КАЗАНСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

КВАНТОВЫЕ ВЫЧИСЛЕНИЯ

Учебное пособие

Казанский федеральный университет
2010

Оглавление

Глава 1

Введение

1.1 Введение

Компьютер родился как машина для счета. До него люди использовали для той же цели арифмометры, еще раньше – логарифмическую линейку, счеты, камешки, палочки, в конце концов – собственные пальцы. Менялись материальные носители, усложнялись конструкции, появлялись новые математические действия. Но неизменными оставались логические принципы, лежащие в основе всякого расчета. Никакой предмет не мог находиться одновременно в двух местах, или быть одновременно больше или меньше некоторой величины. Так строились расчеты, потому что так был устроен мир. В начале XX столетия люди столкнулись с другим миром – квантовым. Его объекты могли одновременно проходить через два отверстия, быть здесь или там и даже существовать лишь отчасти. Но только сегодня человек разобрался в логике странного мира настолько, чтобы попытаться создать основанные на ней счетно-логические устройства.

У истоков идеи квантового компьютера стоял один из самых ярких среди американских физиков-теоретиков послевоенного времени – Ричард Фейнман. В 70-е годы Фейнман занимался проблемой компьютерного моделирования квантово-механических процессов на вычислительных машинах. Он заметил, что с

ростом размерности физической задачи вычислительная сложность увеличивается непропорционально быстро. Строго говоря, вычислительная сложность возрастает экспоненциально. По мнению Фейнмана, проблема заключалась в том, что всякое вычисление – это физический процесс. Даже деление одного числа на другое столбиком карандашом на бумаге выполняется в полном соответствии с физическими законами взаимодействия грифеля с волокнами бумаги. В более сложном случае, когда идет речь об электронно-вычислительном устройстве, функционирование ячеек памяти или регистров процессора происходит в соответствии с законами классической электродинамики. Принципиальной сложностью для такого классического устройства оказывается та самая особенность квантовой механики, которая больше всего возмущает врагов "идеалистической зауми". А именно, настоящее в этой науке определяется не только действительно наблюдавшимся прошлым, но и тем прошлым, которое могло произойти, но не произошло или, по крайней мере, вероятно не произошло. Например, если самолет летит из Петербурга в Москву, то узнать, в каком состоянии прилетит самолет, можно, опираясь на знание его действительной траектории полета. Если бы самолет был квантовым объектом, то, чтобы описать его квантовое состояние в Москве, надо было бы принять во внимание все возможные траектории его движения, включая даже те, на которых немного и ненадолго нарушаются законы сохранения. Чтобы рассчитать движение такого "квантового самолета" на классическом компьютере, придется вычислить все его классические траектории, а потом их определенным образом просуммировать. Даже незначительное усложнение размеров исходной задачи очевидным образом приведет к значительному усложнению ее решения. Но если бы компьютер был устроен так, что он сам по себе, в силу своей природы учитывал бы для каждого отдельного самолета все варианты его прошлого - не только

действительного, но и возможного, - такого бы не происходило. Решение задачи с ростом размерности, естественно, тоже усложнялось бы, но не экспоненциально быстро.

1.2 История зарождения квантовых вычислений

Принцип работы классического компьютера был описан задолго до его появления. Можно утверждать, что Алан Тьюринг (1912 - 1954) – отец информатики, а пророк ее – Чарльз Бэббидж (1791 – 1871). Бэббидж определил большинство существенных элементов современного компьютера, хотя в то время еще не существовало технологий для осуществления его идей. Прошло столетие, и Тьюринг усовершенствовал "аналитическую машину" Бэббиджа, предложив в середине 30-х годов универсальную Машину Тьюринга – полную модель классического вычисления. Гений Тьюринга заключается в том, что он точно определил возможности вычислительной машины и в еще большей степени, чем Бэббидж, подчеркивал роль программирования или, другими словами, программного обеспечения. По словам Фейнмана, Тьюринг "исчерпал законы карандаша и бумаги".

Современные компьютеры не являются ни машинами Тьюринга, ни машинами Бэббиджа, хоть и имеют с ними много общего. Но при этом их вычислительная мощность эквивалентна (в техническом смысле) мощности машины Тьюринга. Весь процесс развития затрагивает лишь размеры и быстродействие, но не касается основных принципов структуры или работы компьютера. Однако квантовая механика ставит вопрос о возможности подобных изменений.

Квантовая механика – это математическая структура, которая в принципе охватывает всю физику. В поведении всех систем, даже тех, которые называются классическими, лежит квантовая механика. ("Даже отвертка является квантово-механичес-

кой системой" – Landauer, 1995). Квантовая механика является наиболее точной теорией об окружающем мире, известной человечеству. Модели вычислений, использующие в основе законы квантовой физики, поэтому, возможно, являются наиболее разумными.

Начиная с изобретения электронных ламп вплоть до разработки сверхбольших интегральных схем, непрестанно происходило усовершенствование вычислительных машин. Сейчас ключевым фактором увеличения быстродействия вычислительных машин является уменьшение размеров транзисторов в используемых современных процессорах. Современные технологии в электронной промышленности являются микронными технологиями. Микрон – одна тысячная миллиметра (1mc= 10^{-3} mm). Эпоха микронных технологий заканчивается. Наступает время нанотехнологий.

Термин "нанотехнология" применяют к технологиям, позволяющим оперировать объектами размерами сто нанометров и менее. Нанометр – одна миллионная миллиметра (1nm = 10^{-6} mm). Один из последних процессоров (процессор Pentium 4) фирмы Intel разработан на основе микронной технологии 0.13 mc или (в терминах нанотехнологий) на основе нанотехнологии 130 nm. Весной 2004 года Intel объявила о создании процессора Prescott, выполненного по технологии 90 nm. В нем около 10 миллиардов транзисторов располагаются на одном квадратном сантиметре электронного чипа. Тактовая частота процессора 3.4 GHz. Эксперты в области электронных нанотехнологий прогнозируют, что к 2020 производители электронных систем подойдут к уровню технологий 10 nm и менее. То есть к 20-м годам на основе кремниевых соединений будут сконструированы транзисторы минимально возможных размеров. Это уменьшение, однако, не может продолжаться бесконечно. Если размер транзистора становится слишком маленьким, сопоставимым с

размерами атома, включается эффект так называемого “квантового туннелирования”, когда электроны начинают свободно проходить между полюсами транзисторов. Означает ли это, что данные эффекты, по-видимому, представляют принципиальные ограничения для существующих в настоящий момент компьютерных технологий? По-видимому, эра кремниевой микроэлектроники, бурно начавшаяся во второй половине XX века, близится к своему завершению. В ближайшем будущем на смену нанотехнологиям неизбежно придут биотехнологии, оперирующие молекулами ДНК, и квантовые технологии, оперирующие ионами атомов и элементарными частицами. Квантовые технологии, готовящиеся прийти на смену классическим технологиям, будут использовать законы квантовой механики. Физика квантового мира такова, что квантовый компьютер, построенный на основе квантовых частиц и функционирующий в соответствии с законами квантовой механики, будет способен реализовывать вычисления, недоступные в реальное время для классических компьютеров. Это будет возможно, в частности, благодаря эффекту “квантового параллелизма”. Необходимо отметить, что результаты вычислений квантовых компьютеров будут проявляться в макромире (в нашем мире) вероятностно.

Связь между квантовой механикой и теорией информации возникла после осознания того факта, что такие простые свойства квантовых систем, как неизбежное их нарушение при измерениях, могут использоваться в практике квантовой криптографии. Квантовая криптография объединяет несколько идей, из которых неизменной является идея квантовой передачи кода. Это несложный метод, где передаваемые квантовые состояния используются для выполнения совершенно особенной задачи связи: установить в двух отстоящих друг от друга точках пару идентичных, но, с другой стороны, случайных последовательностей двоичных цифр так, чтобы они (последовательно-

сти) оставались неизвестными для третьего лица. Этот метод очень полезен, поскольку подобная случайная последовательность может использоваться как криптографический код для обеспечения защищенной связи. Важной особенностью здесь является то, что принципы квантовой механики обеспечивают такой вид сохранения квантовой информации, когда при ее поступлении к сторонам, желающим установить случайный код, эти лица могут быть уверены, что информация не поступила к кому-либо еще (например, к шпиону).

В то время, пока анализировалась и демонстрировалась квантовая криптография, на свет незаметно появился квантовый компьютер. Первые идеи рассмотрения вычислений с квантово-механической точки зрения заключались в преобразовании работы Машины Тьюринга к эквивалентному обратимому процессу. Эти идеи появились в работе Беннетта (Bennett, 1973), где он показал, что универсальная классическая вычислительная машина (такая, как Машина Тьюринга) может быть обратимой без потери своей простоты. Бениофф и др. (Benioff, 1980, 1982) предложили подобные Машины Тьюринга в начале 80-х годов. Хотя их идеи не позволяли провести полный анализ квантового вычисления, они показывали, что унитарная квантовая эволюция по крайней мере обладает такой же вычислительной мощностью, что и классический компьютер.

В начале 1990-х годов несколько авторов занимались поиском задач, чье решение с помощью квантового компьютера было бы более эффективно, чем их решение посредством обычного классического компьютера. Первоначально были обнаружены лишь небольшие различия в работе алгоритмов: в случае, когда квантовая механика, при условии, что на квантовую систему не действуют помехи, могла дать определенный ответ, вероятностный классический компьютер мог получить тот же ответ с высокой вероятностью. В 1994 году Саймон сделал важное открытие,

описав эффективный квантовый алгоритм для решения (в некотором смысле абстрактной) задачи, не имеющей эффективных классических решений даже при использовании вероятностных методов. Это открытие вдохновило Шора, и он поразил общественность, когда в 1994 году описал алгоритм, который был не только эффективен при его реализации на квантовом компьютере, но также был нацелен на решение фундаментальной задачи информатики: разложению на множители простых целых чисел. Шор, применяя метод квантового преобразования Фурье, описал как разложение на множители, так и задачу нахождения дискретного логарифма.

Задача факторизации имеет важную, но специфическую область применения – создание систем шифрования с открытым ключом. Отметим, что современные системы шифрования с открытым ключом (RSA системы) являются основой систем "электронных подписей". Эти системы основаны на задачах, требующих для их решения больших временных затрат классических компьютеров (задача факторизации – пример такой задачи). Алгоритмы дешифрования известны, но для расшифровки (подделывания подписи) нужно потратить сотни недель работы классических компьютеров. Создание квантового компьютера и реализация на нем алгоритма Шора будет означать возможность быстрой расшифровки сообщения, основанного на задаче факторизации.

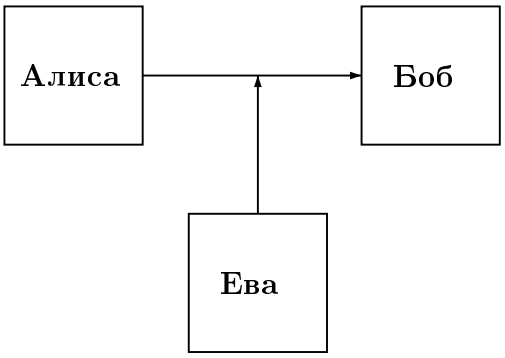
Существующий только на бумаге квантовый компьютер не способен выполнить какие-либо практические вычисления. В конце концов, разрешить спорный вопрос относительно осуществимости построения квантового компьютера можно только путем создания самого квантового компьютера. В настоящий момент ведутся активные исследования в этой области различными группами многих стран (квантовые точки, ионные ловушки, метод ядерного магнитного резонанса и т. д.). На сегодняшний

день не создан еще ни один квантовый компьютер, если рассматривать его с точки зрения алгоритма Шора и требовать от него разложения на множители больших чисел. Однако, если вместо вышеуказанного требуется устройство для проверки идей квантовой теории информации, то для этих целей необходимо устройство, работающее на нескольких квантовых битах. И такие реализации на сегодняшний день уже есть.

1.3 Области применения квантовой информатики

Среди огромного множества задач, для решения которых применение квантового компьютера может оказаться принципиальным, одна представляется наиболее важной. Речь идет о проблеме безопасности в передаче информации. Вряд ли надо объяснять, сколь остро эта проблема стоит на сегодняшний день. Утечка информации приводит к разного рода накладкам – от пропажи денег с электронных карт до террористических актов. Сегодня телефонные разговоры шифруются почти всегда, при этом совершенно незаметно для тех, кто говорит. Всякая передача информации предполагает наличие ключа. В большей части алгоритмов используется так называемый симметричный ключ. Алиса использует ключ для кодировки исходного сообщения. Боб расшифровывает его при помощи того же самого ключа. Но сообщение может прочитать и некто посторонний (будем называть его Ева), если догадается, какой используется ключ, или попросту его украдет.

Для того чтобы шифр был устойчив по отношению к криптоанализу, длина ключа должна быть не меньше длины сообщения. При этом каждый ключ желательно использовать не более одного раза. Тут-то и возникает одна из главных проблем криптографии. “Одноразовый блокнот”, то есть достаточно длинная коллекция достаточно длинных ключей однократного



использования, конечно, гарантирует конфиденциальность сообщений. Но как Алисе и Бобу запастись одинаковыми одноразовыми блокнотами? Ответом на этот вопрос стал принцип шифрования с открытым ключом, предложенный в 1976 году Уитфилдом Диффи и Мартином Хеллманом. Практическое его применение предполагает открытую пересылку некоторого числа, представляющего собой произведение двух очень больших простых чисел. Если бы Еве удалось опознать, что это за числа, она бы без труда расшифровала сообщение Алисы. Но с ростом количества знаков объем операций, необходимых для разложения его на множители, растет по экспоненциальному закону. Компания RSA регулярно предлагает всем желающим подобные задачи. Одно из предложенных ею 155-значных чисел удалось разложить, затратив 36 процессорных лет. Если только у Евы нет гигантских вычислительных ресурсов, к тому времени, когда она расшифрует сообщение, оно уже не будет представлять никакого интереса.

Надо сказать, что операции с огромными числами представляют определенную трудность и для самих участников переписки. Поэтому шифрование с открытым ключом часто применяется как дополнительное средство. Например, Алиса и Боб могут обменяться при его помощи одноразовым блокнотом и дальше пользоваться малозатратными симметричными ключами, надеясь, что конфиденциальность блокнота гарантирована. Однако в 1994 коллега Диффи по Массачусетскому университету Питер

Шор показал, что при разложении N-значного числа на множители число операций квантового компьютера пропорционально примерно N в кубе (за свою работу Питер Шор был удостоен в 1998 году высшей награды в теоретической информатике - премии Неванлинны, присуждаемой раз в 4 года одновременно с медалью Филдса по чистой математике). То есть для квантового компьютера задача разложения на множители того же класса сложности, что и простое умножение.

Но прежде чем вооруженные алгоритмом Шора квантовые компьютеры начнут угрожать конфиденциальности переписки, на помощь Алисе и Бобу придут методы квантовой криптографии. Тогда при перехвате пересылаемый блокнот окажется уничтоженным, и им просто не удастся воспользоваться.

Квантовая криптография. Фактически на рынке уже несколько лет имеются устройства для создания квантовых линий связи. Формально можно считать, что эти устройства выполняют квантовое вычисление, хотя и очень специфическое, оперируя одним кубитом. Это лишь первые шаги технологии, но уже они обеспечивают совершенно новый тип защищенности линий связи. Квантовый канал используется для передачи не самих сообщений, а только секретного криптографического ключа (общего секретного ключа Алисы и Боба), обычно длиной 28 или 256 битов. Когда ключ сгенерирован, стороны переходят на обычный способ связи, обмениваясь сообщениями, зашифрованными этим ключом (обновляемым тысячи раз в секунду). Шифры, использующие секретные ключи, считаются очень надежными, и им, в отличие от шифров, основанных на трудности задачи факторизации, не угрожает даже квантовый компьютер. Основная трудность при использовании таких шифров – необходимость заранее обменяться достаточным количеством секретных ключей. А развозить абонентам защищенных сетей

связи диски, набитые секретными ключами – дело крайне дорогое, хлопотное и даже опасное. Именно эту проблему призваны решить современные квантовые коммуникации. Для краткости такой тип связи называют квантовой криптографией, хотя, по существу, речь идет лишь о квантовой передаче ключей для обычного "крипто". Пионеры коммерческого использвания квантовой криптографии на общедоступном рынке – компании ID Quantique (Швейцария) и MagiO Technologies (США) – используют примерно одинаковые технологии. Так, MagiO Technologies применяет протокол BB84, разработанный Чарльзом Бенетом и известным криптографом Жилем Брассаром. Секретный ключ представляется в виде потока битов – определенным образом поляризованных единичных фотонов (нулю и единице соответствуют различные поляризации). Любое подслушивание неминуемо (согласно законам квантовой механики) внесет изменения в структуру принимаемого на другом конце потока, и факт прослушивания можно будет установить по окончании работы протокола. Протокол же требует после пересылки криптофотонов обменяться еще кое-какой несекретной информацией по обычному открытому каналу. Только тогда у обеих сторон появляется общий секретный ключ для надежного шифрования.

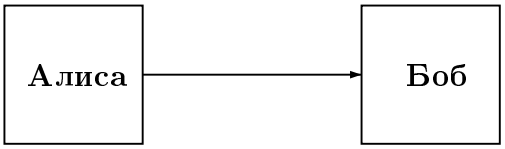
Вторая волна протоколов квантовой связи использует иную физику, гораздо ближе к той, что требуется для квантовых вычислений. Речь идет о так называемых запутанных (entagled) состояниях нескольких частиц – важнейшем механизме квантовых вычислений и одновременно самом загадочном феномене квантовой физики. Если несколько квантовых частиц составляют единую квантовую систему, то теоретически они могут быть разнесены хоть на световые годы, не теряя своего квантового единства. А это означает, что любое воздействие на одну из них автоматически меняет состояние другой точно так же, как если бы она была совсем рядом. Подольский и Розен в своей знамени-

той работе в 1935 году рассматривали мысленный эксперимент с такими частицами как парадокс, дающий основание для серьезной концептуальной критики квантовой механики. В конечном счете эта критика стимулировала глубокий анализ и дальнейшее развитие квантовой теории.

