Квантовые алгоритмы: возможности и ограничения. Лекция 8: Факторизация чисел

М. Вялый

Вычислительный центр
им. А.А.Дородницына
Российской Академии наук

Санкт-Петербург, 2011

План

Основная схема (схема для косинуса)

U — унитарный оператор, $|\psi\rangle$ — его собственный вектор с собственным числом $\lambda = \exp(2\pi i\varphi)$:

$$U|\psi\rangle = \lambda|\psi\rangle.$$

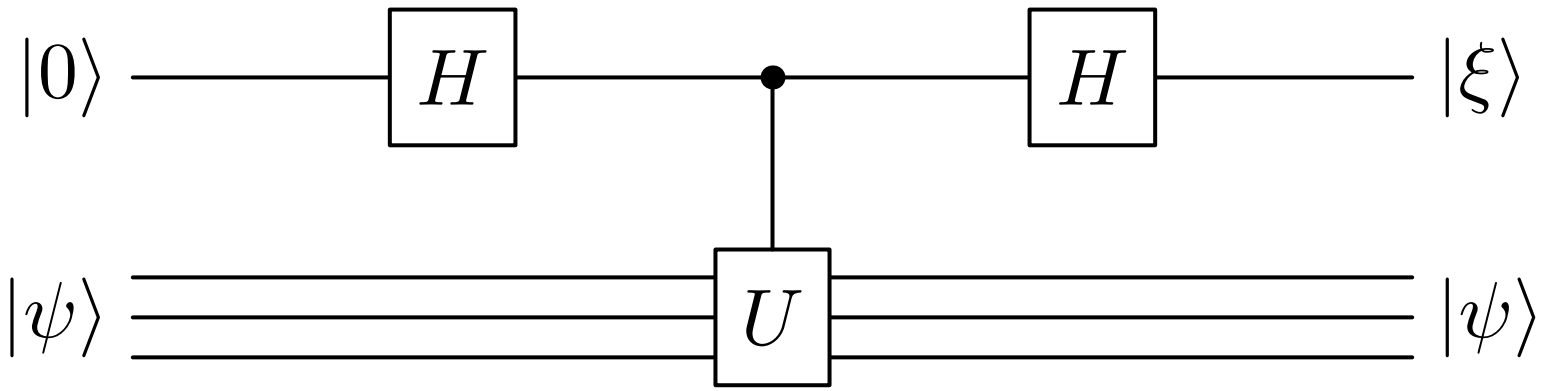
Как определить собственное число? Рассмотрим такую схему:

Основная схема (схема для косинуса)

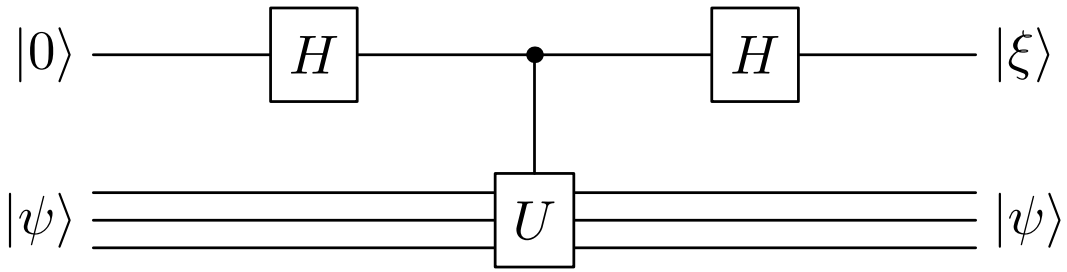
U — унитарный оператор, $|\psi\rangle$ — его собственный вектор с собственным числом $\lambda = \exp(2\pi i\varphi)$:

$$U|\psi\rangle = \lambda|\psi\rangle.$$

Как определить собственное число? Рассмотрим такую схему:



Действие на управляющем кубите

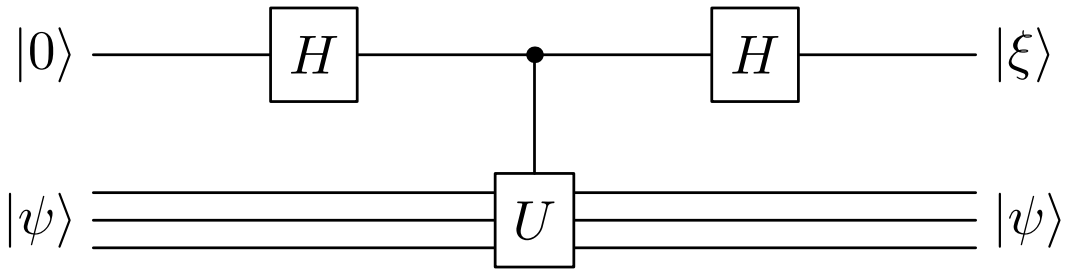


$$|\xi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & \lambda\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}|0\rangle =$$

$$= \frac{1}{2}\begin{pmatrix}1+\lambda & 1-\lambda\\ 1-\lambda & 1+\lambda\end{pmatrix}|0\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\ 1-\lambda\end{pmatrix}.$$

*) $U|\psi\rangle = \lambda|\psi\rangle$.

Действие на управляющем кубите

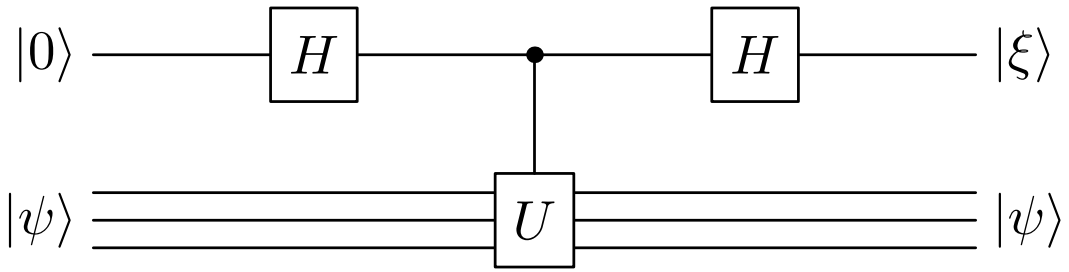


$$|\xi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & \lambda\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}|0\rangle =$$

$$= \frac{1}{2}\begin{pmatrix}1+\lambda & 1-\lambda\\ 1-\lambda & 1+\lambda\end{pmatrix}|0\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\ 1-\lambda\end{pmatrix}.$$

*) $U|\psi\rangle = \lambda|\psi\rangle$.

Действие на управляющем кубите



$$|\xi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & \lambda\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}|0\rangle =$$

$$= \frac{1}{2}\begin{pmatrix}1+\lambda & 1-\lambda\\ 1-\lambda & 1+\lambda\end{pmatrix}|0\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\ 1-\lambda\end{pmatrix}.$$

*) $U|\psi\rangle = \lambda|\psi\rangle$.

Измерение управляющего кубита

Вероятность исхода 0:

$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1+\lambda}{2}\right|^2 = \frac{1}{4}\left((1+\cos(2\pi\varphi))^2 + \sin(2\pi\varphi)^2\right) =$$

$$= \frac{1+\cos(2\pi\varphi)}{2}$$

- $|\xi\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\1-\lambda\end{pmatrix}$;
- $\lambda = \exp(2\pi i\varphi)$;
- раскрываем скобки;
- приводим подобные.

Измерение управляющего кубита

Вероятность исхода 0:

$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1+\lambda}{2}\right|^2 = \frac{1}{4}\left((1+\cos(2\pi\varphi))^2 + \sin(2\pi\varphi)^2\right) =$$

$$= \frac{1+\cos(2\pi\varphi)}{2}$$

- $|\xi\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\1-\lambda\end{pmatrix}$;
- $\lambda = \exp(2\pi i\varphi)$;
- раскрываем скобки;
- приводим подобные.

Измерение управляющего кубита

Вероятность исхода 0:

$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1+\lambda}{2}\right|^2 = \frac{1}{4}\left((1+\cos(2\pi\varphi))^2 + \sin(2\pi\varphi)^2\right) =$$

$$= \frac{1+\cos(2\pi\varphi)}{2}$$

- $|\xi\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\1-\lambda\end{pmatrix}$;
- $\lambda = \exp(2\pi i\varphi)$;
- раскрываем скобки;
- приводим подобные.

Измерение управляющего кубита

Вероятность исхода 0:

$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1+\lambda}{2}\right|^2 = \frac{1}{4}\left((1+\cos(2\pi\varphi))^2 + \sin(2\pi\varphi)^2\right) =$$

$$= \frac{1+\cos(2\pi\varphi)}{2}$$

- $|\xi\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda\\1-\lambda\end{pmatrix}$;
- $\lambda = \exp(2\pi i\varphi)$;
- раскрываем скобки;
- приводим подобные.

