Квантовые вычисления

Учебно-методическое пособие

Ю.И.Ожигов

Москва, МГУ,
факультет ВМиК, 2003

Аннотация

Это пособие посвящено проблеме квантового компьютера. Дается точное определение квантового вычисления - абсолютного и с оракулом, описываются известные квантовые алгоритмы: Гровера, Шора, и ряд их обобщений и модификаций, а также приводятся некоторые нижние оценки для сложности квантовых вычислений. Кратко рассмотрен вопрос о коррекции квантовых ошибок и реалистические схемы квантового компьютера. Изложение доступно студентам первого-второго курсов естественных и технических специальностей университетов.

Предисловие

Эта книга посвящена проблеме сложности вычислений с точки зрения физики. Классическая теория алгоритмов, возникшая в 30-х годах двадцатого века (немного позднее квантовой теории), сейчас представляет вполне сформировавшуюся область, хорошо применимую к тем задачам, в которых мы можем пренебречь физической природой компьютера. Эта дисциплина удовлетворительно справлялась с абстрактным описанием вычислений в условиях действия закона Мура, согласно которому вычислительные мощности возрастали вдвое ежегодно. Это достигалось, естественно, почти экспоненциальным по времени уменьшением размеров элементарной вычислительной ячейки. Ясно, что такой закон не может действовать слишком долго, так как возможное уменьшение размеров лимитировано размерами атомов, для которых действуют уже не законы классической физики, а законы квантовой механики; это проявляется в "поправках"к закону Мура: например, увеличение "периода удвоения производительности"до двух лет и более. Таким образом, мы сталкиваемся с глобальным информационным вызовом, состоящим в принципиальной ограниченности доступных нам вычислительных ресурсов. Этот вызов остро ощущается в прикладных областях, связанных с нанотехнологиями - биохимии, наноэлектронике, новых материалах, фармацевтике и медицине, там где необходимы сложные компьютерные модели молекулярных процессов. Однако впервые с таким вызовом столкнулась физика. Здесь он проявляется в том, что размерность гильбертова пространства состояний квантовой системы растет экспоненциально с ростом числа входящих в нее частиц. Уже для молекулы аммиака эта размерность будет достигать астрономических величин, так что нет никакой надежды получить даже для движений и реакций молекул такой сложности компьютерную модель, учитывающую все взаимодействия, известные в физике.

Для решения этой глобальной проблемы есть два пути. Первый состоит в том, чтобы попытаться "спасти", или даже усилить действие закона Мура, используя новые для вычислительной практики физические принципы; этот путь сохраняет все возможности формализма квантовой физики, но предполагает изменение понятия алгоритма. Второй состоит в том, чтобы опираться только на классические принципы вычислений, но "урезать"квантовое описание физических процессов так, чтобы их можно было моделировать на классических компьютерах, например, рассматривать только амплитуды, принимающие дискретный набор значений с некоторой минимальной ненулевой разностью. До сих пор неясно, насколько существенным должно быть такое "урезание"абстрактного квантового формализма, чтобы сделать его практическим инструментом моделирования.

Одним из первых, кто ясно осознал информационный вызов, был Р.Фейнман, который и высказал идею о возможности создания полномасштабного квантового компьютера. Такой прибор, пока гипотетический, и является потенциальным решением этой проблемы по первому пути. Исследования в этом направлении представляют сейчас уже довольно обширную научную область, основные части которой мы и попытаемся изложить.

В первой главе очень схематично представлены элементы квантовой теории. Квантовая физика дала нам возможность детально объяснять и предсказывать явления микромира, которые были недоступны для физики классической, такие как строение атомов, молекул и взаимодействия элементарных частиц. Некоторые законы, как закон сохранения энергии или невозможность передачи информации со скоростью, превышающей световую, приобрели при этом всеобщий характер. Однако за переход к квантовой механике пришлось заплатить дорогую цену: пришлось отказаться от ряда принципов, на которых держалась физическая интуиция и которые обеспечивали (и до сих пор обеспечивают) многочисленные приложения. Среди таких принципов - локальность физических взаимодействий и возможность одновременного измерения произвольных величин, характеризующих рассматриваемую систему. Отказ от этих положений с необходимостью вытекает из матема-

тического аппарата квантовой теории, и воспринимается как неизбежное неудобство, с которым в общем можно примириться, так как это не приводит к логическим противоречиям. Гораздо хуже другое. Формальный аппарат квантовой физики не позволяет нам изучать поведение хоть сколько-нибудь сложных систем частиц даже с помощью численного моделирования на компьютерах. Это происходит из-за того, что размерность Гильбертова пространства состояний изучаемой системы растет экспоненциально с ростом числа частиц в ней. Все существующие численные методы, применяющиеся, скажем, в исследовании молекулярной динамики, по существу являются классическими. Существующие квантовые расчеты охватывают не более 3-4 частиц, и у нас нет никакой возможности достоверной оценки влияния квантовых эффектов на поведение сложных систем, которые представляют реальный интерес в приложениях - нанотехнологиях и биохимии. Этот информационный барьер вытекает из квантового формализма, и приближение к нему дает возможность выяснять скрытые возможности и границы применимости самого формального аппарата, а также пути его развития для приложений к технологически новым областям. Одна из таких областей - квантовый компьютер. Этот, пока гипотетический, прибор дает возможность использовать квантовый формализм для принципиального ускорения классических вычислений. Квантовое ускорение имеет место для широкого круга алгоритмических задач, таких как задача перебора, дискретной оптимизации, поиска собственных значений, распознавание молекулярных и нано-структур и решения уравнений математической физики. Я выбрал некоторые, с моей точки зрения наиболее типичные задачи из этого списка для подробного разбора во второй главе, которая начинается с краткого и очень схематичного введения в классическую теорию алгоритмов. В третьей главе рассматриваются ограничения на эффективность квантовых алгоритмов, которые исходят из самого формального аппарата квантовой теории. Здесь мы увидим, что с абстрактной точки зрения квантовое ускорение - это своеобразное произведение искусства, которое встречается не так часто. Очертив таким образом внутренние границы возможностей квантового компьютера, мы сможем яснее понять его замысел и возможности.

Проблема построения квантового компьютера рассматривается в книге как фундаментальная проблема физики. Она сводится к вопросу о том, какой вычислительный ресурс для исследования Природы мы можем теоретически использовать. И если в данной книге мы изучаем эту проблему в "оптимистическом"аспекте, то есть предполагая возможность создания масштабируемого квантового компьютера, то в принципе существует и альтернативная возможность, то есть ситуация, когда возможен только ограниченно - масштабируемый квантовый компьютер, то есть классический вычислительный ресурс, находящийся в нашем распоряжении, в принципе ограничен. Этот путь значительно менее разработан. Вычислительные задачи квантовой физики традиционно считаются слишком сложными для того, чтобы пытаться решать их даже на существующих суперкомпьютерах. Одна из причин такой традиции лежит в истории. Квантовая механика появилась в самом начале 20 века, тогда как теория алгоритмов возникла только в 30-х годах. Эта 30-летняя задержка, по-видимому, и ответственна за то, что мощь классического алгоритмического подхода осознается научной общественностью далеко не в той же мере, как для квантовой теории. Разработки эффективных компьютерных моделей для квантовых явлений представляет интереснейшую задачу не только для физиков, но и для программистов и математиков. Читатель, интересующийся этим направлением, может обратиться, например, к работе [Ozh], а также к web-ресурсам.

Важнейшими областями квантовой информатики, отсутствующими в этой книге, являются теория квантовых каналов и квантовая криптография, а также квантовая энтропия, это читатель может найти в ряде других руководств (см., например, [Ho]). Приведенный в конце книги список литературы не претендует на полноту, тем более что по этой теме постоянно появляются новые интересные работы. Укажем на сайт http://xxx.lanl.gov , раздел quant-ph как на основное храни-

лище новых статей по этой проблематике. Ряд интересных ссылок можно найти также в разделе web-ресурсов.

Глава 1

Элементы квантовой механики

1.1 Амплитудное описание систем элементарных частиц. Кубиты.

Вся квантовая механика основана на принципе интерференции, или наложении амплитуд физических величин. Этот принцип имеет волновую природу, так ведут себя реальные волны - например волны на воде, или радиоволны, так что квантовую механику называли также волновой. Опыты по интерференции света делал еще Ньютон, так что это свойство было известно давно, но только в начале 20 века обнаружилось, что для правильного описания всех микрообъектов необходимо применять волновые представления, иными словами, любые частицы (даже больших размеров) обладают волновыми свойствами. Кроме этого, изменение во времени состояний микрообъектов подчиняется линейным законам. Для описания таких состояний и свойств удобнее всего использовать язык комплексных Гильбертовых пространств. Вообще, почти все в теоретической физике определяется возможностями используемого математического аппарата, который в принципе нельзя отделить от собственно физики. Этот аппарат содержит в себе ряд допущений и игнорирует целый ряд деталей, которые в момент появления квантовой механики считались несущественными. Например, представление о потенциальной бесконечности пространства-времени и о его неограниченной делимости. Таким образом, не следует пытаться в рамках этого аппарата ответить на все возникающие разумные вопросы, как и придавать слишком большое значение тем его формальным следствиям, которые все равно никогда не удастся проверить на практике. У нас же сейчас нет лучшего выбора, кроме как принять этот аппарат и постараться изучить его основы наиболее простым образом.

В квантовой физике состояние частицы характеризуется так называемой волновой функцией Ψ которая принимает комплексные значения, называемые амплитудами. Аргументом волновой функции является время, а также некоторый набор физических параметров, например, координаты этой частицы, или координаты и ее энергия, или ее вектор импульса и энергия, или внутренний момент импульса (спин) п.д. Эти параметры неравноправны. Например, в физике считается, что координаты могут принимать любые вещественные значения, кроме оговоренных в условии задачи, тогда как значения энергии, вообще говоря, принимают значения из некоторого спектра, который определяется при решении волнового уравнения, и часто может быть дискретным. Физический смысл амплитуд сложен, его обсуждение мы пока отложим. Заметим только, что природа амплитуд ради-

кально отличается от знакомых вам параметров - аргументов волновой функции, а именно, имея в распоряжении всего лишь один экземпляр частицы, можно путем измерения узнать значения любого параметра - аргумента, но никак невозможно узнать ее амплитуду !

В теории алгоритмов принято оперировать с конечными объектами, описание которых в принципе умещается в память компьютера. Так что при изучении квантовой теории вычислений нам придется перейти от принятых в теорфизике бесконечномерных моделей к конечномерным. Это можно сделать по-разному, в зависимости от того, какой из параметров мы будем делать дискретным. Поскольку физический смысл амплитуд для нас пока туманен, проще всего считать дискретными те величины, которые можно измерить в одном эксперименте, а это как раз аргументы волновой функции: время, координаты, импульсы, спины и т.д. Заметим, что вопрос о том, существует ли минимальное значение времени, координаты, импульса или энергии открыт, так что может случиться, что такая дискретизация как раз отражает реальную картину микромира.

Покажем, как при этой дискретизации возникает понятие квантового бита или кубита. Для простоты рассмотрим случай, когда волновая функция $\Psi(x,t)$ зависит только от координаты x и времени t, которое мы пока зафиксируем, чтобы разобраться с зависимостью от координаты. Пусть сначала x может принимать только два значения, скажем 0 и 1. Это соответствует случаю, когда частица может находиться только в двух различных точках. Можно посмотреть на этот случай иначе. Предположим, что в нашем распоряжении имеется только один бит для хранения информации о местонахождении частицы, которая в действительности находится где-то на отрезке $[0,1]$. Тогда мы примем $x=0$ если она находится в левой половине этого отрезка, и $x=0$, если в правой. Рассмотрим две вспомогательные функции - $|0\rangle$ и $|1\rangle$. Первая равна единице при $x=0$ и нулю при $x=1$, а вторая наоборот. Тогда нетрудно видеть что любую функцию $\Psi(x)$ можно единственным образом записать в форме $\lambda_0|0\rangle+\lambda_1|1\rangle$. Эта запись в точности соответствует разложению вектора двумерного комплексного пространства по базису $|0\rangle$, $|1\rangle$. Если к тому же считать эти базисные вектора ортогональными и единичной длины, у нас получится, что волновая функция есть вектор двумерного комплексного пространства с выделенным ортонормированным базисом. Такую частицу мы будем называть кубитом.

Теперь мы можем поговорить о физическом смысле амплитуд λ_0 и λ_1. Здесь мы должны затронуть очень серьезный вопрос, решения которого, удовлетворительного во всех отношениях, пока не найдено. Дело в том, что единственный способ для нас узнать что-либо о том, в какой же точке находится наша частица в данный момент времени - это измерить ее волновую функцию. Измерение даст нам любой из векторов $|0\rangle$, $|1\rangle$ - каждый с вероятностью $|\lambda_0|^2$ или $|\lambda_2|^2$ соответственно. Итак, физический смысл амплитуд состоит в следующем:

Если взять квадрат модуля амплитуды при заданном значении параметра-аргумента, то получится вероятность того, что этот аргумент примет заданное значение

Это означает, что если мы можем каким-то образом получать все новые и новые частицы, находящиеся в состоянии Ψ, и производим последовательно над каждой из них по одному измерению, записывая полученный результат, то доля полученных результатов $|j\rangle$ при фиксированном $j=0,1$ будет все более и более точно приближаться к $|\lambda_j|^2$ с ростом числа измерений. Частица оказывается как бы размытой по двум точкам, так что с уверенностью нельзя сказать, где же она, собственно, в данный момент находится. Тогда мы должны наложить на амплитуды следующее условие нормировки:

$$|\lambda_0|^2+|\lambda_1|^2=1 \tag{1.1}$$

выражающее тот факт, что суммарная вероятность должна быть равна единице. Итак, измерение (его еще называют наблюдением) - это вероятностный процесс, исход которого никак не может быть

предсказан в рамках квантовой механики. Таких процессов не существует в классической физике. Случайность там проистекает только из-за того, что мы не можем точно определить всех параметров процесса или учет таких параметров слишком сложен. Однако в принципе, зная все такие "скрытые"параметры, мы бы могли, если очень захотели, определить исход классического измерения. Такие процессы называют еще псевдослучайными, и другой случайности в классической физике нет. Классическая физика, как говорят, детерминистична. В квантовом же мире мы встречаемся с истинной случайностью, поскольку мы не просто не знаем скрытых параметров, которые бы определили результат наблюдения - при некоторых условиях, которые довольно естественно было бы наложить на параметры таких параметров просто не может быть. Этому удивительному утверждению можно придать более строгую форму. Таким образом, квантовый формализм не может в принципе ответить на очень простой вопрос: что же в точности происходит с системой, если мы (или кто-нибудь другой) ее понаблюдаем. Мы умеем только вычислять вероятности получения при наблюдении того или иного результата. Увы, нам придется смириться с этим довольно неприятным открытием, чтобы пойти дальше.

Теперь рассмотрим более сложную ситуацию, когда у нас имеется два бита для хранения информации о положении частицы. Это соответствует ситуации, когда мы сначала делим отрезок $[0, 1]$ пополам, используя первый бит для определения в какой из половинок находится наша частица, а затем каждую из половинок - снова пополам, и используем второй бит для аналогичной локализации частицы внутри отрезка длины 0.5. Теперь у нашей волновой функции будет всего четыре значения аргументов, обозначим их так: $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$. Можно считать, что наша частица может находиться в четырех точках, соответствующих разбиению первоначального отрезка сначала пополам, а потом каждую из половин - еще раз пополам. Тогда волновая функция будет иметь такой вид:

$$\Psi = \lambda_0|00\rangle + \lambda_1|01\rangle + \lambda_2|10\rangle + \lambda_3|11\rangle. \tag{1.2}$$

В результате измерения этого состояния мы получим любое из значений $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ - каждое с соответствующей вероятностью: $|\lambda_0|^2$, $|\lambda_1|^2$, $|\lambda_2|^2$, $|\lambda_3|^2$. Условие нормировки при этом выглядит так:

$$|\lambda_0|^2 + |\lambda_1|^2 + |\lambda_2|^2 + |\lambda_3|^2 = 1. \tag{1.3}$$

Мы видим, что если как и прежде, считать $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ ортонормированным базисом в четырехмерном комплексном пространстве, то волновая функция будет вектором этого пространства единичной длины.

Теперь взглянем на эту математику немного с другой стороны. Представим себе, что у нас есть ансамбль из двух частиц, каждая из которых существует на своем отрезке длиной 1, и всего два бита для описания их положения - по биту на частицу. Тогда положение каждой из частиц мы сможем описать только с точностью 0.5. Будем трактовать наши базисные состояния $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ так: обе частицы в левых половинках своих отрезков, первая в левой а вторая в правой, первая в правой а вторая в левой, обе в правых. Тогда волновая функция 1.3 будет состоянием нашего ансамбля из пары частиц. Измерение теперь даст нам координаты каждой из частиц.

Этот процесс можно обобщить на n частиц. Но от такого обобщения будет не много пользы. В случае n частичных ансамблей у нас будут получаться пространства размерности 2^n, и с ростом n эта размерность будет расти так, что даже для небольших по меркам химии молекул, состоящих из нескольких десятков атомов для полного описания их волновых функций не хватит даже памяти суперкомпьютеров. В действительности эта трудность - фундаментальной природы. Именно из-за этого методы точных наук - физики и математики не проникли в биологию, имеющую дело с гигантскими молекулами.

Однако, вернемся к нашему двухчастичному ансамблю. Можно заметить один недостаток предложенного описания состояний. Обе частицы в этом описании как бы неразделимы - мы не рассматриваем их по отдельности. Немного подумав, мы придем к выводу, что это как раз и приводит к экспоненциальному росту размерности пространства для описания состояний. Что если попытаться получить описание ансамбля естественным путем, через некую комбинацию состояний отдельных частиц? Здесь мы придем к понятию тензорного произведения пространств. Это определение получается, если прямо обобщить уже имеющуюся у нас конструкцию на случай произвольных линейных пространств. Пусть H_1 и H_2 - два линейных пространства с базисами $e_1, e_2, \ldots, e_k$ и $h_1, e_2, \ldots, h_s$ соответственно. Образуем из них новое пространство, базисом которого будем считать такие формальные произведения: $e_i \bigotimes e_j$, где $i = 1, 2, \ldots, k$, $j = 1, 2, \ldots, s$, которые мы будем обозначать, по аналогии с предыдущим, через : $|e_i e_j\rangle$, просто опуская знак тензорного произведения. Обозначим это пространство через $H_1 \bigotimes H_2$. Мы видим, что при такой процедуре размерности исходных пространств перемножаются, а не складываются, как было бы в случае декартова произведения. Если первоначальные пространства были Гильбертовыми и их базисы были ортонормированными, то образованный из них базис произведения тоже будем считать ортонормированным базисом в Гильбертовом пространстве $H_1 \bigotimes H_2$. Всегда можно считать, что первое пространство соответствует состояниям первой частицы, а второе - второй. Имея такое определение естественно было бы определить тензорное произведение произвольных состояний $\Psi_1 \in H_1$ и $\Psi_2 \in H_2$ обеих частиц. Будем считать, что операция $\bigotimes$ обладает свойством дистрибутивности по отношению к умножению, то есть можно раскрывать скобки по обычным правилам, нельзя только переставлять сомножители. Тогда, например, $(\lambda_0|0\rangle + \lambda_1|1\rangle) \bigotimes (\mu_0|0\rangle + \mu_1|1\rangle) = \lambda_0\mu_0|00\rangle + \lambda_0\mu_1|01\rangle + \lambda_1\mu_0|10\rangle + \lambda_1\mu_1|11\rangle$.

Ясно, что если бы любой вектор из произведения пространств представлялся в виде произведения векторов из этих пространств, то мы сумели бы обойти трудность, связанную с экспоненциальным ростом размерности. Действительно, тогда для хранения информации о состоянии ансамбля, то есть вектора в пространстве - произведении достаточно было бы хранить информацию о каждом из сомножителей, а значит, рост числа необходимых бит был бы линейным, и мы смогли бы хранить состояния квантовых ансамблей в памяти обыкновенного компьютера. Однако эта надежда тщетна - Вы можете сами убедиться в том, что, например, состояние $|00\rangle + |11\rangle$ нельзя представить в виде произведения однокубитных состояний. Хуже того: ряд экспериментов, проведенных в последнее время показывает, что такие состояния (их еще называют ЭПР - пары, Белловские состояния или кошки Шредингера) действительно существуют в природе, причем для самых различных частиц, как массивных так и безмассовых. Это окончательно хоронит надежды на то, что нам удастся избежать рассмотрения пространств экспоненциальной размерности, и надо набраться терпения для того, чтобы их изучать.

Итак, состояние физической системы описывается вектором Гильбертова пространства H единичной длины с ортонормированным базисом $|e_1\rangle, |e_2\rangle, \ldots$, вообще говоря, бесконечномерного и даже континуальной размерности (это как раз та самая абстракция, диктуемая математическим аппаратом). Физический смысл базисных векторов $|e_j\rangle$ - возможные значения какого-либо параметра, который можно измерить. Например, это могут быть значения координат материальных точек. Или это может быть количество частиц, находящихся в определенных более элементарных одночастичных состояниях (это бывает в представлении вторичного квантования), и т.д. Самый общий вид состояния нашей системы имеет вид

$$|\xi\rangle = \sum_j \lambda_j |e_j\rangle, \tag{1.4}$$

где λ_j - комплексные числа, называемые амплитудами. Таким образом, волновое представление

состояния означает, что система может одновременно находиться во многих состояниях, но при этом в каждом - с соответствующей амплитудой. После измерения это состояние превращается в одно из $|e_j\rangle$ - в каждое с соответствующей вероятностью $|\lambda_j|^2$.

Установим следующие постоянные обозначения. Пусть $\bar{a} \in H$, - - вектор из Гильбертова пространства состояний. Через $|\bar{a}\rangle$ будем обозначать столбец его координат в заранее выбранном базисе $\{e_j\}$, который мы всегда будем предполагать ортонормированным. Через $\langle\bar{a}|$ будем обозначать строку, полученную из $|\bar{a}\rangle$ транспонированием и комплексным сопряжением. Тогда можно рассмотреть скалярное произведение векторов $\bar{a}$ и $\bar{b}$, записав его в виде матричного произведения $\langle\bar{a}| \cdot |\bar{b}\rangle = \sum\limits_j \bar{\lambda}_j^a \lambda_j^a$, где λ_j обозначают координаты векторов $\bar{a}$ и $\bar{b}$, а черта сверху - комплексное сопряжение. Опуская знак скалярного произведения и объединяя вертикальные черточки, мы будем записывать это просто через $\langle\bar{a}|\bar{b}\rangle$. Например, квадрат длины вектора $\bar{a}$ будет иметь вид $\langle\bar{a}|\bar{a}\rangle$. Однако мы можем перемножить строку и столбец и в обратном порядке. Посмотрим, к чему это приведет для одного вектора $\bar{a}$: $|\bar{a}\rangle\langle\bar{a}| = \rho_a$. Это будет матрица размера N×N, где N- размерность пространства H. Эта матрица называется матрицей плотности состояния α и полностью характеризует все его физические свойства. Поскольку на ее главной диагонали стоят вероятности, ее след равен единице. Продемонстрируем удобство наших обозначений, вычислив квадрат этой матрицы: $\rho^2 = |\bar{a}\rangle\langle\bar{a}|\bar{a}\rangle\langle\bar{a}| = \rho$, поскольку вектор состояния $\bar{a}$ единичной длины, и $\langle\bar{a}|\bar{a}\langle = 1$. Таким образом, квадрат матрицы плотности совпадает с ней самой. Из ее определения также следует, что она совпадает со своей сопряженной, т.е. эрмитова.

Заметим, что до сих пор мы считали фиксированным базис $\{|e_j\rangle\}$, полученный с помощью вышеописанной процедуры определения декартовых координат частиц с помощью кубитов. Однако в Гильбертовом пространстве состояний существует не один, а бесконечное множество различный ортонормированных базисов. Будет ли иметь физический смысл выбор какого-либо иного базиса $\{|e'_j\rangle\}$ в том же самом пространстве состояний? Оказывается, да. Например, можно выбрать базис с помощью аналогичной процедуры, но примененной не к координатам наших частиц, а к их импульсам! Можно также выбирать и другие базисы, имеющие физический смысл, но этот смысл будет уже не столь прозрачен, как в перечисленных двух случаях. Здесь уместно напомнить, что мы сделали очень серьезное предположение, а именно, мы предположили, что Гильбертово пространство состояний служит адекватным описанием физики. Надо сказать, что пока мы не привели еще серьезных доводов в пользу этого предположения, поскольку формализм Гильбертовых пространств нами никак не связывался с экспериментами. Чтобы подкрепить это наше предположение, прежде всего придется рассмотреть ситуацию, когда мы производим измерение состояния Ψ нашего ансамбля в другом базисе $\{|e'_j\rangle\}$. Если наше предположение верно, то вероятности получения при измерении всякого $|e'_j\rangle$ должна составить $|\langle e'_j|\Psi\rangle|^2$. Эксперимент полностью подтверждает это. Таким образом, будет совершенно законным рассматривать разложение исходного состояния по разным базисам:

$$| \Psi\rangle = \sum_j \lambda_j |e_j\rangle, \ | \Psi\rangle = \sum_j \lambda'_j |e'_j\rangle, \tag{1.5}$$

где в любом случае амплитуды можно найти из простого соотношения: $\lambda_j = \langle e_j \mid \Psi\rangle$, для получения которого надо умножить первое из приведенных разложений на $\langle e_j \mid$ слева.

Итак, мы видим, что процедура физического измерения связана обязательно с некоторым базисом, и у него есть прозрачный геометрический смысл: это проекции вектора измеряемого состояния на соответствующие координатные оси.

Мы обосновали корректность использования Гильбертовых пространств. Однако все еще их использование выглядит необоснованным усложнением, ибо мы еще фактически никак не применяли

сложение векторов. В следующем параграфе мы увидим, что Гильбертово пространство - абсолютно адекватная квантовой механике структура, поскольку амплитуды меняются со временем по линейным законам.

1.2 Интерференция амплитуд - основной закон квантовой механики. Волновое уравнение

Теперь мы приступаем к изучению особенно интересного вопроса: как меняются амплитуды состояний квантовых ансамблей во времени? Пусть Ψ_0 - вектор состояния нашего ансамбля в нулевой момент, и U_{t_1} - оператор временной эволюции, переводящий это состояние в состояние Ψ_1 той же самой системы в момент t_1. Надо сказать, что здесь мы делаем еще одно важное допущение: считаем, что наше Гильбертово пространство состояний не меняется со временем. Это предположение для описанного в предыдущей секции "координатного"базиса представляется совершенно естественным. Однако есть ситуации, когда это не совсем прозрачно. Например, рассмотрим процесс прохождения кванта света - фотона через тонкую поляризационную пластинку. Любой фотон может либо пройти через нее, либо поглотиться в ней. Пренебрежем пространственной составляющей состояния фотона, а будем рассматривать лишь состояние его поляризации $|\Psi_{pol}\rangle$. Это состояние можно представлять как вектор в двумерном комплексном пространстве, ортогональном направлению его движения. Тогда у нас получится $|\Psi_{pol}\rangle = \lambda_{ver}|e_{ver}\rangle + \lambda_{hor}|e_{hor}\rangle$, где $|e_{ver}\rangle$ и $|e_{hor}\rangle$ - ортонормированный базис, состоящий из состояния с вертикальной поляризацией и горизонтальной соответственно. А поляризационная пластинка пропускает только фотоны с определенным направлением поляризации, например, горизонтальным и поглощает те, что имеют вертикальную поляризацию. Таким образом, можно трактовать прохождение (или непрохождение) фотона через эту пластинку как измерение состояния его поляризации. Однако фотон всегда движется, и пространства его состояния до и после прохождения пластинки, вообще говоря, различны. Мы однако отождествляем эти пространства и их базисы с помощью параллельного переноса в направлении движения фотона на толщину пластинки. То есть считаем, что и после прохождения пластинки вертикальное и горизонтальное направления имеют тот же смысл. Тогда эксперимент показывает, что вероятность фотона пройти через пластинку равна $|\langle\Psi_{pol} \mid e_{gor}\rangle|^2$ и совпадает с квадратом косинуса угла между вектором поляризации фотона и горизонтальной осью в пространстве поляризации. Значит, амплитуды λ_{gor} и λ_{ver} можно трактовать как направляющие косинусы вектора поляризации фотона в пространстве поляризации. Здесь мы наблюдаем интересную вещь: вещественная часть Гильбертова пространства квантовых состояний совпадает с декартовым пространством, в котором разворачивается эксперимент - такое бывает не часто.

Вернемся к временному закону эволюции U_t. Если первоначальное сстояние было $|\Psi\rangle$, то через время t мы получим состояние - результат действия оператора U_t на $|\Psi\rangle$: U_t $|||Psi\rangle$. Приготавливая новые и новые экземпляры ансамблей в состоянии $|\Psi\rangle$, и измеряя это состояние через время t, мы можем установить модули амплитуд выходного состояния $\langle e_j \mid U_t\Psi\rangle$, и, проделывая эту процедуру для разных Ψ, проверить тем самым гипотезу о линейности U_t. Это и было в принципе сделано для самых различных квантовых ансамблей, причем линейность оператора U_t полностью подтвердилась. Итак, оператор временной эволюции квантовой системы линеен и сохранает длины единичных векторов. Следовательно, он сохраняет длины всех векторов, и является унитарным, то есть его обратный совпадает с его сопряженным (под сопряжением мы, как обычно, понимаем транспонирование и комплексное сопряжение). Применяя теорему о связи унитарных и эрмитовых

операторов, мы получаем, что существует такой, зависящий от времени эрмитов оператор H', что $U_t = e^{-iH'(t)}$. Уточним зависимость этого оператора от времени. Представив временную ось как последовательность малых интервалов длины δt, мы имеем:

$$U_t \approx (U_{\delta t})^{t/\delta t} = (e^{-iH'(\delta t)})^{t/\delta t} = e^{-iH'(\delta t)t/\delta t} = e^{-iHt} \tag{1.6}$$

для некоторого эрмитова оператора H. Этот оператор, умноженный на постоянную Планка h, называется Гамильтонианом, или оператором энергии нашего ансамбля (мы дальше увидим, почему у него такое название). Далее мы выбираем всюду такую систему единиц, в которой эта постоянная равна единице, так что нашим Гамильтонианом будет H. Из этого равенства мы можем заключить, что для любых моментов t и t_0 $|\Psi_t\rangle = e^{-iHt}\ |\Psi_{t_0}\rangle$. Это выражение является формулой для общего решения такого дифференциального уравнения:

$$i\frac{\partial|\Psi\rangle}{\partial t} = H|\Psi\rangle \tag{1.7}$$

Это уравнение называется волновым уравнением, на котором, по существу и основана вся квантовая механика. Самое большое искуство здесь состоит только в том, как правильно записывать Гамильтонианы для конкретных ансамблей.

Однако мы можем в достаточно общем случае продвинуться еще дальше, написав более конкретное выражение для гамильтониана, нежели костатация того несомненного факта, что он эрмитов. Во первых, заметим, что прибавление к гамильтониану диагональной матрицы с равными элементами никак не сказывается на физическом существе задачи, поскольку такая процедура приводит просто к изменению фазы в результирующей волновой функции. Так что мы можем прибавлять к H любое слагаемое вида $\lambda\ I$, где I - единичная матрица в зависимости от удобства. Рассмотрим задачу о движении одиночной частицы вдоль координатной прямой в отсутствии внешних полей. Если мы разобъем эту прямую на малые отрезки равной длины δx точками деления x_j, то амплитуда нахождения нашей частицы в каждой из этих точек будет пропорциональна значениям волновой функции Ψ в этих точках. Далее, разумно принять принцип локального действия, согласно которому на амплитуду в данной точке помимо нее самой могут влиять только амплитуды в соседних точках. Тогда волновое уравнение превратится в систему уравнений вида

$$i\frac{\partial\Psi(x_j)}{\partial t} = A\Psi(x_j) + B\Psi(x_{j+1}) + C\Psi(x_{j-1}) \tag{1.8}$$

для некоторых коэффициентов A, B, C. Далее, поскольку оба направления равноправны, мы должны считать что $B = C$. Теперь, прибавляя нужную диагональную матрицу вида λI к нашему гамильтониану, мы можем добиться, чтобы наши уравнения приняли вид

$$i\frac{\partial\Psi(x_j)}{\partial t} = a(2\Psi(x_j) - \Psi(x_{j+1}) - \Psi(x_{j-1})) \tag{1.9}$$

Теперь, переходя к пределу при $\delta x \longrightarrow 0$, мы получим в правой части выражение для второй производной волновой функции по координате:

$$i\frac{\partial\Psi(x,t)}{\partial t} = a\frac{\partial^2\Psi(x,t)}{\partial x^2}. \tag{1.10}$$

Эксперимент показывает, что коэффициент a равен $-h^2/2m$, где h - постоянная Планка, а m - масса частицы. Это и есть уравнение Шредингера для движения свободной одномерной частицы. Его обобщение на трехмерный случай очевидно. Заметим, что это уравнение очень похоже на

уравнение теплопроводности при отсутствии источников тепла. Отличие состоит только в том, что коэффициент i в нашем случае - чисто мнимое число, а не действительное, как было бы в случае теплопроводности. Тем не менее, в эвристических рассуждениях вполне разумны аналогии между изменением амплитуд и температуры.

Что произойдет, если на нашу частицу будет действовать внешнее поле с потенциалом $V(x)$? Мы сейчас дадим эвристический метод для обобщения уравнения Шредингера на этот случай. Обратим внимание на оператор двойного дифференцирования со знаком минус. Его можно представить в виде квадрата оператора $p = \frac{h}{i}\frac{\partial}{\partial x}$. Представим временно, что частица движется в классическом смысле, и применим аналогию с процессом теплообмена. Тогда скорость движения такой частицы должна быть пропорциональна разности температур, то есть результату применения оператора p. Поэтому неудивительно, что в квантовой механике p называют оператором импульса частицы. Можно показать, что при соответствующем усреднении его значения действительно получается классический импульс частицы. Но тогда гамильтониан свободной частицы нужно отождествить с ее кинетической энергией ! Если же частица не свободна, то есть движется в потенциальном поле, то к кинетической энергии логично было бы прибавить еще и потенциальную. Мы таким образом приходим к уравнению Шредингера для частицы в потенциальном поле $V(x)$:

$$ih\frac{\partial|\Psi\rangle}{\partial t} = (-\frac{P^2}{2m} + V)|\Psi\rangle \tag{1.11}$$

где $P^2 = -h^2(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2})$ обозначает скалярный квадрат оператора импульса $P = \frac{h}{i}(\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z})$.

Наконец, мы можем обобщить это уравнение на случай многочастичного ансамбля, состоящего из n взаимодействующих между собой частиц:

$$ih\frac{\partial|\Psi\rangle}{\partial t} = (\sum_{j,k=1}^{n}\frac{P_j^2}{2m_j} + V_j + V_{j,k})|\Psi\rangle \tag{1.12}$$

где V_j потенциал внешнего поля в точке нахождения j-частицы, $V_{j,k}$ - потенциал взаимодействия jй и kй частиц.

1.3 Обобщения уравнения Шредингера

Естественным обобщением уравнения Шредингера является учет электромагнитного поля с вектор-потенциалом $A = (A_1, A_2, A_3, iA_0)$. Переход к электромагнитному полю осуществляется в классической электродинамике с помощью преобразования $\bar{p} \longrightarrow \bar{p} - \frac{e}{c}A,\ E \longrightarrow E - eA_0$, где $\bar{p} = (p_1, p_2, p_3)$, и уравнение Шредингера приобретает вид:

$$ih\frac{\partial\Psi}{\partial t} = (\frac{1}{2m}(\bar{p} - \frac{e}{c}\bar{A}) + V)\Psi.$$

Далее, если учитывать энергию покоя частицы, то мы получим релятивистское уравнение для свободной частицы, которое получается так же, как и рассмотренные выше уравнения, но только если за основу взять выражение $E = \sqrt{M^2c^4 + c^2p^2}$. В релятивистские уравнения будет входить вторая производная по времени, но они будут оставаться линейными по волновой функции. Если же

мы ограничимся только рассмотрением нерелятивистского случая, который выделяется условием $|h\frac{\partial\Psi}{\partial t}|,\ |eA_0\Psi| \ll |Mc^2\Psi|$, то уравнение Шредингера всегда будет иметь вид

$$i\frac{\partial\Psi}{\partial t} = H\Psi \tag{1.13}$$

для некого эрмитова оператора H, называемого гамильтонианом системы. Уравнением вида (1.13) можно описать и релятивистские объекты, например, фотоны. Для них гамильтониан будет иметь вид

$$H = (h_{\alpha,\beta}),\ h_{\alpha,\beta} = k(\delta_{\alpha\beta} - \frac{k_\alpha k_\beta}{k^2}), \tag{1.14}$$

где $\bar{k} = (k_1, k_2, k_3)$ - вектор импульса фотона, $k = |\bar{k}|$. Обозначим его решение через $f_{\bar{k}}$. Тогда уравнение (1.14), вместе с начальным условием $\bar{k}f_{\bar{k}} = 0$, означающим поперечность фотона, будет эквивалентно уравнениям Максвелла в отсутствии зарядов:

$$rot\ \bar{E} = -\frac{\partial H}{\partial t},\ rot\ \bar{H} = \frac{\partial E}{\partial t},\ div\ \bar{H} = div\ \bar{E} = 0, \tag{1.15}$$

где $E = -\frac{\partial\bar{A}}{\partial t} - grad\ A_0,\ H = rot\ \bar{A}$. В этом несложно убедиться, если совершить над векторами электрического и магнитного поля преобразование Фурье $E = \int E_k a^{i\bar{k}\bar{r}} d^3k,\ H = \int H_k a^{i\bar{k}\bar{r}} d^3k$, так что $E_{\bar{k}}$ и $\frac{d}{dt}E_k$ пропорциональны $f_{\bar{k}} + f^*_{\bar{k}}$ и $f_{\bar{k}} - f^*_{-\bar{k}}$ соответственно. Это построение можно обобщить и на случай многих фотонов взаимодействующих с зарядами, а также и на другие взаимодействия, интересующихся деталями мы отошлем к специальной литературе ([AB], [BS]).

Мы видим, что форма уравнения Шредингера (1.13) является очень общей. Именно на таких эволюциях и построен квантовый компьютинг. Иными словами, квантовый компьютер опирается на наиболее фундаментальные положения квантовой механики, так что идею квантовых вычислений невозможно отвергнуть, не подвергая при этом существенной перестройке саму квантовую механику, во всяком случае, в ее нынешнем понимании.

1.4 Решение волнового уравнения методом возмущений

Решение волнового уравнения обычно находят так. Сначала рассматривают случай постоянного Гамильтониана $H = H_0$. В этом случае, как нетрудно убедиться непосредственным дифференцированием, решение уравнения 1.7 имеет вид:

$$\Psi(t) = \exp(-iHt)\Psi(t_0) \tag{1.16}$$

Рассмотрим теперь случай, когда Гамильтониан H зависит от времени. Если бы H был просто одномерным оператором умножения на число, мы могли бы решить уравнение 1.7 как обыкновенное дифференциальное уравнение, получив ответ

$$\Psi(t) = \exp(-i\int_{t_0}^{t} H(t)dt)\Psi(t_0). \tag{1.17}$$

Можно ли использовать эту формулу и для многомерного Гамильтониана, соответствующего настоящей квантово-механической задаче? Вопрос этот осмыслен, поскольку все операции, входящие

в 1.17 имеют смысл и для многомерных эрмитовых операторов H. Чтобы подставить Ψ в уравнение 1.7, надо продифференцировать эту функцию. Мы имеем

$$\Psi' = -i \exp(-i \int_{t_0}^{t} H(t)dt) H(t)\Psi(t_0).$$

Это выражение будет совпадать с $H(t)\Psi(t_0)$ только в том случае, когда операторы $H(t)$ при различных значениях t коммутируют. Очевидно, что в случае многомерных операторов это почти всегда не выполнено. Таким образом, формула 1.17 для многомерных Гамильтонианов не выполняется и нам придется искать более сложный путь решения уравнения 1.7.

Предположим, что Гамильтониан исходной задачи можно представить в виде $\tilde{H} = H_0 + H_1(t)$, где постоянная часть Гамильтониана H_0 не зависит от t, а возмущение H_1 - зависит. Тогда мы можем применить метод вариации произвольных постоянных, и искать решение в виде $\Psi(t + t_0) = \exp(-iH_0t)\Phi(t)$, где $\Phi(t)$ - новая неизвестная волновая функция с начальным условием $\Phi(0) = \Psi_0 = \Psi(t_0)$. Если мы введем новое обозначение

$$H = e^{iH_0t} H_1(t) e^{-iH_0t}$$

то для волновой функции Φ мы получаем задачу Коши вида:

$$i\frac{\partial}{\partial t}\Phi = H\Phi, \quad \Phi(0) = \Psi_0, \tag{1.18}$$

в чем нетрудно убедиться непосредственным подсчетом производной по времени. Эта задача называется представлением взаимодействия для волновой функции. Мы уже видели, что формула 1.17 для этого случая неприменима, поэтому мы должны поискать что-то другое. Самое простое - применить метод итераций, непосредственно вытекающий из вида уравнения 1.18. Мы могли бы применить этот прием непосредственно к волновой функции, и это было бы абсолютно аналогично тому, что мы сделаем: применим итерации к так называемой S-матрице. Эта матрица определяется с помощью соотношения

$$\Phi(t) = S(t, t_0)\Phi(t_0). \tag{1.19}$$

Если мы возьмем $t = +\infty$, $t_0 = -\infty$, то у нас получится матрица рассеяния, которая описывает процесс перехода взаимодействующих частиц из состояния в бесконечном прошлом в состояние в бесконечном будущем. При этом предполагается, что взаимодействие на бесконечности пренебрежимо мало. Из 1.18 вытекает аналогичная задача Коши для S- матрицы при $t_0 = 0$:

$$i\frac{\partial}{\partial t}S(t) = H(t)S(t), \quad \S(0) = I, \tag{1.20}$$

Применим итерационный процесс к $S(t)$. В качестве нулевого приближения возьмем единичную матрицу I. Если у нас уже есть $n-1$-е приближение S_{n-1}, то подставляя его в правую часть уравнения 1.20 и интегрируя, мы найдем n-е приближение по формуле

$$S_n = -i \int_0^t H(t)S_{n-1}(t)dt \tag{1.21}$$

из которой немедленно следует разложение в ряд вида

$$S = \sum_{n=0}^{\infty} (-i)^n \int_0^t H(t_1)dt_1 \int_0^{t_1} H(t_2)dt_2 \ldots \int_0^{t_n} H(t_n)dt_n \tag{1.22}$$

Общий член этого ряда можно представить в виде

$$(-i)^n \int\limits_{t \geq t_1 \geq t_2 \geq \ldots \geq t_n \geq 0} H(t_1)H(t_2)\ldots H(t_n)dt_1dt_2\ldots dt_n,$$

или в таком символическом виде

$$\frac{(-i)^n}{n!} T\{\int_0^t \int_0^t \ldots \int_0^t H(t_1)H(t_2)\ldots H(t_n)dt_1dt_2\ldots dt_n\}, \tag{1.23}$$

причем последняя форма означает, что при вычислении интегральной суммы, соответствующей данному интегралу, вместо подынтегральной функции надлежит брать так называемое хронологическое произведение, определяемое как

$$\begin{aligned} T\{ \quad & H(t_1)H(t_1)\ldots H(t_n)\} = H(t_{i_1})H(t_{i_2})\ldots H(t_{i_n}), \\ & t \geq t_1 \geq t_2 \geq \ldots \geq t_n \geq 0. \end{aligned} \tag{1.24}$$

Действительно, факториал в знаменателе формулы 1.23 появляется из-за того, что число всевозможных способов упорядочения n- элементного множества операторов вида $H(t_i),\ i = 1, 2, \ldots, n$, соответствующих одному хронологическому упорядочению, равно $n!$. Таким образом, мы получаем результирующую формулу для S- матрицы в виде

$$S = \sum_{n=0}^{\infty} \frac{(-i)^n}{n!} T\{\int_0^t \int_0^t \ldots \int_0^t H(t_1)H(t_2)\ldots H(t_n)dt_1dt_2\ldots dt_n\}, \tag{1.25}$$

что служит непосредственным обобщением формулы 1.16 на случай Гамильтонианов, зависящих от времени.

