科学计算中的量子算法：线性方程组的量子算法 2

安冬

北京大学北京国际数学研究中心（BICMR）

andong@bicmr.pku.edu.cn

24-25 学年第 2 学期

大纲

- LCU 算法

量子线性方程组问题

经典: 令 A 是一个 N 乘 N 的厄米矩阵，b 是一个 N 维向量，求

$$x = A^{-1}b$$

- 非厄米情况：考虑 $\begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix}\begin{pmatrix} 0 \\ x \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix}$

量子: 制备一个量子态，以不超过 ϵ 的误差逼近

$$|x\rangle = \frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}$$

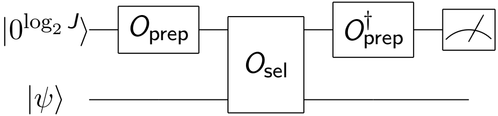
- 假设 $\|A\| = 1$，并且我们已知 A 的 $(1, a, 0)$-block-encoding 和 b 的态制备 oracle

参数: 维数 N, 误差 ϵ, 条件数 $\kappa = \|A\|\|A^{-1}\|$

LCU 算法

目标：改进线性方程组算法中关于 ϵ 的依赖

思路：考虑函数 $1/x$ 的展开，并利用 LCU 实现



$$O_{\mathsf{prep}} : |0\rangle \mapsto \frac{1}{\sqrt{\|\vec{c}\|_1}} \sum_{j=0}^{J-1} \sqrt{c_j}\, |j\rangle\,, \quad O_{\mathsf{sel}} = \sum_{j=0}^{J-1} |j\rangle\langle j| \otimes U_j$$

LCU 算法：Fourier

思路：将 $1/x$ 展开成 e^{-ixt} 的线性组合的形式

任取一个奇函数 $f(y)$，满足 $\int_0^\infty f(y)dy = 1$，那么

$$\frac{1}{x} = \int_0^\infty f(xy)dy$$

取 $f(y) = ye^{-y^2/2}$，并利用 Fourier 变换

$$f(y) = \frac{i}{\sqrt{2\pi}} \int_{-\infty}^\infty ze^{-z^2/2} e^{-iyz} dz,$$

我们有

$$\frac{1}{x} = \frac{i}{\sqrt{2\pi}} \int_0^\infty dy \int_{-\infty}^\infty dz\ ze^{-z^2/2} e^{-ixyz}$$

LCU 算法：Fourier

考虑一阶积分离散，可用 LCU 实现

$$\begin{aligned} A^{-1} &= \frac{i}{\sqrt{2\pi}} \int_0^{\infty} dy \int_{-\infty}^{\infty} dz\ z e^{-z^2/2} e^{-iyzA} \\ &\approx \frac{i}{\sqrt{2\pi}} \int_0^{Y} dy \int_{-Z}^{Z} dz\ z e^{-z^2/2} e^{-iyzA} \\ &\approx \frac{i}{\sqrt{2\pi}} \sum_{j=0}^{J-1} \sum_{k=-K}^{K-1} z_k e^{-z_k^2/2} e^{-iy_j z_k A} \Delta y \Delta z \end{aligned}$$

- Y, Z：两个正实数
- J, K：两个正整数
- $\Delta y = Y/J$, $\Delta z = Z/K, y_j = j\Delta y$, $z_k = k\Delta z$

LCU 输出：A^{-1} 的一个 (α, a, ϵ')-block-encoding

访问复杂度：与 α, Y, Z 和逼近误差 ϵ' 有关（与 J, K 的选取无关，我们选充分大的 J, K 使得积分离散的误差充分小）

LCU 算法：Fourier

$$\alpha = \frac{1}{\sqrt{2\pi}} \sum_{j=0}^{J-1} \sum_{k=-K}^{K-1} |z_k| e^{-z_k^2/2} \Delta y \Delta z \approx \frac{1}{\sqrt{2\pi}} \int_0^Y dy \int_{-Z}^{Z} dz \, |z| e^{-z^2/2} = \mathcal{O}(Y)$$

LCU 算法：Fourier

$$
\begin{aligned}
&\left|\frac{1}{x}-\frac{i}{\sqrt{2\pi}}\int_0^Y dy\int_{-Z}^{Z} dz\; z e^{-z^2/2} e^{-ixyz}\right| = \left|\frac{1}{x}-\frac{1}{\sqrt{2\pi}x}\int_{-Z}^{Z} dz\; e^{-z^2/2}(1-e^{-ixYz})\right| \\
&\lesssim \frac{1}{|x|}\int_Z^{\infty} e^{-z^2/2}dz + \frac{1}{|x|}\left|\int_{-\infty}^{\infty} e^{-z^2/2}e^{-ixYz}dz\right| \\
&\lesssim \frac{1}{|x|}e^{-Z^2/2} + \frac{1}{|x|}e^{-(xY)^2/2} \\
&\lesssim \kappa e^{-Z^2/2} + \kappa e^{-(Y/\kappa)^2/2}
\end{aligned}
$$

为了使误差小于 ϵ'，可取

$$
Y = \mathcal{O}(\kappa\sqrt{\log(\kappa/\epsilon')}), \quad Z = \mathcal{O}(\sqrt{\log(\kappa/\epsilon')})
$$