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из s управляющих кубитов.
- Поскольку состояние управляющих кубитов после применения основной схемы $|\xi\rangle^{\otimes s}$, результаты измерений независимы, вероятность 0 в каждом кубите равна $p = (1 + \cos(2\pi\varphi))/2$.
- Отношение числа нулей среди исходов измерения к s дает приближенное значение p.

Оценка Чернова

Пусть проведена серия из s испытаний Бернулли с вероятностью успеха p. Вероятность отклонения частоты $\nu = (\text{число успехов})/s$ от вероятности оценивается как

$$\mathbf{Pr}\left[|\nu - p| > \delta\right] < 2e^{-2\delta^2 s}.$$

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из s управляющих кубитов.
- Поскольку состояние управляющих кубитов после применения основной схемы $|\xi\rangle^{\otimes s}$, результаты измерений независимы, вероятность 0 в каждом кубите равна $p = (1 + \cos(2\pi\varphi))/2$.
- Отношение числа нулей среди исходов измерения к s дает приближенное значение p.

Оценка Чернова

Пусть проведена серия из s испытаний Бернулли с вероятностью успеха p. Вероятность отклонения частоты $\nu = (\text{число успехов})/s$ от вероятности оценивается как

$$\mathbf{Pr}\left[|\nu - p| > \delta\right] < 2e^{-2\delta^2 s}.$$

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из s управляющих кубитов.
- Поскольку состояние управляющих кубитов после применения основной схемы $|\xi\rangle^{\otimes s}$, результаты измерений независимы, вероятность 0 в каждом кубите равна $p = (1 + \cos(2\pi\varphi))/2$.
- Отношение числа нулей среди исходов измерения к s дает приближенное значение p.

Оценка Чернова

Пусть проведена серия из s испытаний Бернулли с вероятностью успеха p. Вероятность отклонения частоты $\nu = (\text{число успехов})/s$ от вероятности оценивается как

$$\mathbf{Pr}\left[|\nu - p| > \delta\right] < 2e^{-2\delta^2 s}.$$

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из s управляющих кубитов.
- Поскольку состояние управляющих кубитов после применения основной схемы $|\xi\rangle^{\otimes s}$, результаты измерений независимы, вероятность 0 в каждом кубите равна $p = (1 + \cos(2\pi\varphi))/2$.
- Отношение числа нулей среди исходов измерения к s дает приближенное значение p.

Оценка Чернова

Пусть проведена серия из s испытаний Бернулли с вероятностью успеха p. Вероятность отклонения частоты $\nu = (\text{число успехов})/s$ от вероятности оценивается как

$$\mathbf{Pr}\left[|\nu - p| > \delta\right] < 2e^{-2\delta^2 s}.$$

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из s управляющих кубитов.
- Поскольку состояние управляющих кубитов после применения основной схемы $|\xi\rangle^{\otimes s}$, результаты измерений независимы, вероятность 0 в каждом кубите равна $p = (1 + \cos(2\pi\varphi))/2$.
- Отношение числа нулей среди исходов измерения к s дает приближенное значение p.

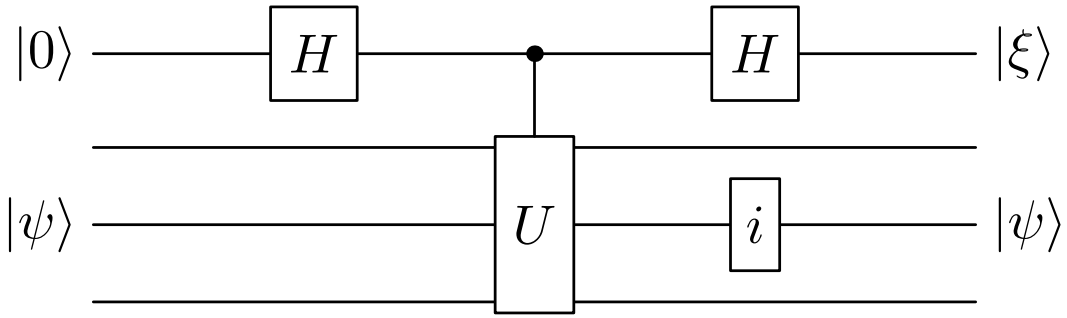
Оценка Чернова

Пусть проведена серия из s испытаний Бернулли с вероятностью успеха p. Вероятность отклонения частоты $\nu = (\text{число успехов})/s$ от вероятности оценивается как

$$\mathbf{Pr}\left[|\nu - p| > \delta\right] < 2e^{-2\delta^2 s}.$$

Схема для синуса

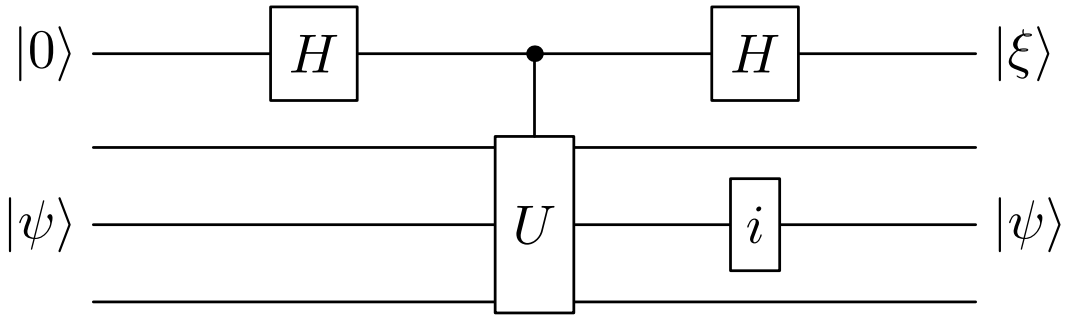
Основная схема позволяет оценивать косинус фазы φ. Для оценки синуса нужно поменять местами действительную и мнимую части λ:



$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1+i\lambda}{2}\right|^2 = \frac{1-\sin(2\pi\varphi)}{2}$$

Схема для синуса

Основная схема позволяет оценивать косинус фазы φ. Для оценки синуса нужно поменять местами действительную и мнимую части λ:



$$\mathbf{Pr}(|\xi\rangle, 0) = \left|\frac{1 + i\lambda}{2}\right|^2 = \frac{1 - \sin(2\pi\varphi)}{2}$$

Точность оценки фазы

Утверждение

Фаза φ собственного числа λ оценивается с точностью δ и вероятностью ошибки $< \varepsilon$ за $2s$ применений схем косинуса и синуса, где $s = O(\delta^{-2} \log(1/\varepsilon))$.

Вероятность ошибки уменьшается очень быстро. Но точность уменьшается медленно: оценка с точностью 2^{-n} требует экспоненциального времени (размера схемы).

Точность оценки фазы

Утверждение

Фаза φ собственного числа λ оценивается с точностью δ и вероятностью ошибки $< \varepsilon$ за $2s$ применений схем косинуса и синуса, где $s = O(\delta^{-2} \log(1/\varepsilon))$.

Вероятность ошибки уменьшается очень быстро. Но точность уменьшается медленно: оценка с точностью 2^{-n} требует экспоненциального времени (размера схемы).

Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения
$2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.

Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения
$2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.

Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения
$2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.

Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения
$2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.

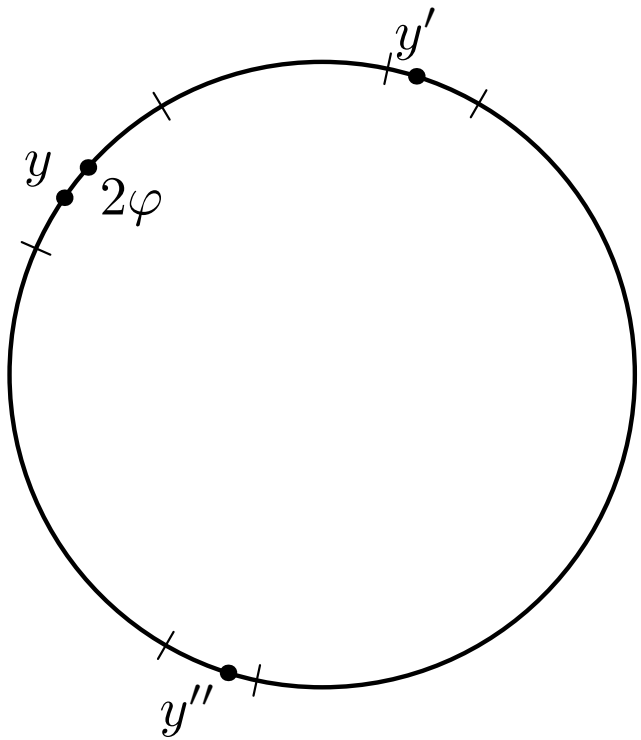
Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения $2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.