Сегодня запутанные состояния (ЭПР-состояния) – не просто экспериментально подтвержденный, а промышленно используемый феномен. Запутанные фотоны рассылаются по квантовым линиям связи (как оптоволоконным, так и воздушным, то есть при помощи лазера прямо через атмосферу) на десятки километров (каждый из собеседников получает по одному фотону, которые вместе образуют ЭПР-пару, единую квантовую систему). Первый протокол (E91) квантовой передачи ключей при помощи ЭПР-пар фотонов был разработан в 1991 году учеником Дэвида Дойча (ныне директором Центра квантовых вычислений Кембриджского университета) Артуром Экертом. В 2004 году по шестой рамочной программе Евросоюза был запущен крупный международный проект SECOQS разработки глобальной сети для защищенной связи на основе квантовой криптографии.

Телепортация. В Гарвардском университете ведутся следующие эксперименты. Фотон света движется между узлами в некоторой среде. В этих узлах он испытывает поглощение. Важно, что спустя некоторое время он переизлучается вновь, оказавшись в том же самом состоянии. При этом можно считать, что это тот же самый фотон, ведь с физической точки зрения переизлученный фотон неотличим от поглощенного. Среда, в которой распространяется свет, может быть моделью квантовой вычислительной сети, если ее атомы, поглощая фотоны, могут потом излучить их в том же квантовом состоянии, в каком поглотили. Фотон, поглощенный атомом, перестает существовать,

однако атом сохраняет информацию о его состоянии. Когда атом переходит из возбужденного состояния в основное, эта информация “считывается”, и рождается новый фотон, неотличимый от старого. Если бы знать в точности исходное квантовое состояние, то исходный фотон можно было бы скопировать, создав его неотличимого двойника. Но такое “клонирование” запрещают законы квантовой механики: измерить состояние фотона можно, только уничтожив сам фотон.



Алиса может воспользоваться этим свойством света, чтобы предохранить пересылаемую информацию от кражи. После проведения Алисой измерения, разрушившего ее фотон, она пересылает результат Бобу (по обычному, не квантовому каналу). И тут выясняется, что Боб может “навязать” измеренное Алисой состояние первому попавшемуся кванту света, который сразу после этого превратится в точную копию уничтоженного Алисой фотона. Такую операцию называют квантовой телепортацией: исчезнувшая из-за манипуляций Алисы частица возрождается у Боба. Необходимое условие такой операции – наличие у Алисы и Боба еще двух фотонов, находящихся, по меткому определению одного из главных творцов квантовой механики Вольфганга Паули, в “запутанных состояниях”. Строго говоря, запутанное состояние неприменимо ни к одному из них в отдельности, можно говорить только о состоянии пары. Дальность телепортации, таком образом, ограничивается тем расстоянием, на которые могут быть разведены два “запутанных” фотона.

Говоря о квантовых компьютерах, как правило, думают об алгоритме Шора, хотя в квантовой информатике есть приложения менее глобальные, но не менее полезные. Например, кван-

товая криптография. Эта вещь по существу уже работает. Но и в ней есть сложность, связанная с ограничением на расстояние. В классических системах есть так называемые репиторы, которые позволяют почистить и усилить сигнал. А настоящие квантовые криптографические системы, построенные на передаче одного фотона, фундаментально ограничены короткими расстояниями. Но оказывается, можно построить и квантовый репитор, который может это расстояние сильно увеличить. И для построения этого квантового репитора нужен маленький квантовый компьютер с памятью для фотонов. И тут даже двух кубит может хватить. Точнее, по два кубита на каждые двадцать километров. В этом смысле квантовый компьютер, пусть даже и минимальный, – это принципиально новый способ использования природы. Благодаря ему абстрактная теория станет частью повседневной реальности – хотя по-прежнему останется недоступной воображению.

1.4 Физические эксперименты по созданию квантового компьютера

В 2001 году группой Айзека Чуаня была произведена экспериментальная реализация алгоритма Шора. В этом эксперименте при помощи квантового компьютера было проведено разложение числа 15 на множители 3 и 5. “Квантовый компьютер” состоял из единственной молекулы, синтезированной специально для этого случая. Кубиты были реализованы ядрами входивших в состав этой молекулы пяти атомов изотопа фтора и двух атомов изотопа углерода. Ноль и единица представлялись одним из возможных направлений спина этих ядер. Вычисления (использовавшие ансамбль независимых идентичных молекул) выполнялись манипуляциями кубитов при помощи примерно трехсот радиочастотных импульсов, а финальное измерение – при помо-

щи ЯМР-спектроскопии (широко применяемой в медицине для проведения томографии) .

Аналогичная система была произведена в 2006 году (в совместной работе Института теоретической физики в Ватерлоо, Канада, и Массачусетского технологического института в Кембридже, США) для того, чтобы получить 12 кубитов в "управляемом" запутанном состоянии.

Также идут исследования по другим методам создания кубитов. Перечислим некоторые из них, которые считаются наиболее практичными:

- Ионные ловушки в толще полупроводника – с их помощью уже получены запутанные состояния десятка ионов меди и магния.
- Квантовые точки, они же искусственные атомы – макрообъекты, состоящие из примерно миллиона атомов в полупроводнике, внутри которых удерживается один или несколько электронов. Кубитом может быть пара квантовых точек, значения 0 и 1 соответствуют нахождению электронов в той или иной из них.
- Цепочки ядерных спинов – структуры в полупроводнике размером в несколько нанометров. В узкий канал в кремнии имплантируется цепочка ионов фосфора, содержащая от десяти тысяч до миллиона атомов - она образует один кубит.
- π-контакт – кубит создается на границе высокотемпературных сверхпроводников разных типов; потенциальная энергия такого перехода имеет два локальных минимума, соответствующих базовым состояниям кубита.

В России исследования по всем этим направлениям ведут специалисты Физико-технологического института РАН (ФТИАН) во

главе с академиком РАН Камилем Ахметовичем Валиевым. Из российских теоретиков нельзя не упомянуть Александра Холево (автора выполненных еще в 1970-е годы пионерских работ по квантовой теории информации), Юрия Манина (первым в мире поставил в 1980 году проблему исследования вычислительного потенциала квантовых автоматов), Алексея Китаева (ему принадлежит революционная идея "топологического квантового вычисления"). Однако, несмотря на успехи многочисленных исследовательских групп, до сих пор никто не может ответить на главный вопрос: какая же технология окажется масштабируемой, то есть позволит легко наращивать число кубитов. За словом "масштабируемость" могут скрываться фундаментальные проблемы.

Процесс вычисления на будущем квантовом компьютере можно условно описать следующим образом. Сначала кубиты, кодирующие задачу, приводят в состояние, в котором они образуют единую квантовую систему. Согласно квантовой механике, это состояние содержит одновременно все возможные начальные условия. Унитарные преобразования воздействуют одновременно на все эти начальные условия (квантовый параллелизм), в некотором смысле двигаясь в поисках ответа сразу по всем возможным путям. В тех задачах, для которых известны квантовые алгоритмы, удалось так подобрать эти преобразования, что приближение к ответу происходит очень быстро. К сожалению, таких ситуаций в математике и информатике буквально единицы. Однако в физике есть важнейший класс задач, которые прекрасно решаются такими средствами – это моделирование квантовых систем. Квантовый компьютер идеально приспособлен для решения таких задач, в то время, как для классического они запредельно трудны. Так, для моделирования поведения электрона в трехмерном пространстве надо взять хотя бы сто точек по каждому измерению. Это дает уже миллион узлов сет-

ки. А если в системе два электрона - потребуется миллион миллионов узлов. Даже для суперкомпьютера это уже сложно. А ведь два электрона - это всего лишь атом гелия, и то без учета движения ядра.

Глава 2

Основные определения, понятия и свойства

2.1 Используемые математические понятия

Пусть $\mathcal{H}^d$ – d-мерное гильбертово пространство с нормой $||.||_2$. Для $z \in \mathcal{H}^d$, $z = \{z_1, \ldots, z_d\}$, $||z|| = \sqrt{\sum_{i=1}^{d} |z_i|^2}$. Расстояние между двумя элементами $z, z' \in \mathcal{H}^d$ определяется как $\rho(z, z') = ||z - z'||$. В квантовой механике принято использовать обозначения Дирака $\langle\,|$ и $|\,\rangle$. Дирак ввел свое обозначение на заре зарождения квантовой механики как удобное средство манипулирования векторами. $|\,\rangle$ – это просто обозначение вектора-столбца. Можно также использовать привычное векторное обозначение $|0\rangle = (1, 0)^T$. $\langle\,|$ обозначает вектор-строку.

Элемент ψ пространства $\mathcal{H}^d$ будем обозначать $|\psi\rangle$, где $|\,\rangle$ обозначает вектор-столбец (кет-вектор). Для вектора $|\psi\rangle = (z_1, \ldots, z_d) \in \mathcal{H}^d$ через $\langle\psi|$ обозначается вектор-строка (бра-вектор), такая что $\langle\psi| = (z_1^*, \ldots, z_d^*)$. При этом выполняется следующее. Если $|\phi\rangle = U|\psi\rangle$, то $\langle\phi| = \langle\psi|U^\dagger$.

Операторы. Каждый линейный оператор U, действующий в пространстве $\mathcal{H}^d$ с базисом $|i\rangle$, $i = 1, \ldots, d$, может быть представлен $d \times d$ матрицей (будем также обозначать ее через U), строки и столбцы которой помечены $i, i = 1, \ldots, d$, и на пересечении i-й строки и j-го столбца стоит элемент $\langle i|U|j\rangle$.

Определение 2.1 *Оператор U называется унитарным, если $UU^\dagger = U^\dagger U = I$ (I – единичная матрица).*

Свойство 2.1 *Унитарный оператор U обладает свойством сохранять норму и расстояние:*

$$|| \, |\psi\rangle|| = ||U|\psi\rangle||,$$
$$\rho(|\psi\rangle, |\phi\rangle) = \rho(U|\psi\rangle, U|\phi\rangle)$$

.

Оператор проекции. Пусть $\mathcal{H}^d = W_1 \oplus \cdots \oplus W_k$, $k \leq d$ – ортогональная декомпозиция пространства $\mathcal{H}^d$ на подпространства $W_1, \ldots, W_k$. Тогда произвольный элемент $|\psi\rangle \in \mathcal{H}^d$ может быть однозначно представлен как линейная суперпозиция проекций $|\psi\rangle$ на подпространства $W_1, \ldots, W_k$ следующим образом:

$$|\psi\rangle = |\psi_{W_1}\rangle + \cdots + |\psi_{W_k}\rangle,$$

где $|\psi_{W_1}\rangle \in W_1, \ldots, |\psi_{W_k}\rangle \in W_k$. В этом случае преобразования $P_{W_1}(|\psi\rangle) = |\psi_{W_1}\rangle, \ldots, P_{W_k}(|\psi\rangle) = |\psi_{W_k}\rangle$ называются *операторами проекции* на подпространства $W_1, \ldots, W_k$, соответственно.

Определение 2.2 *Тензорное (правое кронекерово) произведение векторов a, b – это вектор $a \otimes b$, определенный следующим образом. Для $a = (a_1, \ldots, a_d)$ и $b = (b_1, \ldots, b_l)$: $a \otimes b = (a_1 b_1,\ a_1 b_2,\ \ldots,\ a_1 b_l,\ a_2 b_1,\ \ldots,\ a_d b_l)$.*

Определение 2.3 *Тензорное (правое кронекерово) произведение матриц A, B – матрица $A \otimes B$, определенная следующим образом.*

$$\textit{Для } A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \textit{ и } B = \begin{pmatrix} b_{11} & \cdots & b_{1k} \\ \vdots & \ddots & \vdots \\ b_{l1} & \cdots & b_{lk} \end{pmatrix}$$

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

2.2 Квантовый бит

В классических компьютерах фундаментальной единицей информации является классический бит, который может принимать значение 0 или 1. Для представления набора таких битов компьютер содержит набор соответствующих физических систем, каждая из которых может находиться в одном из двух взаимно различимых физических состояний, ассоциируемых со значениями 0 и 1 того абстрактного бита, который данная система представляет. Такой физической системой может быть, например, магнит, чья намагниченность может быть ориентирована в двух различных направлениях – вверх (0) или вниз (1).

Аналогично классическому компьютеру, который оперирует последовательностью классических битов, квантовый компьютер оперирует последовательностью квантовых битов. Таким образом, фундаментальной единицей информации в квантовых вычислениях является квантовый бит – *кубит*. Для представления кубита может использоваться некоторая квантовая система с двумя устойчивыми состояниями, например спин электрона $1/2$ в магнитном поле. Квантовое состояние спина есть непрерывная величина, определяемая двумя вещественными числами, и, в принципе, способная хранить бесконечное количество классической информации. Однако измерение спина дает лишь одно число, способное принимать два значения (спин вверх, спин вниз), и таким образом, не существует какого-либо доступа к тому бесконечному объему информации, который может храниться данным квантовым состоянием. Мы будем обозначать состояние каждого кубита символом $|\,\rangle$, внутри которого будем помещать значение 0 или 1, представляемое данным состоянием.

Таким образом, в отличие от классического бита, который в каждый момент времени может находиться в одном из двух со-

стояний – в состоянии 0 или в состоянии 1, квантовый бит, или "кубит" q, кроме двух устойчивых состояний ($|0\rangle$ и $|1\rangle$, соответственно), может находиться одновременно в суперпозиции этих двух состояний:

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle,$$

где α, β – комплексные числа, $|\alpha|^2 + |\beta|^2 = 1$. Таким образом, квантовый бит может принимать бесконечно много значений, но, как результат измерения, мы получим либо состояние $|0\rangle$ с вероятностью $|\alpha|^2$, либо состояние $|1\rangle$ с вероятностью $|\beta|^2$ (рис. 2.1).

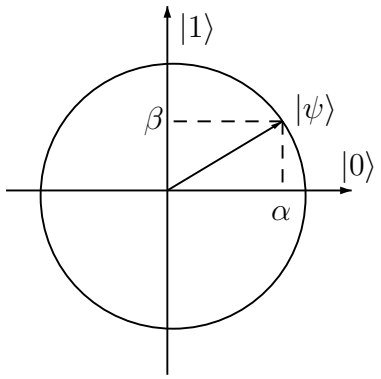


Рис. 2.1: Состояние кубита – единичный вектор в двумерном комплексном пространстве в базисе $\{|0\rangle, |1\rangle\}$

2.3 Квантовая система

Состояние квантовой системы из n кубитов $q_1, \ldots, q_n$ описывается как

$$|q_1\rangle \ldots |q_n\rangle = (\alpha_1|0\rangle + \beta_1|1\rangle) \otimes \cdots \otimes (\alpha_n|0\rangle + \beta_n|1\rangle) =$$

$$z_0(|0\rangle \ldots |0\rangle|0\rangle) + z_1(|0\rangle \ldots |0\rangle|1\rangle) + \cdots + z_{2^n-1}(|1\rangle \ldots |1\rangle|1\rangle),$$

где $z_i (i = 1, \ldots, 2^n)$ – комплексные числа, $\sum_{i=1}^{2^n} |z_i|^2 = 1$. То есть n кубитов могут одновременно находиться в суперпозиции 2^n

состояний, каждое из которых описывает одно из возможных *устойчивых состояний* n-кубитной системы. Эти устойчивые состояния системы следующие: состояние системы, когда значения всех n кубитов равны нулю (это состояние в суперпозиции представлено с амплитудой z_0), состояние, когда значения всех кубитов, кроме последнего, равны нулю, последний n-й кубит равен единице (это состояние в суперпозиции представлено с амплитудой z_1) и так далее. Последнее возможное устойчивое состояние – значения всех кубитов равны единице (амплитуда данного состояния в суперпозиции равна z_{2^n-1}).

Эти устойчивые состояния системы

$$\{|0\rangle|0\rangle\dots|0\rangle, |0\rangle|0\rangle\dots|1\rangle, \dots |1\rangle|1\rangle\dots|1\rangle\}$$

мы также будем кратко обозначать как $\{|00\dots0\rangle, |00\dots1\rangle, \dots |11\dots1\rangle\}$, или даже $\{|0\rangle, |1\rangle, \dots |2^n-1\rangle\}$.

Каждому такому устойчивому состоянию удобно сопоставить базисный вектор из ортонормированного базиса в 2^n-мерном гильбертовом пространстве $\mathcal{H}^{2^n}$ (будем называть его "стандартным базисом"). Тогда состояние $|\psi\rangle = |q_1\dots q_n\rangle$ n-кубитной квантовой системы представляется как вектор единичной длины в гильбертовом пространстве $\mathcal{H}^{2^n}$ в базисе $\{|0\rangle, \dots |2^n-1\rangle\}$. То есть

$$|\psi\rangle = \sum_{i=0}^{2^n-1} z_i|i\rangle$$

или коротко $|\psi\rangle = (z_0, \dots, z_{2_n-1})^T$. При этом выполняется

$$||\,|\psi\rangle||_2 = \sqrt{\sum_{i=0}^{2^n-1} |z_i|^2} = 1.$$

Пространство $\mathcal{H}^{2^n}$ является тензорным произведением n двумерных гильбертовых пространств $\mathcal{H}^2$, каждое из которых есть пространство состояний каждого из n кубитов квантовой системы.

Всюду далее будем называть $|\psi\rangle$ *конфигурацией* или *суперпозицией квантовых состояний*. Комплексное число z_i называется *амплитудой* базисного состояния $|i\rangle$ квантовой системы. При этом, если производится измерение, $|z_i|^2$ дает вероятность обнаружения квантовой системы QS в состоянии $|i\rangle$.

2.4 Преобразования квантовой системы

Все преобразования, осуществляемые квантовым компьютером, – обратимые. Существует только один тип необратимых преобразований – измерение, и это есть лишь извлечение полезной информации из кубита, после того как он достиг своей финальной формы. В обратимых преобразованиях каждое финальное состояние получается из единственного начального состояния. К примеру, классическая операция ERASE, которая выдает состояние 0 независимо от того, какое состояние было начальное — 0 или 1. Эта операция необратима в том смысле, что мы не можем по выходу операции распознать, каково было начальное состояние. Существует только одна нетривиальная обратимая операция на одном классической бите – это операция NOT. Она обратима – применение операции к финальному состоянию бита восстанавливает исходное значение бита. Любое преобразование квантовой системы (за исключением измерения) является унитарным и, следовательно, обратимым преобразованием, и задается при помощи унитарной матрицы.

