Эффективные квантовые алгоритмы: краткое введение

М.Н. Вялый

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

Переславль-Залесский, ИПС РАН,
4 октября 2013 года

План

1. Определение квантового алгоритма
   - Модели вычисления и ресурсы
   - Квантовые схемы, локальные операторы
   - Сравнение с вероятностным вычислением
2. Оценка фазы (собственного числа) унитарного оператора
3. Приложения: разложение числа на простые множители
   - Классические сводимости и теоретико-числовые процедуры
   - Квантовый алгоритм нахождения периода
4. Обобщения: стабилизатор действия абелевой группы

«Классическое вычисление»

1. Детерминированный алгоритм, основная модель —- машина Тьюринга.
2. Время как основной ресурс вычисления. Для машины Тьюринга это количество тактов работы.
3. Вероятностный алгоритм: имеет дополнительно доступ к последовательности независимых случайных битов.
   «Подбрасывание монеты» происходит за один такт работы.
4. Алгоритм успешен, если вероятность ошибки мала (стандартное определение: меньше 1/3).
   Вероятность ошибки можно уменьшать быстро.

С точки зрения физики

Детерминированное и вероятностное вычисления уже основаны на законах квантовой механики.

«Классическое вычисление»

1. Детерминированный алгоритм, основная модель —- машина Тьюринга.
2. Время как основной ресурс вычисления. Для машины Тьюринга это количество тактов работы.
3. Вероятностный алгоритм: имеет дополнительно доступ к последовательности независимых случайных битов.
   «Подбрасывание монеты» происходит за один такт работы.
4. Алгоритм успешен, если вероятность ошибки мала (стандартное определение: меньше 1/3).
   Вероятность ошибки можно уменьшать быстро.

С точки зрения физики

Детерминированное и вероятностное вычисления уже основаны на законах квантовой механики.

«Классическое вычисление»

1. Детерминированный алгоритм, основная модель —- машина Тьюринга.
2. Время как основной ресурс вычисления. Для машины Тьюринга это количество тактов работы.
3. Вероятностный алгоритм: имеет дополнительно доступ к последовательности независимых случайных битов. «Подбрасывание монеты» происходит за один такт работы.
4. Алгоритм успешен, если вероятность ошибки мала (стандартное определение: меньше 1/3).
   Вероятность ошибки можно уменьшать быстро.

С точки зрения физики

Детерминированное и вероятностное вычисления уже основаны на законах квантовой механики.

«Классическое вычисление»

1. Детерминированный алгоритм, основная модель —- машина Тьюринга.
2. Время как основной ресурс вычисления. Для машины Тьюринга это количество тактов работы.
3. Вероятностный алгоритм: имеет дополнительно доступ к последовательности независимых случайных битов. «Подбрасывание монеты» происходит за один такт работы.
4. Алгоритм успешен, если вероятность ошибки мала (стандартное определение: меньше $1/3$).
   Вероятность ошибки можно уменьшать быстро.

С точки зрения физики

Детерминированное и вероятностное вычисления уже основаны на законах квантовой механики.

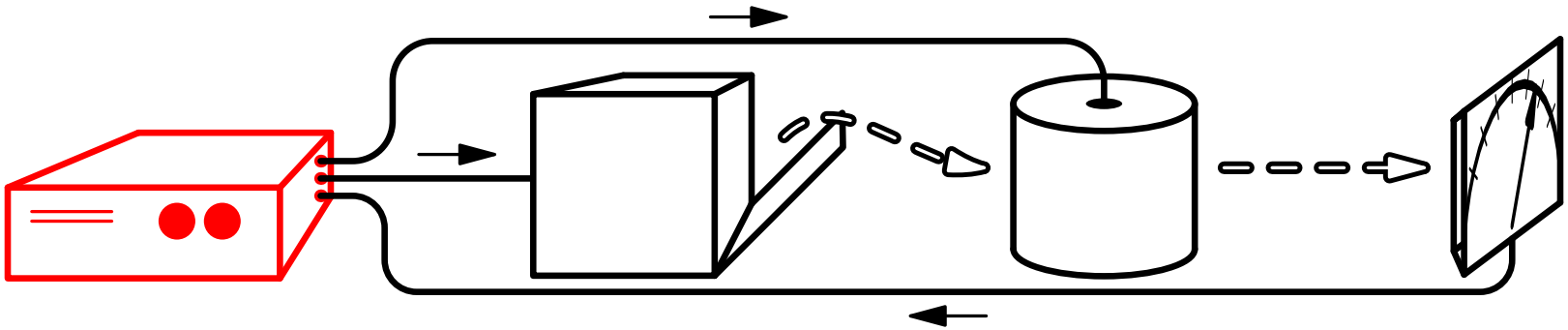
«Классическое вычисление»

1. Детерминированный алгоритм, основная модель —- машина Тьюринга.
2. Время как основной ресурс вычисления. Для машины Тьюринга это количество тактов работы.
3. Вероятностный алгоритм: имеет дополнительно доступ к последовательности независимых случайных битов. «Подбрасывание монеты» происходит за один такт работы.
4. Алгоритм успешен, если вероятность ошибки мала (стандартное определение: меньше $1/3$).
   Вероятность ошибки можно уменьшать быстро.

С точки зрения физики

Детерминированное и вероятностное вычисления уже основаны на законах квантовой механики.

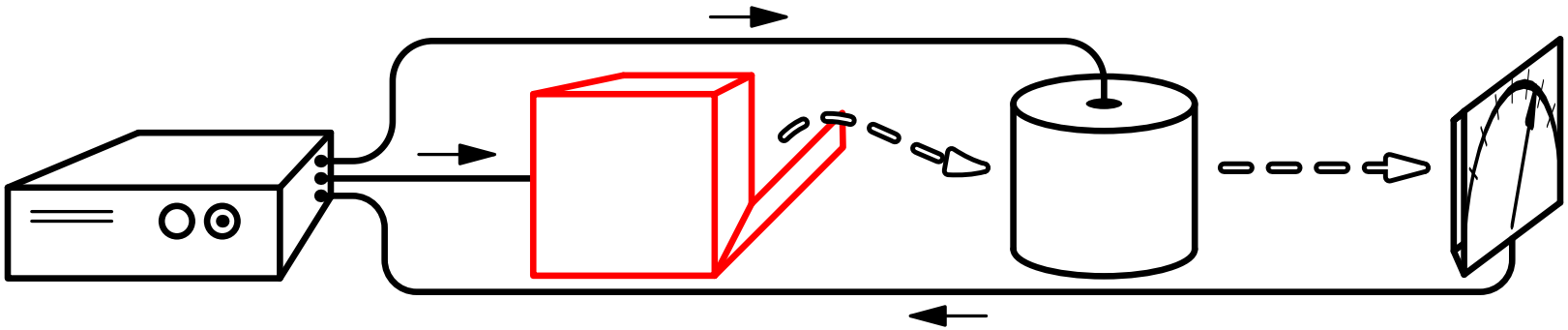
Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

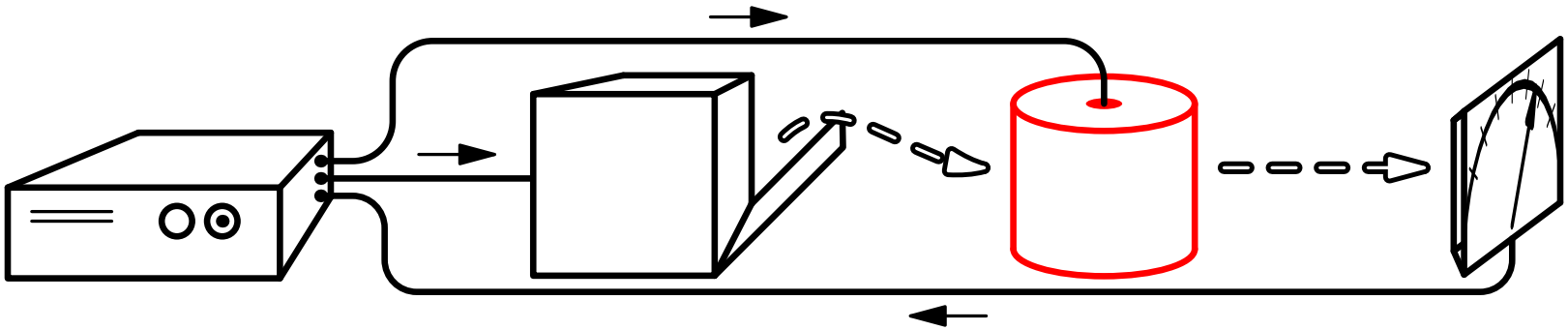
Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

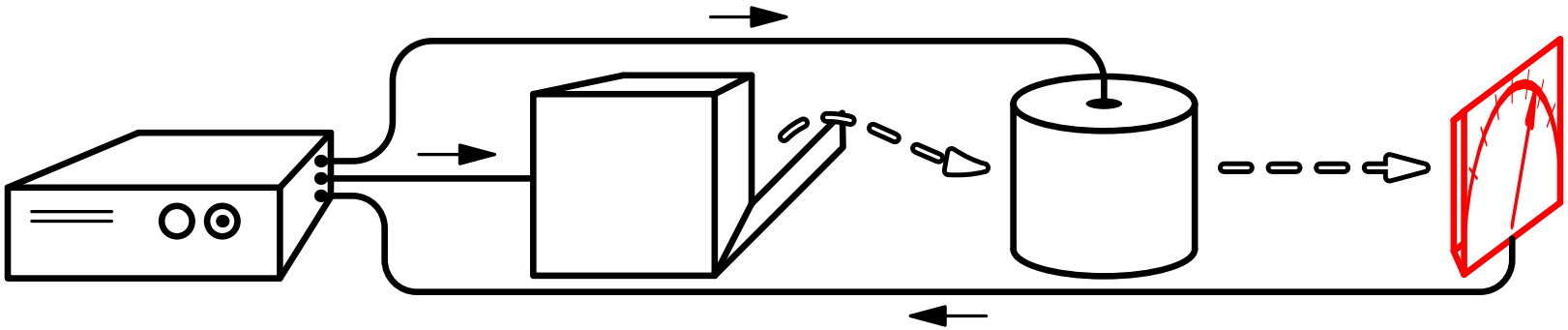
Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