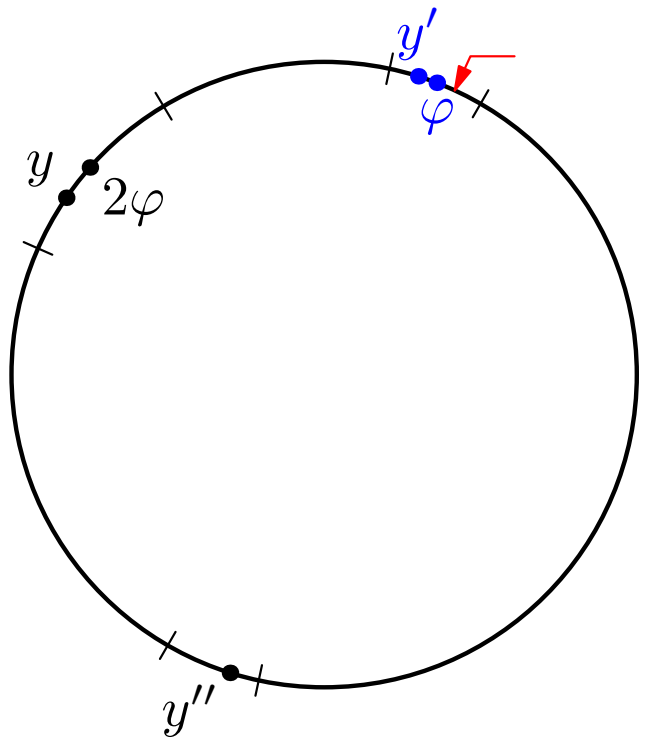
Уточнение значения: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < \pi/8$ фазы $\varphi_k = 2^k\varphi$ степеней U^{2^k} при $k = 0, \ldots, n$.
- Утверждение. По этим данным можно оценить фазу $\varphi = \varphi_1$ с точностью $\pi/2^{n+3}$.

Лемма

Если $|y - 2\varphi| < \delta < \pi$, то
либо $|y' - \varphi| < \delta/2$,
либо $|y'' - \varphi| < \delta/2$,
где y', y'' — решения уравнения
$2x \equiv y \pmod{2\pi}$.

Одно из двух решений выбирается исходя из δ-приближения φ.



Работа алгоритма на произвольном входе

Вопрос

Что будет, если на ход алгоритма оценки фазы подать не собственный вектор?

- Пусть $|\psi\rangle = \sum_{k=1}^{N} c_k |\psi_k\rangle$, где $|\psi_k\rangle$ — собственные вектора оператора U единичной длины: $U|\psi_k\rangle = \exp(2\pi i \varphi_k)|\psi_k\rangle$.
- Утверждение. Алгоритм оценки фазы с вероятностью $|c_k|^2$ выдает оценку фазы собственного числа λ_k.
- Собственные векторы унитарного оператора ортогональны.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на ход алгоритма оценки фазы подать не собственный вектор?

- Пусть $|\psi\rangle = \sum_{k=1}^{N} c_k |\psi_k\rangle$, где $|\psi_k\rangle$ — собственные вектора оператора U единичной длины: $U|\psi_k\rangle = \exp(2\pi i \varphi_k)|\psi_k\rangle$.
- Утверждение. Алгоритм оценки фазы с вероятностью $|c_k|^2$ выдает оценку фазы собственного числа λ_k.
- Собственные векторы унитарного оператора ортогональны.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\Pr(|\psi\rangle, 0) = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на ход алгоритма оценки фазы подать не собственный вектор?

- Пусть $|\psi\rangle = \sum_{k=1}^{N} c_k |\psi_k\rangle$, где $|\psi_k\rangle$ — собственные вектора оператора U единичной длины: $U|\psi_k\rangle = \exp(2\pi i \varphi_k)|\psi_k\rangle$.
- Утверждение. Алгоритм оценки фазы с вероятностью $|c_k|^2$ выдает оценку фазы собственного числа λ_k.
- Собственные векторы унитарного оператора ортогональны.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на ход алгоритма оценки фазы подать не собственный вектор?

- Пусть $|\psi\rangle = \sum_{k=1}^{N} c_k |\psi_k\rangle$, где $|\psi_k\rangle$ — собственные вектора оператора U единичной длины: $U|\psi_k\rangle = \exp(2\pi i \varphi_k)|\psi_k\rangle$.
- Утверждение. Алгоритм оценки фазы с вероятностью $|c_k|^2$ выдает оценку фазы собственного числа λ_k.
- Собственные векторы унитарного оператора ортогональны.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на ход алгоритма оценки фазы подать не собственный вектор?

- Пусть $|\psi\rangle = \sum_{k=1}^{N} c_k |\psi_k\rangle$, где $|\psi_k\rangle$ — собственные вектора оператора U единичной длины: $U|\psi_k\rangle = \exp(2\pi i \varphi_k)|\psi_k\rangle$.
- Утверждение. Алгоритм оценки фазы с вероятностью $|c_k|^2$ выдает оценку фазы собственного числа λ_k.
- Собственные векторы унитарного оператора ортогональны.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа на произвольном входе (продолжение)

- Что происходит при серии измерений? Раскладывая по ортогональной системе собственных векторов, убеждаемся, что амплитуда для набора исходов $x_1, \ldots, x_s$ равна

$$\sum_k c_k \prod_{j=1}^{s} (1 + (-1)^{x_j} \lambda_k)/2$$

(для каждого собственного вектора в управляющих кубитах получаем разложимое состояние $|\xi_k\rangle^{\otimes s}$).

- Вероятность наблюдения набора исходов $x_1, \ldots, x_s$

$$\mathbf{Pr}(|\psi\rangle, (x_1, \ldots, x_s)) = \sum_k |c_k|^2 \prod_{j=1}^{s} \frac{1 + (-1)^{x_j} \cos(2\pi\varphi_k)}{2}.$$

- Такую же вероятность дает классический процесс: выбрать с вероятностью $|c_k|^2$ параметр $p_k = \frac{1+\cos(2\pi\varphi_k)}{2}$ и провести s испытаний Бернулли с этим параметром.

Работа на произвольном входе (продолжение)

- Что происходит при серии измерений? Раскладывая по ортогональной системе собственных векторов, убеждаемся, что амплитуда для набора исходов $x_1, \ldots, x_s$ равна

$$\sum_k c_k \prod_{j=1}^{s} (1 + (-1)^{x_j} \lambda_k)/2$$

(для каждого собственного вектора в управляющих кубитах получаем разложимое состояние $|\xi_k\rangle^{\otimes s}$).

- Вероятность наблюдения набора исходов $x_1, \ldots, x_s$

$$\mathbf{Pr}(|\psi\rangle, (x_1, \ldots, x_s)) = \sum_k |c_k|^2 \prod_{j=1}^{s} \frac{1 + (-1)^{x_j} \cos(2\pi\varphi_k)}{2}.$$

- Такую же вероятность дает классический процесс: выбрать с вероятностью $|c_k|^2$ параметр $p_k = \frac{1+\cos(2\pi\varphi_k)}{2}$ и провести s испытаний Бернулли с этим параметром.

Работа на произвольном входе (продолжение)

- Что происходит при серии измерений? Раскладывая по ортогональной системе собственных векторов, убеждаемся, что амплитуда для набора исходов $x_1, \ldots, x_s$ равна

$$\sum_k c_k \prod_{j=1}^{s} (1 + (-1)^{x_j} \lambda_k)/2$$

(для каждого собственного вектора в управляющих кубитах получаем разложимое состояние $|\xi_k\rangle^{\otimes s}$).

- Вероятность наблюдения набора исходов $x_1, \ldots, x_s$

$$\mathbf{Pr}(|\psi\rangle, (x_1, \ldots, x_s)) = \sum_k |c_k|^2 \prod_{j=1}^{s} \frac{1 + (-1)^{x_j} \cos(2\pi\varphi_k)}{2}.$$

- Такую же вероятность дает классический процесс: выбрать с вероятностью $|c_k|^2$ параметр $p_k = \frac{1+\cos(2\pi\varphi_k)}{2}$ и провести s испытаний Бернулли с этим параметром.