Сделаем напоследок одно важное замечание, касающееся практического применения формулы 1.25. В приближенных вычислениях (которые только и возможны в большинстве случаев) по формуле 1.25 используются так называемые Фейнмановские диаграммы, для применения которых Гамильтониан удобно представлять в виде конечной суммы произведений не более чем трех операторов рождений и уничтожений частиц (см. раздел "Формализм вторичного квантования"). Для операторов рождения и уничтожения бозонов определение 1.24 сохраняется, а для фермионов мы должны приписать еще множитель $(-1)^P$, где P есть четность соответствующей перестановки. Фермионные операторы входят в выражение Гамильтониана парами, поэтому равенство 1.24 для Гамильтонианов у нас сохранится. Для практического суммирования используется также технические приемы, называемые теоремами Вика. Подобные приемы актуальны для численного моделирования квантовых задач, частично мы коснемся их в последней главе, а читателям, которые интересуются более специальными деталями, можно рекомендовать обширную литературу по вторичному квантованию, например, ([BS]).

Глава 2

Элементы классической теории алгоритмов

Понятие алгоритма является одним из самых фундаментальных научных понятий. В точных дисциплинах алгоритмы использовались всегда, но строгое определение этого понятия было дано только в 30-х годах 20 века в трудах Тьюринга, Поста и Маркова (младшего). Наша точка зрения состоит в том, что понятие алгоритма при описании физических явлений не менее фундаментально, чем собственно физические понятия, такие как пространство и время. Поэтому любое существенное видоизменение этого понятия должно рассматриваться как попытка перестроить основы естествознания. Проект квантового компьютера, как мы увидим ниже, является именно такой попыткой. Драматизм ситуации усугубляется тем, что в квантовой физике в ее нынешнем понимании нет никакого запрета на существование этого, пока гипотетического, устройства. Если оно будет построено на практике, нам придется существенно пересмотреть наше понимание алгоритма. А пока было бы хорошо познакомиться поближе с существующей теорией алгоритмов.

2.1 Машина Тьюринга и другие модели алгоритмов

Алгоритмы всегда описывались (и до сих пор описываются) не слишком формально, а именно, их задают в виде некоторой инструкции о последовательности элементарных вычислительных действий. Строгое понятие алгоритма и вычисления понадобилось только тогда, когда у математиков возникли сомнения в том, что алгоритмы для решения некоторых задач вообще существуют. Мы сейчас рассмотрим модель вычислений, предложенную А.Тьюрингом. Машина Тьюринга состоит из ленты, разбитой на ячейки, и головки, способной обозревать содержимое одной ячейки, около которой она расположена, а также перемещаться на одну ячейку влево или вправо по ленте. Лента предполагается потенциально бесконечной, что означает, что мы можем при необходимости в любой момент добавлять к ней любое конечное число дополнительных ячеек. Фиксируется также: алфавиты $\omega_0 = \{a_0, a_1, \ldots, a_m\}$, $\omega_1 = \{q_0, q_1, \ldots, q_k\}$ для ячеек ленты и для состояний головки, так что в каждой ячейке ленты стоит ровно одна буква из ω_0 (а во вновь присоединяемых ячейках - всегда символ пробела a_0), а головка имеет в любой момент времени какое-либо состояние из ω_1. Состояние машины Тьюринга - это состояние ее головки и полное содержимое ее ленты, записанное в виде слова в алфавите ω_0, слева направо. Можно условиться, что новые ячейки всегда присоединяются

с правого конца ленты, а начальное состояние - это такое состояние, в котором головка обозревает первую слева ячейку и находится в состоянии q_0.

Вычислением на машине Тьюринга будет называться последовательность ее состояний, начинающаяся с одного из начальных состояний, кончающаяся на такое состояние, в котором головка находится в состоянии q_k, причем в этой последовательности нет состояний, предшествующих конечному, в котором бы головка находилась бы в состоянии q_k, и такая, что каждое состояние получается из предыдущего применением шага вычисления - формальной операции, которую мы сейчас определим. Шаг вычисления описывается с помощью набора команд вида $a_i, q_j \longrightarrow a_{k(i,j)}, q_{l(i,j)}, (R, L)$ для всех $i = 0, 1, \ldots, m;\ j = 0, 1, \ldots, k$. Каждое такое правило означает буквально следующее: если головка обозревает ячейку с содержимым a_i, находится в состоянии q_j, то происходит запись в данную ячейку символа $a_{k(i,j)}$, (при этом старое содержимое стирается), переход головки в состояние $q_{l(i,j)}$, и сдвиг ее по ленте вправо (R), влево (L), или вообще ее место не меняется (когда после запятой ничего нет). Машина считается детерминистической, если k и l - однозначные функции, и недетерминистической, если многозначные. По умолчанию все машины предполагаются детерминистическими.

Вычисление считается правильным, если при любом начальном состоянии, где на ленте написано слово (оно называется входным словом) x в алфавите $\omega' \subset \omega_0$, не содержащее пробелов, машина заканчивает работу на таком состоянии, где на ленте снова написано слово, в котором все пробелы стоят справа от слова в алфавите ω'. Это слово y является таким образом некоторой функцией от x, которую наша машина и вычисляет. Функция, для которой существует какая-либо вычисляющая ее машина Тьюринга, называется вычислимой. Множество считается вычислимым (или разрешимым), если его характеристическая функция вычислима.

Из соображений мощности ясно, что далеко не все словарные функции вычислимы. Приведем пример одной из таких функций. Каждую машину Тьюринга можно однозначно записать в виде слова в некотором конечном алфавите (достаточно алфавита, состоящего всего из двух букв; даже из одной - только в последнем случае эта запись была бы экспоненциально длиннее суммы длин команд). Код машины Тьюринга T обозначим через $[T]$. Если для некоторого входного слова наша машина заканчивает работу при правильном вычислении, мы скажем, что эта машина признает данное слово, и обозначаем это через $T!x$. Рассмотрим множество A пар слов вида $[T], x$, таких что $T!x$. Если бы это множество было бы разрешимо, то было бы разрешимо и множество A_0 таких натуральных чисел n, что $\exists T : \ [T] = n \ \& T!n$. Но тогда мы могли бы построить такую машину Тьюринга T_0, что для всякого n $T_0!n$ тогда и только тогда, когда $\exists T : \ n = [T] \ \& T \not! n$. Для этого достаточно просто запускать машину, разрешающую A_0, и в случае если данный номер не принадлежит A_0 просто искусственно зацикливать вычисление. Тогда в силу нашего определения T_0 обе возможности: $T_0![T_0]$ и $T_0 \not![T_0]$ приводят к противоречию. Таким образом, множества A и A_0 неразрешимы, и у нас появился первый пример алгоритмически неразрешимой проблемы: проблемы применимости данного алгоритма к данному числу. Отсюда можно сделать такой вывод: если задан алгоритм (то есть код машины Тьюринга) и некоторое число n (в виде начального слова), то единственный универсальный способ определить результат работы этого алгоритма на этом слове - это запустить его, выполняя последовательно все его шаги, и ждать результат.

Сложность вычисления машины Тьютинга T на данном входном слове x есть число всех шагов машины Тьюринга при работе начиная с этого слова. Если n некоторое фиксированное число, то через $f_T(n)$ будем обозначать максимальную сложность вычисления T на словах длины не превосходящей n. Функция f_T называется временной сложностью данной машины T.

Предположим, что имеется некоторая функция F неизвестной природы, которая может быть и невычислимой. Определим важное понятие вычисления с оракулом F. Для этого на ленте нашей

машины надо выделить некоторую заранее фиксированную ячейку, называемую регистром вопроса, а также некоторую часть, называемую местом для вопроса, и некоторую другую часть, называемую местом для ответа. Если в вопросном регистре содержится какая-либо буква кроме некоторой специальной, например, a_1, то делается очередной шаг вычисления так, как мы это описали выше. Но если в вопросном регистре стоит a_1, то вместо обычного шага вычисления мы делаем следующее. Задается вопрос оракулу: каково значение функции F на слове, стоящем в месте для вопроса, затем мы ждем, когда оракул сообщит нам значение функции F на этом слове, и его ответ мы записываем в место для ответа. такая процедура называется запросом к оракулу, или вызовом оракула на вопросном слове. После этого в вопросный регистр помещается пробел и мы продолжаем вычисления так, как это было описано. В ходе вычислений как место для вопроса, так и место для ответа и вопросный регистр могут быть изменены, и в следующий раз оракул быдет вызван уже на какомто, вообще говоря другом слове, и т.д. В этом случае сложностью мы будем считать число вызовов оракула. Это связано с тем, что оракул считается некоторым внешним устройством, сложность которого намного превосходит нашу машину Тьюринга, так что сложность нашего вычисления с оракулом определяется не шагами нашего алгоритма, а числом запросов к оракулу.

2.2 Тезис Черча. Сложность классических алгоритмов

Приведенная формализация понятия алгоритма не является единственно возможной. Известны нормальные алгоритмы Маркова, машины Поста, алгоритмы Колмогорова-Успенского и др. формализации. Однако понятие вычислимой функции во всех этих формализациях одно и то же. Это означает, что если некоторая функция является вычислимой в одной из этих формализаций, то она будет вычислимой и во всех других формализациях. Более того, все формализации алгоритма полиномиально сводимы одна к другой. Это означает, что для любых двух формализаций существует полином, такой что сложность вычисления любой функции в любой из этих формализаций не превосходит значения этого полинома, взятого от сложности вычисления этой же функции в другой формализации. В частности, если какая то функция вычислима за полмномиальное время в какой-то из формализаций, она будет вычислимой за полиномиальное время в любой другой формализации. Все эти однотипные утверждения обычно подытоживают в виде так называемого тезиса Черча, Поста и Тьюринга (или просто тезиса Черча). Он кратко формулируется так:

Понятие вычислимости универсально и не зависит от выбора формализации.

В более узкой трактовке, относительно сложности это означает также и то, что понятие полиномиально вычислимых функций также не зависит от выбора формализации. Здесь речь идет, конечно, только о детерминистических моделях алгоритма. Таким образом, мы можем рассмотреть класс P всех множеств, характеристические функции которых имеют полиномиальную сложность. Это определение не будет зависеть от выбора формализации алгоритмов. Рассмотриим какую-либо характеристическую функцию B (ее еще называют предикатом) множества, принадлежащего классу P. Разобъем агрумент этой функции на две части x и y, так что значением $B(x, y)$ будет 0 или 1. Зафиксируем также некоторый полином p. Тогда мы вправе рассмотреть множество A слов вида x, таких что существует y с длиной записи, не превосходящей значения p от длины записи x, такой что $B(x, y) = 1$. Все такие множества A образуют класс множеств NP[1]. Пусть мы зафиксировали некоторое множество A из класса NP. Для того, чтобы определить, принадлежит ли какое-нибудь слово x множеству A, нам нужно перебрать все слова y, с длиной не превосходящей некоторого

[1] Такое обозначение происходит из того, что это в точности те множества, которые распознаются за полиномиальное время, но на недетерминистических машинах Тьюринга.

полинома от длины записи x, и для каждого из таких проверить, совпадает ли $B(x, y)$ с единицей, или нет. Поскольку B считается бостаточно быстро, то мы имеем дело с переборной задачей типа подбора кода к замку, где роль замка играет x и B, а роль кода - y. Таким образом, мы можем утверждать, что класс P содержится в классе NP. Верно ли, что эти классы вообще совпадают? Ответ на этот вопрос до сих пор неизвестен.

Попробуем применить понятие алгоритма для того, чтобы оценить перспективы классического моделирования квантовых эволюций. Поскольку дело сводится к вычислению экспоненты от матрицы, пропорциональной гамильтониану рассматриваемой системы (в случае, когда последняя зависит от времени это будет хронологическая экспонента), все элементы которого в принципе можно посчитать со сколь угодно большой точностью (при неограниченном запасе времени и памяти), мы можем со всей определенностью утверждать, что **любая квантовая эволюция может быть в принципе рассчитана на классическом компьютере с любой наперед заданной точностью**, если, конечно, не обращать внимания на сложность вычисления. Это означает, что нет никакой надежды построить с помощью квантовой механики вычислительную машину, которая бы "вычисляла"невычислимые функции (например, распознавала бы применимость алгоритма к данному входному слову). Иными словами, тезис Черча (в части, не относящейся к сложности) остается справедливым и для квантовых вычислительных устройств.

Глава 3

Квантовый компьютер и квантовые вычисления

3.1 Почему компьютеры тесно связаны с квантовой механикой

Мы уже видели, что тезис Черча остается справедливым и для всех возможных устройств, основанных на квантовой эволюции. У нас нет, таким образом, никакой надежды решить с помощью квантовой механики какую-либо из алгоритмически неразрешимых задач. Но понятие алгоритмической разрешимости не очень-то практично. Действительно, какой смысл считать разрешимой задачу, требующую для своего решения время, превосходящее возраст Вселенной или памяти, превосходящей число всех элементарных частиц в ней? Более разумно классифицировать задачи по сложности их решения. В качестве меры сложности можно избрать либо память, либо время. Можно также рассматривать универсальную меру сложности, взяв в качестве ее меры сумму: память + время. При наличии большого ресурса времени его можно часто (но не всегда!) конвертировать в память, т.е. восполнить недостаток памяти. Обратное конвертирование в большинстве случаев невозможно. Таким образом, в теории алгоритмов именно время считается наиболее ценным ресурсом. Мы будем в данной книге придерживаться этой точки зрения, характерной вообще для динамических картин мира. Однако, справедливости ради, отметим, что существует и иной, более общий подход к понятию сложности, введенный Колмогоровым - сложность описания конечных объектов. В рамках этого подхода сложностью объекта называется длина слова, которое задает его точное описание. Например, сложность последовательности вида $0101\ldots01$ состоящей из N пар 01 будет иметь порядок $\log N$, поскольку для ее описания достаточно указать ее длину и описать период.

Займемся, однако, квантовыми компьютерами. В электронике известен так называемый закон Мура, согласно которому размеры вычислительных элементов уменьшаются вдвое каждый год, а значит, с той же скоростью увеличивается тактовая частота вычислений. Значит ли это, что вычислительные мощности, доступные человечеству, будут возрастать неограниченно? Увы, ответ на этот вопрос отрицательный. Дело в том, что у нынешних компьютеров существует фундаментальный физический предел уменьшению их элементарных ячеек памяти - размер атомов. Для элементов,

размеры которых сравнимы с атомными (около 1 ангстрема) начинают действовать квантовые законы микромира, описанные нами в первой главе. Это значит, в частности, что при попытке управлять этими элементами в манере, характерной для традиционной электроники, мы столкнемся с их неповиновением, или, как это называется на научном языке - с квантовым хаосом. Таким образом, уже для управления объектами таких размеров необходимо применение аппарата квантовой механики. Современные элементы памяти компьютеров достигают величины примерно в 50 раз превосходящих атомные, так что (при сохранении закона Мура) эта проблема станет препятствием уже через 6 - 7 лет. Но это еще не все. Существующие компьютеры используют в своей основе классические алгоритмы и способы хранения информации. А эти методы для ряда очень важных задач требуют астрономических размеров памяти и таких же запасов времени. Например, если мы хотим с приемлемой точностью и с учетом квантовых эффектов (которые в этом случае принципиально важны) промоделировать эволюцию молекулы ДНК, состоящую из нескольких сотен миллиардов элементарных частиц (даже при отбрасывании несущественных деталей), то в силу экспоненциального квантового порога это означает, что мы должны одновременно управлять более чем 2^{10^7} элементами памяти, а это превышает число элементарных частиц во Вселенной! Так что даже если мы и научимся оперировать элементарными частицами, ряд принципиальных задач все равно будут недоступны нашим компьютерам. Молекулу ДНК мы здесь взяли только для наибольшей наглядности, в действительности классическим путем нельзя рассчитать с приемлемой точностью даже поведение одного основания, входящего в состав этой молекулы - таких оснований всего четыре: аденин, гуанин, тимин и цитозин, в каждом из них и в их соединениях явным образом проявляются квантовые эффекты.

Что же мы имеем в итоге? Все развитие электроники, а значит и новых подходов в технике, биологии, медицине и многих других ключевых отраслях, связано с миниатюризацией элементов, ответственных за обработку информации. Именно миниатюризация таких элементов представляет глобальный вектор повышения эффективности всей мировой цивилизации. И именно здесь мы сталкиваемся с фундаментальным препятствием, которое ставит предел такому повышению. Принципиальная идея, помогающая преодолеть это препятствие, была высказана в 80-х годах Фейнманом, который предложил для исследования квантовых систем использовать не обычный, а квантовый компьютер. В те же годы Бениофф также высказал идею об использовании гамильтонианов квантовых систем для производства вычислений. Однако первым, кто математически строго сформулировал понятие квантового вычисления, был Дейч. Основные квантовые вычислительные "трюки"были изобретены Шором (квантовое преобразование Фурье) и Гровером (квантовый перебор). Дальнейшее развитие этой области связано с комбинированием этих и подобных идей с известными подходами теории алгоритмов. Такое комбинирование позволило построить для очень широкого круга традиционных вычислительных задач быстрые квантовые алгоритмы, которые по своей эффективности качественно превосходят все возможные классические аналоги. Более того, эффективность квантовых вычислений настолько велика, что им поддаются и те задачи, которые в настоящее время даже не рассматриваются в качестве решаемых на компьютере! Примерами являются задачи моделирования сложных многочастичных систем, или задачи криптографии. В этой главе мы дадим абстрактную схему квантового компьютера, расскажем о двух основных квантовых трюках и некоторых их приложениях.

3.2 Абстрактная схема КК

3.2.1 Вычисления без дополнительных устройств

Идея квантового компьютера очень проста. Предположим, что нам надо найти некоторое неизвестное натуральное число p. Это может быть закодированное разложение данного натурального числа на простые множители, или решение некоторого целочисленного уравнения, или еще что нибудь, но для нас сейчас важно только, что это число нам неизвестно, и что оно соответствует какому-то базисному состоянию квантовой системы, находящейся в нашем распоряжении. Представим себе, что мы можем каким-то хитрым способом подбирать Гамильтониан взаимодействия для этой системы H, так что если системы начинает свою эволюцию с состояния ψ_0, то в момент времени t она будет находиться в состоянии $\psi(t) = e^{-iHt}\psi_0$. Этот способ подбора взаимодействия называется управлением квантовым вычислением. Разложим это состояние по базису $e_0, e_1, \ldots, e_{N-1}$ так что у нас получится $\psi(t) = \sum_{j=0}^{N-1} \lambda_j(t)|e_j\rangle$. Так что весь закон эволюции заключается в функциях $\lambda_j(t)$. Будем считать, что мы нашли наше p, если при измерении состояния данной системы мы с высокой вероятностью получили состояние $|e_p\rangle$. Такое соглашение очень хорошо соответствует житейскому смыслу слова "найти". Например, при решении переборной задачи на определение пароля доступа к чужому интернет - сайту , нам было бы вполне достаточно, если бы кто-то подсказал нам этот пароль, поскольку проверить его правильность - дело совсем простое. Так вот, квантовый компьютер и будет играть роль такого "подсказчика". Но какова вероятность, что его ответ при измерении будет верным? Она, как мы знаем, равна $|\lambda_p|^2$. Составим график роста этой вещественной функции во времени - пусть он имеет пик в некоторой точке t_{quant}, как изображено на рисунке 2, причем $t_{quant} < t_{class}$, где t_{class} есть время, которое тратит классический компьютер на поиск p. Тогда, имея такую квантовую систему, мы можем узнать p раньше, чем на классическом компьютере: запустив нужное взаимодействие и подождав t_{quant} мы наблюдаем систему в состоянии, соответствующем максимуму модуля амплитуды искомого состояния, и получаем p раньше, чем это сделает наш конкурент, вооруженный классическим компьютером. Это и есть квантовое ускорение для классических алгоритмов. Для ряда очень важных задач, например, для задачи перебора или поиска собственных чисел операторов, это ускорение так велико, что для его демонстрации достаточно совсем небольшой квантовой системы: несколько десятков кубит. Для иных классов задач, например, для так называемой задачи определения равновесия значений данной булевской функции (PARITY) это ускорение совсем небольшое - в два раза. Существуют и такие задачи, для которых применение квантового компьютера не может дать ускорения. К числу последних относится нахождение результата итерации (последовательного применения) выбранной наугад классической функции F, т.е. нахождение $F(F(\ldots F(x_0)\ldots))$, при условии, что число ее последовательных применений не слишком велико. Хотя такие экзотические примеры теоретически составляют значительную часть всех возможных формальных задач [1], мы не будем ими заниматься. Важно то, что для огромного числа практически значимых задач возможно серьезное квантовое ускорение, и именно это сулит в обозримом будущем настоящий переворот в информатике.

Приведенная идея квантового компьютера слишком схематична. Вы можете спросить: а каким образом сравнивать классическое и квантовое времена, ведь их природа различна. Квантово-механическое время непрерывно, тогда как классическое измеряется числом элементарных операций, то есть дискретно. Непонятно также, что означает "выбрать гамильтониан", и как это связано

[1] Их можно сравнить с множеством звезд на небе - хоть их и очень много, для нас ценность представляет в основном одна ближайшая к нам.

с условиями, определяющими неизвестное число p.

Давайте теперь уточним нашу идею, построив настоящую, хотя и абстрактную, схему квантового компьютера. Прежде всего нужно разделить в нем две части: классическую, предназначенную для управления, и квантовую, служащую собственно для производства вычислений. Как могут эти части влиять друг на друга? Классическая часть может влиять на квантовую, определяя взаимодействие в ней, то есть задавая Гамильтониан эволюции квантовой части. Влияние же квантовой части может проявляться только через получение результата ее измерения. Поскольку измерение, как мы знаем, необратимо разрушает состояние квантовой части, можно было бы ограничиться случаем, когда измерение происходит только в конце вычисления, т.е. классическая часть в ходе работы компьютера изменяется по совершенно самостоятельным, классическим законам, и совершенно не зависит от квантовой части. Эволюция же квантовой части полностью определяется состоянием классической. Таким образом, в таком одностороннем вычислительном устройстве классическую часть можно сравнить с неопытным водителем автомобиля, а квантовую - с самим автомобилем. Однако возможен и компьютер боле общего типа: с обратной связью, т.е. когда обе его части - классическая и квантовая взаимно влияют друг на друга в ходе самого вычисления. Такой компьютер отличается от одностороннего только тем, что в ходе вычисления он применяет промежуточные измерения, и затем использует результат этих измерений в своей дальнейшей работе. Здесь наш водитель будет уже опытным, он своевременно реагирует на состояние своего автомобиля. В любом случае ясно одно: программа (или алгоритм) для квантового компьютера всегда является классической программой, предназначенной для его управляющей, классической части. В такой программе должно явно указываться: как приготовить начальное состояние обеих частей компьютера, какое преобразование и над какими кубитами квантовой части нужно совершить в каждый момент вычисления, когда надо измерить квантовую часть, как использовать результат этого измерения, и наконец - когда остановить вычисление и выдать ответ. Наличие обратной связи предоставляет в некоторых случаях дополнительные возможности в вычислениях, и мы это увидим на примере переборной задачи с неизвестным числом решений. Сейчас же давайте рассмотрим подробнее схему одностороннего квантового компьютера.

Квантовая часть нам уже знакома. Она состоит из n кубит, так что мы можем свободно обращаться к каждому из них, причем они могут находиться в произвольном квантовом состоянии вида

$$\psi = \sum_{j=0}^{N-1} \lambda_j |e_j\rangle, \tag{3.1}$$

где базисные состояния имеют вид

$$\begin{array}{ll} |e_0\rangle & = |00\ldots00\rangle \\ |e_1\rangle & = |00\ldots01\rangle \\ |e_2\rangle & = |00\ldots10\rangle \\ |e_3\rangle & = |00\ldots11\rangle \\ \ldots & \ldots \\ |e_{N-1}\rangle & = |11\ldots11\rangle, \end{array} \tag{3.2}$$

и предполагается, что мы можем производить все измерения квантовой части в данном базисе. Напомним, что $N = 2^n$, так что мы можем свободно оперировать со всеми n кубитами, но число N всех состояний квантовой части нам недоступно, так что сумма в равенстве (6.7) носит чисто теоретический характер и может становиться доступной для нас только при измерениях квантовой части компьютера. В частности, мы не имеем право при построении квантовых алгоритмов

что-то менять в этой сумме, пусть даже с помощью какого-либо классического алгоритма. Таким изменениям нельзя даже в принципе сопоставить какой-либо физический процесс, и поэтому они не представляют никакой ценности. Единственный способ обращения с квантовой памятью заключается в использовании специального управления, которое мы и собираемся описать.

Зафиксируем для начала список простых унитарных преобразований $\bar{U} = \{U_1, U_2, \ldots, U_s, \ldots\}$ над ограниченным числом кубит, и назовем эти операции элементарными. Мы будем считать, что в их числе есть все однокубитовые, и те из двух- и трех- кубитовых, которые нам понадобятся в дальнейшем, при явном описании квантовых алгоритмов. В действительности можно было бы обойтись всеми однокубитовыми и одним произвольно выбранным двухкубитовым преобразованием (с такой ситуацией мы еще познакомимся, когда будем изучать простые способы квантового управления), но это не всегда удобно, особенно вначале. Если мы каким-то образом назначим для некоторых элементарных операций $U_{i_1}, U_{i_2}, \ldots, U_{i_k}$ определенные кубиты квантовой части, над которыми надо провести эти операции, так чтобы каждый кубит участвовал ровно в одной операции, то мы сможем совершить преобразование $U_{i_1} \bigotimes U_{i_2} \bigotimes \ldots \bigotimes U_{i_k}$ над всей квантовой частью. Такое преобразование будем называть рабочим.

Управление назначением кубитов для элементарных операций будет заниматься классическая часть компьютера. Мы можем считать, что каждой элементарной операции сопоставлен ее двоичный код, и аналогичный код указывает на назначение кубитов для такой операции. Тогда набор таких кодов однозначно определит рабочее преобразование. Так вот, в рабочей части будет храниться такой большой код, который от шага к шагу будет меняться под действием какого-либо классического алгоритма. Можно считать, что классическая часть представляет собой обыкновенный компьютер, на котором запущена некая программа, причем состояние его памяти всякий раз указывает, какую манипуляцию в данный момент нужно сделать с квантовой системой. Кроме того, эта управляющая классическая машина указывает, в какой момент нужно измерить состояние квантовой части и как дальше использовать результат измерения, если этот квантовый компьютер обладает обратной связью.

Для вычислений на нем применяются те же меры сложности по памяти и по времени, что и для классических вычислений. Однако надо сказать, что смысл этих понятий здесь будет иной. Этот новый смысл связан с основной особенностью квантовой механики - с интерференцией амплитуд. За счет этой интерференции один такт квантового вычисления - рабочее преобразование - делает гораздо большую работу, чем один такт классического вычисления. Этот тезис совершенно не очевиден. Его настоящее обоснование мы получим чуть позже, когда начнем изучать быстрые квантовые алгоритмы, и убедимся, что описанное нами устройство действительно способно делать совершенно невероятные с житейской точки зрения вещи. Пока же это можно косвенно проиллюстрировать на таком примере. Представим себе, что мы решили бы моделировать квантовое вычисление вида (??) на классической машине. Тогда нам надо было бы хранить в памяти всю матрицу рабочего преобразования, которая имеет размерность $N \times N$, тогда как ресурс памяти на квантовом компьютере, потребный для такой цели, составляет всего лишь порядка n кубит, т.е. в логарифм раз меньше.

3.3 Квантовый оракул как дополнительное устройство

Описанный квантовый компьютер вычисляет сам по себе, без обращения к какой либо внешней системе. В то же время очень полезно для приложений рассмотреть более общий случай, когда в ходе вычисления компьютер может обращаться к некоторому внешнему устройству с запросом на выполнение определенной операции, которая не входит в число элементарных, и следовательно не

может быть непосредственно выполнена самим компьютером. Такое внешнее устройство называется оракулом. Так с древнейших времен называли прорицателей, имеющих способность предсказывать будущее. Естественно считать, что оракул может обладать внутренней сложностью, намного превосходящей сложность нашего компьютера. Следовательно, для вычислений с оракулом в качестве времени надо брать не общее число операций, а число обращений к оракулу, поскольку каждое такое обращение гораздо более значимо, чем рабочие преобразования.

В квантовой части мы выделим специальный набор кубит, предназначенных для операции оракула. Для наглядности рассмотрим следующий пример. Пусть у нас имеется классическое вычисление, имеющее целью найти решение уравнения $f(x) = 1$, для булевской функции $f : \{0,1\}^n \longrightarrow \{0,1\}$. Покажем, как решить эту задачу, используя эту функцию в качестве оракула. Зададим начальное значение $x = 0$. Алгоритм для решения уравнения состоит из цикла, в котором делаются такие последовательные операции: вызывается оракул на x, если он выдает 1, вычисление останавливается и в качестве ответа выдается x, если он выдает 0, значение x увеличивается на единицу. При этом в памяти компьютера должно быть предусмотрено место для ответа, который выдает оракул - в этом случае один бит для значения $f(x)$.

Как должен действовать квантовый оракул, производящий унитарное преобразование над несколькими кубитами? Рассмотрим снова поиск решения уравнения $f(x) = 1$. Такое преобразование должно быть взаимно-однозначным, как унитарное, а с другой стороны оно должно нести в себе полную информацию о функции f. Значит, просто использовать в качестве квантового оракула саму функцию f не получится - ведь она наверняка не является взаимно однозначной при $n > 1$! Немного подумав, мы бы решили, что нас удовлетворило бы такое определение квантового оракула, соответствующего функции f:

$$Qu_f \ |\bar{a}, b\rangle = |\bar{a}, b \bigoplus f(\bar{a})\rangle, \tag{3.3}$$

где $\bar{a} \in \{0,1\}^n$, $b \in \{0,1\}$. Действительно, так определенная операция отображает базисные состояния в базисные, т.е. не порождает никаких линейных комбинаций, которых не было в исходном состоянии, является взаимно-однозначным, так как сохраняет "память"об аргументе функции f, следовательно эта операция унитарна. Наконец, если первоначально b было нулевым, то после применения этой операции там будет содержаться значение $f(\bar{a})$. Равенство (3.3) мы будем считать определением действия квантового оракула, соответствующего классической функции f.

Если немного потрудиться, можно доказать следующий факт. Для любой схемы из функциональных элементов, задающей функцию f, можно построить квантовую схему из функциональных элементов (quantum gate array) реализующую функцию Qu_f. Таким образом, квантовый оракул можно представлять себе как некоторую подпрограмму, код которой открыт для нас, но мы можем работать с ней только запуская ее на некотором входном состоянии.

Итак, работа оракула, или вопросное преобразование, выглядит следующим образом: $Qu = Qu_f \bigotimes I$, где первый сомножитель действует только на кубиты из списка $\bar{a}, b$, а второй сомножитель - идентичное преобразование - действует на все остальные кубиты. Таким образом, эволюция квантовой части при вычислении будет иметь вид:

$$\psi_0 \longrightarrow \ldots \longrightarrow \psi_{h_1} \longrightarrow \psi_{h_1+1} \longrightarrow \ldots \longrightarrow \psi_{h_2} \longrightarrow \psi_{h_2+1} \longrightarrow \ldots\ldots \longrightarrow \psi_{h_r} \longrightarrow \psi_{h_r+1} \longrightarrow \ldots \longrightarrow \psi_R$$

где специально выделены вопросные преобразования. Сложность такого вычисления есть r.

3.4 Формальное определение квантовых алгоритмов

В этом параграфе приводится более формальный вариант описания квантового компьютера. Мы будем определять квантовый компьютер с оракулом для некоторой словарной функции, сохраняющей длину входного слова. Отметим, что дословно аналогичное определение можно было бы сформулировать и для оракула обыкновенной булевской функции. Опишем последовательно обе части квантового компьютера.

Квантовая часть

Она состоит из двух потенциально бесконечных лент: рабочей и вопросной, конечного списка $\mathcal{U}$ унитарных преобразований, о которых предполагается, что их можно легко реализовать на стандартных физических устройствах, и бесконечный набор $F = \bigcup_{n=1}^{\infty} F_n$ унитарных преобразований, называемых оракулом для сохраняющей длину функции $f : \{0,1\}^* \longrightarrow \{0,1\}^*$, где всякое F_n действует на 2^{2n} мерном Гильбертовом пространстве, порожденном векторами $\{0,1\}^{2n}$ следующим образом:

$$F_n|\bar{a},\bar{b}\rangle = |\bar{a}, f(\bar{a}) \bigoplus \bar{b}\rangle, \quad \bar{a},\bar{b} \in \{0,1\}^n,$$

где $\bigoplus$ обозначает покомпонентное сложение по модулю 2.

Ячейки лент будем называть кубитами. Каждый кубит принимает значения из комплексной окружности радиуса 1: $\{z_0\mathbf{0} + z_1\mathbf{1} \mid z_1, z_2 \in \mathtt{C}, |z_0|^2 + |z_1|^2 = 1\}$. Здесь $\mathbf{0}$ и $\mathbf{1}$ есть базисные состояния кубита, образующие базис $\mathtt{C}^2$.

В течение всего времени вычислений обе ленты ограничены каждая двумя маркерами, занимающими постоянные позиции, так что на рабочей (вопросной) ленте только кубиты $v_1, v_2, \cdots, v_\tau$ $(v_{\tau+1}, v_{\tau+2}, \cdots, v_{\tau+2n})$ доступны в вычислении с временной сложностью $\tau = \tau(n)$ на входе длины n. Положим $Q = \{v_1, v_2, \cdots, v_{\tau+2n}\}$. Базисное состояние квантовой части есть функция вида $e : Q \longrightarrow \{0,1\}$. Такое состояние можно закодировать как $|e(v_1), e(v_2), \cdots, e(v_{\tau+2n})\rangle$ и естественно отождествить с соответствующим словом в алфавите $\{0,1\}$. Пусть $K = 2^{\tau+2n}$; $e_0, e_1, \cdots, e_{K-1}$ есть все базисные состояния, взятые в некотором фиксированном порядке, $\mathcal{H}$ есть K мерное Гильбертово пространство с ортонормированным базисом $e_0, e_1, \cdots, e_{K-1}$. $\mathcal{H}$ можно рассматривать как тензорное произведение
$\mathcal{H}_1 \bigotimes \mathcal{H}_2 \bigotimes \cdots \bigotimes \mathcal{H}_{\tau+2n}$ двумерных пространств, где $\mathcal{H}_i$ порождено всевозможными значениями v_i, $i = 1, 2, \cdots, \tau + 2n$. Состояние (чистое) квантовой части есть такой элемент $x \in \mathcal{H}$, что $\|x\| = 1$.

Эволюция рассматриваемой квантовой части во времени определяется двумя типами унитарных преобразований над ее состояниями: рабочими и вопросными. Пусть пара G, U выбрана произвольным образом, так что $G \subset \{1, 2, \cdots, \tau + 2n\}$, $U \in \mathcal{U}$ суть унитарное преобразование в $2^{\mathrm{card}(G)}$ мерном Гильбертовом пространстве.

Рабочее преобразование $W_{G,U}$ на $\mathcal{H}$ имеет форму $E \bigotimes U'$, где U' действует как U на $\bigotimes_{i \in G} \mathcal{H}_i$ в рассматриваемом базисе, E действует как тождественный оператор на $\bigotimes_{i \notin G} \mathcal{H}_i$.

Вопросное преобразование Qu_f на $\mathcal{H}$ имеет форму $E \bigotimes F'_n$, где F'_n действует как F_n на $\bigotimes_{i=\tau+1}^{\tau+2n} \mathcal{H}_i$ и E действует как тождественное на $\bigotimes_{i=1}^{\tau} \mathcal{H}_i$.

Наблюдение над квантовой частью. Если квантовая часть находится в состоянии $\chi = \sum_{i=0}^{K-1} \lambda_i e_i$,

наблюдением называется случайная величина, принимающая значение e_i с вероятностью $|\lambda_i|^2$.

Классическая часть

Классическая часть компьютера также состоит из двух лент: рабочей и вопросной, ячейки которых находятся во взаимно - однозначном соответствии с соответствующими кубитами квантовых лент компьютера и имеют ограничивающие маркеры на соответствующих

позициях. Каждая ячейка классических лент содержит букву из некоторого конечного алфавита ω. Эволюция классической части определяется классической машиной Тьюринга M

с несколькими головками на обеих лентах и множеством объединенных состояний головок: $\{q_b, q_w, q_q, q_o, \cdots\}$. Мы обозначаем через $h(C)$ объединенное состояние головок для состояния C классической части.

Пусть D будет множеством всех состояний классической части.

Правило соответствия между квантовой и классической частями имеет вид $R: D \longrightarrow 2^{\{1,2,\cdots,\tau+2n\}} \times \mathcal{U}$, где $\forall C \in D\ R(C) = \langle G, U\rangle$, U действует на $2^{\mathrm{card}(G)}$ мерном Гильбертовом пространстве так что U зависит только от $h(C)$, и элементы G суть в точности номера ячеек в классической части, содержащих специальную букву $a_0 \in \omega$.

Состояние квантового компьютера есть пара $S = \langle Q(S), C(S)\rangle$ где $Q(S)$ и $C(S)$ - состояния квантовой и классической частей соответственно.

Вычисление на квантовом компьютере. Вычислением мы назовем последовательность преобразований следующего вида:

$$S_0 \longrightarrow S_1 \longrightarrow \cdots \longrightarrow S_\tau, \tag{3.4}$$

где для всякого $i = 0, 1, \cdots, \tau - 1$ $C(S_i) \longrightarrow C(S_{i+1})$ есть переход, определенный машиной Тьюринга M, причем выполнены следующие свойства:

если $h(C(S_i)) = q_w$ то $Q(S_{i+1}) = W_{R(C(S_i))}(Q(S_i))$,

если $h(C(S_i)) = q_q$ то $Q(S_{i+1}) = \mathrm{Qu}_f(Q(S_i))$,

если $h(C(S_i)) = q_b$ то $i = 0$, $Q(S_0) = e_0$, $C(S_0)$ суть фиксированное начальное состояние, соответствующее входному слову $a \in \{0,1\}^n$,

если $h(C(S_i)) = q_o$ то $i = \tau$,

во всех других случаях $Q(S_{i+1}) = Q(S_i)$.

Мы скажем, что этот квантовый компьютер (QC) вычисляет функцию $F(a)$ с вероятностью $p \geq 2/3$, если для вычисления (1.1) на всяком входном слове a наблюдение S_τ с последующей заранее фиксированной рутинной процедурой обработки результата дает $F(a)$ с вероятностью p. Заметим, что при $p < 1$ всегда можно достичь любого большего значения вероятности $p_0 > p$ если выполнить повторные вычисления с тем же входным словом и затем взять в качестве окончательного ответа тот результат, который встречается чаще всех других. Это приведет только к линейному замедлению вычислений по сравнению со старым уровнем вероятности p. Описанные вычисления называются вычислениями с ограниченной вероятностью ошибки. В случае $p = 1$ мы будем иметь точные вычисления.

Если в вычислении используется оракул, то в качестве основной меры сложности вычисления (1.1) мы будем брать не τ, а число вопросных преобразований в данном вычислении. Таким образом, становится непринципиальным выбор машины Тьюринга в качестве модели эволюции классической части.

Вместо нее можно выбрать любую другую модель классических вычислений: клеточный автомат, нормальный алгоритм и т. п., сложность вычислений от этого не изменится.

Глава 4

Быстрый квантовый перебор

4.1 Почему КК делает перебор вариантов необыкновенно быстро

В этом параграфе мы рассматриваем знаменитый квантовый алгоритм для нахождения решения уравнения $f(x) = 1$, придуманный американским математиком Ловом Гровером в 1996 году. Этот алгоритм находит решение за время порядка квадратного корня из классического времени, т.е. для нахождения решения нужно $O(\sqrt{N})$ обращений к оракулу соответствующему f. Этот алгоритм имеет огромное число потенциальных приложений в вычислительных задачах, поскольку к этой задаче сводятся все задачи на прямой перебор вариантов. Действительно, предположим, что нам надо найти какую-либо строку, обладающую некоторым легко проверяемым свойством P. Несмотря на легкость проверки этого свойства для данной конкретной строки, найти строку с этим свойством очень даже не просто. Для этого надо перебрать довольно большое число всяких других строк, последовательно проверяя каждую. Можно доказать (мы не будем этим утруждать читателя), что при разумном уточнении формулировок иной, кроме перебора, универсальный метод поиска, вообще говоря, невозможен. Такое свойство P можно оформить в виде булевской функции f так что этому свойству будут удовлетворять в точности решения уравнения $f(x) = 1$. Более того, если P задано в виде алгоритма, то можно и f представить в виде схемы из функциональных элементов, а значит и составить схему из квантовых функциональных элементов, реализующую квантовый оракул, соответствующий f. Для построения искомого алгоритма GSA (Grover search algorithm) нам понадобится две важных подпрограммы.

