

时间演化矩阵乘积算符方法及其在量子开放系统中的应用

陈若凡

Time-evolving matrix product operator method and its applications in open quantum system

Chen Ruo-Fan

引用信息 Citation: *Acta Physica Sinica*, 72, 120201 (2023) DOI: 10.7498/aps.72.20222267

在线阅读 View online: <https://doi.org/10.7498/aps.72.20222267>

当期内容 View table of contents: <http://wulixb.iphy.ac.cn>

您可能感兴趣的其他文章

Articles you may be interested in

耗散响应理论及其在开放系统中的应用

Dissipative linear response theory and its applications in open quantum systems

物理学报. 2021, 70(23): 230306 <https://doi.org/10.7498/aps.70.20211687>

一种与开放系统初态无关的非马尔科夫度量

Non-Markovian measure independent of initial states of open systems

物理学报. 2022, 71(21): 210303 <https://doi.org/10.7498/aps.71.20221053>

量子扩散通道中Wigner算符的演化规律

Time evolution law of Wigner operator in diffusion channel

物理学报. 2020, 69(9): 090301 <https://doi.org/10.7498/aps.69.20200106>

不同外加磁场中Kaplan-Shekhtman-Entin-Wohlman-Aharony相互作用对量子失协非马尔科夫演化的影响

Effects of Kaplan-Shekhtman-Entin-Wohlman-Aharony interaction on quantum discord of non-markovian dynamics under different magnetic fields

物理学报. 2021, 70(22): 223401 <https://doi.org/10.7498/aps.70.20211277>

半开放系统中的粒子逃逸问题

Particle escape problem in semi-open system

物理学报. 2022, 71(16): 167302 <https://doi.org/10.7498/aps.71.20220450>

矩阵形式的不变本征算符方法以及几种介观电路的本征频率

The invariant eigen-operator method in matrix form and the eigenfrequency of several mesoscopic circuits

物理学报. 2019, 68(22): 220301 <https://doi.org/10.7498/aps.68.20190651>

综述

时间演化矩阵乘积算符方法及其在量子开放系统中的应用*

陈若凡†

(四川师范大学物理与电子工程学院, 计算科学中心, 成都 610068)

(2022 年 11 月 27 日收到; 2023 年 4 月 4 日收到修改稿)

量子开放系统在量子科学的发展中占据着举足轻重的地位, 因此对其数值计算方法进行研究具有重要的意义. 对于量子开放系统, 在 20 世纪 90 年代发明的准绝热传播子路径积分方法是为数不多的精确的数值计算方法, 但其计算复杂度随着系统的大小和关联时间长度呈指数增长, 因此在实际计算中它所能计算的物理模型比较受限. 近年来, 张量网络的研究和应用有了长足的进展. 使用张量网络来表达该方法可以使其计算复杂度变成多项式增长, 极大地提高了计算效率. 由此发展出的新方法则被称为时间演化矩阵乘积算符方法, 是一种高效的、数值精确的、并且非马尔可夫的计算方法, 在量子开放系统的研究中有广泛的应用前景. 本文首先综述了准绝热传播子路径积分方法, 接着介绍了矩阵乘积态的基本思路, 然后利用矩阵乘积态来表述准绝热传播子路径积分方法, 从而对时间演化矩阵乘积算符方法进行介绍; 最后综述了该方法在量子开放系统中的应用, 并以自旋-玻色子系统中的关联函数和热流计算为例对该方法进行了展示.

关键词: 量子开放系统, 准绝热传播子路径积分, 时间演化矩阵乘积算符, 张量网络, 非马尔可夫演化

PACS: 02.70.-c, 03.65.Yz, 05.10.-a, 68.65.Hb

DOI: 10.7498/aps.72.20222267

1 引言

孤立的量子系统是一种理想的模型, 它忽略了环境对系统的影响. 在实际情况下, 不管多么微弱, 环境总是不停对系统产生扰动, 使系统产生退相干和弛豫现象. 因此研究环境对量子系统的影响有着重要的物理意义, 并由此产生一个被广泛关注和研究的领域, 一般称之为量子开放系统或量子耗散系统. 关于量子开放系统的研究具有很长的历史, 相应的综述可参考文献 [1, 2].

1981 年, Caldeira 和 Leggett^[3] 指出在量子耗散过程中, 环境可采用一系列的玻色子来描述, 并由此提出里程碑式的 Caldeira-Leggett 模型^[4,5], 深入解释了量子耗散过程. 尽管 Caldeira-Leggett 模

型已经是一个较为简单的模型, 但对其精确求解仍然是一件困难的事情, 即使其中系统部分是最简单的两能级系统. 在量子力学中, 两能级系统是最简单的非平凡物理系统. 因为两能级系统可以用一个二分之一自旋来等效描述, 这样的模型被广泛称为自旋-玻色子模型 (spin-boson model). 尽管看起来简单, 自旋-玻色子模型的物理性质丰富, 并常常作为研究量子开放系统的出发点, 因此在该领域具有着重要的地位^[1,6]. 对路径积分采用 noninteracting-blip 近似, Leggett 等^[6] 对自旋-玻色子系统进行了深入研究并得到系列的结果. 直到现在, 该近似仍然是计算自旋-玻色子模型的一个重要方法, 但其在某些参数区域得到的结果并不准确.

1994 年, Makarov 和 Makri^[7,8] 研究发现在 Caldeira-Leggett 模型的路径积分表达式中, 时间

* 国家自然科学基金 (批准号: 12104328) 资助的课题.

† 通信作者. E-mail: physcrf@sicnu.edu.cn

的关联长度是有限的,甚至在很低的温度时依然如此.于是他们在时间关联上进行截断,并发展出一种迭代的张量乘法算法来对路径积分进行数值求解,该方法一般被称为准绝热传播子路径积分(quasi-adiabatic propagator path integral, QUAPI)方法.该方法完整包含了环境的非马尔可夫效应,并且是数值精确的,因此被广泛应用于研究自旋-玻色子模型的时间演化^[9-17].但该方法的计算复杂度随着系统大小和时间关联长度呈指数增长,因此它实际所能计算的物理模型比较受限,大多数情况都局限于自旋-玻色子模型.

近年来,张量网络的研究和应用有了长足的发展^[18,19].2018年,Strathearn等^[20]利用矩阵乘积算符(matrix product operator, MPO)将QUAPI算法中的路径积分表达了成矩阵乘积态(matrix product state, MPS),使其计算复杂度从指数增长变成多项式增长,极大地提高了其计算效率和扩展了其应用范围.由此发展的方法被称为时间演化矩阵乘积算符(time-evolving matrix product operator, TEMPO)方法.TEMPO是一种高效的、数值精确的、非马尔可夫的算法,在量子开放系统领域具有广泛的应用前景.

本文对TEMPO算法及其应用进行综述.第2节首先给出Caldeira-Leggett模型和它的路径积分表达式.第3节给出QUAPI算法的一般性理论描述.第4节简单回顾了矩阵乘积态的概念.第5节给出了TEMPO算法的描述.第6节以自旋-玻色子为例展示了TEMPO算法的计算结果.第7节展示了如何使用TEMPO算法计算时间关联函数.第8节展示了多个浴的情况以及浴之间热流的计算.最后,第9节给出总结.

2 Caldeira-Leggett 模型的路径积分

在Caldeira-Leggett模型中,环境由一个由谐振子,即玻色子,构成的浴(bath)来描述,其哈密顿量可以分为3个部分:

$$\hat{H} = \hat{H}_S + \hat{H}_E + \hat{H}_{SE}. \quad (1)$$

这里 $\hat{H}_S$ 是系统的哈密顿量,其形式由具体的系统决定. $\hat{H}_E$ 是浴的哈密顿量,由一系列玻色子来描述:

$$\hat{H}_E = \sum_k \omega_k \hat{b}_k^\dagger \hat{b}_k, \quad (2)$$

其中 $\hat{b}_k^\dagger$ ($\hat{b}_k$)是状态 k 玻色子的产生(湮灭)算符.

系统和浴的耦合项为

$$\hat{H}_{SE} = \hat{s} \sum_k V_k (\hat{b}_k + \hat{b}_k^\dagger), \quad (3)$$

式中 $\hat{s}$ 是一个系统算符,为强度 V_k 与状态 k 的玻色子耦合.浴的性质由一个谱函数来描述:

$$J(\omega) = \sum_k V_k^2 \delta(\omega - \omega_k). \quad (4)$$

假设在初始时刻 $t = 0$,总的密度矩阵可以分离成系统和浴部分,即:

$$\hat{\rho}(0) = \hat{\rho}_S(0) \hat{\rho}_E, \quad (5)$$

式中 $\hat{\rho}_S(0)$ 是初始的系统密度矩阵.在初始时刻,可认为浴是处于热平衡的,其密度矩阵为

$$\hat{\rho}_E = e^{-\hat{H}_E/T}, \quad (6)$$

其中 T 代表浴的温度.为简单起见,令普朗克常数和玻尔兹曼常数均为1,即 $\hbar = k_B = 1$.

总密度矩阵根据冯·诺伊曼方程演化,在 t 时刻其形式解为 $\hat{\rho}(t) = e^{-i\hat{H}t} \hat{\rho}(0) e^{i\hat{H}t}$.这时对环境自由度求迹就可得到系统的约化密度矩阵 $\hat{\rho}_S(t) = \text{tr}_E[\hat{\rho}(t)]$.已知约化密度矩阵 $\hat{\rho}_S(t)$,任意系统算符 $\hat{f}$ 在 t 时刻的期望值都可以通过如下公式得到

$$\langle \hat{f}(t) \rangle = Z_S^{-1} \text{tr}_S [\hat{\rho}_S(t) \hat{f}], \quad (7)$$

其中 $Z_S = \text{tr}_S[\hat{\rho}_S(t)]$ 是系统的配分函数,这里 tr_S 代表对系统的自由度求迹.我们的任务便是计算出系统的约化密度矩阵.

