

## I. 密度矩阵简介及其基本性质

在量子力学中，当体系的状态可以用波函数进行表示时，我们称之为纯态。纯态总是可以被表示为一系列本征态的线性叠加，它描述的是与外界环境彻底隔绝地、或者与一个时间无关的外界势场相互作用的系统。得益于量子变分算法在嘈杂中等规模量子计算机时代（noisy intermediate-scale quantum era, NISQ）的优势，量子计算的算法研究突飞猛进 [1, 2]，但目前仍主要集中于对于纯态的计算方面，例如以量子变分特征值求解器（variational quantum eigensolver, VQE）为代表的变分算法对于化学体系的基态能量求解 [3]。

但是，在现实世界中不存在完全封闭的系统，我们所感兴趣的系统都会与外界存在相互作用。举例来说，对于固体中的任一个分子（不妨称其为系统）来说，它周围的分子充当了环境，系统与环境之间存在静电与晶格的相互作用，同时温度效应带来热涨落从而影响量子态的分布，产生了统计系综（statistical ensemble），这最终影响系统的量子动力学行为。与纯态相对应的，此类与外界环境相互作用的状态被称为混合态，密度矩阵就是描述此类问题的数学工具。接下来，首先介绍密度矩阵的理论形式及相应的物理性质 [4]。

假设有系统（system, S）与（Environment, E），在二者不存在相互作用时，体系总的量子态表示为二者的直积态：

$$|\Psi_0\rangle = |\psi_S\rangle \otimes |\psi_E\rangle = \sum_n c_n |n\rangle \otimes \sum_i \alpha_i |i\rangle = \sum_{n,i} c_n \alpha_i |n\rangle |i\rangle \quad (1)$$

其中， $|n\rangle$  与  $|i\rangle$  分别对应于 S 与 E 的本征态。当 S 与 E 存在相互作用之后，总的哈密顿量表示为

$$H = H_S + H_E + H_{S-E} \quad (2)$$

S 与 E 之间的纠缠使得状态是由密度矩阵进行表示

$$\rho = \sum_k p_k |\phi_k\rangle \langle \phi_k| \quad (3)$$

而不是一个相干的叠加状态

$$|\Psi\rangle \neq \sum_k p_k |\phi_k\rangle \quad (4)$$

从而对于纯态  $|\psi\rangle$  也可以定义对应的密度矩阵  $\rho = |\psi\rangle \langle \psi|$ 。密度矩阵的重要之处在于它决

定了任一（可观测量）算符  $\hat{A}$  的期望值

$$\langle A \rangle = \sum_k \langle \phi_k | \hat{A} | \phi_k \rangle = \text{Tr}(\rho \hat{A}) \quad (5)$$

密度矩阵是具有半正定的性质，并满足  $\text{Tr} \rho = 1, \rho = \rho^\dagger$ ，对于混合态它满足  $\text{Tr} \rho^2 < 1$ 。密度矩阵的西变换可通过先对本征矢做酉变换得到：

$$\rho' = \sum_k p_k \hat{U} | \phi_k \rangle \langle \phi_k | \hat{U}^\dagger = \hat{U} \rho \hat{U}^\dagger \quad (6)$$

进一步的，我们给出密度矩阵的矩阵表示。将  $| \phi_k \rangle$  写为完备正交基的线性叠加，

$$| \phi_k \rangle = \sum_m c_m^{(k)} | m \rangle \quad (7)$$

代入式 (3)，可得

$$\rho = \sum_k p_k \sum_{m,n} c_m^{(k)} | m \rangle c_n^{(k)*} \langle n | = \sum_{m,n} \left( \sum_k p_k c_m^{(k)} c_n^{(k)*} \right) | m \rangle \langle n | = \sum_{mn} \rho_{mn} | m \rangle \langle n | \quad (8)$$

从这里可以容易证明前面所提到的  $\rho$  的厄米性质，即  $\rho_{mn} = \rho_{nm}^*$ ，以及  $\text{Tr} \rho = 1$ （利用  $\sum_k p_k = 1$  以及  $\sum_m c_m^{(k)*} c_m^{(k)}$ ）。 $\rho$  的非对角元代表的是非常重要的 **相干** 性质 (coherence)，对角元则可以理解为混合态在单个基矢上的占据概率 (population)，量子动力学的研究很大程度集中于观察 coherence 与 population 的演化并进行理解与解释 [5, 6]。

即使系统 S 与环境 E 存在相互作用 (entangled)，在只对于系统 S 部分进行观测的情况下，可以得到的全部信息都被包含在约化密度矩阵中。（但这并不代表约化密度矩阵影响整个体系的动力学性质，因为不同的环境 E 也可能会带来同样的 S 的约化密度矩阵。）约化密度矩阵是通过将  $\rho$  中的环境部分求迹 (partial trace) 得到的矩阵，

$$\rho_S = \text{Tr}_E(\rho). \quad (9)$$

以一个贝尔态对应的密度矩阵举例说明，

$$| \psi \rangle = \frac{1}{\sqrt{2}} (| 0_S 0_E \rangle + | 1_S 1_E \rangle) \quad (10)$$

$$\rho = | \psi \rangle \langle \psi | = \frac{1}{2} (| 0_S 0_E \rangle \langle 0_S 0_E | + | 0_S 0_E \rangle \langle 1_S 1_E | + | 1_S 1_E \rangle \langle 1_S 1_E | + | 1_S 1_E \rangle \langle 0_S 0_E |) \quad (11)$$

$$\rho_S = \text{Tr}_E(\rho) = \frac{1}{2} (| 0_S \rangle \langle 0_S | + | 1_S \rangle \langle 1_S |) = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (12)$$

