量子算法概述：数学结构与理论边界

邵长鹏

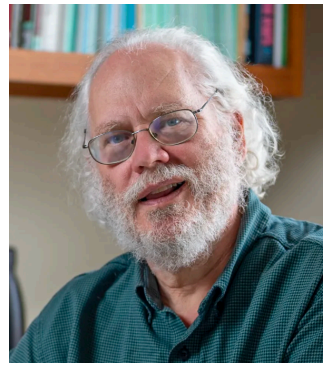
中科院数学院 – 数学机械化重点实验室

2025 年 6 月 27 日

关于这个报告

- 关于量子算法的一个简要综述报告（一条主线：线性代数）
- 偏数学的角度介绍
- 展现量子计算优势
- 希望通过这个报告让大家对量子算法产生新的认识

量子算法的重要性



- 1994 年，Peter Shor 提出分解大整数的量子算法
- 复杂度对比：分解 n-比特整数

	复杂度	$n = 2048$
Shor 算法（量子）	$O(n^2)$	几小时到几天
数域筛法（经典）	$O(e^{1.9n^{1/3}\log^{2/3} n})$	上亿年

- 影响：
  说明了量子计算在某些问题上可以击败经典计算——研究量子算法的一个核心目标

  RSA 等主流公钥加密体系受到威胁

Shor 算法改进

- 要求高：Shor 算法需要上百万个高质量的量子比特，目前只有几百个 (还不考虑量子纠错) [arXiv:2505.15917]
- O. Regev [arXiv:2308.06572]
  将量子门个数从 $O(n^2)$ 降低到 $O(n^{1.5})$
- G. Kahanamoku-Meyer, S. Ragavan, V. Vaikuntanathan, K. Van Kirk [arXiv:2412.12558]
  分解形如 p^2q 的整数，量子门个数 $O(n)$
- L. Brenner, L. Caha, X. Coiteux-Roy, R. Koenig [arXiv:2412.13164]
  连续时间的量子计算模型下，只需要 1 个量子比特，但这个过程所需的能量比一个拥有百万量子比特的量子计算机还要高出指数级

量子态

想象一下手里握着一枚硬币，会有两个状态：正面和反面

量子态模拟这两个状态，是它们的线性叠加

$$|\text{量子硬币}\rangle = \frac{1}{\sqrt{2}}|\text{正面}\rangle + \frac{1}{\sqrt{2}}|\text{反面}\rangle$$

- 观测：打开手看一下，看到正面或反面，概率各 1/2。

  观测会导致量子坍缩：从 2 个不确定的状态到 1 个确定的状态。

量子态

- 酉算子操作：保持单位性。例如作用 Hadamard 算子

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

则得到状态

$$|\text{正面}\rangle$$

这时将以概率 1 得到正面，也即是一个确定的状态了。

- 不难想象，当有 n 个硬币时，状态空间大小为 2^n（指数增加）

量子态（严格的数学定义）

一个 n 量子比特的量子态：

$$|\psi\rangle = \sum_{i_1,\ldots,i_n\in\{0,1\}} \psi_{i_1,\ldots,i_n}|i_1,\ldots,i_n\rangle,$$

满足

$$\sum_{i_1,\ldots,i_n\in\{0,1\}} |\psi_{i_1,\ldots,i_n}|^2 = 1, \quad \psi_{i_1,\ldots,i_n} \in \mathbb{C}.$$

$\psi_{i_1,\ldots,i_n}$ 称为振幅。

所允许的操作：

- 对 $|\psi\rangle$ 进行观测，则以概率 $|\psi_{i_1,\ldots,i_n}|^2$ 得到 $|i_1,\ldots,i_n\rangle$
- 对 $|\psi\rangle$ 作用酉算子 U，它满足 $UU^\dagger = U^\dagger U = I$

量子态：注解

有时为方便，直接记

$$|\psi\rangle = \sum_{x=0}^{2^n-1} \psi_x |x\rangle,$$

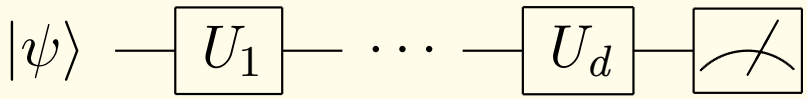
方便理解，从向量角度看，对应

$$\begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{bmatrix}$$

★ 但是，量子态和向量有着本质的差别

量子傅里叶变换 (quantum Fourier transform, QFT)

一个量子算法：从一个简单态 (如 $|0^n\rangle$) 出发，经过一系列酉算子操作，最后做观测，目标是以很大的概率得到想要的结果。



QFT 是量子算法设计中非常重要的一个酉算子

作用在 n 个量子比特上的 QFT:

$$|x\rangle \to \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i x y/2^n} |y\rangle$$

量子傅里叶变换 (quantum Fourier transform, QFT)

矩阵形式

$$\frac{1}{\sqrt{N}}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{bmatrix}$$

其中 $N = 2^n, \omega = e^{2\pi i/N}$

在量子线路上的实现代价：$O(n \log n)$. 对比 FFT: $O(2^n \log(2^n))$.

