

# 科学计算中的量子算法：哈密顿量模拟

安冬

北京大学北京国际数学研究中心 (BICMR)

*andong@bicmr.pku.edu.cn*

25-26 学年第 2 学期

# 大纲

- ▶ 不含时哈密顿量模拟 (Time-independent Hamiltonian simulation)
  - ▶ 乘积公式 (Product Formula)
  - ▶ 截断泰勒方法 (Truncated Taylor method)
  - ▶ QSVT
  - ▶ 其他方法
- ▶ 含时哈密顿量模拟 (Time-dependent Hamiltonian simulation)

# 不含时哈密顿量模拟

$$i\frac{d|\psi(t)\rangle}{dt} = H|\psi(t)\rangle, \quad |\psi(0)\rangle = |\psi_0\rangle.$$

$$|\psi(t)\rangle = U(t)|\psi_0\rangle, \quad U(t) = e^{-iHt}.$$

**目标：** 已知厄米矩阵  $H$  的一个输入模型，构造量子线路  $V$  使得

$$\|V - U(T)\| \leq \epsilon.$$

**复杂度：** 关于  $H$  的访问复杂度，门复杂度，线路深度

**参数：** 范数  $\|H\|$ ，模拟时间  $T$ ，误差  $\epsilon$

# 乘积公式

假设

$$H = H_0 + H_1$$

并假设  $e^{-iH_0 t}$  和  $e^{-iH_1 t}$  比较简单.

例子:

- ▶ Transverse field Ising model (TFIM):

$$H = - \sum_{j=0}^{n-2} Z_j Z_{j+1} - g \sum_{j=0}^{n-1} X_j$$

- ▶ 时空薛定谔方程:

$$H = -\Delta + V(x)$$

# 一阶乘积公式

思路：把  $e^{-iHt}$  分解成  $e^{-iH_0s}$  和  $e^{-iH_1s}$  的乘积

Lie-Trotter 乘积公式：

$$e^{-iHT} = \lim_{L \rightarrow \infty} \left( e^{-iH_1 T/L} e^{-iH_0 T/L} \right)^L$$

1 阶 Trotter 公式：

$$U_1(t) = e^{-iH_1 t} e^{-iH_0 t}$$

算法：把  $[0, T]$  等分成  $L$  份，时间步长  $t = T/L$ ,

$$U(T) \approx (U_1(t))^L$$

## 一阶乘积公式：误差分析

局部舍入误差（可用泰勒展开证明）：

$$\|e^{-iHt} - e^{-iH_1 t} e^{-iH_0 t}\| \leq \mathcal{O}(t^2)$$

全局误差：

$$\|U(T) - U_1(t)^L\| \leq \mathcal{O}(Lt^2) = \mathcal{O}(T^2/L)$$

为了使误差小于  $\epsilon$ ，访问复杂度为

$$L = \mathcal{O}(T^2/\epsilon)$$

## 二阶乘积公式

2 阶 Trotter/对称 Trotter/Strang 公式:

$$U_2(t) = e^{-iH_0 t/2} e^{-iH_1 t} e^{-iH_0 t/2}$$

局部舍入误差 (可用泰勒展开证明):

$$\|e^{-iHt} - e^{-iH_0 t/2} e^{-iH_1 t} e^{-iH_0 t/2}\| \leq \mathcal{O}(t^3)$$

全局误差:

$$\|U(T) - U_2(t)^L\| \leq \mathcal{O}(Lt^3) = \mathcal{O}(T^3/L^2)$$

为了使误差小于  $\epsilon$ , 访问复杂度为

$$L = \mathcal{O}(T^{3/2}/\epsilon^{1/2})$$

## 高阶乘积公式：Trotter-Suzuki

Suzuki 迭代：

$$U_{2p+2}(t) = U_{2p}(s_p t)^2 U_{2p}((1 - 4s_p)t) U_{2p}(s_p t)^2, \quad s_p = (4 - 4^{1/(2p+1)})^{-1}$$