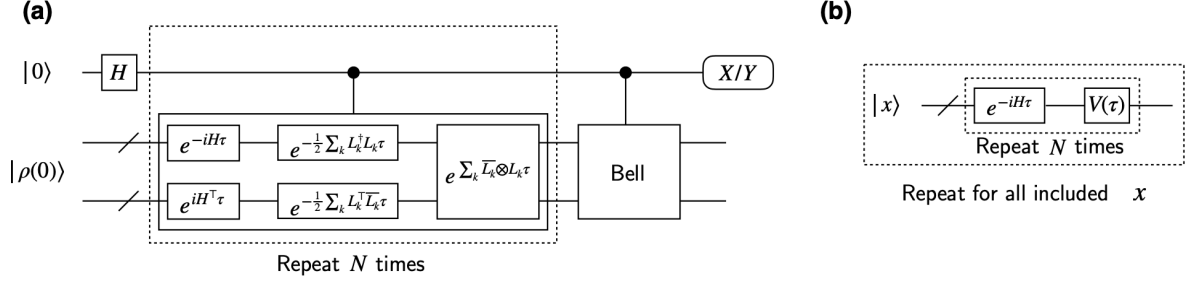


FIG. 1.  $N$  步 Trotter 分解对于密度矩阵进行时间演化的两种方案的量子电路图示：(a) 向量化的密度矩阵  $|\rho\rangle$  的时间演化，(b) 基于纯化的方案 [7]

## II. 密度矩阵的演化：LINDBLAD 主方程

密度矩阵的含时演化服从量子刘维尔（Liouville）方程，也叫冯诺依曼（von Neumann）方程：

$$\frac{d\rho_{\text{tot}}}{dt} = -\frac{i}{\hbar}[H_{\text{tot}}, \rho_{\text{tot}}] \quad (13)$$

其中  $H_{\text{tot}}$  为包含系统、环境与相互作用的总的哈密顿量（见式 (2)）。在我们只关心系统  $S$  部分的动力学演化行为时，将环境部分求迹，从而得到系统的密度矩阵所满足的 Lindblad 主方程，它描述了系统在演化过程中受到环境的作用产生的耗散（dissipation）动力学行为，

$$\frac{d\rho}{dt} = -\frac{i}{\hbar}[H, \rho] + \sum_k [L_k \rho L_k^\dagger - \frac{1}{2}\{L_j^\dagger L_j \rho\}] \quad (14)$$

其中  $\{, \}$  为反对易子， $H$  为感兴趣的系统部分的哈密顿量， $L_j$  被称为 Lindblad 算子，包含了系统与环境之间的耦合，其推导有赖于一些近似，包括玻恩（Born）近似、马尔科夫（Markov）近似与久期（secular）近似 [4]，这些近似是基于系统与环境之间存在弱的耦合并忽略环境带来的记忆效应。Lindblad 主方程演化的初始状态一般表示为由系统与环境部分构成的直积态，即式 (1)，表示系统尚不受到环境的影响。

实现 Lindblad 主方程的动力学演化具有多种策略 [7–10]，如虚时演化方案处理非酉算符的演化（见 Fig. 1(a)），其将维度为  $2^n \times 2^n$  的密度矩阵进行向量化长度为  $4^n$  的  $|\rho\rangle$  ( $4^n$ )，其满足如下的演化方程：

$$\frac{d|\rho(t)\rangle}{dt} = \left[ -i\mathbb{I} \otimes H + iH^\top \otimes \mathbb{I} + \sum_k (\bar{L}_k \otimes L_k - \frac{1}{2}\mathbb{I} \otimes (L_k^\dagger L_k) - \frac{1}{2}(L_k^\top \bar{L}_k) \otimes \mathbb{I}) \right] |\rho\rangle \quad (15)$$

$$|\rho(t)\rangle = [\exp(-iH_1\tau)\exp(-H_2\tau)]^N |\rho(0)\rangle + \mathcal{O}(\tau^2 N) \quad (16)$$

式 (16) 为时间演化的 Trotter 分解, 演化步长  $\tau = \frac{t}{N}$ ,  $H_1$  是厄米的, 因此这一部分的演化是可以利用标准的量子计算方案加以实现,  $iH_2$  是非厄米的,  $\exp(-H_2\tau)$  为非酉的, 可以通过虚时演化的方案实现。这一方案使用的量子比特数与线路深度对于 near term 的量子计算机并不现实, 还有采用变分的纯化演化方案 Fig. 1(b),

$$\rho = \sum_{x \in I} p_x U|x\rangle \langle x| U^\dagger \rightarrow |\rho\rangle = \sum_{x \in I} p_x U|x\rangle \otimes \bar{U}|x\rangle \quad (17)$$