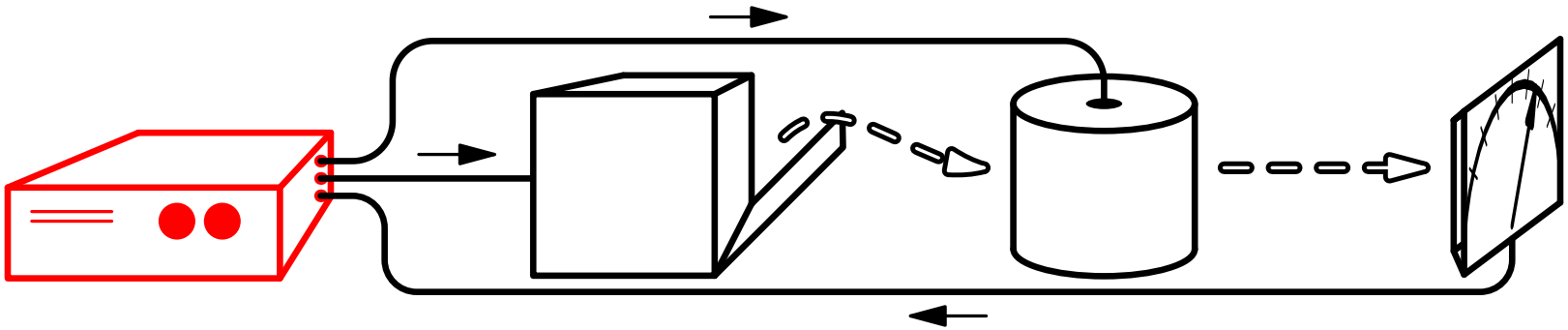
Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

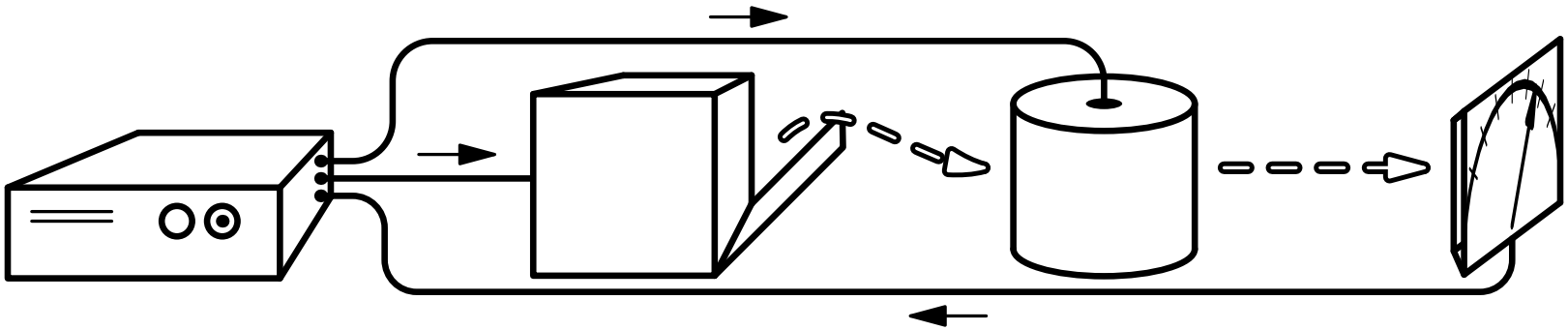
Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

Использование квантового ресурса в алгоритмах



Основные этапы:

- классическое вычисление;
- приготовление «большой» замкнутой квантовой системы в фиксированном состоянии;
- заданная эволюция замкнутой квантовой системы;
- измерение состояния системы после эволюции;
- обработка результатов измерения классическими средствами;
- циклическое повторение предыдущих шагов при необходимости.

Абстрактная замкнутая квантовая система

Пространство состояний

Проективное комплексное пространство $P\mathbb{C}^{n-1}$.

Координатное представление.

Вычислительный) базис: $|1\rangle, \ldots, |n\rangle$. (Выделен средствами наблюдения за системой.)

Состояние задается амплитудами α_i

$$|\psi\rangle = \sum_i \alpha_i |i\rangle, \qquad \sum_i |\alpha_i|^2 = 1$$

с точностью до сдвига фазы (одновременного умножения амплитуд на $e^{i\varphi}$).

Эволюция состояний

Задается унитарным оператором $U\colon \mathbb{C}^n \to \mathbb{C}^n$, $U^\dagger U = I$.

Абстрактная замкнутая квантовая система

Пространство состояний

Проективное комплексное пространство $P\mathbb{C}^{n-1}$.

Координатное представление.

Вычислительный) базис: $|1\rangle, \ldots, |n\rangle$. (Выделен средствами наблюдения за системой.)

Состояние задается амплитудами α_i

$$|\psi\rangle = \sum_i \alpha_i |i\rangle, \qquad \sum_i |\alpha_i|^2 = 1$$

с точностью до сдвига фазы (одновременного умножения амплитуд на $e^{i\varphi}$).

Эволюция состояний

Задается унитарным оператором $U\colon \mathbb{C}^n \to \mathbb{C}^n$, $U^\dagger U = I$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \to V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \to V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \longrightarrow V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \longrightarrow V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \longrightarrow V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Линейная алгебра в обозначениях Дирака

- Кет-векторы: $|\psi\rangle \in V$ (вектор-столбцы).
- Бра-векторы: $\langle\psi| \in V^*$ — линейные функционалы на V (вектор-строки).
- Эрмитово скалярное произведение: $\langle\psi|\xi\rangle$. Индуцирует изоморфизм $V \longrightarrow V^*$:

$$|\psi\rangle \mapsto \langle\psi|; \quad \langle\psi|(|\xi\rangle) = \langle\psi|\xi\rangle.$$

- Эрмитово сопряженный оператор

$$\langle A^\dagger x \mid y\rangle = \langle x \mid A \mid y\rangle.$$

- Операторы можно задавать матрицами в ортонормированном базисе:

$$A = \sum_{j,k} a_{jk}|j\rangle\langle k|, \quad \text{где } a_{jk} = \langle j|A|k\rangle \text{ — матричный элемент.}$$

- Матрица сопряженного оператора $(A^\dagger)_{jk} = (A_{kj})^*$.

Измерение в выделенном базисе

Результат измерения

Состояние перед измерением: $|\psi\rangle = \sum_i \alpha_i |i\rangle$ (нормированное разложение по выделенному базису).

Результат измерения: случайная величина, принимающая значение i с вероятностью $|\alpha_i|^2$.

Состояние системы после измерения: $|i\rangle$ с вероятностью $|\alpha_i|^2$.

Полезное соотношение

Вероятность события $S \subseteq \{1, \ldots, n\}$ выражается формулой

$$\mathbf{Pr}[S] = \langle\psi|\Pi_S|\psi\rangle,$$

где Π_S — ортогональный проектор на подпространство, порожденное векторами $|i\rangle$, $i \in S$.

Измерение в выделенном базисе

Результат измерения

Состояние перед измерением: $|\psi\rangle = \sum_i \alpha_i |i\rangle$ (нормированное разложение по выделенному базису).

Результат измерения: случайная величина, принимающая значение i с вероятностью $|\alpha_i|^2$.

Состояние системы после измерения: $|i\rangle$ с вероятностью $|\alpha_i|^2$.

Полезное соотношение

Вероятность события $S \subseteq \{1, \ldots, n\}$ выражается формулой

$$\mathbf{Pr}[S] = \langle\psi|\Pi_S|\psi\rangle,$$

где Π_S — ортогональный проектор на подпространство, порожденное векторами $|i\rangle$, $i \in S$.

Объединение систем

Основное правило

Пространство состояний составной системы является тензорным произведением пространств состояний ее частей.

Тензорное произведение: простое определение

Тензорное произведение $U \otimes V$ пространств с выделенными базисами $U = (u_1, \ldots, u_n)$ и $V = (v_1, \ldots, v_k)$ имеет выделенный базис

$$u_j \otimes v_\ell, \quad 1 \leqslant j \leqslant n;\ 1 \leqslant \ell \leqslant k,$$

$$|j, \ell\rangle \quad \text{(в обозначениях Дирака)}.$$

Тензорное произведение операторов $X \otimes Y$

Действие на разложимых векторах $(X \otimes Y)(u \otimes v) = (Xu) \otimes (Yv)$, продолжается на остальные по линейности.

Объединение систем

Основное правило

Пространство состояний составной системы является тензорным произведением пространств состояний ее частей.

Тензорное произведение: простое определение

Тензорное произведение $U \otimes V$ пространств с выделенными базисами $U = (u_1, \ldots, u_n)$ и $V = (v_1, \ldots, v_k)$ имеет выделенный базис

$$u_j \otimes v_\ell, \quad 1 \leqslant j \leqslant n;\ 1 \leqslant \ell \leqslant k,$$

$$|j, \ell\rangle \quad \text{(в обозначениях Дирака)}.$$

Тензорное произведение операторов $X \otimes Y$

Действие на разложимых векторах $(X \otimes Y)(u \otimes v) = (Xu) \otimes (Yv)$, продолжается на остальные по линейности.

Объединение систем

Основное правило

Пространство состояний составной системы является тензорным произведением пространств состояний ее частей.