量子相位估计算法 (quantum phase estimation, QPE)

- Alexei Kitaev 在 1995 年提出 [arXiv:quant-ph/9511026]
- 是量子计算中最核心、最强大的算法之一，它构成了很多重要算法的基础，比如：
  - 大整数分解的 Shor 算法
  - 求解线性方程组的 HHL 算法

相位估计问题

给定一个酉算子 U 和一个本征态 $|\psi\rangle$ 满足

$$U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$$

目标：计算 $\theta \in [0, 1)$.

量子相位估计算法 (quantum phase estimation, QPE)

- 初始化 (n 由误差 ε 决定, $\varepsilon \approx 1/2^n$)

$$|0^n\rangle \otimes |\psi\rangle$$

- 作用 $H^{\otimes n}$ 产生叠加态

$$\frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle \otimes |\psi\rangle$$

- 作用控制变换 $\sum_j |j\rangle\langle j| \otimes U^j$

$$\frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle \otimes U^j|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{2\pi i\theta j}|j\rangle \otimes |\psi\rangle$$

量子相位估计算法 (quantum phase estimation, QPE)

- 作用量子傅里叶逆变换

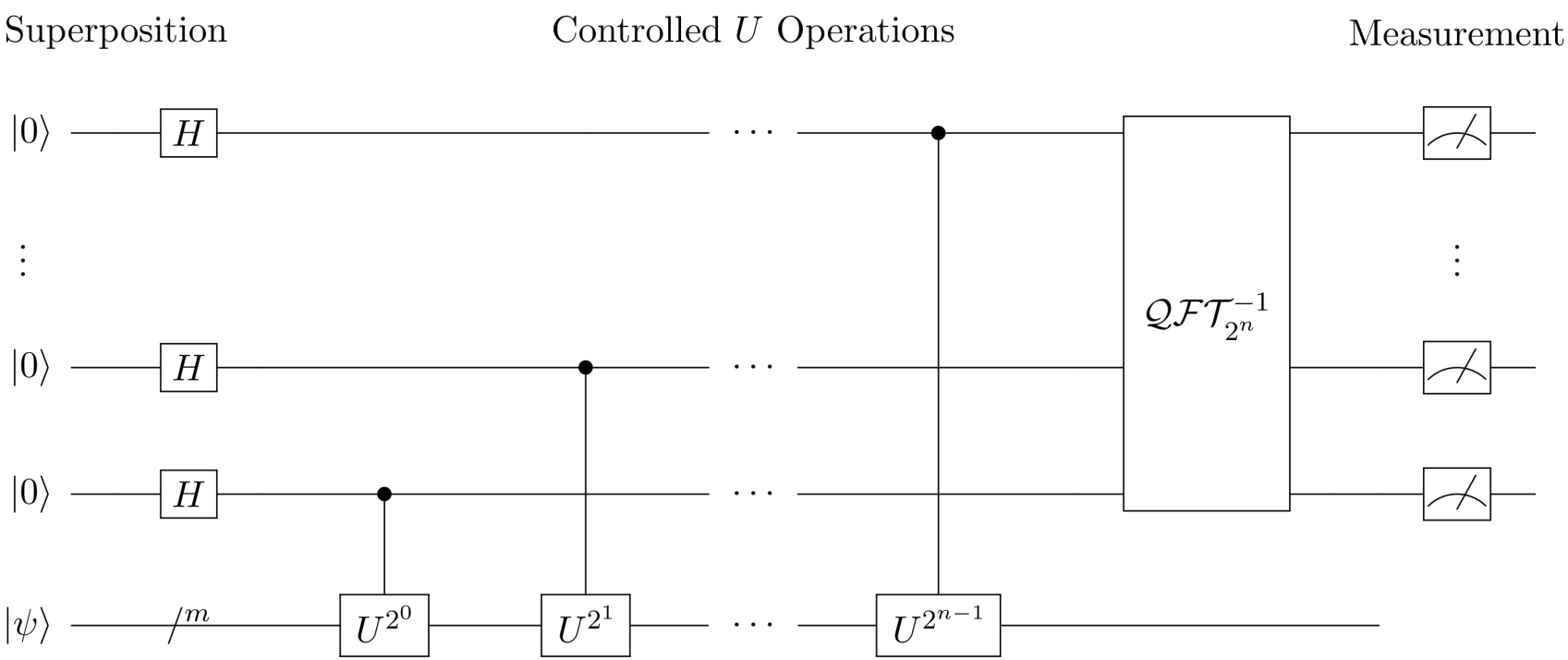
$$\frac{1}{2^n}\sum_{j,k=0}^{2^n-1} e^{2\pi i\theta j - 2\pi ijk/2^n}|k\rangle \otimes |\psi\rangle$$