可以对其调研加以复现。此外, 密度矩阵的演化可以映射为 McLachlan 定律 (McLachlan's rule) 进行变分处理, 考虑  $d\rho/dt = \mathcal{L}[\rho]$  的演化方程, 从而变为最小化  $\min_{\theta} \|(\frac{d}{dt} - \mathcal{L})\rho(\theta)\|$  的问题。

### III. 研究计划与研究目标

- (6.16-6.30): 文献调研、学习 mindquantum 软件
- (7.1-8.1) 完成基于密度矩阵的各种量子门的作用, 接口保证与 mindquantum 现有模拟器接口一致。
- (8.1-9.1) 完成基于密度矩阵的变分量子算法, 确定密度矩阵的演化策略。
- (9.1-9.30) 完成基于密度矩阵的动力学演化、主方程求解。

### IV. 项目成果: 基于 MINDQUANTUM 实现密度矩阵变分量子算法完成主方程演化

纯态可以被表示为一系列本征态的线性叠加, 它描述的是与外界环境彻底隔绝地、或者与一个时间无关的外界势场相互作用的系统, 因此纯态的波函数演化遵循含时薛定谔方程; 而混合态描述了系统与外界环境 (噪声、热浴等) 耦合的情况, 混合态无法用一个波函数表示, 其数学表示为密度矩阵。密度矩阵的时间演化遵循量子刘维尔方程, 其动力学过程也被称为开放系统的动力学。与含时薛定谔方程不同的是, 密度矩阵的时间演化是非酉的, 然而量子计算机中的门操作都是酉矩阵, 因此需要设计巧妙的量子电路对其进行求解。结合本篇教程, 可以自然地推广至虚时演化的方案, 虚时演化被用来获取吉布斯态, 它描述了体系的微观状态的玻尔兹曼热布居。

主要成果内容包括如下几个部分:

#### 1. 含时变分原理介绍

2. 利用纯化的方法得到密度矩阵，以及求解密度矩阵之间的重叠积分
3. 利用 mindquantum 实现密度矩阵的（非酉）时间演化，求解主方程
4. Ising 模型的 Lindblad 主方程求解实例

### A. 含时变分原理

密度矩阵  $\rho$  的开放系统演化方程的一般形式为：

$$\frac{d\rho}{dt} = \mathcal{L}(\rho), \quad (18)$$

$\mathcal{L}$  被称作超算符（super-operator），它的一般表达式为  $\mathcal{L} = \sum_i g_i S_i \rho T_i^\dagger$ ，其中  $g_i$  为系数， $S_i$  与  $T_i^\dagger$  为酉算符。我们将  $\rho$  参数化为  $\rho(\vec{\theta}(t))$ ，其中  $\vec{\theta} = [\theta_1, \theta_2, \dots]$ 。根据 McLachlan 含时变分原理，我们需要

$$\delta \left\| \sum_i \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \dot{\theta}_i - \mathcal{L}(\rho) \right\| = 0, \dot{\theta}_i = \frac{d\theta_i}{dt} \quad (19)$$

这本质上是将实际上  $\rho$  对时间的导数投影到切空间中，切空间为我们所选定的拟设可以表示的空间，即  $\left\{ \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right\}$  所张成的空间。基于此，我们得到

$$\sum_j M_{i,j} \dot{\theta}_j = V_i \quad (20)$$

$$M_{i,j} = \text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_j} \right] \quad (21)$$

$$V_i = \text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger \mathcal{L}(\rho) \right] \quad (22)$$

我们将设计量子线路求解  $M_{i,j}$  与  $V_i$ ，然后利用经典算法求线性方程得到各个  $\theta$  对时间的导数。 $M_{i,j}$  与  $V_i$  可以写为一个一般的表达式，即  $\text{Tr}[\rho_1 \rho_2]$ 。

### B. 通过纯化得到混合态的密度矩阵表示

我们无法将一个系统 S 对应的混合态用一个波函数进行表示，但可以通过引入一个辅助空间 A 对其进行纯化，从而可以用一个波函数  $|\psi_{AS}\rangle$  描述这个扩大的空间。S 部分对