Тензорное произведение: простое определение

Тензорное произведение $U \otimes V$ пространств с выделенными базисами $U = (u_1, \ldots, u_n)$ и $V = (v_1, \ldots, v_k)$ имеет выделенный базис

$$u_j \otimes v_\ell, \quad 1 \leqslant j \leqslant n;\ 1 \leqslant \ell \leqslant k,$$

$$|j, \ell\rangle \quad \text{(в обозначениях Дирака)}.$$

Тензорное произведение операторов $X \otimes Y$

Действие на разложимых векторах $(X \otimes Y)(u \otimes v) = (Xu) \otimes (Yv)$, продолжается на остальные по линейности.

Преобразования за такт работы: локальные операторы

1. Кубит: квантовая система с двумя состояниями, координатное представление пространства состояний $\mathbb{C}^2 = \mathbb{C}(|0\rangle, |1\rangle)$.
2. Система из n кубитов: объединение n кубитов — пространство $(\mathbb{C}^2)^{\otimes n}$.
3. Локальный оператор: действует на ограниченное число кубитов (достаточно двух кубитов). То есть имеет вид

$$U[S] \otimes I[\bar{S}],$$

где U действует на подмножестве кубитов S, а тождественный оператор I — на остальных.

Обозначим $x[S]$ подпоследовательность битов, стоящих на местах из множества S. Тогда

$$U[S] \otimes I[\bar{S}] = \sum_{x,y \in \{0,1\}^n : x[\bar{S}] = y[\bar{S}]} u_{x[S],y[S]} |x\rangle\langle y|.$$

Преобразования за такт работы: локальные операторы

1. Кубит: квантовая система с двумя состояниями, координатное представление пространства состояний $\mathbb{C}^2 = \mathbb{C}(|0\rangle, |1\rangle)$.
2. Система из n кубитов: объединение n кубитов — пространство $(\mathbb{C}^2)^{\otimes n}$.
3. Локальный оператор: действует на ограниченное число кубитов (достаточно двух кубитов). То есть имеет вид

$$U[S] \otimes I[\bar{S}],$$

где U действует на подмножестве кубитов S, а тождественный оператор I — на остальных.

Обозначим $x[S]$ подпоследовательность битов, стоящих на местах из множества S. Тогда

$$U[S] \otimes I[\bar{S}] = \sum_{x,y \in \{0,1\}^n : x[\bar{S}] = y[\bar{S}]} u_{x[S],y[S]} |x\rangle\langle y|.$$

Преобразования за такт работы: локальные операторы

1. Кубит: квантовая система с двумя состояниями, координатное представление пространства состояний $\mathbb{C}^2 = \mathbb{C}(|0\rangle, |1\rangle)$.
2. Система из n кубитов: объединение n кубитов — пространство $(\mathbb{C}^2)^{\otimes n}$.
3. Локальный оператор: действует на ограниченное число кубитов (достаточно двух кубитов). То есть имеет вид

$$U[S] \otimes I[\bar{S}],$$

где U действует на подмножестве кубитов S, а тождественный оператор I — на остальных.

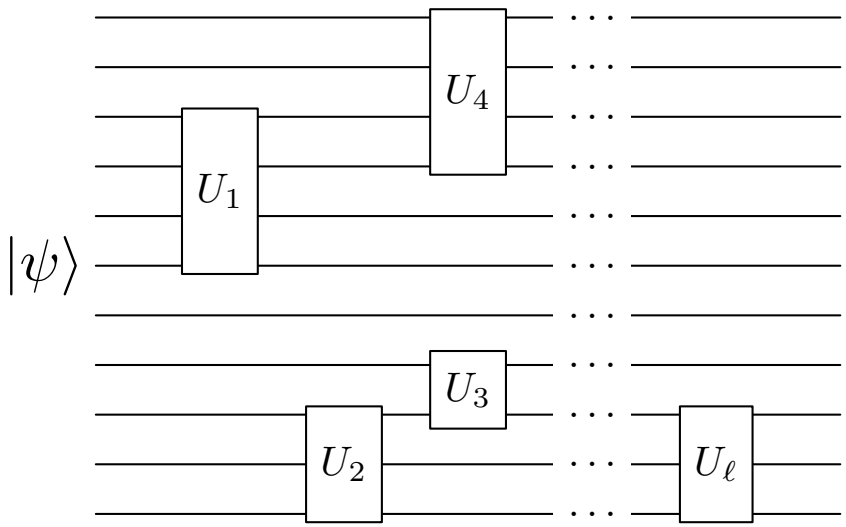
Обозначим $x[S]$ подпоследовательность битов, стоящих на местах из множества S. Тогда

$$U[S] \otimes I[\bar{S}] = \sum_{x,y \in \{0,1\}^n : x[\bar{S}] = y[\bar{S}]} u_{x[S],y[S]} |x\rangle\langle y|.$$

Квантовая схема над базисом $\mathcal{B}$

Базис

Набор унитарных операторов $\mathcal{B}$, описывающих элементарные действия.



$$U = U_\ell[S_\ell] \dots U_2[S_2] U_1[S_1]$$

Квантовая схема

Последовательность операторов

$$U_1[S_1], U_2[S_2], \dots, U_\ell[S_\ell], \quad \text{где } U_k \in \mathcal{B},\ S_k \subseteq \{1, \dots, n\}.$$

Конечные базисы

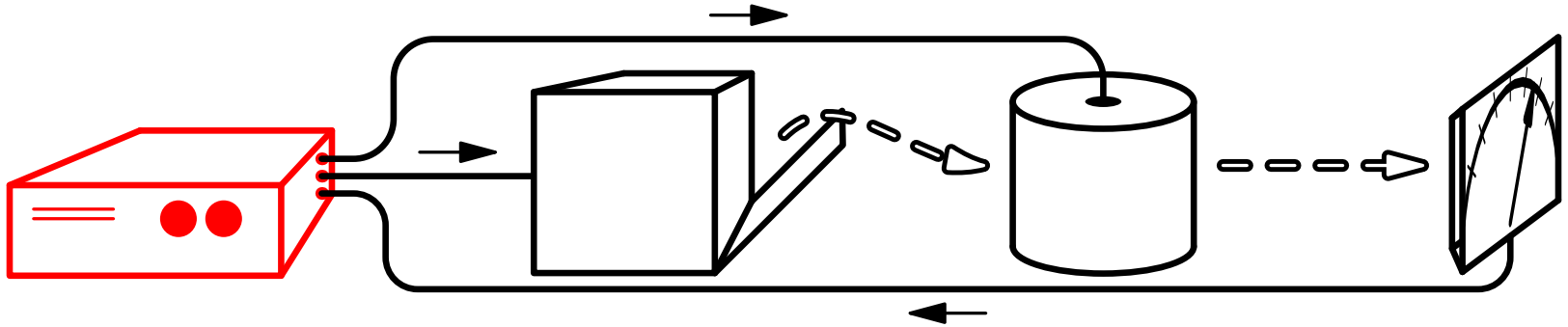
Для квантовых алгоритмов достаточно приближенной реализации унитарных операторов.
Базис называется универсальным, если в нем любой локальный оператор допускает сколь угодно точную приближенную реализацию (схемой небольшого размера).

Теорема об универсальном конечном базисе

Базис $\{\text{c-NOT}, H, K(\pi/4)\}$ — универсальный. Здесь

$$\text{c-NOT}: |x,y\rangle \mapsto |x, y\oplus x\rangle, \quad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad K(\frac{\pi}{4}) = \begin{pmatrix} 1 & 0 \\ 0 & e^{\pi i/4} \end{pmatrix}.$$

