

От QUBO к QAOA

Квантовый приближенный алгоритм оптимизации

Чуруксаев Иван 27.10.25

Что мы уже знаем: QUBO

Мы научились формулировать задачи оптимизации в виде QUBO:

$$\min_{x \in \{0,1\}^n} C(x) = \sum_i c_i x_i + \sum_{i < j} c_{ij} x_i x_j$$

Что это значит?

$x_i \in \{0, 1\}$ — бинарные переменные

c_i, c_{ij} — коэффициенты задачи

$C(x)$ — функция стоимости

Что мы умеем?

- ✓ Формулировать задачу как QUBO
- ✓ Решать классическими методами
- ✓ Имитация отжига, генетические алгоритмы

Вопрос: Как использовать квантовый компьютер для решения?

Зачем нужен квантовый компьютер?

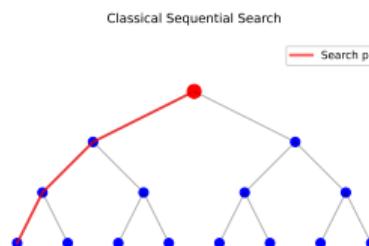
Проблема классических методов: пространство решений растёт экспоненциально — 2^n вариантов

Классический компьютер

В каждый момент времени находится в **одном** состоянии:

$$x = (1, 0, 1, 0, \dots)$$

Требуется последовательный перебор или применение эвристик



Квантовый компьютер

Находится в **суперпозиции** всех состояний одновременно:

$$|\psi\rangle = \sum_x \alpha_x |x\rangle$$

Позволяет исследовать всё пространство решений параллельно



Шаг 1: От битов к кубитам

Что такое кубит?

Кубит — квантовый бит, который может находиться в состояниях $|0\rangle$ и $|1\rangle$, а также в их суперпозиции

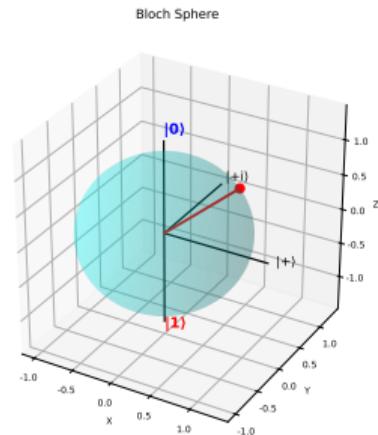
Классический бит: $x_i = 0$ или $x_i = 1$ (либо одно, либо другое)

Квантовый кубит:

$$|\psi_i\rangle = \alpha |0\rangle + \beta |1\rangle$$

Одновременно 0 И 1 с амплитудами α, β

Ключевая идея: Каждой переменной x_i сопоставляем кубит. n переменных = n кубитов = суперпозиция всех 2^n решений!



Сфера Блоха

Геометрическое представление состояния кубита

Шаг 2: Операторы Паули — работа с кубитами

Зачем они нужны? Для выполнения операций над кубитами используются операторы Паули — базовые квантовые преобразования.

Оператор \hat{Z} (измерение)

$$\hat{Z} |0\rangle = +1 \cdot |0\rangle$$

$$\hat{Z} |1\rangle = -1 \cdot |1\rangle$$

Что делает: дает $+1$ для $|0\rangle$ и -1 для $|1\rangle$

Зачем: закодировать значение переменной

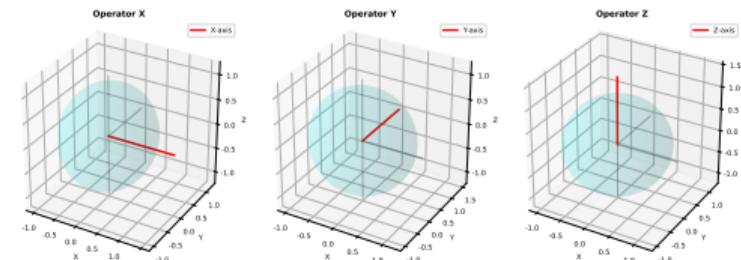
Оператор \hat{X} (переворот)

$$\hat{X} |0\rangle = |1\rangle$$

$$\hat{X} |1\rangle = |0\rangle$$

Что делает: меняет $|0\rangle \leftrightarrow |1\rangle$

Зачем: позволяет переходить между состояниями



Действие операторов Паули на сфере Блоха

Шаг 3: Кодируем переменные через операторы

Как связать $x_i \in \{0, 1\}$ и оператор \hat{Z}_i ?

