arXiv:2603.02127v1 [quant-ph] 2 Mar 2026

用于格子玻尔兹曼方法的量子算法及其在真实量子设备上的应用

ANTONIO BASTIDA-ZAMORA*、LJUBOMIR BUDINSKI*¬、OSKARI KERPPO*、VALTTERI LAHTINEN*†、OSSI NIEMIM ¨AKI*、WILLIAM STEADMAN*、ROBERTO ZAMORA-ZAMORA*、PIERRE SAGAUT¶、VLADYSLAV BOHUN‡、MACIEJ KOCH-JANUSZ‡和ILLIA LUKIN‡§

摘要。我们介绍一种基于单步简化格子玻尔兹曼方法的新型格子玻尔兹曼方法（LBM）量子算法。与早期的格子玻尔兹曼方法量子算法相比，该算法的结构使其在模拟不同物理现象时更加灵活，同时保持了量子门和量子比特复杂度方面的计算效率。该新算法尤其有望实现线性问题的端到端量子效用。我们讨论了线性声学实例的实现，以及一个在混合仿真循环中通过IBM量子处理器求解的非线性纳维-斯托克斯问题。

1. 引言

计算流体力学（CFD）已成为工程领域日常使用的普遍工具，几乎涵盖所有工业领域，同时也应用于生命科学、地球科学和天体物理学，甚至还有一些较为特殊的扩展用途，例如模拟人群等多粒子系统以及创新的扩散。自洛斯阿拉莫斯在曼哈顿计划期间开展开创性研究以来， [8, 94, 122, 123], 人们提出了大量方法，用于离散化并求解连续的控制方程（通常是欧拉方程和纳维-斯托克斯方程）；其中一些最常见的方法包括有限差分法、有限体积法、有限元法和谱方法 [52, 53, 83, 89, 101]。相关数值算法的发展主要由效率驱动，而效率综合体现了稳健性、精度以及充分发挥处理单元特性的能力。时间推进显式方法的兴起便是很好的例证：这类方法采用非常小的时间步，在包含大量网格点（或单元）的计算网格上进行计算，非常适合基于内存较小、计算能力有限的处理器构建的大规模并行计算机。相比之下，用于稳态计算的隐式方法更适合基于单个或少数几个功能强大的处理器的计算机，例如 20 世纪 90 年代的 CRAY计算机。随着集成数千万乃至数亿个图形处理器或异构众核处理器的计算机兴起，前沿计算迈上了一个新台阶 [127]。这使得人们能够对存在湍流的多物理场、全尺寸且细节完备的系统开展日益逼真的模拟。

*Quanscient Oy，Peltokatu 34，33100 坦佩雷，芬兰
†工程科学学院，拉彭兰塔-拉赫蒂理工大学，邮政信箱 20，53851 拉彭兰塔，芬兰

‡Haiqu公司，95 Third Street，旧金山，CA 94103，美国§阿赫泽尔理论物理研究所，乌克兰国家科学院哈尔科夫物理技术研究所，Akademichna 1，61108 哈尔科夫，乌克兰
¶艾克斯-马赛大学、法国国家科学研究中心、马赛中央理工学院、M2P2联合研究单位7360，马赛，法国¬诺维萨德大学技术科学学院，TrgDositejaObradovica 6，诺维萨德，塞尔维亚电子邮箱： valtteri.lahtinen@quanscient.com、ossi.niemimaki@quanscient.com。日期：2026年3月3日。

为了寻找适用于大规模并行计算机的方法，格子玻尔兹曼方法（LBM） [66, 109] 因其数学结构有利于高效并行化而广受欢迎。从根本上说，LBM 可表示为一组耦合的线性平流方程，并辅以局部非线性松弛项。这些方程通常采用非常紧凑的空间模板求解，并结合一种源自平流算子与松弛算子之间 Strang 分裂的显式两步时间积分方法。这两个关键要素最大限度地减少了处理器之间的数据交换，即使使用数千万乃至数亿个处理器，也更容易实现线性加速比。一些里程碑式成果便体现了这一点，例如 [133] 已展示的城市物理学模拟，使用多达 2×10^{12} 个单元，运行于 155×10^6 异构众核 CPU 上。

格子玻尔兹曼方法的另一项非常有趣的特性是其多功能性。虽然最初提出它们是为了改进格子气自动机（一种旨在模拟流体动力学现象的元胞自动机） [50, 51, 79, 90]，但后来证明，至少其中一些方法可以严格地从经典玻尔兹曼方程离散化推导得出 [1, 49, 103]。最后一个但非常重要的步骤是证明，它们可以被视为求解一般偏微分方程 (PDE) 的一种巧妙而高效的方法 [10, 18, 136]，，从而为其在固体力学 [14, 31, 84, 85, 88]，、燃烧 [13, 54, 115, 128, 135]，、传热 [59, 126]，、电动力学 [43, 46, 80]，和量子物理学 [57, 68, 86, 110, 140]，中的应用开辟了道路，也拓展至行人流与交通流 [81, 82, 104]，、人群动力学 [134]，和流行病 [28]等更为特殊的领域。

尽管现代高性能计算以及格子玻尔兹曼方法等方法提供了强大的模拟能力，计算流体力学中的许多问题仍然难以处理，或极其耗费资源。高雷诺数与湍流、复杂几何形状、多相流以及多物理场耦合，即便在一流的计算机集群上，也可能给开展计算流体力学带来难以克服的限制。例如，湍流对雷诺数 Re具有$\mathcal{O}(Re^3)$的时空依赖关系，而完整的飞机模拟所涉及的雷诺数达到 10^8，量级，精确求解大约需要 10^{24} 次浮点运算 [112]。随着摩尔定律因原子尺度限制而在硅芯片上触及物理极限，量子计算正成为计算流体力学等计算密集型领域中一种前景广阔、甚至可能是唯一的替代方案 [11, 32, 96, 117, 132]。

量子计算可以利用纠缠、干涉和叠加等量子原生特性，从而避免普通计算机所面临的资源瓶颈。一种高效的量子算法能够在量子比特线性增加的情况下提供指数级的自由度，同时所需的计算操作远少于相应的经典方法。此类加速有望使显著更复杂的模拟在量子计算机上得以实现。此外，某些量子计算架构固有的紧凑性和更低的功耗需求，有助于使日益增长的计算需求更加可行且更加环保。

已有许多用于求解偏微分方程，尤其是计算流体力学问题的量子算法。其中许多算法以量子线性方程组求解器为基础，或是其变体，例如 HHL [45]或变分量子线性求解器 [17]。其他算法则基于相互作用粒子的涌现非线性动力学，例如格子气自动机 [5, 36, 37, 39, 137]。HHL 基于量子相位估计，这是一种子程序，其计算复杂度随 ϵ 所需精度的提高而按 $\mathcal{O}(1/\epsilon)$ 增长。此外，该方法的复杂度还随算子条件数（最大与最小特征值之比）按 $\mathcal{O}(\kappa^2)$ 增长。因此，HHL 只有在算子的条件数相对于网格点数量呈多对数级增长时才高效

。在离散化偏微分方程的背景下，量子相位估计产生的误差必须低于网格离散误差，而要降低误差，就需要更多的量子比特。随着网格分辨率提高，条件数也会增大，从而给模拟的扩展性带来严重问题 [60]。为了解决计算流体力学中常见的时间步进问题，还需要将时间维度嵌入线性系统，纳入所需的全部时间步，例如采用有限差分表述，并使用时间参数化叠加态 [71]。另一方面，在变分量子算法中，量子计算与经典计算之间的迭代会形成瓶颈，难以进行大规模、长时间的模拟，因为态制备和测量的复杂度会随网格点数量呈指数增长。此外，针对复杂问题的变分量子算法可能因贫瘠高原而产生极深的量子电路 [69]。

用于格子玻尔兹曼方法的量子算法为计算流体力学提供了线性系统求解器之外的另一种选择。一般而言，格子玻尔兹曼方法通过将场演化表述为虚拟粒子的介观概率分布的传播与碰撞，绕开了直接求解线性系统；随后可由此恢复偏微分方程的解。这种结构使格子玻尔兹曼方法特别适合量子算法实现：简化的碰撞过程可局域化，使其彼此独立于格点，而传播则可通过量子游走来实现 [111]。经典格子玻尔兹曼方法往往受制于巨大的内存需求，但这对量子实现而言并不构成真正的问题。因此，人们提出了多种适用范围各异的 LBM 量子算法 [15, 16, 39, 97–100, 125, 130]。