约化密度矩阵 $\hat{\rho}_S(t)$ 可以很自然地表示成路径积分.将演化时间 t 分成 N 步,有 $\delta t = t/N$,并使 $N \rightarrow \infty$,则总密度矩阵可以写成:

$$\hat{\rho}(t) = e^{-i\hat{H}\delta t} \dots e^{-i\hat{H}\delta t} \hat{\rho}(0) e^{i\hat{H}\delta t} \dots e^{i\hat{H}\delta t}. \quad (8)$$

在每一时间点插入系统和浴的单位算符,并把从 $\hat{\rho}(0)$ 开始到最左边的时间点依次标记为 $(t_0^+, t_1^+, t_2^+, \dots, t_N^+)$,到最右边的时间点标记成 $(t_0^-, t_1^-, t_2^-, \dots, t_N^-)$.因为迹在循环置换后不变:

$$\begin{aligned} \langle \hat{f}(t) \rangle &= \text{tr} [e^{-i\hat{H}t} \hat{\rho}(0) e^{i\hat{H}t}] \\ &= \text{tr} [e^{i\hat{H}t} \hat{f} e^{-i\hat{H}t} \hat{\rho}(0)], \end{aligned} \quad (9)$$

可以认为系统从0时刻开始,经过 $e^{-i\hat{H}t}$ 的演化,在 t 时刻进行 $\hat{f}$ 算符的测量,然后经过 $e^{i\hat{H}t}$ 的演化回到0时刻.因此所有的时刻 $(t_0^+, t_1^+, t_2^+, \dots, t_N^+, t_N^-, t_{N-1}^-, t_{N-2}^-, \dots, t_0^-)$ 形成了闭合的时间回路 C ,一般

被称为 Keldysh 回路 [21–24], 见 图 1. 从 t_0^+ 到 t_N^+ 的路径对应着正向演化 $e^{-i\hat{H}t}$, 因此可称其为正向分支, 类似地, 称从 t_N^- 到 t_0^- 的路径为反向分支.

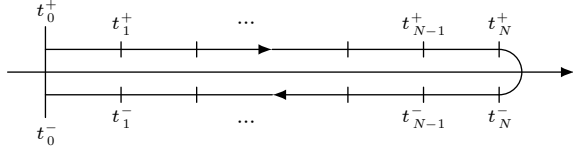


图 1 Keldysh 回路 C , 上部为正向分支, 下部为反向分支 [25]

Fig. 1. Keldysh Contour C , the upper is forward branch and the lower is backward branch [25].

为了方便描述路径积分, 可以在 Keldysh 回路上定义自由浴的格林函数. 首先在回路上定义顺序: 如果 t' 在 t'' 的前面, 就写为 $t' \succ t''$. 有了这个顺序, 可以在回路上定义出回路顺序的自由浴格林函数:

$$G_k(t', t'') = \langle T_C \hat{b}_k(t') \hat{b}_k^\dagger(t'') \rangle_0, \quad (10)$$

式中 T_C 是回路顺序算符, $\langle \cdot \rangle_0$ 对应着自由浴下的期望值. 具体来说:

$$G_k(t', t'') = \begin{cases} \langle \hat{b}_k(t') \hat{b}_k^\dagger(t'') \rangle_0, & t' \succeq t'', \\ \langle \hat{b}_k^\dagger(t'') \hat{b}_k(t') \rangle_0, & t'' \succ t'. \end{cases} \quad (11)$$

根据 t', t'' 在哪个分支, 回路顺序的格林函数可以分成非平衡格林函数的 4 个部分:

$$G_k(t', t'') = \begin{bmatrix} G_k^{++}(t', t'') & G_k^{+-}(t', t'') \\ G_k^{-+}(t', t'') & G_k^{--}(t', t'') \end{bmatrix}. \quad (12)$$

在 $G^{++}(G^-)$ 中, t', t'' 都在正向 (反向) 分支上. 在 $G^{+-}(G^{-+})$ 中, t' 在正向 (反向) 分支上, t'' 在反向 (正向) 分支上.

按照定义, 约化密度矩阵的矩阵元为

$$\langle s' | \hat{\rho}_S(t) | s'' \rangle = \text{tr}_E \left[\langle s' | e^{-i\hat{H}t} \hat{\rho}(0) e^{i\hat{H}t} | s'' \rangle \right]. \quad (13)$$

标记 $s' = s(t_N^+) = s_N^+$ 和 $s'' = s(t_N^-) = s_N^-$, 那么集合 $(s_0^+, s_1^+, s_2^+, \dots, s_N^+, s_N^-, s_{N-1}^-, s_{N-2}^-, \dots, s_0^-)$ 在回路 C 上形成了一个连续的路径 $s(t)$. 通过标准流程 [1,26–28], (13) 式约化密度矩阵的矩阵元可表示成路径积分:

$$\rho_S(s_N^\pm) = Z_E^{(0)} \int \mathcal{D}[s] K[s] I[s], \quad (14)$$

其中 $Z_E^{(0)}$ 是自由浴的配分函数, 对符号 $\mathcal{D}$ 的积分表示对在回路上边界条件为 $s_N^+ = s'$ 和 $s_N^- = s''$ 的所有可能路径的求和. $K[s]$ 是自由系统的传播子:

$$\begin{aligned} K[s] &= K(s_N^+, \dots, s_0^+, s_0^-, \dots, s_N^-) \\ &= \langle s_N^+ | e^{-i\hat{H}\delta t} | s_{N-1}^+ \rangle \cdots \langle s_1^+ | e^{-i\hat{H}\delta t} | s_0^+ \rangle \\ &\quad \times \langle s_0^+ | \hat{\rho}_S(0) | s_0^- \rangle \langle s_0^- | e^{i\hat{H}\delta t} | s_1^- \rangle \cdots \\ &\quad \times \langle s_{N-1}^- | e^{i\hat{H}\delta t} | s_N^- \rangle. \end{aligned} \quad (15)$$

表达式 $I[s]$ 被称为费曼-弗农影响泛函 (Feynman-Vernon influence functional), 其解析表达式可以写成在回路上的积分:

$$I[s] = \exp \left(- \int_C dt' \int_C dt'' s(t') \Delta(t', t'') s(t'') \right), \quad (16)$$

其中

$$\Delta(t', t'') = \sum_k V_k^2 G_k(t', t''). \quad (17)$$

根据谱函数的定义 (4), $\Delta(t', t'')$ 也可以写成:

$$\Delta(t', t'') = \int d\omega J(\omega) G_\omega(t', t''), \quad (18)$$

式中 $G_\omega(t', t'')$ 是 $\omega_k = \omega$ 时的 $G_k(t', t'')$.

我们也可以把积分写在普通的时间轴上. 在时间轴上, 自由浴的格林函数具体形式为

$$G_k^{++}(t' - t'') = \begin{cases} (1 + n_k) e^{-i\omega_k(t' - t'')}, & t' \geq t'', \\ n_k e^{-i\omega_k(t' - t'')}, & t' < t'', \end{cases} \quad (19)$$

$$G_k^-(t' - t'') = \begin{cases} n_k e^{-i\omega_k(t' - t'')}, & t' \geq t'', \\ (1 + n_k) e^{-i\omega_k(t' - t'')}, & t' < t'', \end{cases} \quad (20)$$

$$G_k^{+-}(t' - t'') = n_k e^{-i\omega_k(t' - t'')}, \quad (21)$$

$$G_k^{-+}(t' - t'') = (1 + n_k) e^{-i\omega_k(t' - t'')}, \quad (22)$$

式中, $n_k = (e^{\omega_k/T} - 1)^{-1}$ 是玻色-爱因斯坦分布函数. 将以上格林函数代入 (16) 式中, 并注意到面积分 $\int_0^t dt' \int_0^t dt''$ 可拆分成 $\int_{t' \geq t''} dt' dt''$ 和 $\int_{t' < t''} dt' dt''$, 影响泛函在时间轴上写成 [1,4,26]:

$$\begin{aligned} I[s] &= \exp \left\{ - \int_0^t dt' \int_0^t dt'' [s^+(t') - s^-(t')] \right. \\ &\quad \times [\alpha(t' - t'') s^+(t'') - \bar{\alpha}(t' - t'') s^-(t'')] \left. \right\}, \end{aligned} \quad (23)$$

其中 $\alpha(t)$ 是浴的自相关函数,

$$\alpha(t) = \int d\omega J(\omega) \left[\coth \left(\frac{\omega}{2T} \right) \cos \omega t - i \sin \omega t \right], \quad (24)$$

$\bar{\alpha}(t)$ 是它的复共轭.

3 准绝热传播子路径积分

第2节中的路径积分公式是解析的, 如果想对其进行数值求解, 需要进行离散化. 需要注意的是, 直接在影响泛函(23)中取有限的 δt , 即有限的 N , 并直接采用通常的 Trotter-Suzuki 分解^[29,30], 最后得到的结果偏差会很大. 因此需要一个更好的离散化方案, 也就是所谓的准绝热传播子路径积分(quasi-adiabatic propagator path integral, QUAPI)方法.