Нам нужна формула, которая превращает \hat{Z}_i (собственные значения ± 1) в x_i (значения 0, 1):

$$x_i = \frac{1 - \hat{Z}_i}{2}$$

Проверка:

Если кубит в состоянии $|0\rangle$: $\hat{Z}_i = +1 \Rightarrow x_i = \frac{1-1}{2} = 0 \quad \checkmark$

Если кубит в состоянии $|1\rangle$: $\hat{Z}_i = -1 \Rightarrow x_i = \frac{1-(-1)}{2} = 1 \quad \checkmark$

Состояние кубита	Собственное значение \hat{Z}_i	Классическая переменная x_i
$ 0\rangle$	+1	0
$ 1\rangle$	-1	1

Вывод: Теперь вместо классических переменных x_i используем квантовые операторы \hat{Z}_i

Шаг 4: От функции стоимости к гамильтониану

Подставляем $x_i = \frac{1-\hat{Z}_i}{2}$ в QUBO

Было (классика):

$$C(x) = \sum_i c_i x_i + \sum_{i < j} c_{ij} x_i x_j$$

Стало (квантовая механика):

$$\hat{H}_C = \sum_i h_i \hat{Z}_i + \sum_{i < j} J_{ij} \hat{Z}_i \hat{Z}_j + \text{const}$$

Коэффициенты h_i, J_{ij} выражаются через исходные c_i, c_{ij}

Пример: для Max-Cut

- Классика: $C(x) = \sum_{(i,j) \in E} (x_i + x_j - 2x_i x_j)$
- Квант: $\hat{H}_C = -\frac{1}{2} \sum_{(i,j) \in E} (\mathbb{I} - \hat{Z}_i \hat{Z}_j)$

Главное: \hat{H}_C — гамильтониан стоимости. Энергия состояния $|x\rangle$ равна $C(x)!$

Что такое гамильтониан и зачем он нужен?

Гамильтониан = Энергия системы

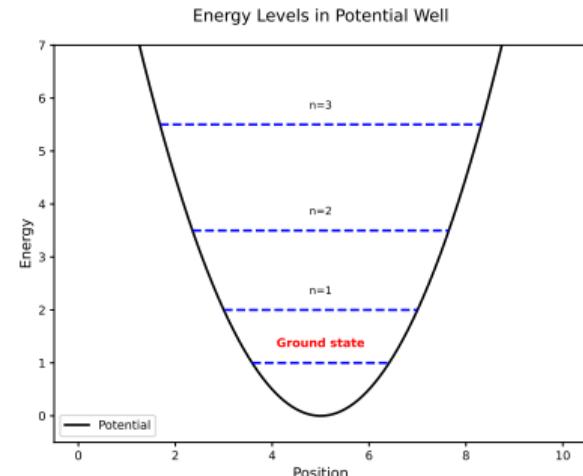
В классической механике: $H = T + V$ — сумма кинетической и потенциальной энергии. Система эволюционирует к минимуму энергии.

В квантовой механике: \hat{H} — оператор энергии. Определяет эволюцию системы во времени.

Основное состояние

Состояние с минимальной энергией называется **основным состоянием** (ground state)

Для оптимизации: \hat{H}_C построен так, что его основное состояние соответствует оптимальному решению QUBO!



Энергетический спектр

Основное состояние соответствует глобальному минимуму

Зачем уравнение Шрёдингера?

Как найти основное состояние \hat{H}_C ?