Алгоритмы оценки фазы: основные свойства

- Используя только оператор U можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\delta^{-2}\log(1/\varepsilon))$.
- Используя операторы U^{2^k}, $k = 0, \ldots, n = O(\log(1/\delta))$, можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\log(1/\delta)\log\log(1/\delta)\log(1/\varepsilon))$ (повторный логарифм возникает из-за необходимости оценивать фазу для каждого U^{2^k} с вероятностью ошибки $< \varepsilon/n$).
- Если применять алгоритм оценки фазы к линейной комбинации собственных векторов $|\psi\rangle = \sum_{k=1}^{N} c_k|\psi_k\rangle$, то в результате работы алгоритма с вероятностью $|c_k|^2$ получается оценка фазы φ_k.

Алгоритмы оценки фазы: основные свойства

- Используя только оператор U можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\delta^{-2}\log(1/\varepsilon))$.
- Используя операторы U^{2^k}, $k = 0, \ldots, n = O(\log(1/\delta))$, можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\log(1/\delta)\log\log(1/\delta)\log(1/\varepsilon))$ (повторный логарифм возникает из-за необходимости оценивать фазу для каждого U^{2^k} с вероятностью ошибки $< \varepsilon/n$).
- Если применять алгоритм оценки фазы к линейной комбинации собственных векторов $|\psi\rangle = \sum_{k=1}^{N} c_k|\psi_k\rangle$, то в результате работы алгоритма с вероятностью $|c_k|^2$ получается оценка фазы φ_k.

Алгоритмы оценки фазы: основные свойства

- Используя только оператор U можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\delta^{-2}\log(1/\varepsilon)$.
- Используя операторы U^{2^k}, $k = 0, \ldots, n = O(\log(1/\delta))$, можно оценить фазу собственного числа с точностью δ и вероятностью ошибки ε за время $O(\log(1/\delta)\log\log(1/\delta)\log(1/\varepsilon))$ (повторный логарифм возникает из-за необходимости оценивать фазу для каждого U^{2^k} с вероятностью ошибки $< \varepsilon/n$).
- Если применять алгоритм оценки фазы к линейной комбинации собственных векторов $|\psi\rangle = \sum_{k=1}^{N} c_k|\psi_k\rangle$, то в результате работы алгоритма с вероятностью $|c_k|^2$ получается оценка фазы φ_k.

План

Задача нахождения периода

Формулировка задачи нахождения периода

Даны: двоичные записи чисел q, a, где $a < q$, $(a, q) = 1$ ((a, q) обозначает наибольший общий делитель).
Найти: *период* a относительно q, т. е. такое наименьшее неотрицательное число t, что $a^t \equiv 1 \pmod{q}$.

Другими словами, период — это порядок числа a в мультипликативной группе вычетов $(\mathbb{Z}/q\mathbb{Z})^*$.
Будем обозначать период числа a относительно q как $\mathrm{per}_q(a)$.
Нас интересует время работы алгоритмов, решающих задачу нахождения периода в зависимости от длины записи входа. Вместо длины записи входа будем использовать $n = 1 + \lceil \log q \rceil$.

Задача нахождения периода

Формулировка задачи нахождения периода

Даны: двоичные записи чисел q, a, где $a < q$, $(a, q) = 1$ ((a, q) обозначает наибольший общий делитель).
Найти: *период* a относительно q, т. е. такое наименьшее неотрицательное число t, что $a^t \equiv 1 \pmod{q}$.

Другими словами, период — это порядок числа a в мультипликативной группе вычетов $(\mathbb{Z}/q\mathbb{Z})^*$.
Будем обозначать период числа a относительно q как $\mathrm{per}_q(a)$.
Нас интересует время работы алгоритмов, решающих задачу нахождения периода в зависимости от длины записи входа. Вместо длины записи входа будем использовать $n = 1 + \lceil \log q \rceil$.

Задача нахождения периода

Формулировка задачи нахождения периода

Даны: двоичные записи чисел q, a, где $a < q$, $(a, q) = 1$ ((a, q) обозначает наибольший общий делитель).
Найти: *период* a относительно q, т. е. такое наименьшее неотрицательное число t, что $a^t \equiv 1 \pmod{q}$.

Другими словами, период — это порядок числа a в мультипликативной группе вычетов $(\mathbb{Z}/q\mathbb{Z})^*$.
Будем обозначать период числа a относительно q как $\mathrm{per}_q(a)$.
Нас интересует время работы алгоритмов, решающих задачу нахождения периода в зависимости от длины записи входа. Вместо длины записи входа будем использовать $n = 1 + \lceil \log q \rceil$.

Теоретико-числовые алгоритмы

Напомним, что существуют эффективные (работающие за полиномиальное время) классические алгоритмы для решения следующих задач (числа заданы двоичными записями):

- Найти значения арифметических операций над целыми числами.
- По x, q найти $x \bmod q$.
- По x, n, q найти $x^n \bmod q$.
- Проверить, является ли x точной степенью (т. е., что существует y и $k \geqslant 2$ такие, что $x = y^k$).
- По x, y найти наибольший общий делитель (x, y).

Теоретико-числовые алгоритмы

Напомним, что существуют эффективные (работающие за полиномиальное время) классические алгоритмы для решения следующих задач (числа заданы двоичными записями):

- Найти значения арифметических операций над целыми числами.
- По x, q найти $x \bmod q$.
- По x, n, q найти $x^n \bmod q$.
- Проверить, является ли x точной степенью (т. е., что существует y и $k \geqslant 2$ такие, что $x = y^k$).
- По x, y найти наибольший общий делитель (x, y).

Теоретико-числовые алгоритмы

Напомним, что существуют эффективные (работающие за полиномиальное время) классические алгоритмы для решения следующих задач (числа заданы двоичными записями):

- Найти значения арифметических операций над целыми числами.
- По x, q найти $x \bmod q$.
- По x, n, q найти $x^n \bmod q$.
- Проверить, является ли x точной степенью (т. е., что существует y и $k \geqslant 2$ такие, что $x = y^k$).
- По x, y найти наибольший общий делитель (x, y).

Теоретико-числовые алгоритмы

Напомним, что существуют эффективные (работающие за полиномиальное время) классические алгоритмы для решения следующих задач (числа заданы двоичными записями):

- Найти значения арифметических операций над целыми числами.
- По x, q найти $x \bmod q$.
- По x, n, q найти $x^n \bmod q$.
- Проверить, является ли x точной степенью (т. е., что существует y и $k \geqslant 2$ такие, что $x = y^k$).
- По x, y найти наибольший общий делитель (x, y).

Теоретико-числовые алгоритмы

Напомним, что существуют эффективные (работающие за полиномиальное время) классические алгоритмы для решения следующих задач (числа заданы двоичными записями):

- Найти значения арифметических операций над целыми числами.
- По x, q найти $x \bmod q$.
- По x, n, q найти $x^n \bmod q$.
- Проверить, является ли x точной степенью (т. е., что существует y и $k \geqslant 2$ такие, что $x = y^k$).
- По x, y найти наибольший общий делитель (x, y).

Собственные числа циклической перестановки

Рассмотрим (классический) оператор на n кубитах

$$U_a\colon |x\rangle \mapsto |ax \bmod q\rangle.$$

Как было сказано, для U_a существует схема полиномиального размера. Более того, существует схема полиномиального от m, n размера для $U_a^{2^m}$, так как $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.
Пространство, натянутое на $|1\rangle, |a\rangle, |a^2\rangle, \ldots, |a^{t-1}\rangle$, является инвариантным пространством U_a, который циклически переставляет указанные векторы.

Утверждение

Собственные числа циклической перестановки $C_t\colon |j\rangle \mapsto |j+1 \bmod t\rangle$ равны $\exp(2\pi ik/t)$.

Собственные векторы: $|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle$.

Собственные числа циклической перестановки

Рассмотрим (классический) оператор на n кубитах

$$U_a\colon |x\rangle \mapsto |ax \bmod q\rangle.$$

Как было сказано, для U_a существует схема полиномиального размера.

Более того, существует схема полиномиального от m, n размера для $U_a^{2^m}$, так как $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Пространство, натянутое на $|1\rangle$, $|a\rangle$, $|a^2\rangle$, $\ldots$, $|a^{t-1}\rangle$, является инвариантным пространством U_a, который циклически переставляет указанные векторы.

Утверждение

Собственные числа циклической перестановки $C_t\colon |j\rangle \mapsto |j+1 \bmod t\rangle$ равны $\exp(2\pi ik/t)$.

Собственные векторы: $|\xi_k\rangle = \dfrac{1}{\sqrt{t}} \sum\limits_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle$.

Собственные числа циклической перестановки

Рассмотрим (классический) оператор на n кубитах

$$U_a\colon |x\rangle \mapsto |ax \bmod q\rangle.$$

Как было сказано, для U_a существует схема полиномиального размера. Более того, существует схема полиномиального от m, n размера для $U_a^{2^m}$, так как $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Пространство, натянутое на $|1\rangle, |a\rangle, |a^2\rangle, \ldots, |a^{t-1}\rangle$, является инвариантным пространством U_a, который циклически переставляет указанные векторы.

