые машины Тьюринга, языки квантового программирования. Приводятся некоторые факты из теории сложности квантовых алгоритмов.

Рассматривается вопрос организации квантового компьютера. В частности, рядом авторов отмечается, что работа квантового компьютера должна осуществляется под управлением классического управляющего устройства, которым может являться классический компьютер.

Рассмотрено применение существующих квантовых алгоритмов в целях криптоанализа. Делается вывод о том, что появление действующих образцов квантовых компьютеров приведет к тому, что многие криптосистемы, прежде всего асимметричные, перестанут быть стойкими. Кратко изложены некоторые факты из квантовой криптографии.

Рассмотрено существующее программное обеспечение для моделирования квантовых вычислений. Такое ПО можно классифицировать следующим образом:

По объекту моделирования:

1. Программное обеспечение, моделирующее поведение квантовой системы, путем решения уравнения Шредингера. Такие программы называют эмуляторами.
2. Программное обеспечение, моделирующее абстрактную математическую модель квантового компьютера. Такие программы называют имитаторами (simulators).

Делается вывод о том, что в связи с тем, что необходимо иметь возможность запуска квантовых алгоритмов, имеет большое значение создание имитатора квантового компьютера. Создание эмулятора признается нецелесообразным, по той причине, что пока неизвестно, каким именно образом будет реализован квантовый компьютер.

Наиболее популярны три способа описания квантового алгоритма:

1. Описание в виде квантовой схемы.
2. Описание в виде квантовой машины Тьюринга.
3. Описание в виде программы на некотором, специализированном языке программирования.

Делается вывод о том, что наиболее перспективным способом описания квантовых алгоритмов является использование языка квантового программирования.

Существует тенденция по созданию универсальных языков программирования, объединяющих в себе как квантовую, так и классическую части. По-видимому, такая тенденция не является правильной в связи с тем, что поскольку квантовый компьютер должен управляться классическим компьютером, необходимо разделять квантовую часть алгоритма и классическую. Классическую часть алгоритма логично реализовывать на традиционном языке программирования, таком как, например, C++, C# или Java, тем более что для этих языков существует большое количество средств, предназначенных для обработки, хранения и визуализации данных. В связи с этим, язык квантового программирования должен быть предназначен для написания процедур вызываемых из программы, работающей на классическом компьютере.

Для разработки новых квантовых алгоритмов, а также для целей обучения, представляется необходимой разработка имитатора квантового компьютера, то есть программного обеспечения, позволяющего выполнять квантовые алгоритмы на обычном компьютере. Скорость работы существующих имитаторов очень мала, а требования к оперативной памяти – велики. Это позволяет использовать такие имитаторы, лишь для моделирования квантовых регистров, длина которых не превышает 20 – 32 кубитов. В связи с этим, можно поставить задачу разработки более быстродействующих алгоритмов моделирования квантового компьютера.

Для успешной имитации квантовых алгоритмов, следует располагать способом описания таких алгоритмов. Таким образом, встает вопрос о разработке языка квантового программирования.

Во второй главе разрабатывается способ машинного представления квантовых регистров и способ имитации квантового компьютера. Рассмотрены различные способы представления квантовых регистров, такие как использование линейного массива и использование связного списка. Сделан вывод о том, что эти способы требуют большого объема памяти.

Предложен способ машинного представления квантовых регистров, основанный на алгебраических решающих диаграммах (АРД). Алгебраическая решающая диаграммы (Algebraic Decision Diagram) представляет собой обобщение двоичной решающей диаграммы (Binary Decision Diagram) на случай произвольного множества значений терминальных вершин. Способ основан на использовании алгебраических решающих диаграмм для хранения векторов состояний квантовых регистров, а также для хранения матриц квантовых преобразований. Использование АРД позволяет значительно уменьшить затраты памяти в случае кодирования матриц и векторов с повторяющимися значениями, за счет использования свойства сокращенности. И в то же время для производства операций над закодированными таким образом матрицами и векторами не требуется декодирование.

АРД предназначены для представления функций вида $f:\{0;1\}^n \to S$, где S – произвольное множество (рассматривается случай линейно упорядоченного множества S). Векторы можно рассматривать в качестве таких функций и, сле-

довательно, представлять их в виде АРД. Матрицы также можно рассматривать в качестве таких функций. При этом на любой цепи дерева несокращенной АРД, представляющей матрицу, вершины, соответствующие номерам строк и номерам столбцов матрицы чередуются.