$$= \sum_{k=0}^{2^n-1}\left(\frac{1}{2^n}\sum_{j=0}^{2^n-1} e^{2\pi ij(\theta - k/2^n)}\right)|k\rangle \otimes |\psi\rangle$$

- 对于满足 $\theta \approx k/2^n$ 的 $|k\rangle$ 的振幅大，因此观测时会以很大的概率得到。

- 复杂度：$O(n + 2^n + n\log n) = O(2^n) = O(1/\varepsilon)$，与维数无关。

QPE 的量子线路图


Superposition
Controlled U Operations
Measurement
$|0\rangle$
$|0\rangle$
$|0\rangle$
$|\psi\rangle$
H
H
H
$/^m$
U^{2^0}
U^{2^1}
$U^{2^{n-1}}$
$\mathcal{QFT}_{2^n}^{-1}$

量子相位估计算法的一些注解

- 可推广到厄米矩阵情形，这时 $U = e^{iH}$，称为哈密顿模拟 (Hamiltonian simulation)，后面会具体说。

- 可推广到的一般的态，而非本征态。具体地，设 U 的特征值和特征向量为 $\{e^{2\pi i\theta_j}, |\psi_j\rangle\}$，则任何一个态 $|\Psi\rangle$ 可分解成

$$|\Psi\rangle = \sum_{j=1}^{2^n} \beta_j |\psi_j\rangle$$

QPE 算法仍然适用，且不需要知道这个分解的具体形式。最后会近似得到

$$\sum_{j=1}^{2^n} \beta_j |\tilde{\theta}_j\rangle |\psi_j\rangle,$$

其中 $|\theta_j - \tilde{\theta}_j| \leq \varepsilon$.

计算特征值的一些进展

1. 正规矩阵 [arXiv:1610.06546]
2. 可对角化矩阵 [arXiv:1912.08015]
3. 矩阵只有实特征值或 Jordan 标准型 [arXiv:2401.06240]
4. 广义特征值问题 [arXiv:2010.15027]
5. 复特征值问题 [arXiv:2502.18119]
6. 幂法算法的量子加速 [arXiv:2405.14765]

量子相位估计算法的应用：求解线性方程组的 HHL 算法

由 Aram Harrow, Avinatan Hassidim 和 Seth Lloyd 在 2008 年提出 [arXiv:0811.3171]

求解线性方程组

给定一个 $n \times n$ 稀疏厄米矩阵 A 和一个向量 b，制备 $|x\rangle = |A^{-1}b\rangle$.

算法复杂度结果比较

HHL 算法	$O((\log n)\kappa^2/\varepsilon)$ 得到 $\|x\rangle$
共轭梯度法	$O(n\kappa)$ 得到 x

★ 通过 $|x\rangle$，可有效计算 $\langle x|M|x\rangle$ 或者 $\langle x|y\rangle$.

HHL 算法简述

- 设 A 的特征值和特征向量为 $\{\lambda_j, |u_j\rangle\}$，则 $|b\rangle = \sum_j \beta_j |u_j\rangle$.
- 由 QPE，可得到

$$\sum_{j=1}^{2^n} \beta_j |\tilde{\lambda}_j\rangle |\psi_j\rangle,$$

- 做受控旋转，得到 (选取 C 使得 $|C/\tilde{\lambda}_j| \leq 1$)

$$\sum_{j=1}^{2^n} \beta_j |\tilde{\lambda}_j\rangle |\psi_j\rangle \left(\frac{C}{\tilde{\lambda}_j} |0\rangle + \sqrt{1 - \frac{C^2}{\tilde{\lambda}_j^2}} |1\rangle \right)$$