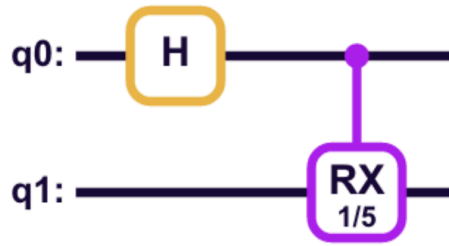


FIG. 2. 纯化：辅助空间比特 + 物理空间比特构成扩大空间的纯态波函数

应的密度矩阵即

$$\rho_S = \text{Tr}_A |\psi_{AS}\rangle \langle \psi_{AS}| \quad (23)$$

这里给一个简单的例子进行说明。 $q_0$  是引入的辅助比特,  $q_1$  对应于我们真正感兴趣的物理系统的比特。通过将扩大空间的密度矩阵对辅助空间求 ‘Trace’, 我们可以得到物理系统对应的约化密度矩阵  $\rho = 0.995|0\rangle\langle 0| + 0.0497i|0\rangle\langle 1| - 0.0497i|1\rangle\langle 0| + 0.00498|1\rangle\langle 1|$ , 显然, 量子比特  $q_1$  的状态并不是一个纯态 (若为纯态, 则  $\rho$  的非对角元应为 0, 对角元代表混合态在不同基矢的占据, 非对角元代表基矢间的相干)。

```
1 from mindquantum import Circuit, RX, H, X
2 import numpy as np
3 circ = Circuit()
4 circ += H.on(0)
5 circ += RX(0.2).on(1, 0)
6 circ.svg() # 见 FIG.2.
```

```
1 psi = circ.get_qs()
2 # mindquantum 的开发后续可依下面的逻辑, 对部分 qubit 求 partial trace 来得到混合态
   的密度矩阵
3 psi = psi.reshape(2, 2)
4 rho = np.tensordot(psi, psi.conj(), axes=(1, 1))
5 print(rho)
6 [[0.99501664+0.j          0.          +0.04966733j]
7   [0.          -0.04966733j  0.00498336+0.j          ]]
```

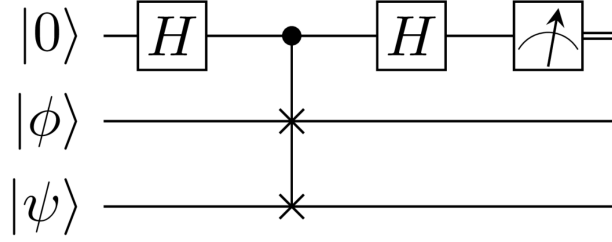


FIG. 3. 利用 Hadamard 测试求纯态波函数的重叠积分

### C. 设计量子线路计算两个密度矩阵的重叠积分： $\text{Tr}[\rho_1 \rho_2]$

对于纯态的波函数的重叠积分的求解是简单的，密度矩阵的重叠积分则会更为复杂，但二者的逻辑有相通之处。前者是采用了变种的 Hadamard test，后者为变种的 Swap test。为了便于理解，首先可以对纯态波函数的重叠积分的求解做一下介绍，即如何求  $\langle \psi_1 | \psi_2 \rangle$ 。

FIG.3. 的量子线路可被用来测量  $\langle \psi | \phi \rangle$ ，其最终的量子态为

$$\frac{1}{\sqrt{2}}(|\phi\rangle|\psi\rangle + |\psi\rangle|\phi\rangle)|0\rangle + \frac{1}{\sqrt{2}}(|\phi\rangle|\psi\rangle - |\psi\rangle|\phi\rangle)|1\rangle \quad (24)$$

于是可以得到最后对辅助比特做测量的结果为

$$P_0 = \frac{1}{2}(1 + |\langle \psi | \phi \rangle|^2), P_1 = \frac{1}{2}(1 - |\langle \psi | \phi \rangle|^2) \quad (25)$$

在了解了纯化的原理以后，我们现在尝试计算密度矩阵的重叠积分  $\text{Tr}[\rho_1 \rho_2]$ 。不妨假设最为一般的表达式

$$\rho_1 = \text{Tr}_A [U_1 |\psi_{AS}\rangle \langle \psi_{AS}| U_2^\dagger] \quad (26)$$

$$\rho_2 = \text{Tr}_A [U_3 |\psi_{AS}\rangle \langle \psi_{AS}| U_4^\dagger] \quad (27)$$

$$\text{Tr}[\rho_1 \rho_2] = \text{Tr}_S [\rho_1 \rho_2] \quad (28)$$

这里我们依然采用  $|\psi_{AS}\rangle$  来表示扩大空间（辅助空间 A+ 物理空间 S）的波函数。我们用 a 标记  $\rho_1$  对应的量子线路中所有的物理比特，用 b 标记  $\rho_1$  对应的量子线路中所有的辅助比特；用 c 标记  $\rho_2$  对应的量子线路中所有的物理比特，用 d 标记  $\rho_1$  对应的量子线路中所有的辅助比特。除此之外，我们再额外引入一个辅助比特  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  作测量。接下来，我们需要首先构造出

$$\frac{1}{\sqrt{2}}|0\rangle U_1 |\psi_{AS}\rangle U_3 |\psi_{AS}\rangle + \frac{1}{\sqrt{2}}|1\rangle U_2 |\psi_{AS}\rangle U_4 |\psi_{AS}\rangle. \quad (29)$$

该式可以被展开为：

$$\sum_{abcd} \frac{1}{\sqrt{2}} |0\rangle C_1^{ab} C_3^{cd} |abcd\rangle + \frac{1}{\sqrt{2}} |1\rangle C_2^{ab} C_4^{cd} |abcd\rangle \quad (30)$$

接下来将  $\rho_1$  与  $\rho_2$  部分对应的物理比特进行 control-swap:

$$\sum_{abcd} \frac{1}{\sqrt{2}} |0\rangle C_1^{ab} C_3^{cd} |abcd\rangle + \frac{1}{\sqrt{2}} |1\rangle C_2^{cb} C_4^{ad} |abcd\rangle \quad (31)$$