Для умножения матриц, представленных в виде АРД, разработан алгоритм, основанный на следующем. Если v – вершина, имеющаяся либо в АРД матрицы A, либо, матрицы B, то в случае, если переменная вершины v входит в состав номера столбца матрицы A (и, следовательно, номера строки матрицы B), выполняется:

$$A_v B_v = A_{high(v)} B_{high(v)} + A_{low(v)} B_{low(v)},$$

где запись A_v означает матрицу, задаваемую АРД с корневой вершиной v в АРД матрицы A.

В обратном случае (т.е. в случае если v входит в состав номера строки матрицы A или номера столбца матрицы B), выполняется:

$$A_v B_v = high(v) A_{high(v)} B_{high(v)} + low(v) A_{low(v)} B_{low(v)}$$

В случае если A_v и B_v – константы, произведение $A_v B_v$ может быть вычислено непосредственно.

С помощью изложенного способа становится возможным осуществлять умножение матриц, представленных с помощью алгебраических решающих диаграмм, без декодирования. На основе этого алгоритма умножения матриц, разработан алгоритм для умножения матрицы, представленной в виде АРД на вектор, представленный в виде АРД. На входе этот алгоритм получает вершину АРД матрицы (обозначим ее A), вершину АРД вектора (обозначим его x) и номер переменной этой вершины – k. Будем обозначать младшего потомка вершины x как low(x), старшего – как high(x), номер переменной, соответствующей вершине x – как var(x).

Если A и x – терминалы, то результат – терминал, равный Ax.
Если var(x) = k, то
    x_0=low(x)
    x_1=high(x)
    p_x=0.
Иначе p_x=1
Если var(A)=2x, то
    A_0=low (A)
    A_1=high(A)
    p_a=0
    Если var(A_0)=2k–1, то
        A_{00}=low(A_0)
        A_{01}=high (A_0)
        p_{a0}=0
    иначе

p_{a0}=1
Если var(A_1)=2k–1, то
A_{10}=low(A_1)
A_{11}=high (A_1)
p_{a1}=0
иначе p_{a1}=1
Иначе
p_a=1

Результат выбирается из табл. 1 (если выходом является АРД с двумя потомками, они приводится через точку с запятой; иначе - приводится одно значение). Умножения матрицы на вектор, приведенные в этой таблице вычисляются по тому же алгоритму, при этом на его вход подается в качестве номера переменной (k–1).

Результатом является АРД, представляющая вектор, равный искомому произведению.

Предложен способ для вычисления кронекеровых произведений матриц, представленных в виде АРД. Пусть надо вычислить кронекерово произведение $A \otimes B$. Способ состоит в замене каждой терминальной вершины АРД матрицы А, на АРД матрицы $B \cdot t$, где t – значение, сохраненное в заменяемой терминальной вершине.

Таблица 1.

К алгоритму умножения матрицы на вектор

p_a	p_{a0}	p_{a1}	p_x	
			0	1
0	0	0	$A_{00}x_0+A_{01}x_1$; $A_{10}x_0+A_{11}x_1$	$(A_{00}+A_{01})x$; $A_{10}x_0+A_{11}x_1$
0	0	1	$A_{00}x_0+A_{01}x_1$; $A_1(x_0+x_1)$;	$(A_{00}+A_{01})x$; $2A_1x$
0	1	0	$A_0(x_0+x_1)$; $A_{10}x_0+A_{11}x_1$	$2A_0x$; $A_{10}x_0+A_{11}x_1$
0	1	1	$A_0(x_0+x_1)$; $A_1(x_0+x_1)$;	$2A_0x$; $2A_1x$
1			$A_0(x_0+x_1)$;	$2Ax$;

Исследуется вопрос о строении матрицы преобразования, переставляющего разряды с номерами i и j у квантового регистра длины n. Это преобразование можно записать следующим образом:

$$P_{n,i,j} = I_2^{[i]} \otimes P_{(j-i+1),0,(j-i)} \otimes I_2^{[n-j-1]},$$

где $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\otimes$ – кронекерово произведение матриц, $A^{[n]} = \underbrace{A \otimes A \otimes \ldots \otimes A}_{n \text{ раз}}$.