LCU 算法：Fourier

$$A^{-1} \approx \frac{i}{\sqrt{2\pi}} \sum_{j=0}^{J-1} \sum_{k=-K}^{K-1} z_k e^{-z_k^2/2} e^{-iy_j z_k A} \Delta y \Delta z$$

LCU 输出：A^{-1} 的一个 (α, a, ϵ')-block-encoding，$\alpha = \mathcal{O}(Y) = \mathcal{O}(\kappa\sqrt{\log(\kappa/\epsilon')})$

构造 A^{-1}：需要控制版本的 $e^{-iy_j z_k A} = e^{-i(y_j/Y)(z_k/Z)AYZ}$，精度为 ϵ'/κ

- 采用截断 Taylor，关于 A 的访问复杂度为 $\mathcal{O}(YZ\log(\kappa/\epsilon')) = \mathcal{O}(\kappa\log^2(\kappa/\epsilon'))$

LCU 算法：Fourier

算法：LCU 构造 A^{-1} 的 block-encoding，作用在 $|b\rangle$ 上，测量或振幅放大后测量

$$\frac{1}{\alpha}|0\rangle\,\|\widetilde{x}\|\,|\widetilde{x}\rangle + |\perp\rangle, \quad \widetilde{x} \approx A^{-1}|b\rangle, \quad \|\widetilde{x}\| \geq 1$$

- 为了使得 $|A^{-1}b\rangle$ 的误差小于 ϵ，可取 $\epsilon' \sim \epsilon$

总复杂度：

	A 的访问复杂度	$\|b\rangle$ 的访问复杂度	线路深度
无振幅放大	$\widetilde{\mathcal{O}}(\kappa^3 \log^3(1/\epsilon))$	$\widetilde{\mathcal{O}}(\kappa^2 \log(1/\epsilon))$	$\widetilde{\mathcal{O}}(\kappa \log^2(1/\epsilon))$
有振幅放大	$\widetilde{\mathcal{O}}(\kappa^2 \log^{2.5}(1/\epsilon))$	$\widetilde{\mathcal{O}}(\kappa \log^{0.5}(1/\epsilon))$	$\widetilde{\mathcal{O}}(\kappa^2 \log^{2.5}(1/\epsilon))$

- 结合变时间振幅放大（variable time amplitude amplification，VTAA），访问复杂度可改进为几乎最优：

$$\mathcal{O}(\kappa \text{ poly} \log(\kappa/\epsilon))$$

LCU 算法：多项式展开

考虑 $1/x$ 在 $[-1,-1/\kappa]\cup[1/\kappa,1]$ 上的多项式逼近

$$\frac{1}{x}\approx\frac{1-(1-x^2)^b}{x}$$

为了使得逼近误差小于 ϵ'，需要取

$$b\geq\kappa^2\log(\kappa/\epsilon')$$

若直接实现 A^k 并利用 LCU 实现多项式 $\frac{1-(1-A^2)^b}{A}$，LCU 中系数的 1 范数 $\sim 2^b$

- 原因：单项式 x^k 不是一组好的基底

改进方向：

- 好的基底
- 降低多项式次数

LCU 算法：Chebyshev

Chebyshev 多项式：

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x),$$

$$T_k(\cos\theta) = \cos(k\theta)$$

- 正交性（带 $\frac{1}{\sqrt{1-x^2}}$ 权重）
- 近似最优的 minimax 逼近

LCU 算法：Chebyshev

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

当 j 比较大时，$T_{2j+1}(x)$ 的系数很小，因此可以进一步截断多项式

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

$$J = \mathcal{O}(\sqrt{b\log(b/\epsilon')}) = \mathcal{O}(\kappa\log(\kappa/\epsilon'))$$

还可计算系数绝对值之和为 $\mathcal{O}(\kappa\log(\kappa/\epsilon'))$

LCU 算法：Chebyshev

$$A^{-1} \approx 4 \sum_{j=0}^{\mathcal{O}(\kappa \log(\kappa/\epsilon'))} (-1)^j \left(\frac{\sum_{i=j+1}^{b} \binom{2b}{b+i}}{2^{2b}} \right) T_{2j+1}(A)$$

难点：如何构造 $T_{2j+1}(A)$ 的 block-encoding（将在后续课程讲解）

- 实现 $T_{2j+1}(A)$ 的访问复杂度为 $\mathcal{O}(j)$

求解线性方程组的总访问复杂度：$\mathcal{O}(\kappa^2 \,\text{poly} \log(\kappa/\epsilon))$

- 与 Fourier 情况类似，也可通过 VTAA 改进为 $\mathcal{O}(\kappa \,\text{poly} \log(\kappa/\epsilon))$

LCU 算法小结

思路：将 $1/x$ 展开，并应用 LCU 实现

- Fourier
- Chebyshev

访问复杂度：$\mathcal{O}(\kappa^2 \,\mathsf{poly}\log(\kappa/\epsilon))$

- 可用 VTAA 改进为 $\mathcal{O}(\kappa \,\mathsf{poly}\log(\kappa/\epsilon))$

阅读

阅读：

- arXiv: 1511.02306