Классические и квантовые вычисления

А. Китаев; написал М. Вялый

5 лекция (5/03/98)

1. Квантовые вычисления

Обычные компьютеры не используют всех возможностей, предоставляемых природой. В них выполняются преобразования на конечных множествах состояний (действия с 0 и 1), а в природе есть возможность делать унитарные преобразования, т. е. действовать на бесконечном множестве. Эта возможность предоставляется квантовой механикой. Устройства (реальные или воображаемые), использующие эту возможность называются квантовыми компьютерами.

Заранее неясно, увеличивает ли переход от преобразований конечных множеств к унитарным преобразованиям конечномерных пространств вычислительные возможности. Сейчас есть основания полагать, что такое увеличение действительно происходит. В качестве примера можно привести задачу о разложении числа на множители: для обычных компьютеров неизвестны полиномиальные алгоритмы, а для квантовых компьютеров такие алгоритмы есть.

Состояние обычного компьютера

$$\begin{array}{ccccc} \square & \square & \cdots & \square & \text{биты} \\ x_1 & x_2 & \ldots & x_n & x_i \in \{0,1\} \end{array}$$

Состояние квантового компьютера

$$\begin{array}{rccccc} & \square & \square & \cdots & \square & \text{q-биты} \\ \text{базисное:} & |\, x_1 & x_2 & \ldots & x_n \,\rangle & x_i \in \{0,1\} \\ \text{произвольное:} & \multicolumn{5}{l}{\sum\limits_{x_1,\ldots,x_n} c_{x_1,\ldots,x_n} |x_1,\ldots,x_n\rangle} \end{array}$$

Обычный компьютер работает с состояниями из конечного числа битов. Каждый бит может находится в одном из двух состояний 0 или 1. Состояние всей системы задаётся указанием значений всех битов. Поэтому множество состояний $\{0,1\}^n$ конечно и имеет мощность 2^n.

Квантовый компьютер работает с конечными наборами элементарных состояний, называемых q-битами. Каждый q-бит имеет два выделенных состояния (если считать q-биты спинами, то это состояния “спин вверх” и “спин вниз”). Указание выделенных состояний для каждого q-бита системы задаёт не все возможные состояния системы, а только базисные. Возможны также любые линейные комбинации базисных состояний с комплексными коэффициентами. Базисные состояния мы будем обозначать $|x_1,\ldots,x_n\rangle$, где $x_i \in \{0,1\}$. Произвольное состояние системы может быть представлено в виде

$$\sum_{(x_1,\ldots,x_n)\in\{0,1\}^n} c_{x_1,\ldots,x_n} |x_1,\ldots,x_n\rangle.$$

Пространство состояний для такой системы — конечномерное (размерности 2^n) пространство над полем комплексных чисел.

Вычисление можно представлять как последовательность выполнения преобразований на множестве состояний системы. Опишем, какие преобразования возможны в классическом, а какие — в квантовом случае.

Классический случай: преобразования — это функции из $\{0,1\}^n$ в $\{0,1\}^n$.	Квантовый случай: преобразования — это унитарные операторы (поэтому пространство состояний системы q-бит — не просто конечномерное комплексное пространство, а конечномерное гильбертово пространство).

2. Определения и обозначения

Пространство состояний системы из n q-бит $\mathbb{C}^n$ можно записать в виде тензорного произведения $\mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2 = \left(\mathbb{C}^2\right)^{\otimes n}$. Сомножители соответствуют пространству состояний одного *q-бита*.

Тензорное произведение двух пространств L и M, в которых фиксированы базисы $\{e_1, \ldots, e_k\}$ и $\{f_1, \ldots, f_s\}$ можно определить как пространство с базисом из элементов $e_i \otimes f_j$. (В данном случае $e_i \otimes f_j$ — это то же самое, что (e_i, f_j), т. е. просто пара векторов). Размерность тензорного произведения равна ks (произведению размерностей сомножителей).

Такое определение *неинвариантно*, т. е. зависит от выбора базисов в перемножаемых пространствах. Можно дать *инвариантное* определение. Для этого рассмотрим вначале (бесконечномерное) пространство с базисом $e \otimes f$, где $e \in L$, $f \in M$ — произвольные вектора из перемножаемых пространств. Тензорное произведение будет факторпространством этого пространства по подпространству, порожденному векторами вида

$$(e_1+e_2)\otimes f - e_1\otimes f - e_2 \otimes f,\ e\otimes(f_1+f_2) - e\otimes f_1 - e\otimes f_2,\ (\lambda e)\otimes f - e\otimes(\lambda f),\ \lambda(e\otimes f) - (\lambda e)\otimes f.$$

Т. е. указанные вектора считаются равными 0.