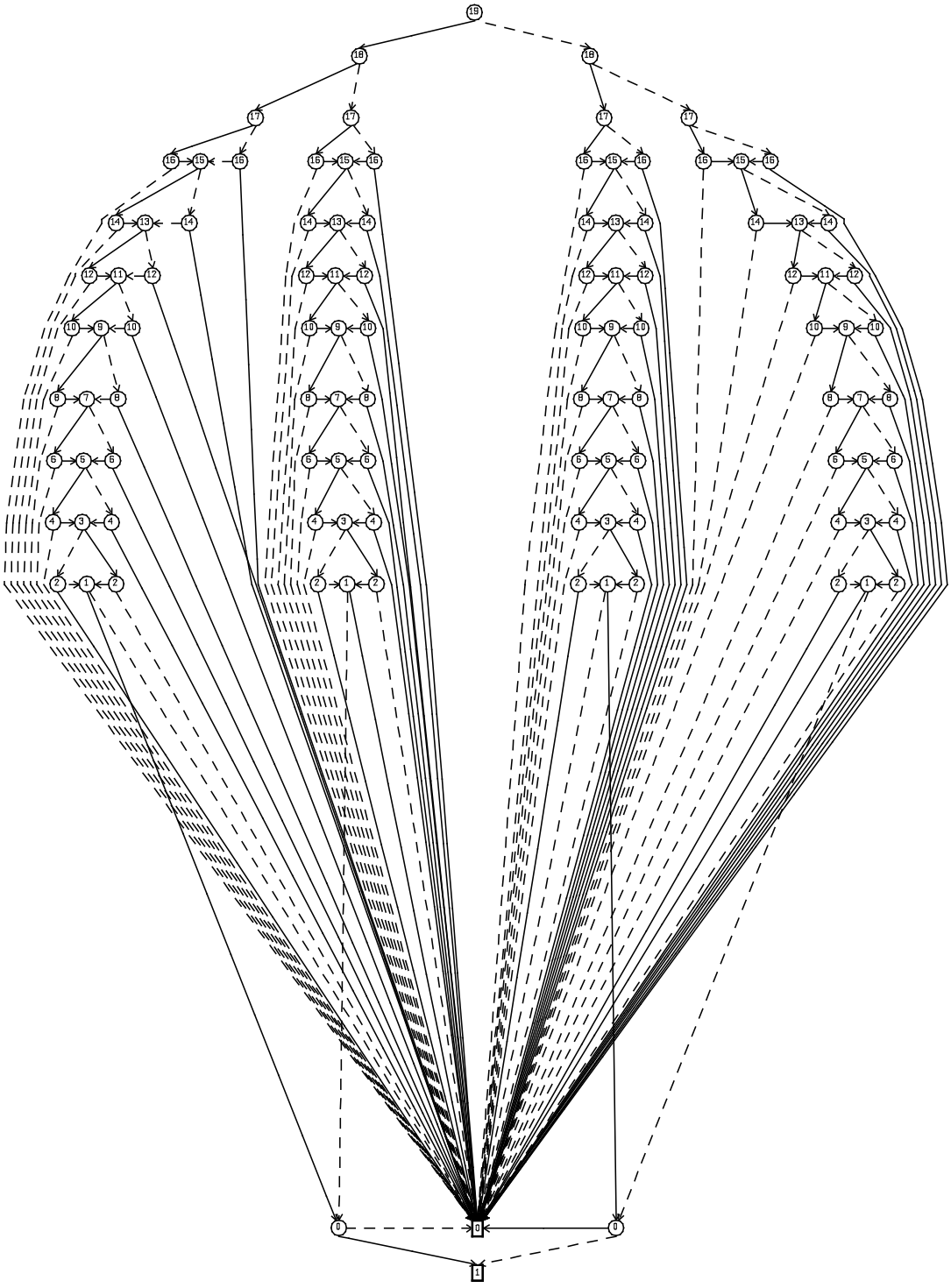

Рис. 1. Алгебраическая решающая диаграмма для матрицы $P_{10,0,9}$ (размера 1024×1024)

Пунктирными линиями обозначены ребра, помеченные 0, сплошными линиями – 1. Кругами (в кругах – номера переменных) обозначены нетерминальные вершины, квадратами (в квадратах – значения) – терминальные.

Сформулированы и доказаны теорема о структуре матриц $P_{(j-i+1),0,(j-i)}$ и теорема о структуре АРД, с помощью которой можно представить матрицу $P_{(j-i+1),0,(j-i)}$. В качестве примера приведем АРД для матрицы $P_{10,0,9}$ (размера 1024×1024) (рис. 1). Эти теоремы позволили разработать алгоритм для построения АРД, соответствующей таким матрицам.

В связи с тем, что размер матрицы преобразования должен быть равен $2^n \times 2^n$, где n – число разрядов моделируемого регистра, перед применением преобразования к регистру следует эту матрицу сформировать, на основе исходной матрицы, с помощью тензорного произведения. После чего следует переставить разряды квантового регистра с помощью матриц $P_{n,i,j}$ а после применения преобразования – переставить их в обратном порядке.

Разрабатывается способ имитации квантовых вычислений. Выполнение квантового алгоритма состоит из следующих этапов: инициализация квантовых регистров; квантовые преобразования и измерения; последнее измерение; получение и обработка результатов управляющим классическим компьютером. В случае необходимости – квантовые вычисления повторяются.