К определению квантового алгоритма

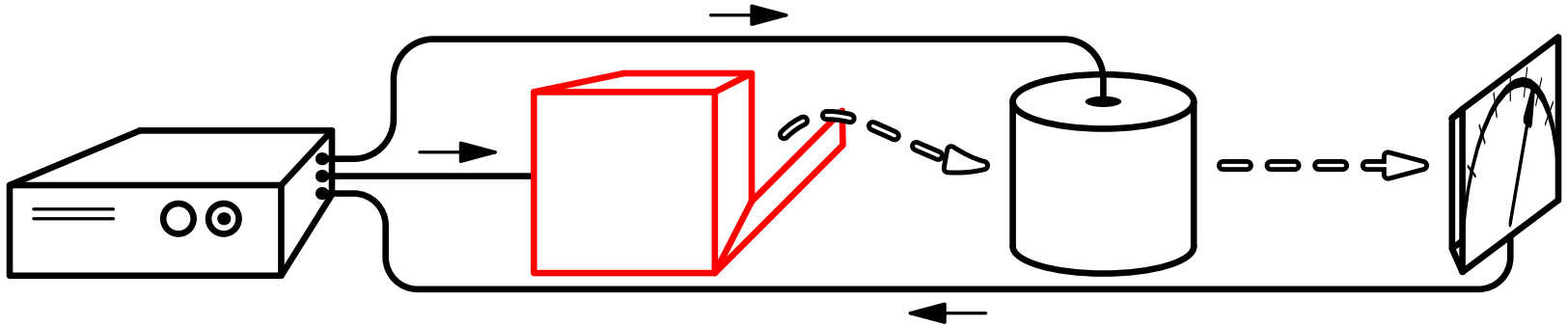


Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

К определению квантового алгоритма

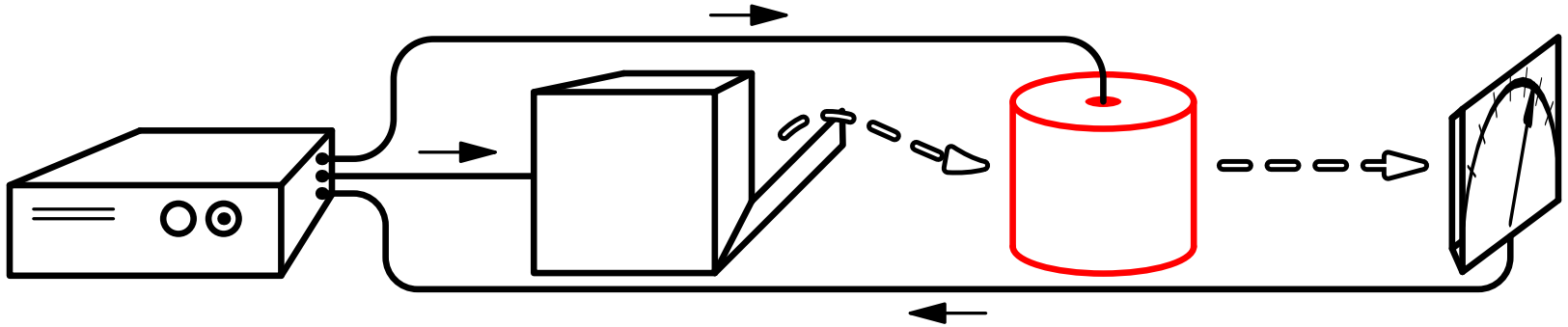


Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

К определению квантового алгоритма

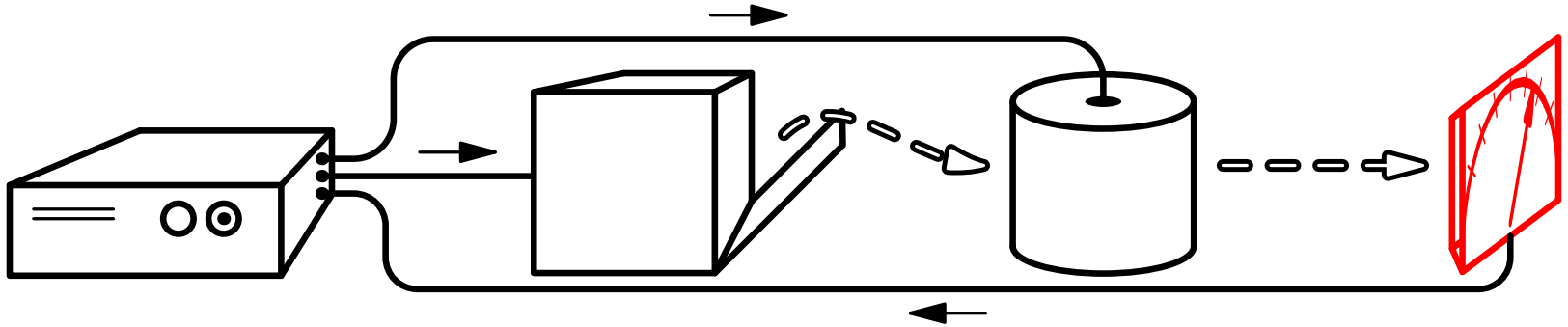


Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

К определению квантового алгоритма

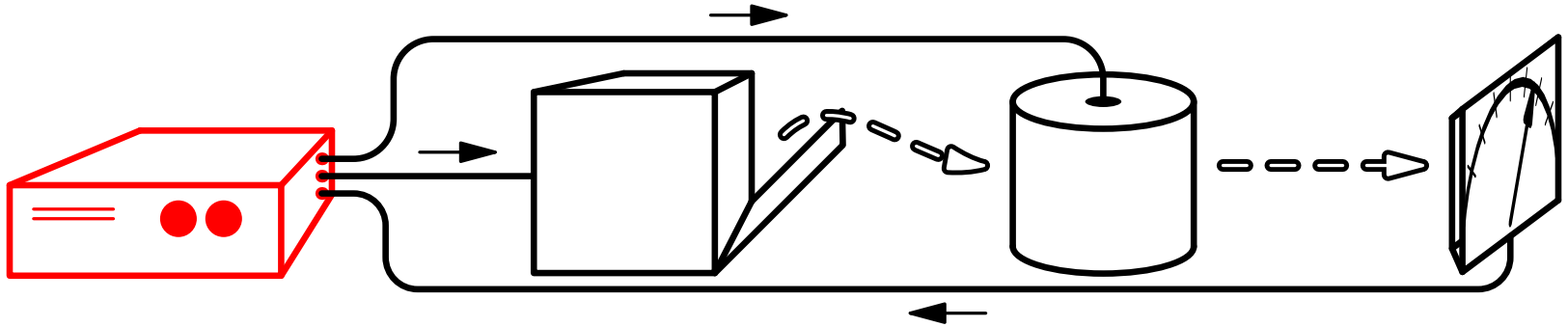


Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

К определению квантового алгоритма

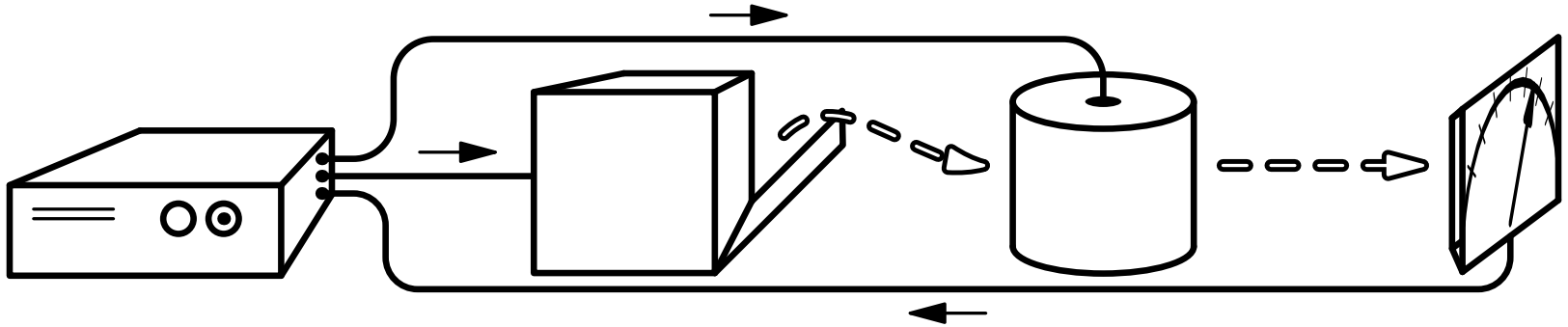


Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

К определению квантового алгоритма



Квантовый алгоритм:

- классическое вычисление;
- приготовление состояния $|0^n\rangle$;
- применение квантовой схемы U в конечном базисе (описание схемы вычисляется классически);
- измерение состояния $U|0^n\rangle$;
- циклическое повторение предыдущих шагов при необходимости.

Время работы при таком определении: количество тактов работы классического алгоритма.

Вероятностное и квантовое вычисление

Общее

1. Пространство состояний экспоненциально велико:

$$(p_{i_0 i_1 \ldots i_{n-1}}), \quad p_{i_0 i_1 \ldots i_{n-1}} \geqslant 0, \quad \sum_{(i_0, i_1, \ldots, i_{n-1}) \in \{0,1\}^n} p_{i_0 i_1 \ldots i_{n-1}} = 1. \quad (P_n)$$

$$\sum_{(i_0, i_1, \ldots, i_{n-1}) \in \{0,1\}^n} \alpha_{i_0 i_1 \ldots i_{n-1}} |i_0, i_1, \ldots, i_{n-1}\rangle, \quad \sum |\alpha_{i_0 i_1 \ldots i_{n-1}}|^2 = 1. \quad (Q_n)$$

2. Приближенная оценка сумм с экспоненциально большим количеством слагаемых (суммирование по путям вычисления).

Различия

1. Интерференция (в квантовой случае суммы со знаками, некоторые пути вычисления исчезают из сумм).
2. Обратимость (в квантовом случае эволюция замкнутой системы обратима).

Вероятностное и квантовое вычисление

Общее

1. Пространство состояний экспоненциально велико:

$$(p_{i_0 i_1 \ldots i_{n-1}}), \quad p_{i_0 i_1 \ldots i_{n-1}} \geqslant 0, \quad \sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} p_{i_0 i_1 \ldots i_{n-1}} = 1. \quad (P_n)$$

$$\sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} \alpha_{i_0 i_1 \ldots i_{n-1}} |i_0, i_1, \ldots, i_{n-1}\rangle, \quad \sum |\alpha_{i_0 i_1 \ldots i_{n-1}}|^2 = 1. \quad (Q_n)$$

2. Приближенная оценка сумм с экспоненциально большим количеством слагаемых (суммирование по путям вычисления).

Различия

1. Интерференция (в квантовой случае суммы со знаками, некоторые пути вычисления исчезают из сумм).
2. Обратимость (в квантовом случае эволюция замкнутой системы обратима).

Вероятностное и квантовое вычисление

Общее

1. Пространство состояний экспоненциально велико:

$$(p_{i_0 i_1 \ldots i_{n-1}}), \quad p_{i_0 i_1 \ldots i_{n-1}} \geqslant 0, \quad \sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} p_{i_0 i_1 \ldots i_{n-1}} = 1. \quad (P_n)$$

$$\sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} \alpha_{i_0 i_1 \ldots i_{n-1}} |i_0, i_1, \ldots, i_{n-1}\rangle, \quad \sum |\alpha_{i_0 i_1 \ldots i_{n-1}}|^2 = 1. \quad (Q_n)$$

2. Приближенная оценка сумм с экспоненциально большим количеством слагаемых (суммирование по путям вычисления).

Различия

1. Интерференция (в квантовой случае суммы со знаками, некоторые пути вычисления исчезают из сумм).
2. Обратимость (в квантовом случае эволюция замкнутой системы обратима).