在 QUAPI 中, 路径 $s^\pm(t)$ 被分成很多等长的小段, 在每一小段 $(j-1/2)\delta t < t < (j+1/2)\delta t$ 中 $s^\pm(t) = s_j^\pm$ 保持不变. 如此, 在离散化后, 影响泛函(23)中的双重积分变成了双重求和:

$$I(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) = e^{-\sum_{j=0}^N \sum_{k=0}^j \phi_{jk}}, \quad (25)$$

$$\phi_{jk} = (s_j^+ - s_j^-)(\eta_{jk}s_k^+ - \bar{\eta}_{jk}s_k^-), \quad (26)$$

其中 η_{jk} 是一个复数, $\bar{\eta}_{jk}$ 是其复共轭. 当 $j \neq k$ 时, 可得

$$\eta_{jk} = \int_{(j-\frac{1}{2})\delta t}^{(j+\frac{1}{2})\delta t} dt' \int_{(k-\frac{1}{2})\delta t}^{(k+\frac{1}{2})\delta t} dt'' \alpha(t' - t''), \quad (27)$$

当 $j = k$ 时, 可得

$$\eta_{jj} = \int_{(j-\frac{1}{2})\delta t}^{(j+\frac{1}{2})\delta t} dt' \int_{(j-\frac{1}{2})\delta t}^{t'} dt'' \alpha(t' - t''). \quad (28)$$

这里展示的是最简单的 η_{jk} 离散方法, 每一时间段的间距相等. 在考虑更高精度的时候, 在一头一尾两端的时间段需要额外的处理, 更多的讨论可以参考文献[31–33].

经过离散化之后, $I(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$ 和 $K(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$ 是两个高维张量, 路径积分(14)可以写成张量的缩并:

$$\rho s(s_N^\pm) = \sum_{s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_{N-1}^\pm} F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm), \quad (29)$$

其中

$$F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) = K(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) I(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm). \quad (30)$$

如果定义

$$\tilde{K}(s_{k-1}^\pm, s_k^\pm) = \left\langle s_k^+ \left| e^{-i\hat{H}_S \delta t} \right| s_{k-1}^+ \right\rangle \left\langle s_{k-1}^- \left| e^{i\hat{H}_S \delta t} \right| s_k^- \right\rangle, \quad (31)$$

那么系统的自由传播子可以写成:

$$K(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) = \rho s(s_0^\pm) \tilde{K}(s_0^\pm, s_1^\pm) \cdots \tilde{K}(s_{N-1}^\pm, s_N^\pm). \quad (32)$$

影响泛函(25)可以写成:

$$I(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) = \prod_{k=0}^N I_0(s_k^\pm, s_k^\pm) \prod_{k=0}^{N-1} I_1(s_k^\pm, s_{k+1}^\pm) \cdots \times \prod_{k=0}^{N-\Delta k} I_{\Delta k}(s_k^\pm, s_{k+\Delta k}^\pm) \cdots I_N(s_0^\pm, s_N^\pm), \quad (33)$$

其中

$$I_{\Delta k}(s_k^\pm, s_{k+\Delta k}^\pm) = e^{-\phi_{k+\Delta k, k}}. \quad (34)$$

此时定义一个张量 A 为

$$A(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_{k+1}^\pm) = \tilde{K}(s_k^\pm, s_{k+1}^\pm) I_{k+1}(s_0^\pm, s_{k+1}^\pm) \cdots \times I_{\Delta k}(s_{k+1-\Delta k}^\pm, s_{k+1}^\pm) \cdots I_0(s_{k+1}^\pm, s_{k+1}^\pm), \quad (35)$$

则根据定义, 不同时间步数的 F 张量之间存在着递归关系

$$F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_{k+1}^\pm) = F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_k^\pm) A(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_{k+1}^\pm). \quad (36)$$

从初始条件 $F(s_0^\pm) = \rho s(s_0^\pm) I_0(s_0^\pm, s_0^\pm)$ 出发, 持续应用上面的递归关系, 原则上就能得到最终所需要的 $F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$.

为了简便, 将指标对 $\{s_k^+, s_k^-\}$ 简写成 s_k . 如果系统的希尔伯特空间大小为 M , 这就意味着将一个 $M \times M$ 大小的矩阵转化成了一个 M^2 大小的矢量. 同时将 F 和 A 张量写成带上标的张量:

$$F^{s_0, s_1, s_2, \dots, s_k} = F(s_0, s_1, s_2, \dots, s_k), \quad (37)$$

如此, 递归关系(36)就写成:

$$F^{s_0, s_1, s_2, \dots, s_{k+1}} = F^{s_0, s_1, s_2, \dots, s_k} A^{s_0, s_1, s_2, \dots, s_{k+1}}. \quad (38)$$

为了更好地使用张量语言, 也为之后引入矩阵乘积态和矩阵乘积算符, 定义一个 B 张量为

$$B_{r_0, r_1, r_2, \dots, r_k}^{s_0, s_1, s_2, \dots, s_{k+1}} = \left(\prod_{i=0}^k \delta_{r_i}^{s_i} \right) A^{s_0, s_1, s_2, \dots, s_{k+1}}, \quad (39)$$

式中 $\delta_{r_k}^{s_k}$ 是克罗内克函数. 那么递归关系(36)就可以写成爱因斯坦求和规则形式:

$$F^{s_0, s_1, s_2, \dots, s_{k+1}} = B_{r_0, r_1, r_2, \dots, r_k}^{s_0, s_1, s_2, \dots, s_{k+1}} F^{r_0, r_1, r_2, \dots, r_k}. \quad (40)$$

张量形式的递归公式 (40) 可以比较形象地用图形来表示, 见图 2(a). 考虑 $N = 4$ 时的情况, 图 2(b) 所示为利用递归关系从初始条件 F^{s_0} 开始演化到 $F^{s_0 s_1 s_2 s_3 s_4}$ 的步骤.

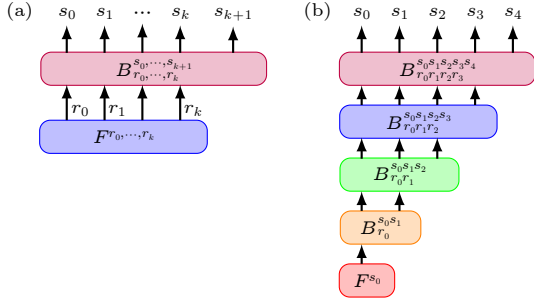


图 2 递归关系的示意图 (a) 递归公式 (40); (b) 从初始条件 F^{s_0} 开始演化 $N = 4$ 步的递推步骤

Fig. 2. An illustration of the recursive relation: (a) Recursive relation (40); (b) the iterative progress with $N = 4$ steps evolution from initial condition F^{s_0} .

虽然理论上可以从 F^{s_0} 出发, 持续采用 (40) 式来递推出最终的 $F^{s_0, s_1, s_2, \dots, s_N}$, 但是 $F^{s_0, s_1, s_2, \dots, s_k}$ 是一个 $(k+1)$ 维的张量, 它所占的内存大小和所需要的计算量都正比于 $M^{2(k+1)}$, 增长非常的快. 因此当 N 较大时, 递推过程得到一个较大的 k 值, 此时就不具备能力对其进行存储和计算.

为了解决这个问题, Makarov 和 Makri^[7,8] 敏锐地察觉到影响泛函里面的自相关函数 $\alpha(t)$ 会随着 $|t|$ 的增大而减小, 即使在零温的情况下依然如此. 也就是说, 虽然 F 张量里面的时间关联是非局域的, 但其关联长度依然有限, 所以可以对其进行截断从而减小存储空间和计算量. 图 3 所示为自相关函数 $\alpha(t)$ 的一种典型形状, 具体参数可以查看文献 [34]. 可以看出, 随着 $|t|$ 增大, $\alpha(t)$ 会衰减到近似为零.

因此, 将 $\alpha(t)$ 离散化而得到的 η_{jk} 会随着 $|j-k|$ 的增大而衰减, 当 $|j-k|$ 到达某个阈值 $\Delta k_{\max}$ 时便可以认为 η_{jk} 约等于零. 于是在张量 (35) 中, 可以认为所有 $\Delta k > \Delta k_{\max}$ 的 $I_{\Delta k} = 1$. 由此, 张量 B 从 $k+1$ 维的截断成了 $\Delta k_{\max} + 1$ 维:

$$B^{s_0, s_1, s_2, \dots, s_{k+1}} \rightarrow B^{s_{k+1-\Delta k_{\max}}, \dots, s_{k+1}}. \quad (41)$$

在这种情况下, 不再需要一直保存完整的 $k+1$ 维的张量 $F^{s_0, s_1, s_2, \dots, s_{k+1}}$, 而是保存一个 $\Delta k_{\max} + 1$ 的

张量 $F^{s_{k+1-\Delta k_{\max}}, \dots, s_{k+1}}$, 递归关系也由此变成:

$$F^{s_{k+1-\Delta k_{\max}}, \dots, s_{k+1}} = \sum_{r_{k-\Delta k_{\max}}} B_{r_{k+1-\Delta k_{\max}}, \dots, r_k}^{s_{k+1-\Delta k_{\max}}, \dots, s_{k+1}} F^{r_{k-\Delta k_{\max}}, \dots, r_k}. \quad (42)$$

假如 $\Delta k_{\max} = 2$, 截断之后的递归关系可以用图 4(a) 来展示. 图 4(b) 所示为 $N = 4$ 时截断之后的递归关系. 图中灰色方格代表 (42) 式的求和, 即对指标 $r_{k-\Delta k_{\max}}$ 的缩并. 这种情况下, F 张量的维度总是保持在 $\Delta k_{\max} + 1$, 极大减小了所需要的存储空间和计算量, 因此对量子开放系统的计算可以到任意长的演化时间, 使得 QUAPI 算法得到了广泛的应用.

