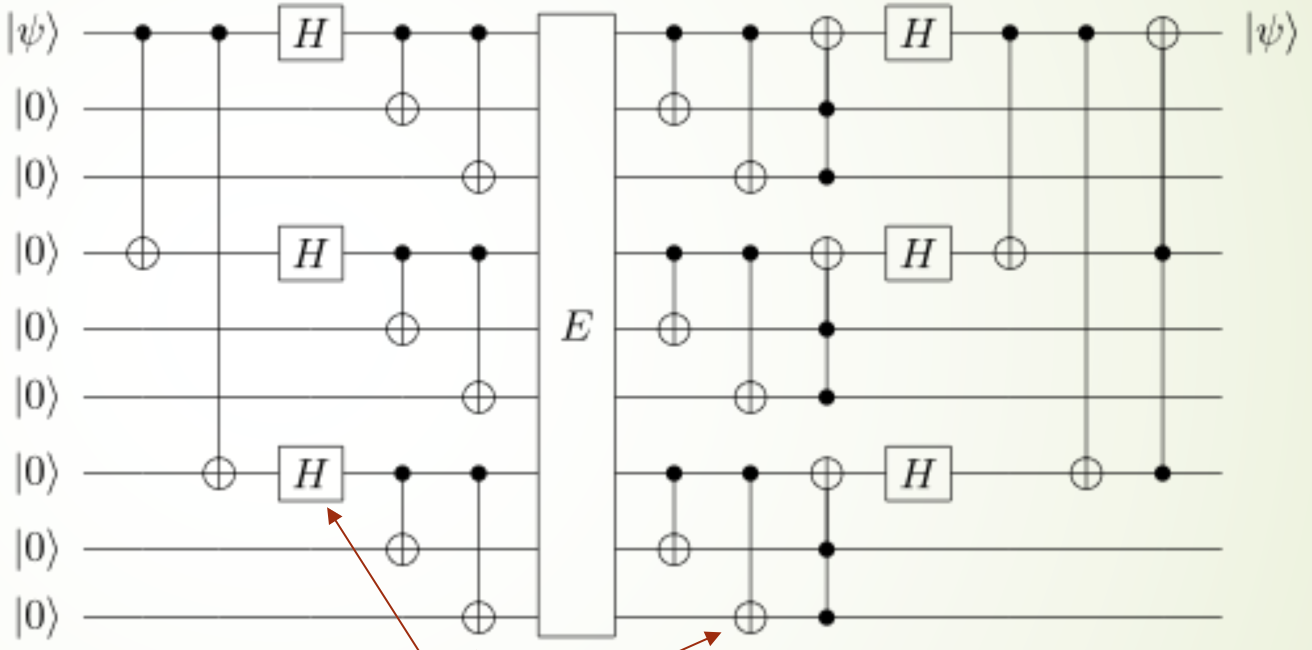
Квантовые вычисления

Модель квантовых вычислений: квантовые схемы



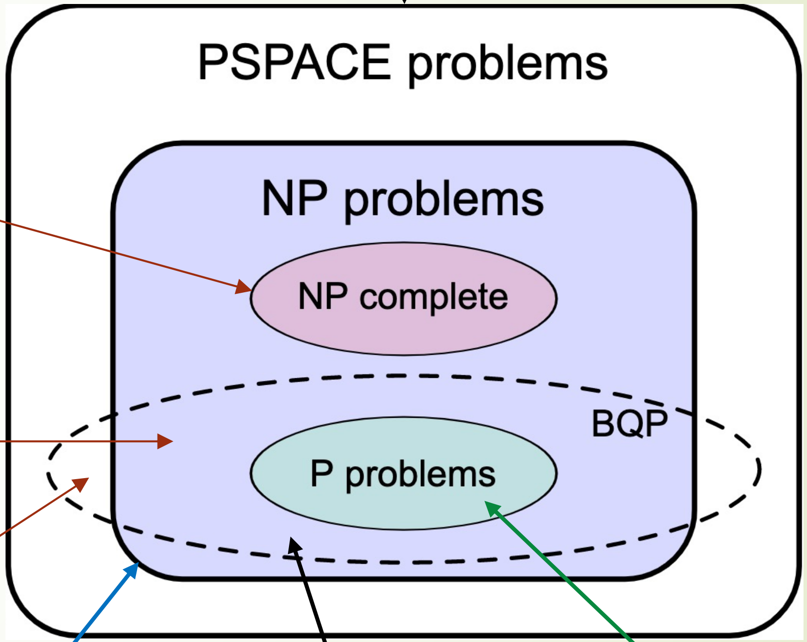
Кубиты

n кубитов кодируют вектор в пространстве $\mathbb{C}^{2^n}$

Квантовые элементы – унитарные операторы: $\mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$