- 做 QPE 的逆过程得到

$$C \sum_{j=1}^{2^n} \frac{\beta_j}{\tilde{\lambda}_j} |\psi_j\rangle |0\rangle + \cdots$$

关于 HHL 算法的一些注解

- 稀疏性要求是为了是的 e^{iA} 能在量子计算机上有效实现
- 厄米不是本质要求，因为可等价考虑

$$\begin{bmatrix} 0 & A \\ A^\dagger & 0 \end{bmatrix} \begin{bmatrix} 0 \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

- 有一系列改进，当前最优算法的复杂度为 $O(\kappa \log(n/\varepsilon))$
  [arXiv:1010.4458, 1511.02306, 1804.01973, 1806.01838, ...]
- 存在很多应用：机器学习、微分方程组求解、优化，等等

HHL 算法的应用：求解 ODE 的量子算法

- 考虑 N 维线性常微分方程组

$$\frac{dx(t)}{dt} = Ax(t) + b, \quad t \in [0, T]$$

已知 $x(0)$，求解 $x(T)$.

- 量子算法构造的核心思路：利用某些离散化方法，把问题转化为线性方程组求解问题，然后利用 HHL 或其改进算法

- 例如采用有限差分有

$$t_j = jh, \quad j = 0, 1, \ldots, M$$

$$\frac{x(t_{j+1}) - x(t_j)}{h} = Ax(t_j) + b, \quad j = 0, 1, \ldots, M-1.$$

HHL 算法的应用：求解 ODE 的量子算法

- 得到一个线性方程组 (例如 $M = 4$，令 $B = I + Ah$)

$$\begin{bmatrix} I & 0 & 0 & 0 & 0 \\ -B & I & 0 & 0 & 0 \\ 0 & -B & I & 0 & 0 \\ 0 & 0 & -B & I & 0 \\ 0 & 0 & 0 & -B & I \end{bmatrix} \begin{bmatrix} x(0) \\ x(h) \\ x(2h) \\ x(3h) \\ x(4h) \end{bmatrix} = \begin{bmatrix} x(0) \\ bh \\ bh \\ bh \\ bh \end{bmatrix}$$

- 得到 $|x(T)\rangle$ 的复杂度为 $\widetilde{O}(\kappa_V^5/\varepsilon^2)$，这里假定 $A = VDV^{-1}$ 可对角化，[arXiv:1010.2745]

其它更精细的离散化方法

泰勒展开 [arXiv:1701.03684]，方程 $\frac{dx(t)}{dt} = Ax(t) + b$ 的显示解

$$\begin{aligned} x(t) &= e^{At}x(0) + (e^{At} - I)A^{-1}b \\ &\approx \sum_{j=0}^{J} \frac{(At)^j}{j!}x(0) + \sum_{j=1}^{J} \frac{(At)^{j-1}}{j!}b \end{aligned}$$

同样需要对时间 t 离散化。复杂度为 $O(\kappa_V \log(N/\varepsilon))$

其它更精细的离散化方法

$$\begin{pmatrix} I & & & & & & & & & & \\ -Ah & I & & & & & & & & & \\ & -Ah/2 & I & & & & & & & & \\ & & -Ah/3 & I & & & & & & & \\ -I & -I & -I & -I & I & & & & & & \\ & & & & -Ah & I & & & & & \\ & & & & & -Ah/2 & I & & & & \\ & & & & & & -Ah/3 & I & & & \\ & & & & -I & -I & -I & -I & I & & \\ & & & & & & & & -I & I & \\ & & & & & & & & & -I & I \end{pmatrix} |x\rangle = \begin{pmatrix} |x_{\text{in}}\rangle \\ h|b\rangle \\ 0 \\ 0 \\ 0 \\ h|b\rangle \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

可以发现：在量子算法设计中，因为算法复杂度关于维数是 polylog 级别的，所以不在意扩大矩阵的规模。

其它更精细的离散化方法

谱方法 [arXiv:1901.00961], 对解的每个分量用切比雪夫多项式逼近

$$x_i(t) \approx \sum_{j=0}^{J} c_{i,j}\, T_j(t), \quad x_i'(t) \approx \sum_{j=0}^{J} d_{i,j}\, T_j(t)$$

得到关于 $c_{i,j}, d_{i,j}$ 的线性方程组。复杂度为 $O(\kappa_V \log(N/\varepsilon))$

求解非线性 ODE 的量子算法 [arXiv:2011.03185]

考虑耗散的 n 维二次 ODE

$$\frac{du}{dt} = F_2(u \otimes u) + F_1 u + F_0, \quad u(0) = u_0, \quad t \in [0, T],$$

其中 $u, F_0 \in \mathbb{R}^n, F_1 \in \mathbb{R}^{n\times n}, F_2 \in \mathbb{R}^{n\times n^2}$. 计算 $x(T)$.