Утверждение

Собственные числа циклической перестановки $C_t\colon |j\rangle \mapsto |j+1 \bmod t\rangle$ равны $\exp(2\pi ik/t)$.

Собственные векторы: $|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle$.

Собственные числа циклической перестановки

Рассмотрим (классический) оператор на n кубитах

$$U_a\colon |x\rangle \mapsto |ax \bmod q\rangle.$$

Как было сказано, для U_a существует схема полиномиального размера. Более того, существует схема полиномиального от m, n размера для $U_a^{2^m}$, так как $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.
Пространство, натянутое на $|1\rangle$, $|a\rangle$, $|a^2\rangle$, $\ldots$, $|a^{t-1}\rangle$, является инвариантным пространством U_a, который циклически переставляет указанные векторы.

Утверждение

Собственные числа циклической перестановки $C_t\colon |j\rangle \mapsto |j+1 \bmod t\rangle$ равны $\exp(2\pi ik/t)$.

Собственные векторы: $|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle$.

Собственные числа циклической перестановки

Рассмотрим (классический) оператор на n кубитах

$$U_a\colon |x\rangle \mapsto |ax \bmod q\rangle.$$

Как было сказано, для U_a существует схема полиномиального размера. Более того, существует схема полиномиального от m, n размера для $U_a^{2^m}$, так как $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.
Пространство, натянутое на $|1\rangle$, $|a\rangle$, $|a^2\rangle$, $\ldots$, $|a^{t-1}\rangle$, является инвариантным пространством U_a, который циклически переставляет указанные векторы.

Утверждение

Собственные числа циклической перестановки $C_t\colon |j\rangle \mapsto |j+1 \bmod t\rangle$ равны $\exp(2\pi ik/t)$.

Собственные векторы: $|\xi_k\rangle = \dfrac{1}{\sqrt{t}} \sum\limits_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle$.

Доказательство утверждения

Каждый из указанных векторов — собственный с указанным собственным числом:

$$C_t|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t) C_t|j\rangle =$$

$$= \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j+1 \bmod t\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ik(j-1)/t)|j\rangle =$$

$$= \exp(2\pi ik/t) \cdot \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle = \exp(2\pi ik/t)|\xi_k\rangle$$

Поскольку найдены t собственных чисел (вида $\exp(2\pi ik/t)$), то других собственных чисел нет.

Доказательство утверждения

Каждый из указанных векторов — собственный с указанным собственным числом:

$$C_t|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t) C_t|j\rangle =$$

$$= \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j+1 \bmod t\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ik(j-1)/t)|j\rangle =$$

$$= \exp(2\pi ik/t) \cdot \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle = \exp(2\pi ik/t)|\xi_k\rangle$$

Поскольку найдены t собственных чисел (вида $\exp(2\pi ik/t)$), то других собственных чисел нет.

Доказательство утверждения

Каждый из указанных векторов — собственный с указанным собственным числом:

$$C_t|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t) C_t|j\rangle =$$

$$= \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j+1 \bmod t\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ik(j-1)/t)|j\rangle =$$

$$= \exp(2\pi ik/t) \cdot \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle = \exp(2\pi ik/t)|\xi_k\rangle$$

Поскольку найдены t собственных чисел (вида $\exp(2\pi ik/t)$), то других собственных чисел нет.

Доказательство утверждения

Каждый из указанных векторов — собственный с указанным собственным числом:

$$C_t|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t) C_t|j\rangle =$$

$$= \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j+1 \bmod t\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ik(j-1)/t)|j\rangle =$$

$$= \exp(2\pi ik/t) \cdot \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle = \exp(2\pi ik/t)|\xi_k\rangle$$

Поскольку найдены t собственных чисел (вида $\exp(2\pi ik/t)$), то других собственных чисел нет.

Доказательство утверждения

Каждый из указанных векторов — собственный с указанным собственным числом:

$$C_t|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t) C_t|j\rangle =$$

$$= \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j+1 \bmod t\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ik(j-1)/t)|j\rangle =$$

$$= \exp(2\pi ik/t) \cdot \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \exp(-2\pi ikj/t)|j\rangle = \exp(2\pi ik/t)|\xi_k\rangle$$

Поскольку найдены t собственных чисел (вида $\exp(2\pi ik/t)$), то других собственных чисел нет.

Описание алгоритма нахождения периода

1. $\ell = 5$ раз применим алгоритм оценки фазы с точностью 2^{-2n-2} и вероятностью ошибки $< 1/32$ к оператору U_a и вектору $|1\rangle$.
   В ответе получим некоторые дроби p_j/q_j, $j = 1, \ldots, \ell$.
2. Для каждой дроби p_j/q_j найдем ближайшую дробь k'_j/t'_j со знаменателем $< 2^n$ (используя разложение в цепную дробь).
3. Выдадим в качестве ответа наименьшее общее кратное чисел t'_j.

Первый шаг алгоритма

В силу анализа алгоритма оценки фазы и возможности вычислять U^{2^m} за полиномиальное от m, n время первый шаг алгоритма выполняется за полиномиальное от длины входа время.

Упражнение

Проверьте, что для циклической перестановки C_t

$$|0\rangle = \frac{1}{\sqrt{t}} \sum_{k=0}^{t-1} |\xi_k\rangle$$

Для оператора U_a вектор $|1\rangle$ играет такую же роль, как вектор $|0\rangle$ для циклической перестановки (по циклу переставляются показатели). Значит, результаты измерений в п. 1 с вероятностью ошибки $< 5/32$ дают приближения с точностью 2^{-2n-2} к числам k_j/t, где каждое k_j распределено равномерно на множестве $\{0, \ldots, t-1\}$.

Первый шаг алгоритма

В силу анализа алгоритма оценки фазы и возможности вычислять U^{2^m} за полиномиальное от m, n время первый шаг алгоритма выполняется за полиномиальное от длины входа время.

Упражнение

Проверьте, что для циклической перестановки C_t

$$|0\rangle = \frac{1}{\sqrt{t}} \sum_{k=0}^{t-1} |\xi_k\rangle$$

Для оператора U_a вектор $|1\rangle$ играет такую же роль, как вектор $|0\rangle$ для циклической перестановки (по циклу переставляются показатели). Значит, результаты измерений в п. 1 с вероятностью ошибки $< 5/32$ дают приближения с точностью 2^{-2n-2} к числам k_j/t, где каждое k_j распределено равномерно на множестве $\{0, \ldots, t-1\}$.

Первый шаг алгоритма

В силу анализа алгоритма оценки фазы и возможности вычислять U^{2^m} за полиномиальное от m, n время первый шаг алгоритма выполняется за полиномиальное от длины входа время.

Упражнение

Проверьте, что для циклической перестановки C_t

$$|0\rangle = \frac{1}{\sqrt{t}} \sum_{k=0}^{t-1} |\xi_k\rangle$$

Для оператора U_a вектор $|1\rangle$ играет такую же роль, как вектор $|0\rangle$ для циклической перестановки (по циклу переставляются показатели). Значит, результаты измерений в п. 1 с вероятностью ошибки $< 5/32$ дают приближения с точностью 2^{-2n-2} к числам k_j/t, где каждое k_j распределено равномерно на множестве $\{0, \ldots, t-1\}$.

Цепные дроби

Напомним факты о цепных дробях.

Утверждение

Каждое действительное число α раскладывается в цепную дробь

$$\alpha = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \dots}},$$

подходящие дроби имеют вид

$$\frac{p_k}{q_k} = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\dots + \cfrac{1}{a_k}}}}.$$

Неравенства с подходящими дробями

Теорема

Для подходящих дробей выполняются неравенства

$$\frac{p_0}{q_0} < \frac{p_2}{q_2} < \cdots < \alpha < \cdots < \frac{p_3}{q_3} < \frac{p_1}{q_1} \quad \text{и} \quad \left|\frac{p_k}{q_k} - \frac{p_{k-1}}{q_{k-1}}\right| = \frac{1}{q_k q_{k-1}}\,.$$

Следствие

$$\left|\alpha - \frac{p_{k-1}}{q_{k-1}}\right| < \frac{1}{q_{k-1}^2}\,.$$

Теорема

Если $|\alpha - p/q| < \dfrac{1}{2q^2}$, то p/q — подходящая дробь для α.