Определение 2.4 Эволюция квантовой системы *(изменение состояния QS за определенный промежуток времени) задается унитарным оператором и описывается следующим образом. Если $|\psi\rangle$ – конфигурация системы QS на текущем шаге, то на следующем шаге конфигурация QS будет $|\psi'\rangle$, где $|\psi'\rangle = U|\psi\rangle$ и U – $(d \times d)$ – унитарная матрица.*

Поскольку унитарное преобразование не меняет длины преобразуемого вектора, и длина вектора состояний $|\psi\rangle$ всегда равна единице, то любое квантовое преобразование состояния $|\psi\rangle$ можно трактовать просто как поворот вектора $|\psi\rangle$ на единичной сфере в пространстве $\mathcal{H}^{2^n}$.

2.5 Измерение квантовой системы

Если вы имеете n классических битов, каждый из которых представляет 0 или 1, вы можете узнать состояние каждого, просто посмотрев их значения. Нет никаких проблем узнать состояние классических битов и, следовательно, узнать результат вычисления, которое вы осуществляли манипуляциями на данных битах. Более того, состояние битов не изменится в результате чтения их содержимого. Совсем иначе обстоит дело, если вы имеете n кубитов в суперпозиции базисных состояний. Существует только один способ извлечь информацию – произвести *измерение*. Произвести измерение — значит осуществить некоторый тест над каждым из кубитов, результатом которого будет 0 или 1. Конкретный набор исходов 0 или 1, полученный в результате теста, вообще говоря, не определяется исходным состоянием кубитов. Исходное состояние определяет вероятности возможных исходов.

Таким образом, извлечение результата вычисления из квантовой системы производится посредством измерения системы. Измерение квантовой системы – это вероятностный процесс, который осуществляется следующим образом:

1. Пусть $|\psi\rangle = (z_0, z_1, \ldots, z_{2^n-1})$ – квантовое состояние и $B = \{|0\rangle, \ldots, |2^n - 1\rangle\}$ – стандартный базис пространства состояний квантовой системы из n кубитов (элементы базиса B соответствуют устойчивым состояниям квантовой системы).

2. Если производится измерение состояния $|\psi\rangle$ по отношению к стандартному базису B, то с вероятностью $|z_i|^2$ результатом измерения является устойчивое состояние $|i\rangle$.

3. После измерения состояние квантовой системы становится

$$|\psi'\rangle = |i\rangle.$$

То есть, когда производится измерение квантовой системы, то состояние, которое имело больший модуль амплитуды, выпадет в результате измерения с большей вероятностью. После измерения амплитуды всех остальных состояний становятся равными нулю, и вся остальная информация, содержащаяся в конфигурации, становится безвозвратно потерянной. Поскольку состояние квантовой системы всегда должно иметь норму, равную единице, результирующая конфигурация нормализуется до единичной длины.

Если производится измерение только некоторых кубитов квантовой системы, то здесь используются обычные правила из теории вероятностей. Например, пусть состояние квантовой системы

$$|\psi\rangle = z_0|00\rangle + z_1|01\rangle + z_2|10\rangle + z_3|11\rangle.$$

Подсчитаем вероятности и исходы измерения первого кубита системы. Вероятность p того, что результатом измерения будет значение $|0\rangle$, подсчитывается как сумма квадратов амплитуд состояний, в которых значение первого кубита равно нулю: $p = |z_0|^2 + |z_1|^2$. Состояние после измерения с исходом $|0\rangle$ равно $|\psi'\rangle = \frac{z_0|00\rangle + z_1|01\rangle}{\sqrt{|z_0|^2+|z_1|^2}}$.

Вероятность того, что результатом измерения первого кубита будет значение $|1\rangle$, равна $|z_2|^2+|z_3|^2$. Состояние после измерения с исходом $|1\rangle$ равно $|\psi'\rangle = \frac{z_2|10\rangle + z_3|11\rangle}{\sqrt{|z_2|^2+|z_3|^2}}$.

Если же мы измерим второй кубит, то вероятность того, что в результате измерения мы получим состояние $|0\rangle$, равна $|z_0|^2 +$

$|z_2|^2$. Состояние после измерения с данным исходом будет $|\psi'\rangle = \frac{z_0|10\rangle+z_2|11\rangle}{\sqrt{|z_0|^2+|z_2|^2}}$.

Если же при измерии второго кубита мы получим результат $|1\rangle$, то вероятность данного исхода будет $|z_1|^2 + |z_3|^2$. Состояние после измерения с данным исходом будет $|\psi'\rangle = \frac{z_1|10\rangle+z_3|11\rangle}{\sqrt{|z_1|^2+|z_3|^2}}$.

Пример. Пусть $|\psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$. Произведем измерение первого кубита. С вероятностью $\frac{1}{2}$ результатом измерения будет значение $|0\rangle$. Состояние после измерения станет $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$.

2.6 Квантовый регистр

Квантовый бит, или кубит, – это вектор в двумерном гильбертовом пространстве, для которого зафиксирован какой-то частичный базис, обозначаемый $|0\rangle, |1\rangle$. Ортогональные базисные состояния могут соответствовать поляризациям $|\uparrow\rangle, |\downarrow\rangle$, или поляризациям $|\nearrow\rangle, |\nwarrow\rangle$ фотона. Или $|0\rangle, |1\rangle$ могут соответствовать состояниям спин вверх и спин вниз электрона. В квантовых вычислениях базисные состояния используются для кодирования значения бита 0 или 1 соответственно. Однако, в отличие от классического бита, кубит может находиться в суперпозиции состояний $|0\rangle$ и $|1\rangle$, такой как

$$\alpha|0\rangle + \beta|1\rangle,$$

где α, β – комплексные числа, такие что $|\alpha|^2 + |\beta|^2 = 1$. Если суперпозиция будет измерена по отношению к базисным состояниям $|0\rangle, |1\rangle$, вероятность того, что измеренное значение будет $|0\rangle$, есть $|\alpha|^2$ и вероятность того, что измеренное значение будет $|1\rangle$, есть $|\beta|^2$. Когда говорят о кубите в квантовых вычислениях, то фиксированный базис, по отношению к которому дела-

ются все утверждения, выбирается заранее. В частности, до тех пор пока не будет особо оговорено, мы будем предполагать, что все измерения производятся в базисе $|0\rangle, |1\rangle$. Хотя кубит может находиться в суперпозиции двух своих различных состояний, возможно закодировать только одно классическое состояние в квантовом бите. С точки зрения теории информации, кубит содержит в точности то же самое количество информации, что и классический бит, несмотря на то что он может иметь бесконечно много различных состояний. Кубит не может содержать большее количество информации, потому что информация из кубита может быть извлечена только посредством измерения. Когда производится измерение кубита, состояние кубита (вектора в двумерном пространстве) сразу же изменяется на одно из базисных состояний, по отношению к которым производилось измерение. Следовательно, как и в классическом случае, для любого измерения кубита существует только два возможных результата. А так как измерение изменяет состояние кубита, невозможно сначала измерить кубит в одном базисе, а затем в другом.

Система из одного квантового бита описывается Гильбертовым пространством $\mathcal{H}^2$. Состояние такой системы (текущее значение квантового бита) описывается вектором нормы 1 в пространстве $\mathcal{H}^2$. Системе из двух квантовых битов соответствует четырехмерное Гильбертово пространство $\mathcal{H}^4$ с ортогональным базисом $\{|0\rangle|0\rangle, |0\rangle|1\rangle, |1\rangle|0\rangle, |1\rangle|1\rangle, \}$. Состояние двухкубитной системы описывается единичным вектором

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$$

таким, что

$$|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$$

Измерение двухкубитной системы выдает как исход измере-

ния один из результатов 00, 01, 10, 11 с вероятностями $|\alpha_{00}|^2$, $|\alpha_{01}|^2$, $|\alpha_{10}|^2, |\alpha_{11}|^2$, соответственно. С другой стороны, если мы выберем измерение только одного из кубитов, применяются стандартные правила вероятностей. Это означает, что измерение первого (соответственно, второго) кубита дает значение 0 или 1 с вероятностями $|\alpha_{00}|^2 + |\alpha_{01}|^2$ и $|\alpha_{10}|^2 + |\alpha_{11}|^2$ (соответственно $|\alpha_{00}|^2 + |\alpha_{10}|^2$ и $|\alpha_{01}|^2 + |\alpha_{11}|^2$).

Определение 2.5 *Квантовый регистр — это упорядоченное множество конечного числа кубитов.*

Определение 2.6 *Стандартный базис $\mathcal{B}$ n-кубитного квантового регистра — это*

$$\mathcal{B} = \{|i\rangle | i \text{ есть } n - \text{битная бинарная строка.}\}$$

Как мы видели, состояние кубита – это есть вектор в двумерном гильбертовом пространстве с базисом $|0\rangle, |1\rangle$. В классическом случае возможное состояние системы из n частиц (регистра из n битов) при условии, что состояние одной частицы описывается вектором в двумерном векторном пространстве, может быть описано вектором размерности 2n. Однако в квантовом случае микросистема из n кубитов формирует пространство состояний размерности 2^n. Это возрастание размерности пространства состояний предполагает возможное экспоненциальное ускорение вычислений на квантовых компьютерах по сравнению с классическими. Рассмотрим разницу между декартовым и тензорным произведением. Пусть V и W – двумерные гильбертовы пространства с базисами $\{v_1, v_2\}$ и $\{w_1, w_2\}$ соответственно.

Декартово произведение $V \times W$ этих двух пространств – это пространство, которое может использовать в качестве базиса объединение базисов исходных пространств – $\{v_1, v_2, w_1, w_2\}$. Отметим, что порядок базиса выбран произвольно. В частности, размерность пространства состояний нескольких классических

частиц возрастает линейно с ростом числа частиц, так как $dim(X \times Y) = dim(X) + dim(Y)$.

Тензорное произведение $V \otimes W$ пространств V и W – это пространство, которое использует в качестве базиса $\{v_1 \otimes w_1, v_1 \otimes w_2, v_2 \otimes w_1, v_2 \otimes w_2\}$. Отметим, что порядок базиса опять же произвольный. Так, пространство состояний двух кубитов, каждый с базисом $\{|0\rangle, |1\rangle\}$, имеет базис $\{|0\rangle \otimes |0\rangle, |0\rangle \otimes |1\rangle, |1\rangle \otimes |0\rangle, |1\rangle \otimes |1\rangle\} = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$. В общем случае, n-кубитная система имеет 2^n базисных векторов. Таким образом, мы наблюдаем экспоненциальное увеличение размерности пространства состояний с ростом числа квантовых частиц. Тензорное произведение $X \otimes Y$ имеет размерность $dim(X) \times dim(Y)$.

2.7 Теорема о неклонировании

Из унитарности квантовых преобразований следует важное и фундаментальное свойство квантовой информации – а именно, то, что квантовое состояние не может быть клонировано. Доказательство данной теоремы (Wooters и Zurek, 1982) есть простое применение свойства линейности унитарных преобразований.

Теорема 2.1 *(Теорема о неклонировании (невозможности копирования)). Неизвестное квантовое состояние не может быть клонировано. То есть не существует унитарного преобразования U такого, что для любого однокубитного состояния $|\psi\rangle$ выполняется*

$$U(|\psi 0\rangle) = |\psi\psi\rangle.$$

Теорема неклонирования верна для любого Гильбертова пространства.

Доказательство: Пусть $|a\rangle$ – произвольное квантовое состояние. Допустим, что существует унитарное преобразование

U, которое может клонировать произвольное квантовое состояние, то есть выполняется

$$U(|a0\rangle) = |aa\rangle.$$

Пусть $|b\rangle$ – состояние, ортогональное состоянию $|a\rangle$.

Поскольку U – клонирующее преобразование, то $U(|b0\rangle) = |bb\rangle$. Рассмотрим состояние $|c\rangle = 1/\sqrt{2}(|a\rangle + |b\rangle)$. По свойству линейности

$$U(|c0\rangle) = 1/\sqrt{2}(U(|a0\rangle) + U(|b0\rangle)) = 1/\sqrt{2}(|aa\rangle + |bb\rangle).$$

Но если U – клонирующее преобразование, то

$$U(|c0\rangle) = |cc\rangle = |c\rangle|c\rangle = 1/2(|aa\rangle + |ab\rangle + |ba\rangle + |bb\rangle),$$

что не равно $1/\sqrt{2}(|aa\rangle + |bb\rangle)$. Поэтому не существует унитарного преобразования, которое может клонировать произвольное квантовое состояние. Теорема доказана. □

Важно понимать, какого сорта клонирование не допускается в квантовых вычислениях. Возможно клонировать известное квантовое состояние. Что утверждает принцип неклонирования, так это принципиальную невозможность клонирования неизвестного квантового состояния. Так, возможно построить на n кубитах полностью запутанное состояние $a|00\ldots0\rangle + b|11\ldots1\rangle$ из состояния $a|0\rangle + b|1\rangle$. При измерении поведение каждой из этих n частиц будет совершенно одинаково (если измерение будет производиться по отношению к стандартному базису $\{|0\ldots0\rangle$, $|0\ldots1\rangle$, $\ldots|1\ldots1\rangle\}$, но не в случае, если измерение будет производиться по отношению к другим базисам). Но невозможно подготовить n-кубитное состояние $(a|0\rangle + b|1\rangle) \otimes \cdots \otimes (a|0\rangle + b|1\rangle)$ из неизвестного состояния $a|0\rangle + b|1\rangle$.

Рассмотрим задачу копирования классического бита. Это может быть сделано при помощи XOR гейта, принимающего на

вход бит, который мы хотим скопировать (в неизвестном состоянии x) и бит памяти, инициализированный в 0. Выходами являются два бита, оба из которых в состоянии x (рис. 2.2).

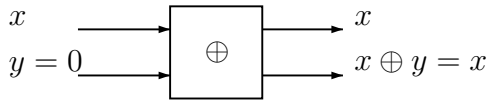


Рис. 2.2: Классический XOR-гейт

Предположим, что мы теперь хотим скопировать кубит в неизвестном состоянии $|\psi\rangle = a|0\rangle + b|1\rangle$ тем же самым способом, используя гейт C_{not}. Входное состояние двух кубитов может быть записано как

$$(a|0\rangle + b|1\rangle) \otimes |0\rangle = a|00\rangle + b|10\rangle$$

Гейт C_{not} инвертирует значение второго кубита, если значение первого есть 1, и поэтому выход гейта есть состояние $|out\rangle = a|00\rangle + b|11\rangle$. Два кубита теперь в одном и том же состоянии! Однако посмотрим, что происходит, если мы измерим один из кубитов. Как мы знаем, получим либо значение 0, либо значение 1 с вероятностью $|a|^2$ и $|b|^2$. Однако, как только один кубит измерен, состояние второго становится полностью детерминированным и никакой добавочной информации о a и b не существует. В этом смысле информация, содержавшаяся в исходном кубите $|\psi\rangle$, была потеряна при измерении и не может быть использована дважды. Фактически это означает, что информация из кубита не была скопирована.

2.8 Запутанные состояния

Представим макроскопический физический объект, который разрушился, и частицы его разлетелись в разных направлениях. Состояние такой системы может быть полностью описано посредством описания состояния каждой из частиц независимо. Одним из удивительных и не интуитивных феноменов квантовой механики является то, что состояние квантовой системы не всегда может быть описано в терминах состояний составляющих ее частиц.

К примеру, состояние $|\psi\rangle = 1/\sqrt{2}(|00\rangle + |11\rangle)$ не может быть разложено в независимые состояния каждой из частиц. Иными словами, мы не можем найти a_1, a_2, b_1, b_2 такие, что

$$(a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) = 1/\sqrt{2}(|00\rangle + |11\rangle),$$

так как

$$(a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) =$$
$$a_1a_2|00\rangle + a_1b_2|01\rangle + a_2b_1|10\rangle + b_1b_2|11\rangle.$$

Последнее выражение не может быть равно $1/\sqrt{2}(|00\rangle + |11\rangle)$, поскольку в этом случае должно быть либо $a_1 = 0$, либо $b_2 = 0$ (аналогично либо $a_2 = 0$, либо $b_1 = 0$), что невозможно, поскольку в этом случае будет $a_1a_2{=}0$, либо $b_1b_2 = 0$.

Состояния, которые не могут быть разложены данным образом, называются запутанными состояниями (entagled states).

Определение 2.7 *Состояние $\psi \in \mathcal{H}^4$ двухкубитной системы называется разложимым, если оно может быть записано как тензорное произведение состояний каждого кубита. В противном случае состояние называется запутанным.*

Состояние $|\psi\rangle = 1/2(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$ является разложимым. Действительно, поскольку $|\psi\rangle = 1/2(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = 1/\sqrt{2}(|0\rangle + |1\rangle) \otimes 1/\sqrt{2}(|0\rangle + |1\rangle)$.

Запутанные состояния описывают ситуацию, для которой нет классического аналога и для которой мы не имеем достаточно интуиции. Это также является одним из преимуществ, которое обеспечивает экспоненциальное возрастание пространства квантовых состояний по сравнению с классическими. Следует также отметить потребность в больших ресурсах для моделирования даже небольшой квантовой системы на традиционном компьютере, поскольку такое моделирование требует хранения траекторий экспоненциально большого количества состояний. Причина потенциальной мощности квантовых компьютеров — возможное использование эволюции квантовых состояний как вычислительного механизма.

Квантовое измерение дает другой взгляд на запутанные состояния.

Определение 2.8 *Состояние $\psi \in \mathcal{H}^4$ двух кубитов называется разложимым, если измерение одного из кубитов не оказывает влияния на измерение состояния другого. В противном случае, состояние называется запутанным.*

Например, состояние $|\psi\rangle = 1/\sqrt{2}(|00\rangle + |11\rangle)$ является запутанным, так как вероятность того, что значение первого кубита есть 0 равно $1/2$, если второй кубит не был измерен. Однако, если второй кубит был предварительно измерен, вероятность, что значение первого кубита при измерении будет $|0\rangle$ равна 1 или 0, в зависимости от того, каков был результат измерения второго кубита – $|0\rangle$ или $|1\rangle$. Поэтому результат измерения первого кубита изменяется в результате предарительного измерения второго. С другой стороны, состояние $|\psi\rangle = 1/\sqrt{2}(|00\rangle + |01\rangle)$ является разложимым, так как любое измерение первого кубита дает результат $|0\rangle$ независимо от того, было или нет произведено измерение второго кубита. Точно так же, измерение второго кубита дает одно из значений $|0\rangle$ или $|1\rangle$ с вероятностью $1/2$ независи-

мо от того, был ли измерен первый кубит. Запутанность в том смысле, что измерение частицы оказывает влияние на измерение другой частицы, эквивалентна предыдущему определению запутанности состояний в терминах разложимости в тензорное произведение индивидуальных состояний. Заметим, что существуют и запутанные состояния, включающие в себя большее число кубитов.