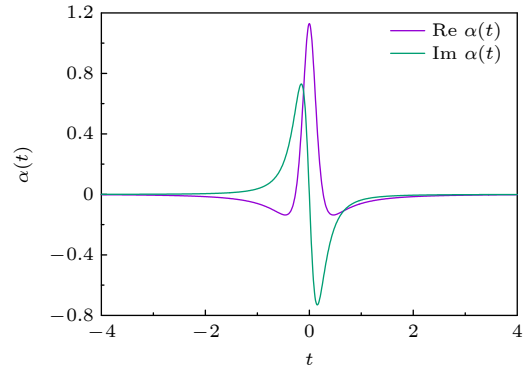


图 3 自相关函数 $\alpha(t)$ 的一种典型形状 [34]

Fig. 3. A typical shape of autocorrelation function $\alpha(t)$ [34].

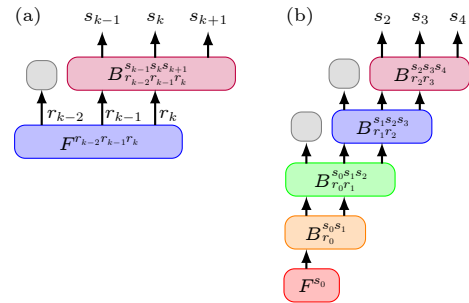


图 4 截断之后的递归关系 (其中灰色方格代表 (42) 式中的求和) (a) 递归公式 (42); (b) 从初始条件 F^{s_0} 开始演化到 $N = 4$ 步时的递推步骤

Fig. 4. An illustration of recursive relation after truncation, here the gray square stands for the summation in Eq. (42): (a) Recursive relation (42); (b) the iterative progress of $N = 4$ steps evolution with initial condition F^{s_0} .

4 矩阵乘积态

从第 3 节可以看出, $\Delta k_{\max}$ 的取值取决于自相关函数 $\alpha(t)$ 的衰减情况以及离散时间间隔 δt 的取值. 以图 3 为例, 在 $|t| > 4$ 的时候进行截断, 即认为 $|t| > 4$ 时 $\alpha(t) = 0$. 如果取一个相对较大的时间间

隔 $\delta t = 0.25$, 那么需要 $\Delta k_{\max} = 8$ 来覆盖 $\alpha(t)$ 中的关联. 对于自旋-玻色子模型, 其系统希尔伯特空间大小为 $M = 2$, 这个 $\Delta k_{\max}$ 在实际计算中是可以接受的. 但是如果取更小的时间间隔, 比如 $\delta t = 0.05$, 那么所需的截断参数变成 $\Delta k_{\max} = 80$. 在实际计算中, 该值就算对于最简单的自旋-玻色子模型也是不可接受的. 对于更加复杂的量子开放系统, 系统会更大, 这样就算比较小的 $\Delta k_{\max}$ 也可能需要非常大的存储空间, 从而无法进行计算. 另外, 在某些情况下自相关函数 $\alpha(t)$ 的关联长度也可能变得很长, 从而在 δt 不变的情况也需要更大 $\Delta k_{\max}$. 因此, 在实际计算中, 第 3 节所介绍的原始的准绝热传播子路径积分方法存在种种限制, 从而影响其适用范围. 也是因为这样的原因, 它的大部分应用都集中在最简单的自旋-玻色子模型上.

一般情况下物理态和物理算符可以用张量来描述, 而张量可以天然地采用张量网络来表达, 如此所对应的结构即是矩阵乘积态和矩阵乘积算符. 张量网络在物理学中具有广泛的应用, 例如一维系统的性质 [35–37]、密度矩阵重整化 [18,38]、张量重整化 [39,40]、二维系统的性质 [41–43]、费米系统高斯态的压缩 [44]、量子主方程 [45]、动力学平均场理论中的杂质求解器 [46–49] 等.

当希望压缩一个张量的体积时, 很自然会想到用矩阵乘积态来表示. 此处笔者简要介绍矩阵乘积态的中心思想, 更多的信息可参考综述 [18, 19]. 假如有一系列的一维张量, 即矢量: $A^{s_0}, A^{s_1}, A^{s_2}, \dots, A^{s_k}$. 可以通过其元素乘积将这些矢量合成一个张量 $A^{s_0} A^{s_1} \dots A^{s_k}$. 如果反过来能把一个张量分解成矢量元素的乘积, 就能给计算和操作张量带来极大的便利. 假如张量可以进行以下分解:

$$A^{s_0, s_1, s_2, \dots, s_k} = A^{s_0} A^{s_1} A^{s_2} \dots A^{s_k}. \quad (43)$$

考虑对张量某个指标, 简单起见取 s_0 的缩并 $\sum_{s_0} A^{s_0, s_1, s_2, \dots, s_k}$. 如果张量不可分解, 对 s_0 的缩并需要涉及到其他所有 s_k , 计算量随着指标个数呈现指数增长. 而如果张量可以分解, 那么这个缩并就变成简单的

$$\sum_{s_0} A^{s_0, s_1, s_2, \dots, s_k} = \left(\sum_{s_0} A^{s_0} \right) A^{s_1} A^{s_2} \dots A^{s_k}. \quad (44)$$

这里只需要处理 A^{s_0} , 这与处理一个矢量所需要的计算量相同. 再考虑对多个, 甚至所有指标进行缩并, 那么可得

$$\begin{aligned} & \sum_{s_0, s_1, s_2, \dots, s_k} A^{s_0, s_1, s_2, \dots, s_k} \\ &= \left(\sum_{s_0} A^{s_0} \right) \left(\sum_{s_1} A^{s_1} \right) \\ & \times \left(\sum_{s_2} A^{s_2} \right) \dots \left(\sum_{s_k} A^{s_k} \right). \end{aligned} \quad (45)$$

可以看到所有的指标均分开处理, 因此整体所需要的计算量随着指标个数呈线性增长. 这种情况可认为张量不同指标之间是完全不耦合的, 不耦合的张量在其他操作上也会有类似的便利.

当然, 一般的张量都不能进行类似分解. 但可以退而求其次, 不再要求 A^{s_k} 里面的元素是简单的标量, 而是允许它们是一维或者二维张量. 也就是说, 可寻求对张量的如下分解:

$$A^{s_0, s_1, s_2, \dots, s_k} = A_{\alpha_0}^{s_0} A_{\alpha_0 \alpha_1}^{s_1} \dots A_{\alpha_{k-2} \alpha_{k-1}}^{s_{k-1}} A_{\alpha_{k-1}}^{s_k}, \quad (46)$$

式中相同的 α_k 指标代表求和. 这个分解的头尾 $A_{\alpha_0}^{s_0}, A_{\alpha_{k-1}}^{s_k}$ 是二维张量, 其他地方 $A_{\alpha_{m-1} \alpha_m}^{s_m}$ 是三维张量. 在固定指标 s_m, s_{m+1} 时, $A_{\alpha_{m-1} \alpha_m}^{s_m}$ 和 $A_{\alpha_m \alpha_{m+1}}^{s_{m+1}}$ 是矩阵, $A_{\alpha_{m-1} \alpha_m}^{s_m} A_{\alpha_m \alpha_{m+1}}^{s_{m+1}}$ 是它们的矩阵乘积 (相同的 α_m 代表求和). 于是张量被分解成一系列矩阵的乘积, 这样的表达方式被称为矩阵乘积态.

尽管矩阵乘积态里面不是简单的标量而是矩阵, 在进行张量操作时其仍然比直接操作整个张量便利得多. 只要矩阵的大小在控制的范围内, 将张量表示成矩阵乘积态可以极大地减少张量操作所需要的计算量. 不同于完全不耦合的张量, 矩阵乘积态里面不同指标之间还存在耦合, 其耦合程度体现在矩阵的大小, 矩阵越大那么其耦合就越强.

除了张量操作上的便利性之外, 矩阵乘积态还有相应的压缩算法, 可以极大地减小其体积. 主流的压缩算法基于矩阵的奇异值分解 (singular value decomposition, SVD), 下面简要介绍该算法. 对于一个三维张量 $A_{\alpha\beta}^s$, 可以把指标 s 和 α 合并, 把这个张量看成一个矩阵 $A_{s\alpha, \beta}$. 对该矩阵进行奇异值分解, 可得

$$A_{s\alpha, \beta} = U_{s\alpha, k} \lambda_k V_{k\beta}, \quad (47)$$

其中, U, V 都是矩阵, λ_k 是相应的奇异值, 指标 k 被求和. 将指标 s 和 α 再次分开, 可得

$$A_{\alpha\beta}^s = U_{\alpha k}^s \lambda_k V_{k\beta}. \quad (48)$$

压缩算法的思路是丢掉可以忽略的奇异值, 具体做

法是设置一个阈值 λ_c , 然后丢掉小于该阈值的奇异值. 考虑矩阵乘积 $A_{\alpha_0\alpha_1}^{s_1} A_{\alpha_1\alpha_2}^{s_2}$, 对前一项进行奇异值分解之后可得

$$\begin{aligned} A_{\alpha_0\alpha_1}^{s_1} A_{\alpha_1\alpha_2}^{s_2} &= U_{\alpha_0k}^{s_1} (\lambda_k V_{k\alpha_1} A_{\alpha_1\alpha_2}^{s_2}) \\ &= \tilde{A}_{\alpha_0k}^{s_1} \tilde{A}_{k\alpha_2}^{s_2}. \end{aligned} \quad (49)$$

这里可以看出, 通过奇异值分解然后重新组合, 可得到新的乘积态. 注意这里 α_1 指标已经被缩并, k 指标充当新的 α_1 指标. 在丢弃部分奇异值之后, k 指标的个数会变少. 在矩阵乘积态 (46) 中, 最简单的压缩的方法^[18]是先从左往右重复上述操作, 但这个时候并不压缩, 然后再从右往左的重复上述操作, 并在这个过程中进行压缩丢弃相应的奇异值. 整个过程结束后就完成了对矩阵乘积态的压缩.