Имитатор должен позволять выполнять эти действия так, чтобы результат был неотличим от того, который был бы получен на квантовом компьютере. В процессе выполнения квантового алгоритма должны производиться квантовые преобразования. Делается вывод, что их целесообразно последовательно применять к квантовому регистру.

Квантовые преобразования выполняются в соответствии с формулой

$$\left|y\right\rangle = P^{-1}(F \otimes I_2^{[n-k]})P\left|x\right\rangle,$$

где x – текущее состояние квантового регистра, длины n; y – новое значение регистра; F – матрица $2^k \times 2^k$ преобразования, действующего на k кубитах; P – матрица подстановки.

Рассмотрим теперь проведение измерений состояния квантовых регистров. Для проведения измерения одного разряда с номером k, требуется получить вероятности того, что значение этого разряда равно нулю $p\{x_k = 0\} = p_0$ и единице $p\{x_k = 1\} = p_1 = 1 - p_0$. При этом $p\{x_k = 0\} = \sum_{x:x_k=0} |a_x|^2$. После случайного выбора результата измерения u, требуется перевести регистр в состояние $\left|\psi\right\rangle = \sum_{x:x_k=u} a_x \left|x\right\rangle$, что достигается использованием алгоритма Restrict с параметрами (v,k,u), где v – корневая вершина АРД, с последующей нормировкой. Алгоритм Restrict – известный из теории решающих диаграмм алгоритм, который преобразует решающую диаграмму с корневой вершиной v, путем приравнивания переменной u к значению k. После выполнения алгоритма Restrict выполняется нормировка вектора состояния, путем вычисления нормы вектора $\left|\psi\right\rangle$ и

деления значений всех терминальных вершин АРД на $\sqrt{\||\psi\rangle|}$. Измерение набора кубитов сводится к последовательному измерению каждого кубита из набора.

Таким образом, был разработан способ машинного представления информации о квантовых состояниях, основанный на алгебраических решающих диаграммах, который позволяет хранить информацию о квантовом состоянии в более компактной форме по сравнению с существующими алгоритмами. Также, предложены алгоритмы для имитации выполнения любых допустимых квантовых преобразований и измерения квантовых регистров. В ходе выполнения этих задач была развита теория алгебраических решающих диаграмм в части способов представления матриц и векторов, а также алгоритмов для работы с матрицами и векторами.

В третьей главе диссертации, на основе результатов второй главы, реализована библиотека функций для работы с квантовыми регистрами на языке функционального программирования Haskell. Эта библиотека функций реализована на основе библиотеки функций для работы с алгебраическими решающими диаграммами, которая также разработана в этой главе. Исходные тексты разработанных библиотек приведены в приложении.

В библиотеке функций для работы с алгебраическими решающими диаграммами для описания типа вершин используется тип, сочетающий в себе два типа вершин:

1. Нетерминальная вершина. Такую вершину можно представить в виде тройки (v,l,h), где $v,l,h \in \mathbb{Z}^+$: v – номер переменной, соответствующей вершине, l – номер дочерней вершины, соответствующей нулевому значению переменной, h – номер вершины, соответствующей единичному значению переменной
2. Терминальная вершина. В такой вершине хранится значение.

Алгебраическая решающая диаграмма хранится в структуре данных, состоящей из двух таблиц:

W – таблица, реализующая функцию, отображающую вершины u в тройки (i,l,h), для нетерминальных вершин; и в значения x, для терминальных вершин.

H – таблица, реализующая функцию, отображающую тройки (i,l,h), для нетерминальных вершин и значения x – для терминальных вершин, в u.

Для эффективной реализации этих таблиц требуется, чтобы множество возможных значений было линейно упорядоченным.

Таким образом, алгебраическая решающая диаграмма хранится в виде структуры данных, содержащей три элемента: таблицу W (представленную с помощью типа Sequence), таблицу H (представленную с помощью типа Map) и номера корневой вершины, заключенного в монаду Maybe. Типы Map и Sequence входят в стандартную библиотеку языка Haskell и реализованы в ней с помощью сбалансированных бинарных деревьев.

Реализующий АРД тип ADD является параметризованным. При этом тип хранимых значений может быть любым типом, являющемся линейно упорядоченным (т.е. реализующим класс Ord), для значений которого реализуется операция сравнения (т.е. тип реализует класс Eq).

В этой библиотеке реализованы функции, необходимые для работы с алгебраическими решающими диаграммами, а также с векторами и матрицами, представленными с помощью алгебраических решающих диаграмм. В частности, реализованы функции для создания АРД, инициированной вектором, а также матрицей; для умножения матриц, умножения матрицы на вектор; кронекерова произведения матриц; тензорного произведения векторов; сложения матриц; сложения векторов; функция, реализующая алгоритм restrict, функция реализующая вычисление функции от двух АРД), а также ряд других функций.