Глава 3

Квантовые гейты

3.1 Квантовые гейты

Эволюция квантовой системы, не подвергающейся измерению, переводит состояние квантовой системы в другое состояние с сохранением нормы. Для Гильбертова пространства преобразование, сохраняющее расстояние, является унитарным преобразованием, определяемым следующим образом. Любое преобразование на d-мерном комплексном векторном пространстве может быть описано при помощи $d \times d$-матрицы. Пусть $M^\dagger$ обозначает транспонированную комплексносопряженную матрицу для матрицы M.

Матрица M называется унитарной (описывающей унитарные преобразования), если $MM^\dagger = I$.

Любое унитарное преобразование квантового пространства состояний есть допустимое квантовое преобразование, и наоборот. Мы можем трактовать унитарное преобразование как повороты в комплексном векторном пространстве.

Важным следствием факта, что любое квантовое преобразование унитарно, является его обратимость. Поэтому квантовые гейты должны быть обратимы.

В классическом компьютере преобразование информации осуществляется логическими гейтами. Логический гейт преобразовывает состояние входных битов в другое состояние соответ-

ственно таблице истинности. Простейший нетривиальный логический гейт – это NOT-гейт, однобитный гейт, который осуществляет отрицание входного бита: 0 заменяет на 1 и наоборот. Известно, что при помощи классических логических гейтов NOT, AND, OR (или даже NOT, AND) можно построить схему для вычисления любой булевой функции, поскольку данный набор гейтов составляет полный базис (см. рис. 3.1, 3.2, 3.3).

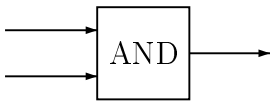


Рис. 3.1: Классический гейт AND

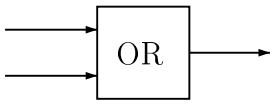


Рис. 3.2: Классический гейт OR

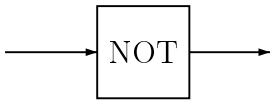


Рис. 3.3: Классический гейт NOT

Подобно классическому, квантовый компьютер может быть построен из схем логических гейтов. Соответствующие квантовые гейты выполняются посредством унитарной операции, которая отображает базисные состояния в соответствующие базисные состояния согласно той же самой таблице истинности.

Определение 3.1 *Квантовый гейт с n входами и n выходами — это преобразование, заданное на n кубитах и определяемое унитарной матрицей U размерности $2^n \times 2^n$.*

Квантовая схема — это квантовая вычислительная модель, построенная из квантовых логических гейтов, в которых вычис-

лительные шаги синхронизированы по времени. Входы квантовых гейтов связаны с входами схемы или с выходами других гейтов. Выходы некоторых гейтов связаны со входами других гейтов. Сложная унитарная операция может быть представлена в виде схемы, состоящей из нескольких квантовых гейтов.

3.2 Однокубитные квантовые гейты

Из трех рассмотренных классических гейтов AND, OR и NOT единственным обратимым гейтом является гейт NOT. Квантовая версия классического гейта NOT – это есть унитарная однокубитная операция U_{NOT} такая, что

$$U_{NOT}|0\rangle = |1\rangle$$

$$U_{NOT}|1\rangle = |0\rangle$$

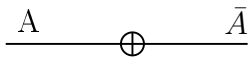


Рис. 3.4: Графическое изображение квантового гейта NOT

Приведем примеры еще некоторых однокубитных квантовых преобразований.

$$\text{I:}\quad \begin{matrix}|0\rangle \to |0\rangle \\ |1\rangle \to |1\rangle\end{matrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\text{X:}\quad \begin{matrix}|0\rangle \to |1\rangle \\ |1\rangle \to |0\rangle\end{matrix} \qquad \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\text{Y:}\quad \begin{matrix}|0\rangle \to -|1\rangle \\ |1\rangle \to |0\rangle\end{matrix} \qquad \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

$$\text{Z:}\quad \begin{matrix}|0\rangle \to |0\rangle \\ |1\rangle \to -|1\rangle\end{matrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Здесь I– это тождественное преобразование, X – отрицание (гейт NOT), Z – операция поворота фазы, Y – комбинация преобразований Z и X. Может быть легко проверено, что эти преобразования являются унитарными. Например,

$$YY^\dagger = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = I.$$

Вследствие линейности данные преобразования полностью определяются их действием на базисные векторы. Отметим, что квантовые гейты преобразуют состояния, находящиеся в суперпозиции, и поэтому не могут рассматриваться аналогично классическим гейтам (см. рис. 3.5).

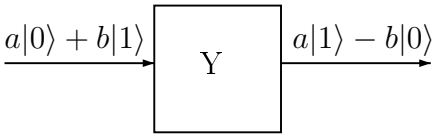


Рис. 3.5: Действие квантового гейта Y на суперпозицию квантовых состояний кубита

Вообще, существует бесконечно много однокубитных гейтов,

все они могут быть образованы из операции поворота

$$U_R(\theta) = \begin{pmatrix} \cos(\theta) & -\sin\theta \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

и операции сдвига фазы

$$U_P(\phi_1, \phi_2) = \begin{pmatrix} e^{i\phi_1} & 0 \\ 0 & e^{i\phi_2} \end{pmatrix}.$$

3.3 Преобразование Уолша-Адамара

Другое важное однокубитное преобразование – это преобразование Адамара, определяемое следующим образом.

$$H : |0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$H : |1\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Преобразование H имеет большое число важных приложений. Когда оно применяется к состоянию $|0\rangle$, это преобразование производит равномерную суперпозицию двух состояний $|0\rangle$ и $|1\rangle$. Если мы имеем регистр из n кубитов, то, применяя преобразование Адамара к каждому из кубитов индивидуально, получаем суперпозицию всех 2^n состояний (которые могут быть представлены как бинарное представление чисел от 0 до $2^n - 1$).

$$(H \otimes H \otimes \cdots \otimes H)|00\ldots0\rangle =$$

$$\frac{1}{\sqrt{2^n}}((|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)) = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$$

Таким образом, при помощи линейного числа операций (n применений преобразования H) мы можем образовать регистр состояний, содержащий экспоненциальное (2^n) число различных

компонент. То есть, используя квантовый регистр, n элементарных операций образуют состояние, содержащее все 2^n возможных различных численных значений регистра. К сравнению, в классическом регистре n элементарных операций могут подготовить только одно состояние, представляющее только одно конкретное число.

Преобразование, которое применяет H к регистру из n кубит, называется преобразованием Уолша-Адамара и обозначается W. Оно может быть определено рекурсивно следующим образом.

$$W_1 = H, W_{n+1} = H \otimes W_n.$$

3.4 Двухкубитные гейты

Известно, что при помощи классических логических гейтов NOT, AND, OR (или даже NOT, AND) можно построить схему для вычисления любой булевой функции, поскольку данный набор гейтов составляет полный базис. К сожалению, классические гейты AND, OR являются необратимыми булевскими операциями. Мы не можем по выходному значению гейта восстановить, каково было входное значение – информация становится необратимо потерянной в результате операции гейта. При обратимых гейтах, булевых функциях, операциях и вычислениях в выходном значении всегда содержится достаточно информации, чтобы однозначно восстановить входное значение. Только такие функции, гейты и операции допускаются в квантовых вычислениях из-за условия обратимости эволюции в квантовой физике.

Квантовый гейт Controlled NOT (C_{not} или XOR) действует на двух кубитах следующим образом. Он изменяет значение второго кубита, если значение первого равно 1, и оставляет второй кубит неизменным в противном случае. Графическое изображение квантового гейта C_{not} приведено на рис. 3.6.

$$C_{not}: \begin{array}{ccc} |00\rangle & \rightarrow & |00\rangle \\ |01\rangle & \rightarrow & |01\rangle \\ |10\rangle & \rightarrow & |11\rangle \\ |11\rangle & \rightarrow & |10\rangle \end{array} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

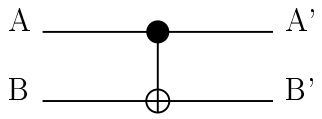


Рис. 3.6: Графическое изображение квантового гейта C_{not}

Преобразование C_{not} является унитарным, поскольку $C_{not}^{\dagger} = C_{not}$ и $C_{not}C_{not} = I$.

Свойство 3.1 *Гейт C_{not} не может быть разложен в тензорное произведение двух однокубитных гейтов.*

3.5 Трехкубитные гейты

В данной главе мы рассмотрим некоторые квантовые гейты, в которых задействовано три кубита. Также покажем существование универсальных квантовых гейтов.

Трехкубитный гейт CC_{not}. Трехкубитный гейт CC_{not} меняет значение третьего кубита, если два первых кубита равны единице, и оставляет третий кубит без изменения в противном случае. Ниже приведены графическое изображение и таблица истинности для данного гейта.

Рис. 3.7: Графическое изображение квантового гейта CC_{not}

A	B	C	A'	B'	C'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

3.6 Универсальные квантовые гейты

Известно, что с помощью классических гейтов AND, NOT, OR можно вычислить любую булеву функцию. То есть этот набор гейтов составляет полный базис. Было показано, что любую обратимую булеву функцию из $\mathcal{B}_n^m$, $n \geq 3$ можно вычислить обратимой схемой, составленной из гейтов NOT, C_{not}, CC_{not}.

В действительности было показано существование универсального квантового гейта. Это один из следующих гейтов с 3 входами и 3 выходами: Тоффоли гейт и Фредкин гейт.

Фредкин гейт устроен следующим образом: он осуществляет перестановку кубитов B и C при условии, что значение кубита A равно 0. Ниже приведены графическое изображение и таблица истинности для Фредкин гейта. (рис. 3.8 и табл. 3.1)

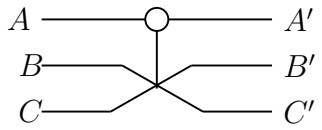


Рис. 3.8: Графическое изображение Фредкин гейта

A	B	C	A'	B'	C'
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	1

Таблица 3.1: Таблица истинности Фредкин гейта

Тоффоли гейт - инвертирует кубит B при условии, что значения кубитов A и C равны 1. Ниже приведены графическое изображение и таблица истинности для гейта Тоффоли (рис. 3.9 и табл. 3.2).

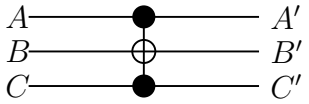


Рис. 3.9: Графическое изображение гейта Тоффоли

Утверждение 3.1 *Гейт Тоффоли является универсальным квантовым гейтом.*

Доказательство: Действительно, в Тоффоли гейте: если $A = 1$, то $B' = B \oplus C$. Если $B = 0$, то $B' = A\&C$. Если $A = 1$,

A	B	C	A'	B'	C'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	1	0	1

Таблица 3.2: Таблица истинности Тоффоли гейта

$C = 1$, то $B' = \bar{B}$. Следовательно, гейты AND и NOT реализуемы гейтом Тоффоли, что и доказывает его универсальность. □

Утверждение 3.2 *Гейт Фредкина является универсальным квантовым гейтом.*

Доказательство: Для гейта Фредкина: если $C = 0$, то $B' = A\&B$. Если $B = 0, C = 1$, то $B' = \bar{A}$.

Следовательно, гейты AND и NOT реализуемы гейтом Фредкина, что и доказывает его универсальность. □

Также можно построить гейт контролируемой унитарной операции, где унитарное преобразование U над несколькими кубитами выполняется или нет в зависимости от значения управляющего кубита (рис. 3.10). На рис. 3.11 представлен гейт, где унитарное преобразование контролируется несколькими кубитами.

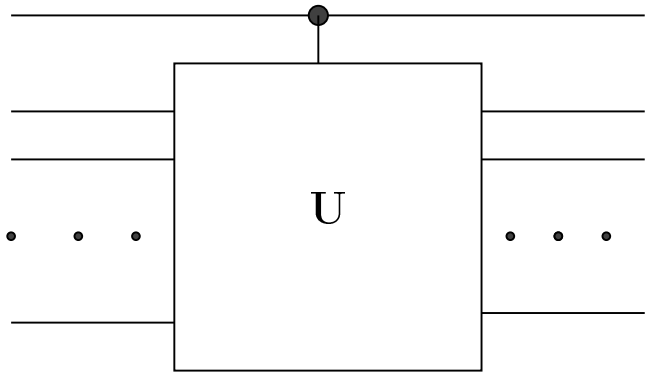


Рис. 3.10: Унитарный гейт, контролируемый одним кубитом

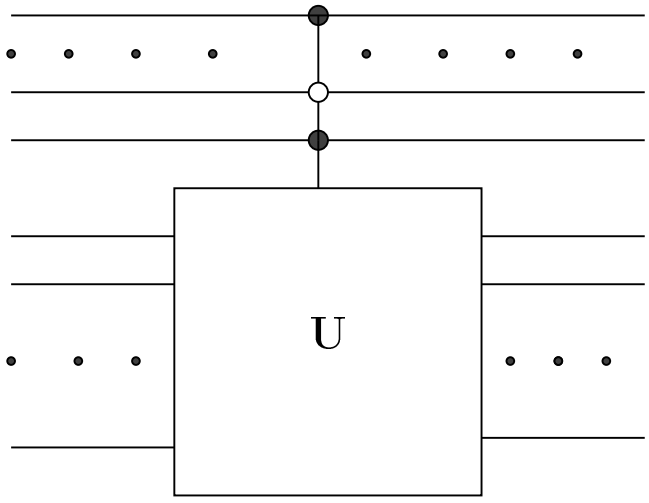


Рис. 3.11: Унитарный гейт, контролируемый несколькими кубитами

Глава 4

Квантовый параллелизм

4.1 Вычисление функций

Рассмотрим функцию $f : \{0, 1, \ldots, 2^n - 1\} \to \{0, 1, \ldots, 2^m - 1\}$, где n, m – положительные числа. Классическое устройство вычисляет f, отображая каждый указанный вход $0, 1, \ldots, 2^n - 1$ в соответствующее значение выхода $f(0), f(1), \ldots, f(2^n - 1)$. Квантовый компьютер, вследствие унитарной (а следовательно, обратимой) природы своей эволюции, вычисляет функции слегка отличным образом. А именно, невозможно вычислить произвольную функцию f посредством унитарной операции, которая отображает $|x\rangle$ в $|f(x)\rangle$: если f не взаимнооднозначное отображение (то есть если $f(x) = f(y)$ для некоторого $x \neq y$), то два ортогональных кет-вектора $|x\rangle$ и $|y\rangle$ перейдут в одно и то же значение $f(x) = f(y)$, что нарушает унитарность. Один из способов вычисления функций, не являющихся взаимнооднозначными отображениями, при котором не нарушается обратимость – это сохранение входного значения. Для достижения этого квантовый вычислитель использует два регистра: один – для входных данных, второй – для выходных данных. Каждый возможный вход $|x\rangle$ – содержимое первого регистра. Аналогично, каждый возможный выход $y = f(x)$, представляемый как $|y\rangle$, есть содержимое второго регистра. Состояния, соответствующие различным входам и различным выходам, ортогональны.

Вычисление функции тогда определяется унитарным оператором U_f, который действует на двух регистрах:

$$U_f(|x\rangle|0\rangle) = |x\rangle|f(x)\rangle.$$

Вычисления, осуществляемые таким образом, являются не только обратимыми, но и квантовыми, и при этом мы можем сделать более, чем вычислять по очереди все значения функции одно за другим. Мы можем вычислить значения функции на всех входных аргументах сразу.

Подготовим суперпозицию всех входных значений как одно состояние следующим образом. Начинаем с состояния $|00\ldots0\rangle$. Применяем преобразование Уолша-Адамара, чтобы получить суперпозицию $\frac{1}{\sqrt{2^n}}(|00\ldots0\rangle + |00\ldots1\rangle + \cdots + |11\ldots1\rangle)$, которую мы можем рассматривать как суперпозицию всех чисел $0 \leq x \leq 2^n - 1$. Так как U_f – линейное преобразование, оно может быть применено ко всем базисным состояниям в суперпозиции независимо и будет образовывать суперпозицию результатов. Таким образом, применяя вычисление U_f только один раз, мы можем вычислить все 2^n значений $f(0), \ldots, f(2^n - 1)$:

$$|\psi\rangle = U_f \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|f(x)\rangle.$$

Этот эффект называется *квантовым параллелизмом*. Так как n кубит позволяют работать одновременно с 2^n состояниями, квантовый параллелизм обладает преимуществом перед классическим параллелизмом, так как обеспечивает экспоненциальный размер вычислительного пространства, используя линейный размер физического пространства.

Это выглядит слишком уж хорошо чтобы быть правдой, так в чем же загвоздка? Как много информации об f содержит $|\psi\rangle$? Как мы и ожидаем, никакое квантовое измерение не может из-

влечь все 2^n значений $f(0)$, $f(1), \ldots, f(2^n - 1)$ из $|\psi\rangle$.

Рассмотрим тривиальный пример - Тоффоли гейт T.

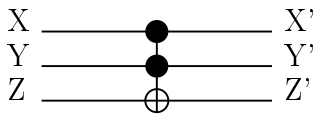


Рис. 4.1: Тоффоли гейт

Теперь возьмем в качестве входа суперпозицию всех возможных комбинаций битов x и y с обязательным 0 в качестве z.

$$H(|0\rangle) \otimes H(|0\rangle) \otimes |0\rangle =$$
$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle =$$
$$\frac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |110\rangle)$$

Суперпозиция на входах приводит к суперпозиции результатов, то есть

$$T(H|0\rangle \otimes H|0\rangle \otimes |0\rangle) = \frac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |111\rangle)$$

На результирующую суперпозицию можно смотреть как на таблицу истинности для конъюнкции. Измерение результата дает одну строчку таблицы истинности. Отметим, что биты могут быть измерены в разном порядке: измерение бита результата спроецирует состояние в суперпозицию множества всех входных значений, для которых f производит один и тот же результат, а измерение входов спроецирует результат на соответствующее значение функции.

Сердцем любого квантового алгоритма является способ, при помощи которого он манипулирует квантовым параллелизмом

таким образом, что желаемый результат получится в результате измерения с высокой вероятностью. Данный сорт манипуляций не имеет классического аналога и требует нетрадиционной техники программирования. Вот список пары известных на сегодняшний день техник.

1. Усиление нужного выходного значения. Общая идея состоит в преобразовании состояния таким образом, чтобы интересующее нас значение имело большую амплитуду и, следовательно, большую вероятность проявиться при измерении.