本文提出了一种新的格子玻尔兹曼方法量子算法。该算法基于单步简化格子玻尔兹曼方法（OSSLBM），这是一种采用两级传播过程的格子玻尔兹曼方法变体 [92]。由于传播基于高效的量子基态移位子程序 [87], ，相较于模型在描述不同物理过程方面灵活性的提升，量子电路复杂度的增加可以忽略不计。传播也是该算法中唯一需要使用分布函数的部分；除此之外，算子直接作用于宏观格点场。这尤其简化了碰撞算子和边界算子。虽然 OSSLBM 不具有标准格子玻尔兹曼方法意义上的碰撞过程，但仍存在一个类碰撞算子，可将格点场映射为与局部离散速度相关联的分布函数。具体而言，该算子保留了局域性。

我们将详细讨论 OSSLBM 的量子算法如何为实现真正的端到端量子优势带来机遇。此外，实现不同物理过程的能力带来了广泛的应用，我们将重点介绍其中几个示例。

在第 2 节中，我们简要介绍格子玻尔兹曼方法，尤其关注单步简化模型。第 3 节概述量子算法中的子程序。第 4 节中，我们讨论该方法的适用范围，并对所需的量子门和量子比特资源进行简要复杂度分析，同时讨论跨多个时间步的完整模拟。第 5 节还展示了一些数值示例，包括线性声学模型；第 6 节展示了在 IBM量子计算机上进行的非线性纳维-斯托克斯模拟。最后，我们将在第 7 节简要展望量子格子玻尔兹曼方法的未来。

2. 格子玻尔兹曼方法及量子格子玻尔兹曼方法的相关前期工作

本节简要回顾格子玻尔兹曼方法的基础知识，重点从碰撞-迁移过程出发，推导单步简化方法。具体而言，我们概述如何将一般的碰撞积分近似为巴特纳加尔–格罗斯–克鲁克（BGK）碰撞算子 [9], ，继而简化分布函数的演化，最终将基础的两步过程提炼为涉及更多变量的单步过程。

2.1. 采用 BGK 碰撞的格子玻尔兹曼方法。由于 OSSLBM 基于对 BGK 碰撞模型的改写，为了清晰起见并确保内容自洽，这里将简要回顾相关内容。BGK 是一种重要的简化方法，可用于模拟多种不同的物理现象；更重要的是，对于数值实现而言，它能将碰撞简化为局部算子。不过请注意，BGK 并不适用于所有模型，可能需要使用更一般的算子 [66,第 10] 章。

构建过程始于基于格子的离散化。首先定义一组离散速度$\boldsymbol{c}_\alpha$及其对应的分布函数 f_α，其中 $\alpha \in \{\mathbf{0}, \dots, m-1\}$。将其与 BGK 模型结合，即可得到离散速度玻尔兹曼方程：

$$\frac{\partial f_\alpha}{\partial t} + \boldsymbol{c}_\alpha \cdot \nabla f_\alpha = -\frac{1}{\tau}\left(f_\alpha(\boldsymbol{x},t) - f_\alpha^{eq}(\boldsymbol{x},t)\right), \tag{1}$$

对于 DnQm格子，该方程可写为：

$$f_\alpha(\boldsymbol{x} + \boldsymbol{c}_\alpha \Delta t, t + \Delta t) = f_\alpha(\boldsymbol{x},t) + \frac{\Delta t}{\tau}\left(f_\alpha(\boldsymbol{x},t) - f_\alpha^{eq}(\boldsymbol{x},t)\right). \tag{2}$$

右端项是一般碰撞积分的 BGK 线性近似，使用平衡分布函数 $f_\alpha^{eq}(\boldsymbol{x},\ t)$、弛豫时间 τ和积分时间步长 Δt。请注意，碰撞表达式完全是局部的。

随后，我们可以按如下方式实现碰撞-迁移过程

$$f_\alpha^{coll}(\boldsymbol{x},t) = f_\alpha^{eq}(\boldsymbol{x},t) + \left(1 - \frac{\Delta t}{\tau}\right) f_\alpha^{neq}(\boldsymbol{x},t) \qquad \text{(collision)} \tag{3}$$

$$f_\alpha(\boldsymbol{x} + \boldsymbol{c}_\alpha \Delta t, t + \Delta t) = f_\alpha^{coll}(\boldsymbol{x},t) \qquad \text{(streaming)} \tag{4}$$

其中，$f_\alpha^{neq}(\boldsymbol{x},\ t)$ 是分布函数 f_α的非平衡部分。这些项的精确表达式取决于碰撞模型。

相应的流体黏度为 $\nu = c_s^2 \Delta t(\tau - 1/2)$，其中 $c_s = c_s^* \Delta x/\Delta t$ 是声速，Δx 为网格尺寸，c_s^* 是内置格子速度。宏观量 V_0 和 $\boldsymbol{V}_1$ （例如，在经典流体力学情形下的密度 ρ 和动量 $\rho\boldsymbol{u}$，见下文）按以下方式重构：

$$\begin{aligned} V_0 &= \sum_\alpha f_\alpha = \sum_\alpha f_\alpha^{eq}, \\ \boldsymbol{V}_1 &= \sum_\alpha \boldsymbol{c}_\alpha f_\alpha = \sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{eq}. \end{aligned} \tag{5}$$

常用格子玻尔兹曼格子（见图1，以及 [129, 第 5],[66, 第 3 章。4] 有关更详细的讨论）主要特征见表1。

平衡函数 $f_\alpha^{eq}(\boldsymbol{x},\ t)$ 表示为宏观变量的显式函数：

$$f_\alpha^{eq}(\boldsymbol{x},t) = \mathcal{F}_\alpha(V_0, \boldsymbol{V_1}) \tag{6}$$

表 1. 格子玻尔兹曼方法中 D2Q9 和 D3Q27 格子的主要特征。两种情况下，格子的特征速度均为 $c_s = 1/\sqrt{3}$ 。

格子	$\boldsymbol{c}_\alpha$	数量	权重 W_α
D2Q9	$(0,\ 0)$	1	$4/9$
	$(\pm1, 0),(0, \pm1)$	4	$1/9$
	$(\pm1,\ \pm1)$	4	$1/36$
D3Q27	$(0, 0,0)$	1	$8/27$
	$(\pm1, 0, 0),\ (0, \pm1,0,\ (0,\ 0, \pm1)$	6	$2/27$
	$(\pm1, \pm1, 0),\ (0, \pm1, \pm1),\ (\pm1,0, \pm1)$	12	$1/54$
	$(\pm1, \pm1, \pm1)$	8	1/216

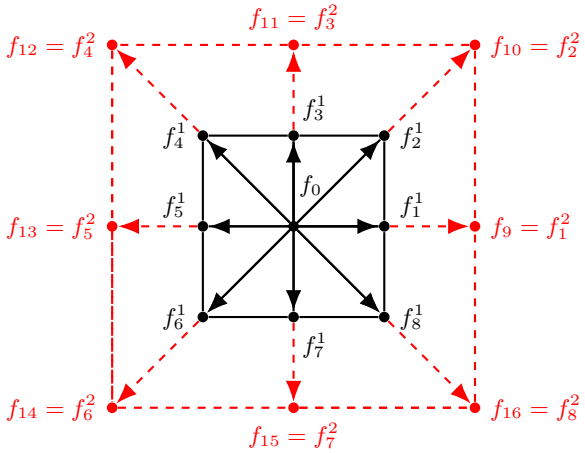

图1. D2Q9 格子（黑线）和 D2Q17格子（添加红线）的配置。对于 OSSLBM 模型，基础配置为 D2Q9，其中包含 f_α^1，另外 8 个速度 f_α^2 则被解释为携带来自第二时间层的数据。

其中 $\mathcal{F}_\alpha$ 包含所研究物理问题的全部信息（流体动力学、声学、传热等），因此该方法具有完全的通用性和灵活性。表 2列出了本文涉及的不同物理问题，以及相应的宏观量和平衡函数。详情见附录A。

2.2. **单步简化格子玻尔兹曼方法。** 由于经典格子玻尔兹曼方法依赖碰撞-迁移方法，因此在处理量子计算时会带来一些编码问题。因此，推导完全量子格子玻尔兹曼算法的一个重要步骤，是对格子玻尔兹曼方法进行重新表述，以便于编码。为此，本文提出研究不依赖碰撞-迁移分步处理的格子玻尔兹曼方法变体，更具体地说，重点关注单步方法。