在时间演化矩阵算符方法中, 奇异值一般依照与最大奇异值的比值来选取. 举例来说, 令 $\lambda_{\max}$ 为最大的奇异值, 那么可丢弃小于 $c\lambda_{\max}$ 的奇异值. 这个比值 c 可以很小, 例如在实际计算时取 $c = 10^{-8}$, 所以压缩过程带来的误差也可以很小.

5 时间演化矩阵乘积算符方法

Strathearn 等^[20]在 2018 年应用矩阵乘积态来表示 QUAPI 算法中的张量, 极大地减小了存储需求和计算量. 他们将由此得到的方法命名为时间演化矩阵乘积算符 (time-evolving matrix product operators, TEMPO) 方法.

假设张量 $F^{s_0, s_1, s_2, \dots, s_k}$ 已被表示成矩阵乘积态:

$$F^{s_0, s_1, s_2, \dots, s_k} = F_{\alpha_0}^{s_0} F_{\alpha_0\alpha_1}^{s_1} \dots F_{\alpha_{k-2}\alpha_{k-1}}^{s_{k-1}} F_{\alpha_{k-1}}^{s_k}. \quad (50)$$

图 5(a) 所示为 (50) 式的示意图. 图中开放的箭头是张量的指标, 封闭的箭头代表缩并. 希望经过递归公式 (40) 之后, 该张量依然保持矩阵乘积态的形式. Strathearn 等^[20]巧妙地观察到, 公式中的张量 $B_{r_0, r_1, r_2, \dots, r_k}^{s_0, s_1, s_2, \dots, s_{k+1}}$ 可以改写成:

$$\begin{aligned} B_{r_0, r_1, r_2, \dots, r_k}^{s_0, s_1, s_2, \dots, s_{k+1}} &= [b_{k+1}]_{r_0}^{s_0\alpha_0} [b_k]_{r_1\alpha_0}^{s_1\alpha_1} \dots [b_{k+1-m}]_{r_m\alpha_{m-1}}^{s_m\alpha_m} \dots \\ &[b_1]_{r_k\alpha_{k-1}}^{s_k\alpha_k} [b_0]_{\alpha_k}^{s_{k+1}}. \end{aligned} \quad (51)$$

在最前面的三阶张量为

$$[b_{k+1}]_{r_0}^{s_0\alpha_0} = I_{k+1}(s_0, \alpha_0) \delta_{r_0}^{s_0}. \quad (52)$$

当 $m < k$ 时, 中间的四阶张量为

$$[b_{k+1-m}]_{r_m\alpha_{m-1}}^{s_m\alpha_m} = I_{k+1-m}(s_m, \alpha_m) \delta_{r_m}^{s_m} \delta_{\alpha_{m-1}}^{\alpha_m}, \quad (53)$$

当 $m = k$ 时, 可得

$$[b_1]_{r_k\alpha_{k-1}}^{s_k\alpha_k} = \tilde{K}(s_k, \alpha_k) I_1(s_k, \alpha_k) \delta_{r_m}^{s_m} \delta_{\alpha_{m-1}}^{\alpha_m}. \quad (54)$$

最后的二阶张量为

$$[b_0]_{\alpha_k}^{s_{k+1}} = I_0(s_{k+1}, s_{k+1}) \delta_{\alpha_k}^{s_{k+1}}. \quad (55)$$

这里相同的指标代表求和. 图 5(b) 所示为 (55) 式的示意图. 可以看到和矩阵乘积态不同, 这个图里每个矩阵块有两个开放的箭头, 对应着两个开放的指标, 这样的张量表达形式被称为矩阵乘积算符.

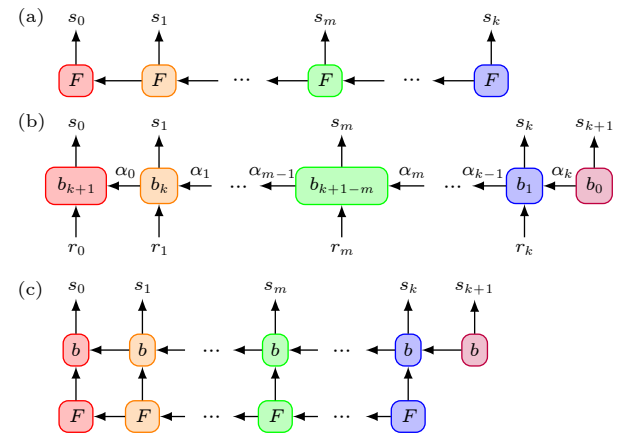


图 5 TEMPO 算法中的一些张量网络图示 (a) F 张量的矩阵乘积态表示; (b) B 张量的矩阵乘积算符表示; (c) 递归关系 (40) 的张量网络表示

Fig. 5. An illustration of some tensor networks in TEMPO: (a) The matrix product states representation of F tensor; (b) the matrix product operators representation of B tensor; (c) the tensor network representation of recursive relation (40).

矩阵乘积算符下面的开放指标可以和矩阵乘积态的指标缩并得到新的矩阵乘积态, 这对应着算符作用到态上. 简单理解就是矩阵乘积算符是把算符用矩阵乘积态表达出来. 因此递归关系 (40) 可以表示成矩阵乘积算符作用到矩阵乘积态上, 见图 5(c). 如此, 重复应用矩阵乘积态的递归关系, 并在每次递推之后应用压缩算法, 就能得到最终 F 张量的矩阵乘积态表达形式.

在这个过程中, 仍然可以利用自相关函数中的时间关联是有限的性质来减少所需要的计算量. 这部分和原始的 QUAPI 算法一致, 只是现在张量用矩阵乘积态来表达. 给定截断参数 $\Delta k_{\max}$ 之后, 张量 B 做相应的截断, 见 (41) 式. 假设 $\Delta k_{\max} = 2$, 从初始条件演化 $N = 4$ 步的张量网络见图 6.

足够覆盖非局域的时间关联. 从图 8 可以看出, 当 $\lambda = 0.1$ 耦合较弱的时候, $\langle \hat{\sigma}_z(t) \rangle$ 呈现衰减的振荡. 当处于较强的耦合 $\lambda = 0.4$ 时, $\langle \hat{\sigma}_z(t) \rangle$ 的振荡消失, 只留下单纯的衰减. 当耦合很强, 即 $\lambda = 1.0$ 时, $\langle \hat{\sigma}_z(t) \rangle$ 最终稳定在一个固定值, 这对应着自旋-玻色子模型的局域化相变. 关于此相变可以参考文献 [1, 6, 64, 65]. 这里需要注意的是, 如果模拟时间足够长, $\langle \hat{\sigma}_z(t) \rangle$ 会慢慢地衰减到零, 也就是说这个算法并不能真实反映该相变. Strathearn 等 [20] 认为这是由于并没有真正考虑到无穷长的关联造成, 并最终采用拟合的方法得到相变点大致的位置.

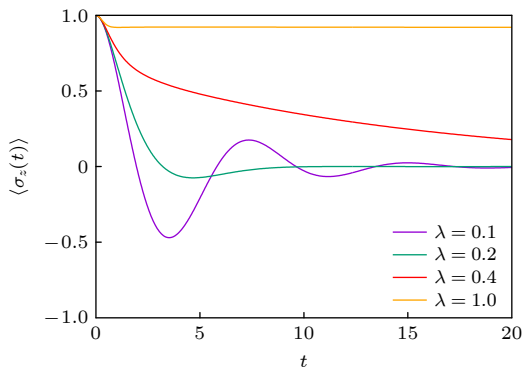


图 8 $\langle \hat{\sigma}_z(t) \rangle$ 在不同耦合强度 λ 下的演化

Fig. 8. Dynamics of $\langle \hat{\sigma}_z(t) \rangle$ with respect to different coupling strength λ .

7 关联函数的计算

时间演化矩阵乘积算符方法不仅可以高效地计算约化密度矩阵, 也可以用来计算系统的关联函数. 令 f 为系统的一个算符, 其多时关联函数的定义为

$$\langle \hat{f}(t_1) \hat{f}(t_2) \cdots \hat{f}(t_n) \rangle = \text{tr} [\hat{\rho}(0) \hat{f}(t_1) \hat{f}(t_2) \cdots \hat{f}(t_n)]. \quad (60)$$

尽管这里 $\hat{f}$ 算符只作用在系统上, 但其关联函数的计算仍然需要考虑环境的影响, 因此其属于非马尔可夫问题.

因为 $\hat{f}$ 算符只涉及到系统, 所以它只改变路径积分公式 (14) 中的系统传播子 K . 也就是说, 只需要在相应的时间点把 $\tilde{K}(s_{k-1}, s_k)$ 从 (31) 式变成:

$$\langle s_k^+ | e^{-i\hat{H}_S \delta t} \hat{f} | s_{k-1}^+ \rangle \langle s_{k-1}^- | e^{i\hat{H}_S \delta t} | s_k^- \rangle \quad (61)$$

或者

$$\langle s_k^+ | e^{-i\hat{H}_S \delta t} | s_{k-1}^+ \rangle \langle s_{k-1}^- | \hat{f} e^{i\hat{H}_S \delta t} | s_k^- \rangle, \quad (62)$$

就可得到新的系统传播子张量 $K(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$. 形象地说, 在指定位置把算符 f 插入系统传播子 $K(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$ 中. 具体是插入正向分支还是反向分支则依据定义或者相应的规定. 有了新的 K , 再应用 (30) 式可以得到新的 F 张量, 记作 $\tilde{F}(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm)$ 张量, 然后应用 (29) 式对其进行缩并可得到一个“约化密度矩阵”:

$$\tilde{\rho}_S(s_N^\pm) = \sum_{s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_{N-1}^\pm} \tilde{F}(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm). \quad (63)$$

对其求迹则得到了所求的关联函数.