2. Нахождение общих свойств всех значений $f(x)$, таких, например, как периодичность функции. Эта идея использовалась в алгоритме Шора.

Глава 5

Простейшие квантовые алгоритмы

Ниже приводятся два алгоритма с использованием простейших квантовых гейтов – алгоритм плотного (сжатого) кодирования и телепортация.

5.1 Плотное кодирование

Dense coding (плотное (сжатое) кодирование) использует один квантовый бит вместе с ЭПР-парой для кодирования и передачи двух классических битов. Так как ЭПР-пара может быть заготовлена и передана заранее, только один кубит (частицу) необходимо физически передать, чтобы передать два бита информации. Это результат, с одной стороны, удивительный, поскольку мы говорили, что один кубит может содержать только одну единицу информации.

Телепортация, в противоположность плотному кодированию, использует два классических бита для передачи одного кубита. Эффект телепортации удивителен в свете принципа неклонирования в квантовой механике и позволяет пересылать неизвестное квантовое состояние.

Ключевой момент, который используется как в плотном кодировании, так и в телепортации - использование запутанных частиц. Начальное состояние одно и то же для обоих процессов. Алиса и Боб хотят обменяться сообщением. Каждому из них пе-

ресылается один из двух кубитов, приготовленных в запутанном состоянии

$$|\psi_0\rangle = 1/\sqrt{2}(|00\rangle + |11\rangle).$$

Например, Алиса получает первую частицу, Боб - вторую. До тех пор пока частица не передана, Алиса может совершать преобразования только на своем кубите, Боб - только на своем.

Алиса: Алиса получает два классических бита, кодирующих число от 0 до 3. В зависимости от значения этого числа, Алиса совершает одно из преобразований $\{I, X, Y, Z\}$ на своем кубите запутанной пары $|\psi_0\rangle$. Результирующее состояние показано в следующей таблице.

Значение	Преобразование	Новое состояние
0	$\|\psi_0\rangle = (I \otimes I)\|\psi_0\rangle$	$\frac{1}{\sqrt{2}}(\|00\rangle + \|11\rangle)$
1	$\|\psi_1\rangle = (X \otimes I)\|\psi_0\rangle$	$\frac{1}{\sqrt{2}}(\|10\rangle + \|01\rangle)$
2	$\|\psi_2\rangle = (Y \otimes I)\|\psi_0\rangle$	$\frac{1}{\sqrt{2}}(-\|10\rangle + \|01\rangle)$
3	$\|\psi_3\rangle = (Z \otimes I)\|\psi_0\rangle$	$\frac{1}{\sqrt{2}}(\|00\rangle - \|11\rangle)$

Алиса передает свой кубит Бобу.

Боб: Боб применяет преобразование Controlled-NOT к двум кубитам запутанной пары.

Нач. состояние	Controlled-NOT	Первый бит	Второй бит
$\frac{1}{\sqrt{2}}(\|00\rangle + \|11\rangle)$	$\frac{1}{\sqrt{2}}(\|00\rangle + \|10\rangle)$	$\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)$	$\|0\rangle$
$\frac{1}{\sqrt{2}}(\|10\rangle + \|01\rangle)$	$\frac{1}{\sqrt{2}}(\|11\rangle + \|01\rangle)$	$\frac{1}{\sqrt{2}}(\|1\rangle + \|0\rangle)$	$\|1\rangle$
$\frac{1}{\sqrt{2}}(-\|10\rangle + \|01\rangle)$	$\frac{1}{\sqrt{2}}(-\|11\rangle + \|01\rangle)$	$\frac{1}{\sqrt{2}}(-\|1\rangle + \|0\rangle)$	$\|1\rangle$
$\frac{1}{\sqrt{2}}(\|00\rangle - \|11\rangle)$	$\frac{1}{\sqrt{2}}(\|00\rangle - \|10\rangle)$	$\frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)$	$\|0\rangle$

Теперь Боб может измерить второй кубит без разрушения квантового состояния. Если результат измерения $|0\rangle$, тогда закодированное значение либо 0, либо 3. Если результат измерения $|1\rangle$ – тогда закодированное значение 1 или 2. Теперь Боб применяет преобразование Адамара к первому кубиту:

Нач. состояние	Первый бит	H (Первый бит)
$\|\psi_0\rangle$	$\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)$	$\frac{1}{\sqrt{2}}(\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle) + \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)) = \|0\rangle$
$\|\psi_1\rangle$	$\frac{1}{\sqrt{2}}(\|1\rangle + \|0\rangle)$	$\frac{1}{\sqrt{2}}(\frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle) + \frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)) = \|0\rangle$
$\|\psi_2\rangle$	$\frac{1}{\sqrt{2}}(-\|1\rangle + \|0\rangle)$	$\frac{1}{\sqrt{2}}(\frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle) + \frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle))\|1\rangle$
$\|\psi_3\rangle$	$\frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)$	$\frac{1}{\sqrt{2}}(\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle) + \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle))\|1\rangle$

Наконец, Боб измеряет результирующий бит, что позволяет ему отличить 0 от 3, и 1 от 2.

5.2 Телепортация

Задача состоит в передаче квантового состояния частицы, используя передачу классических битов с последующим воссозданием точного квантового состояния у получателя. Так как, по Теореме о неклонировании, оно не может быть скопировано, квантовое состояние исходной частицы обязательно будет разрушено. Телепортация одного кубита впервые была реализована экспериментально в 1997 году.

Алиса хочет передать состояние кубита

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

Бобу по классическому каналу. Как и в плотном кодировании, Алиса и Боб каждый владеет одним из битов запутанной пары

$$|\psi_0\rangle = 1/\sqrt{2}(|00\rangle + |11\rangle).$$

Алиса: Алиса применяет декодирующий шаг плотного кодирования к кубиту $|\psi\rangle$, который должен быть передан, и к своей половине запутанной пары. Начальным квантовым состоянием является следующее:

$$|\psi\rangle \otimes |\psi_0\rangle = 1/\sqrt{2}(a|0\rangle \otimes (|00\rangle + |11\rangle) + b|1\rangle(|00\rangle + |11\rangle)) =$$

$$1/\sqrt{2}(a|000\rangle + a|011\rangle + b|100\rangle + b|111\rangle),$$

в котором Алиса контролирует первые два бита, Боб - последний бит. Алиса применяет преобразования $C_{not} \otimes I$ и $H \otimes I \otimes I$ к этому состоянию.

$$(H \otimes I \otimes I)(C_{not} \otimes I)(|\psi\rangle \otimes |\psi_0\rangle) =$$

$$(H \otimes I \otimes I)(C_{not} \otimes I)\frac{1}{\sqrt{2}}(a|000\rangle + a|011\rangle + b|100\rangle + b|111\rangle) =$$

$$(H \otimes I \otimes I)\frac{1}{\sqrt{2}}(a|000\rangle + a|011\rangle + b|110\rangle + b|101\rangle) =$$

$$\frac{1}{2}(a(|000\rangle + |011\rangle + |011\rangle + |111\rangle) + b(|010\rangle + |001\rangle - |110\rangle - |111\rangle)) =$$

$$\frac{1}{2}(|00\rangle(a|0\rangle + b|1\rangle) + |01\rangle(a|1\rangle + b|0\rangle) + |10\rangle(a|0\rangle - b|1\rangle) + |11\rangle(a|1\rangle - b|0\rangle))$$

Алиса производит измерение своих двух кубитов и получает один из результатов $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ с равной вероятностью. В зависимости от исхода измерения текущее состояние проецируется в одно из $a|0\rangle + b|1\rangle$, $a|1\rangle + b|0\rangle$, $a|0\rangle - b|1\rangle$ либо $a|1\rangle + b|0\rangle$, соответственно. Алиса передает результат измерения Бобу, используя при этом два классических бита. Отметим, что в результате измерения Алиса безвозвратно изменяет состояние $|\psi\rangle$ своего оригинального кубита, которое она хотела передать Бобу. Эта потеря оригинального состояния как раз объясняет, почему

эффект телепортации ни коим образом не противоречит Теореме неклонирования.

Боб: Когда Боб получает два классических бита от Алисы, он знает, как сопоставить состояние своей половины запутанной пары с оригинальным состоянием кубита Алисы.

Полученные биты	Состояние	Декодирование
00	$a\|0\rangle + b\|1\rangle$	I
01	$a\|1\rangle + b\|0\rangle$	X
10	$a\|0\rangle - b\|1\rangle$	Z
11	$a\|1\rangle - b\|0\rangle$	Y

Боб может воссоздать оригинальное состояние $|\psi\rangle$ кубита Алисы, применяя декодирующее преобразование к своей части запутанной пары. Отметим, что это кодирующий шаг алгоритма плотного кодирования.

5.3 Алгоритм Дойча

Проблема Дойча – это простейший и один из самых первых примеров задач, для которых квантовый алгоритм является экспоненциально более эффективным, чем классические, как детерминированные, так и вероятностные. Данная проблема – это пример, когда мы жертвуем полной информацией в пользу относительной. Самая ранняя версия этого алгоритма появилась в 1985 году в статье Дэвида Дойча. Она, вместе со статьей Ричарда Фейнмана 1982 года, всколыхнула весь мир. Первый алгоритм, предложенный Дойчем, был вероятностный. С вероятностью $1/2$ он выдавал «не знаю». С вероятностью $1/2$ он выдавал ответ, и в случае выдачи ответа ответ был точный. Позже появилась версия алгоритма, который выдавал точный ответ в 100% случаях. Этот алгоритм мы здесь и изложим. Опишем данную проблему и квантовый алгоритм для нее. Входной и выходной регистры содержат единственный кубит, то есть мы рассматриваем булеву функцию, которая переводит единственный бит в единственный бит. Четыре возможные функций такого типа – следующие:

	x=0	x=1
f_0	0	0
f_1	0	1
f_2	1	0
f_3	1	1

Существуют две трактовки данной задачи.

Первая состоит в следующем. Существует 4 возможных функции, отображающих один бит в один бит, которые представлены в таблице. Предположим, что у нас имеется черный ящих

(black-box), который вычисляет одну из этих четырех функций в обычном квантовом формате, осуществляя квантовое преобразование

$$U_f(|x\rangle|y\rangle) = |x\rangle|y \oplus f(x)\rangle,$$

где первый из двух регистров является входным регистром (i), второй – выходным регистром (o). Легко понять, что

- $U_{f_0} = I_i \otimes I_o$,
- $U_{f_1} = CNOT_{io}$,
- $U_{f_2} = CNOT_{io}(I_i \otimes NOT_o)$,
- $U_{f_3} = I_i \otimes NOT_o$,

где I – тождественное преобразование, $CNOT_{io}$ – controlled-NOT с входным регистром в качестве управляющего и выходным регистром в качестве целевого, NOT_o – преобразование, инвертирующее значение выходного регистра (смотри рис. 5.1 – 5.4).

Предположим, что наш черный ящик вычисляет одну из четырех функций, но мы не знаем, какую из них. Наша задача – узнать, что за операцию выполняет черный ящик. Естественно, мы можем узнать, какую именно операцию выполняет черный ящик, заставив его сработать дважды: сначала на входе $|0\rangle|0\rangle$, а затем -– на входе $|1\rangle|0\rangle$. Но предположим, что нам дозволен доступ к черному ящику лишь единожды. Что мы можем узнать об f? На классическом компьютере при условии, что нам разрешено использование черного ящика лишь один раз, мы можем узнать либо значение $f(0)$, либо значение $f(1)$. В случае если мы выбрали значение $f(0)$, можем сделать вывод, что искомая функция есть f_0 или f_1 (если значение $f(0) = 0$), либо искомая функция есть f_2 или f_3 (если значение $f(0) = 1$). Если мы выбрали значение $f(1)$, можем сделать вывод, что искомая функция

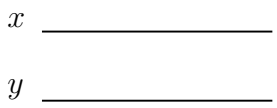


Рис. 5.1: U_{f_0}

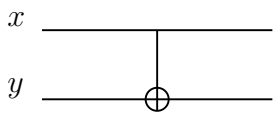


Рис. 5.2: U_{f_1}

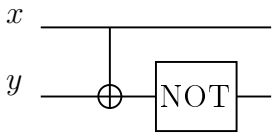


Рис. 5.3: U_{f_2}

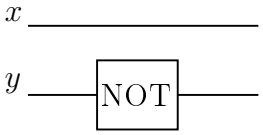


Рис. 5.4: U_{f_3}

есть f_0 или f_2 (если значение $f(1) = 0$), либо искомая функция есть f_1 или f_3 (если значение $f(1) = 1$).

Предположим теперь, что мы хотим узнать, является ли искомая функция константой ($f(0) = f(1)$), что соответствует f_0 или f_3, или неконстантной ($f(0) \neq f(1)$), что соответствует f_1 или f_2. Для классического компьютера не существует способа одновременно вычислить как $f(0)$, так и $f(1)$ и сравнить их. Для того чтобы определить, является ли f константой или нет, нам надо извлечь полную информацию об f. Следовательно, мы должны заставить сработать U_f дважды. Замечательно, но для квантового компьютера мы не должны выполнять U_f дважды, чтобы определить, является ли f константой, или нет. Мы можем сделать это, запустив U_f только один раз. Интересно, что при этом мы не получаем информации, каковы индивидуальные значения $f(0)$ и $f(1)$, но тем не менее мы получаем ответ на вопрос, каковы их взаимные значения: являются ли они одинаковыми или нет. То есть мы получаем меньше информации, чем получили бы, отвечая на данный вопрос с использованием классического компьютера, но, не получая части информации, которая не относится к нужному нам вопросу, мы можем получить ответ только за одно обращение к черному ящику.

Существует другой взгляд на проблему Дойча, который имеет нетривиальный математический контекст. Мы можем трактовать вход x как выбор двух различных входов для выполнения подпрограммы, которая требует много дополнительных кубит. При этом $f(x)$ мы можем трактовать как некоторую характеристику двузначного выхода подпрограммы. Например, $f(x)$ может быть значение милионного бита в бинарном представлении значения $\sqrt{2+x}$, то есть $f(0)$ – значение миллионого бита бинарного представления $\sqrt{2}$, $f(1)$ — значение миллионого бита бинарного представления $\sqrt{3}$. Таким образом, входной регистр осуществляет выбор данных в подпрограмму, а подпрограмма

возвращает данные в выходной регистр.

В интерпретации проблемы Дойча, вопрос о том, является ли f константой или нет, на самом деле представляет собой вопрос о том, как узнать что-то о природе черного ящика, который выполняет U_f, если мы не можем открыть его и заглянуть внутрь. В случае нашего примера интересующий нас нетривиальный вопрос – это: являются ли миллионные биты $\sqrt{2}$ и $\sqrt{3}$ одинаковыми или нет? В этой интерпретации, отвечая на вопрос с использованием классического компьютера, мы не можем сделать ничего лучше, как дважды запустить черный ящик, с 0 и 1 в качестве входов, и сравнить выходы.

В квантовом случае мы можем применить стандартный трюк, подготовив входной регистр в суперпозиции $1/\sqrt{2}(|0\rangle + |1\rangle)$. Тогда финальное состояние входного и выходного регистров будет

$$U_f(H \otimes I)(|0\rangle|0\rangle) = 1/\sqrt{2}|0\rangle|f(0)\rangle + 1/\sqrt{2}|1\rangle|f(1)\rangle.$$

Если мы измерим второй регистр, мы узнаем миллионный бит либо $\sqrt{2}$, либо $\sqrt{3}$. Результат будет равновероятным, и в этом случае мы не получаем преимущества по сравнению с классическим случаем.

Однако было замечено, что существует унитарное преобразование, которое мы можем применить до измерения, такое, что в половине случаев мы можем с точностью сказать, верно ли, что $f(0) = f(1)$. Некоторое время спустя были найдены унитарные преобразования, применение которых позволило получать верный ответ во всех случаях. Опишем данный алгоритм. Раньше в качестве входа к U_f мы брали состояние

$$(H \otimes I)(|0\rangle|0\rangle).$$

Вместо этого теперь стартовое состояние будет следующее:

$$(H \otimes H)(NOT \otimes NOT)(|0\rangle|0\rangle).$$

Так как $NOT|0\rangle = |1\rangle$ и $H|1\rangle = 1/\sqrt{2}(|0\rangle - |1\rangle)$, входом к U_f будет состояние

$$(H \otimes H)(NOT \otimes NOT)(|0\rangle|0\rangle) =$$
$$(1/\sqrt{2}|0\rangle - 1/\sqrt{2}|1\rangle)(1/\sqrt{2}|0\rangle - 1/\sqrt{2}|1\rangle) =$$
$$(|0\rangle|0\rangle - |1\rangle|0\rangle - |0\rangle|1\rangle + |1\rangle|1\rangle)$$

После применения U_f к данному состоянию в силу линейности результирующее состояние будет

$$1/2(U_f(|0\rangle|0\rangle) - U_f(|1\rangle|0\rangle) - U_f(|0\rangle|1\rangle) + U_f(|1\rangle|1\rangle)) =$$
$$1/2\big(|0\rangle|f(0)\rangle - |1\rangle|f(1)\rangle - |0\rangle|\bar{f}(0)\rangle + |1\rangle|\bar{f}(1)\rangle\big).$$

Поэтому, если $f(0) = f(1)$, выходное состояние есть

$$1/2(|0\rangle - |1\rangle)(|f(0)\rangle - |\bar{f}(0)\rangle), \qquad f(0) = f(1).$$

Если же $f(0) \neq f(1)$, значит $f(1) = \bar{f}(0), \bar{f}(1) = f(0)$, и выходное состояние будет

$$1/2(|0\rangle + |1\rangle)(|f(0)\rangle - |\bar{f}(0)\rangle), \qquad f(0) \neq f(1).$$

Применим теперь преобразование Адамара ко входному регистру. Получим

$$|1\rangle\frac{1}{\sqrt{2}}(|f(0)\rangle - |\bar{f}(0)\rangle), \qquad f(0) = f(1),$$
$$|0\rangle\frac{1}{\sqrt{2}}(|f(0)\rangle - |\bar{f}(0)\rangle), \qquad f(0) \neq f(1),$$

Измеряя первый регистр, мы можем дать ответ, являются ли значения $f(0)$ и $f(1)$ одинаковыми или нет. Заметим, что второй регистр совсем не содержит полезной информации, поскольку при измерении мы получим с равной вероятностью значения $|f(0)\rangle$ или $|\bar{f}(0)\rangle$. То есть цена, которую мы заплатили за то, чтобы узнать, одинаковы ли значения $f(0)$ и $f(1)$, – потеря информации об их действительных значениях. На рис. 5.5 показана квантовая схема алгоритма Дойча.