最后我们在对测量比特做一个 Hadamard 门，

$$\sum_{abcd} \frac{1}{2} |0\rangle (C_1^{ab} C_3^{cd} + C_2^{cb} C_4^{ad}) |abcd\rangle + \frac{1}{2} |1\rangle (C_1^{ab} C_3^{cd} - C_2^{cb} C_4^{ad}) |abcd\rangle \quad (32)$$

$$P_0 - P_1 = \sum_{abcd} \text{Re}[C_1^{ab} C_2^{cb*} C_3^{cd} C_4^{ad*}] \quad (33)$$

显然测量  $P_0 - P_1$  就等于我们要求的  $\text{Tr}[\rho_1 \rho_2]$  的实部：

$$\text{ReTr}_S[(\text{Tr}_A U_1 |\psi_{AS}\rangle \langle \psi_{AS}| U_2^\dagger)(\text{Tr}_A U_3 |\psi_{AS}\rangle \langle \psi_{AS}| U_4^\dagger)] \quad (34)$$

若要求解虚部，则在测量比特第一个 Hadamard 门之后施加一个 S dagger 门即可，具体可参考源程序。

#### D. 开放系统动力学实例：Ising 模型的主方程求解

这里利用 mindquantum 完美复现了 Yuan X, Endo S, Zhao Q, et al. Theory of variational quantum simulation[J]. Quantum, 2019, 3: 191. 的结果, <https://doi.org/10.22331/q-2019-10-07-191>, 对于一个 2 个自旋格点的 Ising 模型，哈密顿量为

$$H = X_0 + X_1 + \frac{1}{4} Z_0 Z_1 \quad (35)$$

其主方程为：

$$\frac{d}{dt} \rho = -i[H, \rho] + \sum_{j=0}^1 \mathcal{L}_j(\rho) \quad (36)$$

$$\mathcal{L}_j(\rho) = \frac{1}{2} \{2\sigma_j^- \rho \sigma_j^+ - \sigma_j^+ \sigma_j^- \rho - \rho \sigma_j^+ \sigma_j^-\} \quad (37)$$

$t = 0$  时将所有  $\theta$  设置为为，初始状态对应于一个纯态  $|00\rangle$  的密度矩阵

$$\rho_0 = |00\rangle \langle 00| \quad (38)$$

接下来，我们将展示如何演化这一主方程。

首先，设计  $\rho$  的量子线路 Ansatz 如下，其中  $q_2$  与  $q_3$  为要研究的自旋格点对应的物理比特， $q_0$  与  $q_1$  为引入的辅助比特，该 Ansatz 共由 6 个参数  $(\theta_0, \theta_1, \dots, \theta_5)$  控制。



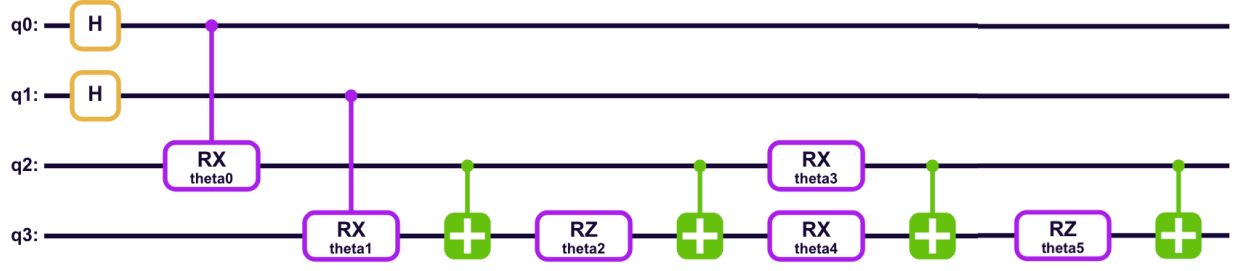


FIG. 4. Ising 模型的密度矩阵参数化的 Ansatz

```

1 from mindquantum import Circuit, RX, RZ, H, X, Y, Z, S
2 from mindquantum.core.gates import SWAP
3 from mindquantum.core.gates import Measure
4 from mindquantum.simulator import Simulator      # 从mindquantum.simulator中导
           入Simulator类
5 import numpy as np
6 from scipy import linalg
7
8 circ = Circuit()
9
10 operations = [H.on(0), H.on(1), RX("theta0").on(2, 0), RX("theta1").on(3, 1),
11              X.on(3, 2), RZ("theta2").on(3), X.on(3, 2), RX("theta3").on(2),
12              RX("theta4").on(3), X.on(3, 2), RZ("theta5").on(3), X.on(3, 2),
13              ]
14
15 for i_op in operations:
16     circ += i_op
17 circ.svg() # 见FIG.4.