从上述讨论可以看出, 对于不同时间点的关联函数, 其影响泛函都一样, 只是系统的传播子不同. 于是为了重复计算不同时间点的关联函数, Jørgensen 和 Pollock [50] 在 2019 年指出可先将影响泛函 (25) 用矩阵乘积态表示出来, 然后再单独计算系统的传播子. 因为非局域的时间关联都包含在影响泛函里面, 其计算部分耗时最长, 而系统传播子的计算只涉及相邻时间点, 其计算耗时较短, 所以他们的做法可以节省大量的计算量.

文献 [34] 考虑自旋-玻色子模型中的量子随机共振现象, 所涉及的系统算符为 σ_z . 在该问题中, $\hat{f}$ 算符在选择系统表象下是对角的, 于是可得

$$\hat{\sigma}_z |s_k^\pm\rangle = s_k^\pm |s_k^\pm\rangle, \quad (64)$$

那么 $\tilde{F}$ 张量就可以简单地写成:

$$\begin{aligned} \tilde{F}(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) \\ = F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) s_{k_1}^- s_{k_2}^- \cdots s_{k_n}^-, \end{aligned} \quad (65)$$

这里 $k_1 \delta t = t_1, \dots, k_n \delta t = t_n$. 在这个问题中关注的是双时关联函数, 因此“约化密度矩阵”为

$$\begin{aligned} \tilde{\rho}_S(s_N^\pm; s_{k_1}^-, s_{k_2}^-) \\ = \sum_{s_0^\pm, \dots, s_{N-1}^\pm} F(s_0^\pm, s_1^\pm, s_2^\pm, \dots, s_N^\pm) s_{k_1}^- s_{k_2}^-, \end{aligned} \quad (66)$$

对其求迹则得到了双时关联函数:

$$\langle \hat{\sigma}_z(t_1) \hat{\sigma}_z(t_2) \rangle = \text{tr}_S [\tilde{\rho}_S(s_N^\pm; s_{k_1}^-, s_{k_2}^-)]. \quad (67)$$

在自旋-玻色子模型中, 如果系统受到周期性的驱动, 最终物理量 $\langle \hat{\sigma}_z(t) \rangle$ 会做周期性的振荡. 环境噪声会对系统造成耗散, 所以一般情况下环境会压制振荡的振幅. 但在某些情况下, 噪声反而会增

强系统的振荡, 这就是量子随机共振现象^[66-68]. 在早期的量子共振现象中, 研究者们关注 $\langle \hat{\sigma}_z(t) \rangle$ 的振幅随着噪声强度的变化, 但在之后研究发现可以用信噪比更好地描述该现象.

周期性驱动的两能级系统的哈密顿量可以写成:

$$\hat{H}_S(t) = \frac{\Delta}{2} \hat{\sigma}_x + \frac{E}{2} \hat{\sigma}_z \cos \Omega t, \quad (68)$$

式中 $\hat{\sigma}_z, \hat{\sigma}_x$ 是泡利矩阵, Δ 为隧穿强度, E 是驱动外场的强度, Ω 是驱动的频率. 因为存在周期性驱动, 在最终稳态下, 系统会做周期性的振荡. 为了求得信噪比, 需要计算稳态下对称关联函数:

$$\begin{aligned} C(t_1, t_2) &= \frac{1}{2} \langle \hat{\sigma}_z(t_1) \hat{\sigma}_z(t_2) + \hat{\sigma}_z(t_2) \hat{\sigma}_z(t_1) \rangle \\ &= \text{Re} \langle \hat{\sigma}_z(t_1) \hat{\sigma}_z(t_2) \rangle. \end{aligned} \quad (69)$$

这个关联函数是双周期的, 即:

$$C(t_1, t_2) = C\left(t_1 + \frac{2\pi}{\Omega}, t_2 + \frac{2\pi}{\Omega}\right). \quad (70)$$

因为 $\hat{\sigma}_z$ 算符是对角的, 因此可以直接把 F 张量, 而

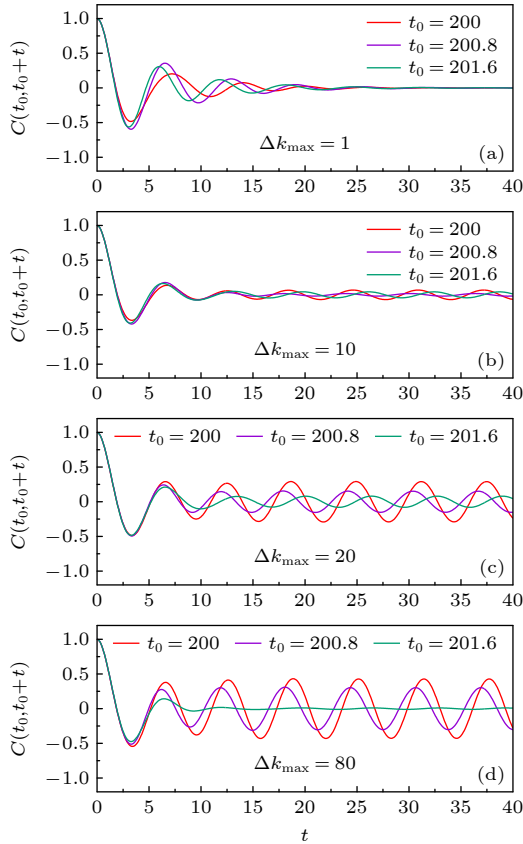


图 9 在不同 $\Delta k_{\max}$ 和 t_0 下的关联函数 $C(t_0, t_0 + t)$ ^[34]

Fig. 9. Correlation function $C(t_0, t_0 + t)$ with respect to different $\Delta k_{\max}$ and t_0 ^[34].

不是影响泛函, 表达成矩阵乘积态. 这样就可以把时间步数取的很长以便达到系统的稳态. 实际计算中, 只需要考虑 $t_2 \geq t_1$ 的情况, 所以考虑 $t > 0$ 时的关联函数 $C(t_0, t_0 + t)$. 以文献^[34]中的模型为例, 图 9 为不同 $\Delta k_{\max}, t_0$ 下的关联函数 $C(t_0, t_0 + t)$. 可以看到 t_0 时刻的选取对关联函数具有很大的影响.

8 多浴之间的热流

TEMPO 算法可以很自然地推广到多个浴的情况. 假如有多个浴, 单个浴的哈密顿量为

$$\hat{H}_\alpha = \sum_k \omega_{\alpha k} \hat{b}_{\alpha k}^\dagger \hat{b}_{\alpha k}, \quad (71)$$

其中 $\hat{b}_{\alpha k}^\dagger$ ($\hat{b}_{\alpha k}$) 产生 (湮灭) α 浴中 k 态的玻色子. 系统和浴之间的相互作用则为

$$H_{S\alpha} = \hat{s}_\alpha \sum_k V_{\alpha k} (\hat{b}_{\alpha k} + \hat{b}_{\alpha k}^\dagger). \quad (72)$$

其中 $\hat{s}_\alpha$ 系统耦合到 α 浴的一个算符, $V_{\alpha k}$ 是相应的耦合强度. 此处可简单认为不同 α 的 $\hat{s}_\alpha$ 之间相互对易. 这种情况下, 总的影响泛函可以写成:

$$I[s] = \prod_\alpha I_\alpha[s], \quad (73)$$

其中 $I_\alpha[s]$ 和单个浴的影响泛函 (23) 有着相同的形式:

$$I_\alpha[s] = \exp\left(-\int_C dt' \int_C dt'' s_\alpha(t') \Delta_\alpha(t', t'') s_\alpha(t'')\right), \quad (74)$$

式中

$$\Delta_\alpha(t', t'') = \sum_k V_{\alpha k}^2 G_{\alpha k}(t', t''), \quad (75)$$

$G_{\alpha k}(t', t'')$ 是 α 浴的自由格林函数. 具体理论推导可以参考文献^[25, 59].

根据多浴的影响泛函 (73), F 张量 (30) 可以很自然地用时间演化矩阵乘积算符方法表达成矩阵乘积态, 影响泛函 $I[s]$ 本身也可以很自然地表达成矩阵乘积态. 于是多浴情况下的约化密度矩阵以及关联函数, 都可以高效地采用 TEMPO 算法来计算.

如果多个浴的温度不一样, 那么热量会通过系统从高温的浴流向低温的浴. 到达稳态时候, 浴之间就会有稳定的热流. 因为热流直接影响纳米器件的性能, 所以如何计算热流一直是个重要的研究领域^[24,69-73].