Вероятностное и квантовое вычисление

Общее

1. Пространство состояний экспоненциально велико:

$$(p_{i_0 i_1 \ldots i_{n-1}}), \quad p_{i_0 i_1 \ldots i_{n-1}} \geqslant 0, \quad \sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} p_{i_0 i_1 \ldots i_{n-1}} = 1. \quad (P_n)$$

$$\sum_{(i_0,i_1,\ldots,i_{n-1})\in\{0,1\}^n} \alpha_{i_0 i_1 \ldots i_{n-1}} |i_0, i_1, \ldots, i_{n-1}\rangle, \quad \sum |\alpha_{i_0 i_1 \ldots i_{n-1}}|^2 = 1. \quad (Q_n)$$

2. Приближенная оценка сумм с экспоненциально большим количеством слагаемых (суммирование по путям вычисления).

Различия

1. Интерференция (в квантовой случае суммы со знаками, некоторые пути вычисления исчезают из сумм).
2. Обратимость (в квантовом случае эволюция замкнутой системы обратима).

Квантовые схемы моделируют классические

Перестановочные операторы

Унитарный оператор называется перестановочным, если он сохраняет множество базисных векторов выделенного базиса.

Классическое обратимое вычисление

Это квантовая схема из перестановочных операторов.

Важный факт (неформально)

Обратимость не накладывает существенных ограничений на трудоемкость алгоритмов.

Квантовые схемы моделируют классические

Перестановочные операторы

Унитарный оператор называется перестановочным, если он сохраняет множество базисных векторов выделенного базиса.

Классическое обратимое вычисление

Это квантовая схема из перестановочных операторов.

Важный факт (неформально)

Обратимость не накладывает существенных ограничений на трудоемкость алгоритмов.

Квантовые схемы моделируют классические

Перестановочные операторы

Унитарный оператор называется перестановочным, если он сохраняет множество базисных векторов выделенного базиса.

Классическое обратимое вычисление

Это квантовая схема из перестановочных операторов.

Важный факт (неформально)

Обратимость не накладывает существенных ограничений на трудоемкость алгоритмов.

Квантовые схемы моделируют классические – 2

По классическому базису $\mathcal{B} = \{f_1, \ldots, f_m\}$ из функций $f_k \colon \{0,1\}^{d_k} \to \{0,1\}$ построим обратимый базис $\mathcal{B}_\oplus = \{f_k^\oplus, \text{c-NOT}\}$,

$$f_k^\oplus \colon (x, y) \mapsto (x, y \oplus f_k(x)), \quad \text{c-NOT} \colon (x, y) \mapsto (x, x \oplus y).$$

Теорема (реализация необратимого отображения)

Если отображение $F \colon \{0,1\}^n \to \{0,1\}^m$ реализуется булевой схемой размера L в базисе $\mathcal{B}$, то существует схема размера $O(L + n + m)$ в базисе $\mathcal{B}_\oplus$, которая реализует отображение

$$F^\oplus \colon (x, y, 0^L) \mapsto (x, y \oplus F(x), 0^L).$$

Теорема (реализация обратимого отображения)

Если обратимое отображение $F \colon \{0,1\}^n \to \{0,1\}^n$ и обратное F^{-1} реализуются булевыми схемами размера $\leqslant L$ в базисе $\mathcal{B}$, то они реализуются в базисе $\mathcal{B}_\oplus$ схемами размера $O(L + n)$.

Квантовые схемы моделируют классические – 2

По классическому базису $\mathcal{B} = \{f_1, \ldots, f_m\}$ из функций $f_k \colon \{0,1\}^{d_k} \to \{0,1\}$ построим обратимый базис $\mathcal{B}_\oplus = \{f_k^\oplus, \text{c-NOT}\}$,

$$f_k^\oplus \colon (x, y) \mapsto (x, y \oplus f_k(x)), \quad \text{c-NOT} \colon (x, y) \mapsto (x, x \oplus y).$$

Теорема (реализация необратимого отображения)

Если отображение $F \colon \{0,1\}^n \to \{0,1\}^m$ реализуется булевой схемой размера L в базисе $\mathcal{B}$, то существует схема размера $O(L + n + m)$ в базисе $\mathcal{B}_\oplus$, которая реализует отображение

$$F^\oplus \colon (x, y, 0^L) \mapsto (x, y \oplus F(x), 0^L).$$

Теорема (реализация обратимого отображения)

Если обратимое отображение $F \colon \{0,1\}^n \to \{0,1\}^n$ и обратное F^{-1} реализуются булевыми схемами размера $\leqslant L$ в базисе $\mathcal{B}$, то они реализуются в базисе $\mathcal{B}_\oplus$ схемами размера $O(L + n)$.

Квантовые схемы моделируют классические – 2

По классическому базису $\mathcal{B} = \{f_1, \ldots, f_m\}$ из функций $f_k \colon \{0,1\}^{d_k} \to \{0,1\}$ построим обратимый базис $\mathcal{B}_\oplus = \{f_k^\oplus, \text{c-NOT}\}$,

$$f_k^\oplus \colon (x, y) \mapsto (x, y \oplus f_k(x)), \quad \text{c-NOT} \colon (x, y) \mapsto (x, x \oplus y).$$

Теорема (реализация необратимого отображения)

Если отображение $F \colon \{0,1\}^n \to \{0,1\}^m$ реализуется булевой схемой размера L в базисе $\mathcal{B}$, то существует схема размера $O(L + n + m)$ в базисе $\mathcal{B}_\oplus$, которая реализует отображение

$$F^\oplus \colon (x, y, 0^L) \mapsto (x, y \oplus F(x), 0^L).$$

Теорема (реализация обратимого отображения)

Если обратимое отображение $F \colon \{0,1\}^n \to \{0,1\}^n$ и обратное F^{-1} реализуются булевыми схемами размера $\leqslant L$ в базисе $\mathcal{B}$, то они реализуются в базисе $\mathcal{B}_\oplus$ схемами размера $O(L + n)$.

План

Неформальная постановка

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

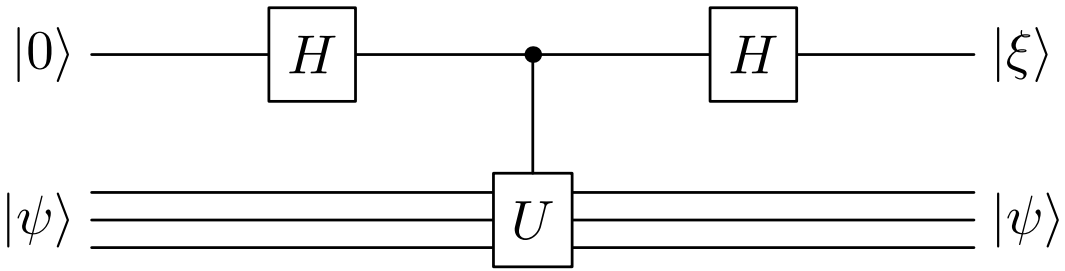
$$U|\psi\rangle = \lambda|\psi\rangle.$$

Как найти собственное число? (Достаточно найти фазу φ.)

Основная схема (схема для косинуса)

Определение условного оператора

$$\text{c-}U\colon |x\rangle \otimes |\psi\rangle = \begin{cases} |1\rangle \otimes U|\psi\rangle, & \text{если } x = 1; \\ |0\rangle \otimes |\psi\rangle, & \text{иначе.} \end{cases}$$



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

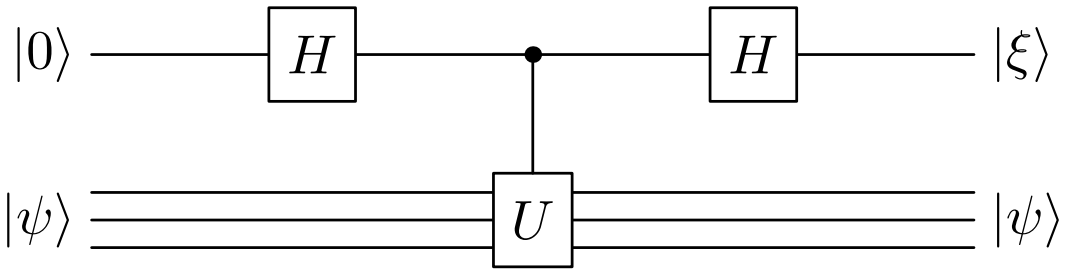
$$|\xi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & \lambda \end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}|0\rangle =$$

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

Основная схема (схема для косинуса)

Определение условного оператора

$$\text{c-}U\colon |x\rangle \otimes |\psi\rangle = \begin{cases} |1\rangle \otimes U|\psi\rangle, & \text{если } x = 1; \\ |0\rangle \otimes |\psi\rangle, & \text{иначе.} \end{cases}$$



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

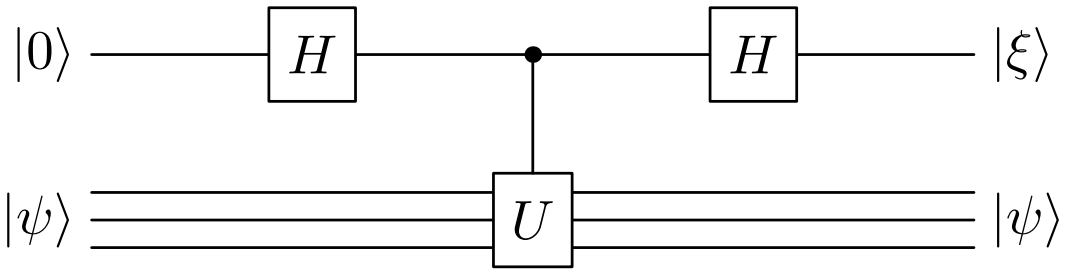
$$|\xi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & \lambda \end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}|0\rangle =$$

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

Основная схема (схема для косинуса)

Определение условного оператора

$$\text{c-}U\colon |x\rangle \otimes |\psi\rangle = \begin{cases} |1\rangle \otimes U|\psi\rangle, & \text{если } x = 1; \\ |0\rangle \otimes |\psi\rangle, & \text{иначе.} \end{cases}$$



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

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

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

Измерение управляющего кубита

Из предыдущего вычисления

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

Вероятность исхода 0:

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

Измерение управляющего кубита

Из предыдущего вычисления

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

Вероятность исхода 0:

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

Серия измерений

- Последовательно применяем основную схему к s различным управляющим кубитам.
- Измеряем каждый из 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.