Carleman 线性化：

$$\begin{aligned}\frac{d(u\otimes u)}{dt} &= \frac{du}{dt}\otimes u + u \otimes \frac{du}{dt} \\ &= (F_2 \otimes I + I \otimes F_2)(u \otimes u \otimes u) \\ &\quad +(F_1 \otimes I + I \otimes F_1)(u \otimes u) \\ &\quad +(F_0 \otimes I + I \otimes F_0)u\end{aligned}$$

求解非线性 ODE 的量子算法 [arXiv:2011.03185]

Carleman 线性化：计算并截断

$$\frac{du^{\otimes 3}}{dt}, \frac{du^{\otimes 4}}{dt}, \cdots, \frac{du^{\otimes N}}{dt}.$$

最后得到关于

$$u, u^{\otimes 2}, u^{\otimes 3}, \ldots, u^{\otimes N}$$

的一个线性 ODE 方程组，其中

$$N \approx \frac{\log(T\|F_2\|/\varepsilon)}{\log(1/\|u_0\|)}$$

求解非线性 ODE 的量子算法 [arXiv:2011.03185]

线性 ODE 方程组形如：

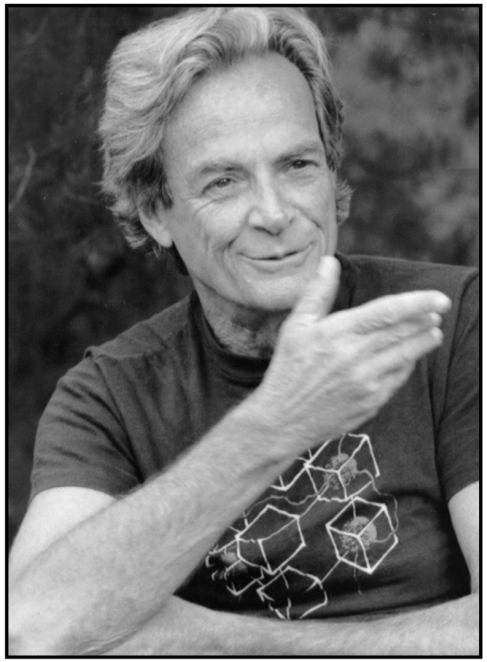
$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\hat{y}_1\\ \hat{y}_2\\ \hat{y}_3\\ \vdots\\ \hat{y}_{N-1}\\ \hat{y}_N\end{pmatrix}=\begin{pmatrix}A_1^1 & A_2^1 & & & & & \\ A_1^2 & A_2^2 & A_3^2 & & & & \\ & A_2^3 & A_3^3 & A_4^3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & A_{N-2}^{N-1} & A_{N-1}^{N-1} & A_N^{N-1} \\ & & & & A_{N-1}^N & A_N^N\end{pmatrix}\begin{pmatrix}\hat{y}_1\\ \hat{y}_2\\ \hat{y}_3\\ \vdots\\ \hat{y}_{N-1}\\ \hat{y}_N\end{pmatrix}+\begin{pmatrix}F_0(t)\\ 0\\ 0\\ \vdots\\ 0\\ 0\end{pmatrix}$$

量子算法复杂度：$\widetilde{O}(T^2/\varepsilon)$.

关于微分方程求解的一些进展

1. 哈密顿模拟的线性组合方法 [arXiv:2312.03916]
2. 薛定谔化方法 [arXiv:2212.13969]
3. Dyson 序列法 [arXiv:2212.03544]
4. Time-marching 方法 [arXiv:2208.06941]
5. 非线性方程的量子算法 [arXiv:2011.06571, 2205.01141]

哈密顿模拟 (Hamiltonian simulation)



"… nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."