Неравенства с подходящими дробями

Теорема

Для подходящих дробей выполняются неравенства

$$\frac{p_0}{q_0} < \frac{p_2}{q_2} < \cdots < \alpha < \cdots < \frac{p_3}{q_3} < \frac{p_1}{q_1} \quad \text{и} \quad \left|\frac{p_k}{q_k} - \frac{p_{k-1}}{q_{k-1}}\right| = \frac{1}{q_k q_{k-1}}\,.$$

Следствие

$$\left|\alpha - \frac{p_{k-1}}{q_{k-1}}\right| < \frac{1}{q_{k-1}^2}\,.$$

Теорема

Если $|\alpha - p/q| < \dfrac{1}{2q^2}$, то p/q — подходящая дробь для α.

Неравенства с подходящими дробями

Теорема

Для подходящих дробей выполняются неравенства

$$\frac{p_0}{q_0} < \frac{p_2}{q_2} < \cdots < \alpha < \cdots < \frac{p_3}{q_3} < \frac{p_1}{q_1} \quad \text{и} \quad \left|\frac{p_k}{q_k} - \frac{p_{k-1}}{q_{k-1}}\right| = \frac{1}{q_k q_{k-1}}\,.$$

Следствие

$$\left|\alpha - \frac{p_{k-1}}{q_{k-1}}\right| < \frac{1}{q_{k-1}^2}\,.$$

Теорема

Если $|\alpha - p/q| < \dfrac{1}{2q^2}$, то p/q — подходящая дробь для α.

Алгоритмические следствия

Утверждение

Подходящие дроби для рационального α вычисляются за полиномиальное от длины записи α время (как в алгоритме Евклида).

Анализ шага 2 алгоритма нахождения периода

Поскольку число p_j/q_j отличается от фазы k_j/t не более, чем на 2^{-2n-2}, а $t < 2^n$, то разложение в цепную дробь даст несократимую дробь $k'_j/t'_j = k_j/t$, где k_j — случайный (равномерно распределенный числитель).

Выполняется за полиномиальное время.

Алгоритмические следствия

Утверждение

Подходящие дроби для рационального α вычисляются за полиномиальное от длины записи α время (как в алгоритме Евклида).

Анализ шага 2 алгоритма нахождения периода

Поскольку число p_j/q_j отличается от фазы k_j/t не более, чем на 2^{-2n-2}, а $t < 2^n$, то разложение в цепную дробь даст несократимую дробь $k'_j/t'_j = k_j/t$, где k_j — случайный (равномерно распределенный числитель).

Выполняется за полиномиальное время.

Шаг 3 алгоритма нахождения периода

Лемма

Пусть по равномерному распределению независимо выбраны числа $0 \leqslant k_j < t$, $1 \leqslant j \leqslant \ell$, $\ell \geqslant 2$.
Тогда вероятность того, что наименьшее общее кратное знаменателей несократимых дробей, равных k_j/t, отлично от t, меньше $\frac{\pi^2}{3} \cdot 2^{-\ell}$.

Доказательство

Вероятность того, что $k_1, \ldots, k_\ell$ имеют общий простой делитель p, не больше, чем $1/p^\ell$.
Поэтому вероятность того, что $k_1, \ldots, k_\ell$ не взаимно просты в совокупности (равносильно тому, что НОК не равен t), не выше

$$\sum_{k=2}^{\infty} \frac{1}{k^\ell} < \frac{\pi^2}{3} \cdot 2^{-\ell}.$$

Шаг 3 алгоритма нахождения периода

Лемма

Пусть по равномерному распределению независимо выбраны числа $0 \leqslant k_j < t$, $1 \leqslant j \leqslant \ell$, $\ell \geqslant 2$.

Тогда вероятность того, что наименьшее общее кратное знаменателей несократимых дробей, равных k_j/t, отлично от t, меньше $\frac{\pi^2}{3} \cdot 2^{-\ell}$.

Доказательство

Вероятность того, что $k_1, \ldots, k_\ell$ имеют общий простой делитель p, не больше, чем $1/p^\ell$.

Поэтому вероятность того, что $k_1, \ldots, k_\ell$ не взаимно просты в совокупности (равносильно тому, что НОК не равен t), не выше

$$\sum_{k=2}^{\infty} \frac{1}{k^\ell} < \frac{\pi^2}{3} \cdot 2^{-\ell}.$$

Шаг 3 алгоритма нахождения периода

Лемма

Пусть по равномерному распределению независимо выбраны числа $0 \leqslant k_j < t$, $1 \leqslant j \leqslant \ell$, $\ell \geqslant 2$.
Тогда вероятность того, что наименьшее общее кратное знаменателей несократимых дробей, равных k_j/t, отлично от t, меньше $\frac{\pi^2}{3} \cdot 2^{-\ell}$.

Доказательство

Вероятность того, что $k_1, \ldots, k_\ell$ имеют общий простой делитель p, не больше, чем $1/p^\ell$.
Поэтому вероятность того, что $k_1, \ldots, k_\ell$ не взаимно просты в совокупности (равносильно тому, что НОК не равен t), не выше

$$\sum_{k=2}^{\infty} \frac{1}{k^\ell} < \frac{\pi^2}{3} \cdot 2^{-\ell} .$$

Анализ алгоритма нахождения периода

- Алгоритм работает за полиномиальное от длины входа время (НОК находится с помощью алгоритма Евклида и формулы $(x, y) \cdot [x, y] = xy$).
- Вероятность ошибки не превосходит

$$\left(\frac{\pi^2}{3} + 5\right) \cdot \frac{1}{32} < \frac{1}{3}.$$

Теорема

Существует полиномиальный квантовый алгоритм нахождения периода, вероятность ошибки которого $< 1/3$.

Анализ алгоритма нахождения периода

- Алгоритм работает за полиномиальное от длины входа время (НОК находится с помощью алгоритма Евклида и формулы $(x, y) \cdot [x, y] = xy$).
- Вероятность ошибки не превосходит

$$\left(\frac{\pi^2}{3} + 5\right) \cdot \frac{1}{32} < \frac{1}{3}.$$

Теорема

Существует полиномиальный квантовый алгоритм нахождения периода, вероятность ошибки которого $< 1/3$.

Анализ алгоритма нахождения периода

- Алгоритм работает за полиномиальное от длины входа время (НОК находится с помощью алгоритма Евклида и формулы $(x, y) \cdot [x, y] = xy$).
- Вероятность ошибки не превосходит

$$\left(\frac{\pi^2}{3} + 5\right) \cdot \frac{1}{32} < \frac{1}{3}\,.$$

Теорема

Существует полиномиальный квантовый алгоритм нахождения периода, вероятность ошибки которого $< 1/3$.

План

Задача факторизации

Задача факторизации числа

Дано: натуральное число y в двоичной записи.
Найти: разложение y на простые множители

$$y = p_1^{\alpha_1} p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}.$$

Прагматическое свидетельство трудности факторизации

На предположении о трудности задачи факторизации основаны практические алгоритмы криптографии (шифрование с открытым ключом, например).
Если бы существовал нетрудоемкий алгоритм ее решения, кто-нибудь уже взломал бы RSA.

Задача факторизации

Задача факторизации числа

Дано: натуральное число y в двоичной записи.
Найти: разложение y на простые множители

$$y = p_1^{\alpha_1} p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}.$$

Прагматическое свидетельство трудности факторизации

На предположении о трудности задачи факторизации основаны практические алгоритмы криптографии (шифрование с открытым ключом, например).
Если бы существовал нетрудоемкий алгоритм ее решения, кто-нибудь уже взломал бы RSA.

Полиномиальный квантовый алгоритм для задачи факторизации

Теорема

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Следствие

Существует полиномиальный квантовый алгоритм факторизации числа.

Замечания

Исходный квантовый алгоритм Шора также использовал эту сводимость. Квантовая часть (нахождение периода) была устроена иначе и основывалась на преобразовании Фурье (переход к базису собственных векторов для циклического сдвига).
Алгоритм Шора также включает использование цепных дробей.

Полиномиальный квантовый алгоритм для задачи факторизации

Теорема

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Следствие

Существует полиномиальный квантовый алгоритм факторизации числа.

Замечания

Исходный квантовый алгоритм Шора также использовал эту сводимость. Квантовая часть (нахождение периода) была устроена иначе и основывалась на преобразовании Фурье (переход к базису собственных векторов для циклического сдвига).
Алгоритм Шора также включает использование цепных дробей.

Полиномиальный квантовый алгоритм для задачи факторизации

Теорема

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Следствие

Существует полиномиальный квантовый алгоритм факторизации числа.

Замечания

Исходный квантовый алгоритм Шора также использовал эту сводимость. Квантовая часть (нахождение периода) была устроена иначе и основывалась на преобразовании Фурье (переход к базису собственных векторов для циклического сдвига).
Алгоритм Шора также включает использование цепных дробей.

Описание сводимости

Мы построим процедуру, которая использует подпрограмму вычисления периода и находит некоторый нетривиальный делитель числа y с вероятностью $\geqslant 1/2$ или сообщает, что y простое.