Hoeffding bound (вариант оценки 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 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.

Hoeffding bound (вариант оценки 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 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.

Hoeffding bound (вариант оценки 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 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.

Hoeffding bound (вариант оценки 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 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.

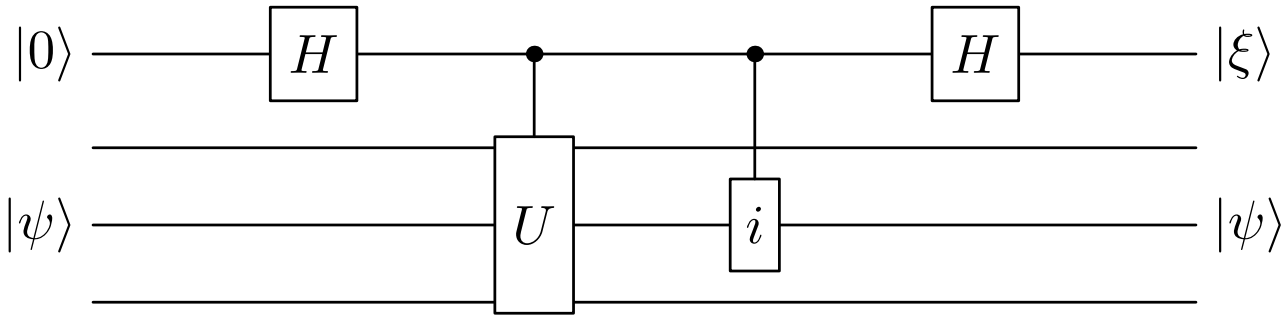
Hoeffding bound (вариант оценки Чернова)

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

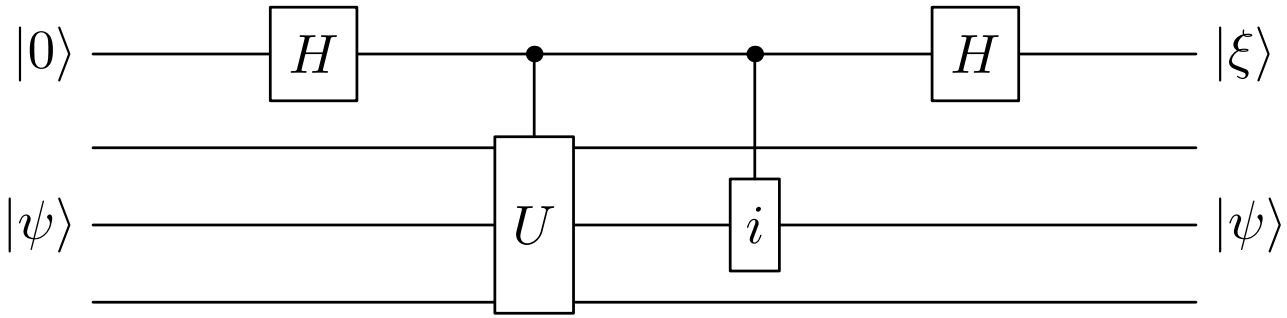
Основная схема позволяет оценивать косинус фазы φ. Для оценки синуса нужно поменять местами действительную и мнимую части λ:



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

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).

Открытый вопрос

Возможно ли применить алгоритм оценки фазы с полиномиальной точностью для построения эффективных квантовых алгоритмов решения интересных задач?

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).

Открытый вопрос

Возможно ли применить алгоритм оценки фазы с полиномиальной точностью для построения эффективных квантовых алгоритмов решения интересных задач?

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).

Открытый вопрос

Возможно ли применить алгоритм оценки фазы с полиномиальной точностью для построения эффективных квантовых алгоритмов решения интересных задач?

Уточнение значения фазы: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < 1/8$ 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 $1/2^{n+3}$.

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 фазы: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < 1/8$ 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 $1/2^{n+3}$.

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 фазы: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < 1/8$ 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 $1/2^{n+3}$.

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.

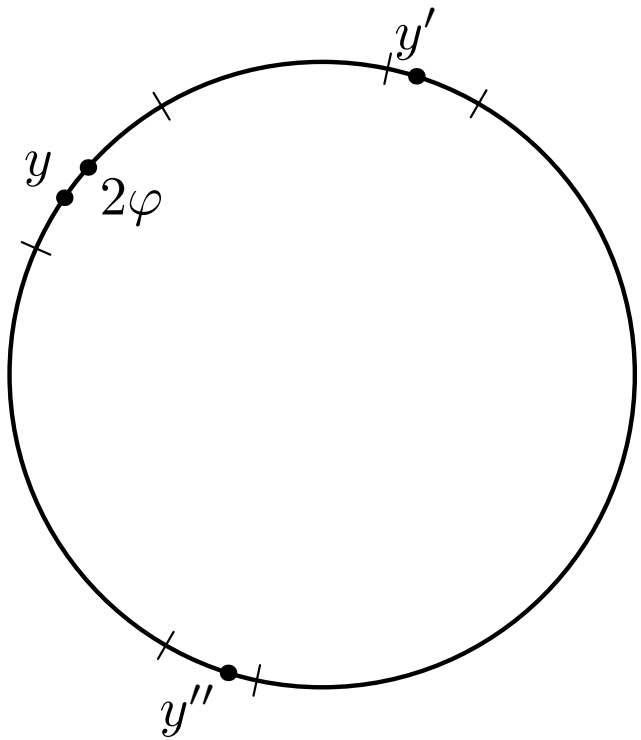
Уточнение значения фазы: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < 1/8$ 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 $1/2^{n+3}$.

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.



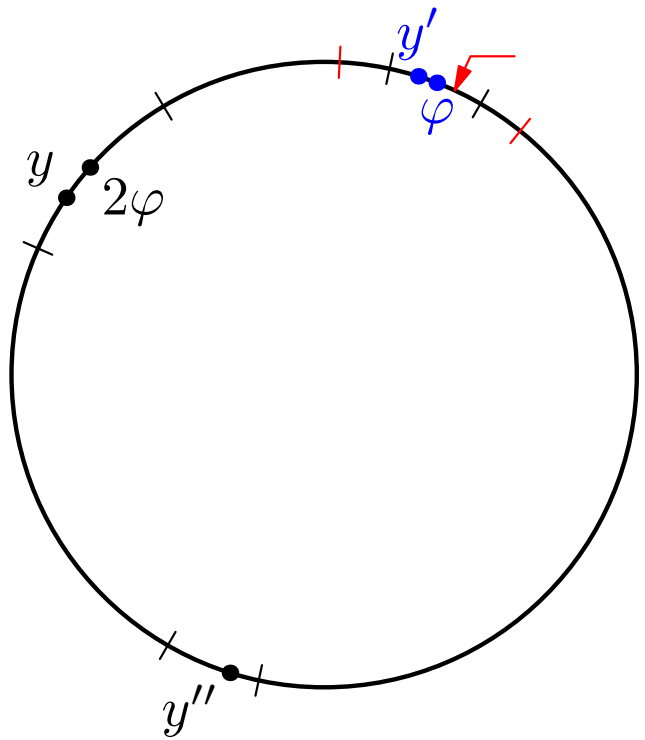
Уточнение значения фазы: алгоритм экспонент

- Если $U|\psi\rangle = \lambda|\psi\rangle$, то $U^k|\psi\rangle = \lambda^k|\psi\rangle$.
- Оценим с точностью $\delta < 1/8$ 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 $1/2^{n+3}$.

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 вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1+\cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1+\cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1+\cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1+\cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1 + \cos(2\pi\varphi_k)}{2}.$$

Работа алгоритма на произвольном входе

Вопрос

Что будет, если на вход 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 $|\psi_k\rangle$ унитарного оператора ортогональны.
- $(\langle\xi| \otimes \langle\psi|)(|\xi'\rangle \otimes |\psi\rangle) = \langle\xi|\xi'\rangle \cdot \langle\psi|\psi'\rangle$.
- Поэтому и векторы $c_k|\xi_k\rangle \otimes |\psi_k\rangle$ ортогональны. Значит,

$$\mathbf{Pr}(|\psi\rangle, 0) = \langle 0| \otimes \langle\psi|\Pi_0 \otimes I|0\rangle \otimes \psi\rangle = \sum_k |c_k|^2 \frac{1+\cos(2\pi\varphi_k)}{2}.$$

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

1. Определение квантового алгоритма
   - Модели вычисления и ресурсы
   - Квантовые схемы, локальные операторы
   - Сравнение с вероятностным вычислением
2. Оценка фазы (собственного числа) унитарного оператора
3. Приложения: разложение числа на простые множители
   - Классические сводимости и теоретико-числовые процедуры
   - Квантовый алгоритм нахождения периода
4. Обобщения: стабилизатор действия абелевой группы

Задача факторизации

Задача факторизации числа

Дано: натуральное число y в двоичной записи.

Найти: разложение y на простые множители $y = p_1^{\alpha_1} p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}$.

Структурная сложность факторизации чисел

В разумном смысле задача факторизации принадлежит классу $\mathrm{NP} \cap \text{co-NP}$ (и поэтому вряд ли является NP-полной).