Возможности квантовых компьютеров

Задачи, которые можно решать с небольшой памятью



- Алгоритм Гровера: (NP-complete) поиск набора, на котором булева фукнция от n переменных, заданная схемой, равна 1 за $O(2^{n/2})$ операций.
- Алгоритм Шора: (QBP ∩ NP) разложение n-битного числа на простые множители за время $O(n^3)$
- Симуляция сложных квантовых систем (BQP)

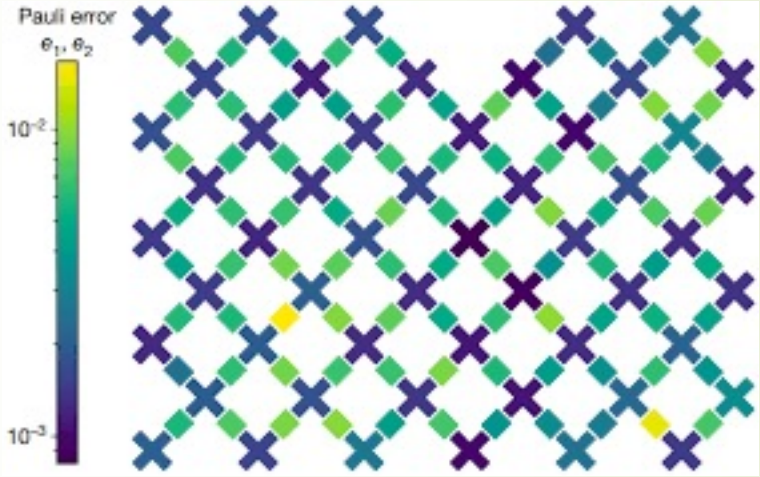
NP: Можно быстро проверить ответ

P: Можно быстро решать

BQP: Можно быстро решать на квантовом компьютере

Основные проблемы на пути создания квантовых компьютеров

- Сложно изолировать квантовую систему от взаимодействия с окружающей средой
- Все операции производятся с определённой вероятностью ошибки
- Двухкубитные операции можно производить только над кубитами, которые физически расположены рядом



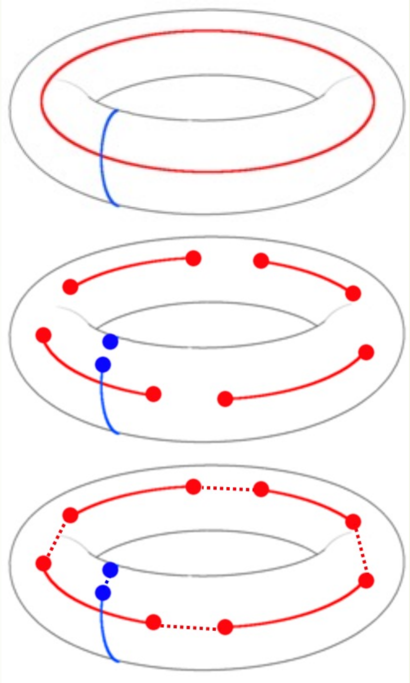
архитектура квантового компьютера Sycamore

Задача 1 – построение надежных квантовых схем: квантовые коды

- Кодирование квантового состояния:
  - Добавляем дополнительные кубиты
  - Периодически производим определенные измерения и выявляем произошедшие ошибки
  - Основываясь на результатах измерений, исправляем ошибки
- Желательные свойства кодов:
  - Локальность: каждое измерение для обнаружения ошибок должно затрагивать кубиты, расположенные в небольшой области.
  - Большое число исправляемых ошибок
  - Возможность эффективной коррекции ошибок

Пример: торический код

- Корректное состояние системы –
 объединение циклов на торе



- Ошибки в системе – разрывы циклов
  Измерение даёт концы фрагментов линий
- Коррекция ошибок – соединение разрывов линиями минимальной длины
- Торический код кодирует систему из 2-х кубитов
- Общее число кубитов N пропорционально площади тора
- Число исправляемых ошибок пропорционально минимальной длине нестягиваемого цикла на торе $\sim \sqrt{N}$

Задача 2 – симуляция квантовых схем

Способы симуляции:

1. Хранить в памяти целиком состояние квантовой системы и умножать на унитарные матрицы. Требует $O(2^n)$ памяти.
2. Свёртка тензорной сети. Квантовая схема представляет собой тензорную сеть, и можно выбрать оптимальный порядок свёртки тензоров так, чтобы минимизировать сложность.