Классические и квантовые вычисления

А. Китаев; написал М. Вялый

9 лекция (09/04/98)

1. Квантовые алгоритмы

Мы закончили с формальным введением и теперь перейдем к построению алгоритмов, существенно использующих квантовые свойства введенной модели вычислений.

Задача. Имеется число q, записывающееся не более, чем n битами ($1 \leq q < 2^n$) и число a такое, что $a < q$, $\text{нод(a,q)} = 1$ (здесь нод обозначает наибольший общий делитель). Нужно найти *период* a относительно q, т. е. такое наименьшее неотрицательное число t, что $a^t \equiv 1 \pmod{q}$.

Период числа будем обозначать $\mathrm{per}_q(a)$.

Вначале несколько комментариев по поводу самой задачи. Вычеты, взаимно простые с q, образуют группу. Это почти очевидно; фактически, нужно доказывать только существование обратного элемента. Обратный элемент можно найти, пользуясь вариантом алгоритма Евклида, который ищет решение уравнения $xa + yq = 1$ в целых числах: из этого уравнения следует, что $x = a^{-1} \pmod{q}$. Напомним также, что алгоритм Евклида полиномиален, как и возведение чисел в степень (время ограничено степенью от длины входа). Поскольку группа вычетов, взаимно простых с q, конечна, каждый ее элемент имеет конечный порядок (это и есть период числа относительно q). Более того, порядок любого элемента не превосходит q.

Можно показать, что с классической точки зрения задач определения периода эквивалентна по сложности задаче факторизации целых чисел. Последняя задача считается сложной настолько, что на предположении о ее сложности основываются алгоритмы криптографии, используемые на практике.

Мы построим квантовый алгоритм для решения задачи о нахождении периода числа.

Основная идея алгоритма

Рассмотрим оператор $U_a : |x\rangle \mapsto |ax \pmod{q}\rangle$ (оператор умножения вычета на a, $\text{нод(a,q)} = 1$). Этот оператор переставляет базисные векторы при $0 \leq x < q$, будем считать, что на остальные базисные векторы он действует тождественно, $U_a : |x\rangle \mapsto |x\rangle$, при $x \geq q$.

Поскольку для умножения вычетов есть обычная булева схема полиномиального — $O(n^2)$ — размера, то существует и квантовая схема примерно такого же размера (использующая напрокат дополнительные регистры, как это объяснялось раньше).

Перестановка, которую задает оператор U_a, разбивается на циклы. Цикл, содержащий a, содержит и 1 (после $\mathrm{per}_q(a) - 1$ шагов мы попадаем из a в 1). Алгоритм, о котором пойдет речь, начинает с состояния $|1\rangle$ и применяет нему оператор U_a по многу раз. Но за пределы орбиты a (цикла перестановки, которому принадлежит a) мы такими преобразованиями не выйдем. Поэтому рассмотрим ограничение оператора U_a на подпространство, порожденное орбитой a.

Собственные числа для U_a : $\lambda_k = e^{2\pi \mathrm{i} \cdot k/t}$, где t — период.

$$\text{Собственные вектора для } U_a: \quad |\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{m=0}^{t-1} e^{-2\pi \mathrm{i} \frac{km}{t}} |a^m\rangle.$$

Проверить, что написанные вектора действительно собственные, легко. Достаточно заметить, что умножение на a приводит к сдвигу индексов в сумме. Если заменить переменную суммирования, чтобы устранить этот сдвиг, получим множитель $e^{2\pi\mathrm{i}\cdot k/t}$.

Если бы мы могли измерять собственные числа оператора U_a, то получали бы числа k/t. Сначала разберем, как это может нам помочь в нахождении периода.

Пусть у нас есть машина M, которая при каждом запуске выдает нам число k/t, где t — искомый период, а k — равномерно распределенное на множестве $\{1,\ldots,t-1\}$ случайное число. Мы предполагаем, что k/t представлено в виде несократимой дроби k'/t' (если бы машина выдавала число в виде k/t, то вообще не было бы проблем).

Получив несколько дробей такого вида k'_1/t'_1, $k'_2/t'_2,\ldots,k'_l/t'_l$, можно с большой вероятностью найти число t, приводя эти дроби к общему знаменателю.