Из этой процедуры уже легко получить полиномиальный алгоритм факторизации:

1. Повторяя процедуру нахождения делителя s раз, уменьшаем вероятность ошибки до 2^{-s}.
2. Применяем такую усиленную процедуру $O(\log y)$ раз, вероятность ошибки уходит до $O(\log y/2^s)$, что всё ещё мало.

Описание сводимости

Мы построим процедуру, которая использует подпрограмму вычисления периода и находит некоторый нетривиальный делитель числа y с вероятностью $\geqslant 1/2$ или сообщает, что y простое.
Из этой процедуры уже легко получить полиномиальный алгоритм факторизации:

1. Повторяя процедуру нахождения делителя s раз, уменьшаем вероятность ошибки до 2^{-s}.
2. Применяем такую усиленную процедуру $O(\log y)$ раз, вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Описание сводимости

Мы построим процедуру, которая использует подпрограмму вычисления периода и находит некоторый нетривиальный делитель числа y с вероятностью $\geqslant 1/2$ или сообщает, что y простое.
Из этой процедуры уже легко получить полиномиальный алгоритм факторизации:

1. Повторяя процедуру нахождения делителя s раз, уменьшаем вероятность ошибки до 2^{-s}.
2. Применяем такую усиленную процедуру $O(\log y)$ раз, вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Описание сводимости

Мы построим процедуру, которая использует подпрограмму вычисления периода и находит некоторый нетривиальный делитель числа y с вероятностью $\geqslant 1/2$ или сообщает, что y простое.
Из этой процедуры уже легко получить полиномиальный алгоритм факторизации:

1. Повторяя процедуру нахождения делителя s раз, уменьшаем вероятность ошибки до 2^{-s}.
2. Применяем такую усиленную процедуру $O(\log y)$ раз, вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Процедура нахождения делителя

1. Проверяем четность y. Если y — четное, то выдаем ответ «2».
2. Проверяем, извлекается ли из y нацело корень k-й степени при $k = 2, \ldots, \log_2 y$. Если $y = m^k$, то ответ «m».
3. Выбираем случайно и равновероятно a среди чисел от 1 до y, вычисляем $r = \mathrm{per}_y(a)$ (используя имеющийся по предположению алгоритм нахождения периода) и, если r — нечетное, то ответ «y — простое».
4. Если же r четное, то находим $d = (a^{r/2} - 1, y)$ алгоритмом Евклида и если $d > 1$, то ответ «d», иначе ответ «y — простое».

Из построения ясно, что алгоритм работает полиномиальное время. Ясно также, что неправильным может быть лишь ответ «y — простое».

Процедура нахождения делителя

1. Проверяем четность y. Если y — четное, то выдаем ответ «2».
2. Проверяем, извлекается ли из y нацело корень k-й степени при $k = 2, \ldots, \log_2 y$. Если $y = m^k$, то ответ «m».
3. Выбираем случайно и равновероятно a среди чисел от 1 до y, вычисляем $r = \mathrm{per}_y(a)$ (используя имеющийся по предположению алгоритм нахождения периода) и, если r — нечетное, то ответ «y — простое».
4. Если же r четное, то находим $d = (a^{r/2} - 1, y)$ алгоритмом Евклида и если $d > 1$, то ответ «d», иначе ответ «y — простое».

Из построения ясно, что алгоритм работает полиномиальное время. Ясно также, что неправильным может быть лишь ответ «y — простое».

Процедура нахождения делителя

1. Проверяем четность y. Если y — четное, то выдаем ответ «2».
2. Проверяем, извлекается ли из y нацело корень k-й степени при $k = 2, \ldots, \log_2 y$. Если $y = m^k$, то ответ «m».
3. Выбираем случайно и равновероятно a среди чисел от 1 до y, вычисляем $r = \mathrm{per}_y(a)$ (используя имеющийся по предположению алгоритм нахождения периода) и, если r — нечетное, то ответ «y — простое».
4. Если же r четное, то находим $d = (a^{r/2} - 1, y)$ алгоритмом Евклида и если $d > 1$, то ответ «d», иначе ответ «y — простое».

Из построения ясно, что алгоритм работает полиномиальное время. Ясно также, что неправильным может быть лишь ответ «y — простое».

Процедура нахождения делителя

1. Проверяем четность y. Если y — четное, то выдаем ответ «2».
2. Проверяем, извлекается ли из y нацело корень k-й степени при $k = 2, \ldots, \log_2 y$. Если $y = m^k$, то ответ «m».
3. Выбираем случайно и равновероятно a среди чисел от 1 до y, вычисляем $r = \mathrm{per}_y(a)$ (используя имеющийся по предположению алгоритм нахождения периода) и, если r — нечетное, то ответ «y — простое».
4. Если же r четное, то находим $d = (a^{r/2} - 1, y)$ алгоритмом Евклида и если $d > 1$, то ответ «d», иначе ответ «y — простое».

Из построения ясно, что алгоритм работает полиномиальное время. Ясно также, что неправильным может быть лишь ответ «y — простое».

Процедура нахождения делителя

1. Проверяем четность y. Если y — четное, то выдаем ответ «2».
2. Проверяем, извлекается ли из y нацело корень k-й степени при $k = 2, \ldots, \log_2 y$. Если $y = m^k$, то ответ «m».
3. Выбираем случайно и равновероятно a среди чисел от 1 до y, вычисляем $r = \mathrm{per}_y(a)$ (используя имеющийся по предположению алгоритм нахождения периода) и, если r — нечетное, то ответ «y — простое».
4. Если же r четное, то находим $d = (a^{r/2} - 1, y)$ алгоритмом Евклида и если $d > 1$, то ответ «d», иначе ответ «y — простое».

Из построения ясно, что алгоритм работает полиномиальное время. Ясно также, что неправильным может быть лишь ответ «y — простое».

Анализ процедуры нахождения делителя

Лемма

Процедура нахождения делителя выдает собственный делитель y с вероятностью не меньше $1 - 1/2^{k-1}$, где k — число различных простых делителей y. (Для простого y эта вероятность равна 0.)

Китайская теорема об остатках

Если $y = \prod_{j=1}^{k} p_j^{\alpha_j}$ — разложение на простые, то

$$(\mathbb{Z}/y\mathbb{Z})^* \cong \prod_{j=1}^{k} (\mathbb{Z}/p_j^{\alpha_j})^*.$$

Цикличность мультипликативной группы вычетов по модулю p^α

Группа $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ — циклическая, т. е. существует такой вычет g, что

$$(\mathbb{Z}/p^\alpha\mathbb{Z})^* = \{g^s : 0 \leqslant s < p^\alpha - p^{\alpha-1}\}.$$

Анализ процедуры нахождения делителя

Лемма

Процедура нахождения делителя выдает собственный делитель y с вероятностью не меньше $1 - 1/2^{k-1}$, где k — число различных простых делителей y. (Для простого y эта вероятность равна 0.)

Китайская теорема об остатках

Если $y = \prod_{j=1}^{k} p_j^{\alpha_j}$ — разложение на простые, то

$$(\mathbb{Z}/y\mathbb{Z})^* \cong \prod_{j=1}^{k} (\mathbb{Z}/p_j^{\alpha_j})^*.$$

Цикличность мультипликативной группы вычетов по модулю p^α

Группа $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ — циклическая, т. е. существует такой вычет g, что

$$(\mathbb{Z}/p^\alpha\mathbb{Z})^* = \{g^s : 0 \leqslant s < p^\alpha - p^{\alpha-1}\}.$$

Анализ процедуры нахождения делителя

Лемма

Процедура нахождения делителя выдает собственный делитель y с вероятностью не меньше $1 - 1/2^{k-1}$, где k — число различных простых делителей y. (Для простого y эта вероятность равна 0.)