本文采用的最终方法，即单步简化格子玻尔兹曼方法（OSSLBM），将分两步推导。首先，通过考虑宏观量的演化方程并去除分布函数方程，得到一种简化格子玻尔兹曼方法（见第 2.2.1节）。其次，将这两个步骤合并为一步，从而得到最终方法（见第 2.2.2节）。

2.2.1. 简化格子玻尔兹曼方法。设计单步简化格子玻尔兹曼方法的第一步是改用简化格子玻尔兹曼方法。在这里，简化是指用宏观量 V_0 和 $\boldsymbol{V}_1$ 的演化方程替代离散分布函数 f_α 的演化方程。关键

表2. 可能的物理模型及其相关计算变量和对应平衡函数列表。变量 h、 ρ 和 $\mathbf{u}$ 分别表示水深、密度和速度（浅水近似情况下为深度平均速度），而 g 表示重力加速度。下标0表示基流物理量，它们是模型的输入。权重 W_α 和特征格子速度 c_s 见表1。

Case	$(V_0, \boldsymbol{V_1})$	$\mathcal{F}_\alpha(V_0, \boldsymbol{V_1})$
Low-Mach athermal flows	$(\rho, \rho\boldsymbol{u})$	$W_\alpha \left(V_0 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)^2}{2V_0 c_s^4} - \frac{\boldsymbol{V}_1 \cdot \boldsymbol{V}_1}{2V_0 c_s^2} \right)$
Incompressible athermal flows	$(\rho', \rho_0\boldsymbol{u})$	$W_\alpha \left(V_0 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)^2}{2\rho_0 c_s^4} - \frac{\boldsymbol{V}_1 \cdot \boldsymbol{V}_1}{2\rho_0 c_s^2} \right)$
Linear acoustics	$(\rho', \rho_0\boldsymbol{u}')$	$W_\alpha \left(V_0 C_0 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u}_0)(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)}{c_s^4} - \frac{\boldsymbol{u}_0 \cdot \boldsymbol{V}_1}{\rho_0 c_s^2} \right)$ $C_0 = \left(1 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{u}_0}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u}_0)^2}{2c_s^4} - \frac{\boldsymbol{u}_0 \cdot \boldsymbol{u}_0}{c_s^2} \right)$
Shallow water	$(h, h\boldsymbol{u})$	$\mathcal{F}_0 = V_0 - \frac{5gV_0^2}{6c_s^2} - \frac{2(\boldsymbol{V}_1 \cdot \boldsymbol{V}_1)}{3V_0 c_s^2}$ $\mathcal{F}_{\alpha=1,8} = D_\alpha \left(\frac{gV_0^2}{6c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)}{3c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)^2}{2V_0 c_s^4} - \frac{(\boldsymbol{V}_1 \cdot \boldsymbol{V}_1)}{6V_0 c_s^2} \right)$ $D_{\alpha=1,4} = 1,\ D_{\alpha=4,8} = 1/4$
Linearised shallow water	$(h', (h\boldsymbol{u})')$	$\mathcal{F}_0 = V_0 \left(1 - \frac{5gh_0}{c_s^2} + \frac{2(\boldsymbol{u}_0 \cdot \boldsymbol{u}_0)}{c_s^2} \right) - \frac{4(\boldsymbol{u}_0 \cdot \boldsymbol{V}_1)}{c_s^2}$ $\mathcal{F}_{\alpha=1,8} = D_\alpha \left(V_0 C_0 + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)}{3c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u}_0)(\boldsymbol{c}_\alpha \cdot \boldsymbol{V}_1)}{c_s^4} + \frac{(\boldsymbol{u}_0 \cdot \boldsymbol{V}_1)}{3c_s^2} \right)$ $C_0 = \frac{gh_0}{3c_s^2} - \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u}_0)^2}{2c_s^4} - \frac{\boldsymbol{u}_0 \cdot \boldsymbol{u}_0}{6c_s^2}$ $D_{\alpha=1,4} = 1,\ D_{\alpha=4,8} = 1/4$

推导简化格子玻尔兹曼方法的步骤如下。首先从.....中的格子玻尔兹曼-BGK模型出发，

方程2，其中 c 并可恢复以下两个宏观量的连续演化方程 s:

$$\frac{\partial V_0}{\partial t} + \nabla \cdot \left(\sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{eq} \right) = 0, \tag{7}$$

$$\frac{\partial \boldsymbol{V}_1}{\partial t} + \nabla \cdot \Pi = 0, \quad \Pi_{ij} = \sum_\alpha \boldsymbol{c}_{\alpha,i} \boldsymbol{c}_{\alpha,j} \left(f_\alpha^{eq} + \left(1 - \frac{1}{2\tau} \right) f_\alpha^{neq} \right). \tag{8}$$

该推导涉及一项冗长的代数运算，需要对零阶和一阶矩进行泰勒级数展开。

非平衡部分为

$$f_\alpha^{neq} = f_\alpha - f_\alpha^{eq} = -\tau \Delta t \left(\frac{\partial}{\partial t} + \boldsymbol{c}_\alpha \cdot \nabla \right) f_\alpha^{eq} = -\tau \Delta t D_\alpha f_\alpha^{eq}, \tag{9}$$

其中，平流算子 D_α 定义为

$$D_\alpha = \left(\frac{\partial}{\partial t} + \boldsymbol{c}_\alpha \cdot \nabla \right). \tag{10}$$

现在，方程（8）可在主导阶展开为

$$D_\alpha f_\alpha^{eq} = -\left(1 - \frac{1}{2\tau} \right) \nabla^2 f_\alpha^{neq} - \frac{1}{\tau \Delta t} f_\alpha^{neq}, \tag{11}$$

可将其改写为

$$f_\alpha^{neq} = -\tau \Delta t \left(D_\alpha f_\alpha^{eq} + \Delta t \left(\frac{1}{2} - \tau \right) D_\alpha^2 f_\alpha^{eq} \right), \tag{12}$$

需要注意的是，在没有外力的情况下，有

$$\sum_\alpha f_\alpha^{neq} = 0, \sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{neq} = 0. \tag{13}$$

这样，方程（10）的右端便可仅表示为 f_α^{eq}的函数，因此无需再求解 f_α 的演化方程，因为平衡函数仅是宏观变量的函数（见表 2）。简化格子玻尔兹曼方法最常见的形式是预测-校正格式，例如参见 [19–24]。

2.2.2. 单步 *SLBM*——合并两个步骤。 作为简化格子玻尔兹曼方法的进一步发展，Qin 等人提出通过将两步预测-校正过程改为单步方法来进一步简化该方法，从而得到一种非常高效的算法 [91–93, 105]。其做法是计算方程（12）的零阶矩和一阶矩，得到：

$$\sum_\alpha D_\alpha f_\alpha^{eq} + \Delta t \left(\frac{1}{2} - \tau \right) \sum_\alpha D_\alpha^2 f_\alpha^{eq} = 0, \tag{14}$$

and

$$\sum_\alpha \boldsymbol{c}_\alpha D_\alpha f_\alpha^{eq} + \Delta t \left(\frac{1}{2} - \tau \right) \sum_\alpha \boldsymbol{c}_\alpha D_\alpha^2 f_\alpha^{eq} = 0. \tag{15}$$

现在利用以下两个二阶泰勒级数展开

$$f_\alpha^{eq}(x \mp \boldsymbol{c}_\alpha \Delta t, t - \Delta t) = f_\alpha^{eq}(x,t) \mp \Delta t D_\alpha f_\alpha^{eq} + \frac{1}{2} \Delta t^2 D_\alpha^2 f_\alpha^{eq} + O(\Delta^3), \tag{16}$$

and

$$f_\alpha^{eq}(\boldsymbol{x} \mp 2\boldsymbol{c}_\alpha \Delta t, t - 2\Delta t) = f_\alpha^{eq}(\boldsymbol{x}, t) \mp 2\Delta t D_\alpha f_\alpha^{eq} + 2\Delta t^2 D_\alpha^2 f_\alpha^{eq} + O(\Delta^3), \tag{17}$$