Лемма. *Если получено l дробей, то вероятность того, что наименьшее общее кратное их знаменателей отлично от t меньше $3\cdot 2^{-l}$.*

Доказательство. Дроби $k'_1/t'_1,\ldots,k'_l/t'_l$ получаются сокращением дробей $k_1/t,\ldots,k_l/t$ (т. е. $k'_j/t'_j = k_j/t$), где $k_1,\ldots,k_l$ — независимо распределенные случайные числа. Достаточно, чтобы эти числа были в совокупности взаимно просты, тогда наименьшее общее кратное $t'_1,\ldots,t'_l$ будет равно t.

Вероятность того, что $k_1,\ldots,k_l$ имеют общий простой делитель p, не больше, чем $1/p^l$. Поэтому вероятность получить не t после приведения к общему знаменателю не превосходит $\sum\limits_{k=2}^{\infty}\dfrac{1}{k^l} < 3\cdot 2^{-l}$ (эта сумма заведомо включает в себя все простые, меньшие t). □

Теперь будем строить машину M. Она должна содержать схему, измеряющую собственные числа оператора U_a. Точнее говоря, нам нужен оператор $U_* : |q,a,x\rangle \mapsto |q,a,ax \pmod q\rangle$, если $\mathrm{нод(a,q)} = 1$. Как оператор U_* действует в остальных случаях, неважно. Его можно доопределить любым вычислительно тривиальным способом. На самом деле, все приведенные ранее рассуждения об имитации классических схем квантовыми сохраняют силу и для имитации схем, вычисляющих частично определенные функции.

Обозначим $\mathcal{L}_k = (|\xi_k\rangle)$ (подпространство, порожденное $|\xi_k\rangle$), тогда искомая схема должна вычислять измеряющий оператор $W = \sum\limits_{k=0}^{t-1} V_k \otimes \Pi_{\mathcal{L}_k}$ с операторами V_k вида $|0\rangle \mapsto \sum_{y,z} c_{y,z}|y,z\rangle$, где y — некоторая несократимая дробь, а z — мусор. (Строго говоря, мы должны написать $W = \sum\limits_{k=0}^{t-1} V_{q,a,k}\otimes \Pi_{\mathcal{L}_{q,a,k}}$. Однако q и a на протяжении всего вычисления не меняются, поэтому их можно опускать в индексах). При этом для условных вероятностей должно выполняться неравенство

$$\mathbf{P}\left(\boxed{\frac{k}{t}}\,\middle|\, k\right) \overset{\mathrm{def}}{=} \sum_z \left|\left\langle \boxed{\frac{k}{t}}, z \,\middle|\, V_k \,\middle|\, 0\right\rangle\right|^2 \;>\; 1-\varepsilon,$$

где $\boxed{\frac{k}{t}}$ обозначает несократимую дробь, представляющую рациональное число k/t.

Построение такой измеряющей схемы довольно сложное, поэтому вначале объясним, как из нее строится машина M. Возьмем состояние $|1\rangle$ в качестве начального. Прямое вычисление (рекомендуется его проделать) показывает, что $|1\rangle = \dfrac{1}{\sqrt{t}}\sum\limits_{k=0}^{t-1}|\xi_k\rangle$. Это равенство гарантирует равномерное распределение числителей дробей. Проведем измерение в этом состоянии, тогда по формуле полной вероятности получаем

$$\mathbf{P}\Big(W(|0\rangle\otimes|1\rangle),y\Big) \;=\; \sum_k \mathbf{P}(y\big|k)\,\mathbf{P}(|1\rangle,\mathcal{L}_k).$$

Вероятности всех $|\xi_k\rangle$ равны: $\mathbf{P}(|1\rangle,\mathcal{L}_k)=|\langle\xi_k|\xi_k\rangle|^2=1/t$, а указанное выше свойство условных вероятностей гарантирует нам, что с вероятностью $1-\varepsilon$ будет получаться $\boxed{\frac{k}{t}}$. Как будет видно в дальнейшем, при построении W можно сделать ε сколь угодно малой.