Richard Feynman (1981)
Simulating physics with computers

图 1: "大自然不是经典的，该死的！如果你想模拟大自然，那你最好用量子力学来模拟。天哪，这真是个了不起的问题，因为它看起来可一点都不简单。"

哈密顿模拟 (Hamiltonian simulation)

哈密顿模拟问题

给定一个哈密顿量 H，时间参数 t 和一个初始状态 $|\psi(0)\rangle$，制备态

$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle,$$

也即求解薛定谔方程

$$i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle$$

哈密顿模拟的重要性

- 费曼提出研究量子计算的动机
- 很多量子算法的“子程序”
- BQP-完全问题的代表之一（量子计算机能够有效解决的最困难的一类问题）
- 在不远的将来，极有可能是展现量子优势的首个实用例子

哈密顿模拟的 Lie product 量子算法

假定 $H = \sum_{\ell=1}^{L} H_\ell$ 是一些简单厄米算子的线性组合，则可以利用 Lie product 公式，例如 $L = 2$

$$\lim_{r\to\infty} \left(e^{-iH_1 t/r} e^{-iH_2 t/r}\right)^r = e^{-i(H_1+H_2)t}$$

$$\left(e^{-iH_1 t/r} e^{-iH_2 t/r}\right)^r = e^{-i(H_1+H_2)t} + O(t^2\|H\|^2/r).$$

则可取 $r = O(t^2\|H\|^2/\varepsilon)$. [S. Lloyd, Science, 1996]

哈密顿模拟的 Lie product 量子算法

高阶公式给出更好的逼近结果 [arXiv:1912.08854]，例如二阶公式

$$(e^{-iH_1 t/2r} e^{-iH_2 t/r} e^{-iH_1 t/2r})^r = e^{-i(H_1+H_2)t} + O(t^3\|H\|^3/r^2).$$

这时 $r = O(t^{3/2}\|H\|^{3/2}/\varepsilon^{1/2})$. 一般 $2k$ 阶公式给出

$$r \approx 5^k L\|H\| t \left(\frac{L\|H\| t}{\varepsilon}\right)^{1/2k}.$$

下面将介绍一个更有效的算法/工具

量子奇异值变换 (Quantum singular value transformation, QSVT)

设 A 是厄米矩阵，$\|A\| \leq 1$,

$$U = \begin{bmatrix} A & * \\ * & * \end{bmatrix}$$

是一个酉矩阵，$f(x) : [-1, 1] \to [-1, 1]$ 是一个次数为 d 的多项式，则 QSVT 可有效构造出一个新的酉矩阵

$$\widetilde{U} = \begin{bmatrix} f(A) & * \\ * & * \end{bmatrix}.$$

且 $\widetilde{U}$ 形如（以 d 是偶数为例，$(\phi_1, \phi_2, \ldots, \phi_d)$ 由 $f(x)$ 决定）

$$\widetilde{U} = \prod_{j=1}^{d/2} \left(e^{i\phi_{2j-1} Z_1} U e^{i\phi_{2j} Z_1} U^\dagger \right)$$ “代价由 d 和 U 决定”

QSVT 的应用一：哈密顿模拟问题 e^{iAt}

设 $g(x) = e^{ixt}$，则 $g(A) = e^{iAt}$. 考虑 e^{ixt} 的多项式逼近

$$\left|\cos(xt) - J_0(t) + 2\sum_{k=1}^{R}(-1)^k J_{2k}(t)\,T_{2k}(x)\right| \le \varepsilon$$

$$\left|\sin(xt) - 2\sum_{k=1}^{R}(-1)^k J_{2k+1}(t)\,T_{2k+1}(x)\right| \le \varepsilon,$$

其中 $J_m(t)$ 是第一类贝塞尔函数，$T_m(x)$ 是切比雪夫多项式，

$$R \approx t + \frac{\log(1/\varepsilon)}{\log\log(1/\varepsilon)}$$

由 QSVT，可在 R 时间内求解哈密顿模拟问题，且最优。

QSVT 的应用二：线性方程组问题 A^{-1}

设 $g(x) = 1/x$, 定义域为 $[-1, 1/\kappa] \cup [1/\kappa, 1]$. 则 $g(A) = A^{-1}$.

考虑 $1/x$ 的多项式逼近

$$\left| \frac{1}{x} - 4 \sum_{j=0}^{J} (-1)^j \frac{\sum_{i=j+1}^{b} \binom{2b}{b+i}}{4^b} T_{2j+1}(x) \right| \leq 2\varepsilon$$

其中 $b \approx \kappa^2 \log(\kappa/\varepsilon), J = \sqrt{b \log(4b/\varepsilon)}$.