局部舍入误差：

$$\|e^{-iHt} - U_{2p}(t)\| \leq \mathcal{O}(t^{2p+1})$$

全局误差：

$$\|U(T) - U_{2p}(t)^L\| \leq \mathcal{O}(Lt^{2p+1}) = \mathcal{O}(T^{2p+1}/L^{2p})$$

为了使误差小于  $\epsilon$ ，访问复杂度为

$$L = \mathcal{O}(T^{1+1/(2p)} / \epsilon^{1/(2p)})$$

- ▶ 每一个  $U_{2p}(t)$  包含指数多项 ( $3 \times 5^{p-1}$  项)
- ▶ 无法通过优化  $p$  改进到  $\mathcal{O}(T \text{ poly log}(T/\epsilon))$

## 高阶乘积公式：一般形式

一般形式（系数  $\alpha_j, \beta_j$  由阶条件确定）：

$$U_{\text{PF}}(t) = \prod_{j=0}^{J-1} e^{-i\beta_j t H_1} e^{-i\alpha_j t H_0}$$

$p$  阶乘积公式的访问复杂度：

$$L = \mathcal{O}(T^{1+1/p}/\epsilon^{1/p})$$

- ▶ 仍必须包含指数多项，因此无法通过优化  $p$  改进到  $\mathcal{O}(T \text{ poly } \log(T/\epsilon))$

## 乘积公式：Commutator scaling

交换子 (commutator):

$$[A, B] = AB - BA$$

$$U_1(t) = e^{-iH_1 t} e^{-iH_0 t}$$

直观：当  $[H_0, H_1] = 0$  时， $U_1(t) = U(t)$

## 乘积公式：Commutator scaling

1 阶 Trotter/2 阶 Trotter:

$$\|e^{-iHt} - U_1(t)\| \leq \mathcal{O}(t^2 \| [H_0, H_1] \|)$$

$$\|e^{-iHt} - U_2(t)\| \leq \mathcal{O}(t^3 (\| [H_0, [H_0, H_1]] + [H_1, [H_1, H_0]] \|))$$

为了使误差小于  $\epsilon$ , 访问复杂度为

$$L_1 = \mathcal{O}(\| [H_0, H_1] \| T^2 / \epsilon)$$

$$L_2 = \mathcal{O}((\| [H_0, [H_0, H_1]] + [H_1, [H_1, H_0]] \|)^{1/2} T^{3/2} / \epsilon^{1/2})$$

► 证明：常数变易公式 + 泰勒展开

## 乘积公式：Commutator scaling

高阶乘积公式：

$$\|e^{-iHt} - U_p(t)\| \leq \mathcal{O}(t^{p+1}\gamma_p), \quad \gamma_p = \sum_{\nu_0, \nu_1, \dots, \nu_p \in \{0,1\}} \|[H_{\nu_p}, \dots [H_{\nu_2}, [H_{\nu_1}, H_{\nu_0}]] \dots]\|$$

访问复杂度：

$$L = \mathcal{O}(\gamma_p^{1/p} T^{1+1/p} / \epsilon^{1/p})$$

注意到  $\gamma_p^{1/p} \leq \mathcal{O}((\|H_0\| + \|H_1\|)^{1+1/p})$ ，但具体例子中往往会更好

## 乘积公式小结

$$e^{-iHt} \approx U_{\text{PF}}(t) = \prod_{j=0}^{J-1} e^{-i\beta_j t H_1} e^{-i\alpha_j t H_0}$$

- ▶ 1 阶 Trotter, 2 阶 Trotter, Trotter-Suzuki
- ▶ 访问复杂度:  $\mathcal{O}(\gamma_p^{1/p} T^{1+1/p}/\epsilon^{1/p})$ , 具有 commutator scaling
- ▶ 实现方式简单, 仍然是最广泛应用的哈密顿量模拟算法 (尤其是低阶 Trotter)

## 截断泰勒方法

$$U(t) = e^{-iHt}$$

假设已知  $H$  的一个  $(\alpha, a, 0)$ -block-encoding, 记为

$$U_H = \begin{pmatrix} H/\alpha & * & \cdots \\ * & * & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