Разработана библиотека функций для имитации квантового компьютера. Это библиотека функций основана на представлении вектора состояния квантовых регистров и матриц квантовых преобразований в виде АРД. Элементами этих векторов и матриц являются комплексные числа, представленные в экспоненциальном виде, причем в качестве отношения порядка используется лексикографическое отношение пар (амплитуда, фаза).

Реализованы типы данных для представления квантового регистра и квантового преобразования. Реализованы функции для создания и инициализации квантовых регистров и квантовых преобразований, функции для имитации применения квантовых преобразований к квантовому регистру, функция для вычисления классической функции от квантового регистра, а также функции для проведения измерений состояния квантового регистра, с помощью которых можно проводить измерения любого набора кубитов из регистра.

В четвертой главе диссертации разработан язык представления квантовых алгоритмов и интерпретатор этого языка.

Основными объектами, с которыми оперирует разработанный язык, являются: классическая переменная, представляющая целое неотрицательное число, квантовый регистр, преобразование.

Основными операциями, которые поддерживает разработанный язык, являются следующие: измерение состояния квантового регистра, линейное преобразование, тензорное произведение квантовых регистров, вычисление классической функции от квантового регистра.

В программы, написанные на разработанном языке, должны передаваться параметры, являющиеся классическими переменными, а из программ должны возвращаться результаты вычислений, являющиеся также классическими переменными.

Программы на разрабатываемом языке генерируются с помощью программ на каком-либо классическом языке программирования. Аналогичный подход используется, например, в задачах доступа к базам данных.

Разработанный язык имеет операторы для:

- определения входной переменной;

- определения квантового преобразования;
- определения квантового регистра;
- вычисления тензорного произведения квантовых регистров;
- применения квантового преобразования к квантовому регистру;
- последовательного применения одного и того же квантового преобразования к различным разрядам квантового регистра;
- измерения состояния набора разрядов квантового регистра;
- вычисления классической функции от квантового регистра;
- передачи результатов вычислений в управляющую программу.

В языке реализованы стандартные квантовые преобразования: отрицание, фазовый сдвиг, фазовый сдвиг с отрицанием, Controlled-NOT, вентиль Тофолли, преобразование Адамара. Кроме того, для увеличения производительности на уровне языка реализованы такие преобразования, как преобразование Фурье и инверсия относительно среднего. Имеется возможность задавать любые другие квантовые преобразования.

Интерпретатор реализован на языке Haskell с помощью генератора парсеров Happy и лексического анализатора Alex.