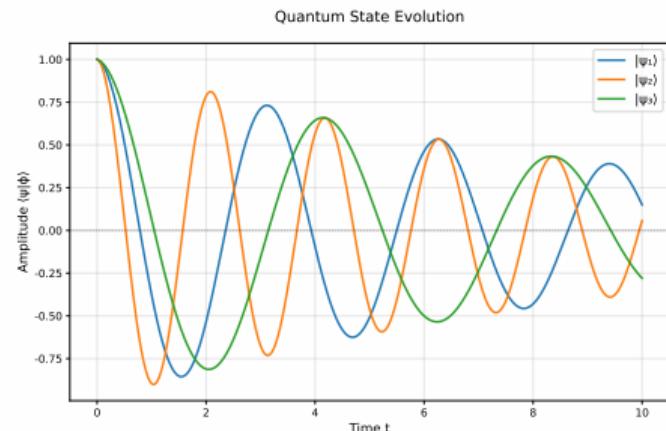
В классических методах мы перемещались по пространству решений. В квантовой механике состояние системы **эволюционирует** согласно уравнению Шрёдингера:

$$i\hbar \frac{\partial |\psi(t)\rangle}{\partial t} = \hat{H}(t) |\psi(t)\rangle$$

Что это значит? Гамильтониан $\hat{H}(t)$ определяет, как изменяется состояние $|\psi(t)\rangle$ со временем

Зачем нам это? При правильном выборе $\hat{H}(t)$ система эволюционирует к основному состоянию = оптимальному решению

Идея: Использовать квантовую эволюцию для поиска минимума энергии



Квантовая эволюция состояния

Адиабатический принцип

Как прийти к основному состоянию?

Адиабатическая теорема: при достаточно медленном изменении гамильтониана система остаётся в основном состоянии

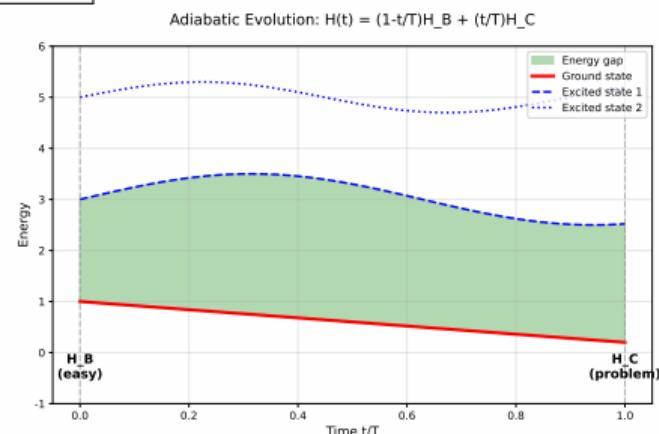
$$\hat{H}(t) = \left(1 - \frac{t}{T}\right) \hat{H}_B + \frac{t}{T} \hat{H}_C$$

В момент $t = 0$: $\hat{H}(0) = \hat{H}_B$ — простой гамильтониан с известным основным состоянием

В момент $t = T$: $\hat{H}(T) = \hat{H}_C$ — гамильтониан задачи

Между: Непрерывная интерполяция от \hat{H}_B к \hat{H}_C

Проблема: Для гарантии адиабатичности требуется $T \sim \exp(n)$ — экспоненциальное время!



Адиабатическая эволюция
Плавный переход от \hat{H}_B к \hat{H}_C

Что такое \hat{H}_B (оператор смешивания)?

Зачем нужен второй гамильтониан?

\hat{H}_B — гамильтониан с легко приготавливаемым основным состоянием

$$\hat{H}_B = - \sum_{i=1}^n \hat{X}_i$$

Основное состояние \hat{H}_B : равновесная суперпозиция всех битовых строк:

$$|s\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle$$

Как приготовить: применением вентилей Адамара ко всем кубитам в состоянии $|0\rangle^{\otimes n}$

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Роль \hat{H}_B : Создание суперпозиции различных решений, что обеспечивает исследование пространства

От адиабатического алгоритма к QAOA

Дискретизация времени

Идея QAOA: заменить непрерывную эволюцию на p дискретных шагов

Адиабатический алгоритм

Непрерывная эволюция от $t = 0$ до $t = T$

Требует $T \rightarrow \infty$

Сложно реализовать практически

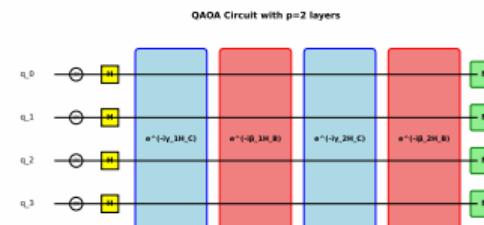
QAOA

p дискретных применений операторов

Фиксированное число шагов

Реализуем на NISQ устройствах

$$|\psi(\gamma, \beta)\rangle = \prod_{k=1}^p e^{-i\beta_k \hat{H}_B} e^{-i\gamma_k \hat{H}_C} |s\rangle$$



Что делают операторы эволюции?