## 截断泰勒方法：算法

考虑泰勒展开：

$$e^{-iHt} \approx \sum_{k=0}^{K-1} \frac{(-iHt)^k}{k!} = \sum_{k=0}^{K-1} \frac{(\alpha t)^k}{k!} (-iH/\alpha)^k$$

算法思路：利用 LCU 实现泰勒展开

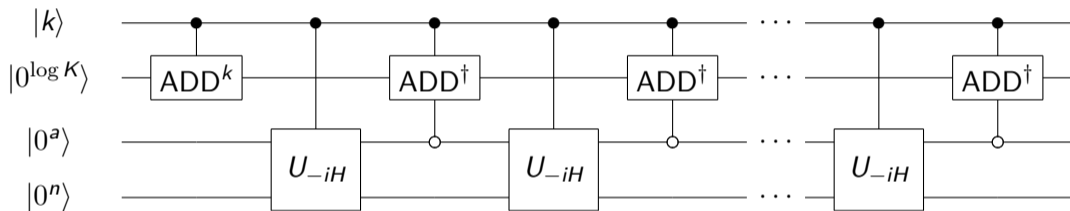
$$O_{\text{prep}} : |0\rangle \mapsto \frac{1}{\sqrt{\|c\|_1}} \sum_{k=0}^{K-1} \sqrt{c_k} |k\rangle, \quad c_k = \frac{(\alpha t)^k}{k!}$$

$$O_{\text{sel}} = \sum_{k=0}^{K-1} |k\rangle \langle k| \otimes U_k,$$

其中  $U_k$  是  $(-iH/\alpha)^k$  的  $(1, a', 0)$ -block-encoding

## 截断泰勒方法：算法

$O_{\text{sel}}$  的实现：compression gadget



- ▶  $\text{ADD}^k$  受第一寄存器控制，可用类似于 QPE 线路的思想实现
- ▶ 第  $j$  组  $U_{-iH}$  和  $\text{ADD}^\dagger$  受控条件为  $k > j, j \in [K]$

访问复杂度： $\mathcal{O}(K)$ ， 额外量子比特： $\mathcal{O}(a + \log K)$

## 截断泰勒方法：算法

$$\begin{aligned}(O_{\text{prep}}^\dagger \otimes I) U_{\text{sel}} (O_{\text{prep}} \otimes I) |0^{\log K}\rangle |0^{a'}\rangle |\psi\rangle &= \frac{1}{\|c\|_1} |0^{\log K}\rangle |0^{a'}\rangle \left( \sum_{k=0}^{K-1} \frac{(-iHt)^k}{k!} \right) |\psi\rangle + |\perp\rangle \\ &\approx \frac{1}{\|c\|_1} |0^{\log K}\rangle |0^{a'}\rangle e^{-iHt} |\psi\rangle + |\perp\rangle\end{aligned}$$

存在的问题：  $t$  不能取太大，否则  $\|c\|_1 \sim e^{\alpha t}$  是指数大的

## 截断泰勒方法：算法

$$(O_{\text{prep}}^\dagger \otimes I) U_{\text{sel}} (O_{\text{prep}} \otimes I) |0^{\log K}\rangle |0^{a'}\rangle |\psi\rangle \approx \frac{1}{\|c\|_1} |0^{\log K}\rangle |0^{a'}\rangle e^{-iHt} |\psi\rangle + |\perp\rangle$$

把  $[0, T]$  划分成  $L = 2\alpha T$  份，时间步长  $t = T/L$  满足  $\alpha t \leq 1/2$

$$\frac{1}{\|c\|_1} |0^{\log K}\rangle |0^{a'}\rangle e^{-iHt} |\psi\rangle + |\perp\rangle \rightarrow \frac{1}{2} |0^{\log K}\rangle |0^{a'+1}\rangle e^{-iHt} |\psi\rangle + |\perp\rangle$$

新的问题：成功概率随时间步数量  $M$  的增长而指数衰减

解决方法：oblivious amplitude amplification (OAA)