作者推导出一些近似式（16）和（17）的线性组合，可改写为

$$V_0(\boldsymbol{x}, t) = \frac{\tau - 1}{2 - \tau} \sum_\alpha f_\alpha^{eq}(\boldsymbol{x} - 2\boldsymbol{c}_\alpha \Delta t, t - 2\Delta t) + \frac{3 - 2\tau}{2 - \tau} \sum_\alpha f_\alpha^{eq}(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t - \Delta t), \tag{18}$$

$$\boldsymbol{V}_1(\boldsymbol{x}, t) = \frac{\tau - 1}{2 - \tau} \sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{eq}(\boldsymbol{x} - 2\boldsymbol{c}_\alpha \Delta t, t - 2\Delta t) + \frac{3 - 2\tau}{2 - \tau} \sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{eq}(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t - \Delta t). \tag{19}$$

该方案采用类 D2Q17 格子（图1），其含义是将节点解释为处于两个不同的时间层：（$t - \Delta t$）与 f_α^1 相对应，而（$t - 2\Delta t$）与 f_α^2相对应。我们将以此为基础，开发用于格子玻尔兹曼方法的量子算法，其中所谓的 Tau1 方法是一个特例。

OSSLBM 模型的主要优势在于，它允许采用超越 Tau1 简化方案的参数化。在量子系统中，增加第二时间层带来的计算开销相对较小，因为算子可以构建为在叠加态下作用于两个时间层。此外，由于算法的主要部分直接处理宏观场变量，而非分布函数，因此处理边界条件等问题更加容易、直接。

文献 [91] 表明，该方法达到二阶精度，比先前单时间步存储需求的方案更准确（见 [29, 30]），且比 [73]等更复杂的方案简单。

2.2.3. *Tau1 方法。* Tau1 方法 [3, 78, 142] 是一种早已为人所知的方法，也称为“无碰撞格子玻尔兹曼方法”、“宏观格子玻尔兹曼方法”或“单步 SLBM”。令 $\tau = 1$ 在式（1）中变为

$$\frac{\partial f_\alpha}{\partial t} + \boldsymbol{c}_\alpha \cdot \nabla f_\alpha = -\left(f_\alpha(\boldsymbol{x}, t) - f_\alpha^{eq}(\boldsymbol{x}, t)\right) \tag{20}$$

而式（2）中的格子玻尔兹曼方法则退化为

$$f_\alpha(\boldsymbol{x} + \boldsymbol{c}_\alpha \Delta t, t + \Delta t) = f_\alpha^{eq}(\boldsymbol{x}, t), \tag{21}$$

这表明，经典格子玻尔兹曼方法可简化为纯迁移步骤，相当于对平流算子 D_i 采用空间和时间均为一阶精度的显式欧拉法或一阶迎风格式离散化。

这种离散化会隐式引入一种耗散，其大小是 Δt、 $\nu = c_s^2 \Delta t/2$的函数，这解释了为何可以通过选择 Δx （因为 Δt 在格子玻尔兹曼方法中受 Δx 支配）来调节有效黏度。文献 [142] 提出了一种通过改变离散速度的大小来显式控制黏度的方法。作者没有采用分量为 0 或 ± 1的离散速度，而是考虑了分量为 $\{0, \pm e\}$的

$$e = 6\nu/\Delta x, c_s = e \Leftrightarrow \nu = e\Delta x/6 = e^2 \Delta t/6,$$

其中， ν 是需要设定的流体黏度。随后调整时间步，使得 $\Delta t = \Delta x/e = \Delta x^2/6\nu$，从而无需对数据进行插值。

恢复对黏度的显式控制的另一种方法，是使用格子动力学格式及其后续格式 [56, 113, 114], ，这些格式可视为 Tau1 方法的线性修改版，引入了一个能够显式控制黏度的参数。为便于量子算法的开发，我们没有采用这类格式。原因在于，将其与格子玻尔兹曼方法结合需要额外执行几个步骤，因为它还会修改 c_s （进而修改马赫数）的值，从而导致与格子玻尔兹曼方法不一致。

取式（21）的零阶和一阶矩，可得

$$V_0(\boldsymbol{x},t)=\sum_\alpha f_\alpha^{eq}(\boldsymbol{x}-\boldsymbol{c}_\alpha\Delta t,t-\Delta t), \tag{22}$$

$$\boldsymbol{V}_1(\boldsymbol{x},t)=\sum_\alpha \boldsymbol{c}_\alpha f_\alpha^{eq}(\boldsymbol{x}-\boldsymbol{c}_\alpha\Delta t,t-\Delta t). \tag{23}$$

还可以验证，在 OSSLBM 中取 $\tau=1$ 可得到这些表达式，从而确保这三种方法在整体上的一致性。

通过考虑沿离散速度 $\boldsymbol{c}_\alpha$ 对应特征线的以下二阶精度时空外推，可以直接从 OSSLBM 的二阶形式中得到 Tau1 方法：

$$f_\alpha^{eq}(\boldsymbol{x}-2\boldsymbol{c}_\alpha\Delta t,t-2\Delta t)=2f_\alpha^{eq}(\boldsymbol{x}-\boldsymbol{c}_\alpha\Delta t,t-\Delta t)-f_\alpha^{eq}(\boldsymbol{x},t) \tag{24}$$

值得注意的是，许多研究人员在推导 LBM 量子算法时采用 Tau1 方法，因为该方法可以大幅简化式（3）中的碰撞项 [15, 16, 64, 118, 125]。

2.3. **边界条件**。标准格子玻尔兹曼方法与 OSSLBM 的一个主要区别在于，后者直接处理宏观变量。因此，边界条件定义在宏观密度和动量项上，与基于纳维-斯托克斯方程的常用方法相同，无需考虑平衡函数。因此，可以采用标准的密度和动量狄利克雷条件与诺伊曼条件组合，对入口、出口、无滑移和滑移边界条件进行建模。为便于迁移到量子计算场景，诺伊曼条件采用一阶精度格式实现。

由于采用了 D2Q17型模板，OSSLBM 无法用于靠近区域边界的第一层网格单元。因此，应使用另一种模板更窄的方法替代。在当前实现中，对于无法使用 OSSLBM 的网格单元，采用 Tau1 方法进行替代——这意味着边界条件主要仅施加在第一时间层，并通过额外的再平衡来恢复方程（18–19）的正确尺度。

值得注意的是，结合方程（24），在第一层网格单元中使用 Tau1 方法，可以理解为将 OSSLBM 与外推技术结合起来，以重建位于计算域外部的虚拟节点处缺失的数据。

2.4. **浸入固体的掩模方法**。由于量子实现使复杂几何的实现困难得多，因此建议采用一种非常简单的方法来处理任意浸入固体。

在当前实现中，固体由掩模方法建模，该方法将位于固体内部单元中的所有涨落置为零。该方法属于体积惩罚

方法 [2, 74, 139] 和虚拟区域方法 [41]。更精确地说，它模拟宏观 Navier-Stokes-Brinkmann 方程，在固体内部以无穷大弛豫参数趋向静止介质。该方法已针对定常障碍物和运动障碍物，就应用于纳维-斯托克斯方程的多种数值方法进行了评估，范围从有限差分到谱方法，参见例如 [61, 63, 65]。

3. 格子玻尔兹曼方法的量子算法

为了将 OSSLBM 表述为量子算法，我们希望将代数表达式转换为量子电路和量子门。然而，由于量子计算本身，尤其是量子比特电路系统的特性，这一过程并非如此直接。因此，我们的目标并非定义经典算法与量子算法之间真正的一一对应关系，而是提出一种更抽象的量子例程设计方法，使其整体结构与 OSSLBM 相对应。此处的大多数子程序具有通用性，即不仅适用于 OSSLBM，也适用于一般的 LBM。因此，例如在变量和算子的命名方式上，两者会有明显差异。

本节将分别分析代表 LBM 算法各个部分的量子子程序。我们重点介绍单个时间步的电路模块，并将在第 4 节稍后描述完整的模拟过程。为了提高算法各部分的效率，我们将研究生成相应子电路的不同技术，而不是尝试生成一个涵盖所有部分的单一幺正算子。我们给出的子电路示例经过编写，可直接连接起来，形成代表模拟单个时间步的完整量子电路。

格子构型。在以下描述中，我们研究两种主要的格子玻尔兹曼模型：第2.2.2节所述的完整 OSSLBM，以及 Tau1 简化模型（第2.2.3节）。示例格子构型为具有指定离散速度的二维和三维模型，其定义采用表3中所述的定义。