Условно работу машины M можно представить в виде такого процесса:

$$|1\rangle \longrightarrow \boxed{\begin{array}{c}\text{случайный выбор } k\\ \text{(бог, играющий в кости)}\end{array}} \xrightarrow{|\xi_k\rangle} \boxed{\text{измерение } W} \longrightarrow \begin{cases} y\neq\boxed{\frac{k}{t}}\ \Big\} < \varepsilon \\[2ex] \boxed{\frac{k}{t}}\ \Big\} > 1-\varepsilon \end{cases}$$

Построение измеряющего оператора

Теперь будем строить измеряющий собственные числа U_a оператор. В силу приведенных выше соображений можно ограничиться изучением действия этого оператора на вход $|\xi_k\rangle$. Действовать будем в три приема.

1) Ищем информацию о $\lambda_k=e^{2\pi\mathrm{i}\varphi_k}$, $\varphi_k=\frac{k}{t}\pmod 1$.

2) Локализуем значение φ с небольшой точностью. Самое время подчеркнуть, что во всех приводимых рассуждениях есть два параметра: вероятность ошибки ε и точность δ. Мы получаем некоторое число как результат измерения и должно выполняться условие $\mathrm{Prob}[|z-\varphi_k|>\delta]<\varepsilon$. На данном шаге нас устроит небольшая точность, скажем $\delta=1/8$.

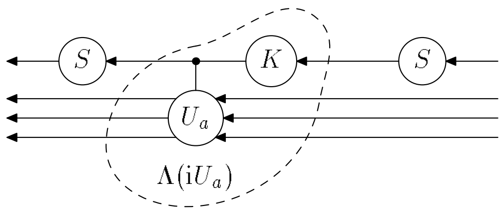
3) Далее нужно увеличить точность. Необходимо уметь отличать друг от друга числа вида $\varphi=k/t$, где $0\le k<t<2^n$. Заметим, что если $k_1/t_1\neq k_2/t_2$, то $\left|\frac{k_1}{t_1}-\frac{k_2}{t_2}\right|>\frac{1}{t_1t_2}>\frac{1}{2^{2n}}$. Поэтому, зная значение $\varphi_k=k/t$ с точностью $1/2^{2n+1}$, мы можем определить его абсолютно точно (в виде несократимой дроби). Чтобы делать это эффективно (за полиномиальное время), можно использовать алгоритм цепных дробей.

1) Как получать информацию о собственном числе

В прошлый раз был введен оператор $\Xi(U_a)=(S\otimes\mathrm{id})\Lambda(U_a)(S\otimes\mathrm{id})$, где $S=\dfrac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}$. Сложность вычисления $\Lambda(U_a)$ почти такая же, как и U_a (мы уже рассматривали реализацию такого оператора). Оператор $\Xi(U_a)$ — измеряющий, $\Xi(U_a)=\sum\limits_k V_{k,a}\otimes\Pi_{\mathcal{L}_k}$, $V_{k,a}=\frac12\begin{pmatrix}1+\lambda_k&1-\lambda_k\\1-\lambda_k&1+\lambda_k\end{pmatrix}$. Запишем действие этого оператора

$$\boxed{0\;\big|\;|\xi_k\rangle} \xrightarrow{\Xi(U_a)} \boxed{\frac{1+\lambda_k}{2}|0\rangle+\frac{1-\lambda_k}{2}|1\rangle\;\Big|\;|\xi_k\rangle}$$

Т. е. $\mathbf{P}(0\big|k)=\left|\dfrac{1+\lambda_k}{2}\right|^2=\dfrac{1+\cos(2\pi\varphi_k)}{2}$ (напомним, что $\lambda_k=e^{2\pi\mathrm{i}\varphi_k}$).



Нам потребуется еще и оператор $\Xi(\mathrm{i}U_a)$. Его также нетрудно реализовать. Реализация, изображенная на врезке, использует оператор $K = \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{i} \end{pmatrix}$, который также был включен в число базисных в стандартном базисе. Обведенный фрагмент реализует оператор $\Lambda(\mathrm{i}U_a)$. Действительно, K умножает на i только $|1\rangle$, но как раз в этом случае применяется оператор U_a (по определению оператора $\Lambda(U_a)$). Для оператора $\Xi(\mathrm{i}U_a)$ условные вероятности равны $\mathbf{P}(0|k) = \dfrac{1-\sin(2\pi\varphi_k)}{2}$.