Прагматическое свидетельство трудности факторизации

На предположении о трудности задачи факторизации основаны практические алгоритмы криптографии (система шифрования с открытым ключом RSA).

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

Структурная сложность факторизации чисел

В разумном смысле задача факторизации принадлежит классу $\mathrm{NP} \cap \text{co-}\mathrm{NP}$ (и поэтому вряд ли является NP-полной).

Прагматическое свидетельство трудности факторизации

На предположении о трудности задачи факторизации основаны практические алгоритмы криптографии (система шифрования с открытым ключом RSA).
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 $y = p_1^{\alpha_1} p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}$.

Структурная сложность факторизации чисел

В разумном смысле задача факторизации принадлежит классу $\mathrm{NP} \cap \text{co-}\mathrm{NP}$ (и поэтому вряд ли является NP-полной).

Прагматическое свидетельство трудности факторизации

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

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Эффективные 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) (алгоритм Евклида).
- По рациональному $\alpha = p/q$ найти разложение в цепную дробь (такие же вычисления, как в алгоритме Евклида).
- Для рационального α найти такую несократимую дробь p/q, что $|\alpha - p/q| < 1/2q^2$ (используется разложение в цепную дробь).

Задача нахождения периода

Формулировка задачи нахождения периода

Даны: двоичные записи чисел q, a, где $a < q$, $(a, q) = 1$ (здесь и далее (a, q) обозначает наибольший общий делитель).

Найти: *период* a относительно q, т. е. такое наименьшее положительное 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 $\mathrm{per}_q(a)$.

Задача нахождения периода

Формулировка задачи нахождения периода

Даны: двоичные записи чисел q, a, где $a < q$, $(a, q) = 1$ (здесь и далее (a, q) обозначает наибольший общий делитель).
Найти: *период* a относительно q, т. е. такое наименьшее положительное 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 $\mathrm{per}_q(a)$.

Сводимость факторизации к нахождению периода

Теорема (Miller, 1976)

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Сводимость основана на процедуре, 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, чтобы найти все простые делители. Вероятность ошибки уменьшится до $O(\log y/2^s)$, но этого достаточно.

Сводимость факторизации к нахождению периода

Теорема (Miller, 1976)

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Сводимость основана на процедуре, 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, чтобы найти все простые делители. Вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Сводимость факторизации к нахождению периода

Теорема (Miller, 1976)

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Сводимость основана на процедуре, 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, чтобы найти все простые делители. Вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Сводимость факторизации к нахождению периода

Теорема (Miller, 1976)

Существует полиномиальная вероятностная сводимость задачи факторизации к задаче нахождения периода.

Сводимость основана на процедуре, 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, чтобы найти все простые делители. Вероятность ошибки ухудшается до $O(\log y/2^s)$, но этого достаточно.

Основное утверждение

Теорема (Shor, 1994)

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

Замечание

Вероятность ошибки можно экспоненциально быстро понижать.

Следствие

Существует полиномиальный квантовый алгоритм факторизации целых чисел.

Применение алгоритмов оценки фазы для нахождения периода (Китаев, 1995)

1. Точная оценка фазы собственного числа оператора умножения на a по модулю q (у Шора вместо этого шага применялось преобразование Фурье).
2. Извлечение информации о периоде a из приближенного значения фазы с использованием цепных дробей (в алгоритме Шора есть похожая часть).

Применение алгоритмов оценки фазы для нахождения периода (Китаев, 1995)

1. Точная оценка фазы собственного числа оператора умножения на a по модулю q (у Шора вместо этого шага применялось преобразование Фурье).
2. Извлечение информации о периоде a из приближенного значения фазы с использованием цепных дробей (в алгоритме Шора есть похожая часть).

Оператор умножения в группе обратимых вычетов

Рассмотрим (классический) оператор на n кубитах ($n = \theta(\log q)$)

$$U_a : |x\rangle \mapsto \begin{cases} |ax \bmod q\rangle, & \text{если } (x, q) = 1,\ x < q, \\ |x\rangle, & \text{иначе.} \end{cases}$$

Следствие теорем об обратимых вычислениях

Для U_a существует квантовая (обратимая) схема полиномиального размера. Более того, существует квантовая схема полиномиального от m, n размера для $U_a^{2^m}$, так как в нетривиальном случае $U_a^{2^m} : |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Наблюдение

Пространство, натянутое на $|1\rangle, |a\rangle, |a^2\rangle, \ldots, |a^{t-1}\rangle$, $t = \mathrm{per}_q(a)$, является инвариантным пространством оператора U_a, и этот оператор циклически переставляет указанные векторы.

Оператор умножения в группе обратимых вычетов

Рассмотрим (классический) оператор на n кубитах ($n = \theta(\log q)$)

$$U_a : |x\rangle \mapsto \begin{cases} |ax \bmod q\rangle, & \text{если } (x, q) = 1,\ x < q, \\ |x\rangle, & \text{иначе.} \end{cases}$$

Следствие теорем об обратимых вычислениях

Для U_a существует квантовая (обратимая) схема полиномиального размера. Более того, существует квантовая схема полиномиального от m, n размера для $U_a^{2^m}$, так как в нетривиальном случае $U_a^{2^m} : |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Наблюдение

Пространство, натянутое на $|1\rangle, |a\rangle, |a^2\rangle, \ldots, |a^{t-1}\rangle$, $t = \mathrm{per}_q(a)$, является инвариантным пространством оператора U_a, и этот оператор циклически переставляет указанные векторы.

Оператор умножения в группе обратимых вычетов

Рассмотрим (классический) оператор на n кубитах ($n = \theta(\log q)$)

$$U_a : |x\rangle \mapsto \begin{cases} |ax \bmod q\rangle, & \text{если } (x, q) = 1,\ x < q, \\ |x\rangle, & \text{иначе.} \end{cases}$$

Следствие теорем об обратимых вычислениях

Для U_a существует квантовая (обратимая) схема полиномиального размера. Более того, существует квантовая схема полиномиального от m, n размера для $U_a^{2^m}$, так как в нетривиальном случае $U_a^{2^m} : |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Наблюдение

Пространство, натянутое на $|1\rangle, |a\rangle, |a^2\rangle, \ldots, |a^{t-1}\rangle$, $t = \mathrm{per}_q(a)$, является инвариантным пространством оператора U_a, и этот оператор циклически переставляет указанные векторы.

Оператор умножения в группе обратимых вычетов

Рассмотрим (классический) оператор на n кубитах ($n = \theta(\log q)$)

$$U_a\colon |x\rangle \mapsto \begin{cases} |ax \bmod q\rangle, & \text{если } (x,q) = 1,\ x < q, \\ |x\rangle, & \text{иначе.} \end{cases}$$

Следствие теорем об обратимых вычислениях

Для U_a существует квантовая (обратимая) схема полиномиального размера. Более того, существует квантовая схема полиномиального от m, n размера для $U_a^{2^m}$, так как в нетривиальном случае $U_a^{2^m}\colon |x\rangle \mapsto |a^{2^m} x \bmod q\rangle$.

Наблюдение

Пространство, натянутое на $|1\rangle$, $|a\rangle$, $|a^2\rangle$, $\ldots$, $|a^{t-1}\rangle$, $t = \mathrm{per}_q(a)$, является инвариантным пространством оператора U_a, и этот оператор циклически переставляет указанные векторы.

Собственные числа циклической перестановки

Утверждение

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

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

Утверждение

Для циклической перестановки C_t

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

Собственные числа циклической перестановки

Утверждение

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

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

Утверждение

Для циклической перестановки C_t

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

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.
- 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.
- 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.
- 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 $\{0, \ldots, t-1\}$.

Второй шаг

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.

Третий шаг 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

Если НОК не равен t, то $k_1, \ldots, k_\ell$ имеют общий делитель > 1.
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, не выше

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

Третий шаг 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

Если НОК не равен t, то $k_1, \ldots, k_\ell$ имеют общий делитель > 1.

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, не выше

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

Третий шаг 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

Если НОК не равен t, то $k_1, \ldots, k_\ell$ имеют общий делитель > 1.
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, не выше

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

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

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

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

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

План

1. Определение квантового алгоритма
   - Модели вычисления и ресурсы
   - Квантовые схемы, локальные операторы
   - Сравнение с вероятностным вычислением
2. Оценка фазы (собственного числа) унитарного оператора
3. Приложения: разложение числа на простые множители
   - Классические сводимости и теоретико-числовые процедуры
   - Квантовый алгоритм нахождения периода
4. Обобщения: стабилизатор действия абелевой группы

Действие абелевой группы

Действие группы G на множестве X: гомоморфизм $G \to S(X)$ в группу биективных преобразований множества X. Другими словами, это функция α: $G \times X \to X$, для которой выполняются условия

- $\alpha(g, \cdot)$ — взаимно однозначное отображение X на X;
- $\alpha(g_1 g_2, x) = \alpha(g_1, \alpha(g_2, x))$.

Часто обозначение функции опускается: $\alpha(g, x)$ обозначается как gx.

Стабилизатор: $S(x_0) = \{g \in G : gx_0 = x_0\}$.

Это подгруппа G.

Утверждение

Любая подгруппа $G < \mathbb{Z}^n$ изоморфна $\mathbb{Z}^k$. Поэтому для G существует базис образующих g_k:

$$G = \left\{ g : g = \sum_k z_k g_k \right\}, \; z_k \text{ определены однозначно по } g.$$

Действие абелевой группы

Действие группы G на множестве X: гомоморфизм $G \to S(X)$ в группу биективных преобразований множества X. Другими словами, это функция α: $G \times X \to X$, для которой выполняются условия

- $\alpha(g, \cdot)$ — взаимно однозначное отображение X на X;
- $\alpha(g_1 g_2, x) = \alpha(g_1, \alpha(g_2, x))$.