## 截断泰勒方法：最终算法

取  $L = 2\alpha T$ ,  $t = T/L = 1/(2\alpha)$

1. 利用 LCU 实现  $V|0\rangle|\psi\rangle \approx \frac{1}{2}|0\rangle e^{-iHt}|\psi\rangle + |\perp\rangle$
2. 利用一步 (robust) OAA 实现  $\tilde{V}|0\rangle|\psi\rangle \approx |0\rangle e^{-iHt}|\psi\rangle$
3. 重复前两步  $L$  次, 实现  $\tilde{V}^L|0\rangle|\psi\rangle \approx |0\rangle e^{-iHT}|\psi\rangle$

访问复杂度:  $\mathcal{O}(LK) = \mathcal{O}(\alpha TK)$

额外量子比特数:  $\mathcal{O}(a + \log K)$

## 截断泰勒方法：复杂度分析

根据泰勒公式（并利用  $K^K \leq e^K K!$ ），

$$\begin{aligned} \left\| e^{-iHt} - \sum_{k=0}^{K-1} \frac{(-iH)^k}{k!} t^k \right\| &= \left\| \int_0^t \frac{(-iH)^K}{(K-1)!} (t-s)^{K-1} ds \right\| \\ &\leq \frac{\|H\|^K t^K}{K!} \leq \frac{1}{2^K K!} \leq \left(\frac{e}{2K}\right)^K \end{aligned}$$

为了使得局部截断误差小于  $\epsilon' = \epsilon/L = \epsilon/(2\alpha T)$ ，可以取

$$K = \mathcal{O}\left(\frac{\log(1/\epsilon')}{\log \log(1/\epsilon')}\right) = \mathcal{O}\left(\frac{\log(\alpha T/\epsilon)}{\log \log(\alpha T/\epsilon)}\right)$$

访问复杂度： $\mathcal{O}\left(\alpha T \frac{\log(\alpha T/\epsilon)}{\log \log(\alpha T/\epsilon)}\right)$

额外量子比特数： $\mathcal{O}(a + \log \log(\alpha T/\epsilon))$

$$i\frac{du}{dt} = Hu, \quad u(t) = e^{-iHt} |u_0\rangle, \quad \|H\| \leq 1$$

$$e^{-iHt} = \cos(Ht) - i\sin(Ht)$$

**Taylor 展开:**

$$\cos(tx) = \sum_{k=0}^{\infty} (-1)^k \frac{t^{2k}}{(2k)!} x^{2k}, \quad \sin(tx) = \sum_{k=0}^{\infty} (-1)^k \frac{t^{2k+1}}{(2k+1)!} x^{2k+1}$$

**Jacobi-Anger 展开:**

$$\cos(tx) = J_0(t) + 2 \sum_{k=1}^{\infty} (-1)^k J_{2k}(t) T_{2k}(x), \quad \sin(tx) = 2 \sum_{k=0}^{\infty} (-1)^k J_{2k+1}(t) T_{2k+1}(x)$$

►  $J_\alpha$  : 第一类 Bessel 函数,  $J_\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m+\alpha+1)} \left(\frac{x}{2}\right)^{2m+\alpha}$

# QSVT

## Lemma (三角函数的多项式近似)

对于任意的  $t \in \mathbb{R}$ ,  $\epsilon \in (0, e^{-1})$ , 存在次数为  $d = \mathcal{O}(t + \frac{\log(1/\epsilon)}{\log(e + \log(1/\epsilon)/t)})$  的偶实多项式  $p_0(x)$  和奇实多项式  $p_1(x)$ , 满足

$$\sup_{x \in [-1, 1]} |\cos(tx) - p_0(x)| \leq \epsilon$$

$$\sup_{x \in [-1, 1]} |\sin(tx) - p_1(x)| \leq \epsilon$$

### 算法:

1. 利用 QSVT 实现  $\frac{1}{1+\epsilon}p_0(H)$  和  $\frac{1}{1+\epsilon}p_1(H)$  的 block-encoding
2. 利用 LCU 实现  $\frac{1}{2(1+\epsilon)}(p_0(H) + ip_1(H))$  的 block-encoding, 即为  $e^{-iHt}$  的  $(2(1 + \epsilon), a + 2, \epsilon)$ -block-encoding