表 3. 量子算法所用的示例格子构型。

	2D	3D
OSSLBM 两个时间层	D2Q17	D3Q53
Tau1 单时间层	D2Q9	D3Q27

特别要注意，我们始终假设单时间层模型遵循 Tau1 构造。文献中可以找到适用于 D2Q9 和 D3Q27 格子构型的更一般模型，但我们将限定于 Tau1。完整模型中的第二时间层实际上使离散非零速度的数量翻倍，不过从碰撞算子和迁移算子的角度来看，这两组速度的演化彼此独立。

模型可以结合 Tau1 格式和完整格式。具体来说，可以先在一些初始时间步中使用 Tau1 格式来稳定模型，然后切换到完整格式，以提高参数化的精度和灵活性。

3.1. **结构概述**。 量子算法由 OSSLBM 的结构直接决定，可分为四个主要步骤（见图 2）：碰撞、传播、积分和边界条件的实施。请注意，由于创建量子计算程序所需的结构有所不同，这些步骤可能无法与通常认为构成 LBM 模型的部分完全对应。

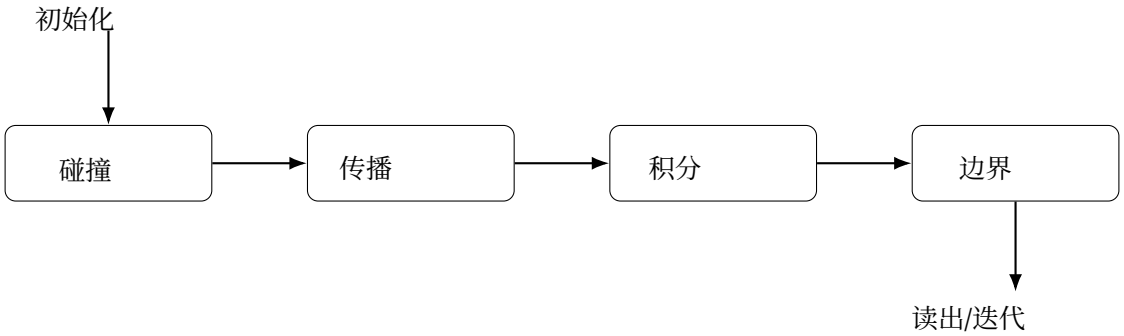

图 2. OSSLBM 量子算法的一般表示。

根据宏观格点场生成初始量子态 ρ 和 $\mathbf{u}$，对应给定的 t_0。我们使用通用术语 密度 和 动量 场来指代 ρ 和$\mathbf{u}$：这些应理解为模型内部的占位符，具体的物理含义则视具体情况而定（通常，这些变量应与第 V_0 节中使用的 $\mathbf{V}_1$ 和2相对应）。假设我们可以构建一个由一对场组成的 LBM 模型，其中包括一个（赝）标量场和一个矢量场；不过，基底也可以更复杂，例如像第 3.3.2节中那样，还包含由这些场产生的非线性项。

格子寄存器上的计算基态用于编码格点位置，而场值则映射为振幅。为了实现高效的编码方法，例如可以采用 [62] 及类似技术；一般而言，编码本身就是一个独立的问题，此处不作深入讨论，只给出几个示例。与一些早期的量子 LBM 架构不同 [15, 16], 振幅不必为正，而且在某些物理模型中，它们也可以是复数。

引入一个类碰撞算子，用于计算平衡分布函数 f_α^{eq}。一般而言，这基于式（6），具体细节取决于实际物理情况。下文中，我们将它们统称为分布函数，并记为 f_α^k，其中 $k \in \{1、2\}$ 表示 OSSLBM 完整模型中使用的特定时间层子集。

传播算子沿局部格子速度对分布数据进行空间和时间上的移位，如图1所示。传播算子通过连接相邻格点，将局部碰撞动力学与全局晶格状态的演化联系起来。

随后，根据式（18–19），通过计算零阶和一阶统计矩，从传播分布中重构宏观场。这是通过对分布函数进行局部积分得到的；由于采用了格子离散化，最终可简化为简单求和。

边界算子沿固体域边界施加外部边界条件。随后，可以使用掩模方法（如第 2.4 节所述）进一步模拟物体浸入区域的情况：我们将其称为物体算子。在图 2 中，边界 算子还包含可能的物体算子，用于施加内部边界条件。

最终态携带着有关新宏观场的信息，可用于启动下一个时间步，或为最终测量做好准备。关于时间步进的更多详情，请参见第 4 节。

为了在更具体的层面上便于阐述，我们引入三个量子寄存器：格子寄存器 q、叠加寄存器 s，以及计算辅助寄存器 a。格子寄存器 q 的大小由格点数目 N 决定：量子比特数为 $n(q) = \log_2 N$。叠加寄存器 s 用于存储格子上的场值，其中量子比特数 $n(s)$ 主要取决于所模拟问题的空间维数：例如，D2Q9 格子至少需要四个量子比特，才能以叠加方式在格子上存储九个分布函数。辅助寄存器 a 的大小取决于电路优化的程度以及问题的维数；详见第4节。一般而言，寄存器 a中的量子比特数可以固定为一个相对较小且与格子大小无关的数目。

我们提出若干新技术，用于寻找实现所需算子的高效量子门序列。具体方法取决于给定算子的结构、问题的维数，以及所采用的 OSSLBM 类型（完整模型或 Tau1）。就效率而言，我们有两个主要目标：以所需量子比特与量子门资源衡量的计算复杂度，以及最小化源于格子玻尔兹曼模型中固有非幺正操作的残余态。减少残余子态的数量与振幅，将显著影响最终测量时的期望概率。更多讨论见第4节。

3.2. **两个时间层的管理。** 格子玻尔兹曼方法的 OSS 模型需要管理对应于两个离散时间层的两组变量。要将场变量从时间 t_i 更新到时间 t_{i+1}，还需要来自时间 t_{i-1} 的场数据。这种双层演化如图 3 所示。主要区别在于分布函数的传播方式：在时间层 t_i，传播步为 ± 1（在格子上），而在时间层 t_{i-1} 传播步为 ± 2。随后对传播分布进行局部积分（在每一层分别进行），再根据式（18–19）将各层组合。

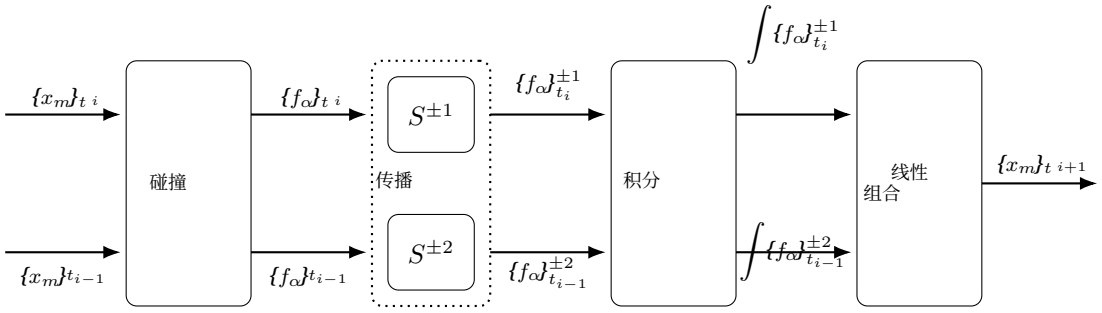

图3。算法中两个时间层的演化。碰撞算子和积分算子并行作用于这两个层（完全叠加），而传播算子则为它们分配不同的移位步长（部分叠加）。这里$\{x_m\}_{t_i}$ 指的是给定时间 t_i 下场变量的通用集合；例如，它们可以是这一对变量（ρ,$\mathbf{u}$）。

跟踪来自不同时间层的变量，需要在某个子态中携带先前计算结果的“副本”，因为没有可用于存储的量子随机存取存储器。 [1] 这一点既体现在碰撞算子中（见式（25）中的矩阵），也体现在对 s 寄存器子态的操作上，以便在计算过程中使这些副本保持完整。

3.3. 碰撞与局部平衡分布。碰撞算子将格子上的宏观场映射为（局部）速度相关分布函数。这模拟了元胞自动机方案中的碰撞，例如格子气自动机：在这类方案中，每个格点上的粒子都被赋予明确的碰撞规则。在标准格子玻尔兹曼方法中，碰撞规则作用于分布函数，并形式化地定义为碰撞积分，用来模拟不同速度分布函数之间的相互作用。这里我们仍将生成传播分布函数的算子称为碰撞，尽管在我们的情形中，该映射并非在分布函数之间进行，而是将格子场映射为分布函数。