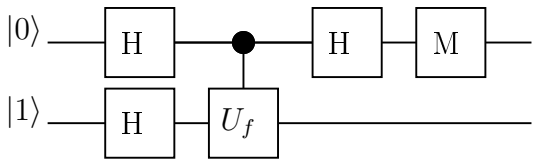


Рис. 5.5: Квантовая схема алгоритма Дойча

5.4 Проблема Дойча-Джозса

Рассмотрим алгоритм, который является обобщением алгоритма Дойча – алгоритм Дойча-Джозса. Пусть имеется функция $f : \{0,1\}^n \to \{0,1\}$. При этом известно, что функция f является либо константной, либо сбалансированной. Под сбалансированной понимается функция, которая принимает значение 0 и значение 1 на равном количестве входов (то есть на 2^{n-1} входах значение функции равно 0, и на 2^{n-1} входах значение функции равно 1). Проблема ставится следующим образом: определить, является ли функция f константой или сбалансированной.

Преобразование Уолша-Адамара. Ключевым квантовым преобразованием алгоритма, который мы сейчас будем рассматривать, является преобразование Уолша-Адамара, определенное нами ранее. Остановимся на нем подробнее. Преобразование Уолша-Адамара – это применение преобразования Адамара независимо к каждому кубиту n-кубитного регистра. Результат применения данного преобразования к произвольному базисному состоянию n-кубитного регистра следующий:

$$H_n|x\rangle = 1/\sqrt{2^n} \sum_{y \in B_n} (-1)^{x \cdot y} |y\rangle,$$

где операция $x \cdot y$ – операция побитного скалярного умножения векторов x и y по модулю 2. То есть $x \cdot y = x_1 y_1 \oplus \cdots \oplus x_n y_n$. При этом, если данная операция применяется к базисному состоянию

$|00\ldots0\rangle$, то результатом является равномерная суперпозиция всех возможных состояний регистра. То есть

$$H_n|00\ldots0\rangle = 1/\sqrt{2^n}\sum_{y\in B_n}|y\rangle,$$

Обратное преобразование к H_n совпадает с самим же преобразованием H_n. То есть выполняется $H_nH_n = I$. Поэтому

$$H_n(1/\sqrt{2^n}\sum_{y\in B^n}|y\rangle) = |00\ldots0\rangle.$$

Вернемся к алгоритму Дойча-Джозса. Как и в алгоритме Дойча, постановка задачи следующая. Имеется черный ящик, который вычисляет функцию f. То есть, подавая на вход черного ящика значение аргумента x, на выходе черного ящика получаем значение $f(x)$ для соответствующего значения аргумента. Квантовый алгоритм, который мы ниже опишем, решает данную задачу точно, используя только одно обращение к черному ящику. Очевидно, что классический алгоритм, который точно решает данную задачу во всех случаях, должен использовать $O(2^n)$ обращений к черному ящику по крайней мере в некоторых случаях. Отметим, что проблема Дойча есть частный случай данной задачи при $n = 1$. Опишем данный алгоритм.

Шаг 0. Исходное состояние $|0^n\rangle \otimes |1\rangle$.

Шаг 1. Применяем преобразование Уолша-Адамара на первом регистре и преобразование Адамара на втором регистре.

$$H_n \otimes H(|0^n\rangle \otimes |1\rangle) = 1/\sqrt{2^n}\sum_{i=0}^{2^n-1}|i\rangle \otimes 1/\sqrt{2}(|0\rangle - |1\rangle).$$

Шаг 2. Применяем преобразование U_f ко второму регистру.

$$U_f(1/\sqrt{2^n}\sum_x |x\rangle) \otimes 1/\sqrt{2}(|0\rangle - |1\rangle) =$$

$$(1/\sqrt{2^n}\sum_x (-1)^{f(x)}|x\rangle) \otimes 1/\sqrt{2}(|0\rangle - |1\rangle) =$$

$$|f\rangle \otimes 1/\sqrt{2}(|0\rangle - |1\rangle),$$

где $|f\rangle = 1/\sqrt{2^n}\sum_x (-1)^{f(x)}|x\rangle$. За один шаг применения U_f мы нашли значение функции f на всех 2^n входных наборах.

Шаг 3. Применяем преобразование Уолша-Адамара к первому регистру. Отметим, что

$$H_n|00\ldots0\rangle = 1/\sqrt{2^n}\sum_x |x\rangle$$

есть равная суперпозиция базисных состояний и она с точностью до знака совпадает с $|f\rangle$, если f – константа. Так как $H_n H_n = I$, то отсюда следует, что $H_n|f\rangle = \pm|00\ldots0\rangle$, если f – константа. Поэтому, если f – сбалансированная функция, то $H_n|f\rangle$ должно быть ортогонально $|00\ldots0\rangle$, то есть лежит в пространстве, натянутом на $\{|x\rangle : x \neq 0\ldots0\}$. Следовательно, чтобы отличить сбалансированную функцию от константной, мы применяем H_n к $|f\rangle$ и затем измеряем кубиты, чтобы увидеть, равны они все нулю или нет.

Шаг 4. Измерение.

5.5 Алгоритм Саймона

Проблема, решаемая алгоритмом Саймона состоит в следующем. Имеется n-битное ненулевое число a, которое определяется подпрограммой U_f. Целью является узнать значение a, используя как можно меньше вызовов подпрограммы. В проблеме Бернштейна-Вазирани классический компьютер должен использовать n вызовов подпрограммы, чтобы определить значение a, в то время как квантовому компьютеру необходимо только одно обращение к подпрограмме. Число вызовов подпрограммы возрастает линейно от n в классическом случае и является независимым от n в квантовом случае. В проблеме Саймона ускорение на квантовом компьютере значительно более существенно. На классическом компьютере количество раз, которое необходимо вызвать подпрограмму U_f, чтобы решить проблему Саймона, возрастает экспоненциально с ростом n, в то время как для квантового компьютера – линейно с ростом n.

Данное удивительное ускорение касается вероятностной характеристики многих квантовых алгоритмов. Такое ускорение на количество вызовов подпрограммы в зависимости от числа бит a мы имеем не для вычисления a точно, а для определения a с высокой вероятностью.

Подпрограмма U_f (черный ящик) в проблеме Саймона вычисляет два в один-функцию $f : \{0,1\}^n \to \{0,1\}^{n-1}$. Данная функция устроена таким образом, что $f(x) = f(y)$ тогда и только тогда, когда n-битные x и y связаны отношением $x = y \oplus a$, или, что то же самое $x \oplus y = a$, где $\oplus$ – операция побитного сложения по модулю 2. То есть $(x_1, \ldots, x_n) \oplus (y_1, \ldots, y_n) = (x_1 \oplus y_1, \ldots, x_n \oplus y_n)$.

Данную проблему часто рассматривают как проблему нахождения периода. Мы можем говорить, что функция f – периодичная относительно операции сложения по модулю 2,

$$f(x \oplus a) = f(x)$$

для всех x. Задача состоит в том, чтобы найти период a.

Именно поэтому проблема Саймона является предшественником алгоритма Шора — более искусного, более впечатляющего и более практически полезного алгоритма нахождения периода, который является сердцем его процедуры факторизации, где необходимо найти неизвестный период функции, которая является периодичной в обычном смысле: $f(x + a) = f(x)$.

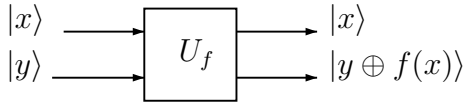
Для того чтобы найти значение a в проблеме Саймона на классическом компьютере, все, что нам надо, это выполнять подпрограмму U_f для списка различных входных значений x_1, x_2, x_3, ... до тех пор, пока мы не наткнемся на значение x_j, для которого значение функции $f(x_j)$ не совпадет с одним из вычисленных ранее значений $f(x_i)$. Тогда мы сможем определить $a = x_i \oplus x_j$. На любом шаге, предшествующем успеху, если мы выбрали m различных значений x, тогда все, что мы знаем, это что $a \neq x_i \oplus x_j$ для всех пар предыдущих значений x. То есть мы исключаем $\frac{1}{2}m(m-1)$ значений a. (Мы можем исключить и меньше значений a, если не будем заботиться, чтобы не выбрать очередной x равный $x_i \oplus x_j \oplus x_k$ для трех уже выбранных значений x). Так как существует $2^n - 1$ различных возможных значений a, наш шанс на успех не будет достаточно ощутимым, пока $\frac{1}{2}m(m-1)$ не будет хотя бы в малой степени сравнимым с 2^n. Мы маловероятно достигнем успеха, пока m не станет порядка $2^{n/2}$, поэтому количество раз, которое подпрограмма должна быть выполнена, чтобы получить достаточную вероятность для определения значения a, возрастает с ростом числа n бит как $2^{n/2}$, то есть экспоненциально. Если a имеет всего 100 бит, то классическому компьютеру понадобится примерно $2^{50} \approx 10^{15}$ раз выполнить подпрограмму, чтобы иметь достаточный шанс правильно определить a. При скорости 10 миллионов операций

в секунду этот процесс займет около трех лет.

В противоположность этому квантовый компьютер может определить a с высокой вероятностью (скажем, с вероятностью ошибиться один на миллион), выполняя подпрограмму не многим более чем n раз (около 120 раз, если a состоит из 100 бит). Этот замечательный трюк осуществляется с помощью следующей стратегии.

Как обычно, процедура U_f, выполняемая черным ящиком, является квантовым (унитарным) преобразованием, определяемым как

$$U_f : |x\rangle|y\rangle \to |x\rangle|y \oplus f(x)\rangle.$$



Пошагово изложим алгоритм Саймона.

Шаг 1. Строим равномерную суперпозицию всех 2^n возможных значений входного регистра:

$$(H_n \otimes H) : |00\ldots0\rangle|0\rangle \to \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|0\rangle.$$

Шаг 2. Применяем преобразование U_f

$$U_f : \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|0\rangle \to \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|f(x)\rangle$$

Шаг 3. Измеряем значение второго регистра и сохраняем результирующее состояние в регистре 1. Согласно определению функции f, состояние первого регистра будет вида

$$1/\sqrt{2}(|x_0\rangle + |x_0 \oplus a\rangle),$$

где x_0 выбрано равновероятно.

Заметим, что в силу определения функции в результирующей суперпозиции первого регистра будет только две ненулевых компоненты, так как $x_0 \oplus a \oplus a = x_0$. Эта суперпозиция содержит искомое значение a вместе с ненужной информацией о равновероятно выбранном x_0. Поэтому если мы просто измерим первый регистр, то получим в результате либо значение x_0, либо значение $x_0 \oplus a$ равновероятно, не получив никакой информации о значении a. Значение a, которое мы хотим узнать, проявляется лишь в отношении между этими двумя равновероятностными числами, только одно из которых мы можем получить в результате измерения. Если бы мы могли клонировать квантовое состояние, то, создав всего лишь 10 копий данного состояния, мы могли бы измерить каждую копию, и с вероятностью 0,998 узнали бы и x_0, и $x_0 \oplus a$, а значит и само a. Но, к сожалению, как мы видели ранее, квантовое состояние не может быть клонировано. Выполнение же данного алгоритма несколько раз не может привести к успеху, так как для каждого очередного запуска алгоритма после измерения на шаге 2 мы будем получать состояние $1/\sqrt{2}(|x_0\rangle + |x_0 \oplus a\rangle)$ для разных (вероятностных) значений x_0.

Однако, если, как в алгоритме Дойча, мы откажемся от получения точной информации относительно значений x_0 и $x_0 \oplus a$, мы можем получить частичную полезную информацию об их отношении (которая и представляет для нас интерес). Трюк который здесь используется, следующий.

Шаг 4. Применяем преобразование Уолша-Адамара к первому регистру.

$$\frac{1}{\sqrt{2^n}}(|x_0\rangle+|x_0\oplus a\rangle) \xrightarrow{H_n} \frac{1}{\sqrt{2^{n+1}}}\sum_y((-1)^{x_0\cdot y}+(-1)^{(x_0\oplus a)\cdot y})|y\rangle.$$

Так как $(-1)^{(x_0\oplus a)\cdot y} = (-1)^{x_0\cdot y}(-1)^{a\cdot y}$, то коэффициент при $|y\rangle$ равен 0, если $a\cdot y = 1$, и равен $2(-1)^{x_0\cdot y}$, если $a\cdot y = 0$. Поэтому результирующее состояние после шага 4 есть

$$\frac{1}{\sqrt{2^{n-1}}}\sum_{y:a\cdot y=0}(-1)^{x_0\cdot y}|y\rangle.$$

Заметим, что в результате действия H_n из состояния исчезла нежелательная информация об x_0, которая перешла в информацию об a, записанную как суперпозиция y таких, что $y\cdot a = 0$. Теперь измерение состояния позволит нам извлечь информацию об a. Тот же самый прием мы увидим позже в алгоритме Шора.

Шаг 5. Измеряем регистр и находим значение y (равновероятно) такое, что $y\cdot a = 0$.

Шаг 6. Повторяем весь описанный процесс достаточное количество раз, чтобы определить значение a, решая линейную систему уравнений

$$\begin{cases} y_1\cdot a &= & 0 \\ \dots & \dots & \dots \\ y_k\cdot a &= & 0 \end{cases}$$

Может быть показано что $O(n^2)$ повторений достаточно, чтобы определить a с любой заданной вероятностью $p < 1$.

Упражнение. Нарисуйте квантовую схему, реализующую алгоритм Саймона.

Глава 6

Квантовый поиск в неупорядоченной базе данных

6.1 Алгоритм Гровера

В 1996 году Лов Гровер разработал квантовый метод, который может быть применен к целому классу проблем, для которых решение трудно найти, но легко проверить (как для проблем из класса NP). Метод Гровера (и его различные обобщения) может быть использован для доказательства того, что квантовые вычислительные модели могут быть более эффективны для класса проблем, для которых известны нижние оценки сложности для классических вычислительных моделей.

Большой класс задач может быть легко переформулирован в терминах задачи поиска. Например,

- *Сортировка последовательности.* Дан n-элементный вектор A, найти перестановку π множества $\{1, \ldots, n\}$ такую, что $a_{\pi(i)} \leq a_{\pi(i+1)}$ для $i = 1, \ldots, n-1$.
- *Раскраска графа.* Дан граф (V, E) с n вершинами V с e ребрами $E \subseteq V \times V$ и множество из k красок C. Найти отображение $c : V \to C$ такое, что $\forall (v_1, v_2) \in E : c(v_1) \neq c(v_2)$.
- *NP-полная проблема SAT.*

Проблема, на решение которой направлен метод Гровера, может быть сформулирована следующим образом. Пусть дана неупорядоченная база данных (список) из N элементов, и пусть в ней существует один элемент, обладающий некоторым свойством (которое легко проверяется). Требуется найти этот элемент.

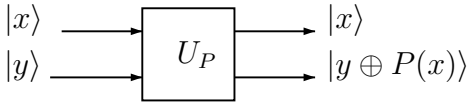
Например, представьте, что нам необходимо найти телефонный номер в телефонном справочнике, содержащем телефоны 1 000 000 абонентов. Мы легко можем найти в таком справочнике номер телефона, зная фамилию абонента. Предположим, что мы забыли фамилию абонента; вся информация, которой мы владеем, – это его адрес. В этом случае не остается ничего другого, как просматривать элементы справочника, пока мы не найдем нужный. В среднем нам понадобится просмотреть 500 000 посторонних людей, пока мы найдем того, кто нам нужен. В наихудший наш день нам понадобится просмотреть 999 999 из них. Компьютер может выполнить данный поиск гораздо быстрее, но алгоритмически это будет все то же самое. В среднем, требуется $N/2$ шагов, чтобы найти нужный элемент в списке из N элементов.

Итак, мы будем формулировать проблему поиска следующим образом в терминах экспоненциально большой неупорядоченной базы данных с $N = 2^n$ элементами, среди которых один элемент специальным образом промаркирован. Проблема состоит в том, чтобы найти этот элемент. Элементарная теория вероятностей показывает, что если мы просмотрим k элементов, то имеем вероятность k/N нахождения нужного нам элемента. Следовательно, необходимо сделать $O(N)$ запросов к базе, чтобы найти нужный элемент с любой константной (не зависящей от N) вероятностью. Алгоритм Гровера позволяет найти нужный элемент с вероятностью, достаточно близкой к 1 за $O(\sqrt{N})$ шагов (более точно, за $O(\sqrt{N})$ итераций выполнения процедуры, но за $O(\sqrt{N}\log N)$ шагов (где $\log N$ шагов необходимо для выполне-

ния преобразования Уолша-Адамара). Было показано, что ускорение в квадратный корень оптимально в контексте квантовых вычислений.

Проблема поиска может быть аккуратно переформулирована в терминах оракульной проблемы. Мы заменим базу данных на оракула (или черный ящик), который вычисляет предикат $P : \{0,1\}^n \to \{0,1\}$. Известно, что $P(x) = 0$ для всех x кроме одного, обозначим его x_0 ($P(x_0) = 1$). Наша проблема состоит в том, чтобы определить x_0. Как обычно, предполагаем, что f выполняется как унитарное преобразование на $n+1$ кубитах, определенное как

$$U_P : |x\rangle|y\rangle \to |x\rangle|y \oplus P(x)\rangle$$



Вычисление P за один шаг для всех возможных входов x_i посредством применения U_p к регистру, содержащему суперпозицию $1/\sqrt{2^n}\sum_x |x\rangle$ всех 2^n возможных значений входов x, вместе с начальным значением выходного регистра 0 приводит к суперпозиции

$$U_P : \frac{1}{\sqrt{2^n}}\sum_x |x\rangle|y\rangle \to \frac{1}{\sqrt{2^n}}\sum_x |x\rangle|P(x)\rangle$$

Здесь $|x\rangle$ – входной регистр, содержащий входное значение x, $|y\rangle$ – выходной регистр (изначально содержащий значение 0).