访问复杂度:  $\mathcal{O}(T + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)})$

## 截断泰勒方法和 QSVT: 小结

截断泰勒方法: 泰勒公式 + LCU + OAA

- ▶ 访问复杂度:  $\mathcal{O}\left(\alpha T \frac{\log(\alpha T/\epsilon)}{\log \log(\alpha T/\epsilon)}\right)$
- ▶ 额外量子比特数:  $\mathcal{O}(a + \log \log(\alpha T/\epsilon))$

QSVT:

- ▶ 访问复杂度:  $\mathcal{O}\left(\alpha T + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$
- ▶ 额外量子比特数:  $a + 2$

注:

- ▶ 复杂度下界:  $\Omega\left(\alpha T + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$  (QSVT 达到了最优)
- ▶ 加性 vs 乘性
- ▶ 截断泰勒方法可推广到含时哈密顿量模拟

## 乘积公式 vs QSVT

乘积公式复杂度:

$$\mathcal{O}(\gamma_p^{1/p} T^{1+1/p} / \epsilon^{1/p}), \quad \gamma_p = \sum_{\nu_0, \nu_1, \dots, \nu_p \in \{0,1\}} \|[H_{\nu_p}, \dots [H_{\nu_2}, [H_{\nu_1}, H_{\nu_0}]] \dots]\|$$

QSVT 复杂度:

$$\mathcal{O}\left(\|H\| T + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$$

	$T$	$\epsilon$	Commutator
乘积公式	$\mathcal{O}(T^{1+1/p})$	$\mathcal{O}(1/\epsilon^{1/p})$	有
QSVT	$\mathcal{O}(T)$	$\mathcal{O}(\log(1/\epsilon))$	无

## 乘积公式 vs QSVT: 例子

$$-\partial_x^2 + V(x), \quad \text{周期边界}, \quad \max |V(x)| = \mathcal{O}(1)$$

采用  $N$  个格点的中心差分离散后得到  $H = H_0 + H_1$

$$\|H\| = \mathcal{O}(N^2), \quad \|[H_0, H_1]\| = \mathcal{O}(N)$$

(后者的连续版本推导:  $[-\partial_x^2, V] = -V'' - 2V\partial_x$ )

一阶乘积公式  $\mathcal{O}(NT^2/\epsilon)$

$$\text{QSVT: } \mathcal{O}\left(N^2 T + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$$

## 其他方法

**LCU 的随机化实现：** 减少线路深度、量子比特数量和控制操作

**多乘积公式：** 结合乘积公式 commutator 依赖和截断泰勒的  $T, \epsilon$  依赖

▶ 思路：外推法

$$U_{\text{MPF}}(t) = \sum_{j=0}^{J-1} a_j U_2(t/k_j)^j$$

**随机化方法（例如 qDRIFT）：** 乘积公式的变种，克服哈密顿量项数过多的问题

.....

## 含时哈密顿量模拟

$$i\frac{d|\psi(t)\rangle}{dt} = H(t)|\psi(t)\rangle, \quad |\psi(0)\rangle = |\psi_0\rangle.$$

$$|\psi(t)\rangle = U(t)|\psi_0\rangle$$

$$U(t) = \mathcal{T}e^{-i\int_0^t H(s)ds} := \sum_{k=0}^{\infty} \frac{(-i)^k}{k!} \int_0^t ds_1 \int_0^{s_1} ds_2 \cdots \int_0^{s_{k-1}} ds_k \mathcal{T}H(s_1)H(s_2)\cdots H(s_k)$$

$\mathcal{T}_{\text{exp}}$ : 序指数 (ordered exponential)

$\mathcal{T}$ : 时序算子 (time-ordering operator)

$$\mathcal{T}(AB) = \begin{cases} A(t_1)B(t_2), & t_1 > t_2 \\ B(t_2)A(t_1), & t_1 < t_2 \end{cases}$$