该算子用于根据宏观场计算分布函数 f_α^1和 f_α^2，这些宏观场为ρ 和$\mathbf{u}$，其中场数据包含两个时间层。分布函数f_α^k 的方程取决于我们所模拟的物理过程类型。例如，若要求解纳维-斯托克斯方程，碰撞算子由方程（42）构造；而在声波传播的情形下，相应方程则为方程（54）。我们区分线性与非线性模型，并首先关注线性碰撞模型。

线性碰撞算子可以描述为一个矩阵，作用于在给定基上编码宏观场数据的向量。碰撞步的局域性意味着该算子的维度由格子构型固定，且其元素对应于特定的物理过程（分布函数）。若将包含单个格点x 处振幅编码场数据的向量记为$\mathbf{m}$ ，并将平衡矩阵记为 C，则单个格点处的分布函数由 C给出：$\mathbf{m}(x,t) \mapsto \mathbf{f}$，其中矩阵 C具有分块形式

$$C = \begin{bmatrix} W & 0 \\ I & 0 \\ 0 & 0 \end{bmatrix}. \tag{25}$$

此处，分块 W 包含由模型物理与格子构型决定的实数权重系数（与表1相关）。关键在于，这些权重与所选格点无关，因此我们可以将该算子作用于编码格子的态叠加。通常，这些权重也被认为独立于时间 t。

模块 W 和 I 的大小取决于格子模型和问题类型， W 中的行数与离散格子速度的数量相对应。例如，D2Q9 和 D3Q27 模型中， W 的行数分别为 9 和 27。对于线性问题，我们通常使用输入向量对变量 ρ 和$\mathbf{u}$进行编码，其中动量场$\mathbf{u}$ 的分量数取决于空间维数——因此，D2Q9 模型中的 W 和 I 的秩均为三，D3Q27 模型中的秩均为四。单位块的维数与未填充输入向量的维数相匹配

[1]所谓复制，我们并非指复制一个量子态，而是指第二组初始振幅被保留在某个子态中（受全局归一化约束）。例如，量子比特上的振幅数组（q_0，……， q_n）在此意义上可以借助另一个量子比特 q_{n+1}，通过在其上施加H门来“复制”。由此所有振幅都会改变，但可通过重新归一化恢复。

以便在具有两个时间层的模型中，将初始态复制并传递到下一个时间步。

矩阵的量子电路嵌入要求其总维度至少达到容纳所有这些系数所需的最小 2 的幂。例如，对于 D2Q9 模型，$\dim(C) = 16$；对于 D3Q27 模型，$\dim(C) = 32$，其余模块以零填充。如有必要，可以自由调整矩阵各行的顺序。这有助于简化后续传播步的量子电路。

鉴于 C 通常是非酉的，因此需要进行酉嵌入，以便综合量子电路。我们可以将非酉算子分解为酉算子的线性组合。此处采用奇异值分解（SVD）作为具体的分解示例。这样可以将矩阵分解为缩放矩阵和旋转矩阵的乘积，形式为 $C = U\Sigma V\dagger$，其中 U 和 V 是酉旋转矩阵，缩放矩阵 Σ是对角矩阵。现在，嵌入的难点转移到了对角矩阵 Σ上，它通常不是酉矩阵。例如，我们可以进一步将矩阵 Σ 分解为两个酉算子 Σ_1 和 Σ_2 的线性组合，具体如下： [131], 其中每个酉算子的权重因子为 0.5。只需在控制量子比特上使用两个哈达玛门，即可高效地实现该嵌入的量子电路。图 4 展示了矩阵整体嵌入的量子电路实现。

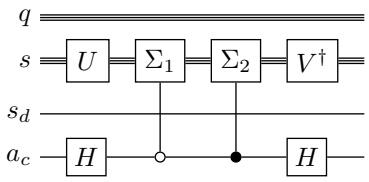

图 4. 使用 SVD 分解碰撞矩阵来实现碰撞步的量子电路示例。寄存器 s 的大小取决于所使用的模型。无论如何，此处标记为 s_d 的寄存器中的最后一个量子比特将用于存储与第二时间层相关的变量。随后，碰撞算子作用于叠加中的两个时间层。

根据矩阵 C的秩，可能可以使用约化SVD矩阵，使其仅作用于寄存器 s的一部分。例如，给定一个仅包含密度 ρ 和动量分量 **u**的输入向量，其秩至多为四。因此，矩阵 U 和 Σ 可以缩减为大小为 4×4的矩阵。不过，算子 $V\dagger$ 始终需要作用于完整寄存器 s。

非酉矩阵的其他分解方式还包括泡利分解过程 [44] 以及 [15, 131]中使用的酉分解技术。不过，由于量子线路简单，SVD 分解是一个特别好的选择。这里需要指出，尽管不同矩阵会在量子门层面产生略有不同的量子线路，但无论模型类型如何——也就是说，无论实现的是完整模型还是 Tau1模型——所述碰撞步过程在本质上都是相同的。碰撞算子仅作用于叠加寄存器，因此，格子上两个时间层的分布函数计算都可由叠加得到。

在 Tau1 模型中，我们只关心时间层 $t - \Delta t$ （见公式xxxxx）。在这种情况下，分布函数 f_α 的计算只需要 $t - \Delta t$ 时间层，我们无需全程携带状态的副本。这意味着矩阵的进一步约化可能

可行：例如，二维模型的算子规模可以减小，从而嵌入本身无需作用于完整的 s 寄存器。[2] 这要求块 W 中的某一行与其他行线性相关，从而可以通过调整编码和矩阵元素来实现这种约化。

3.3.1. 振幅耗散与酉碰撞算子。 方程（25）中的通用碰撞矩阵不是酉矩阵。由于这需要对角 Σ进行酉嵌入，因此预计会出现一些振幅耗散。在嵌入中，算子 Σ_1的构造方法如下：

$$\Sigma_1(k,k) = \sigma_k + j\sqrt{1-\sigma_k^2}, \tag{26}$$

其中， σ_k 是矩阵 C 的奇异值，并按降序排列。结合$\Sigma_2 = \Sigma_1^\dagger$ ，我们可以大致了解耗散情况。奇异值明显偏离最大值 σ_1 ，意味着耗散更强，且耗散量对输入态的敏感性更高。同样，零奇异值也会导致耗散；当矩阵 C 的秩较低时，这可能会成为问题。

由于嵌入误差是相干的，因此可以采用振幅放大方案来减少耗散。本文不对此作进一步分析，而将其留作未来研究中的实现细节。

也有可能构造出满足所需方程的酉矩阵。这取决于这样一个事实：对于给定的模型，振幅向量中的大多数元素都为零，而且格子模型规定的权重恰到好处。特别是，格子速度的选取必须具有对称性，这样才能将整个矩阵正交化。根据表1，一维和二维格子构型是三维模型 [66, 第 3.4 章。7],的投影，这意味着可以进行维度分解。例如，D2Q9 模型中的未约化线性碰撞矩阵的尺寸为 16×16，它作用于一个向量；该向量在每个格点仅包含三个可能非零的标量 $\rho(x)$、 $u_1(x)$ 和$u_2(x)$， x向量中的其余元素均为零。这样便可以为矩阵中的零元素赋值，使算子成为酉算子，同时不改变其对这类状态的作用结果。这样做的显而易见的好处是，可以直接进行无辅助比特电路嵌入，且不会造成振幅耗散。

3.3.2. 非线性碰撞模型。 对于非线性碰撞模型，我们在此考虑一个混合计算循环，其中非线性项在经典预处理过程中构建。随后可以像线性情形那样构造一个类似的矩阵，不同之处在于，该矩阵作用于一个同时编码线性项和非线性项的状态向量。例如，D2Q9 纳维–斯托克斯模型要求输入向量 $\mathbf{m}$ 的维数至少为六（按所包含的场计算），其中各项如下：

$$\{x_m\} = \{\rho, u_1, u_2, u_1^2, u_2^2, u_1u_2\}. \tag{27}$$

利用这些数据，可以构造一个矩阵算子，使其满足式（42）给出的二次展开。

对非线性项进行纯量子实现的探索是一个正在积极研究的课题。值得一提的有趣候选方案要么涉及Carleman线性化 [6] 要么涉及机器学习

[2]仍需对整个 s 寄存器进行一些额外的纠缠操作，以生成完整分布集；我们希望这种纠缠操作的代价低于完整嵌入。

技术 [7, 25, 58, 67]。本研究不会更详细地讨论这些方法；非线性模型的示例结果在第6 节中采用了上文概述的混合方法。

3.4. **分布函数传播**。 遵循格子玻尔兹曼方法所依据的元胞自动机逻辑，碰撞产生的分布函数沿着预定的局部格子速度传播（或流动）。传播将局部动力学扩展到各个格点，从而使全局状态得以演化。例如，图中展示了 D2Q9/Q17 模型的局部速度构型。1，所有格点均适用这一构型。