Основная трудность состоит в извлечении результата из данной суперпозиции. Для любого x_0, для которого $P(x_0) = 1$, состояние $|x_0\rangle|1\rangle$ есть часть суперпозиции $1/\sqrt{2^n}\sum_x |x\rangle|P(x)\rangle$,

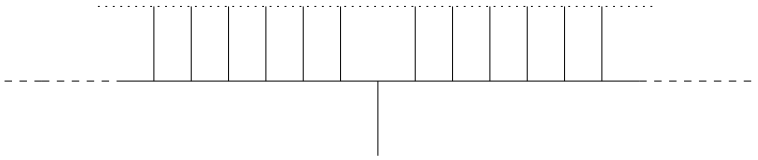
но, так как амплитуда этого состояния равна $1/\sqrt{2^n}$, то вероятность того, что при измерении мы получим результат x_0, равна 2^{-n}. Трюк состоит в изменении полученного квантового состояния $1/\sqrt{2^n}\sum_x |x\rangle|P(x)\rangle$ таким образом, чтобы значительно увеличить амплитуду состояния $|x_0\rangle|1\rangle$ (для которого предикат истинен) и уменьшить амплитуду состояний $|x\rangle|0\rangle$, для которых предикат ложен. Как только такое преобразование квантовой суперпозиции будет осуществлено, остается только провести измерение последнего кубита квантовой суперпозиции, который представляет $P(x)$. Так как амплитуда была изменена, вероятность того, что результат измерения будет 1, высока. Если это событие произойдет, измерение спроецирует состояние $1/\sqrt{2^n}\sum_x |x\rangle|P(x)\rangle$ в подпространство $1/\sqrt{k}\sum_{i=1}^{k} |x_i\rangle|1\rangle$, где k – число решений. Последующее измерение оставшихся кубитов выдаст одно из этих решений. Если в результате измерения кубита $P(x)$ получаем 0, весь процесс вычисления возобновляется вновь, и снова вычисляется суперпозиция $1/\sqrt{2^n}\sum_x |x\rangle|P(x)\rangle$.

Опишем детально алгоритм Гровера по шагам:

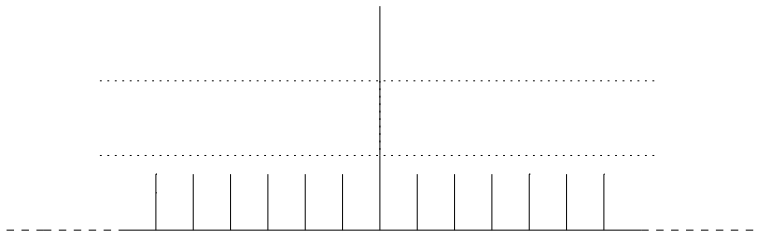
Шаг 1. Подготавливаем входной регистр, содержащий суперпозицию всех возможных значений $x_i \in \{0, \ldots 2^{n-1}\}$.

Шаг 2. Вычисляем $P(x_i)$ на этом регистре.

Шаг 3. Изменяем амплитуду a_i на $-a_i$ для тех значений x_i, для которых $P(x_i) = 1$. Эффективный алгоритм изменения знака обсудим позже.



Шаг 4. Применяем операцию “инверсия относительно среднего”, чтобы увеличить амплитуду тех значений x_i, для которых $P(x_i) = 1$. Квантовый алгоритм инверсии относительно среднего приведем ниже. Результирующие амплитуды выглядят следующим образом, где амплитуды x_i, для которых $P(x_i) = 0$, уменьшились.



Шаг 5. Повторяем шаги 2—4 $k = \pi/4\sqrt{2^n}$ раз.

Шаг 6. Считываем результат.

Было показано, что алгоритм Гровера оптимален с точностью до константного множителя. То есть не существует квантового алгоритма, который осуществляет неструктурированный поиск быстрее. Также было показано, что если решение x_0 единственно, то через $\frac{\pi}{8}\sqrt{2^n}$ итераций шагов 2—4 вероятность ошибки равна 0,5. После $\frac{\pi}{4}\sqrt{2^n}$ итераций вероятность ошибки уменьшается до 2^{-n}. Интересно, что дальнейшее продолжение итераций будет увеличивать вероятность ошибки. Для примера, после $\frac{\pi}{2}\sqrt{2^n}$ вероятность ошибки вновь станет близкой к 1.

Существует множество классических вероятностных алгоритмов, в которых процедура повторяется много раз для достижения лучшего результата. Повторение квантовой процедуры может улучшать результат некоторое время, но после достаточного количества повторений результат снова становится хуже.

Квантовая процедура — это унитарное преобразование, которое осуществляет поворот в комплексном пространстве. Поэтому, в то время как повторное применение квантового преобразования может поворачивать текущее состояние все ближе и ближе к нужному нам состоянию в течение какого-то времени, дальнейшее его применение заставит состояние пройти мимо нужного состояния и уходить все дальше и дальше от него. Поэтому, для того чтобы получить хороший результат при повторяющихся квантовых преобразованиях, очень важно знать, когда нужно остановиться.

6.2 Изменение знака

Вместо использования U_f мы будем использовать эквивалентную операцию на n кубитах. Она определяется следующим образом:

$$I_{x_0}|x\rangle = \begin{cases} |x\rangle & \text{if } x \neq x_0 \\ -|x_0\rangle & \text{if } x = x_0 \end{cases}$$

Для этого, как и в алгоритме Дойча, вместо значения 0 выходного регистра возьмем значение 1 и применим преобразование Адамара.

$$(H_n \otimes H)(|0^n\rangle \otimes |1\rangle) =$$
$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes 1/\sqrt{2}(|0\rangle - |1\rangle) =$$
$$\frac{1}{\sqrt{2^{n+1}}}(\sum_x |x\rangle|0\rangle - \sum_x |x\rangle|1\rangle).$$

Затем к полученной суперпозиции применяем преобразование U_f. В результате получаем

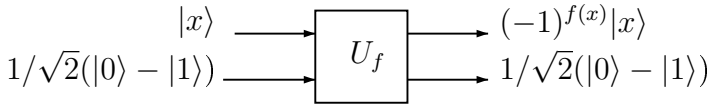
$$U_f(\frac{1}{\sqrt{2^{n+1}}}(\sum_x |x\rangle|0\rangle - \sum_x |x\rangle|1\rangle)) =$$

$$\frac{1}{\sqrt{2^{n+1}}}(\sum_{x\neq x_0}|x\rangle|0\rangle+|x_0\rangle|1\rangle-\sum_{x\neq x_0}|x\rangle|1\rangle-|x_0\rangle|1\rangle)=$$

$$\frac{1}{\sqrt{2^{n+1}}}(\sum_{x\neq x_0}|x\rangle(|0\rangle-|1\rangle)-|x_0\rangle(|0\rangle-|1\rangle))=$$

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x}(-1)^{f(x)}|x\rangle(|0\rangle-|1\rangle).$$

Как и в алгоритме Дойча, второй регистр носит вспомогательный характер. Подготовка выходного регистра в суперпозиции $1/\sqrt{2}(|0\rangle-|1\rangle)$ перед выполнением оператора черного ящика U_f используется, чтобы закодировать значение функции $f(x)$ в знаке $(-1)^{f(x)}$. Это часто используемый прием в квантовых вычислениях. Значение выходного регистра ограничивается только этим.



6.3 Инверсия относительно среднего

Чтобы осуществить инверсию относительно среднего на квантовом компьютере, инверсия должна быть унитарным преобразованием. Более того, чтобы алгоритм в целом решал проблему за $O(\sqrt{N})$ шагов, инверсия должна выполняться эффективно. Как мы покажем ниже, инверсия может быть выполнена с использованием $O(n)=O(\log(N))$ квантовых гейтов. Легко увидеть, что преобразование

$$\sum_i a_i|x_i\rangle\rightarrow\sum_i(2A-a_i)|x_i\rangle$$

описывается матрицей

$$D = \begin{pmatrix} 2/N-1 & 2/N & \dots & 2/N \\ 2/N & 2/N-1 & \dots & 2/N \\ \dots & \dots & \dots & \dots \\ 2/N & 2/N & \dots & 2/N-1 \end{pmatrix}$$

Так как $DD^\dagger = I$, преобразование D унитарно и, следовательно, является допустимым квантовым преобразованием. Обсудим теперь, как эффективно может быть выполнено данное преобразование, и покажем, что оно может быть разложено в $O(n) = O(\log N)$ квантовых гейтов. Мы можем определить D как $D = WRW$, где W – преобразование Уолша-Адамара, а преобразование R определено следующим образом

$$R = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & -1 & 0 & \dots \\ 0 & \dots & \dots & 0 \\ 0 & \dots & 0 & -1 \end{pmatrix}$$

Чтобы увидеть, что $D = WRW$, рассмотрим $R = R' - I$, где I – единичная матрица,

$$R' = \begin{pmatrix} 2 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots \\ 0 & \dots & \dots & 0 \\ 0 & \dots & 0 & 0 \end{pmatrix}$$

Теперь $WRW = W(R' - I)W = WR'W - I$. Легко проверить, что

$$WR'W = \begin{pmatrix} 2/N & 2/N & \dots & 2/N \\ 2/N & 2/N & 2/N & \dots \\ 2/N & \dots & \dots & 2/N \\ 2/N & \dots & 2/N & 2/N \end{pmatrix}$$

И поэтому $WR'W - I = D$.

Глава 7

Квантовый алгоритм факторизации чисел

7.1 Алгоритм RSA-шифрования

В данной главе мы опишем алгоритм RSA-шифрования, основанный на предположении о вычислительной трудоемкости задачи факторизации чисел. В следующем разделе будет приведен квантовый алгоритм Шора факторизации чисел, имеющий полиномиальную сложность.

7.1.1 Сведения из теории чисел

Основной алгебраический объект, лежащий в основе RSA-шифрования – это конечные группы с групповой операцией умножения по модулю некоторого фиксированного N. В арифметике по модулю N все числа, которые отличаются на множитель N считаются одинаковыми, таким образом, существует N различных значений, которые могут быть представлены как $0, 1, \ldots, N-1$. Для примера, $5 \times 6 = 2 \mod 7$ так как $5 \times 6 = 30 = 4 \times 7 + 2$. Мы будем писать $\equiv \mod N$, чтобы подчеркнуть, что равенство выполняется с точностью до множителя N.

Пусть G_N – множество всех положительных чисел, меньших N, не имеющих общих делителей с N. Так как разложение на простые множители единственно, произведение двух чисел из

G_N также принадлежит G_N. То есть G_N замкнуто относительно умножения по модулю N. Если $a, b, c \in G_N$ с условием $ab \equiv ac \mod N$, то $a(b-c)$ кратно N, и так как a не имеет общих делителей с N, то $b-c$ кратно N, значит, $b \equiv c \mod N$. Отсюда следует, что умножение элементов G_N на фиксированное a просто переставляет элементы в множестве G_N. Так как 1 принадлежит G_N, то должен существовать некоторый $d \in G_N$, такой, что $ad = 1 \mod N$, то есть каждый элемент a должен иметь обратный по умножению элемент. Все эти условия означают, что G_N является группой по умножению по модулю N.

Каждый элемент a группы G_N характеризуется своим **порядком** k — наименьшим целым числом, для которого

$$a^k \equiv 1 \mod N$$

Свойство 7.1 *Порядок каждого элемента группы G является делителем числа элементов в G (которое называется* **порядком группы** *G).*

Если p – простое, то группа G_p содержит $p-1$ элементов, так как никакое положительное число, меньшее p, не имеет общих делителей с p. Так как $p-1$ кратно порядку любого элемента a из G_p, то

$$a^{p-1} \equiv 1 \mod p.$$

Это отношение, известное как малая Теорема Ферма, распространяется на произвольные целые a, не делящие p, так как любое такое a представимо в виде $a = mp + a'$ с целым m и $a' < p$.

RSA-шифрование использует это расширение малой Теоремы Ферма на случай, характеризующийся двумя различными целыми p и q. Если целое a не делится ни на p, ни на q, то никакая

степень a также не делится ни на p, ни на q. В частности, a^{q-1} не делится на p, поэтому

$$(a^{q-1})^{p-1} \equiv 1 \mod p.$$

И по той же причине

$$(a^{p-1})^{q-1} \equiv 1 \mod q.$$

Последние соотношения означают, что существуют целые m, n, для которых

$$a^{(p-1)(q-1)} = 1 + mp$$
$$a^{(p-1)(q-1)} = 1 + nq$$

Отсюда следует, что $mp = nq$, что возможно при различных значениях p и q, только если m кратно q и, аналогично, n кратно p. Поэтому

$$a^{(p-1)(q-1)} = 1 + kpq$$

и поэтому

$$a^{(p-1)(q-1)} \equiv 1 \mod pq$$

для любого целого a не делящегося ни на p, ни на q.

Как альтернативное объяснение последнего равенства, отметим, что, так как a не делится ни на p, ни на q, то a не имеет общих делителей с pq и, следовательно, принадлежит группе G_{pq}. Число элементов в G_{pq} равно $pq-1-(p-1)-(q-1) = (p-1)(q-1)$, так как существует $pq-1$ элементов меньших pq, среди которых $p-1$ кратны q, $q-1$ кратны p. Следовательно, порядок $(p-1)(q-1)$ должен быть кратен порядку элемента a.

Преобразуем последнее равенство к виду, который используется в системе RSA. Для этого возьмем произвольную степень s и умножим обе части равенства на a:

$$a^{1+s(p-1)(q-1)} \equiv a \mod pq. \tag{7.1}$$

Отметим теперь, что если c не имеет общего делителя с $(p-1)(q-1)$, тогда c принадлежит группе $G_{(p-1)(q-1)}$, поэтому у него существует обратный элемент, то есть существует d такое, что

$$cd \equiv 1 \mod (p-1)(q-1)$$

Поэтому для некоторого целого s

$$cd = 1 + s(p-1)(q-1). \tag{7.2}$$

Объединяя равенства 7.1 и 7.2, имеем:

$$a^{cd} = a \mod pq \tag{7.3}$$

Поэтому если

$$b \equiv a^c \mod pq, \tag{7.4}$$

то

$$b^d \equiv a \mod pq. \tag{7.5}$$

7.1.2 RSA-шифрование

Предположим, Боб хочет получить сообщение от Алисы, закодированное таким образом, чтобы только он один мог прочитать его. Для этого он выбирает два больших (например, в 200 цифр) простых числа p и q. Боб пересылает Алисе по открытому каналу произведение этих чисел $N = pq$ и большое кодирующее число c, которое он выбирает таким образом, чтобы оно не имело общих делителей с $(p-1)(q-1)$. Поскольку числа p и q – большие, то практически невозможно факторизовать число N (состоящее из 400 цифр) с использованием обычного компьютера, поэтому невозможно ни Алисе, ни злоумышленнику Еве узнать p и q, зная только лишь N. Боб, однако, знает p, q и поэтому знает $(p-1)(q-1)$, следовательно, может найти обратный по умножению элемент d к c. Боб хранит c в строгом секрете.

Алиса кодирует сообщение, представляя его как строку из менее чем 400 цифр, используя некоторую кодировку, например ASCII. Если сообщение имеет большую длину, оно разбивается на несколько блоков. Алиса интерпретирует строку как число $a < N$. Используя кодовое значение c и значение $N = pq$, полученные от Боба, она вычисляет значение $b \equiv a^c \mod pq$ и передает его Бобу по открытому каналу. Когда Боб получает сообщение, он использует свое знание d, чтобы вычислить $b^d \mod pq$, что дает ему исходное Алисино сообщение a. Если бы Ева могла найти разложение N на p и q, то она могла бы декодировать d из открытого значения c так же, как это сделал Боб. Но факторизовать большие числа не позволяют ее вычислительные возможности.

7.2 Проблема факторизации числа

В рассмотренной проблеме Саймона мы имели дело с подпрограммой, вычисляющей функцию $f(x)$, которая удовлетворяла условию $f(x) = f(y)$ для различных x и y тогда и только тогда, когда $y = x \oplus a$, где $\oplus$ – операция побитного суммирования по модулю 2 n-битных чисел x и y. Число вызовов подпрограммы U_f для того, чтобы определить значение a для классического компьютера возрастает экспоненциально с ростом n, в то время как для квантового компьютера это возрастание только линейно. Данная проблема важна, поскольку демонстрирует замечательную мощность квантовых вычислений, но является достаточно искусственной. Проблема Саймона может быть сформулирована как задача нахождения неизвестного периода функции на n-битных числах, которая является "периодической" по модулю 2. Гораздо более естественной проблемой является проблема нахождения периода r функции, которая периодична в обычном смысле. Такая функция удовлетворяет условию: $f(x) = f(y)$ для различных x и y тогда и только тогда, когда x и y отличаются на множитель r. Нахождение периода такой периодической функции является ключевой процедурой в задаче факторизации – естественной математической проблеме с достаточно многочисленными практическим приложениями.

На первый взгляд может показаться, что нахождение периода периодической функции может быть достаточно легкой задачей, но так бывает только когда мы имеем дело с гладкой непрерывной периодической функцией (такой, например, как функция синус), чья структура на малом подмножестве точек внутри периода может дать достаточно информации о том, какой период у этой функции может быть. Наилучший тип периодической функции, который мы всегда далее будем иметь в виду, – это функция на целых числах, значения которой внутри периода

r полностью случайны и поэтому не дают даже намека на то, какой период у функции может быть. Наилучший из известных классических алгоритмов нахождения периода r таких периодических функций требует времени, которое возрастает экспоненциально от числа n количества бит r. Однако в 1994 году Питер Шор, вдохновленный работой Даниела Саймона, открыл, что можно использовать мощь квантовых вычислений, чтобы узнать период r с вероятностью, произвольно близкой к 1, за время, которое с ростом n растет немногим быстрее, чем n^3. Открытие Шора имеет важное практическое значение, поскольку умение эффективно находить период, вместе с некоторыми трюками из области теории чисел, позволит эффективно факторизовать число, которое является произведением двух больших простых чисел. Эта трудоемкая вычислительная процедура требуется во всех известных техниках факторизации, лежащих в основах систем безопасности, использующих RSA-метод шифрования с открытым ключом. (Название RSA возникло по именам создавших в 1977 году данный метод Ronald Rivest, Adi Shamir и Leonard Adelman. RSA-шифрование независимо было изобретено Clifford Cocks четыре года ранее, но его открытие было квалифицировано Британской Разведкой как высочайший секрет, и ему не было позволено открыть его первенство до 1977 года). Любой компьютер, который сможет эффективно находить период, поставит под угрозу безопасность как военных, так и коммерческих систем связей. Этим объясняется то, почему исследования в области возможности построения квантового компьютера — предмет значительного интереса в мире военного дела и бизнеса.