$H(t)$  中的时间依赖对算法设计和复杂度分析都有新的要求

## 含时哈密顿量模拟：乘积公式

考虑  $H(t) = H_0(t) + H_1(t)$ ,  $[0, T]$  分成  $L$  时间步, 时间步长  $\Delta t = T/L$

一阶乘积公式:

$$U_1(t, \Delta t) = e^{-i\Delta t H_1(t)} e^{-i\Delta t H_0(t)}$$

二阶乘积公式:

$$U_2(t, \Delta t) = e^{-i\Delta t H_0(t_m)/2} e^{-i\Delta t H_1(t_m)} e^{-i\Delta t H_0(t_m)/2}, \quad t_m = t + \frac{1}{2}\Delta t$$

Suzuki 迭代:

$$\begin{aligned} U_{2p+2}(t, \Delta t) &= U_{2p}(t + (1 - s_p)\Delta t, s_p\Delta t) U_{2p}(t + (1 - 2s_p)\Delta t, s_p\Delta t) \\ &\quad \times U_{2p}(t + 2s_p\Delta t, (1 - 4s_p)\Delta t) U_{2p}(t + s_p\Delta t, s_p\Delta t) U_{2p}(t, s_p\Delta t), \\ s_p &= (4 - 4^{1/(2p+1)})^{-1} \end{aligned}$$

一般形式:  $U_{\text{PF}}(t, \Delta t) = \prod_{j=0}^{J-1} e^{-i\beta_j\Delta t H_1(t+\nu_j\Delta t)} e^{-i\alpha_j\Delta t H_0(t+\mu_j\Delta t)}$

## 含时哈密顿量模拟：推广乘积公式

一阶推广乘积公式：

$$U_1(t, \Delta t) = e^{-i \int_t^{t+\Delta t} H_1(s) ds} e^{-i \int_t^{t+\Delta t} H_1(s) ds}$$

二阶推广乘积公式：

$$U_2(t, \Delta t) = e^{-i \int_{t+\Delta t/2}^{t+\Delta t} H_0(s) ds} e^{-i \int_t^{t+\Delta t} H_1(s) ds} e^{-i \int_t^{t+\Delta t/2} H_0(s) ds}$$

高阶公式可用 Suzuki 迭代

要求我们能高效计算  $H_0(t)$  和  $H_1(t)$  的积分，常用于控制型哈密顿量：

$$H(t) = f_0(t)H_0 + f_1(t)H_1, \quad f_0, f_1 \in \mathbb{R}$$

## Magnus 展开

$$U(t) = \mathcal{T} e^{-i \int_0^t H(s) ds} = e^{\Omega(t)}$$

$$\Omega(t) = \sum_{k=1}^{\infty} \Omega_k(t)$$

$$\Omega_1(t) = -i \int_0^t H(s) ds$$

$$\Omega_2(t) = -\frac{1}{2} \int_0^t ds_1 \int_0^t ds_2 [A(s_1), A(s_2)]$$

$$\Omega_3(t) = \frac{i}{6} \int_0^t ds_1 \int_0^t ds_2 \int_0^t ds_3 ([H(s_1), [H(s_2), H(s_3)]] + [H(s_3), [H(s_2), H(s_1)]])$$

.....

积分可以离散后用 LCU 实现

## 截断 Dyson 方法

$$\begin{aligned}\mathcal{T}e^{-i\int_0^t H(s)ds} &= \sum_{k=0}^{\infty} \frac{(-i)^k}{k!} \int_0^t ds_1 \int_0^t ds_2 \cdots \int_0^t ds_k \mathcal{T}H(s_1)H(s_2) \cdots H(s_k) \\ &\approx \sum_{k=0}^{K-1} \frac{(-i)^k}{k!} \int_0^t ds_1 \int_0^t ds_2 \cdots \int_0^t ds_k \mathcal{T}H(s_1)H(s_2) \cdots H(s_k)\end{aligned}$$

进一步积分离散后，用 LCU+OAA 实现

访问复杂度：

$$\mathcal{O}\left(\alpha T \frac{\log(\alpha T/\epsilon)}{\log \log(\alpha T/\epsilon)}\right), \quad \alpha \geq \max_t \|H(t)\|$$

# 阅读

阅读:

- ▶ LL: Chapter 5