用于传播的量子程序本质上是一种量子游走，其中方向的叠加由分布函数的集合决定。格子构型决定了格子中链接的数量：下文将以 D2Q9 构型为例进行说明。此外，构造传播序列有多种不同方法。本文介绍一种变体，即分别沿每个格子维度进行传播。尽管这不一定总是最高效的方法，但其结构简单，适合作为示例。由于此步骤的目的并非生成游走分布本身，我们将这些程序称为移位操作 而非游走。有关如何在规则格子上构造移位模式的更多详情，请参见 [87] 。请注意，此处规定的移位操作在构造上是周期性的。如果需要周期性，这可以用作内置边界条件。

对于一个 n 维正交格子，传播方向大致分为两种不同类型。简单的方向只涉及一个维度，因此彼此正交且可分离。我们假设这类方向始终存在。此外，还可能存在组合多个维度的方向，或在格子上进行对角传播的方向。例如，D2Q9 格子包含四个不同的正交方向和四个对角方向，此外还有静止（局部速度为零）方向。正交方向可以分离，相关分布函数可以通过单次移位例程传播；而与对角方向相关的分布函数则需要对所有相关维度同时进行组合移位。

一般方案如下。首先，我们将与给定维度相关的分布函数子态归为一组，并对这些子态执行相应的移位例程。然后，我们根据另一个维度重新归组分布函数子态，并对重新归组后的子态执行该维度对应的移位例程。根据格子构型的需要，重复这一过程。还可以在之前和之后进行额外的重新归组，以便将传播子电路与前面的碰撞以及后续的积分步骤衔接起来。

对于 D2Q9 格子的示例，我们需要两种移位操作：垂直移位和水平移位。根据图 1，与垂直方向相关的分布函数集合为 $\{f_2, f_3, f_4, f_6, f_7, f_8\}$，其中 f_3 和 f_7 仅沿垂直方向。类似地，分布函数集合 $\{f_1, f_2, f_4, f_5, f_6, f_8\}$ 与水平方向相关，而 f_1 和 f_5 仅沿水平方向。随后，我们可以构建传播过程：先对第一组执行垂直移位，再对第二组执行水平移位。一种简单的方法是使用标志量子比特标记需要移位的子态：使用重组算子重新排列子态，使其以正确的顺序与标志量子比特相匹配。将这种 D2Q9 传播扩展到等效的三维 D3Q27 模型十分直接：我们只需为第三个维度再添加一次移位，并相应地匹配重组算子。D2Q9 模型的传播例程如图 5 所示。

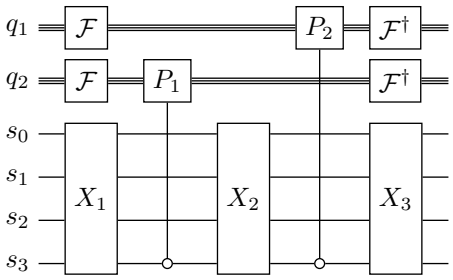

图5. D2Q9传播的一个示例实现，其中格子寄存器被分为 q_1 （水平方向）和 q_2 （垂直方向）。带有$\mathcal{F}$ 的模块表示量子傅里叶变换。我们使用标志量子比特状态 $s_3 = 0$ 来控制这两个移位操作，并用重排算子 X_k 相应地重排分布函数数据。只需额外增加由附加量子比特 s_4控制的相位门模块（P_n），即可将此方案扩展到D2Q17模型；该量子比特用于存储第二时间层的数据 f_α^2 。请注意，第一个重排算子 X_1 至少可以部分地作为先前碰撞步中平衡矩阵 C 的行排序嵌入。

请注意，在完整的 OSSLBM 模型中，该方案在两个时间层上大多以叠加态运行。 f_α^k 这里唯一必要的修改，是确保对应于第二时间层的子态改为移位两倍步长。如果移位例程基于量子傅里叶变换 [102] (QFT)，如图5所示，只需添加一个受控相位门模块，使相关子态的相位平移加倍即可轻松实现。因此，从 Tau1 模型升级到完整双层模型，代价很小。

就门复杂度而言，对于给定的格子构型，重组算子的规模是恒定的，并且不依赖于晶格尺寸。这些算子可以用简单的（可能是多控制的）CX门序列来实现。门复杂度的主要来源是依赖于晶格尺寸的移位算子。在此示例中，我们使用基于量子傅里叶变换的移位；由于量子傅里叶变换例程，其门复杂度相对于格子量子比特数呈二次关系。然而，相位移位门的数量相对于格子量子比特数是线性的。因此，对于每个格子维度，我们需要两个量子傅里叶变换模块（一个量子傅里叶变换模块及其逆变换），每个模块覆盖与该维度相关的量子比特数，以及一组数量与格子量子比特数呈线性关系的相位移位门。第二时间层会使相位移位门的数量加倍，但不需要额外的量子傅里叶变换模块。

可以自然地用其他方法替代基于量子傅里叶变换的移位，例如并行多重受控移位例程 [87]。级联并行移位或许尤其值得关注，因为借助一些辅助量子比特，移位例程可以在线性复杂度下实现（辅助寄存器的大小大致与格子寄存器中最大维度子集的大小相同），也就是说，其复杂度相对于格点数量呈对数级增长。完整模型的双步进在此情形下也是一种简单的扩展：实际上，只需在级联移位中，为标记为第二时间层的子态平衡第一个分解模块即可。与单步进例程相比，这种方法的线性量子门开销相对较低。

3.5. **积分：回到宏观变量**。在传播之后逆转碰撞步，我们现在希望从局部分布函数回到格子上的全局宏观场

。这样，我们就完成了局部碰撞动力学与全局格子状态演化之间的衔接。

实现这一映射的方程很简单（方程（18–19）），其来源是对每个晶格单元内相关分布函数进行局部积分。在离散层面，我们只需将与相关宏观场 ρ 和$\mathbf{u}$对应的分布函数相加。由于积分是局部的，我们可以设计在整个格子上以叠加方式作用的算子。换言之，此处的算子仅作用于寄存器 s，并且可能作用于寄存器 a中的某些辅助量子比特。

我们利用一个简单的观察：哈达玛门作用于基矢的振幅系数时会产生相应结果。首先考虑非常简单的一维模型 D1Q2，其中每个格点上只有两个有贡献的分布函数： f_1 和 f_2。振幅子态可以编码在单个叠加量子比特上，对该量子比特作用一个 H门后，我们得到振幅映射

$$(\alpha, \beta) \mapsto \frac{1}{\sqrt{2}}(\alpha + \beta, \alpha - \beta),$$

其中，我们将 α 和 β 解释为振幅阵列，它们包含相对于格子寄存器的 f_1 和 f_2 数据。暂且不考虑归一化系数，这些振幅现在恰好包含所需宏观密度和动量项 ρ 和 $\mathbf{u}$的数据。

推广到更高维模型相当直接。具体而言，我们只需使用受控哈达玛门算子，就能构造 D2Q9 模型的场分量。事实上，两个受控哈达玛门就足以提取所有相关项，不过得到的是分解形式。若先取寄存器 s 的一个子集，其中分布函数 f_k 按如下方式配对：

$$(f_1, f_5), (f_2, f_6), (f_3, f_7), (f_4, f_8),$$

然后对每一对施加一个以该对为目标的受控 H 门，就会得到可区分的振幅阵列，其中包含

$$(f_1 + f_5, f_1 - f_5), (f_2 + f_6, f_2 - f_6), (f_3 + f_7, f_3 - f_7), (f_4 + f_8, f_4 - f_8).$$