4.1.1 Преобразование Уолша -Адамара

Рассмотрим один из важных и полезных примеров унитарных преобразований - преобразование Уолша - Адамара. Его однокубитовы1 вариант выглядит так:

$$W = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}.$$

n кубитовый вариант определяется как независимое действие W на каждом из кубитов: $W^{\bigotimes n}$. Из определения этого преобразования видно, что амплитуда перехода базисного состояния в любое

другое одинакова, а знак меняется столько раз, сколько единиц стоят на одинаковых местах в обоих этих состояниях. Мы можем написать общую формулу для этого преобразования как $w_{i,j} = (-1)^{(i \cdot j)}$, где в показателе стоит скалярное произведение бинарных записей чисел i и j. Это означает, что если числа i и j имеют вид $\sum_{s=0}^{n-1} i_s 2^s$ и $\sum_{s=0}^{n-1} j_s 2^s$ соответственно, то $i \cdot j = \sum_{s=0}^{n-1} i_s j_s$. Нетрудно видеть, что $(W^n)^{-1} = W^n$.

Это преобразование замечательно тем, что оно не приводит к запутыванию кубитов. Если первоначальное состояние кубитов было незапутанным, таковым же будет и результат преобразования W^n. Если мы применим это преобразование к n кубитам, находящимся в нулевом состоянии, мы получим состояние вида

$$\phi_0 = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |e_i\rangle. \tag{4.1}$$

Это состояние замечательно тем, что все амплитуды в нем одинаковы. В силу сказанного в Главе 2 это последнее свойство однозначно определяет состояние, так как фазовый множитель $e^{i\phi}$ не имеет никакого физического смысла, и мы можем теперь утверждать, что это - единственное состояние с данным свойством.

4.1.2 Операция инверсии и ее реализация на квантовом компьютере

Унитарное преобразование инверсии вдоль состояния a определяется так:

$$I_a|b\rangle = \begin{cases} b, & a \ \ b, \\ -a, & a = b. \end{cases} \tag{4.2}$$

Такое определение формально является неполным, но в силу линейности унитарных операций оно определено данным равенством однозначно. Представив ситуацию геометрически, мы поймем, что это преобразование есть зеркальное отражение всего пространства состояний относительно подпространства, ортогонального вектору a. Как реализовать такую операцию на квантовом компьютере?

Рассмотрим сначала тот случай, когда вектор a нам известен. Для простоты положим $a = 0$ (случаи других базисных состояний отличаются от этого не существенно). Самая простая идея - устроить наш алгоритм так, чтобы он просмотрел последовательно все разряды a на предмет поиска единиц, и если нашел хоть одну, то ничего не делает, а если не нашел ни одной, т.е. $a = 0$, то изменяет знак состояния. Последовательный просмотр устроить очень просто, когда компьютер у нас классический. В квантовом случае есть небольшая трудность, поскольку единиц может быть несколько, и значит при проходе через все число a мы должны как-то это фиксировать, чтобы у нас получилась последовательность только унитарных операций. Наметим такой план. Устроим специальный дополнительный кубит, называемый res, который будет нам сигналить о том, что встретилась единица в числе a. Кроме того, для соблюдения унитарности заведем для каждого значащего кубита его двойник. Все эти дополнительные кубиты первоначально будут находиться в состоянии 0. Проходов будет два: на первом мы выясним, совпадает ли a с нулевой строкой, а на втором, двигаясь в обратном направлении и совершая обратные преобразования, восстановим содержимое всех дополнительных кубитов, т.е. сделаем их снова нулевыми. Эта последняя операция необходима для того, чтобы делать эту процедуру неоднократно. Такая "очистка мусорной корзины"имеет в квантовом компьютинге и иной смысл, аналогов которому нет в классическом случае. Дело в том, что дополнительные кубиты находятся со значащими в запутанном состоянии, а это

значит, что, например, их наблюдение может самым неприятным образом отразиться на кубитах значащих, иначе говоря, испортить необходимую нам информацию.

Итак, мы будем на каждом шаге прохода делать специальную фиксированную операцию V над тремя кубитами: основным, его двойником, и res. На эти кубиты будут указывать специальные указатели. После чего сдвигаем указатели основного кубита и его двойника сдвигаются на один шаг вправо и все повторяется. После первого прохода содержимое res будет сигнализировать о том, встретилась ли нам хотя бы раз единица, или нет. После этого следует условная смена знака, и наконец - все в обратном порядке для чистки мусора. Дело за тем, чтобы определить операцию V. Она должна изменять содержимое res, если первый раз встретила единицу, и при этом быть унитарной. Здесь нам не обойтись без кубитов- двойников. Немного размышлений - и будет видно, что V должна действовать примерно так:

$$\begin{array}{lll} 0\,0\,0 & \longrightarrow & 0\,0\,0 \\ 1\,0\,0 & \longrightarrow & 1\,0\,1 \\ 0\,0\,1 & \longrightarrow & 0\,0\,1 \\ 1\,0\,1 & \longrightarrow & 1\,1\,1 \end{array}$$

На всех остальных состояниях действие этой операции доопределяется так, чтобы она была взаимно-однозначной. Таким образом, операция инверсии относительно нулевого состояния выразится такой последовательностью рабочих преобразований:

$$V_n V_{n-1} \dots V_1 \text{ Sign } V_1^{-1} V_2^{-1} \dots V_n^{-1},$$

где V_j обозначает операцию V совершенную в момент j, а Sign имеет в базисе $|0\rangle, |1\rangle$ матрицу вида

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

В результате мы будем иметь переход от состояния вида $\psi \bigotimes \bar{0}$ к состоянию $\psi' \bigotimes \bar{0}$, где первый сомножитель отличается от второго только знаком при $|e_0\rangle$.

Теперь нам ничего не составляет построить инверсию относительно состояния ϕ_0, определенного равенством (4.1). Используя известное определение Уолша-Адамара, мы получаем равенство: $I_{\phi_0} = W^n I_{\bar{0}} W^n$. Следовательно, и эту инверсию легко можно реализовать на квантовом компьютере. Можно сделать и более общее утверждение, которое легко проверить непосредственно. Пусть вектора $|\bar{a}\rangle$ и $|\bar{b}\rangle$ связаны соотношением $|\bar{a}\rangle = U|\bar{b}\rangle$. Тогда инверсии относительно них будут связаны соотношением $I_{\bar{a}} = U I_{\bar{b}} U^{-1}$. Таким образом, мы можем совершать инверсии относительно любых векторов, если известно, как эти вектора получаются из нулевого. Например, можно случайным образом выбирать преобразование U, и тогда мы будем в состоянии совершить инверсию вдоль случайно выбранного вектора. Отметим, что пока все вектора состояний, относительно которых совершались инверсии, были нам известны в том смысле, что мы располагали способом получить эти состояния нашей квантовой памяти.

А можно ли совершить инверсию относительно того вектора, который нам не известен? Классический компьютер сделать инверсию смог бы только, предварительно найдя этот вектор. Для квантового находить сам вектор нет нужды - достаточно знать, что он является решением уравнения $f(x) = 1$, и иметь в распоряжении оракул для f. Мы сначала опишем не самый хороший способ сделать это. Для простоты мы сначала ограничимся случаем, когда решение уравнения $f(x) = 1$ ровно одно. Обозначим это решение (и соответствующее ему базисное состояние) через tar. Нашей

задачей является построение подпрограммы, реализующей I_{tar}. Применим квантовый оракул, соответствующий f, и имеющий вид (3.3). После этого изменим знак в зависимости от того, совпадает b с единицей или нет, а затем снова применим оракул. Нетрудно видеть, что последнее действие очистит содержимое b, так что знак изменится именно так как надо. В чем недостаток этого метода? Дело в том, что оракул здесь применяется дважды. Можно ли добиться того же эффекта, применив его только один раз?

Оказывается, можно. Дополнительный кубит, или как его называют, анциллу, можно инициализировать в любом состоянии, а не только в нулевом. Важно, чтобы это состояние восстанавливалось после операций. Инициализируем кубит, соответствующий b, в состоянии $\psi_0 = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Тогда после операции Qu_f вида (3.3) мы получим, что к анцилле прибавляется 1 тогда и только тогда, когда a совпадает с искомым корнем уравнения. Но из вида анциллы следует, что при этом как раз будет изменяться ее знак, поскольку ноль и единица меняются местами. Такой способ требует только одного применения оракула, и потому является оптимальным.

Итак, мы познакомились с первым чудом квантовой информатики - возможностью инвертировать пространство состояний относительно вектора, который нам неизвестен!

4.1.3 GSA

Теперь мы уже готовы к тому, чтобы описать метод быстрого квантового перебора. Этот метод удивительно прост: он состоит в последовательном применении преобразования $G = -I_{\phi_0} I_{tar}$[1] к вектору ϕ_0 $\quad t_0 = \left[\frac{\pi\sqrt{N}}{4}\right]$ раз. Получившееся при этом состояние практически совпадает с искомым tar. Чтобы убедиться в этом, рассмотрим, как оператор G действует на произвольный вектор. Проще всего задействовать геометрическое воображение. Во первых, нетрудно понять, что преобразование G преобразует плоскость, порожденную векторами $|tar\rangle$ и ϕ_0 в эту же плоскость (это следует из того, что обе инверсии не выведут нас за пределы этой плоскости). Во-вторых, одна инверсия меняет ориентацию, значит, две инверсии ее сохраняют. Таким образом, G есть ортогональное преобразование плоскости (унитарное, и без комплексных коэффициентов). Следовательно, G есть не что иное, как поворот этой плоскости. Мы приходим к выводу, что G представляет собой поворот на острый угол α, который образован вектором ϕ и вектором $\tilde{0}$, ортогональным $|tar\rangle$. Мы имеем $\alpha \approx \sin\alpha = |\langle tar|\phi\rangle| = 1/\sqrt{N}$, с точностью до $1/N^{3/2}$. Значит, если мы применим G t_0 раз, с точностью до $1/N$ результат совпадет с искомым вектором tar. После этого узнать искомое состояние можно будет, просто измерив состояние квантовой части компьютера.

4.1.4 Как искать решения, если их много?

До сих пор мы занимались случаем единственного решения нашего уравнения $f(x) = 1$. Что изменится, если решений несколько? Во-первых, можно заметить, что если решений очень много, например их число сравнимо с числом всевозможных значений x, то применение квантового компьютера для поиска любого из этих решений лишено всякого смысла, так как можно просто выбрать x наугад. Итак, будем считать, что решений не очень много, например, можно считать, что их число $O(\sqrt{N})$.

Заметим, что формально мы можем применить GSA и для этого случая. Обе инверсии, входящие в его состав, можно формально выполнить и в случае многих решений. Однако теперь геометрический смысл инверсии I_{tar} будет уже несколько иной. Пусть L_f обозначает подпространство,

[1] Знак минус в определении G взят только для красоты.

порожденное всеми решениями $f(x) = 1$. Тогда I_{tar}, которое производится точно так же , как и в случае единственного решения, будет зеркальным отражением всего пространства относительно подпространства, ортогонального к L_f. Действительно, каждое решение будет менять знак, значит и каждая линейная комбинация решений будет менять знак, а вектор, перпендикулярный ко всем решениям, останется без изменений.

Сделаем еще одно поучительное наблюдение. Как нам известно, инверсию на квантовом компьютере можно совершать не только относительно состояния ϕ_0, но и относительно любого состояния $\tilde{\phi}_0$, которое мы можем получить для квантовой части компьютера. Рассмотрим проекцию начального состояния $\tilde{\phi}_0$ на подпространство L_f, и пусть теперь tar обозначает новое целевое состояние, направленное вдоль этой проекции (оно отличается от самой проекции только тем, что имеет единичную длину). Тогда мы снова, как и в предыдущем случае, получаем, что плоскость, образованная векторами tar и $\tilde{\phi}_0$ переходит в себя же при применении Гроверовского преобразования. В нашем геометрическом рассуждении ничего не изменится, если мы просто заменим ϕ_0 на $\tilde{\phi}_0$. Единственное, что придется нам теперь подправить, это время t_0. Рассуждая по аналогии с предыдущим случаем, мы быстро поймем, что теперь время должно быть $\left[\frac{\pi}{4|\langle tar|\tilde{\phi}_0\rangle|}\right]$.

Отсюда следует интересный вывод. В схеме Гровера мы можем использовать инверсию относительно любого начального состояния ϕ_0, даже выбранного наугад! Действительно, при случайном выборе этого состояния модуль скалярного произведения $|\langle tar|\tilde{\phi}_0\rangle|$ будет иметь порядок $1/\sqrt{N}$. Квадрат этого числа есть вероятность получения состояния tar при измерении состояния $\tilde{\phi}_0$ в каком либо базисе, содержащем наш вектор tar. Из соображений симметрии, при случайном выборе начального состояния $\tilde{\phi}_0$ эта вероятность должна быть одинакова для всех векторов этого базиса, и значит, должна равняться $1/N$. Из этого, строго говоря еще не следует, что вероятность получения квадрата из этой величины будет равна $1/\sqrt{N}$[2]. Однако можно доказать, что среднее значение самого модуля будет иметь тот же порядок величины, а именно $1/\sqrt{N}$ с точностью до умножения на константу, не зависящую от N. Так что мы можем смело применять Гроверовскую схему для быстрого получения состояний, если у нас есть в распоряжении какой-либо аппарат, осуществляющий инверсию относительно этого состояния. При этом начальный вектор и соответствующее ему преобразование инверсии можно выбирать произвольно, случайным образом.

Однако остается неясным, когда же все-таки производить конечное измерение? Для этого конечно надо знать $|\langle tar|\tilde{\phi}_0\rangle|$. Иначе мы окажемся в положении пассажира на кольцевой железной дороге, который не знает точно, где ему выходить из поезда. Если мы точно знаем количество l всех решений, немного подумав, мы можем сообразить, что этот модуль равен $\sqrt{l/N}$. А что делать, если число решений неизвестно, или если мы просто хотим по схеме Гровера получить искомое состояние tar, используя соответствующую инверсию? Здесь мы встречаемся с тем случаем, когда удобно применить промежуточные измерения.

4.1.5 Когда бывает удобно часто измерять

Итак, пусть мы не знаем общего числа решений. Если мы остановим наш процесс применения Гроверовского преобразования в произвольный момент, меньший $C\sqrt{N}$, где C достаточно большое число, с какой вероятностью мы получим при измерении искомое нами состояние? Немного подумав, мы должны сообразить, что эта вероятность не должна быть намного меньше $1/2$, поскольку наш текущий вектор состояния и вектора tar и $-tar$ с примерно равными вероятностями образуют

[2]Для тех, кто знает теорию вероятностей, должно быть просто понять, что разница между средним квадрата величины и квадратом среднего ее значения как раз будет дисперсией этой величины.

углы большие и меньшие $\pi/4$. Так что если мы не гонимся за максимально быстрым методом получения целевого состояния, мы можем просто запустить процесс GSA и примерно за $O(\sqrt{N})$ шагов получить ответ с высокой вероятностью, используя повторные попытки. Но при этом мы наверняка упустим возможность быстрого получения ответа тогда, когда в действительносьти решений много. Например, при $l = O(\sqrt{N})$ упущенная выгода составит $O(N^{1/4})$. Здесь нам на помощь придет очень простая идея: сканировать область времени - до $C\sqrt{N}$ включительно, всякий раз выбирая интервал времени, например, в два раза больший, чем на предыдущем шаге. При этом на каждом шаге будем дополнительно применять измерение в произвольно выбранный момент из текущего интервала. Тогда мы, разумеется, не упустим выгоду в том случае, если решений много - ответ будет получен за $O(\sqrt{N/L})$ шагов, как и должно. Это следует из того, что время, потраченное почти впустую на измерения при малых временных интервалах до того, как текущий интервал будет достаточно велик, будет примерно такое же, как и в достаточно большом интервале, ведь сумма геометрической прогрессии примерно совпадает с ее наибольшим членом.

Итак, мы познакомились со вторым чудом квантовой информатики - с возможностью сделать перебор вариантов, не перебирая их все! Эта удивительная возможность целиком опирается на фундаментальные свойства квантовых объектов - интерференцию амплитуд и существование запутанных состояний. Можно показать (см. например, []), что всякое квантовое вычисление, не использующее запутанности, т.е. оперирующее с независимыми кубитами, может быть легко реализовано и на классическом компьютере. В начале и в конце вычисления по схеме GSA мы имеем незапутанные состояния, такие как tar или ϕ_0. Однако в середине этой процедуры итераций запутанность растет, дорастает до некоторого предела, а затем начинает падать до нуля. Итак, запутанность в квантовой информатике является своеобразным и абсолютно необходимым ресурсом, таким же, как солнечный свет для всех живых существ. Физическая природа этого ресурса пока до конца не понятна, и ее исследование представляет собой интереснейшую открытую проблему естествознания.

Структурный поиск

Может ли дополнительная информация о решении уравнения $f(x) = 1$ помочь при его квантовом решении? Мы знаем, что в классическом случае ответ положительный, но применить такого роды информацию в квантовом вычислении - задача непростая. Опишем сначала очевидный, но бесперспективный путь. Пусть g есть некоторый булевский оракул, такой что множество решений $g(x) = 1$ содержит всен ерешения уравнения $f(x) = 1$. Рассмотрим оператор стандартного GSA: GSA $= (WI_0WI_{tar})^t$ где $t = \left[\frac{\pi\sqrt{N}}{4}\right]$ и N есть число всех базисных состояний. Можно просто проверить, что все, что нам нужно от оператора - это что он отображает нуль- состояние в $\frac{1}{|sqrtN}\sum_{\bar{x}}\bar{x}$ и наоборот. В этом случае мы можем найти искомое состояние, применяя GSA к состоянию $W\bar{0}$. Пусть $\mathcal{M}$ есть множество всех искомых состояний, I_{tar} обозначает инверсию всех состоянийиз $\mathcal{M}$. Обозначаем оператор GSA через GSA)$\mathcal{M}$. теперь мы можем использовать вместо W другой оператор типа GSA, индуцированный функцией g, так как такой оператор удовлетворяет этому условию с некоторой малой ошибкой. Убедитесь, что этот прием не дает преимущества по сравнению с обычным гроверовским ускорением. Однако нечто новое случится, если мы рассмотрим двумерно организованное пространство базисных состояний и функцию $f(x, y)$ с единственной точкой (x_0, y_0) где она принимает значение 1, и вспомогательную функцию $g(x)$, такую что x_0 есть одно из решений уравнения $g(x) = 1$. Для этой задачи, которая называется структурным перебором, существует дополнительное по сравнению с квадратным корнем квантовое ускорение. Эта красивая конструкция предложена Фари и Гутманом (см. работу [**?**]) и мы ее сейчас изучим.

Проблема структурного перебора формулируется так. Рассмотрим множество базисных состояний вида $S = \{0, 1, \ldots, L-1\} \times \{0, 1, \ldots, L-1\}$ и пусть x и y обозначают элементы из множества $\{0, 1, \ldots, L-1\}$. Тогда $|x, y\rangle$ будет базисным состоянием. Пусть $F(x, y)$ и $G(x)$ есть две булевские функции, такие что уравнение $F(x, y) = 1$ имеет единственное решение x_0, y_0 а уравнение $G(x) = 1$ имеет M решений: $x_1, x_2, \ldots, x_M$, где x_0 - одно из них. Обозначим множество этих решений через $\mathcal{M}$. Примем, что нам известно L и M и $1 \ll M^2 \ll L$. Проблема состоит в нахождении x_0, y_0. Классически это требует $O(LM)$ шагов: перебирать все x и для таких x, которые удовлетворяют $G(x) = 1$, пробовать всевозможные значения y.

Теперь рассмотрим квантовый алгоритм для этой проблемы. Сначала определим вспомогательный оператор W_x, отображающий $\bar{0}$ в состояние $\frac{1}{\sqrt{M}} \sum_{j=1}^{M} x_j$ и наоборот. Мы можем положить $W_x = \mathrm{GSA}_{\mathcal{M}}$. Затем так определенный оператор будент удовлетворять требуемому условию с точностью $O(\sqrt{\frac{M}{L}})$ и будет иметь сложность $O(\sqrt{\frac{L}{M}})$. Временно отложим вопрос о точности и будем рассматривать так определенный оператор W_x как удовлетворяющий нужному условию точно.

Теперь определим второй вспомогательный оператор V со следующим свойством:

$$V|x\rangle = \begin{cases} |x\rangle, & \text{if } x \neq x_0, \\ -x_0\rangle, & \text{if } x = x_0. \end{cases} \tag{4.3}$$

Теперь свою роль сыграет вторая "координата"y. Обозначим через A оператор, присоединяюший $y = 0$ к рабочему регистру, содержащему x. (Мы могли бы обойтись вообще без этого оператора, если условимся, что все наши операторы действуют на S). В следующем вспомогательном определении все операторы, кроме I_{tar} будут действовать на второй компоненте базисного состояния $|x, y\rangle$.

$$V_1 = (WI_0WI_{tar})^{\tau}WA,$$

где τ есть ближайшее четное натуральное число к $\left[\frac{\pi\sqrt{L}}{4}\right]$. Этот оператор таков что

$$V_1|x\rangle = \begin{cases} |x_0, y_0\rangle, & \text{if } x = x_0, \\ |x, \bar{0}\rangle \end{cases} \tag{4.4}$$

Заметим, что обратный оператор $V^{-1} = A^{-1}W(WI_0WI_{tar})^{\tau}$ может быть также реализован на квантовом компьютере. Мы можем считать, что $\bar{0} \neq y_0$, поскольку в противном случае задача отыскания x_0, y_0 была бы тривиальной. Наш второй вспомогательный оператор V можно определить как

$$V = V_1^{-1}I_{tar}V.$$

Проверка требуемого свойства делается непосредственно.

Имея два оператора W_x и V, мы можем определить наш результирующий оператор, дающий требуемое состояние $|x_0\rangle$ через

$$(W_xVW_xI_0)^{\tau'}W_x$$

где $\tau' = \left[\frac{\pi\sqrt{M}}{4}\right]$. Это есть GSA_{x_0} оператор, и следовательно его применение к начальному вектору из нулей даст $|x_0\rangle$ высокой вероятностью. Так определенный результирующий оператор требует $O((\sqrt{L} + \sqrt{\frac{L}{M}})\sqrt{M})$ шагов, что есть $O(\sqrt{LM})$.

Теперь займемся отложенным вопросом о точности. Только что определенная схема будет работать, если W_x удовлетворяет соответствующему условию точно. Но напомним, что оператор W_x, определенный выше, удовлетворяет этому условию с точностью $(1/\sqrt{M})$. Если мы теперь применим наш оператор GSA_{x_0} к начальному состоянию, то ошибка составит $\sqrt{M}O(\sqrt{\frac{M}{L}}) = o(1)$ ввиду нашего условия $M^2 \ll L$. Мы видим, что алгоритм для структурного поискадает то же ускорение по сравнению с классическим случаем, что и GSA - квадратный корень из классического времени.

Итерационный поиск

Рассмотрим такую ситуацию. Мы собираем мозаику из разбросанных камней в прямоугольной области по имеющейся у нас картинке. Каждый камень имеет уникальную форму. Мы можем собирать мозаику слой за слоем< используя простой перебор среди еще разбросанных камней и заполнять каждый уровень, основываясь на предыдущем, уже собранном. Тогды мы фактически выполняем итерационный поиск классически, так как для того, чтобы подбирать камни к данному слою, надо уже иметь предыдущий слой собранным.

Мы формализуем эту ситуацию как специальный тип итерационного алгоритма: итерационный поиск (IS). Пусть мы имеем последовательность $S_1, S_2, \ldots, S_k$ одинаковых переборных проблем, где проблема S_i состоит в нахождении единственного решения x_i^0 уравнения $f_i(x_i) = 1$ где булевская функция f_i доступна нам тогда и только тогда, когда мы знаем все $x_j^0, \quad j < i$. Пусть $|x_i| = n, \quad N = 2^n, \quad k \ll N$, $|x|$ обозначает длину слова x. Цель состоит в поиске x_k^0, $k \geq 2$. Ввиду результата работы [BBHT] последовательное применение гроверовского алгоритма для $x_1^0, x_2^0, \ldots, x_k^0$ дает ответ за время $\frac{k\pi\sqrt{N}}{4}$ с вероятностью ошибки около k/N. Чтобы это реализовать, мы должны иметь все оракулы $f_i, \quad i = 1, 2, \ldots, k$, где зависимость f_i от всех $x_j, \quad j < i$ может быть включена в f_i. Таким образом, мы можем предположить, что f_i имеет фвид $f_i(x_1, x_2, \ldots, x_i)$ и каждое равенство $f_i(x_1, \ldots, x_i) = 1$ имеет единственное решение $x_1^0, x_2^0, \ldots, x_i^0, \quad i = 1, 2, \ldots, k$. Рассматривая все оракулы f_i как физические устройства, которые нельзя клонировать, мы получаем, что все они должны быть в нашем распоряжении одновременно, для того, чтобы мы могли их применить одновременно. Здесь преимущество получается из-за того, что при одновременном применении этих оракулов возникает квантовая интерференция, которая и дает дополнительное ускорение по сравнению с последовательным режимом. Как возникает это ускорение? Оно возникает из-за утечки амплитуды на каждом шаге последовательного поиска. Амплитуда состояния x_i^0 в поиске номер i возрастает шаг за шагом в течение гроверовского поиска, так что после нескольких первых l шагов она становится примерно равной $\frac{2l+1}{\sqrt{N}}$, а амплитуда других $x_i \neq x_i^0$ убывает. Это преимущество в величине амплитуды состояния x_i^0 (утечка амплитуды) может быть немедленно использована для следующего $i+1$-го поиска.

Мы увидим, как этот эффект можно использовать для решения задачи итерационного поиска за время $O(\frac{k\pi\sqrt{N}}{4\sqrt{2}})$ то есть в $\sqrt{2}$ раз быстрее, чем за k последовательных применений гроверовского алгоритма. Подчеркнем, что это ускорение не требует дополнительных устройств, эффект достигается за счет одновременного действия оракулов. Так мы получим модификацию быстрого квантового поиска - параллельный алгоритм для итерационного поиска, который будет описан ниже. В этом параграфе мы в основном занимаемся частным случаем $k = 2$ of IS, который мы называем проблемой повторного поиска (RS).

Сравним параллельный алгоритм для RS с алгоритмом гнездного квантового поиска (NQS) представленным в работе [?]. Для случая единственного решения для любого f_i NQS эквивалентен последовательному применению простого квантового поиска. Это значит, что NQS не использует

квантовую интерференцию между двумя процессами поиска. Основное достоинство параллельного алгоритма для RS, представленного здесь, состоит в использовании интерференции между поисками решений уравнений $f_1(x) = 1$ и $f_2(x, y) = 1$ и таким образом этот метод достигает $\sqrt{2}$-кратного ускорения.

RS-проблема связана с известной проблемой структурного поиска (SS). Проблема структурного поиска состоит в нахождении единственного решения x_0, y_0 уравнения $f(x, y) = 1$, при условии что мы имеем функцию g, чей носитель $\{x \mid g(x) = 1\}$ мощности M содержит x_0. RS-проблема представляет собой частный случай SS, когда $M = 1$. Случай $1 \ll M \ll N$ был рассмотрен Фари и Гутманом в работе [FG]. Они нашли квантовый алгоритм для этого случая со сложностью $O(\sqrt{MN})$, и также отметили, что наилучшей известной стратегией для случая $M = 1$ является последовательное применение гроверовского алгоритма. Здесь мы покажем, как эта очевидная стратегия может быть улучшена в $\sqrt{2}$ раз. Отметим, что мы используем идею, отличную от идеи [FG]. Если в упомянутой работе использовались только алгебраические свойства гроверовского алгоритма, то мы будем использовать сам вид эволюции амплитуд в ходе вычислений.

Мы будем использовать то обстоятельство, что в нашей модели квантовых вычислений несколько оракулов могут вызываться одновременно, что приводит к интерференции результатов их действия. Именно эта интерференция и приводит к дополнительному ускорению вычислений в случае повторного поиска.

Пусть $u,\ x,\ y$ будут переменными со значениями из трех разных копий $\mathcal{H}_0 = \mathrm{C}^N$, $a = a_1 \bigotimes a_2 \in \mathrm{C}^4$, где $a_1 = a_2 = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Мы примем обозначения $f_1(x),\ f_2(x, y)$ для двух оракулов из повторного квантового поиска и пусть e_1, e_2 будут теми значениями для x, y, которые представляют единственные решения уравнений $f_1 = 1,\ f_2 = 1$. Обозначим соответствующие состояния квантовой части компьютера теми же буквами.

Положим $\mathcal{H} = \mathcal{H}_0 \bigotimes \mathcal{H}_0 \bigotimes \mathcal{H}_0 \bigotimes \mathrm{C}^4$. Let

$$\begin{aligned} F_1|u, x, y, a\rangle &= |u, x, y, a_1 \bigoplus f_1(u), a_2\rangle, \\ F_2|u, x, y, a\rangle &= |u, x, y, a_1, a_2 \bigoplus f_2(x, y)\rangle, \\ P|u, x, y, a\rangle &= |u \bigoplus x, x, y, a\rangle. \end{aligned}$$

Тогда

$$F_1|u, x, y, a\rangle = \begin{cases} |u, x, y, a\rangle, & \text{if } u \neq e_1, \\ -|u, x, y, a\rangle, & \text{if } u = e_1; \end{cases}$$

$$F_2|u, x, y, a\rangle = \begin{cases} |u, x, y, a\rangle, & \text{if } |x, y\rangle \neq |e_1, e_2\rangle, \\ -|u, x, y, a\rangle, & \text{if } |x, y\rangle = |e_1, e_2\rangle; \end{cases}$$

Определим такое вспомогательное унитарное преобразование на $\mathcal{H}$: $\mathcal{R}_0 = I \bigotimes R_{0x} \bigotimes R_{0y} \bigotimes I;\ \mathcal{W} = I \bigotimes W_x \bigotimes W_y \bigotimes I;\ \ \mathcal{F} = P(F_1 \mid_{u,a_1} \bigotimes F_2 \mid_{x,y,a_2})P$, где нижний индекс x, y обозначает соответствующую область для применения преобразования Уолша-Адамара и ротаций фазы 0, а I обозначает идентичное преобразование.

Ключевое унитарное преобразование параллельного алгоритма для RS будет таким

$$Z = \mathcal{W}\mathcal{R}_0\mathcal{W}\mathcal{F}. \tag{4.5}$$

Параллельный алгоритм для RS есть последовательное применение оператора Z начиная с начального состояния

$$\chi_0 = |\bar{0}\rangle \bigotimes \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |e_i\rangle \bigotimes \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |e_i\rangle \bigotimes a$$

$\left[\frac{\pi\sqrt{N}}{2\sqrt{2}}\right]$ раз.

Теорема 1 *Пусть* $t = \left[\frac{\pi\sqrt{N}}{2\sqrt{2}}\right]$. *Тогда измерение* $Z^{\{t\}}(\chi_0)$ *дает* $x = e_1,\ y = e_2$ *с вероятностью ошибки* $O(\frac{1}{\sqrt{N}})$.

Из определения Z следует, что оракулы F_1 и F_2 для функций $f_1,\ f_2$ работают одновременно и параллельнотак что алгоритм требует примерно $\left[\frac{\pi\sqrt{N}}{2\sqrt{2}}\right]$ одновременных запросов для нахождления результата, тогда как последовательное применение простого квантового поиска с оракулом f_1 а затем с f_2 требует $\left[\frac{\pi\sqrt{N}}{2}\right]$ шагов для нахождения результата с той же вероятностью. Заметим, что для обыкновенного поиска GSA мультипликативная константа $\frac{\pi\sqrt{N}}{4}$ не может быть существенно уменьшена (см. [BBHT]).

Предположим, что каждый квантовый поиск выполняется на физическом устройстве (оракуле) специального вида, соответствующего форме запроса. Мы таким образом рассматриваем оракулы как тип устройств. Минимальный набор оракулов, достаточный для решения проблемы RS состоит из одного оракула для f_1 и одного для f_2. С этими оракулами мы можем запустить их параллельно и получить результат в $\sqrt{2}$ раз быстрее чем при последовательном поиске. Но если у нас есть *две копии* каждого оракула, то возможно будет достичь той же производительности и при последовательном поиске, если мы разобьем целую область $\{0,1\}^n$ на две равные части по $N/2$ элементов каждая и применим GSA к сначала с двумя копиями f_1-оракула - по одному для каждой половинки области, а потом, найдя e_1, повторим GSA с двумя копиями f_2-оракула таким же образом. Этот путь будет более дорогим, если каждая копия оракула имеет высокую цену, или вообще невозможным, если каждый оракул уникален, например, основан на некотором природном феномене. Именно в этом случае минимального возможного набора устройств для оракулов $f_1,\ f_2$ применение параллельного квантового алгоритма для RS дает увеличение производительности в $\sqrt{2}$ раз. Это ускорение можно также получить и для IS если применить этот алгоритм для пар $f_i, f_{i+1},\ i = 1, 2, \ldots, k-1$. Результирующая вероятность ошибки будет составлять $O(k/\sqrt{N})$.

Основная особенность параллельного алгоритма - ускорение без дополнительных устройств. Это важно, поскольку гроверовский алгоритм для простого квантового поиска не может быть улучшен даже по мультипликативной константе.

Оставшаяся часть этого параграфа посвящена доказательству Теоремы и перспективам этого подхода.

Заметим, что каждый из $W_y,\ R_{0y}$ коммутирует с $W_x,\ R_{0x},\ P,\ F_1$; P коммутирует с F_2, поэтому Z можно представить как

$$Z = -[(I \bigotimes W_x R_{0x} W_x \bigotimes I) P F_1 P][-(I \bigotimes (W_y R_{0y} W_y) \bigotimes I) F_2],$$

или как

$$Z = \{-W_x R_{0x} W_x \mathcal{F}_1\}\{-W_y R_{0y} W_y F_2\}, \tag{4.6}$$

где

$$\mathcal{F}_1|u,x,y,a\rangle = \begin{cases} |u,x,y,a\rangle & \text{if } x \neq e_1, \\ -|u,x,y,a\rangle & \text{if } x = e_1. \end{cases}$$

Вид (9) представляет собой в точности повторный GSA с оракулами $F_2,\ \mathcal{F}_1$ в этом порядке, так что мы можем применить формулы (4) для результирующих амплитуд этих преобразований. Пусть Z-итерации будут $\chi_0 \longrightarrow \chi_1 \longrightarrow \ldots \longrightarrow \chi_t$, $\chi_{i+1} = Z(\chi_i),\ i = 0, 1, \ldots, t-1$;

$$\chi_i = b_i|e_1e_2\rangle + a_i|e_1N_2\rangle + \alpha_i|N_1N_2\rangle + \beta_i|N_1e_2\rangle, \tag{4.7}$$

где e_1 и e_1, e_2 есть искомые состояния: единственные решения $f_1(x) = 1$ и для $f_2(x, y) = 1$ соответственно, $N_1 = \sum_{i=2}^{N} e_i$, $N_2 = \sum_{i\neq 2} e_i$ (мы опускаем вспомогательные кубиты).

Мы представим оператор $\chi_i \longrightarrow Z(\chi_i)$ как два последовательных шага:

$$\chi_i \xrightarrow{1} Z_1(\chi_i) = \chi_i' \xrightarrow{2} Z_2(\chi_i') = \chi_{i+1},$$

где $Z_1 = -W_yR_{0y}W_yF_2$, $Z_2 = -W_xR_{0x}W_x\mathcal{F}_1$. Чтобы подсчитать изменение амплитуды, происходящее из-за применения Z_1 (или Z_2) мы зафиксируем значение x (или y соответственно).

Шаг 1. Обозначим амплитуды базисных состояний в χ_i' через соответствующие буквы со штрихами:

$$\chi_i' = b_i'|e_1e_2\rangle + a_i'|e_1N_2\rangle + \alpha_i'|N_1N_2\rangle + \beta_i'|N_1e_2\rangle.$$

Тогда для двух существенно различных путей фиксации базисного состояния x: $x = e_1$ or $x+e_j$, $j \neq 1$ мы будем иметь разные выражения для новых амплитуд. Используем свойство преобразования диффузии WR_0W быть инверсией относительно среднего (см. [Gr1]). Пусть λ_{av} есть амплитуда соответствующего квантового состояния.

a). $\underline{x = e_1.}$

$\lambda_{av} = \frac{(N-1)a_i - b_i}{N}$, $b_i' = 2\lambda_{av} + b_i$, $a_i' = 2\lambda_{av} - a_i$,

$$\begin{array}{ll} b_i' = \frac{2(N-1)a_i - 2b_i}{N} + b_i & = b_i(1 - \frac{2}{N}) + 2a_i(1 - \frac{1}{N}), \\ a_i' = \frac{2(N-1)a_i - 2b_i}{N} - a_i & = -b_i\frac{2}{N}) + a_i(1 - \frac{2}{N}). \end{array}$$

b). $\underline{x = e_j,\ j \neq 1.}$

$\lambda_{av} = \frac{(N-1)\alpha_i + \beta_i}{N}$, $\alpha_i' = 2\lambda_{av} - \alpha_i$, $\beta_i' = 2\lambda_{av} - \beta_i$,

$$\begin{array}{ll} \alpha_i' = \frac{2(N-1)\alpha_i + 2b_i}{N} - a_i & = \alpha_i(1 - \frac{2}{N}) + 2\beta_i\frac{2}{N}), \\ \beta_i' = \frac{2(N-1)\alpha_i + 2\beta_i}{N} - \beta_i & = \alpha_i(1 - \frac{1}{N}) - \beta_i(1 - \frac{2}{N}). \end{array}$$

Шаг 2. $\chi_i' \xrightarrow{2} Z_1(\chi_i') = \chi_{i+1}$.

У нас есть два разых пути фиксации базисного состояния y : $y = e_2$ или $y = e_j,\ j \neq 2$.

a). $\underline{y = e_2.}$

$$\lambda_{av} = \frac{(N-1)\beta_i' - b_i'}{N}, \quad \begin{array}{ll} b_{i+1} & = 2\lambda_{av} + b_i' = b_i'(1 - \frac{2}{N}) + 2\beta_i'(1 - \frac{1}{N}), \\ \beta_{i+1} & = 2\lambda_{av} - \beta_i' = \beta_i'(1 - \frac{2}{N}) - b_i'\frac{2}{N}. \end{array}$$

b). $\underline{y = e_j,\ j \neq 2.}$

$$\lambda_{av} = \frac{(N-1)\alpha_i' - a_i'}{N}, \quad \begin{array}{ll} a_{i+1} & = 2\lambda_{av} + a_i' = a_i'(1 - \frac{2}{N}) + 2\alpha_i'(1 - \frac{1}{N}), \\ \alpha_{i+1} & = 2\lambda_{av} - \alpha_i' = \alpha_i'(1 - \frac{2}{N}) - a_i'\frac{2}{N}. \end{array}$$

Следовательно, реккурентные формулы для амплитуд последовательных шагов 1,2 приобретают вид:

$$\begin{array}{ll} b_{i+1} & = b_i(1-\frac{2}{N})^2 + 2a_i(1-\frac{1}{N})(1-\frac{2}{N}) + 4\alpha_i(1-\frac{1}{N})^2 - 2\beta_i(1-\frac{2}{N})(1-\frac{1}{N}); \\ a_{i+1} & = a_i(1-\frac{2}{N})^2 - b_i\frac{2}{N}(1-\frac{2}{N}) + 2\alpha_i(1-\frac{2}{N})(1-\frac{1}{N}) + 2\beta_i\frac{2}{N}(1-\frac{1}{N}); \\ \alpha_{i+1} & = \alpha_i(1-\frac{2}{N})^2 + \beta_i\frac{2}{N}(1-\frac{2}{N}) - a_i(1-\frac{2}{N})\frac{2}{N} + b_i\frac{4}{N^2}; \\ \beta_{i+1} & = 2\alpha_i(1-\frac{1}{N})(1-\frac{2}{N}) - \beta_i(1-\frac{2}{N})^2 - b_i(1-\frac{2}{N})\frac{2}{N} - 2a_i(1-\frac{1}{N})\frac{2}{N}. \end{array}$$

Таким образом, матрица одного шага алгоритма имеет вид

$$Z = \begin{pmatrix} 1 & 2 & 4 & -2 \\ -\frac{2}{N} & 1 & 2 & \frac{4}{N} \\ \frac{4}{N^2} & -\frac{2}{N} & 1 & \frac{2}{N} \\ -\frac{2}{N} & \frac{4}{N} & 2 & -1 \end{pmatrix}.$$

Система реккурентных уравнений может быть записана в виде системы дифференциальных уравнений.

$$\begin{array}{ll} b_{i+1} - b_i & = 2a_i + 4\alpha_i - 2\beta_i + b_iO_1(\frac{1}{N}) + a_iO_2(\frac{1}{N}) + \alpha_iO_3(\frac{1}{N}) + \beta_iO_4(\frac{1}{N}); \\ a_{i+1} - a_i & = -\frac{2}{N}b_i + 2\alpha_i + a_iO_5(\frac{1}{N}) + b_iO_6(\frac{1}{N^2}) + \alpha_iO_7(\frac{1}{N}) + \beta_iO_8(\frac{1}{N}); \\ \alpha_{i+1} - \alpha_i & = -\frac{2}{N}a_i + \beta_iO_{13}(\frac{1}{N}) + a_iO_{14}(\frac{1}{N^2}) + b_iO_{15}(\frac{1}{N^2}) + \alpha_iO_{16}(\frac{1}{N}); \\ \beta_{i+1} - \beta_i & = -\frac{2}{N}b_i + 2\alpha_i - 2\beta_i + a_iO_9(\frac{1}{N}) + \alpha_iO_{10}(\frac{1}{N}) + \beta_iO_{11}(\frac{1}{N}) + b_iO_{12}(\frac{1}{N^2}). \end{array} \tag{4.8}$$

4.1.6 Приближение эволюции амплитуд дифференциальными уравнениями

Пусть $\{\bar{c}_i\}$ есть последовательность векторов из C^k : $\bar{c}_i = (c_i^1, c_i^2, \ldots, c_i^k)$, $c_i^j \in \mathrm{C}$, удовлетворяющая следующей системе дифференциальных уравнений

$$\bar{c}_{i+1} - \bar{c}_i = A\bar{c}_i, \tag{4.9}$$

где A есть матрица размера $k \times k$ с комплексными элементами.