Можно доказать, что данные определения эквивалентны.

В нашем случае имеется естественный выделенный базис (соответствующий выделенным состояниям) и для $\mathbb{C}^2$ — $\{|0\rangle, |1\rangle\}$, — и для $\left(\mathbb{C}^2\right)^{\otimes n}$ — $\{|x_1, \ldots, x_n\rangle\}$, $x_i \in \{0,1\}$. Коэффициенты разложения вектора по этому базису называются *амплитудами*. Их физический смысл состоит в том, что квадрат амплитуды равен вероятности обнаружить систему в данном состоянии (подробнее об этом будет сказано позже).

Мы будем использовать (и уже использовали) принятые в физике обозначения, относящиеся к векторам и скалярному произведению в гильбертовом пространстве (их ввёл Дирак). Вектора обозначаются $|\xi\rangle$, скалярное произведение — $\langle\xi|\eta\rangle$. Скалярное произведение антилинейно по первому аргументу и линейно по второму, т. е.

$$\langle\xi_1+\xi_2|\eta\rangle = \langle\xi_1|\eta\rangle + \langle\xi_2|\eta\rangle,\ \langle\xi|\eta_1+\eta_2\rangle = \langle\xi|\eta_1\rangle + \langle\xi|\eta_2\rangle,\ \langle c\xi|\eta\rangle = c^*\langle\xi|\eta\rangle,\ \langle\xi|c\eta\rangle = c\langle\xi|\eta\rangle.$$

Здесь и далее c^* обозначает комплексное сопряжение.

Если в обозначении скалярного произведения взять левую половину, то получим *бра-вектор* $\langle\xi|$, т. е. линейный функционал на *кет-векторах* (векторах нашего пространства). Из-за антилинейности скалярного произведения по первому аргументу имеем равенство $\langle c\xi| = c^*\langle\xi|$.

Запись $\langle\xi|A|\eta\rangle$ (A — линейный оператор) можно толковать двояко: либо как скалярное произведение вектора $\langle\xi|$ на вектор $A|\eta\rangle$, либо как $\langle\xi|A$ на $|\eta\rangle$. Так появляется сопряженный оператор $A^\dagger$: по определению, $\langle A^\dagger\xi|$ равен линейному функционалу $\langle\xi|A$. Из определения сразу следует, что $\langle A^\dagger\eta|\xi\rangle = \langle\eta|A|\xi\rangle$.

Унитарный оператор — это линейный оператор, сохраняющий скалярное произведение. Условие

$$\langle\eta|\xi\rangle = \langle U\eta|U|\xi\rangle = \langle\eta|U^\dagger U|\xi\rangle$$

эквивалентно тому, что $U^\dagger U = I$ (тождественному оператору).

В дальнейшем будет использоваться *тензорное произведение* операторов. Оно действует в тензорном произведении пространств, на которых действуют сомножители по правилу $(A \otimes B)|\xi\rangle \otimes |\eta\rangle = A|\xi\rangle \otimes B|\eta\rangle$. Если операторы заданы в матричном виде в некотором базисе, т. е. $A = \sum_{ij} a_{ij}|i\rangle\langle j|$, $B = \sum_{ij} b_{ij}|i\rangle\langle j|$ (легко понять, что $|i\rangle\langle j|$ — линейный оператор, $|i\rangle\langle j||\xi\rangle = \xi_j|i\rangle$), то матричные элементы $C = A \otimes B$ имеют вид $c_{(ik)(jl)} = a_{ij}b_{kl}$.

Вычисление состоит из последовательности преобразований, считаемых элементарными (выполняемых за единицу времени).

Элементарное преобразование **в классическом случае**: такая функция из $\{0,1\}^n$ в $\{0,1\}^n$, которая зависит от небольшого (не зависящего от n) числа бит и изменяет также небольшое число бит.	Элементарное преобразование **в квантовом случае**: $U \otimes I$, где U — произвольный унитарный оператор на части сомножителей $\left(\mathbb{C}^2\right)^r$, т. е. преобразование некоторых q-бит в количестве r штук ($r = O(1)$, достаточно рассматривать $r = 2$), а I — тождественный оператор на остальных q-битах.

Тензорное произведение некоторого оператора U, действующего на множестве q-битов A, и тождественного оператора, действующего на остальных q-битах, будем обозначать $U[A]$.