由 QSVT，可在 $\widetilde{O}(\kappa)$ 时间内求解线性方程组问题，且最优。

使用 QSVT 的难点

似乎问题转化成了多项式逼近问题，但是一个关键难点在于：需要事先能够有效地实现如下酉算子 (称为 block encoding)

$$U = \begin{bmatrix} A & * \\ * & * \end{bmatrix}$$

对于绝大部分 A 是不存在有效构造方式的，以下情形存在有效实现方式

1. A 是稀疏的
2. A 是一些简单酉算子的线性组合

QSVT 参考文献：[arXiv:1806.01838]

构造 block encoding 的 LCU 方法

设 $A = \sum_{j=0}^{L-1} \lambda_j U_j$ 是一些酉算子的线性组合，则可通过下述方式实现其 block encoding：

- 假定 $\lambda_j > 0$ 且 $\sum_j \lambda_j = 1$.
- 构造酉算子 V, 使其第一列为 $[\sqrt{\lambda_0}, \ldots, \sqrt{\lambda_{L-1}}]^T$,
- 令 $U = \sum_{j=0}^{L-1} |j\rangle\langle j| \otimes U_j = \mathrm{diag}(U_0, \ldots, U_{L-1})$,
- 则 A 的一个 block encoding 为

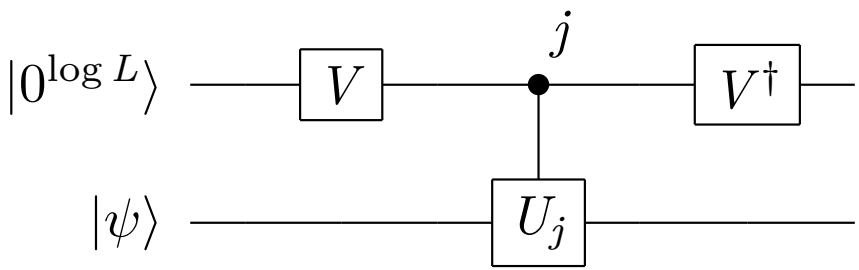
$$(V^\dagger \otimes I) U (V \otimes I) = \begin{bmatrix} A & * \\ * & * \end{bmatrix}$$

构造 block encoding 的 LCU 方法

矩阵形式

$$\begin{bmatrix} \sqrt{\lambda_0} & \sqrt{\lambda_1} & \cdots & \sqrt{\lambda_{L-1}} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} U_0 & & & \\ & U_1 & & \\ & & \ddots & \\ & & & U_{L-1} \end{bmatrix} \begin{bmatrix} \sqrt{\lambda_0} & \cdots & \cdots & \cdots \\ \sqrt{\lambda_1} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \sqrt{\lambda_{L-1}} & \cdots & \cdots & \cdots \end{bmatrix}$$

量子线路图



QSVT 的一些应用及技术改进

应用：

1. 半正定规划的量子加速 [arXiv:1705.01843]
2. 基态制备 [arXiv:2002.12508]
3. 统计学习中的概率分布性质测试 [arXiv:1902.00814]
4. 量子 PCP 问题 [arXiv:2111.09079]

改进：

1. 最优性证明 [arXiv:2311.06999]
2. QSVT 的推广/变形 [arXiv:2308.01501, 2104.01410]
3. 不依赖于 block encoding 的 QSVT 框架 [arXiv:2504.02385]

Bernstein-Vazirani 算法

Ethan Bernstein 和 Umesh Vazirani 在 1997 年提出。

Bernstein-Vazirani 问题

给定实现布尔函数 $f:\{0,1\}^n \to \{0,1\}$ 的一个 Oracle

$$U_f : |x\rangle \to (-1)^{f(x)}|x\rangle.$$

该函数满足 $f(x) = a \cdot x$ 其中 $a \in \{0,1\}^n$, 求 a.