Пусть m есть натуральное число и функция $C(t)$: $\mathrm{R} \longrightarrow \mathrm{C}^k$ есть решение системы дифференциальных уравнений

$$\dot{C}(t) = mAC(t) \tag{4.10}$$

с начальным условием

$$C(0) = \bar{c}_0. \tag{4.11}$$

Тогда точное решение задачи Коши (13),(14) будет $C(t) = R(t)\bar{c}_0$, с матрицей - резольвентой $R(t) = \exp(mAt)$. Система (12) будет системой дифференциальных уравнений, приближающая $C(t)$ по методу Эйлера< если мы рассмотрим $\bar{c}_i$ как приближение для $C(i/m)$, $i = 0, 1, \ldots$. Точность такого приближения может быть найдена из формулы Тэйлора $C(\frac{i+1}{m}) = C(\frac{i}{m}) + \frac{1}{m}\dot{C}(\frac{i}{m}) + \frac{1}{2m^2}\ddot{C}(t_1)$, $\frac{i}{m} < t_1 < \frac{i+1}{m}$. Здесь ошибка ϵ_1 на одном шаге рекурсии (12) есть третье слагаемое $\frac{1}{2m^2}\ddot{C}(t_1) = \frac{1}{2}A^2C(t_1)$. Таким образом, ошибка на первом шаге есть $\frac{1}{2}A^2\exp(mA\theta_1)\bar{c}_0$, на втором шаге: $\frac{1}{2}A^2\exp(mA\theta_2)\bar{c}_1 + \exp(mA\frac{1}{m})\frac{1}{2}A^2\exp(mA\theta_1)\bar{c}0 = \frac{1}{2}A^2\exp(mA\theta_2)(\bar{c}_0 + A\bar{c}_0 + \frac{1}{2}A^2\exp(mA\theta_1)\bar{c}_0) + \exp(A)\frac{1}{2}A^2\exp(mA\theta_1)\bar{c}_0$, и т.д., на шаге i ошибка будет $\epsilon_i \leq \frac{3}{2}\sum_{j=1} i\exp(A\alpha_j)A^2\bar{c}_0$, где $0 < \alpha_j < 1$. Значит, если $\|\bar{c}_0\| \leq h$, то ошибка после шага i будет $\epsilon_i = O(ih)$. В частности, для начальных условий $\|c_0\| = O(\frac{1}{N})$ хорошее приближение можно найти, если $i = o(N)$, так мы можем решить задачу Коши вместо (11) для $i = O(\sqrt{N})$ с ошибкой столь малой, сколь это требуется для достаточно больших N.

Определим новую функцию $B(\tau)$ как $B(tm) = C(t)$. В теминах B задача Коши (13), (14) принимает вид

$$\frac{\mathrm{d}}{\mathrm{d}\tau}B(\tau) = AB(\tau),\ B(0) = c_0. \tag{4.12}$$

Применим это к решению $\bar{c}_i = |b_i, a_i, \alpha_i, \beta_i\rangle$ уравнения (11), где $\bar{c}_0 = |\frac{1}{N}, \frac{1}{N}, \frac{1}{N}, \frac{1}{N}\rangle$. Положим $B = |b, a, \alpha, \beta\rangle$ для скалярных функций b, a, β, α и обозначим аргумент функции B через t. Тогда уравнение (15), приближающее (11), примет вид:

$$\begin{array}{ll} \dot{b} & = 2a + 4\alpha - 2\beta + bO_1(\frac{1}{N}) + \epsilon_1 + aO_0(\frac{1}{N}); \\ \dot{a} & = -\frac{2}{N}b + 2\alpha + \epsilon_2 + O_2(\frac{1}{N})a; \\ \dot{\beta} & = -\frac{2}{N}b + 2\alpha - 2\beta + \epsilon_4 + O_4(\frac{1}{N})a; \\ \dot{\alpha} & = -\frac{2}{N}a + \epsilon_3, \end{array} \tag{4.13}$$

где $\epsilon_i = aO_{0i}(\frac{1}{N^2}) + bO_{1i}(\frac{1}{N^2}) + \beta O_{2i}(\frac{1}{N}) + \alpha O_{3i}(\frac{1}{N}),\ i = 1, 2, 3, 4$, с начальным условием

$$b(0) = a(0) = \beta(0) = \alpha(0) = \frac{1}{N}. \tag{4.14}$$

Тогда для $t = O(\sqrt{N}),\ i = [t]$ вектор ошибки будет $\bar{\delta} = \bar{B}(t) - \bar{c}_i = O(1/\sqrt{N}),\ \ N \longrightarrow \infty$ и с этой точностью мы можем написать $b(i) \approx b_i$ для амплитуды b_i искомого состояния $|e_1, e_2\rangle$.

4.1.7 Тонкий анализ параллельного алгоритма повторного квантового поиска RS

Теперь мы займемся системой линейных дифференциальных уравнений (16) с начальным условием (17). Наша цель - решить его на отрезке $0 \le t \le O(\sqrt{N})$. Система (16) может быть представлена в виде $\dot{B} = MB$, где его матрица $M = Z - 1 = \tilde{A}_0 + E + H$ (1 обозначает идентичную матрицу) для матриц

$$\tilde{A}_0 = \begin{pmatrix} 0 & 2 & 4 & 0 \\ -\frac{2}{N} & 0 & 2 & 0 \\ 0 & -\frac{2}{N} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, E = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{2}{N} & 0 & 2 & -2 \end{pmatrix}, H = \begin{pmatrix} d_1 & d_1 & d_1 & -2 + d_1 \\ d_2 & d_1 & d_1 & d_1 \\ d_2 & d_2 & d_1 & d_1 \\ d_2 & d_1 & d_1 & d_1 \end{pmatrix},$$

где d_l обозначает различные выражения вида $O(N^{-l}),\ l = 1, 2$.

Мы покажем, что вклад $\tilde{A}_0$ в решение (16),(17) - основной, а вклад E и H - пренебрежимый. В чем тут основная трудность? Рассмотрим резольвенту для задачи Коши (16), (17), это решение $R(t)$ дифференциального уравнения для матриц: $\dot{R} = MR$ с начальным условием $R(0) = 1$. Тогда мы имеем $C(t) = RC(0)$. Матрица R имеет вид $\exp(Mt)$. Но в нашем случае матрицы $\tilde{A}_0, E, H$ не коммутируют, так что мы не можем использовать стандартные свойства для экспоненты. Чтобы справиться с этой работой, надо сначала решить задачу Коши, пренебрегая вкладом E и H в основную матрицу M. Законность такого приближения будет обоснована ниже.

Теперь займемся редуцированным уравнением $\dot{C}(t) = AC(t)$ с начальным условием $C(0) = c_0$. Вычеркивая последний столбец и строку, состоящую из нулей, мы получим новую матрицу A_0. Характеристическое уравнение для A_0 будет $\lambda^3 + \frac{8}{N}\lambda - \frac{16}{N^2} = 0$ и его ненулевые решения с точностью

$O(\frac{1}{N})$ будет $\lambda_{1,2} =^{+}_{-} 2\sqrt{2}i/\sqrt{N}$. Стандартные вычисления дают приближения решения вида

$$\begin{aligned} b &= \tfrac{1}{2} - \tfrac{1}{2}\cos\tfrac{2\sqrt{2}t}{\sqrt{N}}, \\ a &= \tfrac{1}{\sqrt{2N}}\sin\tfrac{2\sqrt{2}t}{\sqrt{N}} + \tfrac{1}{N}\cos\tfrac{2\sqrt{2}t}{\sqrt{N}}, \\ \alpha &= \tfrac{1}{2N}\cos\tfrac{2\sqrt{2}t}{\sqrt{N}} + \tfrac{1}{2N} \end{aligned} \tag{4.15}$$

с точностью $|O(\frac{1}{\sqrt{N}}), O(\frac{1}{N}), O(\frac{1}{N\sqrt{N}})\rangle$. Амплитуда b из (18) имеет пик в точке $t_1 = \frac{\pi\sqrt{N}}{2\sqrt{2}}$ где $b(t_1) = 1$ с точностью $O(\frac{1}{\sqrt{N}})$. Принимая, что вклад E и H в решение мал, мы получаем, что амплитуда искомого состояния $|e_1, e_2\rangle$ есть $1 - O(1/\sqrt{N})$ после $\left[\frac{\pi\sqrt{N}}{2\sqrt{2}}\right]$ шагов параллельного алгоритма, что в $\sqrt{2}$ раз меньше времени последовательного квантового поиска.

Сначала обратимся к ортогональному базису $E_1 = |e_1e_2\rangle$, $E_2 = \frac{1}{\sqrt{N}}|e_1N_2\rangle$, $E_3 = \frac{1}{N}|N_1N_2\rangle$, $E_4 = \frac{1}{\sqrt{N}}|N_1e_2\rangle$. Матрица Z в этом базисе примет вид

$$A_1 = \begin{pmatrix} 1 & \frac{2}{\sqrt{N}} & \frac{4}{N} & -\frac{2}{\sqrt{N}} \\ -\frac{2}{\sqrt{N}} & 1 & \frac{2}{\sqrt{N}} & \frac{4}{N} \\ \frac{4}{N} & -\frac{2}{\sqrt{N}} & 1 & \frac{2}{\sqrt{N}} \\ -\frac{2}{\sqrt{N}} & \frac{4}{N} & \frac{2}{\sqrt{N}} & -1 \end{pmatrix}.$$

Сгруппируем каждую пару унитарных преобразований в алгоритме: $\chi_{2k} \longrightarrow \chi_{2k+1} \longrightarrow \chi_{2k+2}$ и обозначим через B соответствующую матрицу: $B_0 = A_1^2$. Достаточно доказать, что $\|B_0^{[\frac{\pi\sqrt{N}}{4\sqrt{2}}]}|0,0,1,0\rangle - |1,0,0,0\rangle\| = O(\frac{1}{\sqrt{N}})$, так как одно применение A_1 способно увеличить ошибку на $O(\frac{1}{\sqrt{N}})$.

Задача Коши для рекурсии $\bar{c}_{i+1} = B_0\bar{c}_i$ имеет вид $\dot{\bar{c}} = (B_0 - 1)\bar{c}$, $\bar{c} = \bar{c}_0$, и его резольвента имеет вид $R = \exp Bt$, где $B = B_0 - 1$. Мы имеем:

$$B \approx \frac{4}{\sqrt{N}}\begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

с точностью $O(\frac{1}{N})$.

Итак, мы можем рассматривать только проекцию $\bar{c}$ на подпространство $\mathcal{H}_1$ порожденное E_1, E_2, E_3. Обозначим D матрицу

$$\begin{pmatrix} 0 & -\frac{i}{\sqrt{2}} & 0 \\ \frac{i}{\sqrt{2}} & 0 & -\frac{i}{\sqrt{2}} \\ 0 & \frac{i}{\sqrt{2}} & 0 \end{pmatrix}.$$

Тогда ограничение B на $\mathcal{H}_1$ имеет вид $\frac{4\sqrt{2}i}{\sqrt{N}}D$. Легко видеть, что $D^{2k+1} = D$, $D^{2k} = D^2$ для $k = 1, 2, \ldots$. Начальное условие есть $|0,0,1\rangle$ с точностью $O(\frac{1}{\sqrt{N}})$. Положим $k = \frac{4\sqrt{2}}{\sqrt{N}}$. Тогда с точностью $O(\frac{1}{\sqrt{N}})$ мы имеем $|b,a,\alpha\rangle \approx C|0,0,1\rangle$, где

$$\begin{aligned} C &= \exp(kiDt) = \cos(kDt) + i\sin(kDt) \\ &= 1 - \tfrac{(kDt)^2}{2} + \tfrac{(kDt)^4}{4!} - \ldots + i(kDt - \tfrac{(kDt)^3}{3!} + \ldots) \\ &= 1 - D^2(1 - \cos kt) + iD\sin kt \end{aligned}$$

что немедленно дает (18). Теорема доказана.

4.1.8 Параллельный квантовый алгоритм для IS

Теперь займемся проблемой IS для произвольного k. Рассмотрим эволюцию амплитуд, имеющую место когда k оракулов работают параллельно. Пусть $\chi_i = a_0^i|N_1, N_2, \ldots, N_k\rangle + a_1^i|e_1, N_2, \ldots, N_k\rangle + \ldots + a_k^i|e_1, e_2, \ldots, e_k\rangle + R_i$ (это обобщает (10)), где R_i содержит только базисные состояния вида $|\ldots, N_p, \ldots, e_q, \ldots\rangle$. Естественное обобщение оператора (8) будет

$$Z_k = (-1)^k \mathcal{W}^{(k)} \mathcal{R}_0^{(k)} \mathcal{W}^{(k)} \mathcal{F}^{(k)},$$

где $\mathcal{W}^{(k)} = W_1 \bigotimes W_2 \bigotimes \ldots \bigotimes W_k \bigotimes I$, каждый W-H оператор W_i действует на x_i, $i = 1, 2, \ldots, k$, $\mathcal{R}_0^{(k)} = R_{01} \bigotimes \ldots \bigotimes R_{0k} \bigotimes I$, каждая ротация фазы нулей 0 R_{0i} действует на x_i, $i = 1, 2, \ldots, k$, $\mathcal{F}^{(k)} = F_1 \bigotimes \ldots \bigotimes F_k \bigotimes I$, каждый F_i действует на x_i и инвертирует знак e_i, идентичные I действуют на анцилле.

Пусть матрица A определяет эволюцию квантового состояния в параллельном алгоритме так что $\chi_i = A\chi_{i-1}$. A представляет оператор в 2^{kn}-мерном пространстве. Мы редуцируем A до оператора A_r, действующего на $k+1$-мерном пространстве, порожденном векторами $|N_1, N_2, \ldots, N_k\rangle, |e_1, N_2, \ldots, N_k\rangle, \ldots, |e_1, e_2, \ldots, e_k\rangle$. Теперь представим A_r как $A_r = A_0 + B$, где A_0 есть якобиан вида

$$\begin{pmatrix} 0 & 2 & 0 & \ldots & 0 \\ -\frac{2}{N} & 0 & 2 & \ldots & 0 \\ & & & \ddots & \\ \ldots & \ldots & \ldots & & \ldots \\ 0 & \ldots & -\frac{2}{N} & 0 & 2 \\ 0 & \ldots & 0 & -\frac{2}{N} & 0 \end{pmatrix}. \tag{4.16}$$

Эта матрица имеет два ненулевых элемента: 2 - над главной диагональю и $-\frac{2}{N}$ - под ней. Ее размер $(k+1)\times(k+1)$. Примем, что эффект редукции: A к A_r и вклад B - пренебрежимы. Для $k > 2$ этот факт можно доказать, используя итерационные приближения (см. Appendix в работе [Oz3]). Эволюция амплитуд может быть приблизительно представлена в виде решения задачи Коши

$$\dot{\bar{a}} = A\bar{a}, \quad a(0) = |N^{-k/2}, \ldots, N^{-k/2}\rangle, \tag{4.17}$$

где $\bar{a}(i) \approx |a_1^i, \ldots, a_k^i\rangle$, i натуральное, мы примем, что $k \ll \sqrt{N}$.

Особенность параллельного алгоритма для IS можно использовать для организации вычислений с дополнительным классическим параллелизмом. Классический тип параллелизма не использует квантовую запутанность между разными процессорами, но может увеличить производительность за счет дополнительных устройств.

Проверка логических формул

Естественное обобщение G-BBHT представляет квантовая проверка логических формул логики предикатов вида

$$\forall x_1 \exists y_1 \forall x_2 \exists y_2 \ldots \exists y_k \; p(x_1, y_2, \ldots, x_k, y_k) \tag{4.18}$$

(пренексная нормальная форма). Эта задача и будет разбираться в этом параграфе.

Пусть $N = 2^n$ где кортеж $x_1 y_1 \ldots x_k y_k$ принадлежит $\{0,1\}^n$.

Теорема 2 *Существует квантовый алгоритм, проверяющий общезначимость (тождественную истинность) формулы вида (1) за время $O(\sqrt{N})$ с ограниченной вероятностью ошибки$<$ с использованием $(Cn)^k$ одновременных запросов к оракулу, где константа C зависит от вероятности ошибки.*

Заметим, что в работе [BCW] доказано, что формула вида (1) может быть проверена за время порядка $\sqrt{N}(\log N)^{k-1}$ с только одним одновременным запросом к оракулу (Теорема 1.15). Мы видим, что допущение одновременных запросов дает здесь соответствующее уменьшение общего времени, что для задач типа поиска совершенно естественно).

Эта теорема является нетривиальным усилением алгоритма G-BBHT. Дело в том, что имея информацию о P, чтобы преодолеть перемену кванторов: $\forall x \exists y$, мы должны использовать чисто классическую процедуру над G-BBHT: запускать его как подпрограмму в последовательном применении некоторой другой программы в разных случаях. Классическая реализация этой процедуры немедленно даст время $O(\sqrt{|y_k|}|x_1||y_1|\dots|x_k|)$ что значительно больше $\sqrt{N}$.

Как ускорить этот процесс на квантовом компьютере? Это обсуждается в следующем параграфе.

4.2 Квантовые подпрограммы

Чтобы ускорить вычисления с одной подпрограммой, параметры которой могут быть разными, мы должны запускать эту подпрограмму одновременно для разных значений этих параметров. Но реализация этого плана встречается с очевидной трудностью. Квантовый алгоритм должен преобразить суперпозицию $\sum_i \lambda_i e_i$ базисных состояний e_i. Если подпрограмма содержит промежуточные измерения, что требует чисто классических процедур, мы должны запустить эту подпрограмму на каждом e_i отдельно, что даст рост времени как в классическом случае. Значит, нам надо, по крайней мере, исключить измерения из подпрограмм. Но этого недостаточно. Чтобы использовать квантовый параллелизм для ускорения вычислений, мы должны также обеспечить специальную единую форму выходных данных для нашей подпрограммы.

Обозначение *Для вещественных положительных чисел ϵ ξ_ϵ обозначает такое состояние χ в гильбертовом пространстве, что $\|\chi - \xi\| < \epsilon$.*

Определение *Унитарный алгоритм, вычисляющий функцию $f: \{0,1\}^n \longrightarrow \{0,1\}$ с вероятностью ошибки p_{err} есть такой квантовый алгоритм, действие которого на входном состоянии $x \in \{0,1\}^n$ есть последовательность унитарных операторов вида*

$$\xi_0 \longrightarrow \xi_1 \longrightarrow \dots \longrightarrow \xi_T,$$

где $\xi_0 = |x, 0\rangle$, $\xi_T = (\tilde{\xi} \bigotimes |f(x)\rangle)_\epsilon$, где $\epsilon < p_{err}/2$, ϵ зависит от x.

Итак, измерение конечного состояния ξ_T дает $f(x)$ с вероятностью большей, чем $1 - p_{err}$. Унитарный алгоритм можно использовать как подпрограмму, так как состояние $\sum_i \lambda_i |x_i, 0\rangle$ преобразуется в $\sum_i \lambda_i \tilde{\xi_i} \bigotimes f(x_i)\rangle + \bar{\epsilon}$, где $\bar{\epsilon} = \sum_i \lambda_i \bar{\epsilon}_{x_i}$ и $\|\bar{e}\| < p\sqrt{N}/2$, где N есть число всех базисных состояний. Примем, что момент времени T для конца унитарного алгоритма вычислен заранее классическим способом. (см. работу [De]).

4.3 Унитарный квантовый поиск

Нашей ближайшей задачей будет построение унитарного алгоритма в временем $O(\sqrt{N_1})$, для стандартной задачи нахождения такого x, что $p(x)$ истинно, для данного предикат p, $N_1 = 2^{|x|}$.

Предложение 1 *Существует унитарный алгоритм, осуществляющи переход от $|0,0,\dots,0\rangle$ к $|0,0,\dots,0,\gamma\rangle_\epsilon$ с оракулом p где $\gamma = 1$ если $\exists x\ p(x)$ и 0 в иных случаях. Этот алгоритм требует*

$\sqrt{N_1}$ шагов с Mn обращениями к p одновременно и использует $M(n+2)+2$ кубит памяти, где $M = \log(1/\epsilon)$.

Доказательство

Напомним алгоритм G-BBHT (см. [BBHT]). Он состоит из следующих шагов.

1. Выбор номера m.

2. Выбор значения натуральной переменной χ, распределенной равномерно на $\{1, 2, \ldots, m\}$.

3. Выполнение гроверовского преобразования WF_0WF_p χ раз на начальном состоянии $\sum_j e_j/\sqrt{N}$.

4. Измерение результата.

Для нашей цели достаточно взять $m = \sqrt{N_1}$. С этим значением требуемое значение x получается при измерении конечного состояния с вероятностью примерно $1/2$ (см. выше). Нам будет нужна такая техническая Лемма.

Лемма 1 *Для любого $m = 1, 2, \ldots$ существует унитарное преобразование, выполняющее переход:*

$$|0 \ldots 0\rangle \longrightarrow \ldots \longrightarrow \frac{1}{\sqrt{m}} \sum_{\chi \leq m} |\chi\rangle,$$

где χ есть натуральное число в его бинарной записи.

Доказательство этой несложной Леммы предоставляется читателю (его также можно найти в работе [Ki]).

В силу Леммы 1 мы можем выполнить пункты 1-3 алгоритма G-BBHT с помощью унитарного алгоритма.

Теперь заведем M независимых блоков по $2n$ кубит каждый и выполним унитарную версию алгоритма G-BBHT на каждом из них независимо. Мы получим состояние $|0 \ldots 0\rangle \bigotimes |x_1 x_2 \ldots x_M\rangle$ где результат работы каждого блока записан в

$$x_i = \sum_j \lambda_j e_j, \qquad \sum_{p(e_j)\ \text{true}} |\lambda_j|^2 \approx 1/2, \tag{4.19}$$

если такое e_j существует. Применяя оракул для p, мы находим состояние

$$|0 \ldots 0\rangle \bigotimes (\sum_j (\lambda_j |e_j\rangle \bigotimes |p(e_j)\rangle)) \bigotimes \ldots \bigotimes (\sum_j (\lambda_j |e_j\rangle \bigotimes |p(e_j)\rangle)), \tag{4.20}$$

где $p(e_j) \in \{0, 1\}$ есть значения вспомогательных кубит. Здесь мы принимаем, что оракул p отображает $|a, b\rangle$ в $|a, b + p(a) \pmod 2\rangle$.

Лемма 2 *Лемма Существует унитарный алгоритм с линейной временной сложностью, осуществляющий переход $|\sigma_1\sigma_2 \ldots \sigma_M 0^{M+2}\rangle \longrightarrow |\sigma_1\sigma_2 \ldots \sigma_M 0^{M+1}\sigma\rangle$, где $\forall i = 1, 2, \ldots, M$ $\sigma, \sigma_i \in \{0, 1\}$ and $\sigma = 1$ iff $\exists i \in \{1, \ldots, M\} :\ \sigma_i = 1$.*

Доказательство Леммы 2.

Рассмотрим классическое обратимое преобразование f с тремя кубитами:
| result, controller, subject $\rangle$, такое что

$$\begin{aligned} |\,0\,0\,0\,\rangle &\xrightarrow{f} |\,0\,0\,0\,\rangle \\ |\,0\,0\,1\,\rangle &\xrightarrow{f} |\,1\,0\,1\,\rangle \\ |\,1\,0\,1\,\rangle &\xrightarrow{f} |\,1\,1\,1\,\rangle \\ |\,1\,0\,0\,\rangle &\xrightarrow{f} |\,1\,0\,0\,\rangle \end{aligned}$$

Такое преобразование может быть выбрано унитарным, так как наши требования к нему допускают взаимную однозначность. Если мы применим его последовательно, так что на шаге номер i кубит "result"всегда является последним вспомогательным кубитом, кубит "controller всегда i -й вспомогательный кубит, а "subject всегда идет под номером σ_i, и $i = 1, 2, \ldots, M$, мы получим состояние $|\delta_1\delta_2\ldots\delta_M\delta_{M+1}\ldots\delta_{2M}\sigma 0\rangle$. Теперь делаем $|\delta_1\delta_2\ldots\delta_{2M}\sigma\sigma\rangle$ через $|\sigma 0\rangle \longrightarrow |\sigma\sigma\rangle$, и наконец выполняем все обратные последовательные преобразования в обратном порядке, что дает $|\sigma_1\sigma_2\ldots\sigma_M 0^{M+1}\sigma\rangle$. Лемма 2 доказана.

Назовем унитарный алгоритм из Лемы 2 EXISTS. Теперь применим Лемму 2 к состоянию (3) со вспомогательными кубитами, игрпающими роль $\sigma_1, \sigma_2, \ldots, \sigma_M$. По (2) это даст состояние

$$(|0\ldots|0\rangle \bigotimes (\sum_j \lambda_j |e_j\rangle \bigotimes |p(e_j)\rangle)) \bigotimes \ldots \bigotimes (\sum_j (\lambda_j |e_j\rangle \bigotimes |p(e_j)\rangle)) \bigotimes \gamma)_\epsilon. \tag{4.21}$$

Мы имеем M независимых блоков по n кубит каждый и выполняем наш унитарный алгоритм на всех этих блоках независимо, поэтому мы найдем требуемое значение x по крайней мере в одном из блоков с вероятностью порядка $1 - \frac{1}{2^M}$. Значит, если имеется значение допустимой вероятности ошибки ϵ, $M = \log\frac{1}{\epsilon}$ будет достаточно. Наконец, применим ко всем кубитам кроме γ в (4) преобразования, обратные к G-BBHT, и найдем $(|0\ldots 0\gamma\rangle)_\epsilon$.

Предложение 1 доказано.

Обозначение *Обозначим унитарный алгоритм из Предложения 1 с оракулом p через* $\mathrm{SEARCH}(p)$.

Заметим, что мы можем выполнить G-BBHT как унитарный алгоритм, если p нам дано как подпрограмма.

4.4 Формулы логики предикатов

Рассмотрим формулу логики предикатов вида (1). Простым обобщением ее является формула со свободными переменными $z_1, z_2, \ldots, z_q$ вида

$$\forall x_1 \exists x_2 \forall x_3 \ldots Q_{k-1} x_{k-1} Q_k x_k p(z_1, z_2, \ldots, z_q, x_1, \ldots, x_k). \tag{4.22}$$

где $Q_1, Q_2 \in \{\exists, \forall\}$.

Предложение 2 *Предложение Существует унитарный алгоритм, выполняющий переход*

$$|\,z_1\ldots z_q 0\ldots 0\rangle \longrightarrow \ldots \longrightarrow |\,z_1\ldots z_q 0\ldots 0\gamma\rangle$$

за $2^{\frac{1}{2}\sum_{i=1}^{k}|x_k|}$ *шагов, использующий* $(Mn)^k$ *одновременных запросов к оракулу, где для всякого значения свободных переменных* $z_1, \dots z_q$

$$\gamma = \begin{cases} 0, \ \textit{if} \ (5) \ \textit{true}, \\ 1, \ \textit{if} \ (5) \ \textit{false}. \end{cases}$$

Доказательство.

Индукция по k. Базис. $k = 0$. Доказывать нечего. Шаг. Предположим, что утверждение верно длязначения k, меньшего заданного, докажем его для данного k. Индуктивная гипотеза говорит о том, что существует унитарный алгоритм с временной сложностью $2^{\frac{1}{2}\sum_{i=1}^{k-1}|x_i|}$, использующий не более чем $(Mn)^{k-1}$ одновременных запросов к оракулу p и $(Mn)^{k-1}$ кубит, корорый вычисляет функцию
$z_1, \dots z_q, x_k \longrightarrow T_1 \in \{0,1\}$, где $T_1 = 1$ тогда и только тогда, когда $\forall x_1 \exists x_2 \dots Q_{k-1} x_{k-1} p(z_1, \dots, z_q, x_1, \dots, x_{k-1}, x_k)$.
Обозначим этот алгоритм через P_{k-1}. Наша цель - построить унитарный алгоритм для функции $z_1, \dots, z_q \longrightarrow T \in \{0,1\}$, где $T = 1$ тогда и только тогда, когда
$\forall x_1 \exists x_2 \dots Q_{k-1} x_{k-1} Q_k x_k p(z_1, \dots, z_q, x_1, \dots, x_k)$.

Рассмотрим два разных случая.

Случай 1: Q_k есть $\exists$. ‘ Тогда требуемый алгоритм - это SEARCH (P_{k-1}). По Предложению 1 этот алгоритм требует $2^{\frac{1}{2}|x_k|}$ шагов с Mn одновременными запросами к P_{k-1}, каждый из которых по индуктивному предположению содержит $2^{\frac{1}{2}\sum_{i=1}^{k-1}|x_i|}$ шагов с $(Mn)^{k-1}$ одновременными запросами к p. Таким образом< мы имеем требуемый унитарный алгоритм с $2^{\frac{1}{2}\sum_{i=1}^{k}|x_1|}$ шагами и $(Mn)^k$ одновременными запросами к p. Число требуемых для этого кубит будет порядка $(Mn)^k$.

Случай 2: Q_k есть $\forall$.

Здесь требуемым алгоритмом будет NOT (SEARCH (NOT (P_{k-1}))), где NOT является отрицанием: $0 \longrightarrow 1$, $1 \longrightarrow 0$. Проверка доказываемого утверждения с использованием индуктивной гипотезы делается так же, как и в предыдущем случае. Предложение 2 доказано.

Теорема является частным случаем Предложения 2. Теорема доказана.

4.5 Обобщения алгоритма Гровера

Рассмотрим более общую формулировку задачи квантового поиска, при которой у нас будут m типов состояний: множество N_1 основных состояний, которые не будут вообще подвергаться изменениям при инверсии, множество N_2 целевых состояний, которые вращаются на π, и являются решениями уравнения $f(x) = 1$, и $m-2$ всех прочих типов состояний числом соответственно: $N_3, N_4, \dots, N_m$ которые будут поворачиваться на на $m-2$ различных углов: $d_1, d_2, \dots, d_{m-2}$ соответственно. Занумеруем элементы из множества N_s так что $N_s = \{x_s^j;\ j = 1, 2, \dots, l_s\}$, $s = 1, 2, \dots, m$. Тогда целевые состояния будут $x_2^1, x_2^2, \dots, x_2^{l_2}$, и число всех базисных векторов $\sum_{s=1}^{m} l_s = N$ есть размерность основного Гильбертова пространства H. Обозначим этот ортонормированный базис H через $\bar{N}$.

Наше предположение о углах поворотов можно переформулировать как ограничение, наложенное на унитарное преобразование U которое будет использовано в определении GSA вместо I_{tar}. Пусть U будет унитарным преобразованием на H с собственными векторами x_j^s $s = 1, 2, \ldots, m,\ j = 1, 2, \ldots, l_s$, и собственными значениями всех x_j^2, равными -1, а для всех x_j^1 - равными 1, а для всех x_j^s $(s \geq 3)$ равными $e^{i(d_k - \pi)},\ d_k \in [0, 2\pi)$. Пусть $v_k = e^{id_k}$.

Определим k как такой индекс, при котором d_k максимален. Пусть $d = d_k$. Определим $\gamma = \sqrt{\frac{l_2}{N}}$. Положим $\gamma \longrightarrow 0$ $(n \longrightarrow \infty)$ поскольку иначе проблема может быть тривиально решена на классическом компьютере ти применение квантового компьютера не имеет смысла. Теперь определим $e_j = \frac{1}{\sqrt{l_j}} \sum\limits_{x \in N_j} |x\rangle$. Таким образом, например e_2 будет суперпозицией всех целевых состояний с равными амплитудами. Определим $\tilde{0}$ как $\frac{1}{\sqrt{N}} \sum\limits_{x \in \bar{N}} |x\rangle$. Это состояние будет начальным для нашего алгоритма. Обозначим через H_0 подпространство, порожденное всеми векторами e_j. Эволюция вектора состояния будет вращением в этом m мерном подпространстве. Из нашего определения этого подпространства непосредственно вытекает, что начальный вектор $\tilde{0}$ принадлежит этому подпространству. С другой стороны действие I_{tar}, ограниченное на H_0 будет совпадать с ограничением преобразования I_{e_2} на H_0.

Наш основной результат будет таким.

Теорема 3 *Пусть* $l_1/N \longrightarrow 1$, $(N - l_1 - l_2)/d\sqrt{Nl_2} \longrightarrow 0$, $\sqrt{\frac{l_2}{N}} = o(d)$ *и* $t = \left[\frac{\pi}{4}\sqrt{\frac{l_2}{N}}\right]$, *тогда итерированные преобразования* $I_{\tilde{0}}U$, *примененные к* $\tilde{0}$ t *раз, с последующим наблюдением, даст нам состояние* e_2 *с исчезающей вероятностью ошибки (* $n \longrightarrow \infty$ *).*

Мы будем работать с подпространством H_0 основного пространства. $I_{\tilde{0}}$ и U сохраняет это подпространство. Идея доказательства следующая. Мы подсчитаем собственные векторы и собственные значения матрицы $G = -I_{\tilde{0}}U$ и сравним их с соответствующими собственными векторами и значениями матрицы стандартного GSA - оператора $I_{\tilde{0}}I_{tar}$. Окажется, что двумерное подпространство of H_0 порожденное состояниями e_2 и $\tilde{0}$, а также собственные векторы и собственные значения G будут совпадать с аналогичными величинами для GSA с высокой точностью. Это означает, что поведение итераций G будет таким же, что и GSA. Иными словами, эволюция вектора состояния, порождаемая нашим алгоритмом, может быть с высокой точностью описано эволюцией в двумерном подпространстве, порожденном векторами e_1 и e_2.

4.5.1 Вычисление матриц

Приближенное вычисление

Оставшаяся часть этого параграфа посвящена, в основном, доказательству Теоремы 1. Мы будем вычислять все матрицы в базисе $e_1, e_2, \ldots, e_m$ подпространства H_0. Определим $\langle\tilde{0}|e_2\rangle = \sqrt{\frac{l_2}{N}} = X$, $\langle\tilde{0}|e_j\rangle = \sqrt{\frac{l_j}{N}} = Y_{j-2}$, $j = 3, 4, \ldots, n$, $x_j = 2X_j$. В этом параграфе мы будем выполнять все вычисления с точностью до $\sum\limits_{j=2}^{m} l_j/N$. Обоснование законности такого подхода будет дано в следующем параграфе.

Положим $\sum\limits_{j\geq 2} l_j/N = \epsilon$. Мы имеем: $\tilde{0} = (\sqrt{1-\epsilon}, X, Y_1, \ldots, Y_{m-2})^T$, $\sqrt{\frac{l_1}{N}} = \sqrt{1-\epsilon} = 1 - \frac{\epsilon}{2} + o(\epsilon)$. Тогда $\tilde{0} \approx \tilde{0}_{app} = (1, X, Y_1, \ldots, Y_{m-2})^T$ с точностью до $O(\epsilon)$, где расстояние между векторами оценивается в Гильбертовом пространстве H_0.

Непосредственно из определений следует, что H_0 является подпространством, инвариантным относительно операторов U и $I_{\tilde{0}}$. В m мерном подпространстве H_0 матрица оператора U будет иметь такой вид.

$$U = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 0 & -1 & 0 & \ldots & 0 \\ 0 & 0 & -v_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & 0 & -v_{m-2} \end{pmatrix}.$$

Чтобы найти матрицу для $I_{\tilde{0}}$ в этом подпространстве, мы представим этот оператор в виде $I_{\tilde{0}} \approx I_{\tilde{0}_{app}} = V^{-1} I_{e_1} V$, где матрица V такова, что $V\tilde{0}_{app} = e_1$. Это происходит потому, что $V^{-1} I_{e_1} V \tilde{0}_{app} = -V^{-1} e_1 = -\tilde{0}_{app}$. Непосредственно проверяется, что матрица V и ее обратная имеют вид

$$V = \begin{pmatrix} 1 & X & Y_1 & Y_2 & \ldots & Y_{m-2} \\ -X & 1 & 0 & 0 & \ldots & 0 \\ -Y_1 & 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ -Y_{m-2} & 0 & 0 & 0 & \ldots & 1 \end{pmatrix},$$

$$V^{-1} = \begin{pmatrix} 1 & -X & -Y_1 & -Y_2 & \ldots & -Y_{m-2} \\ X & 1 & 0 & 0 & \ldots & 0 \\ Y_1 & 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ Y_{m-2} & 0 & 0 & 0 & \ldots & 1 \end{pmatrix}.$$

The substantiation of this assumpsion is done in the next section. Мы далее имеем:

$$I_{e_1} = \begin{pmatrix} -1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \end{pmatrix}$$

И теперь непосредственное умножение матриц дает

$$I_{\tilde{0}_{app}} = V^{-1} I_{e_1} V = \begin{pmatrix} -1 & -x & -y_1 & -y_2 & \ldots & -y_{m-2} \\ -x & 1 & 0 & 0 & \ldots & 0 \\ -y_1 & 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ -y_{m-2} & 0 & 0 & 0 & \ldots & 1 \end{pmatrix},$$

$$G = -I_{\tilde{0}_{app}} U = \begin{pmatrix} 1 & -x & -y_1 v_1 & -y_2 v_2 & \ldots & -y_{m-2} v_{m-2} \\ x & 1 & 0 & 0 & \ldots & 0 \\ y_1 & 0 & v_1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ y_{m-2} & 0 & 0 & 0 & \ldots & v_{m-2} \end{pmatrix}.$$

Точность приближения

Теперь мы покажем, что для целей приближения GSA наша точность до ϵ достаточна. Мы будем работать с обычной матричной нормой заданной равенством

$$\|A\| = \max_{\|\bar{x}\|=1} \|A\bar{x}\|.$$

Точность нашего приближения $G = -I_{\tilde{0}_{app}} U$ к точной матрице $-I_{\tilde{0}} U$ которая будет обозначаться через G_{exact} есть $\|G - G_{exact}\| = \|U\|\|I_{\tilde{0}} - I_{\tilde{0}_{app}}\| = \|I_{\tilde{0}} - I_{\tilde{0}_{app}}\| = \|\tilde{0} - \tilde{0}_{app}\| = O(\epsilon)$. Следовательно, G есть приближение of G_{exact} с точностью до ϵ, и мы можем представить ее как $G = G_{exact} + \Delta$, где $\|\Delta\| = O(\epsilon)$. Пусть $t = \sqrt{\frac{N}{l_2}}$ есть порядок числа итераций в GSA. Оценим число $\nu = t\Delta = \frac{\sum\limits_{j\geq 2} l_j}{\sqrt{N l_2}}$. Если $l_2 = O(\sum\limits_{j\geq 3})$, мы можем опустить l_2 и используя условие Теоремы, получить $\nu = o(1)$. В противном случае $\sum\limits_{j\geq 3} = o(l_2)$ и ν будет порядка $\sqrt{\frac{l_2}{N}}$ что снова $o(1)$. Значит, во всех случаях $\nu = o(1)$. Тогда мы имеем: $G^t = (G_{exact} + \Delta)^t = G^t_{exact} + O(t\Delta G^{t-1}_{exact}) = G^t_{exact} + o(1)$. Это значит, что для исследовния поведения G^t_{exact} будет достаточно использовать приближенгие G к G_{exact}, что и обосновывает наши вычисления.

4.5.2 Нахождение собственных значений

Чтобы найти собственные значения G, нам следует сначала посчитать характеристический многочлен $p_{m-2}(\lambda) = |G - \lambda I|$, где I - идентичная матрица. Затем, решая реккурентно уравнение $p_{m-2}(\lambda) = 0$, мы найдем собственные значения. Мы имеем

$$p_{m-2}(\lambda) = \begin{vmatrix} 1-\lambda & -x & -y_1 v_1 & -y_2 v_2 & \dots & -y_{m-2} v_{m-2} \\ x & 1-\lambda & 0 & 0 & \dots & 0 \\ y_1 & 0 & v_1 - \lambda & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ y_{m-2} & 0 & 0 & 0 & \dots & v_{m-2} - \lambda \end{vmatrix} =$$

$$(-1)^{m+1} y_{m-2} v_{m-2} \begin{vmatrix} x & 1-\lambda & 0 & \dots & 0 \\ y_1 & 0 & v_1 - \lambda & \dots & 0 \\ y_2 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ y_{m-2} & 0 & 0 & \dots & 0 \end{vmatrix} + (v_{m-2} - \lambda) p_{m-3}(\lambda) =$$

$$y^2_{m-2} v_{m-2} (1-\lambda)(v_1 - \lambda) \dots (v_{m-3} - \lambda) + (v_{m-2} - \lambda) p_{m-3}(\lambda).$$

что дает следующее реккурентное соотношение

$$p_{m-2}(\lambda) = (v_{m-2} - \lambda) p_{m-3}(\lambda) + y^2_{m-2} v_{m-2} (1-\lambda)(v_1 - \lambda) \dots (v_{m-3} - \lambda). \tag{4.23}$$

Используя базис рекурсии:
$-p_1(\lambda) = (\lambda - 1 + ix)(\lambda - 1 - ix)(v_1 - \lambda) + v_1 y_1^2 (1-\lambda)$ мы можем вывести из основного уравнения (1) с

помощью непосредственных преобразований общую формулу для характеристического многочлена:

$$\begin{array}{rl} p_{m-2}(\lambda) = & (\lambda-1+ix)(\lambda-1-ix)(v_1-\lambda)(v_2-\lambda)\dots(v_{m-2}-\lambda)+ \\ & v_1y_1^2(1-\lambda)(v_2-\lambda)\dots(v_{m-2}-\lambda)+ \\ & v_2y_2^2(1-\lambda)(v_1-\lambda)(v_3-\lambda)\dots(v_{m-2}-\lambda)+ \\ & \dots+v_{m-2}y_{m-2}^2(1-\lambda)(v_1-\lambda)\dots(v_{m-3}-\lambda). \end{array} \tag{4.24}$$

Обозначим первое слагаемое в (2) через $p_0(\lambda)=(\lambda-1+ix)(\lambda-1-ix)(v_1-\lambda)(v_2-\lambda)\dots(v_{m-2}-\lambda)$, так что $p_{m-2}(\lambda)=p^0+\delta$, где

$$\begin{array}{l} \delta = v_1y_1^2(1-\lambda)(v_2-\lambda)\dots(v_{m-2}-\lambda)+ \\ v_2y_2^2(1-\lambda)(v_1-\lambda)(v_3-\lambda)\dots(v_{m-2}-\lambda)+ \\ \dots+v_{m-2}y_{m-2}^2(1-\lambda)(v_1-\lambda)\dots(v_{m-3}-\lambda). \end{array}$$

Это означает, что p_{m-2} и $p_0(\lambda)$ отличается только сдвигом на δ. Корни $p_0(\lambda)$ будут $\lambda_1=1-ix,\ \lambda_2=1+ix,\ \lambda_3=v_1,\dots,\lambda_m=v_{m-2}$. Обозначим корни p_{m-2} через $\tilde\lambda_1,\tilde\lambda_2,\dots,\tilde\lambda_m$.