С этого места начинается вычислительная сложность. Пусть $r = 2$, тогда U — некоторая матрица 4×4, $U \otimes I$ — матрица (экспоненциально) большой размерности, у которой по диагонали стоят блоки из матриц U. Эта матрица представляет один элементарный шаг. Когда применяется несколько таких операторов (к разным парам q-битов), результат будет выглядеть гораздо сложнее. Не видно способа определить этот результат, кроме прямого перемножения матриц. Поскольку размеры матриц экспоненциально велики, потребуется экспоненциальное время для их перемножения.

Заметим однако, что вычисление матричных элементов возможно на полиномиально ограниченной памяти. Пусть нужно найти матричный элемент U_{xy} оператора

$$U \;=\; U^{(l)}[i_l, j_l]\, U^{(l-1)}[i_{l-1}, j_{l-1}]\, \ldots\, U^{(2)}[i_2, j_2]\, U^{(1)}[i_1, j_1].$$

Поскольку

$$\left(U^{(l)} \ldots U^{(1)}\right)_{x_l x_0} = \sum U^{(l)}_{x_l x_{l-1}} \ldots U^{(1)}_{x_1 x_0},$$

можно организовать рекурсивное вычисление значения матричного элемента, используя l штук вспомогательных регистров, в которых будут хранится промежуточные суммы для матричных элементов частичных произведений.

Определение 1. ***Квантовая схема.*** *Пусть $\mathcal{A}$ — некоторое множество унитарных операторов (базис). Тогда квантовая схема в базисе $\mathcal{A}$ — это последовательность $U_1[A_1], \ldots U_l[A_l]$, где A_i — множества q-битов, $U_i \in \mathcal{A}$.*

Оператор, реализуемый квантовой схемой. *Это оператор $U : \mathbb{B}^{\otimes n} \to \mathbb{B}^{\otimes n}$ (где $\mathbb{B} \stackrel{def}{=} \mathbb{C}^2$ с выделенным базисом $|0\rangle, |1\rangle$; n — общее число q-бит), равный $U_l[A_l] \ldots U_1[A_1]$.*

Это определение не очень хорошо, так не учитывает возможность использования дополнительной памяти в процессе вычисления. Поэтому дадим ещё одно определение.

Оператор $U : \mathbb{B}^{\otimes n} \to \mathbb{B}^{\otimes n}$, реализуемый схемой в расширенном смысле. *Произведение $W = U_l[A_l] \ldots U_1[A_1]$ (действующее на бóльшее, чем n, число q-битов, обозначим его N) удовлетворяет условию $W(|\xi\rangle \otimes |0^{N-n}\rangle) = (U|\xi\rangle) \otimes |0^{N-n}\rangle$ для любого вектора $|\xi\rangle$.*

Последнее определение требует некоторых пояснений. Если подействовать унитарным оператором на вектор вида $|\xi\rangle \otimes |\eta\rangle$, то получится вектор, который не обязательно разложим в произведение $|\xi\rangle' \otimes |\eta\rangle'$. Изометричного отображения пространства бо́льшей размерности на пространство меньшей размерности нет. Условие, которое используется в определении реализации оператора схемой в расширенном смысле, гарантирует, что результат действия W будет разложим по крайней мере для одного вектора в правом сомножителе (в принципе, неважно, какие именно значения принимают q-биты на этом векторе).

Разложимость вектора состояния системы означает, что две её подсистемы находятся в некотором определённом состоянии. При этом можно отсоединить эти две подсистемы и в дальнейшем они смогут существовать порознь. Поэтому условие, сформулированное в определении, можно пересказать нестрогим образом так: мы подключаем на время вычислений дополнительную память (её состояние определено однозначно), а в конце вычисления можем отсоединить дополнительную память, не причинив ущерба состоянию q-битов ответа.

Если вектор состояния неразложим, то отсоединение систем приводит к искажению их состояний. «Нельзя отсоединить квантовый принтер от квантового компьютера, пока идёт работа, иначе один или оба прибора испортятся» (замечание А. Шеня).

3. Соотношение между классическим и квантовым вычислением

Классический объект, соответствующий унитарному оператору — перестановка. Любой перестановке $G: \{0,1\}^k \to \{0,1\}^k$ естественно сопоставляется унитарный оператор $\hat{G}$ в пространстве $\left(\mathbb{C}^2\right)^k$, действующий по правилу $\hat{G}|x\rangle \stackrel{\text{def}}{=} |Gx\rangle$.

Аналогично определению 1 можно определить обратимые классические схемы, реализующие перестановки.