```

接下来主要是通过量子线路求解  $M_{i,j}(0 \leq i, j \leq 5)$  以及  $V_i(0 \leq i \leq 5)$

$$\sum_j M_{i,j} \theta_j = V_i \quad (39)$$

$$M_{i,j} = \text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_j} \right] \quad (40)$$

$$V_i = \text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger \mathcal{L}(\rho) \right] \quad (41)$$

令

$$\frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_j} = -iA + iA^\dagger, \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} = -iB + iB^\dagger \quad (42)$$

则可导出（此处省略推导）：

$$\text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_j} \right] = 2\text{ReTr}[B^\dagger A] - 2\text{ReTr}[BA] \quad (43)$$

下面为求解  $M_{i,j}$  的量子线路，其中 thetas 为描述线路的参数向量  $([\theta_0, \theta_1, \dots, \theta_5])$ ，theta\_i 与 theta\_j 的取值范围为 0 5，qcidx 取 0 或 1，分别对应于上式中的两项： $\text{ReTr}[B^\dagger A], \text{ReTr}[BA]$ 。

```

1 def make_circ_m(thetas, theta_i, theta_j, qcidx):
2     assert qcidx in [0, 1]
3     circ = Circuit()
4     circ += H.on(0)
5     circ += H.on(1)
6     circ += H.on(2)
7     circ += H.on(5)
8     circ += H.on(6)
9
10    apply_dic = {0: [RX(thetas[0]).on(3, 1)],
11                    1: [RX(thetas[1]).on(4, 2)],
12                    2: [X.on(4, 3), RZ(thetas[2]).on(4), X.on(4, 3)],
13                    3: [RX(thetas[3]).on(3)],
14                    4: [RX(thetas[4]).on(4)],
15                    5: [X.on(4, 3), RZ(thetas[5]).on(4), X.on(4, 3)]
16    }
17
18
19    for i in range(6):

```

```

20     if i == theta_i:
21         if qcidx == 1:
22             circ += X.on(0)
23         if i in [0, 1]:
24             circ += X.on(i+3, (0, i+1))
25         elif i in [2, 5]:
26             circ += Z.on(3, 0)
27             circ += Z.on(4, 0)
28         else:
29             circ += X.on(i, 0)
30         if qcidx == 1:
31             circ += X.on(0)
32
33     for i_op in apply_dic[i]:
34         circ += i_op
35
36     apply_dic = {0: [RX(thetas[0]).on(7, 5)],
37                    1: [RX(thetas[1]).on(8, 6)],
38                    2: [X.on(8, 7), RZ(thetas[2]).on(8), X.on(8, 7)],
39                    3: [RX(thetas[3]).on(7)],
40                    4: [RX(thetas[4]).on(8)],
41                    5: [X.on(8, 7), RZ(thetas[5]).on(8), X.on(8, 7)]
42                }
43
44     for i in range(6):
45         if i == theta_j:
46             circ += X.on(0)
47         if i in [0, 1]:
48             circ += X.on(i+7, (0, i+5))
49         elif i in [2, 5]:

```

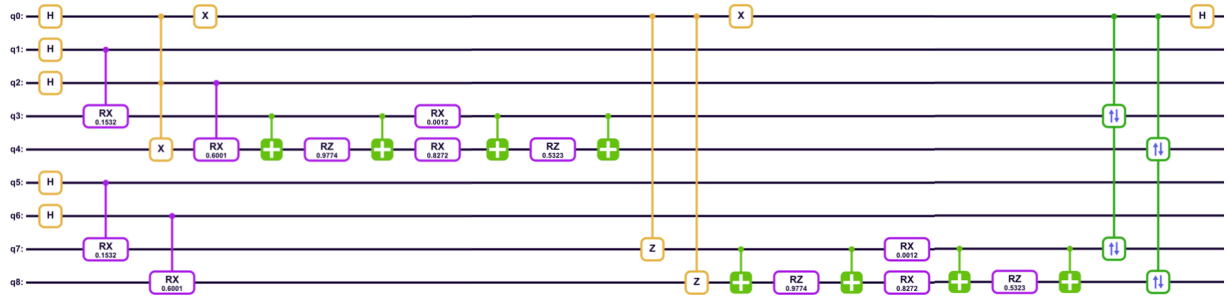


FIG. 5. 求解  $M_{1,2}$  的第一项的量子线路

```

50         circ += Z.on(7, 0)
51         circ += Z.on(8, 0)
52     else:
53         circ += X.on(i+4, 0)
54         circ += X.on(0)
55         for i_op in apply_dic[i]:
56             circ += i_op
57         circ += SWAP.on((3, 7), 0)
58         circ += SWAP.on((4, 8), 0)
59         circ += H.on(0)
60     return circ

```

作为示例，可以看一下求解  $M_{1,2}$  的第一项的量子线路：

```

1 circ = make_circ_m(thetas, 1, 2, 0)
2 circ.svg() # 见FIG.5.