Теперь нам нужно оценить различие между λ_j и $\tilde\lambda_j$, используя оценку δ в окрестности двух корней: $\lambda_{1,2}$ которые играют главную роль в динамике рассматриваемого алгоритма.

Заметим, что $|\lambda_1-\lambda_2|=o(|v_j-\lambda_1|)$. С другой стороны $|\lambda_1-\lambda_2|\gg\delta^2$. Наш полином p можно приблизить квадратичным полиномом q со старшим коэффициентом $A=\lambda-v=\Omega(d)$ в окрестности λ_1 или λ_2 радиуса $|\lambda_1-\lambda_2|$. Производная q этого квадратичного полинома в этой окрестности есть $q=\gamma(v_1-\lambda)\cdots(v_{n-2}-\lambda)$. Обозначим разницу между корнями через $\sigma=|\lambda_1-\tilde\lambda_1|+|\lambda_2-\tilde\lambda_2|$. Тогда $\sigma=O(\delta/q')=\sum\limits_{j=1}^{m-2}\frac{v_jy_j^2(1-\lambda)}{\gamma(v_j-\lambda)}$. Используя равенства $v_j-\lambda=O(d_j)$, $1-\lambda=O(\gamma)$, мы заключаем, что $\sigma=O(\frac{1}{d}\sum\limits_{j\geq3}\frac{l_j}{N})$ что есть $o(\gamma)$, поскольку по условию Теоремы $\sum\limits_{j\geq3}l_j=o(d\sqrt{Nl_2})$. Следовательно, $\tilde\lambda_{1,2}=1+-ix+o(\gamma)$, $\tilde\lambda_3=v+o(\gamma)$.

4.5.3 Нахождение собственных векторов

Сначала займемся собственными векторами для двух первых корней: $\tilde\lambda_{1,2}=1+-ix+o(\gamma)$.

1). $\lambda=1-ix+o(\gamma)$. Пусть собственный вектор есть столбец вида $\bar a=(a,b,w_1,\dots,w_{m-2})^T$. Система линейных уравнений, определяющая $\bar a$ имеет такую форму: $(G-\lambda E)\bar a=\bar 0$ и может быть записана в виде

$$\begin{pmatrix} ix & -x & -y_1v_1 & \dots & -y_{m-2}v_{m-2} \\ x & ix & 0 & \dots & 0 \\ y_1 & 0 & v_1-1+ix & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ y_{m-2} & 0 & 0 & \dots & v_{m-2}-1+ix \end{pmatrix}\begin{pmatrix} a \\ b \\ w_1 \\ \dots \\ w_{m-2} \end{pmatrix}=\begin{pmatrix} o(\gamma) \\ o(\gamma) \\ o(\gamma) \\ \dots \\ o(\gamma) \end{pmatrix}.$$

Переписывая это в виде системы линейных уравнений, находим

$$\begin{cases} ixa & -xb & -y_1v_1w_1 & -\dots & -y_{m-2}v_{m-2}w_{m-2} & =o(\gamma) \\ xa & +ixb & & & & =o(\gamma) \\ y_1a & & +(v_1-1+ix)w_1 & & & =o(\gamma) \\ \dots & \dots & \dots & \dots & \dots & \dots \\ y_{m-2}a & & & & +(v_{m-2}-1+ix)w_{m-2} & =o(\gamma) \end{cases}$$

Предположим, что собственный вектор имеет ограниченную норму, что означает, что все его компоненты ограничены. Решая систему, находим $w_j = \frac{y_j a}{v_j - 1 + ix}$. Применяем условие Теоремы: $\sqrt{\frac{l_2}{N}} = o(d)$ и заключаем, что $w_j = o(1)$, и с точностью до $o(1)$ $a = i, b = -1$.

2). $\lambda = 1 + ix + o(\gamma)$. Соостетствующее вычислениедает $a = i, b = 1$ с точностью до $o(1)$, all $w_j = o(1)$.

3). Для всех других собственных значений мы имеем $a = 0,\ b = 0$ с точностью до $o(\gamma)$, поскольку соответствующие вектора должны быть ортогональны подпространству, порожденному e_1, e_2.

Итак, мы имеем следующую ситуацию:

если $n \longrightarrow \infty$, то собственные векторы, принадлежащие подпространству, порожденному e_1, e_2 отличаются на $o(\gamma)$ от соответствующих собственных значений стандартного GSA, и соответствующие собственные векторы отличаются на $o(1)$ от соответствующих собственных векторов стандартного GSA. Это значит, что если число t итераций будет порядка $1/\gamma$ - как и в GSA, то разница между результирующими состояниями GSA и итерированного применения G будет порядка $o(1)$. Теорема доказана.

4.6 Решение задач дискретной оптимизации

Мы завершим рассмотрение модификаций переборных задач поиском точки экстремума целочисленной функции. Пусть функция $f : \{0,1\}^n \longrightarrow \{0,1\}^n$ задана своим оракулом (или схемой из функциональных элементов). Ее мы будем, как всегда, трактовать как целочисленную функцию. Задача состоит в том, чтобы найти ее точку максимума (или минимума). Заметим, что в случае такой очень общей формулировки задачи дискретной оптимизации мы не можем применить какой либо прием, существенно упрощающий перебор, например, симплекс метод или дифференцирование. Таким образом, классическое решение этой задачи требует порядка $N = 2^n$ действий.

Рассмотрим идею квантового метода ее решения, основанного на применении GSA. Будем находить точку максимума с помощью последовательных приближений. А именно, расположим все аргументы в порядке возрастания значения функции f на них: $f(x_0) \leq f(x_1) \leq \ldots \leq f(x_{N-1}$. На каждом j-м шаге входным значением будет некоторое x_{j_k}. Применим G-BBHT с оракулом, принимающим значение 1 в точности на тех аргументах x_j, для которых $f(x_{j_k}) < f(x_j)$, т.е. на $x_{j'},\ j' > j_k$. После очередных наблюдений и проверок правильности мы получим следующее значение $x_{j_{k+1}}$ и т.д. до тех пор, пока не доберемся до x_{N-1}. Детальный анализ (см. например, []) показывает, что сложность такого алгоритма имеет порядок $\sqrt{N}$, то есть дает такое же ускорение, как и GSA.

Глава 5

Квантовое преобразование Фурье и его применения

5.1 Что общего между цветовым зрением и факторизацией целых чисел?

Каждый читатель наверняка знает, что разница в цвете предметов связана с различной длиной волн (или с различной частотой) отраженного ими света. Мы не будем здесь заниматься описанием очень сложного (и до конца не понятного) процесса различения частот фотонов в зрительном анализаторе человека, а вместо этого дадим очень схематичную формулировку этой задачи и ее решение с использованием преобразования Фурье.

Представим себе, что у нас есть способ генерации какого-либо гармонического колебания вида $e^{2i\pi\ wx}$, причем его частота ω нам не известна. Задача состоит в том, чтобы определить эту частоту. В этой постановке, конечно, можно считать что $\omega \in [0,1)$. Эта задача является очень общей и мы будем в каждом случае ее конкретизировать так, что частными случаями окажутся: поиск собственных чисел операторов, распознавание структур, и даже факторизация целых чисел! В самой общей формулировке она может быть решена одним мощным приемом, изобретенным в начале XIX века Огюстом Фурье - специальным интегральным преобразованием, действующим на функции. Это преобразование и обратное к нему имеет вид:

$$\begin{aligned} F\ f = \phi(\lambda) = &\quad \tfrac{1}{\sqrt{2\pi}} \int\limits_D e^{-i\lambda x} f(x)\ dx, \\ F^{-1}\ \phi = f(x) = &\quad \tfrac{1}{\sqrt{2\pi}} \int\limits_D e^{i\lambda x} \phi(\lambda)\ d\lambda. \end{aligned} \tag{5.1}$$

Пусть на вход этого преобразования подается наше гармоническое колебание $e^{2i\pi\ wx}$. Здесь имеется маленькое затруднение, связанное с тем, что преобразование определено на всей числовой оси $D = R$, и это требует, если рассуждать строго, привлечения обобщенных функций. Чтобы этого избежать, представим себе что D это просто большой интервал прямой, например $-B, B$, где B очень большое число. Тогда мы получим:

$$F\ f = \phi(\lambda) = \frac{1}{\sqrt{2\pi}} \int\limits_D e^{i(2\pi\omega - \lambda)x} f(x)\ dx.$$

Теперь мы можем увидеть, что если аргумент результирующей функции λ близок к $2\pi\omega$, то интеграл очень велик, потому что подынтегральная функция близка к единице, а в противном случае этот интеграл небольшой, так как разные части интервала будут вычитаться, или, как говорят физики, деструктивно интерферировать. Так что результирующая функция будет иметь большой пик в точке $\lambda = 2\pi\omega$. Теперь представим, что мы каким-то образом смогли выполнить преобразование Фурье на квантовом компьютере. Тогда в выходном состоянии вся масса амплитуды будет сосредоточена вокруг числа $2\pi\omega$ и измерение выходного состояния с высокой точностью даст нам значение неизвестного параметра ω. В этом и состоит вторая идея применения квантового компьютера. Нужно лишь додуматься, как реализовать на нем преобразование Фурье.

5.1.1 Квантовое преобразование Фурье и его основное свойство

Попытаемся сначала записать квантовый вариант преобразования Фурье. В силу линейности его действие достаточно определить на базисных элементах. Именно базисные вектора должны играть роль функций f и ϕ в определении. Рассмотрим следующее определение квантового преобразования Фурье.

$$\mathrm{QFT}:\ |a\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{b=0}^{N-1} e^{-\frac{2\pi i\ ab}{N}} |b\rangle, \qquad \mathrm{QFT}^{-1}:\ |a\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{b=0}^{N-1} e^{\frac{2\pi i\ ab}{N}} |b\rangle. \tag{5.2}$$

Это определение построено по аналогии с обычным. Непосредственно проверяется (мы предлагаем читателю это проделать) что это преобразование отображает базисные векторы во взаимно ортогональные векторы единичной длины, из чего следует его унитарность. Далее, также непосредственно можно убедиться, что эти формулы действительно задают взаимно обратные преобразования.

Теперь мы можем продемонстрировать, каким образом это преобразование может выявлять скрытые периоды. Сначала нам понадобится ввести одно новое понятие, играющее важную роль в квантовом компьютинге. Это - условное применение оператора. Идея состоит в следующем. Предположим, что в нашем распоряжении находится некоторый унитарный оператор U. Мы можем для наглядности считать, что он задан нам в виде соответствующей схемы из функциональных элементов, хотя, вообще говоря, это и необязательно. Далее, пусть у нас имеется некий вспомогательный квантовый регистр, состоящий из нескольких кубит, который мы объявим управляющим. Цель состоит в том, чтобы применить оператор U последовательно столько раз, какое число содержится в управляющем регистре. Запишем это формально в виде такого определения:

$$U_{cond}|x,\alpha\rangle \longrightarrow \begin{cases} |U\ x,\alpha\rangle, & \text{if } \alpha = 1, \\ |x,\alpha\rangle & \text{if } \alpha = 0. \end{cases}$$

Мы не будем обсуждать здесь вопросы реализации данного преобразования. Заметим только, что если оператор U задан нам в виде схемы из квантовых функциональных элементов, можно легко построить схему такого же типа, реализующую U_{cond}. Для этого достаточно сделать условным каждый оператор, входящий в данную схему. Предоставляем подробности читателю.

Рассмотрим теперь важную задачу определения собственной частоты оператора U. Эта частота будет появляться при измерении некоторого специального регистра из n кубит, обозначаемого через α, в котором будут храниться последовательные бинарные знаки этой частоты, разумеется, с ограниченной точностью. Мы будем предполагать, что истинная частота может быть записана в этом регистре с абсолютной точностью, это не очень существенно при описании общей схемы рассуждений. Тех читателей, которые заинтересуются общим случаем, мы отсылаем к статье []. Таким

образом, наш компьютер будет работать с двумя регистрами: регистр для аргумента оператора U, и регистр для значения его собственной частоты. В качестве начального состояния выберем $|\xi, \bar{0}\rangle$, где $\xi = \sum_k x_k \psi_k$, где ψ_k есть собственные состояния нашего оператора U, соответствующие собственным частотам w_k.

Центральным приемом для проявления собственных частот будет оператор, который для частного случая оператора численного умножения был придуман Шором, и был обобщен на случай произвольных операторов Абрамсом и Ллойдом. Его определение таково:

$$\mathrm{QFT}_2 \; U_{cond} \; QFT_2. \tag{5.3}$$

Здесь преобразование Фурье применяется ко второму регистру. Посчитаем, что получится в результате применения этой процедуры к нашему начальному состоянию. Первый оператор преобразования Фурье даст равномерное распределение амплитуд во втором регистре: $= \frac{1}{\sqrt{N}} \sum_k \sum_{\alpha=0}^{N-1} x_k |\psi_k, \alpha\rangle$. Оператор условного применения U, с учетом того, что ψ_k - собственные векторы U даст $U_{cond}|\psi_k, \alpha\rangle = |U^\alpha \psi_k, \alpha\rangle = e^{2i\pi w_k \alpha}|\psi_k, \alpha\rangle$, значит, все состояние после применения условного оператора превратится в $\frac{1}{\sqrt{N}} \sum_k \sum_\alpha e^{2i\pi w_k \alpha}|\psi_k, \alpha\rangle$. Наконец последнее применение Фурье даст состояние:

$$\frac{1}{N} \sum_k \sum_c \sum_{\alpha=0}^{N-1} e^{2i\pi\alpha(w_k - \frac{c}{N})}|\psi_k, c\rangle. \tag{5.4}$$

Если c есть как раз список бинарных знаков w_k, то показатель экспоненты - ноль, и у нас получится при суммировании по α сумма одних единиц в количестве N штук, то есть коэффициент при состоянии с таким c будет равен x_k. Из этого, в силу нормировки - сумма модулей в квадрате всех x_k есть 1 - следует, что при базисных векторах с другими c амплитуда равна нулю. Однако в этом можно убедиться и непосредственно: $\sum_{\alpha=0}^{N-1} e^{2i\pi\alpha\beta} = 0$ при $\beta \neq 0$. Действительно, Это есть сумма геометрической прогрессии со знаменателем не равным 1 для которой последний член равен первому.

Итак, результатом нашей процедуры будет состояние

$$\sum_k |\psi_k, w_k\rangle,$$

где под w_k подразумевается его бинарная запись. Тем самым если мы наблюдаем это результирующее состояние в базисе, состоящем из собственных векторов оператора U, то мы получим в качестве "приложения"к полученному собственному состоянию бинарную запись соответствующей собственной частоты. В частности, если исходное состояние ξ само было собственным, мы просто получим его частоту.

5.1.2 Реализация QFT на квантовом компьютере

Для того, чтобы применять изложенный метод поиска собственных частот, нам нужно малое - построить квантовый алгоритм для квантового преобразования Фурье, чем мы сейчас и займемся. Мы будем представлять этот алгоритм в несколько иной форме, чем алгоритм квантового перебора, а именно, в форме квантовой схемы из функциональных элементов, или quantum gate array (qga).

Такая схема представляет собой набор параллельных проводов, располагаемых обычно горизонтально, так что каждый провод представляет собой один определенный кубит из квантовой части компьютера, так что время течет слева направо. Провода соединены функциональными элементами (gates), каждый из которых представляет какое-то элементарное унитарное преобразование. При этом данные преобразования применяются в той последовательности и к тем кубитам, как это показано на данной qga. Таким образом, представление квантовых алгоритмов в форме qga эквивалентно тому, которое мы описали выше. Qga обычно используются просто для наглядности. Договоримся представлять целое число вида $a = a_0 + a_0 2 + \ldots + a_{l-1}2^{l-1}$ базисным состоянием $|a_0\ a_1\ \ldots\ a_{l-1}\rangle$ и располагать все a_j сверху вниз. Такое же соглашение примем и для выхода, только бинарные знаки b_j числа $b = b_0 + b_0 2 + \ldots + b_{l-1}2^{l-1}$ будем писать в обратном порядке - снизу вверх.

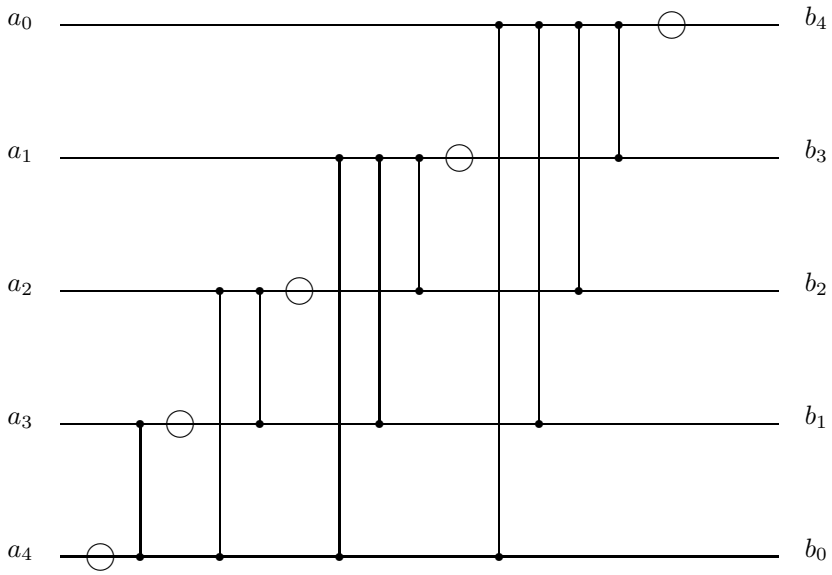

Рисунок 1. Квантовая схема для QFT^{-1}.

Окружности здесь обозначают преобразование W^1, двухкубитовые операции имеют вид:

$$U_{k,j} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\pi/2^{k-j}} \end{pmatrix},\ k > j. \tag{5.5}$$

Чтобы убедиться в этом, мы рассмотрим амплитуду перехода от базисного состояния a к базисному состоянию b. Это понятие законно, так называется соответствующий элемент матрицы рассматриваемого оператора. Здесь нам придется набраться терпения - подсчет идейно простой, но

требует некоторого занудства. Сначала заметим, что модули всех таких амплитуд одинаковы и, так же как в обратном преобразовании Фурье, равны $1/2^{l/2}$, так что следить надо только за фазовым сдвигом, т.е. за аргументом ϕ комплексной амплитуды $e^{i\phi}$. Мы будем учитывать этот набег фазы, суммируя вклады от преобразований Уолша с вкладами от двухкубитных фазовых сдвигов. Введем для упрощения счета такое сокращенное обозначение: $b'_j = b_{l-1-j},\ j = 0, 1, \ldots, l-1$ - это понадобится для того, чтобы в нужный момент учесть обратный порядок расположения бинарных разрядов в a и b. Представим себе, как меняются состояния при продвижении слева направо по проводам нашей схемы. Собственно переход от a к b происходит только при совершении операции Адамара, двухкубитные операции диагональны и базисные состояния не меняют, добавляя только слагаемые к фазе. Вклад от операции Адамара будет таким: $\pi a_j b'_j$. Это число не равно нулю только если оба j-х разряда наших входных и выходных числе равны 1, что в точности соответствует определению преобразования Адамара. Вклад от двухкубитовой операции при $j < k$ будет $\pi a_j b'_k / 2^{k-j}$, потому что состояние a меняется на b только при прохождении устройства Адамара, а как видно из рисунка 1, такая двухкубитовая операция совершается в момент, когда j-й кубит еще в состоянии a_j, а k-й - уже в состоянии b'_k. Суммируя все эти слагаемые фазового сдвига, и учитывая, что целое кратное π можно вообще в расчет не принимать, получаем вот что:

$$\begin{array}{l} \pi \sum\limits_{l>k>j\geq 0} \frac{a_j b'_k}{2^{k-j}} + \pi \sum\limits_{l>j\geq 0} a_j b'_j = \\ 2\pi \sum\limits_{l>j+k\geq 0} \frac{a_j b_k 2^{j+k}}{2^l} = \\ 2\pi \sum\limits_{l>j,k\geq 0} \frac{a_j b_k 2^{j+k}}{2^l} = \\ \frac{2\pi}{2^l} \sum\limits_{l>j\geq 0} a_j 2^j \sum\limits_{l>k\geq 0} b_k 2^j = \frac{2\pi}{2^l}. \end{array} \tag{5.6}$$

Это как раз то, что требуетсяв определении обратного преобразования Фурье. Если нам понадобится совершить прямое преобразование, достаточно обратить порядок функциональных элементов в рассматриваемой схеме и поставить знак минус перед фазовым сдвигом в определении двухкубитных операций.

А теперь посмотрим на то, что мы только что совершили. Построенная нами схема, реализующая преобразование Фурье, содержит порядка l^2 функциональных элементов. Заметим, что если мы не будем гнаться за точностью этого преобразования, то можно будет отбросить все двухкубитные операции, в которых участвуют слишком далекие друг от друга кубиты. Действительно, знаменатель в $\pi/2^{k-j}$ для них делает всю дробь пренебрежимо малой, экспонента будет почти единицей, т.е. такие преобразования почти единичны и их можно отбросить. Схема тогда значительно упростится - ее размер вообще будет линейным - порядка $C\ l$, где константа C будет, конечно, зависеть от выбранной нами точности.

Посмотрим на это с другой стороны. Матрица преобразования Фурье имеет размер $N \times N$, т.е. недоступна нам для непосредственного манипулирования. Памяти классического компьютера не хватит даже для того, чтобы записать ничтожную часть этой матрицы, не говоря уже о том, чтобы с ней что-то делать. Мы же только что построили схему квантового компьютера доступного (и даже очень маленького) размера, который делает эту колоссальную работу - оперирует с матрицей преобразования Фурье. А это значит, что этот небольшой вычислитель вполне может справиться и с таким заданием, как определение собственных чисел операторов, распознавание молекулярных структур и их спектров и делать все задачи, которые на этом основаны - а это огромный класс задач! Это - третье чудо квантовой информатики.

5.2 Факторизация, оптимизация, моделирование и распознавание

5.2.1 Факторизация целых чисел

Общий метод нахождения собственных значений, который мы изложили в предыдущем разделе, был первоначально изобретен Шором для частного случая, появляющегося в задаче факторизации целых чисел. Факторизация, или разложение целого числа на простые множители, является известнейшей вычислительной задачей. Известные классические алгоритмы ее решения требуют порядка $e^{a\ n^{1/3}}$ шагов (алгоритм Ленгстры). Таким образом, эта задача относится к категории трудно решаемых. Здесь мы изложим квантовый алгоритм для решения задачи факторизации, изобретенный Петером Шором. Этот алгоритм исторически был первым быстрым квантовым алгоритмом, решающим задачу огромной практической важности. Дело в том, что на факторизации целых чисел основан криптографический протокол RSA, который используется в огромном количестве коммерческих приложений, например в системе защиты популярной операционной системы Windows. При этом сам факт того, что задача факторизации считается трудноразрешимой, обеспечивает надежность защиты данных в этом протоколе. В криптографической защите Windows предусмотрен высший уровень защиты, такой, что для его преодоления нужно факторизовать числа с 200 десятичными знаками. Эта задача абсолютно не посильна даже для современных суперкомпьютеров. Квантовый компьютер всего с 1000 кубит менее чем 1ГГц частоты способен справиться с такой задачей за несколько минут. Таким образом, практическое создание такого рода машин означало бы конец современной криптографии.

Однако у алгоритма Шора есть и большое чисто теоретическое значение. На этом примере можно проиллюстрировать важность того контекста, в котором применяется преобразование Фурье, а именно вспомогательных преобразований. Дело в том, что основное время в этом алгоритме затрачивается не на сложное с классической точки зрения преобразование Фурье, а на умножение целых чисел!

Итак, займемся задачей факторизации. Пусть нам надо найти неизвестное нетривиальное разложение $q = q_1 q_2$ известного натурального числа q на множители. Эта задача может быть сведена к задаче отыскания наименьшего мультипликативного периода r произвольного целого числа y по модулю q: $y^r \equiv 1\ (mod\ q)$. Кратко говоря это сведение таково. Пусть у нас есть метод нахождения r. Будем выбирать y случайно, и находить r. Тогда с не исчезающе малой вероятностью окажется, что r четно. В этом случае получается $y^r - 1 = (y^{r/2} - 1)(y^{r/2} + 1) \equiv 1\ (mod\ q)$, тогда один из множителей является делителем кратного q числа и мы получаем с неисчезающей вероятностью факторизацию самого q. Таким образом, нам достаточно научиться быстро находить r при заданных q и y. Здесь можно усмотреть аналогию с нахождением неизвестного периода, только в качестве оператора U будет выступать операция умножения на число y.

Мы подберем n так, чтобы $2^{n-1} \leq q < 2^n$ и будем работать с квантовой памятью из n кубит. Рассмотрим такой оператор U: $U|x\rangle \longrightarrow |yx\ (\mathrm{mod}\ q)\rangle$, где yx есть просто числовое умножение. Чтобы сделать так, что он действует на всех наших базисных векторах, мы будем считать, что это равенство определяет оператор только на числах, менбших q, а на всех остальных: $q, q+1, \ldots, 2^n - 1$ пусть он действует как тождественный оператор. Имеется еще одно небольшое затруднение: такой оператор может быть не унитарным. Если y и q имеют общие делители, некоторые элементы будут "склеиваться". Для исключения такой неприятности будем считать что они взаимно простые: $(y, q) = 1$. Поскольку y выбирается случайно, то с неисчезающей вероятностью именно так и окажет-

ся. Теперь все в порядке. Мы можем применить к нашему оператору мощную технику квантового компьютинга, развитую нами. Собственные вектора U имеют вид $\frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} \exp(-2\pi ikj/r)|y^j \pmod q\rangle$ и соответствующие собственные значения будут $\exp(2\pi ij/r)$. Если мы применим процедуру проявления собственных частот из предыдущей части, то после измерения получим число j/r[1]. Зная с высокой точностью двоичное приближение дроби, легко найти ее знаменатель, если предположить, что дробь несократима. Формальный алгоритм для такого поиска основан на методе непрерывных дробей и может быть найден, например, в книге []. А эта дробь действительно будет несократима с неисчезающей вероятностью, поскольку всевозможные j здесь появляются равновероятно в том случае, если мы позаботимся о том, чтобы начальное состояние ξ для процедуры проявления (см. предыдущий параграф) выбиралось бы произвольно. Тогда мы, повторяя описанную процедуру многократно, будем достаточно часто получать значения j, взаимно простые с r, и таким образом сможем найти само r. Это и есть алгоритм Шора.

Теперь наступил самый интересный момент - мы будем подсчитывать, насколько же этот алгоритм хорош. Что касается преобразования Фурье, здесь все понятно - оно делается очень быстро, вообще говоря за время, линейное относительно длины записи числа q, которое нам надо факторизовать. Но имеется другая очень рутинная операция, которая грозит свести на нет все преимущество квантового преобразования Фурье. Это операциа умножения на число y, которая входит в оператор условного применения U_{cond}. Для получения U^α нам надо умножать на y α раз, а это порядка q действий. Эта трудность в общем случае применения преобразования Фурье в квантовом компьютинге носит принципиальный характер. Для произвольного оператора U ее нельзя устранить. Однако в случае факторизации нам фантастически везет - можно осуществить условное применение за время порядка $\log^2 q$. Чтобы умножить на число y^α мы будем получать само число y^α последовательным возведением в квадрат, начиная с y: $y, y^2, y^4, \ldots$. Разумеется, на каждом шаге мы будем брать остаток при делении на q. Так можно довести дело до ближайшей к y степени двойки: 2^{l_1}. Затем возьмем частное $q/2^{l_1}$ и поступим с ним так же, и т.д. Тогда мы доберемся до y за время порядка логарифма от q, то есть за число шагов, порядка длины записи числа q. На каждом шаге надо потратить $log^2 q$ действий для вычисления произведения чисел методом "столбика", и в итоге мы получим реализацию оператора условного применения U за время $O(\log^3 q)$. Это и будет сложностью алгоритма Шора. Мы видим, что наиболее трудоемкой частью этого алгоритма является рутинная операция, входящая в процесс подготовки входного состояния для преобразования Фурье - последовательное умножение натуральных чисел.

5.3 Моделирование квантовых систем

Теперь обратимся к задаче квантового моделирования физических систем. Это как раз и есть та задача, которую имел в виду Фейнман, выдвигая идею квантового компьютера. Состояние многочастичной системы может быть описано набором чисел, выражающих значение физических величин, таких как массы, координаты, скорости, время и т.д. Эти числа (в отличие от амплитуд) вещественные. Более того, при надлежащем ограничении области рассмотрения и разрешимости измеряющего прибора можно считать, что все они представляются в виде $\frac{l}{2^n}$, где n разумной величины число. Тогда базисное состояние рассматриваемой многочастичной системы можно представить как базисный же вектор в пространстве состояний квантовой памяти из n кубит. Соответственно, линейной

[1] В действительности мы получим приближение этого числа с точностью до $O(1/N)$ с высокой вероятностью. Детали доказательства можно найти в статье []

комбинации базисных состояний изучаемой системы будет отвечать состояние квантового компьютера с точно такими же амплитудами. Кубиты нашего квантового компьютера для моделируемой системы носят виртуальный характер, т.е. мы не можем приписать им никакого естественного физического смысла. Однако в нашем квантовом компьютере, который будет моделировать изучаемую систему, это реальные, "физические"кубиты. Такой подход к описанию физических систем можно назвать "кубитовым". Мы увидим, что такой подход к описанию физики принципиально более эффективен, чем традиционный "битовый"подход, используемый при численном моделировании многочастичных процессов на классических компьютерах. Для этого попробуем решить на квантовом компьютере уравнение Шредингера. Здесь снова ключевую роль будет играть быстрое преобразование Фурье, но использоваться будет немного иное его свойство, чем раньше. Это свойство состоит в том, что преобразование Фурье переводит операцию дифференцирования в операцию умножения на независимую переменную с мнимым коэффициентом. Таким образом, если применить его к волновой функции, получится, что оператор двойного дифференцирования, входящий в Гамильтониан, превратится для Фурье-образа волновой функции в оператор умножения на квадрат независимой переменной этого Фурье-образа с неким коэффициентом, а эта переменная есть не что иное как импульс. Эта идея, хорошо известная физикам, вручную решающим волновое уравнение, великолепно работает и для квантового компьютера. Надо лишь убедиться, что квантовое преобразование Фурье обладает аналогичным свойством, связанным с операцией дифференцирования (для нашего квантового симулятора роль дифференцирования будет играть соответствующая конечная разность). Это следует из того, что QFT является приближением оператора настоящего преобразования Фурье при переходе к кубитовому представлению волновой функции.

Нашей целью будет получение состояния нашего квантового компьютера, соответствующего состоянию изучаемой системы в некоторый момент времени t. Нам нужно с помощью рабочих преобразований приблизить действие оператора эволюции e^{-iHt} на волновую функцию ψ_0, где $H = H_p + H_q$, $H_p = \frac{p^2}{2m}$, $H_q = V(q)$, $p = \frac{1}{i}\frac{\partial}{\partial q}$ и потенциал $V(q)$ есть вещественная функция. Для простоты возьмем время t равным единице. Реализовать на квантовом компьютере действие H_q просто. Поскольку матрица этого оператора (а значит и e^{iH_q}) диагональна, для этого надо всего лишь изменить фазы в зависимости от вида базисных состояний, - а это делается примерно так же, как инверсия нулевого состояния в алгоритме Гровера. Однако со вторым слагаемым Гамильтониана придется повозиться. Трудность заключается в том, что оператор H_p не будет диагональным в выбранном нами "координатном"базисе. Однако мы уже знаем, как свести дело к простому диагональному случаю: надо перейти к импульсному базису, иными словами, совершить преобразование Фурье - а это у нас очень хорошо получается. Для этого выберем маленький интервал времени Δt представим приближенно наш эволюционный оператор таким образом:

$$e^{-iH} \approx (e^{-iH_q \Delta t}\ e^{-iH_p \Delta t})^{1/\Delta t}. \tag{5.7}$$

В справедливости этого представления легко убедиться, воспроизведя выкладки, ведущие ко второму замечательному пределу, служащему определением числа e. Мы выбрали "координатный"базис, так что H_q имеет диагональный вид. Применяя квантовое преобразование Фурье: QFT : $f \longrightarrow \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-ipq} f(q)\ dq$ и его свойство переводить дифференцирование $\partial/\partial\ q$ в уиножение на ip, мы можем представить действие импульсной части оператора как $e^{-iH_p} = \mathrm{FT}^{-1}\ e^{-ip^2\Delta t/2m}\ \mathrm{FT}$, где средний оператор имеет диагональный вид. Теперь последовательные применения QFT и фазового сдвига на $-p^2/2m$ при реализации последовательности (5.9) дают требуемое приближение.

Сравним квантовый метод решения уравнения Шредингера с традиционным методом конечных разностей. При описанном методе у нас получается экспоненциальный выигрыш по памяти,

поскольку для разностного метода нужно хранить в памяти всю волновую функцию, а квантовая память позволяет обойтись только логарифмическим числом кубит. Выигрыш по времени будет примерно таким же. Более подробный анализ (см. []) показывает, что эволюцию на отрезке времени t можно квантово моделировать за время порядка t^2. Важно то, что с ростом числа частиц в исследуемой системе (q может с аналогичным успехом изображать и набор многих координат) при фиксированной точности моделирования число необходимых кубит симулятора растет линейно. Таким образом, возможно эффективное моделирование многочастичных систем на квантовом компьютере разумного размера.

Интересно, что в некоторых случаях можно не просто моделировать поведение исследуемой многочастичной системы, но и предсказывать ее состояние. Иначе говоря, моделировать с опережением, когда для получения состояния, соответствующего системе в момент t на симуляторе требуется меньшее время $t' < t$. Такая возможность открывается в случае, если у Гамильтониана исследуемой системы редкий спектр, то есть собственные частоты группируются вокруг небольшого (по сравнению с размерностью задачи) числа характерных частот. Соответствующий метод описан в работе [].

Рассмотрим одномерное движение точечной частицы единичной массы в поле с потенциалом $V(t, X)$, где t - время, X - координата. Обозначим импульс ее через P. Гамильтониан для такой частицы будет иметь вид $H = \frac{P^2}{2} + V(t, X)$ как в классическом, так и в квантовом случае, только в последнем $P = \frac{1}{i}\frac{\partial}{\partial X}$. Квантовое описание этого движения мы дадим в форме имитации на квантовом компьютере, которая предложена в работе [Za, Wi], и которая полностью эквивалентна прямому решению краевой задачи для уравнения Шредингера методом конечных разностей. Эта краевая задача имеет вид:

$$i\frac{\partial \Psi}{\partial t} = \frac{P^2}{2} + V(t, X), \qquad \Psi(0) = \Psi_0. \tag{5.8}$$

Ее решение дается формулой $\Psi(t, X) = \exp(-i\int\limits_0^t H(t, X)dt)\Psi_0$. Метод имитации на квантовом компьютере состоит в следующем.

Мы будем приближать действие оператора e^{-iHt} на on волновой функции ψ_0 где $H = H_p + H_q$, $H_p = \frac{p^2}{2m}$, $H_q = V(q)$, $p = \frac{1}{i}\frac{\partial}{\partial q}$ и потенциал $V(q)$ есть вещественная функция. Без ограничения общностимы можем взять $t = 1$. Чтобы иметь полезное приближение, мы должны работать в координатном или импульсном базисе в пространтве векторов состояний, причем ни в одном из этих случаев Гамильтониан не будет диагональным. Чтобы свести проблему к простому диагональному случаю, мы выберем малый промежуток времени Δt и представим наш оператор эволюции приближенно как

$$e^{-iH} \approx (e^{-iH_q\Delta t}\ e^{-iH_p\Delta t})^{1/\Delta t}. \tag{5.9}$$

Выбирая, скажем, координатный базис, мы имеем диагональный оператор H_q. Применив преобразование Фурье FT : $f \longrightarrow \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} e^{-ipq} f(q)\ dq$ и его свойство заменять производную $\partial/\partial\ q$ на сомножитель ip мы можем представить действие импульсного слагаемого Гамильтониана как $e^{-iH_p} = \mathrm{FT}^{-1}\ e^{-ip^2\Delta t/2m}\ \mathrm{FT}$ где средний оператор диагонален. Если мы можем осуществить преобразование Фурье и фазовый сдвигна $-p^2/2m$, то последовательное выполнение этих операторов из (5.9) даст нам требуемое приближение. Пусть волновая функция $\psi(q)$ определена на сегменте $(-A, A)$ и ее импульсное представление FT ψ определено на сегменте $(-B, B)$. Выбирая малые значения Δq и Δp, мы можем приблизить ее через $\sum\limits_{a=0}^{2A/\Delta q} \psi(q_a)\delta_a$ где $\delta_a(q)$ принимает значение 1 на

сегменте $(q_a, q_a + \Delta q)$ и нуль для других q. Теперь мы можем приблизить преобразование Фурье с помощью линейного оператора, который при действии на δ_a дает $\frac{1}{\sqrt{2\pi}}\Delta q \sum_{b=0}^{2B/\Delta p} e^{-ip_b q_a}\sigma_b(p)$ где $\sigma_b(p)$ есть одноступенчатая импульсная функция, аналогичная δ_a. Вводя новые одноступенчатые функци для координат и импульсов через $d_a(q) = \delta_a(q-A)$, $s_b(p) = \sigma_b(p-B)$, мы перепишем преобразование Фурье в виде

$$d_a \longrightarrow \frac{1}{\sqrt{2\pi}}\Delta q \sum_{b=0}^{2B/\Delta p} e^{-i\ ba\Delta q\ \Delta p} s_b \tag{5.10}$$

что выглядит похоже на квантовое преобразование Фурье. Предположим, что физическое пространство зернисто как в координатном, так и в импульсном представлении с размерами зерен Δq и Δp соответственно. Тогда рассматриваемая частица может находиться только в точках вида q_a или может иметь импульс только вида p_b. Мы ассоциируем положение q_a; $a = 0, 1, \ldots, N = 2^l$ с базисным состоянием $|a\rangle$ l кубитной квантовой системы. Для простоты выберем такую единицу длины, что $\Delta q = \Delta p = \sqrt{2\pi}/\sqrt{N}$ и пусть $A = B = \sqrt{\pi N/2}$. Тогда (5.10) будет соответствовать квантовому преобразованию Фурье вида

$$\text{QFT}: \ |a\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{b=0}^{N-1} e^{-\frac{2\pi i\ ab}{N}} |b\rangle, \tag{5.11}$$

и фазовый сдвиг на $-p^2\Delta t/2m$ из (5.9) будет соответствовать фазовому сдвигу на $-\pi b^2\Delta t/mN$. Обратное преобразоване Фурье будет при нашей дискретизации задачи соответствовать обратному к квантовому преобразованию Фурье. При моделировании s_1 частичной системы мы должны завести s_1 копий квантового регистра для одной частицы и выполнить описанную процедуру для полученной объединенной квантовой памяти.

Аналогично рассматривается случай вращательного движения твердого тела относительно некоторой точки (как правило - центра масс). Здесь Гамильтониан будет иметь вид

$$H^1 = H^1_{kin} + V(\mu, \nu, \eta), \quad H^1_{kin}(\mu, \nu, \eta) = \frac{J_x^2}{2I_x} + \frac{J_y^2}{2I_y} + \frac{J_z^2}{2I_z}$$

где потенциальная и кинетическая энергии вращения выражены через углы Эйлера μ, ν, η, определяющие положение вращающегося вокруг одной неподвижной точки тела, I обозначает моменты инерции относительно соответствующих осей, оператор момента $\bar{J} = [\bar{r} \times \bar{p}]$, где r и p - операторы координаты и импульса. Пусть уровень r фиксирован. Шаг моделирующего алгоритма является элементарным рассеянием неупругого типа, в котором может измениться не только внутренняя энергия, но и химический состав расеивающихся частиц.

Оператор эволюции имеет вид $U = \exp -iH\delta t$, где гамильтониан $H_j = H_j^0 + H_j^1$ состоит из слагаемого $H_j^0 = \frac{P_j^2}{2m_j} + V(s_j, X_j)$, соответствующего движению X_j^0, и H_j^1 соответствует вращению j- го участка как целого вокруг точки X_j. Последнее рассматривается совершенно аналогично движению точечной частицы, и мы ограничимся этим последним случаем.

При дискретизации импульсы и координаты будут представляться в виде бинарных чисел x и p из интервала $[0, 1)$, которые связаны с физическими величинами равенствами: $P = \sqrt{2\pi/N}p$, $X = \sqrt{2\pi/N}x$. Мы будем предполагать, как это всегда делается, что производная потенциала $V'(X)$

меняется достаточно медленно, а именно, в пределах точности определения X эту производную можно считать постоянной. Цепочка, соответствующая H_p, имеет вид

$$x \longrightarrow \sum_p e^{\frac{-2\pi i\ xp}{N}}|p\rangle \longrightarrow \frac{1}{\sqrt{N}}\sum_p e^{\frac{-2\pi i\ xp}{N}} e^{iP^2\delta t\pi/N}|p\rangle$$

Пусть $\theta_p(\kappa) = a$. Это означает, что мы должны сделать замену P на $P + A$ в амплитудах предыдущего выражения, $A = \sqrt{2\pi/N}a$. Это даст после обратного преобразования Фурье :

$$\frac{1}{N}\sum_y \lambda_y|y\rangle, \qquad \lambda_y = \sum_p e^{ip^2\delta t\pi/N} e^{-2\pi i\ xa/N} e^{\frac{ip(-2\pi X + 2\pi A\delta t + 2\pi Y)}{N}}|y\rangle$$

Первые два сомножителя в выражении для амплитуды не имеют физического смысла, так как дают общий фазовый сдвиг. Мы имеем $\lambda_y = 0$ для всех y кроме того, что соответствует положению частицы $Y = X - A\delta t$, то есть мгновенная скорость будет численно равна P, что означает наличие одного из уравнений классической динамики: $X' = \partial H/\partial P$. Для получения другого уравнения рассмотрим переход, соответствующий применению координатной части гамильтониана: $H_x = V(X)$. Начнем с состояния $|p\rangle$. Тогда применение оператора QFT H_x QFT^{-1} даст такой переход

$$|p\rangle \longrightarrow \frac{1}{\sqrt{N}}\sum_x e^{\frac{2\pi ixp}{N}}|x\rangle \longrightarrow \frac{1}{\sqrt{N}}\sum_x e^{\frac{2\pi ixp}{N}} e^{iv(x)}|x\rangle \longrightarrow \frac{1}{N}\sum_{p_1}\lambda_{p_1}|p_1\rangle,$$

где

$$\lambda_{p_1} = \sum_x e^{2\pi ix(\frac{p}{N} + \frac{v(x)\delta t}{2\pi x} - \frac{p_1}{N}}.$$

Тогда $\lambda_{p_1} \neq 0$ будет тогда и только тогда, когда выполняется равенство

$$P_1 = P + \delta t\frac{V(X)}{X},$$

что в силу нашего предположения о медленности изменения градиента потенциала дает нам второе уравенение классической динамики:

$$P' = \frac{\partial H}{\partial x}.$$

5.3.1 Распознавание структур и функций

Список задач, в которых применение квантовых вычислений дает принципиальный выигрыш, слишком велик для того, чтобы приводить его в этой небольшой книжке. Мы завершаем эту тему рассмотрением задач распознавания устройств, которые можно кратко сформулировать как прямую и обратную задачи:

1) по заданной схеме устройства определить его функции,

2) по заданным функциям восстановить схему устройства.