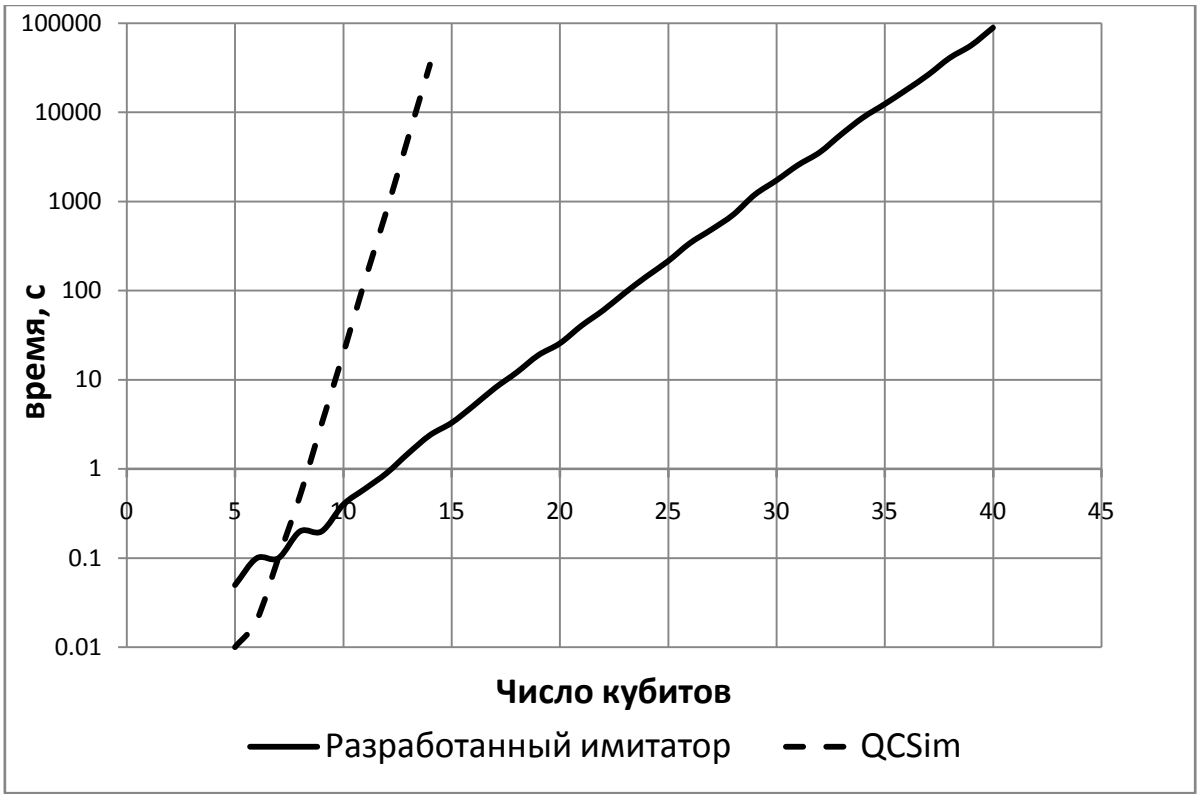

Рис. 2. Время выполнения имитации одной итерации алгоритма Гровера

Кроме того, в этой главе приводятся замеры производительности разработанного интерпретатора, при моделировании алгоритма Гровера и квантового преобразования Фурье. Его производительность для случая алгоритма

Гровера сравнивается с производительностью имитатор QCSim, разработанного в NIST (Национальный институт стандартов, США). В соответствии с этими замерами, производительность разработанного интерпретатора (рис. 2) существенно выше, чем производительность имитатора квантовых вычислений QCSim, который использует массивы для хранения информации о квантовых состояниях. Требования к памяти разработанного имитатора – существенно ниже, чем для QCSim (рис. 3) и являются линейными относительно длины моделируемого квантового регистра.

Квантовое преобразование Фурье, при достаточно большой длине квантового регистра, также имитируется с более высокой скоростью, чем с помощью QCSim (рис. 4) и при этом, требует меньше памяти (рис. 5).

Разработанный интерпретатор и является имитатором квантового компьютера.