```

**mindquantum** 内置了量子线路的测量方案，依然以上图中所示的线路为例：

```

1 circ = make_circ_m(thetas, 1, 2, 0)
2 circ += Measure('q0').on(0)
3 sim = Simulator('projectq', 9)
4 sim.reset()
5 result = sim.sampling(circ, shots=1000)
6 res = result.data

```

```

7 print(res)
8 {'0': 535, '1': 465}

```

我们也可以不用采样的方法，直接利用张量乘法拿到精确的“测量值”：

```

1 circ = make_circ_m(thetas, 1, 2, 0)
2 vec = circ.get_qs()
3 vec = vec.reshape(-1, 2)
4 res = np.einsum("ab, ab -> b", np.conj(vec), vec).real
5 print(res)
6 [0.53349583 0.46650417]

```

这么做的好处是相当节省时间，第一种做采样的方法如果采 100000 次可能需要 10 分钟，而第二种方法可以瞬间完成。

$V_i$  的求解会更为繁琐一些，因为涉及到  $\mathcal{L}(\rho)$  的项数较多：

$$\mathcal{L}(\rho) = -iH\rho + i\rho H + \frac{1}{2}(2\sigma_0^- \rho \sigma_0^+ - \sigma_0^+ \sigma_0^- \rho - \rho \sigma_0^+ \sigma_0^-) + \frac{1}{2}(2\sigma_1^- \rho \sigma_1^+ - \sigma_1^+ \sigma_1^- \rho - \rho \sigma_1^+ \sigma_1^-) \quad (44)$$

进一步展开：

$$\begin{aligned} \mathcal{L}(\rho) = & (-iX_0 - iX_1 - \frac{i}{4}Z_0Z_1)\rho + \rho(iX_0 + iX_1 + \frac{i}{4}Z_0Z_1) \\ & + (-\frac{i}{4}X_0Y_0 - \frac{i}{4}X_1Y_1)\rho + \rho(-\frac{i}{4}X_0Y_0 - \frac{i}{4}X_1Y_1) \\ & + \frac{1}{4}(X_0\rho X_0 + X_1\rho X_1 + Y_0\rho Y_0 + Y_1\rho Y_1) \\ & + \frac{i}{4}(X_0\rho Y_0 - Y_0\rho X_0 + X_1\rho Y_1 - Y_1\rho X_1) - \rho \end{aligned}$$

在这里只举其中几项的求解为例进行说明，其余的依照相同的逻辑可以得到。

$$\text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger (-iX_0\rho) \right] + \text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger (i\rho X_0) \right] = -2\text{ReTr}[BX_0\rho] + 2\text{ReTr}[B\rho X_0] \quad (45)$$

$$\text{Tr} \left[ \left( \frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} \right)^\dagger (X_0\rho X_0) \right] = 2\text{ImTr}BX_0\rho X_0 \quad (46)$$

其中，

$$\frac{\partial \rho(\vec{\theta}(t))}{\partial \theta_i} = -iB + iB^\dagger \quad (47)$$

这里不再展示求其中单独某一项的量子电路，其思路与前面求解  $M$  矩阵是一致的，对于虚部的求解需要做一些微调。你可以在 ‘mindquantum/tutorials/source’ 中找到所有的代码。

最后，我们可以模拟实时间的动力学行为，在这里直接调用包含所有线路测量的程序 `dynamics.py`。

```
1 from dynamics import *

1 thetas = [0] * 6
2 z0 = [measure_z0(thetas)]
3 for i in range(1000):
4     grads = get_grad(thetas) #给定thetas, 得到thetas关于时间的导数
5     grads = [igrad * 2 for igrad in grads]
6     thetas = [np.real(thetas[j]+grads[j]*0.01) for j in range(len(grads))]
7     z0.append(measure_z0(thetas))
8 fig, axes = plt.subplots()
9 plt.plot(np.arange(0, 10.01, 0.01), z0, label='vqa-dynamics')
10 axes.legend()
11 axes.set_xlabel("time")
12 axes.set_ylabel(r"$\langle Z_0 \rangle$")
```

为了便于可读，本项目文档只展示了主要过程与部分实例，本项目的源代码参见 mindquantum 的 gitee 仓库中的 Pull Request 链接：<https://gitee.com/mindspore/mindquantum/pulls/1251>

## V. 遇到的问题及解决方案

上文中介绍的量子变分方法的关键在于设计量子线路去求解  $\text{Tr}[\rho_1 \rho_2]$ ，最后我们需要调用 `mindquantum` 内置的 `Simulator` 做测量来拿到  $q_0$  量子比特在  $|0\rangle$  与  $|1\rangle$  取到的概率，而采样的数目 `shots` 动辄需要  $10^5 \sim 10^6$  次，这使得对于每个电路的测量时间非常长：当设置 `shots=100000` 时，每求一次所有  $\theta$  对  $t$  的导数需要约 10 分钟，而对于 FIG.6 我们需要演化 1000 步，因此这使得计算时长非常久 ( $\sim 10^4$  min)。

于是为 `mindquantum` 增加了新的功能，即不经过采样，通过直接做张量乘积得到精确的测量结果，借鉴了 IBM 的 Qiskit 的功能，可以在秒量级输出测量结果。