Оператор стоимости $e^{-i\gamma \hat{H}_C}$

Действие: добавляет фазу $e^{-i\gamma C(x)}$ к каждому состоянию $|x\rangle$

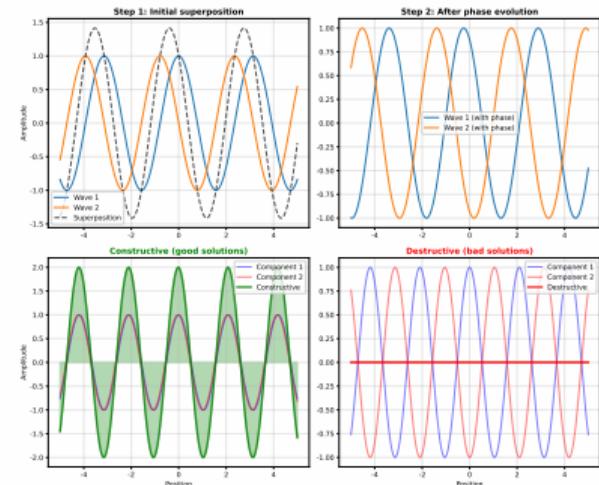
Эффект: состояния с малым $C(x)$ получают малую фазу, с большим — большую

Оператор смешивания $e^{-i\beta \hat{H}_B}$

Действие: создаёт суперпозицию различных битовых строк

Эффект: различные фазы приводят к интерференции состояний

Результат: Амплитуды оптимальных решений растут, субоптимальных — уменьшаются



Квантовая интерференция

Конструктивная интерференция для хороших решений

Откуда берутся параметры γ, β ?

Вариационный подход

Параметры $\gamma = (\gamma_1, \dots, \gamma_p)$ и $\beta = (\beta_1, \dots, \beta_p)$ подбираются для минимизации энергии:

$$F(\gamma, \beta) = \langle \psi(\gamma, \beta) | \hat{H}_C | \psi(\gamma, \beta) \rangle \rightarrow \min$$

Процесс оптимизации:

Выбираем начальные γ, β (случайно или эвристически)

Подготавливаем состояние $|\psi(\gamma, \beta)\rangle$ на квантовом процессоре

Проводим измерения ($N \sim 1000\text{--}10000$), вычисляем среднюю энергию F

Классический оптимизатор (COBYLA, Adam и др.) предлагает новые параметры

Итерируем до достижения сходимости

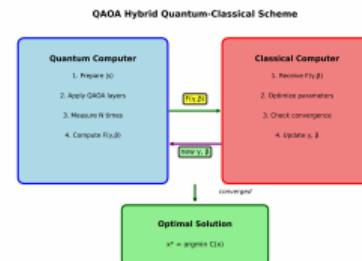
Полная схема QAOA

Квантовая часть

- 1 Начальное состояние: $|s\rangle$
- 2 Применяем p слоёв:
 $e^{-i\beta_k \hat{H}_B} e^{-i\gamma_k \hat{H}_C}$
- 3 Измеряем N раз
- 4 Вычисляем $F \approx \frac{1}{N} \sum C(x_i)$

Классическая часть

- 1 Получаем значение $F(\gamma, \beta)$
- 2 Оптимизатор предлагает новые параметры
- 3 Отправляем на квантовый процессор
- 4 Повторяем до сходимости



Гибридный алгоритм: Квантовая часть готовит состояние и измеряет, классическая — оптимизирует параметры