笔者在文献^[25]中展示了如何应用时间演化

矩阵乘积算符来计算不同浴之间的热流. 从 α 浴流出的热流可以定义成该浴能量变化率的相反数

$$I_\alpha(t) = -\langle \dot{\hat{H}}_\alpha(t) \rangle = -i \langle [\hat{H}, \hat{H}_\alpha(t)] \rangle. \quad (76)$$

于是可得

$$\begin{aligned} I_\alpha(t) &= i \sum_k V_{\alpha k} \omega_{\alpha k} \langle \hat{s}_\alpha(t) [\hat{b}_{\alpha k}(t) - \hat{b}_{\alpha k}^\dagger(t)] \rangle \\ &= -2\text{Im} \sum_k V_{\alpha k} \omega_{\alpha k} \langle \hat{s}_\alpha(t) \hat{b}_{\alpha k}(t) \rangle. \end{aligned} \quad (77)$$

从这里可以看出, 计算热流需要计算系统算符和浴算符的关联函数. 为求得此关联函数, 可以定义一个有源约化密度矩阵为

$$\hat{\rho}_S^{\xi_\alpha}(t) = \text{tr}_E \left[\hat{\rho}(t) \exp \left(\xi_\alpha \sum_k V_{\alpha k} \omega_{\alpha k} \hat{b}_{\alpha k} \right) \right]. \quad (78)$$

对 ξ_α 进行微分就有

$$I_\alpha(t) = -2\text{Im} Z_S^{-1} \text{tr}_S \left[\frac{\delta \hat{\rho}_S^{\xi_\alpha}(t)}{\delta \xi_\alpha} \right] \Big|_{\xi_\alpha=0}. \quad (79)$$

在 t 时刻, 从 α 浴流出的总热量 $Q_\alpha(t)$ 则由热流的时间积分给出:

$$Q_\alpha(t) = \int_0^t I_\alpha(t') dt'. \quad (80)$$

热量 $Q_\alpha(t)$ 一部分流入系统和其他浴, 剩下的保存在系统和浴之间的相互作用里面, 在这里称其为相互作用能量. 系统与 α 浴之间在时刻 t 的相互作用能量写为

$$W_\alpha(t) = \langle \hat{H}_{S\alpha}(t) \rangle = 2\text{Re} \sum_k V_{\alpha k} \langle \hat{s}_\alpha(t) \hat{b}_{\alpha k}(t) \rangle. \quad (81)$$

这个相互作用能量在输运过程起着重要的作用^[74-77]. 类似地, 为了求 $W_\alpha(t)$ 可以定义出另外一个有源约化密度矩阵:

$$\hat{\rho}_S^{\chi_\alpha}(t) = \text{tr}_E \left[\hat{\rho}(t) \exp \left(\chi_\alpha \sum_k V_{\alpha k} \hat{b}_{\alpha k} \right) \right], \quad (82)$$

对 χ_α 微分则有

$$W_\alpha(t) = 2\text{Re} Z_S^{-1} \text{tr}_S \left[\frac{\delta \hat{\rho}_S^{\chi_\alpha}(t)}{\delta \chi_\alpha} \right] \Big|_{\chi_\alpha=0}. \quad (83)$$

系统本身的能量变化, 可采用

$$E(t) = \langle \hat{H}_S(t) \rangle - \langle \hat{H}_S(0) \rangle \quad (84)$$

来表示. 那么模型的能量守恒就可以用 $Q(t)$, $E(t)$ 和 $W(t)$ 表达成:

$$\sum_\alpha Q_\alpha(t) = E(t) + \sum_\alpha W_\alpha(t), \quad (85)$$

或者用相应的时间微分表达成:

$$\sum_\alpha I_\alpha(t) = \dot{E}(t) + \sum_\alpha \dot{W}(t). \quad (86)$$

在路径积分表达中, 有源密度矩阵充当了生成函数的角色, 可以用 TEMPO 算法将其表达成矩阵乘积态, 然后通过数值微分就可以求得热流和相互作用能量. 还是以自旋-玻色子模型为例, 考虑有两个浴的情况, 分别为浴 1 和浴 2, 具体的模型和参数见文献^[25]. 当浴温度不同时, 最终会在两浴之间形成稳定的热流, 其相应的过程见图 10. 在这里浴 1 的温度 T_1 远高于浴 2 的温度 T_2 , 从图 10(a) 可以看到, 在最开始的一段时间里 $I_1(t)$ 和 $I_2(t)$ 都是负值, 这表明了浴的能量其实是在增大的. 这是因为浴和系统开始耦合形成负的相互作用能量 $W_1(t)$ 和 $W_2(t)$, 见图 10(d), 于是能量就流入了浴. 当 t 足够长的时候, 系统和环境的耦合趋于稳定, 于是就有了稳定的热流 $I_1(t)$ 和 $I_2(t)$. 其中 $I_1(t)$ 为正, $I_2(t)$ 为负, 并且有 $I_1(t) + I_2(t) = 0$, 表明形成了从浴 1 到浴 2 的稳定热流.

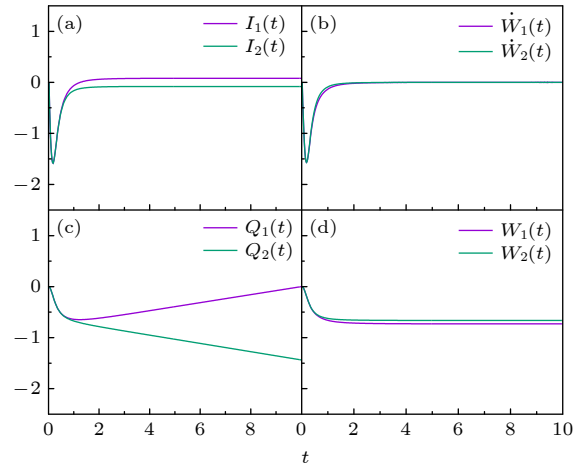


图 10 系统和浴在 T_1 远大于 T_2 时的演化过程^[25]

Fig. 10. Evolution of system and bath when T_1 is much larger than T_2 ^[25].

现在考虑两个浴处于相同温度, 即 $T_1 = T_2$, 但系统和浴耦合强度不同, 即 $\lambda_1 \neq \lambda_2$, 的情况. 这里令 λ_1 远大于 λ_2 , 其演化过程见图 11. 从图 11(a) 可以看到, 在刚开始的一段时间里, $I_1(t)$ 和 $I_2(t)$ 也是负的, 表示热量是流入浴的, 这是系统和浴之间的耦合造成的. 这时 $I_1(t)$ 的绝对值远大于 $I_2(t)$ 的绝对值, 因为系统和浴 1 的耦合强度要大得多, 同

样, 最终的相互作用能量也是 $W_1(t)$ 的绝对值远大于 $W_2(t)$ 的绝对值, 见图 11(d). 当演化时间足够长时, $I_1(t)$ 和 $I_2(t)$ 都归零, 这表明尽管耦合强度不同, 温度相同的两个浴之间也不存在热流.

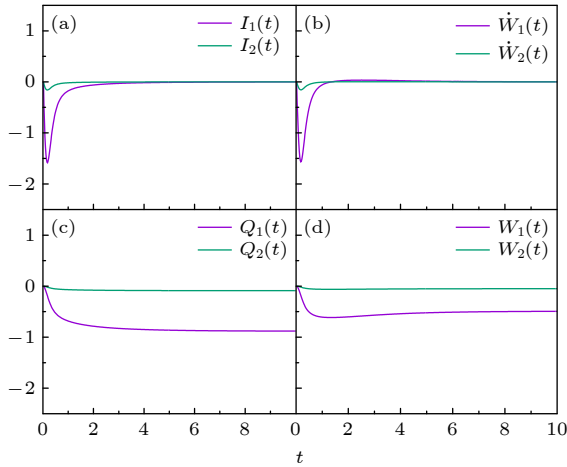


图 11 系统和浴在 $T_1 = T_2$ 但 λ_1 远大于 λ_2 时的演化过程^[25]

Fig. 11. Evolution of system and bath when $T_1 = T_2$ but λ_1 is larger than λ_2 ^[25].

9 总 结

准绝热传播子路径积分方法发明于 20 世纪 90 年代, 是为数不多的数值精确的计算量子开放系统的方法. 但是它的计算复杂度随着系统大小和时间关联长度呈现指数增长, 因此极大限制了其应用范围. 近年来, 应用张量网络的理论, 研究者们把准绝热传播子路径积分方法中的张量用矩阵乘积态表达, 从而得到了所谓的时间演化矩阵乘积算符方法. 该方法的计算复杂度呈现多项式增长, 极大减小所需要的计算量, 因此极大扩展了可应用范围. 总的来说, 时间演化矩阵乘积算符方法是一个高效且数值精确的量子开放系统的计算方法, 并且完全包含了环境所造成的非马尔可夫效应. 该方法在问世的这几年来内, 已经取得了一些关于量子开放系统的重要成果, 并在未来有着更广阔的应用前景. 本文对该算法及其相关应用进行综述, 并用量子随机共振现象中的关联函数以及多浴模型中的热流的计算来展示该方法的计算效果.