Определение 2. ***Обратимая классическая схема.*** *Пусть $\mathcal{A}$ — некоторое множество перестановок $\{0,1\}^n \to \{0,1\}^n$. Обратимая классическая схема в базисе $\mathcal{A}$ — это последовательность перестановок $U_1[A_1], \dots U_l[A_l]$, где A_i — множества битов, $U_i \in \mathcal{A}$.*

Перестановка, реализуемая обратимой схемой. *Это композиция перестановок $U_l[A_l] \dots U_1[A_1]$.*

Перестановка U, реализуемая в расширенном смысле. *Произведение перестановок $W = U_l[A_l] \dots U_1[A_1]$ (действующее на бо́льшее, чем n, число битов, обозначим его N) удовлетворяет условию $W(x, 0^{N-n}) = \left(Ux, 0^{N-n}\right)$ для любого $x \in \{0,1\}^n$.*

В каких случаях функцию, заданную булевой схемой, можно реализовать обратимой схемой? Обратимые схемы реализуют только перестановки. Преодолеть эту трудность можно так. Вместо вычисления функции $F: \{0,1\}^n \to \{0,1\}^m$ будем вычислять функцию $F_\oplus: \{0,1\}^{n+m} \to \{0,1\}^{n+m}$, заданную соотношением $F_\oplus(x,y) = (x, y \oplus F(x))$ (здесь $\oplus$ означает сложение по модулю 2). Тогда значение $F(x)$ можно получить так: $F_\oplus(x,0) = (x, F(x))$.

Чтобы можно было вычислять функции, заданные булевыми схемами в полном базисе, недостаточно взять базис для обратимых схем из перестановок на двух битах. Оказывается, что любая перестановка на двух битах $g: \{0,1\}^2 \to \{0,1\}^2$ является линейной функцией: $g(x,y) = (ax \oplus by \oplus c, dx \oplus ey \oplus f)$, где $a,b,c,d,e,f \in GF(2)$. Поэтому все функции, вычисляемые обратимыми схемами в базисе из перестановок на двух битах, являются линейными.

А вот перестановок на трёх битах уже достаточно, чтобы реализовать любые функции. Более того, оказывается, что достаточно включить в базис одну лишь функцию $\wedge_\oplus : (x,y,z) \to (x, y, z \oplus xy)$ (она называется элементом Тоффоли), чтобы можно было реализовывать любую функцию. При этом имеется в виду реализуемость в расширенном смысле, т. е. можно брать напрокат биты в состоянии 0 и возвращать из после окончания вычислений в том же состоянии.

Лемма 1. *Пусть $F : \mathbb{B}^n \to \mathbb{B}^m$, реализуемая булевой схемой размера L в некотором базисе $\mathcal{A}$. Тогда можно реализовать функцию $(x, 0) \mapsto (F(x), G(x))$ обратимой схемой размера $O(L)$ в базисе $\mathcal{A}_\oplus$, состоящем из функций $f_\oplus$ $(f \in \mathcal{A})$, а также функции $\neg_\oplus : (x, y) \mapsto (x, x \oplus y)$.* (Заметим, что перестановки битов выражаются через $f_\oplus$).

$G(x)$ — это “мусор”, образованный при вычислении.

Доказательство. Возьмём схему, вычисляющую F. Пусть $x_1, \ldots, x_n$ — входные переменные; $x_{n+1}, \ldots, x_{n+L}$ — вспомогательные переменные схемы и биты результата. Сопоставим им дополнительные биты (в начальном состоянии все они имеют значение 0).

Каждое присваивание в схеме имеет вид $x_{n+k} := f_k(x_{i_k}, \ldots, x_{j_k})$, $f_k \in \mathcal{A}$, $i_k, \ldots, j_k < n+k$. В обратимой схеме вместо f_k будем использовать $(f_k)_\oplus$: $(x_{i_k}, \ldots, x_{j_k}, x_{n+k}) := (f_k)_\oplus(x_{i_k}, \ldots, x_{j_k}, x_{n+k})$, то есть $x_{n+k} := x_{n+k} \oplus f_k(x_{i_k}, \ldots, x_{j_k})$.

Поскольку начальные значения дополнительных переменных были равны 0, их конечные значения будут такими же, как и в булевой схеме.

Осталось поменять местами биты, чтобы получить указанный в условии формат ответа. Для этого можно использовать функцию $\neg_\oplus$, а именно, $[i \leftrightarrow j] = \neg_\oplus[i, j]\, \neg_\oplus[j, i]\, \neg_\oplus[i, j]$.