Роль параметра глубины p

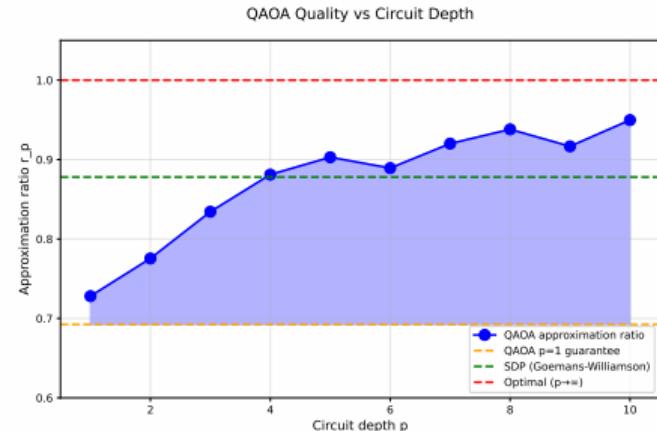
Что происходит при росте p ?

p	Параметры	Качество	Глубина
1	2	Грубое	Малая
2–3	4–6	Хорошее	Средняя
$\rightarrow \infty$	$2p$	Точное	Большая

Теорема (Farhi et al., 2014)

При $p \rightarrow \infty$ QAOA сходится к адиабатическому алгоритму и находит точное решение

На практике: Выбираем $p = 2–3$ как компромисс между качеством решения и глубиной схемы (шумы растут с глубиной)



Зависимость качества от p

Больше слоёв — точнее аппроксимация

Конкретный пример: Max-Cut

Задача: граф с 4 вершинами. Разделить на два множества, максимизируя число рёбер между ними.

Классическая формулировка:

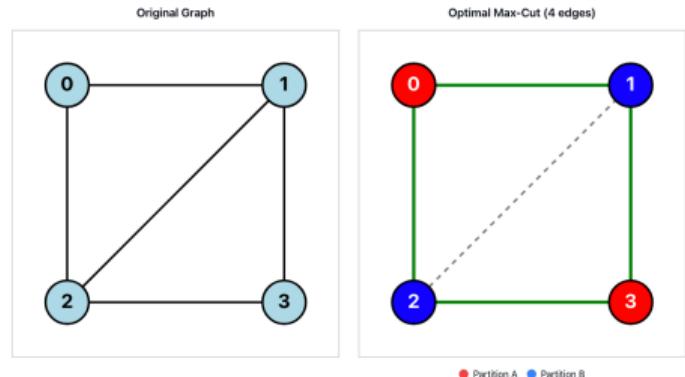
$$C(x) = \sum_{(i,j) \in E} (x_i + x_j - 2x_i x_j), \quad x_i = 0, 1$$

Квантовая формулировка:

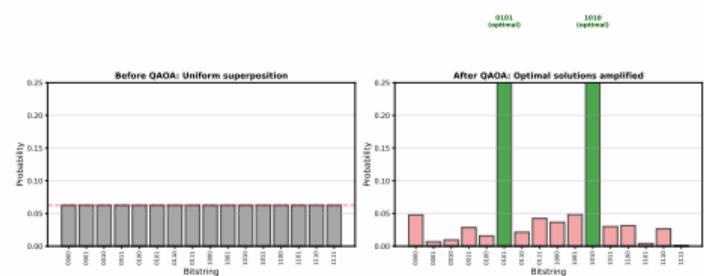
$$\hat{H}_C = -\frac{1}{2} \sum_{(i,j) \in E} (\mathbb{I} - \hat{Z}_i \hat{Z}_j)$$

Начальное состояние:

$$|s\rangle = \frac{1}{4} \sum_{x \in \{0,1\}^4} |x\rangle$$



Пример графа



Распределение после QAOA

Почему QAOA работает?

Квантовая интерференция

Суперпозиция: начальное состояние — равномерная суперпозиция всех возможных решений

Фазовая эволюция: оператор $e^{-i\gamma \hat{H}_C}$ присваивает различные фазы состояниям в зависимости от их энергии

Интерференция: оператор $e^{-i\beta \hat{H}_B}$ создаёт интерференцию между состояниями

Усиление: состояния с низкой энергией интерферируют конструктивно (амплитуды складываются), с высокой — деструктивно (компенсируются)

Измерение: при измерении с высокой вероятностью получаем одно из оптимальных решений

Суть: Квантовая интерференция используется для усиления вероятности оптимальных решений

Насколько хорошо QAOA решает задачи?