Задачи такого типа часто встречаются в инженерной практике. Мы используем понятие собственных частот для уточнения этих формулировок. Будем считать, что устройством будет схема из квантовых элементов, а функцией этого устройства будем называть унитарный оператор, который она генерирует. Тогда функция будет полностью задана, если мы зададим собственные частоты,

и для каждой собственной частоты укажем соответствующее ей подпространство собственных векторов. Сначала мы сузим задачу и будем рассматривать только собственные частоты, игнорируя собственные векторы. Здесь получаются такие вопросы:

а) По данной схеме, генерирующей оператор U, и числу $w \in [0,1)$ определить, является ли оно собственной частотой оператора U.

б) По данному набору чисел и схеме определить, входит ли спектр (набор собственных частот) оператора, генерируемого этой схемой в данный набор.

в) По данному набору частот построить схему, генерирующую соответствующий оператор.

Другого типа задача получится, если использовать физическую функциональность прибора с неизвестной схемой:

г) Имея оракул для оператора U или U_{cond} найти генерирующую его квантовую схему.

Эти задачи расположены в порядке нарастающей трудности. Рассмотрим вкратце общую схему решения первой из них. Обозначим построенный ранее оператор, выявляющий собственные частоты, через Rev. Аналогичным способом мы можем построить обратный оператор. Теперь, имея состояние $\sum\limits_k x_k|\psi_k\rangle$, мы можем обратить знак при том ψ_k, который соответствует заданной частоте w. Делается это так: сначала с помощью Rev мы проявляем значение частоты ω_k - она появляется в дополнительном регистре, затем меняем знак при условии $w = \omega_k$ - это делается так же, как и инверсия относительно нуля в GSA, а затем с помощью Rev^{-1} очищается дополнительный регистр. Теперь, имея инверсию относительно подпространства собственной частоты w, мы можем применить схему GSA и получить вектор из этого подпространства, после чего, опять таки с помощью Rev, проверить, действительно ли этот вектор - собственный с собственной частотой w. Если w - собственная частота, все упомянутые инверсии будут нетривиальными, если же это - не собственная частота, инверсии будут тождественными, и мы в результате вместо вращения текущего вектора по методу GSA получим просто отсутствие всякого движения - исходный вектор алгоритма будет и результирующим. При этом будут появляться неточности из-за незнания нами момента окончания процесса GSA. С ними можно успешно бороться, используя параллельные вычисления[2]

Перейдем ко второй задаче. Здесь нам придется устраивать два вложенных GSA- процесса, действующих на двух разных регистрах. Внешний (основной) GSA - процесс действует на регистре для частот, его цель - найти частоту, входящую в заданный список и не являющуюся собственной. Для этого необходима инверсия относительно таких "нехороших"частот. Для построения этой инверсии используется алгоритм из предыдущего пункта, содержащий внутренний GSA - процесс, действующий при фиксированном содержимом регистра частот на регистре для аргумента U.

Наконец, для решения третьего типа задач придется использовать GSA -процессы тройной вложенности. Для основного внешнего процесса мы заведем новый регистр, в котором будут храниться двоичные коды возможных схем. Рассмотренный в предыдущем пункте алгоритм войдет во внутренний процесс, задачей которого будет инверсия вдоль кода тех самых схем, которые генерируют операторы с подходящим спектром. Эта инверсия используется во внешнем GSA - процессе, получающем код искомой схемы.

В рассмотренных задачах использовались по-существу только коды схем. Задача типа г) имеет существенное отличие. Здесь нам потребуется оракул для преобразования U_{cond}. Мы не будем здесь обсуждать вопрос о возможности использования самого оператора U. Для подбора схемы, соответствующей данному оракулу, опять используется GSA - процессы тройной вложенности. Смысл регистров будет таким же, как и в предыдущем случае.

[2]Мы отсылаем интересующихся подробностями к статье [].

Глава 6

Реалистические модели квантовых компьютеров

Здесь мы изучим фермионные вычисления в формализме чисел заполнения, предложенные в статье quant-ph/0003137. Показано, что для выполнения произвольного квантового вычисления с допущением только незначительного замедления достаточно управлять только туннелированием. При этом взаимодействие между кубитами диагонального типа будет оставаться непрерывным и неуправляемым. Обоснование этого подхода дано с помощью редукции к стандартной модели квантовых вычислений в Гильбертовом пространстве с использованием результатов статьи quant-ph/0202030 об однокубитном управлении.

6.1 Введение

Квантовый компьютер является беспрецедентным экзаменом для современной физики, поскольку он требует такого уровня контроля над объектами наноразмеров, который никогда не достигался искусственным путем. В то время как математическая теория квантового компьютинга хорошо развита, его физическая реализация представляет серьезный вызов нашему пониманию Природы. Вот почему очень важно искать простейшие возможные способы такой реализации, так чтобы они базировались только на основных принципах квантовой механики и содержали бы минимум всевозможных технологических трудностей. Два требования можно сформулировать для таких схем: адекватное описание состояний, формирующих вычислительное Гильбертово пространство, и реалиичтический метод управления вычислениями. Обычно вычислительный элемент - кубит представляется как некая характеристика подобная спину, заряду или положению некоторой элементарной частицы. Этот подход хорошо работает для изолированного кубита. Для системы из нескольких кубитов этот подход встречает серьезные трудности. Эти трудности происходят из фундаментального физического принципа неразличимости (или тождественности) частиц одного вида. Чтобы управлять вычислением, мы должны иметь возможность обращения к отдельному кубиту, тогда как различные частицы в принципе являются неразличимыми. Конечно, мы можем различить частицы, размещая их на достаточно большом расстоянии друг от друга, однако в этом случае нам будет очень затруднительно держать их в запутанном состоянии, что совершенно необходимо для

производства квантовых вычислений. Одно из решений этой дилеммы было найдено Китаевым и Бравым, которые предложили использовать Фоковское пространство чисел заполнения для описания квантовых вычислений (см. [KB]). Здесь используется естественная идентификация кубитов с энергетическими уровнями в Фоковском пространстве, так что единица трактуется как занятый уровень, а нуль - как свободный. Такой подход дает универсальные квантовые вычисления за довольно дорогую цену: требуется управлять не только внешним полем и туннелированием, но также и посроянным диагональным взаимодействием между кубитами, а также еще и контактом со сверхпроводником, то есть управление коэффициентами α, β, γ в (6.9) и управление дополнительным слагаемым $\delta a_k^+ a_j^+ + \delta^* a_k a_j$.

В настоящей работе мы увидим, как уменьшить эту цену, используя идею о непревывном и неконтролируемом взаимодействии. Для этого нам понадобится две вещи: предположение о том, что исходный Гамильтониан взаимодействия для вычислительной системы состоит только из диагонильных и туннельных членов, а также модифицированное соответствие между состояниями в пространстве чисел заполнения и вычислительным Гильбертовым пространством. Тогда для контроля над квантовыми вычислениями нам понадобится только управлять внешним полем и туннелированием. Такой тип контроля в принципе осуществим посредством лазеров. Основная схема дана ниже. Она базируется на идее о непрерывном неуправляемом взаимодействии, предложенной в работах [Oz, OF] и адаптированной к языку Фоковского пространства чисел заполнения.

6.1.1 Однокубитовое управление в квантовых вычислениях

Основная трудность в практической реализации квантовых вычислений заключается в том, что технически очень сложно выполнить двухкубитные преобразования, играющие принципиальную роль в таких вычислениях. Для того, чтобы выполнить такое преобразование, мы должны управлять степенью запутанности частиц, которая определяется перекрытием пространственных частей их волновых функций. Однако, для производства вычислений требуется также и четкое различение частиц, что возможно лишь если перекрытие волновых функций достаточно мало. Таким образом, налицо противоречивость требований к физической реализации квантовых вычислений. Мы видим, что двухкубитные преобразования совершать во всяком случае значительно сложнее, чем однокубитные. Здесь уместно предложить такой подход. Поскольку взаимодействие частиц с изменением степени запутанности вытекает из волнового уравнения и подтверждается в экспериментах, то двухкубитные преобразования происходят в ходе естественной временной эволюции квантовой системы. Управлять же такой системой можно при помощи только однокубитных преобразований, которые гораздо проще контролировать. Итак, мы будем производить вычисления, воздействуя на систему только однокубитовыми импульсами, а двухкубитовые преобразования будут совершаться в фоновом режиме, и мы ими управлять не будем. В этом суть предлагаемой модели вычислений с однокубитным управлением. Такая модель является гораздо более реалистической, чем абстрактная схема квантового компьютера, предполагающая управление двухкубитным взаимодействием. Отложим пока вопрос о возможностях такого подхода, и продемонстрируем, как можно в рамках предлагаемой модели решить конкретную задачу - моделировать поведение многочастичной системы с квадратичным взаимодействием с диагональной матрицей. Основная трудность предлагаемой модели с однокубитовым управлением состоит в том что двухкубитное взаимодействие происходит неконтролируемым образом, в частности и с посторонними кубитами, что серьезно искажает схемы квантовых алгоритмов. Для производства вычислений в такой модели нам нужно создать методы коррекции "нежелательных"преобразований с помощью однокубитных импульсов. Для демонстрации возможностей такого подхода мы сначала покажем, как осуществить квантовое преобразование

Фурье в рамках такой модели. Нашим основным предположением будет то, что матрица Гамильтониана двухкубитного взаимодействия имеет диагональную форму. Однако сначала для удобства мы наложим также некоторые ограничения на скорость убывания этого взаимодействия с расстоянием. А именно, будем предполагать, что потенциал взаимодействия падает с расстоянием как потенциал Юкавы. Затем развитый метод мы сможем применить и для более широкого класса диагональных взаимодействий. Более того, можно будет обобщить этот подход и на тот случай, когда разные пары кубитов взаимодействуют по-разному. Наконец, данный подход будет применен к решению волнового уравнения для системы из нескольких частиц с потенциалом квадратичного типа. potentials.

Реализация квантового преобразования Фурье на однокубитовом управлении

Квантовое преобразование Фурье - это ключевая подпрограмма в квантовом компьютинге. Она используется в большом числе других алгоритмов (см. например [Sh, AL, Oz]). Простейшая схема квантовых функциональных элементов, реализующая это преобразование, дана на рисунке 1. Она была использована Шором для быстрой квантовой факторизации (см. [Sh]). Мы, как и прежде, договоримся представлять целое число вида $a = a_0 + a_0 2 + \ldots + a_{l-1}2^{l-1}$ базисным состоянием $|a_0\ a_1\ \ldots\ a_{l-1}\ \rangle = |a\rangle$. Эти состояния формируют ортонормированный базис для входных состояний квантовой схемы функциональных элементов. Располагаем их сверху вниз. Такое же соглашение примем и для выходных состояний схемы, только двоичные b_j знаки числа $b = b_0 + b_0 2 + \ldots + b_{l-1}2^{l-1}$ будем располагать в противоположном порядке. Эта схема выполняет обратное преобразование за QFT^{-1} in $O(l^2)$ шагов, тогда как его матрица $N = l^2$ - мерна. Однако в этой схеме требуется двухкубитовое управление - она непосредственно не может быть реализована в рамках нашей модели. Мы покажем, как это можно сделать. Мы будем рассматривать взаимодействия, имеющие вид

$$\text{A) } H = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \rho \end{pmatrix}, \quad \rho > 0, \qquad \text{B) } H = \begin{pmatrix} \rho_1 & 0 & 0 & 0 \\ 0 & \rho_2 & 0 & 0 \\ 0 & 0 & \rho_3 & 0 \\ 0 & 0 & 0 & \rho_4 \end{pmatrix}, \tag{6.1}$$

где все $\rho = \rho_0 \frac{e^{-br}}{r}$; $b = const$; r есть расстояние между кубитами-частицами и $\rho_1 + \rho_4 \neq \rho_2 + \rho_3$. Расположим l кубитов на одной линии с равными интервалами. Пусть взаимодействие между jм и kм кубитами имеет Гамитониан $H_{j,k}$ вида (6.1). Этот тип Гамильтонианов возникает, например, в модели Изинга для частиц со спином $1/2$. Требуемое уменьшение взаимодействия с расстоянием можно достигнуть, помещая кубиты в подходящую потенциальную яму. Выбрав подходящую единицу длины, мы можем добиться $b = 1$. Сначала будем изучать случай взаимодействия вида (6.1, A) а затем распространим результаты на случай (6.1, B).

Реализация QFT с точностью до фазового сдвига

Примем, что QFT и его обратное имеют вид:

$$QFT:\ |a\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{b=0}^{N-1} e^{-\frac{2\pi i\ ab}{N}} |b\rangle, \qquad QFT^{-1}:\ |a\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{b=0}^{N-1} e^{\frac{2\pi i\ ab}{N}} |b\rangle. \tag{6.2}$$

Тогда обратное преобразование можно выполнить с помощью такой схемы.

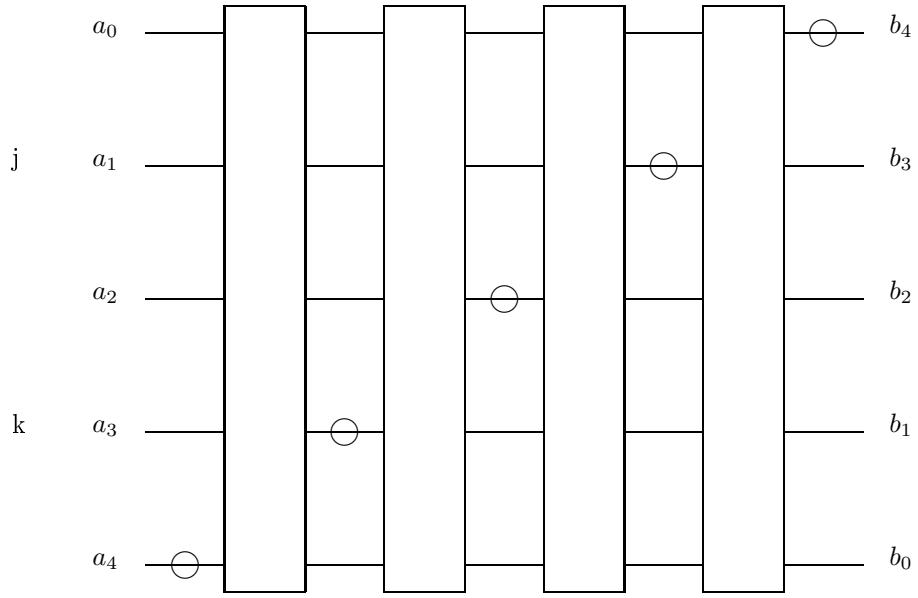

Рисунок 2. Прямоугольники обозначают непрерывное взаимодействие вида (6.1, A), кружки - операции Адамара

Здесь прямоугольники обозначают унитарные преобразования вида $U = e^{-i\tilde{H}}$, где $\tilde{H} = \sum_{l>j>k\geq 0} \tilde{H}_{j,k}$, а каждое из $\tilde{H}_{j,k}$ имеет вид (6.1 , A) с $\rho_0 = \pi$, $r = j - k$. Если мы выберем единицу времени так, чтобы постоянная Планка, умноженная на ρ_0 равнялась π и единицу длины так, чтобы $r = j - k$, то U будет в точности преобразованием вектора состояния, индуцируемым рассматриваемым Гамильтонианом в единичное время. Здесь мы предполагаем, что время производства однокубитных операций пренебрежимо мало по сравнению с единицей, так что взаимодействие пар кубит не может сильно изменить фазу за это время. Эта схема может быть получена из предыдущей вставкой недостающих элементов, соответствующих взаимодействию, происходящему в реальных системах с таким Гамильтонианом. Чтобы доказать, что эта схема действительно выполняет QFT^{-1}, мы применим метод подсчета амплитуд, предложенный в статье Шора [Sh]. Если дано базисное входное состояние $|a\rangle$, мы рассмотрим соответствующее выходное состояние. Это выходное состояние будет линейной комбинацией базисных состояний $|b\rangle$ с некоторыми амплитудами. Все модули этих амплитуд одинаковы и равны $1/\sqrt{L}$ и нам достаточно только следить за их фазами. Для простоты введем обозначение $a'_j = a_{l-1-j},\ j = 0, 1, \ldots, l-1$. В процессе применения нашей схемы значения кубитов с номерами j и $k \leq j$ проходят через элементы из рисунка 2 слева направо. Следуя этому направлению мы выделим следующие 4 типа взаимодействий: взаимодействие a'_j с собой и a'_k с собой в операции Адамара, заимодействие a'_j с a'_k $(j > k)$, взаимодействие a'_j с b_k для $j > k$, и взаимодействие b_j с b_k $(j > k)$. Время этих взаимодействий будет таким: ноль, k, $j - k$ и $l - 1 - j$ соответственно. Складывая вклады этих взаимодействий в фазу, мы получим результирующую фазу

вида

$$\pi \sum_{l>j>k\geq 0} \frac{a'_j a_k k}{2^{j-k}(j-k)} + \pi \sum_{l>j>k\geq 0} \frac{a'_j b_k (j-k)}{2^{j-k}(j-k)} + \pi \sum_{l>j\geq 0} a'_j b_j + \pi \sum_{l>j>k\geq 0} \frac{b_j b_k (l-j-1)}{2^{j-k}(j-k)}. \tag{6.3}$$

Обозначая первое и последнее слагаемые через A и B соответственно. Их вклад соответствует действию диагональных Гамильтонианов на $|a\rangle$ и $|b\rangle$ соответственно. Пока оставим эти вклады. Займемся суммой второго и третьего члена этой суммы. После замены j на $l-1-j$ эта часть приобретет вид

$$\pi \sum_{l-1>k+j\geq 0} \frac{a_j b_k 2^{j+k}}{2^{l-1}} + \pi \sum_{l-1\geq j\geq 0} a_{l-1-j} b_k \quad = 2\pi \sum_{l>k+j\geq 0} \frac{a_j b_k 2^{j+k}}{2^l} = 2\pi S + 2\pi \sum_{l>k,j\geq 0} \frac{a_j b_k 2^{j+k}}{2^l} = 2\pi S + 2\pi \frac{ab}{2^l} \tag{6.4}$$

для некоторого целого S. Первое слагаемое здесь фазы не меняет и мы получаем то, что требовалось с точностью до вклада A и B.

Коррекция сдвига фазы

Для того, чтобы учесть вклад диагональных слагаемых A и B в фазу, мы применим один трюк. Сначала рассмотрим только слагаемое A. Оно состоит из членов вида $A_{j,k} = c_{j,k} a'_j a'_k$, где $c_{j,k}$ зависит только от j и k, но не от a. Объявим jй и kй кубиты выделенными. Мы будем применять однокубитное преобразование NOT несколько раз ко всем кубитам, кроме выделенных, с целью подавить все взаимодействия, кроме взаимодействия между выделенными кубитами. Сначала рассмотрим пару невыделенных кубитов с номерами $p, q,\ q > p$. Их непрерывное взаимодействие за время Δt дает слагаемое $d_{p,q}\Delta t\ a'_p a'_q$ в фазу, где вещественное число $d_{p,q}$ зависит только от того, как быстро взаимодействие падает с расстоянием, но не от a'_p, a'_q. Например, для убывания типа Юкавы мы будем иметь $d_{p,q} = e^{-|q-p|}/|q-p|$. Теперь инвертируем один из этих двух кубитов, неважно какой, допустим, qth с помощью операции NOT. Его состояние будет $1-a'_q$. Теперь второй период длительностью Δt непрерывного взаимодействия даст слагаемое $d_{p,q}\Delta t\ a'_p(1-a'_q)$ в фазу. Наконец, восстановим содержимое qго кубита вторым применением NOT. Результирующий фазовый сдвиг в этих четырех действиях составит $d_{p,q}\Delta t\ a'_p$ и он зависит только от содержимого pго кубита. Теперь мы можем компенсировать этот фазовый сдвиг с помощью только одного однокубитного преобразования. Если рассмотреть пару кубитов с номерами p, q где один, скажем pй выделен, а другой - нет, тогда мы можем скомпенсировать их взаимодействие, используя только однокубитные операции - два NOT-а для qго и некоторый фазовый сдвиг для pго. Теперь нам надо так модифицировать этот метод, чтобы скомпенсировать все влияния невыделенных кубитов одновременно. Для этого мы будем выполнять операции NOT над каждым таким кубитом с достаточно малыми интервалами так что вклады в фазу невыденных кубитов будут друг друга сокращать. Есть два способа сделать это: использовать случайный процесс для генерации моментов для однокубитных операций, или осуществлять их периодически с разными периодами для разных кубитов. Сначала рассмотрим первый подход.

Метод случайных процессов

Для каждого невыделенного кубита с номером p рассмотрим Пуассоновский процесс $\mathcal{A}_p$, генерирующий моменты времени $0 < t_1^p < t_2^p < \ldots < t_{m_p}^p < 1$ с некоторой фиксированной плотностью $\lambda \gg 1$. Пусть все $\mathcal{A}_p$ независимы. Теперь выполним операции NOT над каждым кубитом с номером p

в моменты t_m^p последовательно. В момент 1 выполняем NOT над pм кубитом еслим и только если m_p нечетно. Таким образом, после этой процедуры каждый кубит восстановит свое исходное значение. Посчитаем фазовый сдвиг, порождаемый этой процедурой. Взаимодействия между выделенными кубитами остаются незатронутыми. Зафиксируем некоторый невыделенный кубит и посчитаем его вклад в фазу. Он состоит из двух слагаемых: первое происходит из-за взаимодействия с выделенными, а второе - из-за взаимодействия с невыделенными кубитами. Посчитаем их последовательно. 1. Ввиду большой плотности λ Пуассоновского процесса $\mathcal{A}_p$ около половины времени pй кубит будет находиться в состоянии a'_p, а оставшуюся половину - в состоянии $1-a'_p$. Его взаимодействие с выделенным кубитом, скажем, с jм, даст вклад $\frac{1}{2}d_{p,j}a'_pa'_j+\frac{1}{2}d_{p,j}(1-a'_p)a'_j$ то есть $\frac{1}{2}d_{p,j}a'_j$. 2. Рассмотрим различные невыделенные кубиты с номерами $q\neq p$. Ввиду независимости моментов времени совершения NOT- операций над pм и qм кубитами и большой плотности λ, эти кубиты будут в каждом из состояний $(a'_p,\ a'_q)$, $(a'_p,\ 1-a'_q)$, $(1-a'_p,\ a'_q)$, $(1-a'_p,\ 1-a'_q)$ приблизительно четверть всего времени Результирующий вклад будет $\frac{1}{4}d_{p,q}[a'_pa'_q+a'_p(1-a'_q)+(1-a'_p)a'_q+(1-a'_p)(1-a'_q)]=\frac{1}{4}d_{p,q}$. Общий фазовый сдвиг, происходящий из-зи присутсвия невыделенных кубитов, находится теперь суммированием значений из пунктов 1 и 2 для всех $p\notin\{j,k\}$. Это будет

$$\frac{1}{2}[\sum_{p\notin\{j,k\}}d_{p,j}a'_j+\sum_{p\notin\{j,k\}}d_{p,k}a'_k]+\frac{1}{4}\sum_{p,q\notin\{j,k\}}d_{p,q}.$$

Этот сдвиг может быть скомпенсирован только однокубитными преобразованиями поскольку первые два слагаемых зависят только от значений кубитов, а другие вообще константы. Таким образом, мы получили схему с непрерывным двухкубитным взаимодействием, и однокубитными операциями, выполняющую фазовый сдвиг на $d_{j,k}a'_ja'_k$. Если взять временной промежуток Δt вместо единицы времени в этой процедуре, мы получим фазовый сдвиг на $\Delta t\ d_{j,k}a'_ja'_k$. Если мы хотим получить сдвиг на $-\Delta t\ d_{j,k}a'_ja'_k$, мы сначала должны применить NOT к jму кубиту, затем предыдущую процедуру, затем опять NOT к jму кубиту и наконец прибавить $-\Delta t\ d_{j,k}a'_k$ однокубитной операцией. Итак, мы можем сделать любой добавок к фазе вида $c\cdot a'_ja'_k$ для вещественного c независимо от его знака. Подходящая комбинация этих схем даст фазовый сдвиг

$$\sum_{j,k}c_{j,k}a'_ja'_k \tag{6.5}$$

для любого $c_{j,k}$. Размещая эти операции перед и после QFT^{-1} в процедуре из предыдущего пункта, мы компенсируем слагаемые A и B в фазе и получим схему, реализующую QFT^{-1}. Ошибки, возникающие в этой схеме, происходят из возможной некачественности Пуассоновских процессов, генерирующих моменты совершения операций NOT, а также из взаимодействия, происходящего в течении этих операций. Они могут быть минимизированы с увеличением плотности λ и уменьшением времени NOT операций по сравнению с обычным временем двухкубитных преобразований, определяемых взаимодействием $d_{j,k}$. Оценим замедление, вызванное вставками NOT-ов в большой плотностью по сравнению с абстрактной реализацией квантовых алгоритмов на схемах из функциональных элементах. Это надо так. Зафиксируем единицу времени так, что применение одной операции в схеме требует единичного времени. Пусть время разделено на равные короткие интервалы длины δt единиц, NOT-ы можно выполнять только в моменты вида $k\delta t$ для всякого целого k с вероятностью $p=1/\lambda$, где λ есть плотность процесса. Пусть время вычисления равно T, и $M=T/\delta t$. Ошибка в сдвиге фазы проистекающая из некачественности этой модели случайного процесса, будет $\delta t\ D$ где D есть дисперсия суммы случайных веоичин, принимающих значения 1

и 0 с вероятностями p и $1-p$ что есть $O(\sqrt{M})$. Значит, результирующая ошибка будет порядка $T/\sqrt{M}$ и должна быть пренебрежимо малой. Для QFT мы имеем $T = O(\log N)$ и мы получаем что $M = O(\frac{\log^2 N}{\epsilon})$ будет достаточно для исчезающе малой ϵ. Итак, видно, что метод случайных процессов дает немного более квадратичного замедления по сравнению со стандартной моделью, что вполне допустимо в случае таких быстрых алгоритмов, как QFT. Теперь мы докажем универсальность предложенной модели квантовых вычислений. Предположим, что взаимодействие между кубитами зависит только от их пространственного расположения, которое мы будем считать фиксированным. Единственное условие, налагаемое нами на взаимодействие заключается в том, что оно должно быть диагональным. Таким образом если j и k обозначают номера двух кубитов, то Гамильтониан их взаимодействия имеет один из видов

$$\text{A) } H_{j,k} = \begin{pmatrix} E_1^{j,k} & 0 & 0 & 0 \\ 0 & E_2^{j,k} & 0 & 0 \\ 0 & 0 & E_3^{j,k} & 0 \\ 0 & 0 & 0 & E_4^{j,k} \end{pmatrix}, \qquad \text{B) } H_{j,k} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & E_{j,k} \end{pmatrix}, \ E_{j,k} > 0. \quad (6.6)$$

Сначала заметим, что всякое взаимодействие общего вида (6.1, A) может быть сведено к виду (6.1, B) прибавлением подходящих однокубитных Гамильтонианов $H'_{j,k}$, матрицы которых имеют виды

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & b & 0 \\ 0 & 0 & 0 & b \end{pmatrix}, \qquad \begin{pmatrix} \alpha & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \alpha & 0 \\ 0 & 0 & 0 & \beta \end{pmatrix}.$$

Эта добавка приводит Гамильтониан вида (6.6, A) к (6.6, B) и может быть реализована быстрыми однокубитными преобразованиями, поскольку все диагональные матрицы коммутируют. Заметим, что разные пары кубитов могут взаимодействовать по разному, могут быть расположены на различных расстояниях, не обязательно на прямой и т.п. Чтобы доказать универсальность вычислительной модели с непрерывным взаимодействием, мы должны показать, как выполнить произвольную двухкубитовую операцию. Пусть дана униратная операция, индуцированная Гамильтонианом (6.6, B) в течение единичного времени: $U_{j,k} = \exp(-iH_{j,k})$ (Постоянную Планка мы, как всегда, считаем равной 1). Покажем как выполнить эту операцию над двумя кубитами: jм и kм, сохраняя все другие нетронутыми. Именно это последнее условие трудно обеспечить в случае непрерывного взаимодействия. Если мы сможем это сделать, то мы будем в состоянии реализовать любую двухкуюитную операцию над любой парой кубитов. Тогда для дальнего взаимодействия мы будем иметь, самое большее, линейное замедление по сравнению со стандартной моделью квантовых вычислений, а для короткого взаимодействия нам нужно будет выполнять операции SWAP, чтобы свести требуемую пару кубитов вместе, и таким образом, мы получим ко времени вычисления множитель, пропорциональный размеру памяти. Для осуществления преобразования $U_{j,k}$ надо применить описанный в предыдущем пункте метод преобразований NOT над невыделенными кубиами в моменты времени, которые генерируются независимыми Пуассоновскими случайными процессами большой плотности. Однако теперь выгода от такого метода будет не столь очевидна, как в случае QFT, поскольку, например, квантовый перебор требует все таки не логарифмического времени а квадратного корня из времени классического вычисления. Для таких случаев можно применить следующую модификацию нашего приема.

Метод периодических NOTов

Мы будем совершать преобразования NOTs на каждом из jх кубитов в моменты времени вида $jk\delta t$ для целых k, где δt есть снова малый временной промежуток. Тогда мы можем повторить конструкцию, описанную выше и избавиться от нежелательных фазовых сдвигов с помощью надлежащего выбора δt. Этот метод даст замедление в виде сомножителя порядка n^2 по сравнению со сложностью абстрактной модели квантовых функциональных схем. Теперь осталось показать, как с помощью преобразований $U_{j,k}$ можно совершить любое двухкубитовое преобразование. Например, продемонстрируем, как реализуется операция CNOT над данной парой кубитов. Пусть j, k фиксированы, и будем опускать индексы. Обозначим $\Delta E = E_1 - E_2 - E_3 + E_4$. Если $\frac{\Delta E}{\pi} \notin Q$ ($\frac{\Delta E}{\pi}$ не рационально), то (так как физические параметры нашей системы, влияющие на фазы, например, периоды циклов, могут быть немного изменены, чтобы избежать рациональности этого параметра, мы можем предполагать, что он иррационален без ограничения общности) мы можем осуществить операцию CNOT

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

над выбранной парой близлежащих кубитов, используя только однокубитовые преобразования и фиксированную двухкубитовую диагональную операцию E

$$E = \begin{pmatrix} \exp(iE_1) & 0 & 0 & 0 \\ 0 & \exp(iE_2) & 0 & 0 \\ 0 & 0 & \exp(iE_3) & 0 \\ 0 & 0 & 0 & \exp(iE_4) \end{pmatrix}$$

следующим образом. I. Обозначим последовательность ротаций фазы первого кубита через

$$A = \begin{pmatrix} 1 & 0 \\ 0 & \exp(i(E_1 - E_3)) \end{pmatrix},$$

второго кубита через

$$B = \begin{pmatrix} \exp(-iE_1) & 0 \\ 0 & \exp(-iE_2) \end{pmatrix},$$

и операцию E через U

$$U = E(A \bigotimes B) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \exp(i\Delta E) \end{pmatrix}.$$

II. Используя иррациональность $\frac{\Delta E}{\pi}$ можно показать, что

$$\forall \varepsilon > 0 \exists m \in N \exists n \in N : |\Delta E n - \pi(2m+1)| < \varepsilon,$$

т.е. прилюбой выбранной точности ε существует $n = n(\varepsilon)$ такой что U^n приближает преобразование П

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

с заданной точностью. III. Используя соотношение

$$(I \bigotimes H)\Pi(I \bigotimes H) = CNOT,$$

где I есть тождественная матрица

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

и H - операция Адамара

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

или, в матричной форме,

$$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

мывидим, что CNOT получается как последовательность

$$(I \bigotimes H) \left(E\,(A \bigotimes B) \right)^n (I \bigotimes H)$$

однокубитных ротаций и операции E.

6.1.2 Формализм чисел заполнения

Рассмотрим систему, состоящую из n одинаковых частиц. Сначала сделаем не вполне физичное предположение о том, что можем их надежно различать. Тогда состояние этой системы принадлежит Гильбертову пространству с базисом $\psi(r_1, r_2, \ldots, r_n) = \psi_{j_1}(r_1)\psi_{j_2}(r_2)\ldots\psi_{j_n}(r_n)$ где $\{\psi_j\}$ суть некоторый базис для состояний одной частицы, j_s принадлежит некому фиксированному множеству индексов $1, 2, \ldots, J$, так что r_j включает как пространственные, так и так называемые спиновые координаты[1] Выбор базиса просто означает, что система после измерения может быть найдена только в каком-либо из базисных состояний.

Однако в реальной системе из идентичных частиц они не могут быть различены. Следовательно, всякое базисное состояние должно содержать все слагаемые вида $\psi_{j_1}(r_1)\psi_{j_2}(r_2))\ldots\psi_{j_n}(r_n)$ с некоторыми коэффициентами. Теперь нам понадобится некая информация о природе рассматриваемых частиц. Они могут быть либо фермионами (как электроны или протоны), либо бозонами (как фотоны). Разница между этими типами частиц состоит в том, что максимальное значение спина фермионов - полуцелое (1/2, 3/2 и т.д.) а у бозонов - целое (0,1,2, и т.д.). Для нас же существенно то, что волновая функция системы фермионов должна менять свой знак при перестановке любых двух частиц, а у системы бозонов - сохранять его. Тогда очень естественно было бы для системы n фермионов представлять базисное состояние в виде детерминанта вида

$$\Psi = \frac{1}{\sqrt{n!}} \begin{vmatrix} \psi_{j_1}(r_1) & \psi_{j_1}(r_2) & \ldots & \psi_{j_1}(r_n) \\ \vdots & \vdots & \vdots & \vdots \\ \psi_{j_n}(r_1) & \psi_{j_n}(r_2) & \ldots & \psi_{j_n}(r_n) \end{vmatrix}, \tag{6.7}$$

[1] Спин это внутренний момент количества движения частицы, который имеет релятивистскую природу и может принимать разные значения в зависимости от типа рассматриваемых частиц. Например, для электронов проекция спина на выбранную ось после измерения принимает только два возможных значения.

а для системы бозонов - в виде перманента матрицы аналогичного вида[2]. Такое состояние можно рассматривать как ситуацию, при которой только состояния ψ_{j_s} for $s = 1, 2, \ldots, n$ заняты частицами из нашей системы, а остальные ψ_k for $k \in \{1, 2, \ldots, J\}$, не имеющие вида j_s - свободны. Если ψ с индексами обозначает собственный вектор одночастичного Гамильтониана, мы говорим о занятых или свободных энергетических уровнях, но, вообще говоря, ψ_k могут образовывать произвольный ортонормированный базис в пространстве состояний одной частицы. Состояние вида (6.7) может быть представлено как символ $|\bar{n}_\Psi\rangle = |n_1, n_2, \ldots, n_J\rangle$ где n_k есть единица если kый энергетический уровень занят, и нуль, если этот уровень свободен. Это - естественное представление состояния фермионовского ансамбля в терминах чисел заполнения. Такие векторы $\bar{n}$ составляют базис Фоковского пространства и общая форма состояния нашей системы будет иметь вид $\sum_{\bar{n}} \lambda_{\bar{n}} |\bar{n}\rangle$ с амплитудами λ.

Здесь уместно сделать одно замечание относительно обозначений. В литературе по приложениям квантовой механики, например к молекулярным расчетам, часто встречается немного иное описание детерминанта Фока-Слэтера (6.7) для системы тождественных фермионов. А именно, скажем, при значении спина $1/2$ этот детерминант представляют в виде

$$\Psi = \frac{1}{\sqrt{(2n)!}} \begin{vmatrix} \psi_{j_1}(r_1)\alpha_1 & \psi_{j_1}(r_2)\alpha_1 & \ldots & \psi_{j_1}(r_{2n})\alpha_1 \\ \psi_{j_1}(r_1)\beta_1 & \psi_{j_1}(r_2)\beta_1 & \ldots & \psi_{j_1}(r_{2n})\beta_1 \\ \vdots & \vdots & \vdots & \vdots \\ \psi_{j_n}(r_1)\beta_n & \psi_{j_n}(r_2)\beta_n & \ldots & \psi_{j_n}(r_n)\beta_n \end{vmatrix}, \tag{6.8}$$

где α_j и β_j следует понимать как символы, означающие соответственно: jя частица имеет спин $1/2$ и jя частица имеет спин $-1/2$. При этом в j_k уже не будет содержаться спиновых координат. Амплитуды непосредственно не имеют физического смысла, так что наличие таких символов в волновой функции не должно смущать читателя. Надо только иметь в виду, что при вычислении наблюдаемых величин с помощью Ψ- функции, например выражений вида $\int \langle\Psi| \; H \; |\Psi\rangle d\bar{r}$ данные символы должны подчиняться таким правилам: $\alpha_j^* \alpha_j = \beta_j^* \beta_j = 1$, $[\alpha_j \alpha_k] = [\beta_j \beta_k] = [\alpha_j, \beta_k] = 0$ при $j \neq k$, и $\alpha_j \alpha_j = \beta_j \beta_j = 0$ при любых j, которые выражают нормированность и принцип запрета Паули. При таком понимании вычисления в обозначениях (6.7) и (6.8) будут эквивалентными.

Оператор уничтожения a_j частицы в состоянии jго уровня и его сопряженный оператор a_j^+ (создание) определяются как $a_j|n_1, \ldots, n_J\rangle = \delta_{1,n_j}(-1)^{\sigma_j}|n_1, \ldots, n_{j-1}, n_j - 1, n_{j+1}, \ldots, n_J\rangle$ где $\sigma_j = n_1 + \ldots + n_j$. Они обладают известными коммутационными соотношениями: $a_j^+ a_k + a_k a_j^+ = \delta_{j,k}$, $a_j a_k + a_k a_j = a_j^+ a_k^+ + a_k^+ a_j^+ = 0$.

Предположим, что всякое взаимодействие в Природе затрагивает не более двух частиц. Тогда всякое взаимодействие в многочастичном ансамбле может быть разложено в сумму одно - и двух - частичных взаимодействий вида $H = H_{one} + H_{two}$ с соответствующими потенциалами $V_1(r)$ и $V_2(r, r')$. Каждый из нихможет быть представлен операторами рождения и уничтожения в виде $H_{one} = \sum_{k,l} H_{k,l} a_k^+ a_l$, $H_{two} = \sum_{k,l,m,n} H_{k,l,m,n} a_l^+ a_k^+ a_m a_n$ где

$$H_{k,l} = \langle\psi_k| \; H_{one} \; |\psi_l\rangle = \int \psi_k^*(r) V_1(r) \psi_l(r) dr,$$
$$H_{k,l,m,n} = \langle\psi_l, \psi_k \; |H_{two}| \; \psi_m \psi_n\rangle = \int \psi_k^*(r) \psi_l^*(r') V_2(r, r') \psi_m(r) \psi_n(r') dr dr'.$$

Значит, если даны потенциалы всех взаимодействий и все базисные состояния ψ_i, то мы можем в принципе найти их представление в терминах операторов рождения и уничтожения, то есть на языке чисел заполнения.

[2]Перманент матрицы отличается от ее детерминанта только тем, что при его вычислении нет никаких вычитаний - одни сложения, так что он не меняет величины при перестановке строк или столбцов матрицы.

Рассмотрим ансамбль с Гамильтонианом вида $H = \sum_i H^i_{ext.f.} + \sum_{i,j}(H^{i,j}_{diag.} + H^{i,j}_{tun.})$, где Гамильонианы внешних полей, диагонального взаимодействия и туннелированияпредставляются в терминах операторов рождения и унитожения как

$$\begin{array}{ll} H^i_{ext.f.} & = \alpha_i a_i^+ a_i, \quad \alpha_i \in \mathrm{R}, \\ H^{i,j}_{diag.} & = \beta_{i,j} a_i^+ a_i a_j^+ a_j, \quad \beta_{i,j} \in \mathrm{R}, \\ H^{i,j}_{tun.} & = \gamma_{i,j} a_i^+ a_j + \gamma^*_{i,j} a_j^+ a_i. \end{array} \tag{6.9}$$

Заметим что осуществить управление диагональным Гамильтонианом было бы непросто, поскольку такое взаимодействие должно было бы затрагивать две произвольные частицы рассматриваемого ансамбля, которые являются неразличимыми по принципу идентичности. Таким образом, логично считать, что такое взаимодействие постоянно, и не подвластно нашему контролю, в то время как мы можем эффективно управлять туннельным взаимодействием. При такой форме управления возможно реализовать любое квантовое вычисление. Этот тип контроля представляется более реалистичным, так как туннелированием можно в принципе управлять с помощью лазерных импульсов.

6.1.3 Вычисления, управляемые в помощью туннелирования

Для того, чтобы доказать универсальность предлагаемой упрощенной схемы управления фермионными вычислениями, мы должны сделать однотехничесое приготовление, а именно, установить несколько иное соотношение между Гильбертовским пространством кубитом и Фоковским пространством чисел заполнения, чем то естественное соответствие, о котором говорилось выше.