2) Как оценивать вероятности

Чтобы оценить вероятности, которые приведены в предыдущем пункте, мы будем применять операторы $\Xi(U_a)$ и $\Xi(\mathrm{i}U_a)$ к различным “приборам” (дополнительным q-битам). Рассуждения одинаковы для обоих операторов, поэтому ограничимся случаем $\Xi(U_a)$.

У нас есть “квантовый” регистр A, в котором находится $|\xi_k\rangle$ (на самом деле, там вначале был $|1\rangle$, а потом при действиях многочисленных операторов вида $\Xi(U_a)$ получается некоторое запутанное (entangled) состояние, но мы рассматриваем $|\xi_k\rangle$ по отдельности; это корректно в силу вида измеряющего оператора). Заведем большое количество s вспомогательных регистров длиной в 1 бит. Каждый из этих регистров будет использоваться для применения оператора $\Xi(U_a)$.

Как было доказано на предыдущей лекции, условные вероятности в таком случае перемножаются. Для оператора $\prod\limits_{r=1}^{s}\Xi(U_a)[r,A]$ условные вероятности будут равны $\mathbf{P}(y_1,\ldots,y_s|k) = \prod\limits_{r=1}^{s}\mathbf{P}(y_r|k)$, здесь через y_r обозначено значение в r-том бите.

Дальнейшие действия будут классическими с битами, в которых записаны результаты “экспериментов”. Поскольку условные вероятности перемножаются, можно считать, что у нас есть “кривая” монета, и мы оцениваем вероятность выпадения 1 после серии испытаний Бернулли.

Если монета брошена s раз, то оценкой $\mathbf{P}(1|k)$ считаем частоту выпадения 1 в эксперименте: $(\sum y_r)/s$. С какой точностью верна такая оценка? Из теории вероятностей известно, что при заданном δ выполнено $\mathrm{Prob}\left[\left|\dfrac{\sum_{r=1}^{s} y_r}{s} - \mathbf{P}(1|k)\right|\right] < 2e^{-c(\delta)s}$, при некотором $c(\delta) > 0$. Это показывает, что при любом фиксированном δ можно добиться вероятности ошибки ε за $O(|\log\varepsilon|)$ испытаний.

Итак, мы научились находить с некоторой точностью δ синус и косинус от φ_k. Теперь подберем δ таким, чтобы значение φ_k можно было установить по значениям синуса и косинуса с точностью 1/8. Этим мы достигаем второй из поставленных вначале целей.

3) Экспоненциально точное определение собственных чисел

Для увеличения точности нам потребуется использовать одну особенность рассматриваемого оператора, а именно $(U_a)^b : |x\rangle \mapsto |a^b x\rangle = U_{a^b}|x\rangle$. Числа мы можем быстро возводить в степень, а операторы, вообще говоря, — нет. Но оператор умножения на число U_a обладает тем приятным свойством, что его степени могут быть получены эффективно (схемой полиномиального размера). Для этого нужно возвести в степень число a, а потом использовать полученный результат как параметр для построения оператора U_{a^b}. (Напомним, что в действительности мы имеем дело с оператором U_*, который содержит в себе U_d для всех d сразу).

Вернемся к схеме предыдущего пункта. Мы находили собственное число $\lambda_k = e^{2\pi\mathrm{i}\varphi_k}$ для некоторого собственного вектора $|\xi_k\rangle$. Этот же вектор останется собственным и для любой степени оператора U_a, поэтому можно одновременно (на одном и том же квантовом регистре) искать собственное число для $U_a^2 = U_{a^2}$, оно равно $\lambda_k = e^{2\pi \mathrm{i} 2\varphi_k}$; для $U_a^4 = U_{a^4}$ оно равно $\lambda_k = e^{2\pi \mathrm{i} 4\varphi_k}$; ...