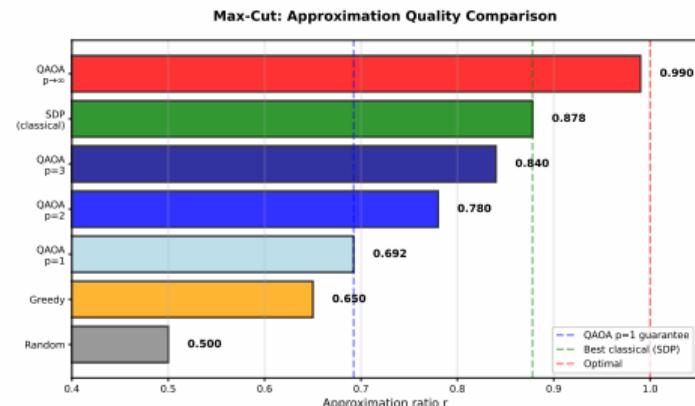
Апроксимационное соотношение

$$r_p = \frac{\langle \hat{H}_C \rangle_{\text{QAOA}}}{\min_x C(x)}$$

Показывает близость QAOA к оптимуму

Результаты для Max-Cut

Метод	Гарантия r
Случайный разрез	≈ 0.5
QAOA ($p = 1$)	≥ 0.6924
SDP (Goemans-Williamson)	≥ 0.878
QAOA ($p \rightarrow \infty$)	$\rightarrow 1.0$



Сравнение методов

QAOA показывает конкурентные результаты

Где мы сейчас: эра NISQ

Noisy Intermediate-Scale Quantum

Возможности устройств

50–1000 кубитов

Время когерентности: 10–100 мкс

Точность одиночных вентилей: 99.5–99.9%

Точность двухкубитных: 95–99%

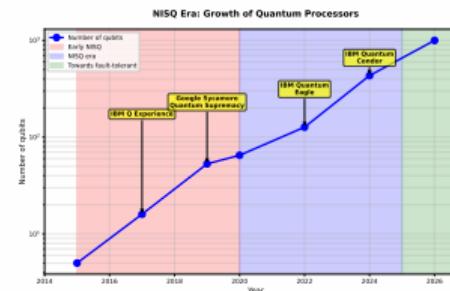
Почему QAOA подходит?

Короткие схемы ($p = 1–3$)

Вариационный подход

Тolerантность к шумам

Гибридная архитектура



Текущее состояние: QAOA реализован на реальных квантовых процессорах для задач до 50 кубитов. Пока уступает лучшим классическим методам, но разрыв сокращается.

Резюме: от QUBO к QAOA

Начали с QUBO — классической задачи оптимизации над битами $x_i \in \{0, 1\}$

Перешли к кубитам — каждой переменной x_i сопоставили кубит, получили суперпозицию решений

Использовали операторы Паули \hat{Z}_i для кодирования переменных

Построили гамильтониан \hat{H}_C — квантовую версию функции стоимости

Применили уравнение Шрёдингера для эволюции к основному состоянию

Дискретизировали адиабатическую эволюцию → получили QAOA

Применили квантовую интерференцию для усиления оптимальных решений

Гибридная оптимизация параметров γ, β даёт финальное решение

Литература и ресурсы

Основополагающие работы:

E. Farhi, J. Goldstone, S. Gutmann. *A Quantum Approximate Optimization Algorithm*. arXiv:1411.4028, 2014

E. Farhi, J. Goldstone, S. Gutmann, M. Sipser. *Quantum Computation by Adiabatic Evolution*. arXiv:quant-ph/0001106, 2000

Обзорные статьи:

L. Zhou et al. *Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices*. Physical Review X 10, 021067, 2020

S. Hadfield et al. *From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz*. Algorithms 12(2), 34, 2019

Реализации и эксперименты:

A. Bengtsson et al. *Improved Success Probability with Greater Circuit Depth for QAOA*. Physical Review Applied 14, 034010, 2020

IBM Quantum Experience: quantum-computing.ibm.com

Qiskit tutorials: qiskit.org/textbook

Спасибо за внимание!

Вопросы?