Зафиксируем некоторое разбиение всех энергетических уровней на две равные части и выберем некоторое взаимно-однозначное соответствие между этими частями. Для определенности мы можем рассматривать kй уровень вниз от уровня Ферми ϵ_F и условиться, что он соответствует kму уровню вверх от ϵ_F. Мы обозначим jй уровень внизот границы Ферми через обычную букву, а jй уровень вверх от этой границы через букву со штрихом j'. Назовем первый уровень jм нижним уровнем, а второй уровень - jм верхним уровнем. Фоковское пространство $\mathcal{F}$ можно представить как $\mathcal{F} = \mathcal{F}_1 \bigotimes \mathcal{F}_2 \bigotimes \ldots \bigotimes \mathcal{F}_k$ где каждый $\mathcal{F}_j$ соответствует jй паре соответсвующих энергетических ровней. Рассмотрим подпространство F_j в $\mathcal{F}_j$, которое порождено двумя следующими векторами. Первый будет таким: "j'й уровень занят, а jй свободен", второй же будет таким: "jй уовень занят, а j'й свободен". Обозначаем их через $|1\rangle_j$ и $|0\rangle_j$ соответственно. Мы будем работать с подпространством $F = F_1 \bigotimes F_2 \bigotimes \ldots \bigotimes F_k$в Фоковском пространстве $\mathcal{F}$. Определим функцию θ, которая отображает наше Гильбертово пространство $\mathcal{H}$ в F с помощью следующего определения действия на базисных состояниях: $\theta(|\xi_1, \xi_2 \ldots \xi_n\rangle) = |\xi_1\rangle_1 \bigotimes |\xi_2\rangle_2 \bigotimes \ldots \bigotimes |\xi_n\rangle_n$ где все ξ_j есть нули и единицы. Тогда θ устанавливает нестандартное сооветствие между Гильбертовским и Фоковским пространствами (см. рисунок 2).

Однокубитовое состояние в Гильбертовом пространстве соответствует двухкубитовому состоянию в обычном отождествлении с кубитами (один уровень - один кубит). Мы увидим, что такое отождествление лучше соответствует нашим целям, чем естественное. Итак, теперь все готово для представления унитарных преобразований в Гильбертовом пространстве в терминах операторов, действующих в пространстве чисел заполнения. Рассмотрим эрмитов оператор H в двумерном Гильбертовом пространстве состояний одного кубита $\mathcal{H}$. Он имеет вид $H_0 + H_1$, где

$$H_0 = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}, H_1 = \begin{pmatrix} 0 & d \\ \bar{d} & 0 \end{pmatrix}.$$

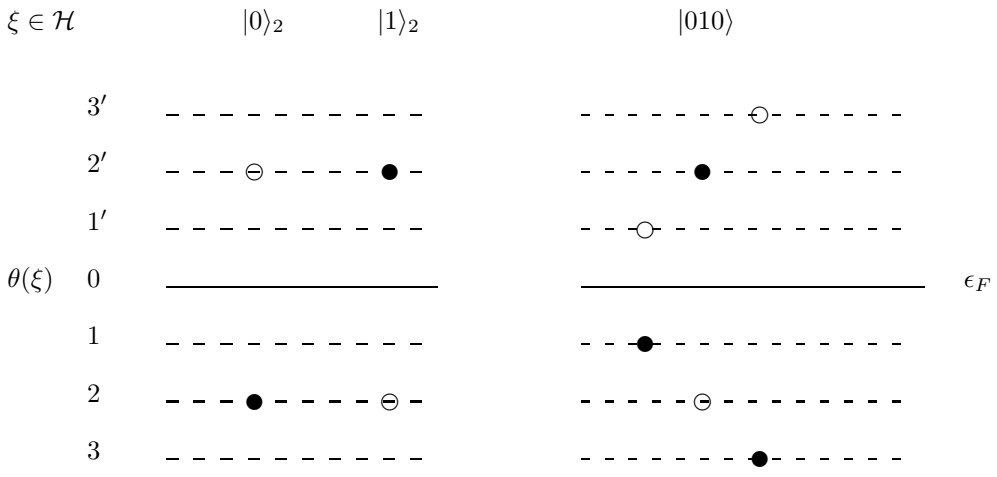

Рисунок 2. Соответствие между Фоковским и Гильбертовым пространствами

Непосредственно проверяется, что для операторов $\tilde{H}_0 = d_1 a_k^+ a_k + d_2 a_{k'}^+ a_{k'}$ и $\tilde{H}_1 = d a_k^+ a_{k'} + \bar{d} a_{k'}^+ a_k$ (внешнего поля и туннелирования) выполняются равенства $\tilde{H}_i \theta = \theta H_i$ для $i = 0, 1$. Используя линейность θ, мы находим $(\tilde{H}_0 + \tilde{H}_1)\theta = \theta H$. Теперь рассмотрим однокубитовое унитарное преобразование U в Гильбертовом пространстве. Оно имеет форму e^{-iH} для Гамильтониана H (мы выбрали подходящую единицу времени для того, чтобы избавиться от постоянной Планка и времени). В силу линейности θ и равенства $\theta^{-1} H^s \theta = (\theta^{-1} H \theta)^s$ для натуральных s мы находим, что для вского однокубитного унитарного оператора U можно эффективно найти соответствующий Гамильтониан в Фоковском пространстве, содержащий только внешнее поле и туннелирование, который делает диаграмму A из рисунка 2 замкнутой.

Займемся двухкубитными преобразованиями в Гильбертовом пространстве. Поскольку все диагональные матрицы коммутируют, для всех диагональных преобразований в пространстве $\mathcal{F}_k \bigotimes \mathcal{F}_j$ мы можем эффективно найти соответствующий диагональный оператор в Гильбертовом пространстве, который делает диаграмму B из рисунка 3 замкнутой.

Теперь все готово для переноса приема из работы [OF] с однокубитным управлением на Фоковское пространство. Комбинация диаграмм из рисунков 3 даст диаграмму из рисунка 4.

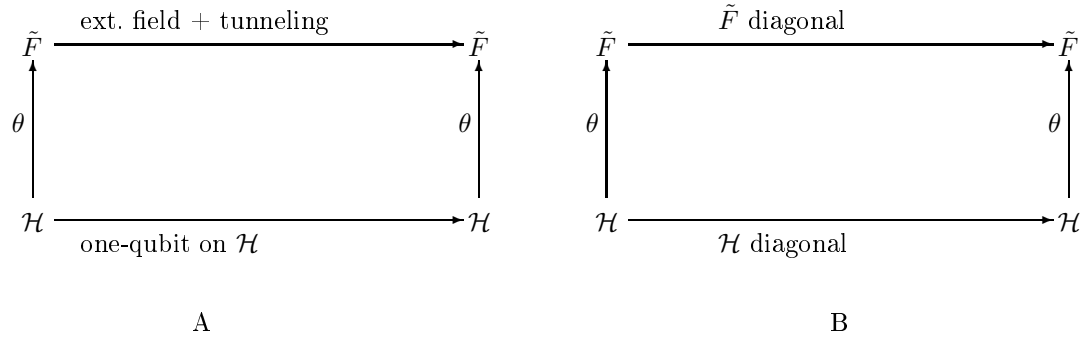

Рисунок 3. Соответсвие операторов в
Фоковском и Гильбертовом подпространствах. $\tilde{F} = F_j \bigotimes F_k$.

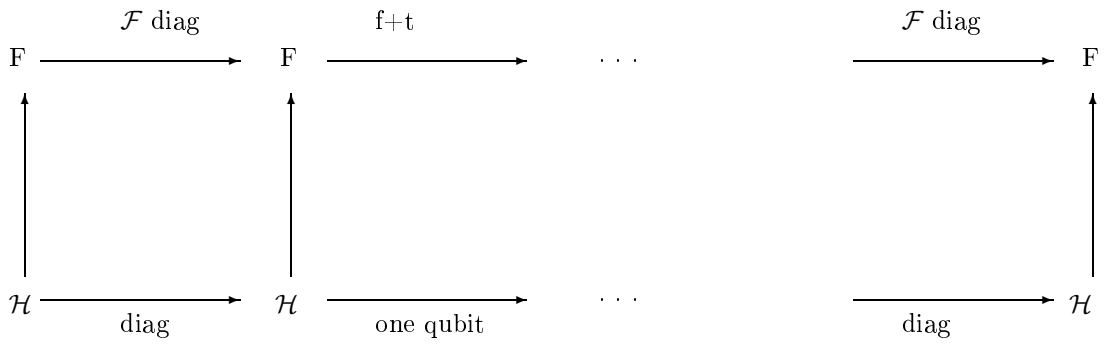

Рисунок 4. Соответствие вычислений в
Фоковском и Гильбертовом пространствах

Пусть диагональная часть Гамильтониана взаимодействия в Фоковском пространстве фиксирована и действует непрерывно в неуправляемом режиме. Тогда мы можем подыскать соответству-

ющее диагональное взаимодействие в Гильбертовом пространстве, делая замкнутыми все "диагональные"части диаграммы из рисунка 4. В силу результата [OF] мы можем выбрать однокубитные преобразования, реализующие управление произвольным квантовым алгоритмом в Гильбертовом пространстве, в виде нижней последовательности преобразований в диаграмме. Наконец, мы можем найти управление вида field + tunneling над состояниями Фоковского пространства, делающие замкнутой всю диаграмму. Заметим, что все операторы рождений и уничтожений, рассматриваемые в целом Фоковском пространстве нелокальны благодаря множителю $(-1)^{\sigma_j}$, который зависит от заданного состояния. Для диагонального оператора $a_j^+ a_j a_k^+ a_k$ и внешнего поля такие множители компенсируются. Туннельный оператор $a_j^+ a_{j'}$ в пространстве F приносит множитель $(-1)^{\sigma'}$ где $\sigma' = \sum_{s+j}^{j'-1} n_s = j' - j$ который не зависит от данного состояния $|\bar{n}\rangle \in F$, поскольку для такого состояния ровно половина уровней между j и j' заняты фермионами. Так как знак может быть вынесен из всех состояний и проигнорирован.

Так мы получим универсальный квантовый компьютер на состояниях в пространстве чисел заполнения, управляемый только внешним полем и туннелированием.

6.2 Коррекция ошибок в квантовых вычислениях

Своеобразие квантовых состояний проявляется в том, что они очень чувствительны к внешним воздействиям. Это объясняется их сверхмалыми размерами. Например, невозможно полностью изолировать группу атомов в кристаллической решетке от теплового воздействия со стороны других атомов. Мы видели, что есть два принципиально разных типа таких воздействий: внешние поля, включаемые в уравнение Шредингера, и измерения системы. При этом если воздействие внешнего поля вызывает унитарное, то есть обратимое возмущение состояния системы, то измерение всегда приводит к необратимому изменению этого состояния. Таким образом, в рамках формализма гильбертовых пространств невозможно единым образом эффективно описывать два этих типа внешних воздействий на изучаемую систему[3]. Таким образом, ошибки в квантовых состояниях бывают двух типов: обратимые, которые могут происходить и в классических состояниях, и необратимые, специфически квантовые. Однако, такая классификация ошибок не точно отражает суть дела. Рассмотрим, для примера состояние $\frac{1}{\sqrt{2}}|000\ldots0\rangle + \frac{1}{\sqrt{2}}|111\ldots1\rangle$. Ошибка в первом кубите может иметь, например, форму: $|0\rangle \longrightarrow |1\rangle$, $|1\rangle \longrightarrow |0\rangle$ или $|0\rangle \longrightarrow |0\rangle$, $|1\rangle \longrightarrow |0\rangle$. Первая будет унитарным преобразованием, а вторая - нет. Наше состояние тогда приобретет форму $\xi' = \frac{1}{\sqrt{2}}|100\ldots0\rangle + \frac{1}{\sqrt{2}}|011\ldots1\rangle$ или $\xi' = \frac{1}{\sqrt{2}}|000\ldots0\rangle + \frac{1}{\sqrt{2}}|011\ldots1\rangle$ соответственно. Видно, что фактически первоначальное и "испорченное"состояния системы очень близки по виду, и обе ошибки легко скорректировать унитарным преобразованием (во втором случае нужно, конечно, использовать кубиты, не затронутые внешним воздействием). Это происходит из-за того, что обе эти ошибки по существу затрагивают содержимое только одного кубита. Рассмотрим теперь измерение одного, скажем, первого кубита. Пусть система находилась в состоянии $\lambda|0,0,0\rangle + \mu|1,1,1\rangle$. Тогда в результате мы получим одно из состояний $|0,0,0\rangle$ или $|1,1,1\rangle$, потеряв таким образом всю информацию

[3] Можно конечно описать эти воздействия формально единообразным образом, например в виде так называемых супероператоров - унитарных операторов, действующих в расширенном гильбертовом пространстве. Но от такого описания будет не много пользы, потому что в нем не будет даже той ограниченной детерминистичности, который присутствует в стандартном квантовом формализме, как возможность в принципе точно вычислить волновую функцию изучаемой системы в любое заданное время.

о первоначальных амплитудах. Мы видим, что если кубиты находятся в запутанном состоянии, то измерение одного из них неизбежно порождает ошибку во всех других ! Таким образом, более правильным было бы классифицировать все возможные ошибки на локальные, т.е. затрагивающие содержимое только одного кубита, и глобальные, затрагивающие неограниченное число кубитов. При этом в случае запутанного состояния локальные воздействия (измерение) способны привести, как мы только что видели, к глобальным ошибкам. В ходе вычислений естественно будут возникать оба эти типа ошибок, и нам придется как то с ними бороться. Ясно, что корректировать локальные ошибки должно быть намного проще, чем глобальные. Можно также предположить, что для коррекции глобальных ошибок, возникающих при локальных воздействиях на систему, необходимо будет использовать чисто квантовый прием, поскольку этот тип не встречается в мире классических систем. И, если такой прием существует, его придется использовать многократно в ходе вычислений, поскольку компьютер всегда испытывает воздействие внешней среды, приводящее к возникновению таких ошибок. Таким образом, программа коррекции ошибок в квантовом компьютере должна запускаться в фоновом режиме и постоянно коректировать все возникающие ошибки. Естественно, воздействие внешней среды должно быть локальным и его интенсивность должна достаточно низкой, для того чтобы наша программа коррекции успела исправить все ошибки. Оказывается, такой метод коррекции квантовых ошибок существует. В этом разделе мы приведем упрощенную схему такого метода.

Сначала рассмотрим принцип коррекции классических ошибок. Этот принцип по существу один - избыточное кодирование. Состояние одного бита кодируется m битами, в которых мы просто дублируем содержимое исходного бита. Теперь если в результате ошибок поменялось на противоположное содержимое $\left[\frac{m-1}{2}\right]$ кодирующих битов, мы можем легко восстановить исходное значение, если возьмем то значение в кодирующих битах, которое принимается большинством. Посмотрим, можно ли применить этот прием для коррекции квантовых состояний. Для этого, во первых, нужно уметь "размножать"квантовые состояния так же легко, как мы можем копировать состояние классического бита. Здесь нас ожидает первая ловушка. Оказывается, не существует физического процесса, копирующего квантовые состояния ! Этот факт известен под громким названием "no cloning theorem", однако он очень простой, и мы его сейчас установим. Действительно, мы хорошо знаем, что все физические процессы описываются или унитарными преобразованиями, или носят вероятностный характер. Предположим, что существует физический процесс W, который клонирует квантовые состояния. Чтобы от него была хоть какая-то польза, он должен действовать на все состояния в гильбертовом пространстве, то есть иметь вид функции $W: H \bigotimes H \longrightarrow H \bigotimes H$, такой что $W(|\Psi, 0\rangle = \lambda|\Psi, \Psi\rangle$ для любого состояния $|\Psi\rangle$ где константа C зависит от Ψ. Эта функция не должна быть вероятностной так как мы рассматриваем чистые состояния. Значит W должен быть унитарным оператором. Но тогда у нас в силу линейности W получилось бы $\lambda|\Psi, \Psi\rangle + \mu|\Psi_1, \Psi_1\rangle = W|\Psi, 0\rangle + W|\Psi_1, 0\rangle = W|\Psi + \Psi_1, 0\rangle = \nu|\Psi + \Psi_1,\ \Psi + \Psi_1\rangle = \nu(|\Psi, \Psi\rangle + |\Psi_1, \Psi_1\rangle + |\Psi, \Psi_1\rangle + |\Psi_1, \Psi\rangle)$, для любой пары состояний, что, разумеется неверно. Итак, клонирование квантовых состояний невозможно.

Но тогда, быть может, мы могли бы использовать естественное "псевдоклонирование"с помощью CNOT и анциллы: $CNOT\ (\lambda|0\rangle + \mu|1\rangle) \bigotimes 0\rangle = \lambda|0, 0\rangle + \mu|1, 1\rangle$? При этом, конечно, мы не будем иметь в результате клонированного состояния исходного кубита: $(\lambda|0\rangle + \mu|1\rangle) \bigotimes (\lambda|0\rangle + \mu|1\rangle)$ (убедитесь в этом сами используя дистрибутивность тензорного произведения). Однако в каком-то смысле наш исходный кубит окажется закодированным в два новых. Далее, можно повторить эту процедуру со следующим вспомогательным кубитом, получив состояние $\lambda|0, 0, 0\rangle + \mu|1, 1, 1\rangle$. Такая процедура в случае классического состояния кубита: 0 или 1 - действительно приводит к клонированию. Однако пользы от этой кодировки в случае глобальных ошибок будет немного, как следует из

рассмотренного выше случая измерения одного кубита. Этот пример также иллюстрирует запрет на клонирование квантовых состояний, так как при попытке "клонировать"такое состояние новые "копии"окажутся в запутанном состоянии с "оригиналами"и наблюдение "оригиналов"неизбежно разрушит "копии". Для коррекции такого воздействия надо применять немного более сложный прием, который мы здесь опишем далее для простейшей ситуации.

Фактически два выделенных типа ошибок - локальные и наблюдения единичных кубитов, исчерпывают все возможные типы ошибок вообще. Рассмотрим, например, нарушение фазы $|0\rangle \longrightarrow |0\rangle$, $|1\rangle \longrightarrow -|1\rangle$ в первом кубите. Тогда преобразование Адамара приведет к тому же самому, что и инверсия содержимого этого кубита. Если мы скорректировали ошибку в содержимом этого кубита, то следующее применение преобразования Адамара восстановит нам прежнее состояние. Рассмотрим теперь такое воздействие окружения, которое можно представить в виде $U: \alpha \bigotimes \xi \longrightarrow \chi$ где α - состояние окружения, и U действует на α и на первый кубит в нашей системе. На первый взгляд этот тип ошибок более общий. Однако его можно свести к комбинации измерения одного кубита и локальной ошибки.

Рассмотрим еще раз схему коррекции локальных ошибок. Мы закодируем наше квантовое состояние с использованием некоторого числа анцилл, и устроим специальную анциллу, играющую роль "мусорного ящика". После того, как произошла ошибка, мы с помощью стандартных унитарных преобразований перемещаем содержимое затронутых ошибкой кубитов в этот "мусорный ящик", так что первоначальное состояние испорченных кубитов восстанавливается, и только "мусорный ящик"хранит информацию о том, какая ошибка произошла.

Эта схема будет работать и в случае глобальных ошибок. Пусть интенсивность локальных воздействий окружения на нашу квантовую систему ограничена. Мы переведем исходное состояние в соответствующее ему закодированное состояние системы с анциллами и можем использовать процедуру коррекции снова и снова через короткий интервал времени Δ так что вероятность того, что в течение этого времени окружение будет воздействовать более чем на один кубит, будет исчезающе малой. Так, варьируя Δ, мы можем сколь угодно долго поддерживать нашу систему в первоначально закодированном состоянии. Конечно, процедура коррекции должна зависеть от метода кодировки, и мы сначала опишем этот метод в применении к состоянию одного кубита ξ. В случае нескольких кубитов мы можем просто закодировать каждый кубит по отдельности и выполнить нашу процедуру коррекции над всеми кодирующими ансамблями одновременно и независимо. Наша процедура коррекции восстановит закодированное состояние и сохранит запутанность, так что ее одновременное выполнение для всех кодирующих ансамблей сохранит закодированный вид первоначального ансамбля.

Итак, сконцентрируемся на методе кодировки состояния одного кубита $\xi = \alpha|0\rangle + \beta|1\rangle$. Этот кубит будет кодироваться тремя кубитами так, что если один из них будет измерен (мы не знаем какой), то наша процедура, проведенная над результатом измерения, восстановит исходное закодированное состояние. Кодирование будет линейной операцией, так что достаточно определить его для базисных состояний $|0\rangle$ и $|1\rangle$. Их коды будут, соответственно, таковы:

$$\begin{aligned} \tilde{0} = &\quad \tfrac{1}{2\sqrt{2}}(|000\rangle + |100\rangle + |010\rangle + |001\rangle + |110\rangle + |101\rangle + |011\rangle + |111\rangle) \\ \tilde{1} = &\quad \tfrac{1}{2\sqrt{2}}(|000\rangle - |100\rangle - |010\rangle - |001\rangle + |110\rangle + |101\rangle + |011\rangle - |111\rangle). \end{aligned} \tag{6.10}$$

Таким образом, код состояния ξ будет трехкубитным состоянием вида $\tilde{\xi} = \alpha\tilde{0}+\beta\tilde{1}$. Теперь посмотрим, что произойдет, если мы понаблюдаем один из кубитов. У нас есть три кубита, два состояния, кодирующих базисные однокубитные состояния: $\tilde{0}$ and $\tilde{1}$ и два различных результата наблюдения: 0 или 1. То есть, существует 12 различных результатов измерения наших двух кодирующих состояний,

если произошло измерение одного кубита, плюс 2 неизмененных базисных состояния, если вообще не произошло никаких измерений. Обозначим результаты через $\tilde{i}^j_k,\ i=0,1,\ j=\emptyset,1,2,3,\ k=\emptyset,0,1$ что означает "состояние, полученное при измерении j-го кубита состояния $\tilde{i}$, если k - результат этого измерения". Если измерений вообще не было, то $j=k=\emptyset$ и $\tilde{i}^j_k=\tilde{i}$. Во всех случаях происхедшего измерения результирующее состояние будет иметь вид:

$$\begin{array}{ll}
\tilde{0}^1_0 & =\frac{1}{2}(|000\rangle+|010\rangle+|001\rangle+|011\rangle,\\
\tilde{0}^1_1 & =\frac{1}{2}(|100\rangle+|110\rangle+|101\rangle+|111\rangle,\\
\tilde{0}^2_0 & =\frac{1}{2}(|000\rangle+|100\rangle+|001\rangle+|101\rangle,\\
\tilde{0}^2_1 & =\frac{1}{2}(|010\rangle+|110\rangle+|011\rangle+|111\rangle,\\
\tilde{0}^3_0 & =\frac{1}{2}(|000\rangle+|010\rangle+|100\rangle+|110\rangle,\\
\tilde{0}^3_1 & =\frac{1}{2}(|001\rangle+|011\rangle+|001\rangle+|101\rangle,\\
\tilde{1}^1_0 & =\frac{1}{2}(|000\rangle-|010\rangle-|001\rangle+|011\rangle,\\
\tilde{1}^1_1 & =\frac{1}{2}(-|100\rangle+|110\rangle+|101\rangle-|111\rangle,\\
\tilde{1}^2_0 & =\frac{1}{2}(|000\rangle-|100\rangle-|001\rangle+|101\rangle,\\
\tilde{1}^2_1 & =\frac{1}{2}(-|010\rangle+|110\rangle+|011\rangle-|111\rangle,\\
\tilde{1}^3_0 & =\frac{1}{2}(|000\rangle-|010\rangle-|100\rangle+|110\rangle,\\
\tilde{1}^3_1 & =\frac{1}{2}(-|001\rangle+|011\rangle-|001\rangle+|101\rangle
\end{array} \tag{6.11}$$

Определим теперь действие корректирующего оператора на данных векторах как

$$U_{rest}:\ \tilde{i}^j_k\bigotimes\bar{0}\longrightarrow\tilde{i}\bigotimes\tilde{0}^j_k. \tag{6.12}$$

Убедимся, что так определенный корректирующий оператор сохраняет все углы между векторами i^j_k. Обозначим подпространства, порожденное всеми векторами видов $\tilde{0}^j_k$ и $\tilde{1}^j_k$ через O и I соответственно. Мы имеем: $\tilde{0}\perp\tilde{1}$, $O\perp I$, все другие углы будут равны $\frac{\pi}{3}$, поскольку $\langle\tilde{1}^1_0|\tilde{1}^2_0\rangle=\langle\tilde{1}^1_0|\tilde{1}^2_1\rangle=\frac{1}{2}$ и то же самое имеет место для базиса подпространства O. Мы видим, что все углы сохраняются. Значит, опреатор U_{rest} может быть продолжен до унитарного оператора на всем пространстве шести кубитов $C^8\bigotimes C^8$, который мы будем обозначать той же буквой. Отметим, что его действие было нами определено лишь для состояний с нулевыми анциллами. Это означает, что мы должны после каждого применения этого оператора U_{rest} позаботиться о восстановлении нулевого состояния анцилл, необходимого для правильного действия следующего коректирующего оператора. Это можно сделать, измеряя анциллы и затем помещая в них нуль (что не является унитарным оператором. Обозначим эту процедуру через A_{rest}. Тогда полный цикл "декогеренция - восстановление"принимает форму

$$\alpha\tilde{0}+\beta\tilde{1}\overset{\text{observation}}{\longrightarrow}\alpha\tilde{0}^j_k+\beta\tilde{1}^j_k\overset{U_{rest}}{\longrightarrow}(\alpha\tilde{0}+\beta\tilde{1})\bigotimes\tilde{0}^j_k\overset{A_{rest}}{\longrightarrow}\alpha\tilde{0}+\beta\tilde{1}, \tag{6.13}$$

где мы опускаем нулевые анциллы в записи тензорного произведения. Отметим, что наше унитарное корректирующее преобразование никак не зависит от того, какая произошла ошибка - его роль состоит только в том, чтобы переместить ошибку, если таковая имеется вообще, в "мусорную корзину", образованную анциллами, и очистить тем самым исходное закодированное состояние, которое никак не изменится при измерении и обновлении анцилл, поскольку оно выделяется в виде тензорного сомножителя. Пусть δ_{rest} обозначает время, требуемое для этой последовательности операторов, а Δ - интервал, с которым мы совершаем операции коррекции. Предположим, что интенсивность внешнего воздействия не слишком велика, так что вероятность измерения более одного кубита из наших трех за время Δ пренебрежимо мала по сравнению с вероятностью измерения ровно одного,

а анциллы мы постоянным измерением вообще поддерживаем в нулевом состоянии всегда кроме промежутков δ_{res}. Кроме того, пусть вероятность ошибки в любом кубите за время δ_{rest} также пренебрежимо мала. Тогда описанная итерация оператора коррекции с большой вероятностью будет поддерживать наше закодированное состояние одного кубита несмотря на декогерентность. Для сохранения многокубитных состояний достаточно применять эту закодировать все кубиты по отдельности и применять описанную процедуру коррекции к каждому кодирующему шестикубитному ансамблю по отдельности.

6.2.1 Замечания о кодах коррекции квантовых ошибок

Мы получили элементарную процедуру коррекции квантовых ошибок, состоящую из примения одного унитарного преобразования и следующих за ним измерения анцилл и переводе их в нулевое состояние. Вторая часть этой повторяющейся процедуры явно не унитарна. Конечно, мы можем представлять себе, что "испорченные"анциллы всякий раз просто выбрасываются, и вместо них мы используем новые. Можно ли корректировать ошибки с помощью только одних унитарных преобразований, т.е. не прибегая к измерениям и заменам содержимого анцилл? Предположим, ято это возможно, и пусть U будет соответствующим унитарным корректирующим оператором, заменяющим собой всю цепочку A_{rest}. Тогда он должен выполнять свою функцию с произвольными анциллами, так как теперь анциллы тоже подвержены декогерентности. Обозначим регистры, содержащие все кодирующие кубиты и все анциллы через a и b соответственно. Для начального состояния вида $a \bigotimes b$ с произвольными b и результатов a_1 или a_2 наблюдений с любыми b мы должны иметь $U\ a_i \bigotimes b = a \bigotimes b_i$ для некоторых состояний b_i, затронутых процессом декогеренции. Результаты a_1 и a_2 измерения ортогональны, поскольку b_1 и b_2 должны быть ортогональными в силу унитарности U. Пусть размерность пространства анцилл есть k. Выберем некоторый ортогональный базис $b^1, b^2, \dots b^k$ в пространстве анцилл и пусть $U\ a_i \bigotimes b^j = a \bigotimes b_i^j$ для каких то состояний b_i^j анцилл. Подпространства, порожденные векторами b_1^j и b_2^j должны быть ортогональными. С другой стороны, всякая пара $b_i^j,\ b_i^{j'}$ должна быть ортогональна для $i = 1, 2;\ j, j' = 1, 2, \dots k,\ j \neq j'$, поскольку U унитарно. Мы получаем $2k$ ортогональных ненулевых векторов b_i^j в k мерном пространстве, что и дает искомое противоречие.

Итак, чисто унитарная коррекция квантовых ошибок невозможна. Значит, данная нами элементарная схема квантовой коррекции не может быть радикально упрощена. В любом случае корректирующая процедура должна содержать необратимые элементы, такие как измерения или приток новых надлежаще инициализированных анцилл.

В заключение сделаем одно общее замечание о кодах коррекции квантовых ошибок. Коды коррекции квантовых ошибок к настоящему времени разрослись до солидной научной дисциплины, цель которой - дать эффективный метод борьбы со спонтанным разрушением квантовых состояний. Реальная коррекция ошибок при вычислениях типа GSA предполагает иерархическую структуру, так что каждый ярус корректирует те ошибки, которые возникают в нижележащем ярусе (см., например, ([AB]))[4]. Однако все такие конструкции основываются на принципиальной возможности реализации широкого класса квантовых состояний, в частности, состояний, включающих слагаемые со сколь угодно малыми но ненулевыми амплитудами. Кроме этого, предполагается, что

[4]Это в полной мере относится и к так называемым адиабатическим квантовым вычислениям, в которых элементарные преобразования явно не рассматриваются, но принципы коррекции будут теми же, а также и к коррекциям малых неточностей, возникающих при выполнении элементарных преобразований; в последнем случае годится общая схема коррекции ошибок, возникающих в отдельных кубитах - см., например, (**??**).

некоторые ключевые квантовые преобразования, например, такие как CNOT, можно совершить с большой точностью и без всяких кодов коррекции. Таким образом, несмотря на всю важность, коды коррекции квантовых ошибок не могут служить исчерпывающим утвердительным ответом на принципиальный вопрос о реализуемости полномасштабного квантового компьютера.

Глава 7

Почему квантовый компьютер не может считать слишком быстро

Каково общее соотношение между минимальным временем классического и квантового вычисления? Ситуация, при которой квантовое время существенно короче классического, называется квантовым ускорением. С другой стороны, мы видели, что любое классическое вычисление можно превратить в квантовое практически без всякого замедления. Это ставит принципиальный вопрос: является ли квантовое ускорение широко распространенным феноменом, или представляет собой редкое исключение? Иными словами, какова доля тех вычислительных задач, для которых возможно существенное (или, по крайней мере, какое-либо) квантовое ускорение? Оказывается, что в общем случае для подавляющего большинства возможных вычислительных задач квантовое время не может быть существенно меньше квадратного корня из классического. Если же принять во внимание еще и размер памяти вычислительного устройства, то можно разделить относительно короткие и относительно длинные вычисления. Тогда можно даказать и более тонкое утверждение. Если время классического вычисления не превышает корня седьмой степени от числа всевозможных состояний памяти компьютера, то подавляющее большинство таких вычислений не может быть ускорено на квантовом компьютере даже на один шаг! Эти две нижние границы времени квантовых вычислений показывают два обстоятельства. Во первых, от применения квантовых вычислений можно ожидать эффекта, в основном, только для трудных задач, то есть таких задач, которые при решении вовлекают большую часть всевозможных состояний компьютера. Во вторых, каждый случай квантового ускорения представляет своеобразное и редкое произведение искусства. Так что полезно было бы извлекать наибольшую пользу из известных квантовых "трюков". Что, конечно, не означает бесполезности поиска новых случаев квантового ускорения.

В этой главе мы установим нижние оценки квантовой сложности решения для двух типов задач: перебор и итерационные алгоритмы. Для получения нижних оценок в этих задачах использованы методы, идея которых восходит к статье [BBBV], и которая была также применена в работе [BBHT]. Примем следующие обозначения для квантового компьютинга. Каждое состояние квантового компьютера с n кубитами является точкой $\chi = \sum_j \lambda_j e_j, \quad \|\chi\| = 1$ в 2^n мерном гильбертовом пространстве с ортонормированным базисом $\{e_j\}$, где λ_j есть комплексные числа, называемые амплитудами. Вычисление имеет вид $\chi_0 \longrightarrow \chi_1 \longrightarrow \ldots \longrightarrow \chi_t$ где всякий переход $\chi_i \longrightarrow \chi_{i+1}$ есть унитарное преобразование, зависящее от оракула.

7.1 Эффект изменения оракула для квантового вычисления

Для установления нижних оценок для поиска точки экстремума нам нужны несколько технических понятий и свойств, качающихся влияния изменения оракула на результат квантовых вычислений. Здесь мы обобщим ряд результатов из работы [Oz], которые и будут применяться в следующем параграфе.

Мы обозначаем базисное состояние буквой e с индексами. Пусть результат действия оракула на базисном состоянии $e = |\ldots, a, b, \ldots\rangle$ есть состояние $|\ldots, a, \phi(a) + b, \ldots\rangle$ где a и b есть места для вопроса и ответа соответственно, и $+$ означает побитовое сложение по модулю 2. Это - унитарное преобразование, обозначаемое через Qu_ϕ. Обозначим это слово a через $q(e)$.

Вопросное состояние χ спрашивает оракул (или вызывает его) на всех словах $q(e)$ с некоторыми (вообще говоря, разными) амплитудами. Положим $\mathcal{K} = \{0, 1, \ldots, K-1\}$. Пусть $\chi = \sum\limits_{j \in \mathcal{K}} \lambda_j e_j$. Если дано слово $a \in \{0,1\}^n$ для вопросного состояния χ мы определим:

$$\delta_a(\chi) = \sum_{j:\ q(e_j)=a} |\lambda_j|^2.$$

Это - вероятность того, что состояние χ вызывает оракул на слове a. В частности, $\sum\limits_{a \in \{0,1\}^n} \delta_a(\chi) = 1$.

Каждое вопросное состояние χ индуцирует метрику на множестве всех оракулов, если для функций f, g вида $\{0,1\}^n \longrightarrow \{0,1\}^n$ определить расстояние между ними как

$$\delta_\chi(f,g) = \left(\sum_{a:\ f(a) \neq g(a)} \delta_a(\chi) \right)^{1/2}.$$

Лемма 3 *Пусть* Qu_f, Qu_g *будут унитарными преобразованиями на квантовой части QC, соответствующими функциям* f, g; χ *- вопросным состоянием. Тогда*

$$\|\mathrm{Qu}_f(\chi) - \mathrm{Qu}_g(\chi)\| \leq 2\delta_\chi(f,g).$$

Доказательство

Положим $\mathcal{L} = \{j \in \mathcal{K} \mid f(q(e_j)) \neq g(q(e_j))\}$. Имеем: $\|\mathrm{Qu}_f(\chi) - \mathrm{Qu}_g(\chi)\| \leq 2(\sum\limits_{j \in \mathcal{L}} (|\lambda_j|)^2)^{1/2} \leq 2\delta_\chi(f,g)$. Lemma доказана.

Квантовое вычисление имеет вид

$$\chi_0 \longrightarrow \chi_1 \longrightarrow \ldots \longrightarrow \chi_t,$$

где каждый шаг $\chi_i \longrightarrow \chi_{i+1}$ есть суперпозиция вопросного преобразования и следующих унитарных преобразований U_i, зависящих только от i: $\chi_i \xrightarrow{\mathrm{Qu}_f} \chi_i' \xrightarrow{U_i} \chi_{i+1}$. Мы будем обозначать $U_i(\mathrm{Qu}_f(\chi))$ через $V_{i,f}(\chi)$, тогда $\chi_{i+1} = V_{i,f}(\chi_i)$, $i = 0, 1, \ldots, t-1$. Здесь t есть число вопросных преобразований в рассматриваемом вычислении. Мы скажем, что число t есть временная сложность этого вычисления.

Пусть $\delta_a(\chi) = \sqrt{\delta_a(\chi)}$.

Лемма 4 *Если* $\chi_0 \longrightarrow \chi_1 \longrightarrow \ldots \longrightarrow \chi_t$ *есть вычисление с оракулом для* f*, функция* g *отличается от* f *только на одном слове* $a \in \{0,1\}^n$ *и* $\chi_0 \longrightarrow \chi'_1 \longrightarrow \ldots \longrightarrow \chi'_t$ *есть вычисление на том же* QC *с новым оракулом для* g*, то*

$$\|\chi_t - \chi'_t\| \leq 2\sum_{i=0}^{t-1} \delta_a(\chi_i).$$

Доказательство

Индукция по t. Бызис очесвиден. Шаг. Ввиду того, что $V_{t-1,g}$ унитарно, Леммы 1 и индуктивной гипотезы, мы имеем

$$\begin{array}{l} \|\chi_t - \chi'_t\| = \|V_{t-1,f}(\chi_{t-1}) - V_{t-1,g}(\chi'_{t-1})\| \leq \\ \|V_{t-1,f}(\chi_{t-1}) - V_{t-1,g}(\chi_{t-1})\| + \|V_{t-1,g}(\chi_{t-1}) - V_{t-1,g}(\chi'_{t-1})\| \leq \\ 2\delta_a(\chi_{t-1}) + \|\chi_{t-1} - \chi'_{t-1}\| = 2\delta_a(\chi_{t-1}) + 2\sum_{i=0}^{t-2} \delta_a(\chi_i) = 2\sum_{i=0}^{t-1} \delta_a(\chi_i). \end{array}$$

Lemma доказана.

В дальнейшем мы предполагаем, что все вычисления выполняются с фиксированной вероятностью ошибки p_{err}. Это значит, что если B есть множество номеров целевых состояний, то вероятность $\sum_{j\in B} |\lambda_j|^2$ получито одно из них в результате наблюдения конечного состояния есть $\chi_t = \sum_j \lambda_j e_j$ не меньше, чем $1 - p_{err}$.

7.2 Сильная нижняя оценка для квантового перебора

начала займемся проблемой поиска экстремума булевской функции. Если дан оракул для функции $\phi: \{0,1\}^n \longrightarrow \{0,1\}$ из достаточно широкого класса S, что будет нижней оценкой для временной сложности квантового поиска ее точки экстремума? Мы будем требовать, чтобы наши алгоритмы давали правильный ответ не обязательно для всех функций ϕ но только для йункций из некоторого множества $G \subseteq S$. Пусть у нас есть две фиксированные константы:

1) максимально допустимая вероятность ошибки $\epsilon > 0$ (для вычислений с оракулом для $\phi \in G$), и

2) вероятность применимости алгоритма: $\mathrm{card}(G)/\mathrm{card}(S)$ так что эта дробь должна быть самое большее p для некоторого $p: 0 < p \leq 1$.

Если S есть множество всех булевских функций, наилучшая возможная нижняя оценка в квантовом случае, равно как и в классическом, будет $O(1)$. Это потому, что простеший классический алгоритм, проверяющий $\phi(0), \phi(1), \ldots, \phi(k)$ даст правильный ответ для функции, выбранной с вероятностью $p = 1 - 2^{-k}$.

Пусть $S = S_b$ есть множество всех булевских функций с ровно b точками x, такими что $\phi(x) = 1$. Пусть далее $n, t(n), b(n)$ меняются так, что $t = o(\sqrt{N/b})$, $n \longrightarrow \infty$, $N = 2^n$. Таким образом, квантовый алгоритм с временной сложностью $t(n)$ будет существенно быстрее, чем G-BBHT. Мы докажем, что если мы применим этот алгоритм к поиску точки экстремума ϕ то он будет давать неправильный ответ для большинства функций ϕ.

Теорема 4 *Путсь* $t(n) = o(\sqrt{N/b(n)})$, $n \longrightarrow \infty$*, и некоторый квантовый компьютер с оракулом для* ϕ *и временной сложностью* $t(n)$ *ищет решения уравнения* $\phi(x) = 1$ *с фиксированной верхней границей* ϵ *для вероятности ошибки* $(0 < \epsilon < 1)$*. Пусть* $p(n)$ *есть вероятность того, что*

этот алгоритм даст верный ответ для оракула ϕ, выбранного случайно из S_b (по умолчанию все распределения мы полагаем равномерными). Тогда $p(n) \longrightarrow 0 \ \ (n \longrightarrow \infty)$.

Proof

Мы применим идею доказательства теоремы 2 из работы [Oz] с некоторыми модификациями. Фиксируем n и положим $\phi_0(x) = 0$. Пусть $X_0 \longrightarrow X_1 \longrightarrow \ldots \longrightarrow X_t$ есть вычисление на рассматриваемом квантовом компьютере. Определим матрицу $a_{ij} = \delta_j(X_i),\ i = 1, 2, \ldots, t;\ j = 1, 2, \ldots, N$, где $N = 2^n$. Тогда мы имеем $\sum\limits_{ij} a_{ij} \leq t$, так как $\forall i \ \sum\limits_{j} a_{ij} \leq 1$.

Пусть T_j есть множество всех целых чисел τ таких что $\sum\limits_{i} a_{i\tau} \leq (j+1)t/N$; примем $T_0 = \emptyset$. Пусть $\hat{b}_j$ обозначает мощность множества $L_j = T_j \setminus T_{j-1}$. Тогда $\sum\limits_{j} \frac{\hat{b}_j (j+1)t}{N} \leq t$.

Выберем произвольно b разных целых чисел из $1, 2, \ldots, N$, обозначаем это множество через D и пусть b_j будет числом тех целых чисел среди них, которые принадлежат множеству L_j. TOгда b_j будет случайной величиной с математическим ожиданием $\mathrm{E}b_j = b\hat{b}_j/N$. Теперь изменим значения ϕ_0 на D на 1. Мы получим новую функцию ϕ_1 и соответственно новое вычисление $X'_0 = X_0 \longrightarrow X'_1 \longrightarrow \ldots \longrightarrow X'_t$ с оракулом для ϕ_1. Норма разности между конечными состояниями $\xi = \|X_t - X'_t\|$ будет таким образом вещественной случайной величиной. Оценим ее математическое ожидание.

Лемма 5 *Для любого $\varepsilon > 0$ $P(\xi > \varepsilon) \longrightarrow 0$ если $n \longrightarrow \infty$.*

Доказательство

Нам понадобится следующее неравенство, справедливое для любой случайной величины: $\mathrm{E}\eta^2 \geq \mathrm{E}^2\eta$.