算法复杂度结果比较

Bernstein-Vazirani 算法	1
经典算法	n

Bernstein-Vazirani 算法

1. 初始化 $|0^n\rangle$
2. 作用 $H^{\otimes n}$

$$\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle.$$

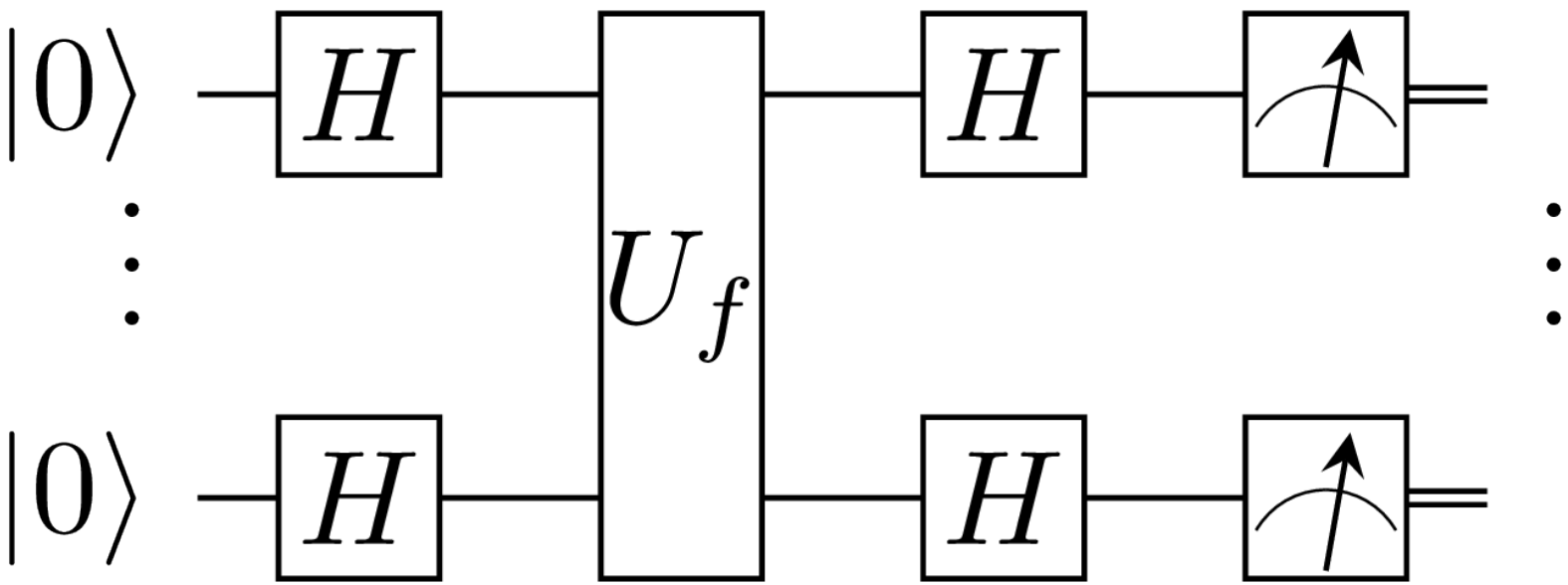
3. 作用 U_f

$$\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle.$$

4. 作用 $H^{\otimes n}$

$$\frac{1}{2^n}\sum_{x,y=0}^{2^n-1}(-1)^{f(x)+x\cdot y}|y\rangle=\frac{1}{2^n}\sum_{y=0}^{2^n-1}\sum_{x=0}^{2^n-1}(-1)^{x\cdot(a+y)}|y\rangle=|a\rangle.$$

Bernstein-Vazirani 算法：量子线路图



Jordan 的量子梯度算法及其改进

- 借助量子傅里叶变换，Bernstein-Vazirani 算法可推广到 $\mathbb{Z}_N$ 上，也即 $f(x) = g \cdot x$, 其中 $g \in \mathbb{Z}_N^d$.
- Jordan 算法 [arXiv:quant-ph/0405146]：近似线性函数的梯度 $f(x) \approx g \cdot x$. $O(1)$
- 高阶差分公式逼近 [arXiv:1711.00465]: $O(\sqrt{d}/\varepsilon)$

$$\sum_{\ell=1}^{m} \frac{\binom{m}{|\ell|}}{\binom{m+|\ell|}{|\ell|}} \frac{f(\ell x) - f(-\ell x)}{\ell} \approx \nabla f(0) \cdot x.$$

- 傅里叶谱方法逼近 [arXiv:2407.03833]: $O(1/\varepsilon)$

$$\frac{1}{N\delta} \sum_{k=0}^{N-1} e^{-2\pi i k/N} f(\delta e^{2\pi i k/N} x) \approx \nabla f(0) \cdot x.$$

一些应用

1. 凸优化的量子加速算法 [arXiv:1809.01731, 1809.00643]
2. 量子层析 (tomography) [arXiv:2207.08800]
3. 量子增强学习算法 [arXiv:2212.09328]
4. 多重期望值估计的量子算法 [arXiv:2111.09283]

其它未涉及到的点

1. 变分量子算法 (variational quantum algorithms)
2. 量子机器学习 (quantum machine learning)
3. 量子查询算法 (quantum query algorithms)
4. 量子复杂性理论 (quantum complexity theory)
5. 量子游走 (quantum walks)
6. ……

参考文献：
[1] Ronald de Wolf. Quantum Computing: Lecture Notes
[2] Andrew M. Childs. Lecture Notes on Quantum Algorithms

非 常 感 谢!