Другими словами, мы можем с точностью $1/8$ определить значения $\varphi_k, 2\varphi_k, \ldots, 2^{2n}\varphi_k$ по модулю 1. Но это позволяет определить φ_k с точностью $1/2^{2n+1}$, причем соответствующее вычисление можно произвести за полиномиальное время.

Идея доказательства. Множество возможных значений φ_k удобно представлять в виде окружности единичной длины. Зная φ_k с точностью $1/8$, мы выделяем дугу в $1/4$ от всей окружности. Знание $2\varphi_k$ с точностью $1/8$ позволяет выделить две дуги длиной $1/8$ каждая, причем только одна из них имеет непустое пересечение с предыдущей дугой.

Важные замечания

1) Существенно, что вектор $|\xi_k\rangle$ не портится во время вычислений.

2) Все вычисление в целом зависит от параметров l и s. Суммарная вероятность ошибки не превышает $3 \cdot 2^{-l} + 4nle^{-cs}$, где $c = \mathrm{const}$. Если требуется получить ответ с вероятностью ошибки $< \epsilon$, следует положить $l = c_1 \log(1/\epsilon)$, $s = c_2 \log(n/\epsilon)$, где c_1 и c_2 — некоторые константы. При этом получится квантовая схема размера $O(n^3 \log(1/\epsilon) \log(n/\epsilon))$.

<u>Вход:</u> a и q

- Вычисление степеней a^{2^j} $(j = 1, \ldots, 2n)$ по модулю q (классическое).
- Создание l штук квантовых регистров, содержащих базисное состояние $|1\rangle$.

$$\boxed{|1\rangle} \qquad \boxed{|1\rangle} \qquad \ldots \qquad \boxed{|1\rangle}$$

$$\begin{array}{|c|c|c|}\multicolumn{3}{c}{2s}\\ \hline 0 & \ldots & 0 \\ \hline 0 & \ldots & 0 \\ \hline \end{array} \quad \ldots \quad \begin{array}{|c|c|c|}\multicolumn{3}{c}{2s}\\ \hline 0 & \ldots & 0 \\ \hline 0 & \ldots & 0 \\ \hline \end{array}$$

$4ns$ дополнительных q-битов (“приборов”)

- Квантовые измерения при помощи операторов $\Xi(U_{a^{2^j}})$ и $\Xi(\mathrm{i}U_{a^{2^j}})$.

$$\begin{array}{|c|c|c|}\hline y_1^{(1)} & \ldots & y_s^{(1)} \\ \hline \tilde{y}_1^{(1)} & \ldots & \tilde{y}_s^{(1)} \\ \hline \end{array} \quad \ldots \quad \begin{array}{|c|c|c|}\hline y_1^{(2n)} & \ldots & y_s^{(2n)} \\ \hline \tilde{y}_1^{(2n)} & \ldots & \tilde{y}_s^{(2n)} \\ \hline \end{array}$$

результаты измерений — нули и единицы

- Подсчет числа нулей и единиц (классический).

$$\begin{array}{ccc} \cos\varphi \pm \delta & & \cos(2^{2n}\varphi) \pm \delta \\ & \ldots & \\ \sin\varphi \pm \delta & & \sin(2^{2n}\varphi) \pm \delta \end{array}$$

- Тригонометрические вычисления (классические).

$$\varphi \pm \tfrac{1}{8} \qquad \ldots \qquad 2^{2n}\varphi \pm \tfrac{1}{8}$$

- Уточнение значения φ по набору чисел $\varphi, \ldots, 2^{2n}\varphi$ (классическое).

$$\varphi \pm 2^{-(2n+1)}$$

- Определение точного значения φ при помощи цепных дробей (классическое).

k'/t' — несократимая дробь, равная φ

$$\frac{k'_1}{t'_1} \qquad \frac{k'_2}{t'_2} \qquad \ldots \qquad \frac{k'_l}{t'_l}$$

- Вычисление наибольшего общего знаменателя (классическое).

<u>Ответ:</u> t (с вероятностью ошибки $< 3 \cdot 2^{-l} + 4nle^{-cs}$, где $c = \mathrm{const}$)

Сх. 1. Общая картина вычислений по описанному алгоритму