Если i принимает все значения $1, 2, \ldots, N$; j принимает все натуральные значения. Мы имеем:

$$\mathrm{E}\xi = 2\mathrm{E}\sum_i \sqrt{\sum_{\tau \in D} a_{i\tau}} \leq 2\mathrm{E}\sqrt{t\sum_j b_j(j+1)t/N} = \tfrac{t\sqrt{b}}{\sqrt{N}} 2\mathrm{E}\sqrt{\sum_j b_j(j+1)/b} \leq$$
$$o(1)\sqrt{\mathrm{E}\sum_j b_j(j+1)/b} \leq o(1)\sqrt{\tfrac{1}{b}\sum_j \tfrac{b\hat{b}_j(j+1)}{N}} = o(1) \ \ (n \longrightarrow \infty).$$

Теперь, применяя неравенство Чебышева $P(\xi \geq \varepsilon) \leq \mathrm{E}\xi/\varepsilon$ мы заключаем, что если ε фиксировано, то $P(\xi \geq \varepsilon)$ может быть сделано произвольно малым для достаточно больного n. Lemma 3 доказана.

Вернемся к доказательству Теоремы 1. Предположим, что наш компьютер дает правильный ответ на всех функциях из G с вероятностью ошибки p_{err}. Без потери общности можно принять $p_{err} = 0.0016,\ N > 1000$. Выберем булевскую функцию $f \in G$, принимающую значение 1 в b точках. Пусть конечное состояние вычисления на нашем компьютере с оракулом f имеет вид $X_t = \sum\limits_j \lambda_j e_j$. Пусть $B = \{j \mid f(e_j) = 1\}$, $\varepsilon_0 = \sum\limits_{j \notin B} |\lambda_j|^2$. Мы имеем

$$\varepsilon_0 \leq p_{err}, \tag{7.1}$$

поскольку конечное наблюдение X_t должно давать результат $e_j,\ j \in B$ с вероятностью ошибки p_{err}. Фиксируем такую f и положим $c_j = j/N,\ j = 0, 1, \ldots;\ \ L_j = \{j \in B \mid \ c_j \leq |\lambda_j|^2 < c_{j+1}\}, \zeta_0 = \sum\limits_j \hat{l}_j c_j$ где $\hat{l}_j = \mathrm{card}(L_j)$. Мы имеем

$$|1 - \zeta_0| \leq \varepsilon_0 + \frac{b}{N} < 2p_{err} \ \ (N \longrightarrow \infty). \tag{7.2}$$

Теперь выберем вторую функцию $f' \in S_b$ случайно. Пусть $B' = \{j \mid f'(e_j) = 1\}$. Определим случайные величины l_j, зачасящие от f':

$$l_j = \text{card } \{j \mid j \in L_j \cap B'\}.$$

Мы имеем $\text{E}l_j = b\hat{l}_j/N$, поскольку вероятность выбора f' распределена равномерно по всему S_b. Наконец, определим $\zeta = \sum\limits_j l_j c_j$. Это тоже будет случайной величиной, закисящей от f'. Ее математическое ожидание будет

$$\text{E}\zeta = \sum_j c_j \text{E}l_j = \sum_j \frac{c_j b\hat{l}_j}{N} = O(1)b/N = o(1) \;\; (N \longrightarrow \infty)$$

ввиду (2). Тогда неравенство Чебышева $P(\zeta \geq 0.9) \leq \frac{10}{9}\text{E}\zeta$ gives

$$P(\zeta \geq 0.9) \longrightarrow 0 \;\; (N \longrightarrow \infty). \tag{7.3}$$

Теперь предположим, что $\text{card}(G)/\text{card}(S_b) = \epsilon_0 = \text{const}$.

Пусть $X'_t = \sum\limits_j \lambda'_j e_j$ есть конечное состояние вычисления с оракулом для выбранной функции f'. Если $f' \in G$ (т.е. с вероятностью ϵ_0) то мы имеем

$$0 \leq \sum_{j \notin B'} |\lambda'_j|^2 \leq p_{err}. \tag{7.4}$$

ы получим, что с вероятностью 1

$$\xi^2 \longrightarrow 0 \;\; (N \longrightarrow \infty). \tag{7.5}$$

Мы имеем

$$\begin{aligned} \xi^2 = \|X_t - X'_t\|^2 = \sum_{j \in B \setminus B'} |\lambda_j - \lambda'_j|^2 + \sum_{j \in B' \setminus B} |\lambda_j - \lambda'_j|^2 + \\ \sum_{j \notin B \cup B'} |\lambda_j - \lambda'_j|^2 + \sum_{j \in B \cap B'} |\lambda_j - \lambda'_j|^2. \end{aligned} \tag{7.6}$$

Положим $\sum\limits_{j \in B' \setminus B} |\lambda'_j|^2 = q'$, $\sum\limits_{j \notin B \cup B'} |\lambda'_j|^2 = z'$, $\sum\limits_{j \in B' \setminus B} |\lambda_j|^2 = q$, $\sum\limits_{j \notin B \cup B'} |\lambda_j|^2 = z$, $\sum\limits_{j \in B \cap B'} |\lambda_j|^2 = r$, $\sum\limits_{j \in B \cap B'} |\lambda'_j|^2 = r'$.

Теперь ввиду (1) $q \leq \varepsilon_0 \leq p_{err}$ and $z \leq p_{err}$. Мы будем использовать неравенство $\|a - b\| \geq |\|a\| - \|b\||$ для двух векторов a, b в гильбертовом пространстве. Используя это неравенство, мы заключаем, что второй член в (6) не меньше чем $\delta = |\sqrt{q'} - \sqrt{q}|^2$. Третий член не меньше чем $|\sqrt{z'} - \sqrt{z}|^2$. Пусть N достаточно большой, так что

$$\xi^2 < p_{err}. \tag{7.7}$$

Такой N существует по (5). Тогда мы имеем $q' < 4p_{err}$. Действительно, в противном случае: $q' \geq 4p_{err}$ ввиду (6) мы бы имели $p_{err} > \delta \geq (\sqrt{q'} - \sqrt{p_{err}})^2 \geq p_{err}$ что и дает противоречие. Аналогично, $z' < 4p_{err}$. Таким образом, асимптотически, при $N \longrightarrow \infty$ with probability 1: $\sum\limits_{j \notin B} |\lambda'_j|^2 = q' + z' <$

$8p_{err}$. Следовательно< с этой вероятностью $\sum_{j\in B}|\lambda'_j|^2 > 1-8p_{err}$. Принимая во внимание (4), мы получаем, что с вероятностью ϵ_0

$$r' > 1-9p_{err}. \tag{7.8}$$

Из определения L_j следует, что

$$|\zeta - r| < \frac{1}{N}. \tag{7.9}$$

С другой стороны, (6) и (7) дают $|r-r'| = |\sqrt{r}-\sqrt{r'}|(\sqrt{r}+\sqrt{r'}) < 2\sqrt{s} \le 2\sqrt{p_{err}}$, где s в четвертом члене суммы (6). Теперь (9) дает $|\zeta - r'| < \frac{1}{N} + 2\sqrt{p_{err}}$, и по (8) $\zeta > 1-9p_{err}-2\sqrt{p_{err}}-\frac{1}{N} > 0.903$ с вероятностью ϵ_0, что противоречит (3). Теорема 1 доказана.

7.2.1 Как много задач можно ускорить квантово

Однако, есть естественные задачи, для которых квантовый компьютер не может ускорить классический. Пусть ω^* обохначает множество всех слов в алфавите ω. Для сохраняющей длину слова функциии $f:\ \{0,1\}^* \longrightarrow \{0,1\}^*$ and $x \in \{0,1\}^n$ результат k итерированных применений f определяется следующей индукцией $f^{\{0\}}(x) = x,\ f^{\{k+1\}}(x) = f(f^{\{k\}}(x))$. В работе [Oz97] доказано, что результат этого вычисления:

$$x \longrightarrow f(x) \longrightarrow f(f(x)) \longrightarrow \ldots \longrightarrow \underbrace{f(\ldots f}_{T}(x)\ldots) = f^{\{T\}}(x) \tag{7.10}$$

не может быть найден на квантовом компьютере существено быстрее, чем на классическом, если $T = O(2^{n/7})$.

В чем важность этой модели "черного ящика"? Дело в следующем неформальном принципе, вытекающем из классической теории алгоритмов.

Принцип релятивизации *Всякий общий принцип, который может быть релятивизован, остается истинным и после релятивизации*[1].

Из этого общего принципа следует, что если дан код классического алгоритма, то единственным путем получить результат его действия на входном слове x длины n является запуск этого алгоритма на x. При этом в ходе вычисления код алгоритма может применяться только как "черный ящик", поскольку в общем случае мы не можем успешно проанализировать его внутреннюю структуру на предмет "предугадывания"результата вычисления. Таким образом, мы можем принять, что типичное классическое вычисление имеет вид (1), где сохраняющая длину функция f используется как оракул. Временная сложность этого вычисления есть T с точностью до мультипликативной константы. Результат работы [Oz97] был усилен в работах [FGGS98] и [BBCMW98] до произвольных времен T. А именно, в обеих этих работах было независимо доказано, что любое квантовое вычисление метафункции PARITY : $\mathrm{Par}\,(g) = \bigoplus_x g(x)$ от ыункции $g:\ \{0,1\}^n \longrightarrow \{0,1\}$ требует ровно 2^{n-1} обращений к g (наполовину меньше, чем в классическом случае).

В работе [BBCMW98] изучались вычисления метафункций вида $F:\ \{g\} \longrightarrow \{0,1\}$, где $\{g\}$ есть множество функций вида $g:\ \{0,1\}^n \longrightarrow \{0,1\}$. В частности, было доказано, что если $T =$

[1] Под релятивизацией мы понимаем здесь добавление к условию решения проблемы возможности обращаться к произвольному но фиксированному оракулу, то есть попросту - добавление оракула для некоторой функции или множества к арсеналу допустимых средств решения.

$o(2^n)$, $n \longrightarrow \infty$, то только исчезающе малая часть таких метафункций может быть вычислена с ровно T обращениями к g на квантовом компьютере. Единственный возможный способ извлечения нижней границы для итерированного применения "черного ящика"из нижней границы для метафункций есть вычисление PARITY. Алгоритм вычисления Par (g) может быть представлен как итерированное применение специфического "черного ящика", использующего g как подпрограмму. Множество таких специфических "черных ящиков"типа PARITY имеет вероятностную меру нуль среди всех возможных "черных ящиков", так что две упомянутые работы вполне допускали возможность того, что для достаточно большой части оракулов существует квантовй алгоритм, ускоряющий их итерацию.

Здесь мы докажем, что если T не слишком велико, то множество "черных ящиков", чье итерированное применение T раз допускает квантовое ускорение, имеет вероятностную меру нуль.

Теорема 5 *Если $T = O(2^{\frac{n}{7+\varepsilon}})$, $\varepsilon > 0$, то для "черного ящика"f, выбранного случайно, с вероятностью 1 любое квантовое вычисление T его итераций требует T обращений к f.*

А вот другая интерпретация Теоремы 1. Любое слово длины n можно рассматривать как состояние конечной классической системы. Тогда функция f, сохраняющая длину, будет некоторым законом, определяющим эволюцию (1) такой системы во времени. Если дано некоторе начальное состояние этой системы, соответствующее слову x, то конечное состояние эволюции будет соответствовать искомому слову $f^{\{T\}}(x)$. Если для такого закона f некоторый квантовый компьютер находит искомое слово быстрее, чем за времы T, это означает, что с таким компьютером мы можем узнать конечное состояние нашей системы при заданной эволюции быстрее, чем эта эволюция закончится. В этом случае мы скажем, что данный квантовый компьютер предсказывает эволюцию классической системы. Теоремы 1 утверждает, что все, кроме слишком длинных, эволюции классических систем не предсказуемы на квантовом компьютере.

Для произвольного числа T итераций справедлива более слабая нижняя оценка для квантового моделирования, устанавливаемая следующей теоремой.

Теорема 6 *Для "черного ящика"f, выбранного произвольно, с вероятностью 1 любое квантовое вычисление T итераций f требует $\Omega(\sqrt{T})$ обращений к f.*

7.3 Основы вероятностного метода

Для анализа модели "черных ящиков"нам нужны некоторые понятия теории вероятностей. Если дано множество $\mathcal{N}$, мы скажем, что некоторое множество $\S \subseteq 2^{\mathcal{N}}$ его подмножеств есть σ-алгебра (алгебра) на $\mathcal{N}$ если $\emptyset, \mathcal{N} \in \S$ и $\S$ замкнуто относительно операций вычитания: $A \setminus B$ и счетных (конечных) объединений и пересечений: $\bigcup_{i=0}^{\infty} A_i$, $\bigcap_{i=0}^{\infty} A_i$. Элементы $\S$ называются событиями.

Вероятностная мера на $\S$ есть такая вещественная функция на соытиях $P : \S \longrightarrow [0,1]$ что $P(\emptyset) = 0$, $P(\mathcal{N}) = 1$, и для всякого списка $\{A_i\}$ взаимно несовместных событий выполнена следующая аксиома аддитивности.

$$P\left(\bigcup_{i=0}^{\infty} A_i\right) = \sum_{i=0}^{\infty} P(A_i).$$

Минимальная σ-алгебра, содержащая данную алгебру $S \subseteq 2^{\mathcal{N}}$ обозначается через $\S(S)$. Если вероятностная мера на алгебре S может быть расширена до вероятностной меры на $\S(S)$, мы будем обозначать ее той же буквой P.

Пусть M_n обозначает множество всех отображений $g: \{0,1\}^n \longrightarrow \{0,1\}^n$. Пусть $\mathrm{card}(M_n) = v_n$. Мы имеем $v_n = 2^{n2^n}$. Пусть F будет множеством всех оракулов. Элемент F будет сохраняющей длину функцией $f: \{0,1\}^* \longrightarrow \{0,1\}^*$, которую можно рассматривать как список $g_1, g_2, \ldots$ функций $g_i \in M_i$. Мы готовы определить вероятностную меру, равномерно распределенную на множестве всех оракулов. Для всякого фиксированного $g_i \in M_i \quad i = 1, 2, \ldots, n$ положим $A(g_1, g_2, \ldots, g_n) = \{f \mid f = (g_1, g_2, \ldots, g_n, \ldots)\}$ и определим $P(A(g_1, \ldots, g_n)) = (v_1 v_2 \ldots v_n)^{-1}$. Тогда аксиома аддитивности будет выполнена для расширения P на минимальную алгебру S, содержащую все $A(g_1, \ldots, g_n)$ для всех n и $g_1, g_2, \ldots, g_n$, и следовательно, до вероятностной меры на $\S(S)$.

Definition Вероятностная мера на оракулах, распределенная равномерно, есть вероятностная мера P на σ-алгебре $\S = \S(S)$.

Пример Пусть дано n и два слова $x, y \in \{0,1\}^n$. Тогда вероятность того, что $f(x) = y$, есть $P(B_{xy})$, где $B_{xy} = \{f \mid f(x) = y\}$. Эта вераоятность равна 2^{-n}.

Для событий $A, B \in \S,\ P(B) \neq 0$ условная вероятность определяется как $P(A \mid B) = P(A \cap B)/P(B)$. Полная группа событий для A есть такое множество $F_1, F_2, \ldots, F_m$ событий с ненулевыми вероятностями, что $F_i \cap F_j = \emptyset$ для $i \neq j$ и $A \in \bigcup_{i=1}^{m} F_i$. В этом случае $P(A) = \sum_{i=1}^{m} P(A \mid F_i) P(F_i)$ (формула полной вераоятности).

7.4 Невозможность квантового ускорения для большинства коротких вычислений

Доказательство Теоремы 1

Пусть $t(n), T(n)$ есть целочисленные функции, $T = O(2^{\frac{n}{7+\varepsilon}})$, $\varepsilon > 0$, C есть квантовый компьютер. Обозначим через $S(C, n, t, T)$ множество таких функций $f \in M_n$, что C вычисляет $f^{\{T\}}(\bar{0})$, используя не больше чем t обращений к f, где $\bar{0}$ есть слово, состоящее из нулей.

Лемма 6 *Для любого квантового компьютера C и $\epsilon > 0$ существует такой номер n что $P(S(C, n, T-1, T)) < \epsilon$.*

Доказательство Леммы 3

Для доказательства Лемы 3 нам нужно несколько технических предложений. Положим $\alpha = 5 + \frac{\varepsilon}{2}$. Фиксируем номер n.

Мы сейчас определим список вида $\zeta_i = \langle \xi_i, f_i, \mathcal{T}_i, x_i \rangle$, где ξ_i есть состояние из $\mathcal{H}_1$, $|\xi_i| = 1$, $f_i \in M_n$, $x_i \in \mathcal{T}_i \subseteq \{0,1\}^n$, с помощью следующей индукции по i.

Определение

Базис: $i = 0$. Пусть $\xi_0 = \chi_0$, а $f_0 \in M_n$ выбран случайно, $x_0 = \bar{0}$, $\mathcal{T}_0 = \{0,1\}^n$.

Шаг. Положим

$$\xi_{i+1} = V_{i,f_i}(\xi_i),$$
$$\mathcal{T}_{i+1} = \mathcal{T}_i \cap R_i, \quad R_i = \{a \mid \delta_a(\xi_{i+1}) < \tfrac{1}{T^\alpha}\},$$

Мы определим x_{i+1} как произвольно выбранный элемент из $\mathcal{T}_{i+1}$, и положим

$$f_{i+1} = \begin{cases} f_i(x), & \text{if } x \neq x_i, \\ x_{i+1}, & \text{if } x = x_i. \end{cases}$$

Список вида ζ_i определен необнозначно, и мы обозначим множество всех таких списков ζ_i через D_i, $i = 1, 2, \ldots$. Пусть N_i есть множество таких функций $f_i \in M_n$, что существуют $\xi_i, \mathcal{T}_i, x_i$, такие что $\langle \xi_i, f_i, \mathcal{T}_i, x_i \rangle \in D_i$.

Предложение 3 *Если $i \leq T$, $n \longrightarrow \infty$, то*

$$P(N_i) = 1 - O\left(\frac{T^{\alpha+1} i}{2^n}\right).$$

Доказательство Предложения 1

Индукция по i. Базис вытекает из определения ζ_0. Шаг. Если задан некоторый список $\zeta_i = \langle \xi_i, f_i, \mathcal{T}_i, x_i \rangle$, в переходе к ζ_{i+1} единственный произвольный выбор - это выбор x_{i+1}. Этот выбор может быть сделан корректно с вероятностью $\frac{2^n - T^{\alpha+1}}{2^n}$, так как $\mathrm{card}(\mathcal{T}_i) > 2^n - T^\alpha i \geq 2^n - T^{\alpha+1}$. Значит, ввиду индуктивного предположения результирующая вероятность будет $\left(1 - O\left(\frac{T^{\alpha+1} i}{2^n}\right)\right)\left(1 - \frac{T^{\alpha+1}}{2^n}\right) = \left(1 - O\left(\frac{T^{\alpha+1}(i+1)}{2^n}\right)\right)$ с некоорой константой. Предложение 1 доказано.

Вернемся к доказательству Леммы 3. Пусть в дальнейшем $t = T - 1$. Если даны списки ζ_i, мы введем следующие обозначения: $V_i = V_{i,f_t}$, $V_i^* = V_{i,f_i}$. Пусть унитарный оператор V^i определяется следующей индукцией: $V^0(x) = V_0(x)$, $V^i(x) = V_i(V^{i-1}(x))$, и унитарный опрератор $\tilde{V}_i$ определяется как $\tilde{V}_0 = V_0^*$, $\tilde{V}_i(x) = V_i^*(\tilde{V}_{i-1}(x))$. Тогда $\xi_{i+1} = \tilde{V}_i(\xi_0)$.

Пусть $\xi_0' = \xi_0$, $\xi_{i+1}' = V^i(\xi_0)$, $\partial_i = |\xi_i - \xi_i'|$, $\Delta_i = |V_i^*(\xi_i) - V_i(\xi_i)|$. Из определения следует, что f_i отличается от f_t самое большее на множестве $X_i = \{x_i, x_{i+1}, \ldots, x_{t-1}\}$, где $\forall a \in X_i \ \delta_a(\xi_i) < \frac{1}{T^\alpha}$. Следовательно, применяя Лемму 1, мы получим

$$\Delta_i \leq \frac{2t^{1/2}}{T^{\alpha/2}}. \tag{7.11}$$

Предложение 4 $\partial_i \leq \sum_{k<i} \Delta_k$.

Доказательство

ИНдукция по i. Базис вытекает из определений. Шаг.

$$\partial_{i+1} = |\tilde{V}_i(\xi_0) - V^i(\xi_0)| = |V_i^*(\tilde{V}_{i-1}(\xi_0)) - V_i(V^{i-1}(\xi_0))| \leq$$
$$\leq |V_i^*(\xi_i) - V_i(\xi_i)| + |V_i(\xi_i) - V_i(\xi_i')| = \Delta_i + \partial_i.$$

Применяя индуктивное предположение, мы завершаем доказательство.

Таким образом, ввиду (3) Предложение 2 дает

$$\forall i = 1, \ldots, t \quad \partial_i \leq \frac{2it^{1/2}}{T^{\alpha/2}}. \tag{7.12}$$

Из определения функций f_i следует, что $\forall i \leq t \;\; \delta_{x_t}(\xi_i) < \frac{1}{T^\alpha}$. Принимая во внимание неравенство (7.12), мы заключаем, что для $x = x_t$

$\delta_x(\xi_i - \xi_i') \leq \frac{2it^{1/2}}{T^{\alpha/2}}$, $\delta_x(\xi_i) < \frac{1}{T^{\alpha/2}}$, $\delta_x(\xi_i') \leq \delta_x(\xi_i - \xi_i') + \delta_x(\xi_i)$.

Следовательно, мы имеем

$$\delta_x(\xi_i') \leq \frac{3t^{3/2}}{T^{\alpha/2}}. \tag{7.13}$$

Теперь рассмотрим некоторый оракул $f_{t+1} = f_T$. Если $\xi_0 \longrightarrow \xi_1'' \longrightarrow \ldots \longrightarrow \xi_t''$ есть вычисление $f_{t+1}^{\{T\}}(\bar{0})$ на нашем QC с оракулом для f_{t+1}, то Lemma 2 и неравенство (7.13) дают

$$|\xi_t' - \xi_t''| < 2\sum_{i \leq t} \delta_x(\xi_i') \leq \frac{6t^{5/2}}{T^{\alpha/2}} < \gamma(n)$$

для $\alpha = 5 + \frac{\varepsilon}{2}$, где $\gamma(n)$ может быть сделано произвольно малым для подходящих n. Значит, наблюдения состояний ξ_t' и ξ_t'' дают одинаковые результаты с близкими вероятностями. Тогда если наш компьютер вычисляет $f_{t+1}^{\{T\}}(\bar{0}) = a$, то амплитуды базисных состояний в ξ_t' должны концентрироваться на только одном единственном базисном состоянии, соответствующем a.

Пусть $P(\text{not} \mid f_t)$ есть вероятность выбора оракула вида f_T,такого что $f_T^{\{T\}}(\bar{0}) \neq f_{T-1}^{\{T\}}(\bar{0})$ дает f_t. Ввиду определения вычисления это есть вероятность того, что с выбранным f_t наш компьютер не вычисляет $f_T^{\{T\}}(\bar{0})$ правильно. Мы имеем $P(\text{not} \mid f_t) = \frac{2^n - T^{\alpha+1}}{2^n} \longrightarrow 1 \; (n \longrightarrow \infty)$ Для всякого выбора f_t, поскольку есть не менее $2^n - T^{\alpha+1}$ подходящих способов выбора x_{t+1}. Далее, пусть $\tilde{p}$ есть вероятность выбрать оракул f_T, такой что наш компьютер не вычисляет $f_T^{\{T\}}(\bar{0})$ правильно. По формуле полной вероятности и Предложению 1 мы имеем

$$\tilde{p} = \sum_{f_t} P(\text{not} \mid f_t)p(f_t) = \frac{2^n - T^{\alpha+1}}{2^n}\left(1 - O\left(\frac{T^{\alpha+2}}{2^n}\right)\right) \longrightarrow 1 \; (n \longrightarrow \infty).$$

Наконец, вероятность p_{not} выбора оракула f, так что $f^{\{T\}}(\bar{0})$ не вычисляется на нашем компьютере, будет равна $p_{\text{not}} \geq \tilde{p}$, и $p_{\text{not}} \longrightarrow 1 \; (n \longrightarrow \infty)$.

Lemma 3 доказана.

Теперь вернемся к доказательству Теоремы 1. Пусть $C_1, C_2, \ldots$ - все квантовые компьютеры с оракулом, взятые в некотором фиксированном порядке, $R(C, n, t, T)$ обозначает множество всех функций $g_n \in M_n$, таких что C не вычисляет $g_n^{\{T\}}(\bar{0})$, используя не более чем t обращений к g_n. Возьмем произвольно $\epsilon > 0$. Применяя Лемму 3, найдем для любого $i = 1, 2, \ldots$ такой номер n_i, что $P(R(C_i, n_i, T-1, T)) > 1 - \epsilon 2^{-i}$. Далее, если $\mathcal{N}$ обозначает множество оракулов, чье T итерированное применение не допускает квантового ускорения, мы имеем $\bigcap\limits_{i=1}^{\infty} R(C_i, n_i, T-1, T) \subseteq \mathcal{N}$. Для дополнительного множества $\bar{\mathcal{N}} \subseteq \bigcup\limits_{i=1}^{\infty} S(C_i, n_i, T-1, T)$. По аксиоме аддитивности $P(\bar{\mathcal{N}}) \leq \sum\limits_{i=1}^{\infty} P(S(C_i, n_i, T-1, T)) = \sum\limits_{i=1}^{\infty} \epsilon 2^{-i} = \epsilon$. Теорема 1 доказана.

7.5 Нижняя оценка для квантового моделирования в общем случае

Доказательство Теоремы 2

Как и в предыдущем параграфе, будет достаточно доказать такую лемму

Лемма 7 *Для любого квантового компьютера C, $\epsilon > 0$ и функций $t(n), T(n)$: $t^2 = o(T)$ $(n \longrightarrow \infty)$ существует номер n, такой что $P(S(C,n,t,T)) < \epsilon$.*

Доказательство

В наших обозначениях для выбранного случайно оракула f и номера n положим $f^k = f^{\{k\}}(\bar{0})$, $k = 0,1,\ldots,T$. Определим матрицу (a_{ij}) следующим образом: $a_{ij} = \delta_{f^j}(\chi_i)$, $i = 0,1,\ldots,t$; $j = 0,1,\ldots,T$.

Тогда мы имеем для всякого $i = 0,\ldots,t \quad \sum\limits_{j=0}^{T} a_{ij} \leq 1$, следовательно, $t \geq \sum\limits_{i=0}^{t}\sum\limits_{j=0}^{T} a_{ij} = \sum\limits_{j=0}^{T}\sum\limits_{i=0}^{t} a_{ij}$. Зафиксируем некоторое $q > 0$. Существует по крайней мере $T(1-1/q)$ таких $\tau \in \{0,1,\ldots,T\}$, что $\sum\limits_{i=0}^{t} a_{i\tau} \leq \frac{qt}{T}$. Выберем одно из таких τ.

Изменив произвольно значение f только на одном слове f^τ, мы получим новую функцию g, где $g^{\{T\}}(\bar{0}) \neq f^{\{T\}}(\bar{0})$ с вероятностью $p_n \longrightarrow 1$ $(n \longrightarrow \infty)$. Пусть $\chi_0 \longrightarrow \chi'_1 \longrightarrow \ldots \longrightarrow \chi'_t$ есть вычисление на QC с оракулом для g. Тогда для такого выбора g с вероятностью p_n мы будем иметь $|\chi_t - \chi'_t| \geq 1/4$, если $f \in S(C,n,t,T)$.

С другой стороны, Lemma 2 дает $|\chi_t - \chi'_t| \leq 2\sum\limits_{i=0}^{t}\sqrt{a_{i\tau}} \leq 2\sqrt{t\sum a_{i\tau}} \leq 2t\sqrt{q}/T^{1/2} < \gamma(n) \longrightarrow 0$ $(n \longrightarrow \infty)$. Здесь q может быть сколь угодно большим. Тогда по опредлелению вычисления с вероятностью $\tilde{p}_n \longrightarrow 1$ $(n \longrightarrow \infty)$ $\quad g^{\{T\}}(\bar{0})$ не вычисляется на данном квантовом компьютере. Lemma 4 доказана. Теорема 2 вытекает из Леммы 4 точно так же, как Теорема 1 из Леммы 3. Теорема 2 доказана.

7.5.1 Значение нижних оценок квантовой сложности

Итак, мы познакомились с некоторыми нижними оценками на время работы квантовых алгоритмов, которые показывают принципиальные границы возможностей квантовых вычислений даже в случае их успешной реализации. Изложенные здесь результаты представляют только малую часть известных нижних оценок квантовой сложности, и интересующиеся читатели могут найти целую серию результатов, помимо уже процитированных, в электронном архиве http://xxx.lanl.gov, раздел quant-ph. Из приведенных нижних оценок вытекают два важных вывода. Первый состоит в том, что бесполезно пытаться с помощью квантового компьютера совершить чудо, то есть решить произвольную переборную проблему за полиномиальное время. Второй вывод состоит в том, что бесполезно пытаться изобрести некий универсальный способ квантового ускорения для классических алгоритмов. Оба этих вывода далеко не тривиальны, и стали осознаваться широкими кругами научной общественности сравнительно недавно[2]. Мы видим, что квантовое ускорение классических вычислений представляет собой очень редкий феномен, своего рода произведение искусства, так что всякий новый случай квантового ускорения вызывает большой интерес. С другой стороны (я думаю, читатель согласится со мной), если бы квантовый компьютер давал невероятно большое

[2]До сих пор в электронном архиве появляются статьи, авторы которых пытаются сделать то, что сделать невозможно. Я периодически натыкаюсь на подобные работы, и раньше не ленился находить в них конкретные ошибки и сообщать об этом авторам этих работ, но теперь махнул на это дело рукой; для читателя же подобное занятие могло бы представить интерес, - и я предлагаю интересующимся заняться этим в качестве упражнения.

ускорение, да еще для очень широкого класса задач, то сама реализуемость этого устройства была бы сомнительна. Нижние оценки квантовой сложности очерчивают теоретические границы возможностей квантовых вычислений, что придает дополнительную весомость и реалистичность самому проекту квантового компьютера.

Заключение

Мы познакомились с основами теории квантовых вычислений и квантового компьютера. Эта теория представляет собой, по-существу, концентрированную "гильбертологию"[3], и четко указывает те цели, которые можно достигнуть путем систематического применения квантового формализма. Надеюсь, что мне удалось передать читателю то ощущение красоты и мощи квантового компьютера, которое возникло у меня самого при изучении этого удивительного объекта. Еще раз подчеркнем, что в современной физике нет никакого запрета на существование масштабируемого квантового компьютера. Поэтому все то, что мы изучили, даже если пока и не реализовано в действующих физических устройствах, имеет законное право на существование в теории.

С другой стороны надо ясно понимать, что все те следствия физических теорий, которые до сих пор подтверждались в экспериментах, выводятся из самих теорий с помощью классических алгоритмов полиномиальной сложности. Иными словами, квантовая механика проверена на практике только в алгоритмически эффективной области. Квантовый компьютер определенно выходит за пределы этой области[4] Это означает, что попытка построить квантовый компьютер в лаборатории представляет самое серьезное испытание для квантовой теории со времени ее возникновения - причем в такой области, в которой она ранее никогда не испытывалась. Никакой из известных в физике экспериментальных методов не может проверить успех в построении квантового компьютера, так как используемый в нем метод принадлежит к совершенно новой области науки и не имеет никаких прецедентов. Единственный критерий успеха - правильное решение тех задач, которые из-за своей сложности не поддаются решению на обычном компьютере. Уникальность ситуации состоит, во-первых, в том, что возможный успех физиков изменит существенные черты современной математики, а во-вторых в том, что физические методы сами по себе не могут быть здесь арбитром: критерий здесь не физический, а алгоритмический. Это означает также и то, что мы никак не можем отказаться от идеи квантового компьютера просто в силу "технологических трудностей"его изготовления, без того, чтобы начать серьезный пересмотр наших взглядов на саму квантовую теорию и, в особенности, на границы ее применимости. Именно поэтому квантовые вычисления представляют собой одну из самых притягательных областей современного естествознания, которая привлекает к себе все больше энтузиастов, и которая, по всей видимости, будет значительной ступенью в истории физики.

Читателям, заинтересовавшимся этим предметом, можно порекомендовать обратиться к электронному архиву статей в http://xxx.lanl.gov, раздел quant-ph, а также сайт кафедры квантовой информатики факультета ВМиК МГУ: http://qi.cs.msu.su.

[3] Термин принадлежит С.Н.Молоткову.

[4] Здесь мы полностью полагаемся на опыт математиков, а не только на полученные ими результаты, см. ([Oz3]).

Литература

[AB] А.И.Ахиезер, В.Б.Берестецкий, Квантовая электродинамика, М., Наука, 1969.

[Be] P.Benioff, Tight Binding Hamiltonians and Quantum Turing Machines, Phys.Rev.Lett. 78 (1997) 590-593.

[Ev] Everett III.H.,Relative state formulation of quantum mechanics, Reviev of modern physics 29, 1957, pp. 454-467. (Reprint in DeWitt and Graham, 1973.)

[Oz] Y.Ozhigov, lanl e-print quant-ph/0303127

[Ozh] Y.Ozhigov, Simulation of quantum interference by reactions of chemical type, lanl e-print quant-ph/

[Pe] Р.Пенроуз, Новый ум короля. - М.:Едиториал УРСС, 2003. - 384 с.

[KB] S.Bravyi, A.Kitaev, Fermionic quantum computation, lanl e-print quant-ph/0003137

[OF] Y.Ozhigov, L.Fedichkin, Quantum Computer with Fixed Interaction is Universal, lanl e-print quant-ph/0202030

[Oz] Yuri Ozhigov, Implementation of Quantum Fourier Transform and Simulation of Wave Functions by Fixed Interaction, lanl e-print quant-ph/0201132

[AL] D.S.Abrams, S.Lloyd, A quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors *lanl e- print quant-ph/9807070*

[Fe] R Feynman, Simulating physics with computers, *J. Theoret. hys., 1982, 21, pp. 467-488*

[FG] E. Farhi, S. Gutmann, An Analog Analogue of a Digital Quantum Computation, *lanl e-print quant-ph/9612026*

[Gr] L. K. Grover, A fast quantum mechanical algorithm for database search. *Proceedings, STOC 1996, 212-219. Philadelphia PA USA, lanl e-print quant-ph/9605043*

[Oz1] Y. Ozhigov, Quantum recognition of eigenvalues, structure of devices and thermodynamic properties *lanl e-print quant-ph/0103073*

[Sh] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Comp. 1997, 26, No. 5, 1484-1509, lanl e-print quant-ph/9508027*

[Wi] S. Wiesner , Simulations of Many- Body Quantum Systems by a Quantum Computer *lanl e-print quant-ph/9603028*

[Za] C. Zalka, Efficient Simulation of Quantum Systems by Quantum Computers, *lanl e-print quant-ph/9603026*

[BBBV] C. H. Bennett, E. Bernstein, G. Brassard, U. Vazirani, Strengths and Weaknesses of Quantum Computing. *SIAM Journal on Computing 1997, 26(5), 1510-1523, lanl e-print quant-ph/9701001*

[BBHT] M. Boyer, G. Brassard, P. Hoyer, A. Tapp, Tight bounds on quantum searching. In *Fourth Workshop on Physics and Computation* (ed. T. Toffoli & M. Biaford & J. Leao), 1996, pp. 36-43. New England Complex Systems Institute.

[Gr1] L. K. Grover, A fast quantum mechanical algorithm for database search. *Proceedings, STOC 1996, 212-219. Philadelphia PA USA.*

[Gr2] L. K. Grover, Rapid sampling through quantum computing *lanl e-print quant-ph/9912001*

[GYWC] Gui Lu Long, Yan Song Li, Wei Lin Zhang, Chang Cun Tu, Intrinsic Limitations on the size of quantum database, *lanl e-print quant-ph/9910076*

[GYWL] Gui Lu Long, Yan Song Li, Wei Lin Zhang, Li Niu, Phase Matching in Quantum Searching, *lanl e-print quant-ph/9906020*

[Mo] С.Н.Молотков,

[MDA] Michael Mussinger, Aldo Delgado, Gernot Alber, Error avoiding quantum codes and dynamical stabilization of Grover's algorithm, *lanl e-print quant-ph/0003141*

[Oz1] Y.I.Ozhigov, Quantum computers speed up classical with probability zero *Chaos, Solitons and Fractals, 1147, (1999), lanl e-print quant-ph/9803064*

[Oz2] Y.I.Ozhigov, Lower bounds of quantum search for extreme point *Proc. Royal. Soc. A (1999) 455, pp. 2165-2172*

[PR] B.Pablo-Norman, M.Ruitz-Altaba, Noise in Grover's quantum search algorithm, *lanl e-print quant-ph/9903070*

[SC] Sixia Yu, Chang-Pu Sun, Quantum searching's underlying SU(2) structure and its quantum decoherence effects, *lanl e-print quant-ph/9903075*

[Ho] P. Hoyer, On Arbitrary Phases in Quantum Amplitude Amplification, *lanl e-print quant-ph/0006031*

[Hol] А.С.Холево,

[Mal] А.А.Мальцев, Алгоритмы и рекурсивные функции, М.: Наука, 1986.

[Ro] Х.Роджерс, Теория рекурсивных функций и эффективная вычислимость, М.: Мир, 1972.

[BS] Н.Н.Боголюбов, Д.В.Ширков, Квантовые поля, М.: Наука, 1980.

[Da] А.С.Давыдов, Квантовая механика, м.: Физматгиз, 1963.

[QC] Introduction to quantum computation and information, ed. H.Kwong Lo, S.Popescu, T.Spiller, World Scientific, 1998.

[Li] H.Lipkin, Quantum mechanics, Elsevier, NY.,1973.

[DFT] Density functional theory at http://www.accelrys.com/technology/qm/erich/dft.htm

[Ac] Accelrys technology at http://www.accelrys.com/technology/

[De] M.Dewar, The moleculat orbital theory of organic chemistry, McGraw-Hill, NY, 1969.

[Zu] L.Zulicke, Quantenchemie, B1, Berlin, 1973.

[Sl] J.C.Slater, Electronic structure of molecules, NY, 1963.

[Wi] S. Wiesner , Simulations of Many-Body Quantum Systems by a Quantum Computer *lanl e-print quant-ph/9603028*

[Za] C. Zalka, Efficient Simulation of Quantum Systems by Quantum Computers, *lanl e-print quant-ph/9603026*

[AL] D.S.Abrams, S.Lloyd, A quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors, *lanl e-print quant-ph/9807070*

[BBHT] Boyer, M., Brassard, G., Hoyer, P., Tapp, A. 1996 Tight bounds on quantum searching. In *Fourth Workshop on Physics and Computation* (ed. T. Toffoli & M. Biaford & J. Leao), pp. 36-43. New England Complex Systems Institute.

[BCW] Buhrman H., Cleve R., Wigderson A., 1998 Quantum vs. Classical Communication and Computation, *Proc. 30th Ann. ACM Symp. on Theory of Computing (STOC 98), pp. 63-68.*

[BHT] G.Brassard, P.Hoyer, A.Tapp, Quantum Counting, *lanl e-print quant-ph/9805082*

[Gr] Grover, L. K. 1996 A fast quantum mechanical algorithm for database search, *Proceedings, STOC 1996, 212-219. Philadelphia PA USA.*

[Oz97] Ozhigov, Y.I., Quantum computer cannot speed up iterated applications of black box, Proceedings of the first NASA conference on quantum computations and quantum communications QCQC'98, Palm Springs, CA, 1998, in Lecture Notes in Computer Science, lanl e-print quant-ph/9712051

[HR] A. Hams, H. de Raedt, Fast Algorithm for Finding the Eigenvalue Distribution of Very Large Matrices, *lanl e-print quant-ph/0004016*

[Ki] A.Kitaev, Quantum measurements and the Abelian Stabilizer Problem, *lanl e-print quant-ph/9511026*

[Oz] Y.Ozhigov, How behavior of systems with sparse spectrum can be predicted on a quantum computer, *lanl e-print quant-ph/0004021*

[Oz3] Y.Ozhigov, Limitation of comutational resource as a physical principle, *lanl e-print quant-ph/*

[Sh] P.W.Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on Quantum Computer, *lanl e-print, quant-ph/9508027 v2 (A preliminary version in Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, Nov. 20-22, 1994, IEEE Computer Society Press, pp 124-134)*

[TM] B.C.Travaglione, G.J.Milburn, Generation of Eigenstates Using the Phase Estimation Algorithm, *lanl e-print quant-ph/0008053*

[Be] P.Benioff, Tight Binding Hamiltonians and Quantum Turing Machines, Phys.Rev.Lett. 78 (1997) 590-593.

[BBCMW98] R. Beals, H. Buhrman, R.Cleve, M. Mosca, R. De Wolf, Tight Quantum Bounds by Polynomials, *lanl e-print quant-ph/9802049*

[FGGS98] E. Farhi, J. Goldstone, S. Gutmann, M. Sipser, A limit on the Speed of Quantum Computation in Determining Parity, *Phys. Rev. Lett., 81, 1998, 5442-5444, lanl e-print quant-ph/9802045*

[Ben] C.H.Bennett, Future directions for quantum information theory, in the book "Introduction to quantum computation and information, World Scientific, 1998, pp. 340-348.

[Gr] Grover, L. K. 1996 A fast quantum mechanical algorithm for database search, *Proceedings, STOC 1996, 212-219. Philadelphia PA USA.*

[Ev] Everett III.H.,Relative state formulation of quantum mechanics, Reviev of modern physics 29, 1957, pp. 454-467. (Reprint in DeWitt and Graham, 1973.)

[Ho] A.S.Holevo, Coding Theorems for Quantum Channels, lanl e-print quant-ph/9809023 .

[Kh] A. Khrennikov, Probabilistic foundations of quantum mechanics and quantum information, lanl e-print quant-ph/0309066.

[Mo] S.N.Molotkov, S.S.Nazin, : Relativistic quantum protocols: "Bit Commitment"and "Coin Tossing", lanl e-print quant-ph/0012075 .

[Oz] Y.Ozhigov, Limitation of computational resources as a physical principle, lanl e-print quant-ph/0303127 .

[Sh] P.W.Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on Quantum Computer, *lanl e-print, quant-ph/9508027 v2 (A preliminary version in Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, Nov. 20-22, 1994, IEEE Computer Society Press, pp 124-134).*

[Pe] R.Penrose, The Emperor's New Mind, Oxford University Press, 384 pp., 1989.