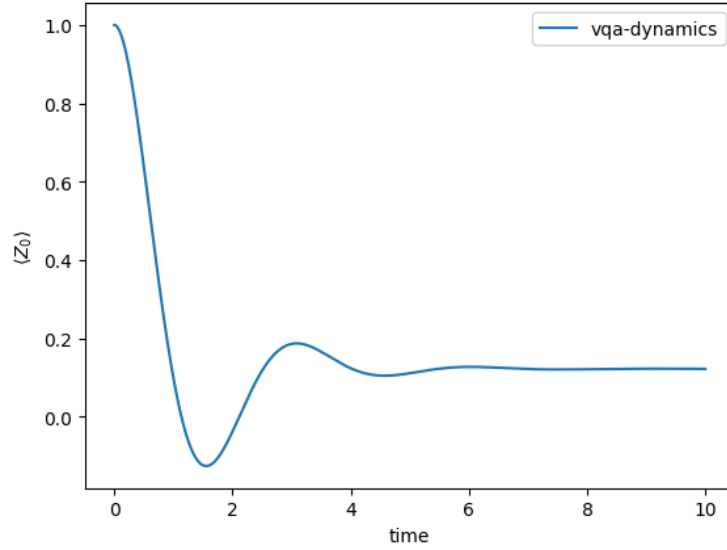


FIG. 6. 开放系统动力学模拟：Ising 自旋模型的主方程求解， $\langle Z_0 \rangle$  在  $t = 0$  到  $t = 10$  的实时演化过程,  $dt = 0.01$

```

1 def get_prob(circ: Circuit, qubit):
2     r"""
3     Args:
4         circuit (Circuit): The circuit that you want to evolution and do
5         sampling.
6         qubit (number): the index of qubit to be measured.
7
8     Returns:
9         Exact probability without making samplings.
10
11     Examples:
12     >>> circ = Circuit().ry(1.1, 0).ry(2.2, 1)
13     >>> res = get_prob(circ)
14     >>> print("exact probability is:", res)
15     >>> circ += Measure('q0_0').on(0)
16     >>> sim = Simulator('projectq', circ.n_qubits)
17     >>> res = sim.sampling(circ, shots=1000, seed=42)

```

```

17     >>> print(res)
18     "exact probability is:" {'0': 0.7267980607127886, '1':
0.27320193928721137}
19     {'0': 707, '1': 293}
20     """
21     state_vector = circ.get_qs()
22     if qubit == 0:
23         vec = state_vector.reshape(-1, 2)
24         res = np.einsum("ab, ab -> b", np.conj(vec), vec).real
25         return {"0": res[0], "1": res[1]}
26     else:
27         tensor_shape = [2**(circ.n_qubits-qubit-1), 2, 2**qubit]
28         vec = state_vector.reshape(tensor_shape)
29         res = np.einsum("abc, abc->b", np.conj(vec), vec).real
30         return {"0": res[0], "1": res[1]}

```

## VI. 后续计划

1. 更为复杂的系统的量子动力学模拟是自然的，但效果如何需要考虑两点：如何设计高效的 ansatz；如何进一步加速计算速度。
2. 后续是计划利用将这种量子变分方案应用到自己的科研课题中，通过与经典算法中的矩阵乘积态相结合，发展量子-经典杂化的计算方法，模拟真实的开放系统动力学过程。

- 
- [1] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.
- [2] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C Benjamin, and Xiao Yuan. Quantum computational chemistry. *Reviews of Modern Physics*, 92(1):015003, 2020.



- [3] Peter JJ O’ Malley, Ryan Babbush, Ian D Kivlichan, Jonathan Romero, Jarrod R McClean, Rami Barends, Julian Kelly, Pedram Roushan, Andrew Tranter, Nan Ding, et al. Scalable quantum simulation of molecular energies. *Physical Review X*, 6(3):031007, 2016.
- [4] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.
- [5] Xin-Yu Chen, Na-Na Zhang, Wan-Ting He, Xiang-Yu Kong, Ming-Jie Tao, Fu-Guo Deng, Qing Ai, and Gui-Lu Long. Global correlation and local information flows in controllable non-markovian open quantum dynamics. *npj Quantum Information*, 8(1):1–6, 2022.
- [6] Gregory D Scholes, Graham R Fleming, Lin X Chen, Alán Aspuru-Guzik, Andreas Buchleitner, David F Coker, Gregory S Engel, Rienk Van Grondelle, Akihito Ishizaki, David M Jonas, et al. Using coherence to enhance function in chemical and biophysical systems. *Nature*, 543(7647):647–656, 2017.
- [7] Hirsh Kamakari, Shi-Ning Sun, Mario Motta, and Austin J Minnich. Digital quantum simulation of open quantum systems using quantum imaginary–time evolution. *PRX Quantum*, 3(1):010320, 2022.
- [8] Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating lindblad evolution. *arXiv preprint arXiv:1612.09512*, 2016.
- [9] Brian William Rost. *Simulating Open Quantum Systems on Near Term Quantum Computers*. PhD thesis, Georgetown University, 2021.
- [10] Anthony W Schlimgen, Kade Head-Marsden, LeeAnn M Sager, Prineha Narang, and David A Mazziotti. Quantum simulation of open quantum systems using a unitary decomposition of operators. *Physical Review Letters*, 127(27):270503, 2021.