Китайская теорема об остатках

Если $y = \prod_{j=1}^{k} p_j^{\alpha_j}$ — разложение на простые, то

$$(\mathbb{Z}/y\mathbb{Z})^* \cong \prod_{j=1}^{k} (\mathbb{Z}/p_j^{\alpha_j})^*.$$

Цикличность мультипликативной группы вычетов по модулю p^α

Группа $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ — циклическая, т. е. существует такой вычет g, что

$$(\mathbb{Z}/p^\alpha\mathbb{Z})^* = \{g^s : 0 \leqslant s < p^\alpha - p^{\alpha-1}\}.$$

Когда процедура отвечает «y — простое»

- Пусть $r_j = \mathrm{per}_{(p_j^{\alpha_j})}\, a = 2^{s_j} r'_j$, где r'_j — нечетное.
- Утверждение. Ответ «y — простое» равносилен

$$s_1 = s_2 = \cdots = s_k.$$

- Если $r = \mathrm{per}_y(a)$ — нечетное, то r_j нечетное для каждого j, поскольку r равен НОК r_j для всех j (китайская теорема). Это случай $s_1 = s_2 = \cdots = s_k = 0$.
- Если $r = \mathrm{per}_y(a)$ — четное, то $(a^{r/2}+1)(a^{r/2}-1) \equiv 0 \pmod{y}$. Так как $a^{r/2} \not\equiv 1 \pmod{y}$, в этом случае процедура выдаст ответ «y — простое» только тогда, когда $a^{r/2} \equiv -1 \pmod{y}$.
- $a^{r_j/2} \equiv -1 \pmod{p_j^{\alpha_j}}$ (используем цикличность $(\mathbb{Z}/p_j^{\alpha_j}\mathbb{Z})^*$). Если $s_1 = s_2 = \ldots = s_k \geqslant 1$, то и $a^{r/2} \equiv -1 \pmod{y}$ (используем китайскую теорему об остатках).

Когда процедура отвечает «y — простое»

- Пусть $r_j = \mathrm{per}_{(p_j^{\alpha_j})}\, a = 2^{s_j} r'_j$, где r'_j — нечетное.
- Утверждение. Ответ «y — простое» равносилен

$$s_1 = s_2 = \cdots = s_k.$$

- Если $r = \mathrm{per}_y(a)$ — нечетное, то r_j нечетное для каждого j, поскольку r равен НОК r_j для всех j (китайская теорема). Это случай $s_1 = s_2 = \cdots = s_k = 0$.
- Если $r = \mathrm{per}_y(a)$ — четное, то $(a^{r/2}+1)(a^{r/2}-1) \equiv 0 \pmod{y}$. Так как $a^{r/2} \not\equiv 1 \pmod{y}$, в этом случае процедура выдаст ответ «y — простое» только тогда, когда $a^{r/2} \equiv -1 \pmod{y}$.
- $a^{r_j/2} \equiv -1 \pmod{p_j^{\alpha_j}}$ (используем цикличность $(\mathbb{Z}/p_j^{\alpha_j}\mathbb{Z})^*$). Если $s_1 = s_2 = \ldots = s_k \geqslant 1$, то и $a^{r/2} \equiv -1 \pmod{y}$ (используем китайскую теорему об остатках).

Когда процедура отвечает «y — простое»

- Пусть $r_j = \mathrm{per}_{(p_j^{\alpha_j})}\, a = 2^{s_j} r'_j$, где r'_j — нечетное.
- Утверждение. Ответ «y — простое» равносилен

$$s_1 = s_2 = \cdots = s_k.$$

- Если $r = \mathrm{per}_y(a)$ — нечетное, то r_j нечетное для каждого j, поскольку r равен НОК r_j для всех j (китайская теорема). Это случай $s_1 = s_2 = \cdots = s_k = 0$.
- Если $r = \mathrm{per}_y(a)$ — четное, то $(a^{r/2}+1)(a^{r/2}-1) \equiv 0 \pmod{y}$. Так как $a^{r/2} \not\equiv 1 \pmod{y}$, в этом случае процедура выдаст ответ «y — простое» только тогда, когда $a^{r/2} \equiv -1 \pmod{y}$.
- $a^{r_j/2} \equiv -1 \pmod{p_j^{\alpha_j}}$ (используем цикличность $(\mathbb{Z}/p_j^{\alpha_j}\mathbb{Z})^*$). Если $s_1 = s_2 = \ldots = s_k \geqslant 1$, то и $a^{r/2} \equiv -1 \pmod{y}$ (используем китайскую теорему об остатках).

Когда процедура отвечает «y — простое»

- Пусть $r_j = \mathrm{per}_{(p_j^{\alpha_j})}\, a = 2^{s_j} r'_j$, где r'_j — нечетное.
- Утверждение. Ответ «y — простое» равносилен

$$s_1 = s_2 = \cdots = s_k.$$

- Если $r = \mathrm{per}_y(a)$ — нечетное, то r_j нечетное для каждого j, поскольку r равен НОК r_j для всех j (китайская теорема). Это случай $s_1 = s_2 = \cdots = s_k = 0$.
- Если $r = \mathrm{per}_y(a)$ — четное, то $(a^{r/2} + 1)(a^{r/2} - 1) \equiv 0 \pmod{y}$. Так как $a^{r/2} \not\equiv 1 \pmod{y}$, в этом случае процедура выдаст ответ «y — простое» только тогда, когда $a^{r/2} \equiv -1 \pmod{y}$.
- $a^{r_j/2} \equiv -1 \pmod{p_j^{\alpha_j}}$ (используем цикличность $(\mathbb{Z}/p_j^{\alpha_j}\mathbb{Z})^*$). Если $s_1 = s_2 = \ldots = s_k \geqslant 1$, то и $a^{r/2} \equiv -1 \pmod{y}$ (используем китайскую теорему об остатках).

Когда процедура отвечает «y — простое»

- Пусть $r_j = \mathrm{per}_{(p_j^{\alpha_j})}\, a = 2^{s_j} r'_j$, где r'_j — нечетное.
- Утверждение. Ответ «y — простое» равносилен

$$s_1 = s_2 = \cdots = s_k.$$

- Если $r = \mathrm{per}_y(a)$ — нечетное, то r_j нечетное для каждого j, поскольку r равен НОК r_j для всех j (китайская теорема). Это случай $s_1 = s_2 = \cdots = s_k = 0$.
- Если $r = \mathrm{per}_y(a)$ — четное, то $(a^{r/2} + 1)(a^{r/2} - 1) \equiv 0 \pmod{y}$. Так как $a^{r/2} \not\equiv 1 \pmod{y}$, в этом случае процедура выдаст ответ «y — простое» только тогда, когда $a^{r/2} \equiv -1 \pmod{y}$.
- $a^{r_j/2} \equiv -1 \pmod{p_j^{\alpha_j}}$ (используем цикличность $(\mathbb{Z}/p_j^{\alpha_j}\mathbb{Z})^*$). Если $s_1 = s_2 = \ldots = s_k \geqslant 1$, то и $a^{r/2} \equiv -1 \pmod{y}$ (используем китайскую теорему об остатках).

Завершение доказательства утверждения

Если не все s_j равны, то при некотором m получим $a^{r/2} \equiv 1 \pmod{p_m^{\alpha_m}}$, т. е. $a^{r/2} \not\equiv -1 \pmod{y}$.

Действительно, r делится на 2^s, где $s = \max_j s_j$. Если не все $s_m < s$, то $2^{s_m} r'_m \mid r/2$, что и означает $a^{r/2} \equiv 1 \pmod{p_m^{\alpha_m}}$.

Завершение доказательства утверждения

Если не все s_j равны, то при некотором m получим $a^{r/2} \equiv 1 \pmod{p_m^{\alpha_m}}$, т. е. $a^{r/2} \not\equiv -1 \pmod{y}$.
Действительно, r делится на 2^s, где $s = \max_j s_j$. Если не все $s_m < s$, то $2^{s_m} r'_m \mid r/2$, что и означает $a^{r/2} \equiv 1 \pmod{p_m^{\alpha_m}}$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

поэтому вероятность $s_1 = s$ не больше $1/2$.

- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

поэтому вероятность $s_1 = s$ не больше $1/2$.

- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

  поэтому вероятность $s_1 = s$ не больше $1/2$.
- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

поэтому вероятность $s_1 = s$ не больше $1/2$.

- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

  поэтому вероятность $s_1 = s$ не больше $1/2$.
- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.

Оценка вероятности события $s_1 = s_2 = \ldots = s_k$

- По китайской теореме об остатках случайный равномерный выбор a есть то же самое, что независимый случайный равномерный выбор всех $a_j \equiv a \pmod{p_j^{\alpha_j}}$.
- Оценим вероятность события $s_1 = s$ при независимом выборе a_1.
- Пусть $p_1^{\alpha_1} - 1 = 2^t q$, q — нечетное, g — образующая $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^*$.
- Тогда

$$|\{a_1 : s_1 = s\}| = |\{g^{2^{t-s}m} : m \text{ — нечетное}\}| =$$

$$= \begin{cases} q, & \text{если } s = 0, \\ (2^s - 2^{s-1})q = \frac{1}{2}2^{s-1}q < \frac{1}{2}2^t q, & \text{если } s > 0, \end{cases}$$

  поэтому вероятность $s_1 = s$ не больше $1/2$.
- Отсюда получаем, что вероятность события $s_1 = s_2 = \ldots = s_k$ не выше $1/2^{k-1}$.
- Лемма доказана. Значит, вероятность ошибки процедуры нахождения делителя $< 1/2$.