Весь процесс вычисления удобно представить следующей схемой (над прямоугольниками подписано количество бит, внутри — их содержимое):

n	$L-m$	m
x	0	0

присваивания по схеме

x	x_{n+1} $\ldots$ x_{L-m}	$F(x)$

перестановки бит

$F(x)$	$G(x)$

□

Замечание. Если базис $\mathcal{A}$ — стандартный полный базис $\{\neg, \wedge\}$, то $\neg_\oplus$ выражается через элемент Тоффоли $\wedge_\oplus$.

Лемма 2 (Очистка мусора). *В условиях леммы 1 можно произвести вычисление функции $F_\oplus$ обратимой схемой размера $O(L + n + m)$ (с привлечением дополнительных битов).*

Доказательство. Для очистки мусора будет использована обратимость. Изобразим процесс вычисления $F_\oplus$ схемой, аналогичной той, что приведена в доказательстве леммы 1.

n	L	m
x	0	y

вычисление по схеме из доказательства леммы 1

m	$L+n-m$	m
$F(x)$	$G(x)$	y

сложение $F(x)$ и y по модулю 2

$F(x)$	$G(x)$	$F(x) \oplus y$

обращение вычислений, сделанных на первом шаге

x	0	$F(x) \oplus y$

□

Замечание. Обратимыми вычислениями заинтересовались при попытке ответить на вопрос, какая энергия необходима для вычислений (классических). Анализ показал, что потери энергии можно устремить к нулю для всех вычислительных операций, кроме необратимых. Когда производится необратимая операция, (например, стирание бита) два различных логических значения (0 и 1) становятся одинаковыми (0). Однако физические законы на микроуровне являются обратимыми, поэтому отличие между между старыми состояниями (0 и 1) должно сохраниться в каких-то неконтролируемых физических степенях свободы. Это можно интерпретировать как возрастание беспорядка (энтропии), которое в конечном счете проявится в окружающей среде в виде тепла. Величина энергии, требуемой для стирания одного бита, очень мала ($\approx kT$), но конечна. Потеря энергии из-за необратимого стирания информации при форматировании жёсткого диска ёмкостью 1 Гб равна 10^{-13} Дж, что примерно соответствует затратам энергии на сдвиг головки диска на половину диаметра атома водорода. Это на много порядков меньше реального перемещения головки при форматировании.

С другой стороны, если ёмкость дисков будет расти столь же быстро, как в настоящее время, то к концу XXIII века для форматирования жёсткого диска потребуется энергия, соответствующая годовому излучению Солнца.

Лемма об очистке мусора показывает, что можно избежать таких потерь энергии, связанных с необратимостью вычислений.

Можно также показать, что любое вычисление, требующее памяти L, можно реализовать обратимым образом с использованием памяти, не превышающей $L^{O(1)}$. Приведём набросок доказательства.

Поскольку задача TQBF $\mathcal{PSPACE}$-полна, то достаточно вычислять на небольшой памяти значение формулы

$$\exists x_1 \forall y_1 \ldots \exists x_M \forall y_M f(x_1, y_1, \ldots, x_M, y_M, z)$$

при условии, что $f(\cdot)$ вычислима булевой схемой небольшого размера L. Покажем, что в этом случае приведенная выше QBF вычислима на памяти $O(L+M)$. Вычисление будет организовано рекурсивно, начиная с внутренних кванторов.

Чтобы вычислить $\forall x\, F(x,z)$, вычислим $F(0,z)$, занесём результат в одну дополнительную ячейку, затем вычислим $F(1,z)$ и занесём результат в другую дополнительную ячейку. После вычислим $\forall x\, F(x,z) = F(0,z) \wedge F(1,z)$ и результат занесём в третью ячейку. Чтобы убрать мусор, прокрутим все вычисления, кроме последнего шага, в обратном направлении.

Разобравшись аналогичным образом с формулой $\exists x\, F(x,z)$, приходим к такому выводу: добавление квантора по булевой переменной увеличивает требуемую память не более чем на константу бит.

В заключение сформулируем теорему о вычислении обратимых функций, которая является прямым обобщением леммы 2.

Теорема. *Пусть F и F^{-1} вычислимы булевыми схемами размеров $\leq L$. Тогда F реализуется обратимой схемой размера $O(L+n)$.*

Доказательство. Даётся следующей схемой вычислений (для простоты схемы биты, которые в начале и конце вычисления равны 0, не указаны).

n	n	
x	0	
		вычисление $F_\oplus$ по схеме из доказательства леммы 2
x	$F(x)$	
		перестановки битов
$F(x)$	x	
		применение $(F^{-1})_\oplus$ (по схеме из доказательства леммы 2) даёт в правом регистре $x \oplus F^{-1}(F(x)) = 0$
$F(x)$	0	

$\square$