参考文献

- [1] Weiss U 1993 *Quantum Dissipative Systems* (Singapore: World Scientific)
- [2] Breuer H P 2007 *The Theory of Open Quantum Systems* (New York: Oxford University Press)
- [3] Caldeira A O, Leggett A J 1981 *Phys. Rev. Lett.* **46** 211
- [4] Caldeira A O, Leggett A J 1983 *Physica A* **121** 587
- [5] Caldeira A, Leggett A 1983 *Ann. Phys.* **149** 374
- [6] Leggett A J, Chakravarty S, Dorsey A T, Fisher M P A, Garg A, Zwenger W 1987 *Rev. Mod. Phys.* **59** 1
- [7] Makarov D E, Makri N 1994 *Chem. Phys. Lett.* **221** 482
- [8] Makri N 1995 *J. Math. Phys.* **36** 2430
- [9] Makarov D E, Makri N 1995 *Phys. Rev. B* **52** R2257
- [10] Makarov D E, Makri N 1995 *Phys. Rev. E* **52** 5863
- [11] Golosov A A, Friesner R A, Pechukas P 1999 *J. Chem. Phys.* **110** 138
- [12] Shao J, Makri N 2002 *J. Chem. Phys.* **116** 507
- [13] Dong K, Makri N 2004 *Chem. Phys.* **296** 273
- [14] Dong K, Makri N 2004 *Phys. Rev. A* **70** 042101
- [15] Thorwart M, Eckel J, Mucciolo E R 2005 *Phys. Rev. B* **72** 235320
- [16] Nalbach P, Thorwart M 2009 *Phys. Rev. Lett.* **103** 220401
- [17] Arceci L, Barbarino S, Fazio R, Santoro G E 2017 *Phys. Rev. B* **96** 054301
- [18] Schollwöck U 2011 *Ann. Phys.* **326** 96
- [19] Orús R 2014 *Ann. Phys.* **349** 117
- [20] Strathearn A, Kirton P, Kilda D, Keeling J, Lovett B W 2018 *Nat. Commun.* **9** 3322
- [21] Keldysh L V 1965 *Sov. Phys. JETP* **20** 1018
- [22] Lifshitz E M, Pitaevskii L P 1981 *Course of Theoretical Physics Volume 10: Physical Kinetics* (Elsevier)
- [23] Kamenev A, Levchenko A 2009 *Adv. Phys.* **58** 197
- [24] Wang J S, Agarwalla B K, Li H, Thingna J 2014 *Front. Phys.* **9** 673
- [25] Chen R 2023 *New J. Phys.* **25** 033035
- [26] Feynman R P, Vernon F L 1963 *Ann. Phys.* **24** 118
- [27] Grabert H, Schramm P, Ingold G L 1988 *Phys. Rep.* **168** 115
- [28] Negele J W, Orland H 1998 *Quantum Many-Particle Systems* (Colorado: Westview Press)
- [29] Trotter H F 1959 *Proc. Amer. Math. Soc.* **10** 545
- [30] Suzuki M 1976 *Commun. Math. Phys.* **51** 183
- [31] Makarov D E, Makri N 1993 *Phys. Rev. A* **48** 3626
- [32] Makri N, Makarov D E 1995 *J. Chem. Phys.* **102** 4600
- [33] Dattani N S, Pollock F A, Wilkins D M 2012 *Quantum Phys. Lett.* **1** 35
- [34] Chen R, Xu X 2023 *Eur. Phys. J. Plus* **138** 194
- [35] Verstraete F, García-Ripoll J J, Cirac J I 2004 *Phys. Rev. Lett.* **93** 207204
- [36] Zwolak M, Vidal G 2004 *Phys. Rev. Lett.* **93** 207205
- [37] Barthel T, Schollwöck U, White S R 2009 *Phys. Rev. B* **79** 245101
- [38] Schollwöck U 2005 *Rev. Mod. Phys.* **77** 259
- [39] Levin M, Nave C P 2007 *Phys. Rev. Lett.* **99** 120601
- [40] Evenbly G, Vidal G 2015 *Phys. Rev. Lett.* **115** 180405
- [41] Ran S J, Li W, Xi B, Zhang Z, Su G 2012 *Phys. Rev. B* **86** 134429
- [42] Czarnik P, Cincio L, Dziarmaga J 2012 *Phys. Rev. B* **86** 245101
- [43] Kshetrimayum A, Weimer H, Orús R 2017 *Nat. Commun.* **8** 1291
- [44] Fishman M T, White S R 2015 *Phys. Rev. B* **92** 075132
- [45] Dorda A, Ganahl M, Evertz H G, Linden W, Arrigoni E 2015 *Phys. Rev. B* **92** 125145
- [46] Ganahl M, Aichhorn M, Evertz H G, Thunström P, Held K, Verstraete F 2015 *Phys. Rev. B* **92** 155132
- [47] Wolf F A, Go A, McCulloch I P, Millis A J, Schollwöck U

- 2015 *Phys. Rev. X* **5** 041032
- [48] Linden N O, Zingl M, Hubig C, Parcollet O, Schollwöck U 2020 *Phys. Rev. B* **101** 041101
- [49] Kohn L, Santoro G E 2021 *Phys. Rev. B* **104** 014303
- [50] Jørgensen M R, Pollock F A 2019 *Phys. Rev. Lett.* **123** 240602
- [51] Carmichael H J 1999 *Statistical Methods in Quantum Optics 1: Master Equations and Fokker-Planck Equations* (Berlin: Springer-Verlag)
- [52] Gardiner C W, Zoller P 2004 *Quantum Noise: A Handbook of Markovian and Non-Markovian Quantum Stochastic Methods with Applications to Quantum Optics* (Berlin: Springer-Verlag)
- [53] Jørgensen M R, Pollock F A 2020 *Phys. Rev. A* **102** 052206
- [54] Cerrillo J, Cao J 2014 *Phys. Rev. Lett.* **112** 110401
- [55] Popovic M, Mitchison M T, Strathearn A, Lovett B W, Goold J, Eastham P R 2021 *PRX Quantum* **2** 020338
- [56] Fux G E, Butler E P, Eastham P R, Lovett B W, Keeling J 2021 *Phys. Rev. Lett.* **126** 200401
- [57] Fux G E, Kilda D, Lovett B W, Keeling J 2022 10.48550/arXiv.2201.05529 [quant-ph]
- [58] Gribben D, Strathearn A, Fux G E, Kirton P, Lovett B W 2021 *Quantum* **6** 847
- [59] Gribben D, Rouse D M, Iles-Smith J, Strathearn A, Maguire H, Kirton P, Nazir A, Gauger E M, Lovett B W 2022 *PRX Quantum* **3** 010321
- [60] Ye E, Chan G K L 2021 *J. Chem. Phys.* **155** 044104
- [61] Chiu Y F, Strathearn A, Keeling J 2022 *Phys. Rev. A* **106** 012204
- [62] Otterpohl F, Nalbach P, Thorwart M 2022 *Phys. Rev. Lett.* **129** 120406
- [63] Bose A 2022 *Phys. Rev. B* **105** 024309
- [64] Florens S, Venturelli D, Narayanan R (edited by Chandra A K, Das A, Chakrabarti B K) 2010 *Quantum Quenching, Annealing and Computation* (Berlin, Heidelberg: Springer) p145
- [65] Hur K L (edited by Carr L D) 2010 *Understanding Quantum Phase Transitions* (New York: CRC Press) p217
- [66] Löfstedt R, Coppersmith S N 1994 *Phys. Rev. Lett.* **72** 1947
- [67] Gammaitoni L, Hänggi P, Jung P, Marchesoni F 1998 *Rev. Mod. Phys.* **70** 223
- [68] Grifoni M, Hänggi P 1998 *Phys. Rep.* **304** 229
- [69] Meir Y, Wingreen N S 1992 *Phys. Rev. Lett.* **68** 2512
- [70] Brandes T, Aguado R, Platero G 2004 *Phys. Rev. B* **69** 205326
- [71] Wang J S 2007 *Phys. Rev. Lett.* **99** 160601
- [72] Segal D 2013 *Phys. Rev. B* **87** 195436
- [73] Lü J T, Christensen R B, Wang J S, Hedegård P, Brandbyge M 2015 *Phys. Rev. Lett.* **114** 096801
- [74] Ludovico M F, Lim J S, Moskalets M, Arrachea L, Sánchez D 2014 *Phys. Rev. B* **89** 161306
- [75] Ludovico M, Arrachea L, Moskalets M, Sánchez D 2016 *Entropy* **18** 419
- [76] Ludovico M F, Moskalets M, Sánchez D, Arrachea L 2016 *Phys. Rev. B* **94** 035436
- [77] Ludovico M F, Arrachea L, Moskalets M, Sánchez D 2018 *Phys. Rev. B* **97** 041416

REVIEW

Time-evolving matrix product operator method and its applications in open quantum system^{*}Chen Ruo-Fan[†]

(Center for Computational Sciences, College of Physics and Electronic Engineering, Sichuan Normal University, Chengdu 610068, China)

(Received 27 November 2022; revised manuscript received 4 April 2023)

Abstract

Open quantum systems play an important role in developing quantum sciences, and therefore the study of corresponding numerical method is of great significance. For the open quantum systems, the quasi-adiabatic propagator path integral invented in 1990s is one of the few numerically exact methods. However, its computational complexity scales exponentially with system size and correlation length, and therefore its application is limited in practical calculation. In recent years, the study and application of tensor network have made rapid progress. Representing the path integral by tensor network makes the computational complexity increase polynomially, thus greatly improving the computational efficiency. Such a new method is called time-evolving matrix product operator. At the very beginning, the reduced density matrix is represented as a matrix product state. Then the time evolution of the system can be achieved by iteratively applying matrix product operators to the matrix product state. The iterative process is amenable to the standard matrix product states compression algorithm, which keeps the computational cost on a polynomial scale. The time-evolving matrix product operator is an efficient, numerically exact and fully non-Markovian method, which has a broad application prospect in the study of quantum open systems. For instance, it is already used in the study of the thermalization, heat statistic, heat transfer and optimal control of the quantum open systems, and conversely it can be also used to investigate the effect of the system on the environment. In addition, the TEMPO method is naturally related to the process tensor, and can be used to calculate the correlation function of the system efficiently. In this article we review this method and its applications. We give a brief introduction of the path integral formalism of Caldeira-Leggett model. According to the path integral formalism, we demonstrate the usage of quasi-adiabatic propagator path integral method. we give the basic idea of matrix product states, and we show how to recast quasi-adiabatic propagator path integral method into time-evolving matrix product operators method by employing the concept of matrix product states and matrix product operators, and give a review of its applications. In addition, we use the calculation results of physical quantities, correlation functions and heat currents in the spin-boson model to illustrate the applications of the time-evolving matrix product operator method.

Keywords: quantum open system, quasi-adiabatic propagator path integral, time-evolving matrix product operators, tensor network, non-Markovian dynamics

PACS: 02.70.-c, 03.65.Yz, 05.10.-a, 68.65.Hb

DOI: 10.7498/aps.72.20222267

^{*} Project supported by the National Natural Science Foundation of China (Grant No. 12104328).

[†] Corresponding author. E-mail: physcrf@sicnu.edu.cn