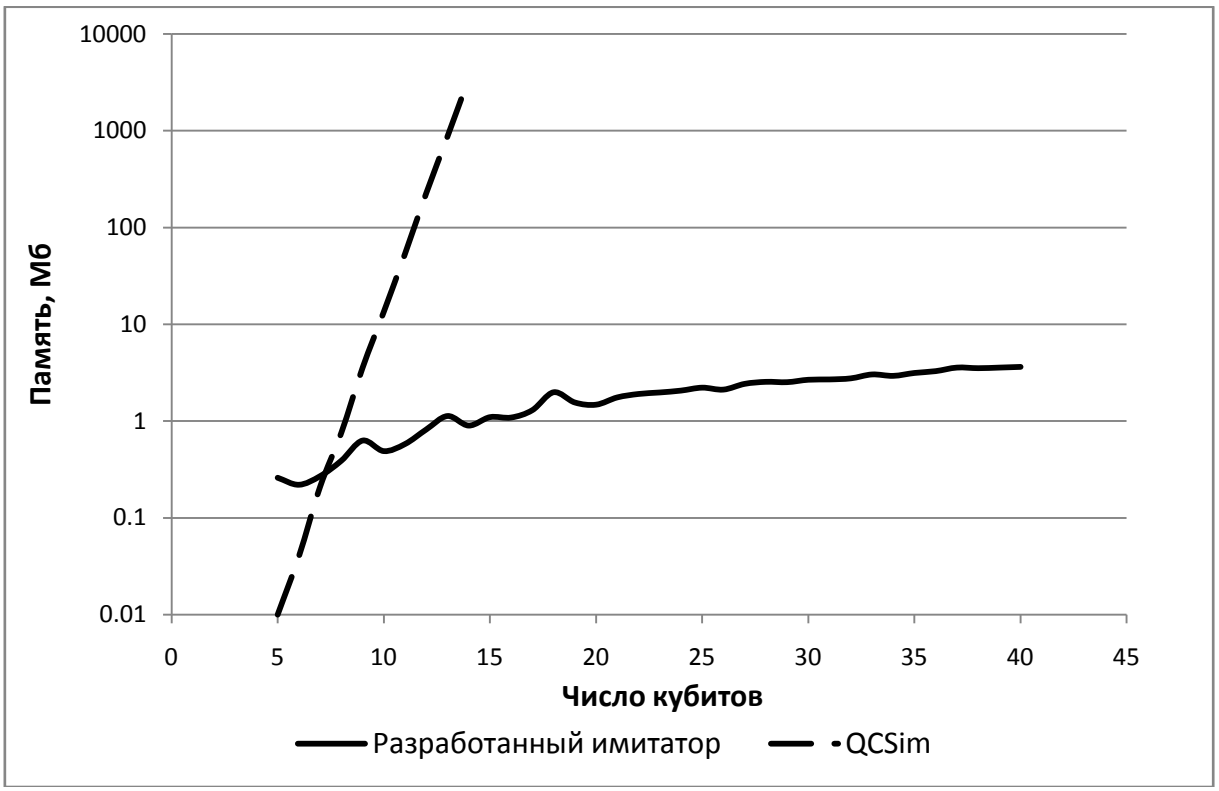

Рис. 3. Требования к памяти, при имитации одной итерации алгоритма Гровера

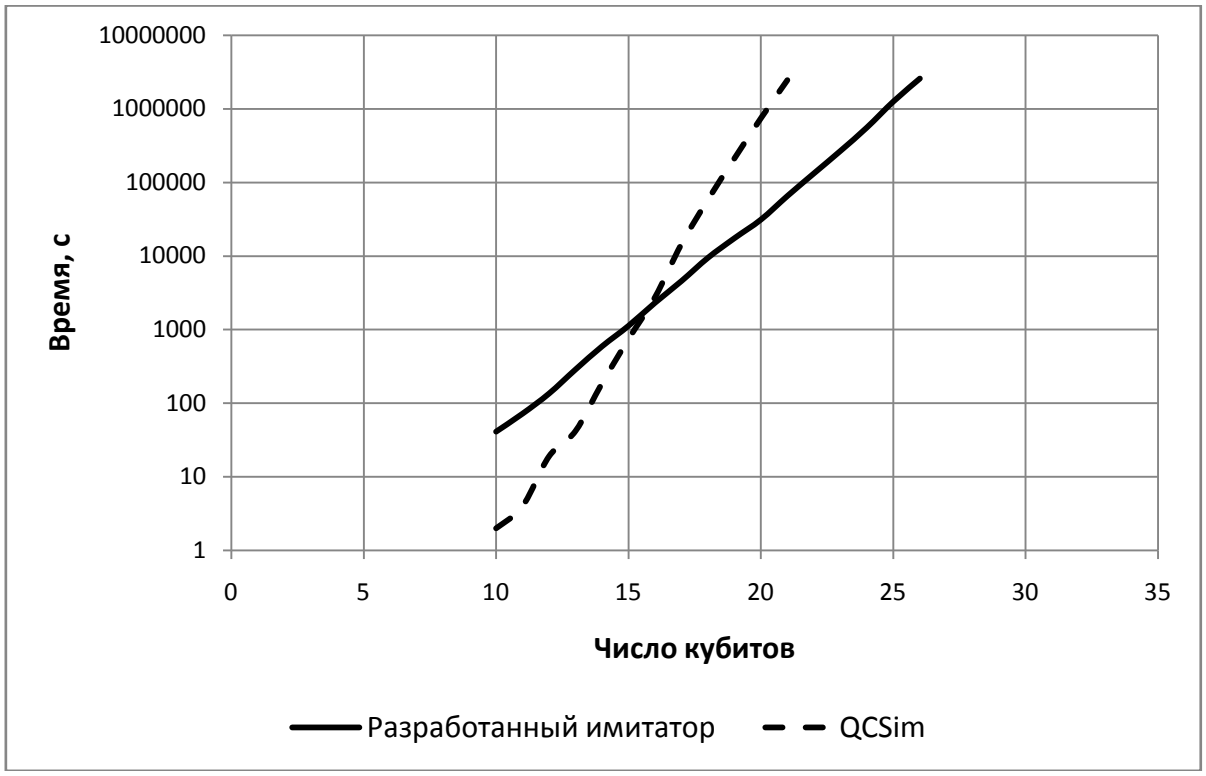

Рис. 4. Время выполнения имитации квантового преобразования Фурье

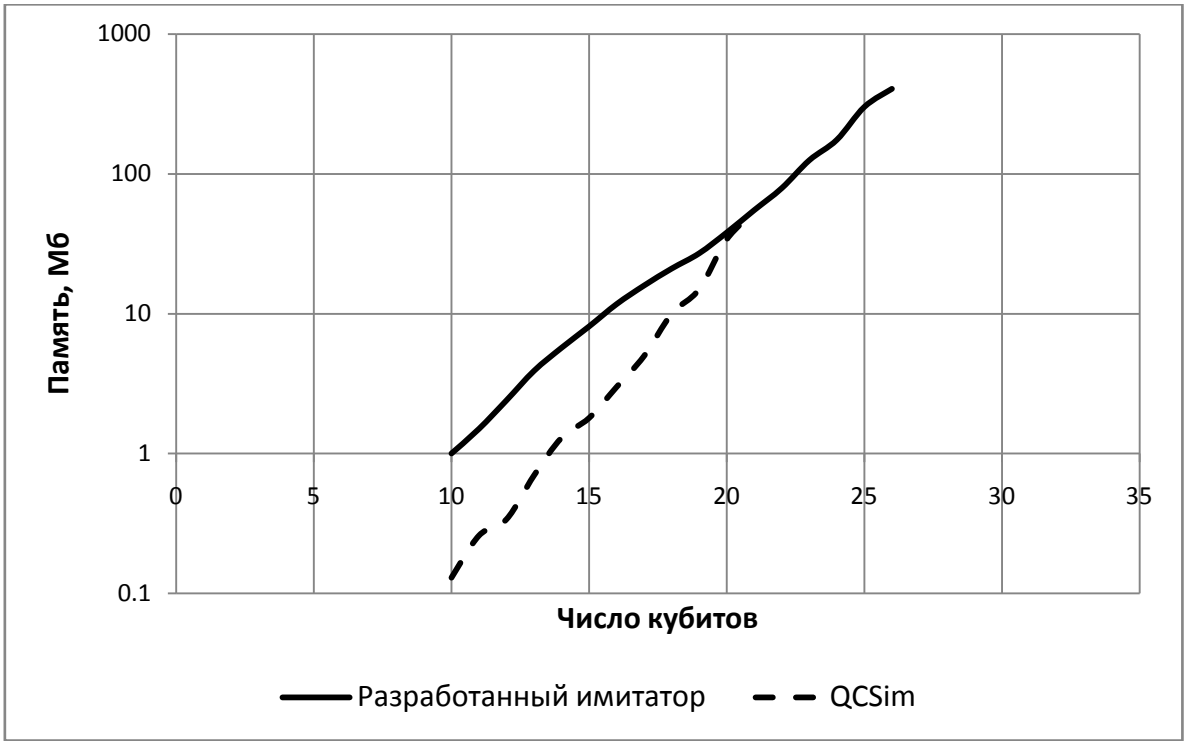

Рис. 5. Требования к памяти для имитации квантового преобразования Фурье

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Разработаны алгоритмы для хранения векторов и матриц в виде алгебраических решающих диаграмм и алгоритмы для действий над ними.
2. Разработаны алгоритмы для имитации квантового компьютера.
3. Разработана библиотека функций, на языке функционального программирования Haskell, для работы с алгебраическими решающими диаграммами, а также с векторами и матрицами, хранимыми в виде алгебраических решающих диаграмм
4. Разработана библиотека функций для имитации квантовых вычислений.
5. Разработан язык для представления квантовых алгоритмов и интерпретатор этого языка.

Результаты настоящей работы будут способствовать разработке новых квантовых алгоритмов.

РАБОТЫ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Ключарев П.Г. Алгебраический подход к диагностике отказов вычислительной системы специального назначения // Компьюлог. – 2002. – №2. – С. 41–43.