В 1994 году Шор придумал полиномиальный по времени алгоритм факторизации n-значного числа на квантовом компьютере, который решает данную проблему вероятностно с ограниченной ошибкой. С 1970-х годов продолжается поиск эффективного

алгоритма для факторизации чисел. Наилучший из известных на сегодняшний день алгоритмов - алгоритм Lenstra and Lenstra (1990), время работы которого – экспоненциально от размера входа. Под входом понимается последовательность цифр в записи факторизуемого числа M. Длина записи $n = \log M$. Хотя на сегодняшний день неизвестно, лежит ли данная проблема в классе P, твердая уверенность в несуществовании эффективного алгоритма для этой задачи имела следствием то, что безопасность многих криптографических систем (таких, как широко используемый RSA-алгоритм) оказалась зависящей от трудности решения данной задачи. Результат Шора поразил общественность, пробудив широкий интерес к квантовым вычислениям. Большинство алгоритмов факторизации, включая алгоритм Шора, используют стандартный прием сведения проблемы факторизации к проблеме нахождения периода функции. Шор использовал эффект квантового параллелизма стандартным образом, чтобы получить суперпозицию всех значений функции за один шаг. Затем он вычисляет квантовое преобразование Фурье функции. Последующее измерение с большой вероятностью позволяет извлечь период, который используется для факторизации числа M. Данное описание алгоритма очень упрощенное. Наибольшую трудность составляет квантовое преобразование Фурье, основанное на алгоритме быстрого преобразования Фурье. Опишем сначала алгоритм квантового преобразования Фурье, являюшийся основой алгоритма Шора.

7.3 Квантовое преобразование Фурье

Определение 7.1 *Комплексным корнем степени n из 1 называют такое комплексное число ω что*

$$\omega^n = 1$$

Имеется ровно n комплексных корней из 1, которые имеют

вид $e^{2\pi ik/n}$ для $k = 0, 1, \ldots, n-1$. Они равномерно распределены на окружности единичного радиуса с центром в нуле. Значение $e^{2\pi i/n}$ называется главным корнем степени n из 1.

Приведем свойства корней степени n из 1.

Свойство 7.2 *Для любых целых* $n \geq 0, k \geq 0, d > 0$

$$\omega_{dn}^{dk} = \omega_n^k.$$

Свойство 7.3 *Для любого четного* $n > 0$

$$\omega_n^{n/2} = w_2 = -1.$$

Свойство 7.4 *(Лемма о сложении) Для любого целого* $n \geq 1$ *и неотрицательного целого* k*, не кратного* n*, выполнено равенство*

$$\sum_{j=0}^{n-1} (\omega_n^k)^j = 0.$$

Определение 7.2 *Пусть* $g = (g_1, \ldots, g_{n-1})$ *– вектор с вещественными или комплексными компонентами. Дискретным преобразованием Фурье (ДПФ) вектора* g *называется вектор* G *длины* n *с комплексными компонентами, задаваемыми равенствами*

$$G_k = \sum_{j=0}^{n-1} \omega^{jk} g_k, k = 0, \ldots, n-1,$$

где ω *– комплексный корень степени* n *из 1.*

Иногда данное равенство записывается в матричном виде $G = Tg$, где матрица T имеет вид

$$T = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \omega & \omega^2 & \ldots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \ldots & \omega^{2(n-1)} \\ \vdots & & & & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \ldots & \omega^{(n-1)(n-1)} \end{pmatrix}$$

Квантовое преобразование Фурье – это вариант дискретного преобразования Фурье, областью значения которого являются равномерно распределенные на интервале $[0, 2\pi)$ точки $k\frac{2\pi}{N}$ для некоторого N. Масштабируя область определения на $\frac{N}{2\pi}$ получаем область значения квантового преобразования Фурье (КПФ) – числа между 0 и $N-1$.

Квантовое преобразование Фурье преобразует амплитуды квантового состояния следующим образом:

$$\sum_x g(x)|x\rangle \to \sum_c G(c)|c\rangle,$$

где $G(c)$ – дискретное преобразование Фурье вектора $g(x)$, x и c двоичные представления чисел, пробегающих различные значения от 0 до $N-1$. Если состояние будет измерено после преобразования Фурье, то вероятность того, что результат измерения будет $|c\rangle$, равна $|G(c)|^2$.

Преобразование Фурье — это общее преобразование из временной области в частотную. Так, преобразование Фурье преобразует функцию периода r в функцию, которая имеет ненулевые значения только на значениях, кратных частоте $1/r$. Поэтому, применяя квантовое преобразование Фурье к периодической функции $g(x)$ с периодом r, мы получим в результате $\sum_c G(c)|c\rangle$, где $G(c)$ равно нулю везде, кроме значений, кратных $\frac{N}{r}$. Поэтому, когда состояние будет измерено, результат измерения будет кратен $\frac{N}{r}$, скажем, $j\frac{N}{r}$.

Квантовое преобразование Фурье работает приближенно. Квантовое преобразование Фурье — это вариант быстрого преобразования Фурье, который работает для N – степени двойки, и дает приближенный результат для периода, который не является степенью двойки. Использование большей степени двойки в основании преобразования позволяет получать более точный результат.

Квантовое преобразование Фурье U_{QFT} с основанием 2^m определяется следующим образом

$$U_{QFT} : |x\rangle \to \frac{1}{\sqrt{2^m}} \sum_{c=0}^{2^m-1} e^{\frac{2\pi icx}{2^m}} |c\rangle.$$

Легко проверить, что U_{QFT} – унитарно. Для того чтобы алгоритм Шора был полиномиальным, необходимо, чтобы квантовое преобразование Фурье выполнялось эффективно. Шор показал, что квантовое преобразование Фурье с основанием 2^m может быть выполнено при помощи $\frac{m(m-1)}{2}$ гейтов. Знаменитый классический алгоритм быстрого преобразования Фурье требует времени, которое возрастает с ростом n как $n2^n$ (в то же время обычное вычисление требует времени $(2^n)^2$). Однако существует квантовый алгоритм выполнения унитарного преобразования U_{QFT} экпоненциально более быстрый, чем алгоритм быстрого преобразования Фурье. Как и во многих других квантовых алгоритмах, секрет здесь в том, что квантовый алгоритм не позволяет нам узнать значения всех Фурье-коэффициентов, как это происходит в алгоритме быстрого преобразования Фурье. Все что мы получаем, это суперпозицию, задаваемую преобразованием, и, как мы неоднократно повторяли, наличие такой суперпозиции не позволяет узнать действительные значения всех коэффициентов.

Отметим сходство квантового преобразования Фурье и преобразования Уолша-Адамара. Так как $-1 = e^{\pi i}$, то H_n можно записать как

$$H_n|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{\pi ixy} |y\rangle.$$

Не рассматривая множитель степени двойки перед суммой, коэффициент перед каждым состоянием $|y\rangle$ – это 1 или -1, опе-

рация xy – это операция обычного скалярного умножения. Поскольку операция обычного умножения xy более тонкая, чем операция $x \cdot y$ скалярного умножения по модулю 2, квантовое преобразование Фурье не может быть выполнено посредством только однокубитных гейтов, как преобразование Уолша-Адамара. Однако замечательно, что оно может быть выполнено посредством однокубитных и двухкубитных гейтов.

Для постоения схемы, выполняющей квантовое преобразование Фурье U_{QFT}, достаточно иметь n-кубитный унитарный оператор Z, диагональный в стандартном базисе

$$Z|y\rangle_n = e^{2\pi iy/2^n}|y\rangle_n.$$

С использованием этого преобразования, а также преобразования Уолша-Адамара U_{QFT} может быть выполнено как

$$U_{QFT}|x\rangle_n = Z^x H_n|0\rangle_n.$$

7.4 Алгоритм Шора факторизации числа N

Алгоритм Шора решает проблему факторизации числа N посредством решения эквивалентной проблемы: имея произвольное число a, взаимно простое с N, найти порядок r элемента $a \mod N$. Для этого выбирается a равновероятно. Используя алгоритм Евклида, можем эффективно определить, является ли a взаимно простым с N. Если a не является взаимно простым с N, то мы нашли делитель числа N. В противном случае порядок r элемента $a \mod N$ будет являться делителем числа N.

Определение 7.3 *Порядком элемента* $a \mod N$ *называется наименьшее целое* r *такое, что*

$$a^r \equiv 1 \mod N$$

Опишем алгоритм Шора.

Шаг 1. Квантовый параллелизм.

Произвольно выбираем число a. Если a не является взаимно простым с N, то мы нашли делитель N. В противном случае применяем следующий алгоритм.

Пусть n будет таково, что $N^2 \leq 2^n < 2N^2$. (Такой выбор n нужен, чтобы аппроксимация для значения периода, не равного степени двойки при преобразовании Фурье, была достаточной для оставшейся части алгоритма.) Используем квантовый параллелизм для вычисления функции $f(x) = a^x \mod N$ для всех значений x от 0 до $2^n - 1$. Результатом будет квантовое состояние

$$\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle |f(x)\rangle.$$

Шаг 2. Измеряем второй регистр.

Получаем равновероятно какое-то значение u. Само значение u нас не интересует. Результат измерения проецирует текущее состояние в подпространство, сопоставимое с результатом измерения. Поэтому после измерения текущая суперпозиция становится

$$\frac{1}{\sqrt{m}} \sum_{x} g(x)|x\rangle|u\rangle,$$

где m – нормализационный коэффициент (m равно количеству x: $f(x) = u$), $m \approx \frac{2^n}{r}$

$$g(x) = \begin{cases} 1, & \text{если } f(x) = u \\ 0, & \text{если } f(x) \neq u \end{cases}$$

Отметим, что значения x распределены в полученной суперпозиции равномерно (расстояние между соседними x равно

периоду r). Если бы мы могли измерить два соседних x в данной суперпозиции, мы получили бы период. К сожалению, законы квантовой механики позволяют получить только один результат измерения. Если же мы еще раз повторим весь процесс, то после первого измерения мы получим уже другой u, и, соответственно, второе измерение выдаст уже совсем другие x.

Шаг 3. Применение преобразования Фурье.

Второй регистр нас больше не интересует (он играл только вспомогательную роль), и можем больше его не рассматривать. Применяем преобразование Фурье к первому регистру (это аналог преобразования Уолша-Адамара в алгоритме Саймона):

$$U_{QFT} : \sum_x g(x)|x\rangle \to \sum_c G(c)|c\rangle$$

Запишем это в следующем виде

$$U_{QFT}\frac{1}{\sqrt{m}}\sum_{k=0}^{m-1}|x_0+kr\rangle = \frac{1}{\sqrt{2^n}}\sum_{y=0}^{2^n-1}\frac{1}{\sqrt{m}}e^{2\pi i(x_0+kr)y/2^n}|y\rangle =$$

$$= \frac{1}{\sqrt{2^n}}\sum_{y=0}^{2^n-1}e^{2\pi i x_0 y/2^n}\frac{1}{\sqrt{m}}\sum_{k=0}^{m-1}e^{2\pi i k r y/2^n}|y\rangle.$$

Расмотрим два случая.

1. Пусть r есть степень двойки.

   В этом случае, согласно стандартному Фурье анализу, ненулевые амплитуды будут иметь только состояния вида $|y\rangle = |j\frac{2^n}{r}\rangle$, то есть результирующая суперпозиция будет вида

$$C \sum_j z_j |j\frac{2^n}{r}\rangle,$$

где $|z_j| = 1$, $C = \sqrt{\frac{r}{2^n}} = \frac{1}{\sqrt{2^n/r}}$.

Действительно, рассмотрим амплитуду z_y при состоянии $|y\rangle$:

$$z_y = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{2\pi i k r y / 2^n}.$$

При $y = j\frac{2^n}{r}$

$$z_y = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{\frac{2\pi i k r j 2^n}{2^n r}} = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{2\pi i k j} = \sqrt{m}.$$

При $y \neq j\frac{2^n}{r}$

$$z_y = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{2\pi i k r y / 2^n} = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{2\pi i k y / (2^n/r)} =$$

$$\frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} e^{2\pi i k y / m} = 0,$$

где равенство нулю выполняется согласно Лемме о сложении.

2. Если r не является степенью двойки, то преобразование аппроксимирует точный результат таким образом, что наибольшие амплитуды сосредоточены в значениях $|y\rangle$ близких к $|j\frac{2^n}{r}\rangle$.

Шаг 4. Извлечение периода.

Измеряем состояние регистра в стандартном базисе и получаем результат v. Когда период есть степень двойки, его легко извлечь. В этом случае $v = j\frac{2^n}{r}$ для некоторого j. В

большинстве случаев j и r будут взаимно простые, и тогда сокращение дроби $\frac{v}{2^n}$ приведет к дроби, чей знаменатель будет значением r. В общем случае, когда r не есть степень двойки, квантовое преобразование Фурье дает только приближенное значение кратного $\frac{2^n}{r}$. При таком условии период извлекается посредством представления дроби в виде бесконечной последовательности цепных дробей.

Шаг 5. Нахождение делителя числа N.

Если найденный нами период r – четный, то используем алгоритм Евклида для определения, имеет ли $a^{r/2}+1$ или $a^{r/2}-1$ нетривиальный общий делитель с N. Если r – период функции $f(x) = a^x \mod N$, значит $a^r = 1 \mod N$, так как $a^r a^x = a^x \mod N$ для всех x. Если r четно, то

$$(a^{r/2}+1)(a^{r/2}-1) = 1 \mod N.$$

Поэтому если ни $a^{r/2}+1$, ни $a^{r/2}-1$ не кратно N, то либо $a^{r/2}+1$, либо $a^{r/2}-1$ имеет общий делитель с N.

Шаг 6. Повторение алгоритма, если необходимо.

Следующие события могут привести к тому, что мы не сможем получить делитель N.

1. значение v недостаточно близко к кратному $\frac{2^n}{r}$;
2. период r и множитель j не являются взаимно простыми, так что при сокращении дроби на шаге 4 мы получаем некоторый делитель периода, а не сам период r;
3. на шаге 5 мы получаем N как делитель N;
4. период функции $f(x) = a^x$ нечетный.

В этом случае повторяем алгоритм с самого начала.

Несколько повторений алгоритма (а именнно $O(\log N)$ раз) позволяют найти делитель N с любой заданной вероятностью $p < 1$.

7.5 Сведение задачи факторизации к задаче нахождения периода

Лемма 7.1 *Если существует полиномиальный (детерминированный, вероятностный, квантовый) алгоритм нахождения нетривиального ($x \not\equiv \pm 1 \mod N$) решения уравнения $x^2 \equiv 1 \mod N$, то существует полиномиальный (детерминированный, вероятностный, квантовый) алгоритм факторизации числа N.*

Доказательство: Пусть существует a такое, что $a^2 \equiv 1 \mod N$. Тогда

$$(a-1)(a+1) \equiv 0 \mod N.$$

Если N – не простое, то простой делитель числа N должен быть простым делителем либо $(a-1)$, либо $(a+1)$. Применяем алгоритм Евклида к $(a-1, N)$, $(a+1, N)$ для его нахождения и за $O(\log N)$ шагов получаем простой делитель числа N. □

Алгоритм нахождения делителя N.

1. Выбираем равновероятно значение y: $1 < y < N$.
2. Вычисляем НОД(y, N). Если НОД$(y, N) \neq 1$, то y – делитель числа N. В противном случае – продолжаем.
3. Находим период функции $y^k \mod N$.
4. Если период r – нечетный, или $y^{r/2} \equiv \pm 1 \mod N$, переходим на шаг 1. Иначе – КОНЕЦ.

Если алгоритм остановился, то $y^{r/2}$ – решение (нетривиальное) уравнения $x^2 \equiv 1 \mod N$.

Свойство 7.5 *Если существует полиномиальный (вероятностный, квантовый) алгоритм нахождения периода, то существует полиномиальный алгоритм факторизации числа.*

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Валиев В.А. Квантовые компьютеры: надежды и реальность / В.А. Валиев, А.А. Кокин. – Ижевск: НИС “Регулярная и хаотическая динамика”, 2001. – 352 с.

2. Китаев А. Классические и квантовые вычисления / А. Китаев, А. Шень, М. Вялый. – М.: МЦНМО, ЧеРО, 1999. – 192 с.

3. Манин Ю. Вычислимое и невычислимое / Ю. Манин. – М.: Сов. радио, 1982. – 128 с.

4. Feynman R. Simulating physics with computers / R. Feynman // International Journal of Theoretical Physics, 1982. – V. 21, N. 6,7. – P.467-488.

5. Simon D. On the power of quantum computation / D. Simon // SIAM Journal of Computing, 1977. – V. 26, N. 5. – P.1474-1483.

6. Grover L. A fast quantum mechanical algorithm for database search / L. Grover // Proc. of 28th STOC, 1996. – P.Philadelphia PA USA, 2996. – P. 212-219.

7. Gruska J. Quantum computing / J. Gruska. – McGraw-Hill Publishing Company, 1999. – 419 p.

8. Shor P. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer / P. Shor // SIAM J. on Computing, 1997. – V. 26, N 5. – P. 1484-1509.

9. Rieffel E., Polak W. An Introduction to Quantum Computing for Non-Physists [Электронный ресурс] / E. Rieffel, W. Polak // режим доступа: http:// xxx.lanl.gov / archive / quant-ph. - 1998. - quant-ph/9809016, проверено 15.05.2009.

10. Jozsa R. Quantum Algorithms and the Fourier Transforms [Электронный ресурс]/ Jozsa R. // режим доступа: http:// xxx.-lanl.gov / archive / quant-ph. - 1997. - quant-ph/9707033, проверено 15.05.2009.

11. Mermin D. Lecture Notes on Quantum Computation [Электронный ресурс]/ Mermin D. // http: // people.ccmr.cornell.edu / mermin / qcomp / SC483.html, проверено 15.05.2009.

12. Баюк Д. Двойная жизнь квантовых компьютеров / Д. Баюк //Что нового в науке и технике, 2006. – N 12. – С. 72-77.

ОСНОВЫ КВАНТОВЫХ ВЫЧИСЛЕНИЙ
Составитель
Гайнутдинова Аида Фаритовна
Учебное пособие

Редактор И.Г.Кондратьева

Подписано в печать 26.05.2009.
Форм. бум. 60 × 841/16. Гарнитура "Таймс". Печать ризографическая
Печ.л. 6,25. Тираж 50. Заказ 207.

Лаборатория оперативной полиграфии Издательства КГУ
420045, Казань, Кр.Позиция, 2а
Тел. 231-52-12