Часто обозначение функции опускается: $\alpha(g, x)$ обозначается как gx.
Стабилизатор: $S(x_0) = \{g \in G : gx_0 = x_0\}$.
Это подгруппа G.

Утверждение

Любая подгруппа $G < \mathbb{Z}^n$ изоморфна $\mathbb{Z}^k$. Поэтому для G существует базис образующих g_k:

$$G = \left\{ g : g = \sum_k z_k g_k \right\}, \; z_k \text{ определены однозначно по } g.$$

Действие абелевой группы

Действие группы G на множестве X: гомоморфизм $G \to S(X)$ в группу биективных преобразований множества X. Другими словами, это функция $\alpha\colon G \times X \to X$, для которой выполняются условия

- $\alpha(g, \cdot)$ — взаимно однозначное отображение X на X;
- $\alpha(g_1 g_2, x) = \alpha(g_1, \alpha(g_2, x))$.

Часто обозначение функции опускается: $\alpha(g, x)$ обозначается как gx.
Стабилизатор: $S(x_0) = \{g \in G : gx_0 = x_0\}$.
Это подгруппа G.

Утверждение

Любая подгруппа $G < \mathbb{Z}^n$ изоморфна $\mathbb{Z}^k$. Поэтому для G существует базис образующих g_k:

$$G = \left\{ g : g = \sum_k z_k g_k \right\},\ z_k \text{ определены однозначно по } g.$$

Нахождение стабилизатора действия $\mathbb{Z}^n$

Нахождение стабилизатора действия $\mathbb{Z}^n$

Дано: эффективное действие группы $\mathbb{Z}^n$ на множестве $X = \{0,1\}^m$. Это означает, что задан алгоритм вычисления $\alpha(z,x)$, работающий за полиномиальное время. Кроме того, указан элемент $x_0 \in X$.
Найти: базисную систему образующих для $S(x_0)$.

Частный случай: нахождение периода

Действие $\mathbb{Z}^1$: $(n,x) \mapsto (a^n x \bmod q)$. Стабилизатор 1 состоит из чисел, кратных $\mathrm{per}_q(a)$.

Теорема (Китаев, 1995)

Существует полиномиальный квантовый алгоритм нахождения стабилизатора эффективного действия абелевой группы.

Нахождение стабилизатора действия $\mathbb{Z}^n$

Нахождение стабилизатора действия $\mathbb{Z}^n$

Дано: эффективное действие группы $\mathbb{Z}^n$ на множестве $X = \{0,1\}^m$. Это означает, что задан алгоритм вычисления $\alpha(z,x)$, работающий за полиномиальное время. Кроме того, указан элемент $x_0 \in X$.
Найти: базисную систему образующих для $S(x_0)$.

Частный случай: нахождение периода

Действие $\mathbb{Z}^1$: $(n,x) \mapsto (a^n x \bmod q)$. Стабилизатор 1 состоит из чисел, кратных $\mathrm{per}_q(a)$.

Теорема (Китаев, 1995)

Существует полиномиальный квантовый алгоритм нахождения стабилизатора эффективного действия абелевой группы.

Нахождение стабилизатора действия $\mathbb{Z}^n$

Нахождение стабилизатора действия $\mathbb{Z}^n$

Дано: эффективное действие группы $\mathbb{Z}^n$ на множестве $X = \{0,1\}^m$. Это означает, что задан алгоритм вычисления $\alpha(z,x)$, работающий за полиномиальное время. Кроме того, указан элемент $x_0 \in X$.
Найти: базисную систему образующих для $S(x_0)$.

Частный случай: нахождение периода

Действие $\mathbb{Z}^1$: $(n,x) \mapsto (a^n x \bmod q)$. Стабилизатор 1 состоит из чисел, кратных $\mathrm{per}_q(a)$.

Теорема (Китаев, 1995)

Существует полиномиальный квантовый алгоритм нахождения стабилизатора эффективного действия абелевой группы.

Ещё один пример: дискретный логарифм

Определение

Пусть g — образующая $(\mathbb{Z}/p\mathbb{Z})^*$, p — простое. Дискретный логарифм $\log_g(a)$: наименьшее положительное k такое, что $g^k = a$.

Теорема (Shor, 1994)

Существует полиномиальный квантовый алгоритм вычисления дискретного логарифма.

Следствие из теоремы Китаева об абелевом стабилизаторе

Задача сводится к нахождению стабилизатора. Действие $\mathbb{Z}^2$:

$$(z_1, z_2)(x) = g^{z_1} a^{z_2} x \pmod{p}.$$

Далее нужно найти в $S(1)$ элементы вида $(k, -1)$ (решением системы диофантовых линейных уравнений).

Ещё один пример: дискретный логарифм

Определение

Пусть g — образующая $(\mathbb{Z}/p\mathbb{Z})^*$, p — простое. Дискретный логарифм $\log_g(a)$: наименьшее положительное k такое, что $g^k = a$.

Теорема (Shor, 1994)

Существует полиномиальный квантовый алгоритм вычисления дискретного логарифма.

Следствие из теоремы Китаева об абелевом стабилизаторе

Задача сводится к нахождению стабилизатора. Действие $\mathbb{Z}^2$:

$$(z_1, z_2)(x) = g^{z_1} a^{z_2} x \pmod{p}.$$

Далее нужно найти в $S(1)$ элементы вида $(k, -1)$ (решением системы диофантовых линейных уравнений).

Ещё один пример: дискретный логарифм

Определение

Пусть g — образующая $(\mathbb{Z}/p\mathbb{Z})^*$, p — простое. Дискретный логарифм $\log_g(a)$: наименьшее положительное k такое, что $g^k = a$.

Теорема (Shor, 1994)

Существует полиномиальный квантовый алгоритм вычисления дискретного логарифма.

Следствие из теоремы Китаева об абелевом стабилизаторе

Задача сводится к нахождению стабилизатора. Действие $\mathbb{Z}^2$:

$$(z_1, z_2)(x) = g^{z_1} a^{z_2} x \pmod{p}.$$

Далее нужно найти в $S(1)$ элементы вида $(k, -1)$ (решением системы диофантовых линейных уравнений).

Открытые вопросы

Что мы уже умеем?

Какие задачи сводятся к нахождению стабилизатора действия абелевой группы?

Примеры выше принадлежат $\mathrm{NP} \cap \text{co-NP}$. Сводится ли весь этот класс к стабилизаторам действия абелевых групп? Есть ли интересные примеры за пределами $\mathrm{NP} \cap \text{co-NP}$?

Стабилизатор действия S_n

Дано: действие симметрической группы $S_n \times \{0,1\}^n \to \{0,1\}^n$ задано схемой вычисления, элемент $x_0 \in \{0,1\}^n$.
Найти: $S(x_0)$.
(К этой задаче сводится изоморфизм графов.)

Уже для диэдральной группы D_n наилучший известный квантовый алгоритм поиска стабилизатора (Куперберг, 2003) требует $2^{O(\sqrt{\log n})}$.

Открытые вопросы

Что мы уже умеем?

Какие задачи сводятся к нахождению стабилизатора действия абелевой группы?

Примеры выше принадлежат $\mathrm{NP} \cap \text{co-}\mathrm{NP}$. Сводится ли весь этот класс к стабилизаторам действия абелевых групп? Есть ли интересные примеры за пределами $\mathrm{NP} \cap \text{co-}\mathrm{NP}$?

Стабилизатор действия S_n

Дано: действие симметрической группы $S_n \times \{0,1\}^n \to \{0,1\}^n$ задано схемой вычисления, элемент $x_0 \in \{0,1\}^n$.
Найти: $S(x_0)$.
(К этой задаче сводится изоморфизм графов.)

Уже для диэдральной группы D_n наилучший известный квантовый алгоритм поиска стабилизатора (Куперберг, 2003) требует $2^{O(\sqrt{\log n})}$.

Открытые вопросы

Что мы уже умеем?

Какие задачи сводятся к нахождению стабилизатора действия абелевой группы?

Примеры выше принадлежат $\mathrm{NP} \cap \text{co-}\mathrm{NP}$. Сводится ли весь этот класс к стабилизаторам действия абелевых групп? Есть ли интересные примеры за пределами $\mathrm{NP} \cap \text{co-}\mathrm{NP}$?

Стабилизатор действия S_n

Дано: действие симметрической группы $S_n \times \{0,1\}^n \to \{0,1\}^n$ задано схемой вычисления, элемент $x_0 \in \{0,1\}^n$.
Найти: $S(x_0)$.
(К этой задаче сводится изоморфизм графов.)

Уже для диэдральной группы D_n наилучший известный квантовый алгоритм поиска стабилизатора (Куперберг, 2003) требует $2^{O(\sqrt{\log n})}$.

Открытые вопросы

Что мы уже умеем?

Какие задачи сводятся к нахождению стабилизатора действия абелевой группы?

Примеры выше принадлежат $\mathrm{NP} \cap \text{co-}\mathrm{NP}$. Сводится ли весь этот класс к стабилизаторам действия абелевых групп? Есть ли интересные примеры за пределами $\mathrm{NP} \cap \text{co-}\mathrm{NP}$?

Стабилизатор действия S_n

Дано: действие симметрической группы $S_n \times \{0,1\}^n \to \{0,1\}^n$ задано схемой вычисления, элемент $x_0 \in \{0,1\}^n$.
Найти: $S(x_0)$.
(К этой задаче сводится изоморфизм графов.)

Уже для диэдральной группы D_n наилучший известный квантовый алгоритм поиска стабилизатора (Куперберг, 2003) требует $2^{O(\sqrt{\log n})}$.

Благодарю за внимание!