2. Ключарев П.Г. Информационные и телекоммуникационные системы, их воздействие на общественное и индивидуальное сознание // Студенческая научная весна – 2000: Тез. докл. научно-технической конференции. - М.: МГТУ им. Н.Э. Баумана, 2001. - С. 57-58.
3. Ключарев П.Г. Квантовый компьютер и криптографическая стойкость современных систем шифрования // Вестник МГТУ им. Н.Э. Баумана. Серия Естественные науки. – 2007. – №2. – С. 113-120.
4. Ключарев П.Г. Криптоаналитические возможности квантового компьютера // Прикаспийский журнал: управление и высокие технологии. – 2008. – №2. – С. 7-13.
5. Ключарев П.Г. Основы квантовых вычислений и квантовой криптографии // Вестник МГТУ им. Н.Э. Баумана. Серия Приборостроение. – 2006. – №2. – С. 36-46.
6. Ключарев П.Г. Программная реализация двоичных регистров сдвига с обратной связью // Студенческая научная весна: Тез. докл. научно-технической конференции. - М.: МГТУ им. Н.Э. Баумана, 2002 -С. 125-126.
7. Ключарев П.Г. Свобода и безопасность личности в условиях информационного общества // Тезисы Второго Международного конгресса Молодежь и наука – третье тысячелетие, 2001. - С. 125–126.
8. Ключарев П.Г. Социокультурная компетентность как одно из условий обеспечения информационной безопасности в Российской Федерации // Реформы в России и мире: компаративный анализ: Тез. докл. научно-технической конференции. - М.: МГТУ им Н.Э. Баумана, 2000. - С. 63-65.