进一步重排后，可得到以下子集

$$\begin{aligned}
&(f_1 + f_5, f_2 + f_6, f_3 + f_7, f_4 + f_8) \\
&(f_1 - f_5), (f_3 - f_7) \\
&(f_2 - f_6, f_4 - f_8),
\end{aligned}$$

这已经显露出所需的结构：第一个子集中包含密度 ρ 的所有分量，但分布函数 f_0除外，它被单独保留。第二个子集给出了动量 $\mathbf{u}$ 的部分分量；对最后一个子集中的一对施加受控 H 门，即可补全这些分量。因此，总的来说，我们可以得到以下子集

$$\rho : (f_0, f_1 + f_5, f_2 + f_6, f_3 + f_7, f_4 + f_8) \tag{28}$$

$$u_1 : (f_1 - f_5, f_2 - f_4 - f_6 + f_8) \tag{29}$$

$$u_2 : (f_3 - f_7, f_2 + f_4 - f_6 - f_8), \tag{30}$$

其中每个元素都可作为寄存器 s 中的一个子态加以区分。

原则上，这些数据已经足够——无需将这些子集合并成最终的场。边界条件同样可以施加在这些叠加态分量上，而碰撞在

下一次迭代也可以相应地调整。这样做的好处是，进一步化简不会导致振幅耗散。不过，需要留意每个H门所伴随的归一化系数。例如，动量项 u_k 在第一个分量上的权重为 $1/\sqrt{2}$ ，在第二个分量上的权重为 $1/2$ 。

尽管如此，有时仍可能需要完成这些求和，例如以便与某种测量操作相匹配。为此可以使用额外的哈达玛门，以及可能的一些重缩放旋转。图6中勾勒了一个示例实现，它既可用于 D2Q9 模型，也可用于 D2Q17模型，同时还包含 D2Q9 模型的一种具体电路实现。

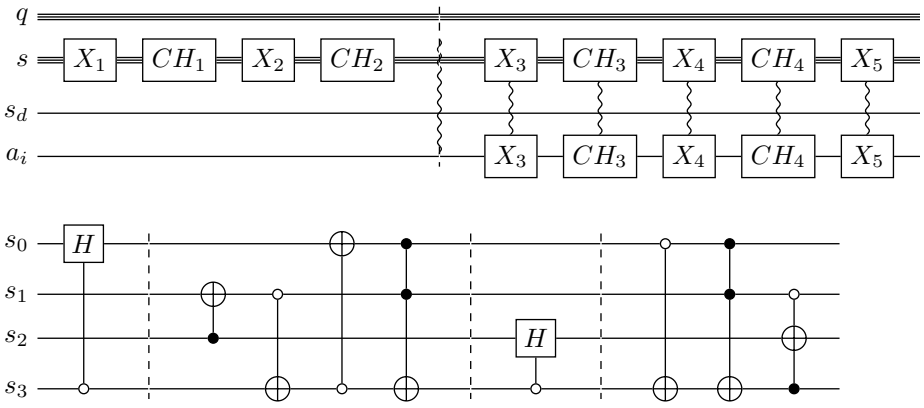

图6. D2Q9/Q17积分的量子电路示意图。上方图中的屏障将保留振幅的部分积分区段，与以振幅耗散为代价完成全积分的部分分隔开来。辅助量子比特 a_i 用于协助构建受控H操作所需的正确子态对，并最终可用于标记由全积分产生的错误子态。子态重组算子 X_k 由一系列多重受控 X 门构成。下方量子电路图更详细地展示了 Tau1（D2Q9）部分积分区段的一种可能实现，并假设寄存器 s 中的子态按（f_1，f_5，f_2，f_6，f_3，f_7，f_4，f_8，0_4，f_0，0_3）排序，其中 0_n 表示 n零振幅模块。

我们使用重组算子 X_k ，以便为求和恰当地排列分布项。与传播中的重组算子一样，我们可以在寄存器 s 上使用一系列多重受控 X 门（也可以在一小组辅助量子比特上使用，以简化子态管理）。在这种情况下，额外的扩散不可避免，因为场现在携带了所需振幅的 $1/4$（尽管进行了预先缩放）。

从 Tau1 量子电路扩展到完整模型很直接。如非活动量子比特 s_d 所示，局部求和会同时作用于叠加中的两个时间层。Tau1 与完整模型的主要区别在于，后者需要管理保存初始数据的额外子态。这可能需要增加一些控制，并使算子 X_k的模块稍微复杂一些。

由于局部积分步骤实际上是碰撞算子的逆过程，因此不依赖于格子的大小。因此，对于给定的格子构型，其计算复杂度可视为常数。量子门复杂度取决于 m 离散速度的数量，因为每增加一个速度，方程（18–19）中的求和项就会增多。这种依赖关系是线性的，重组算子的复杂度为 $\mathcal{O}(\log^2(m))$。与第 6 节中的 D2Q9 情形一样，对于理想（全连接）量子电路，所需的 CX 门数量约为 50。

3.6. 施加边界条件。 边界条件可分为两类：跨越计算域的外部边界，以及勾勒出域内物体／障碍物轮廓的内部边界。本节将讨论区域边界。

OSSLBM 中的边界条件通过宏观场变量实现（见第 2.3 节），这与标准格子玻尔兹曼方法不同，后者需要直接处理分布函数。这意味着，可以通过密度场和动量场的标准狄利克雷条件和诺伊曼条件来施加计算域边界条件。然而，两个时间层要求对两层边界单元进行特殊处理 [92]。此外，传播子电路带有固有的周期性条件，会影响这两层，因此需要加以修正。我们的处理方式是，基本上忽略这些格点上第二时间层的振幅数据，并且只在第一时间层施加相应的边界条件。

为此，我们需要修改格子域边界部分的振幅。这个过程分为三步：将相关振幅数据定位并移至格子上的零振幅子空间，根据规定的条件对其进行修改，最后将修改后的数据移回原位置。

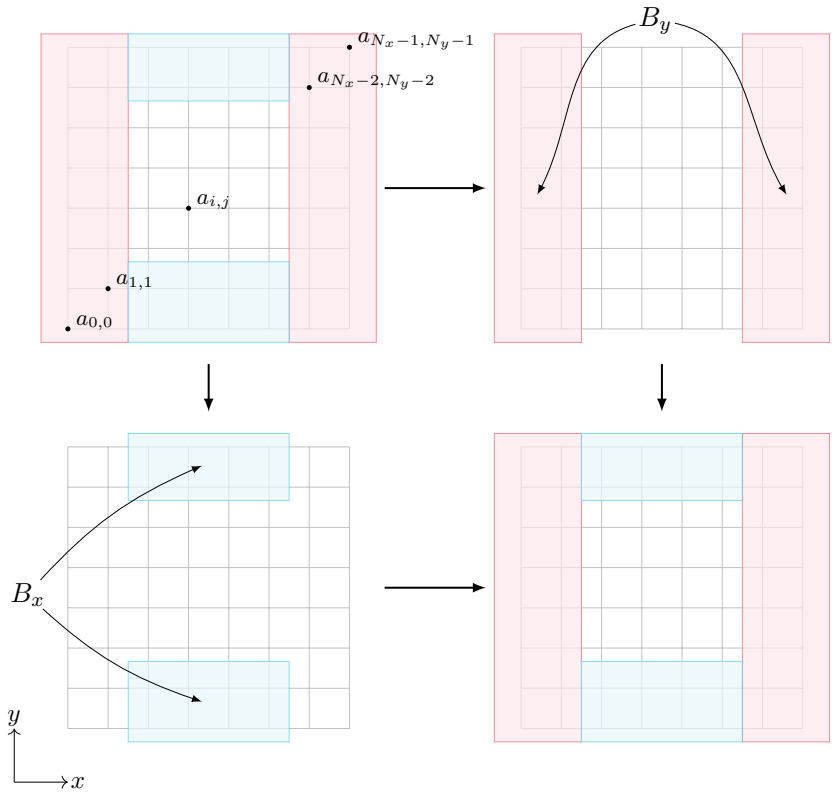

图 7. 将二维数组 a 上的边界数据映射到尺寸为$N_x \times N_y$的晶格网格上。首先，找到与两单元边界层相对应的格点（左上）。随后，将这些层上的振幅数据移入零子空间。算子 B_y 和 B_x（右上和左下）用于对振幅数据施加边界条件。最后，将振幅传回原来的位置（右下）。扩展到三维很简单。

首先，在格子上定位相关数据包括选择与边界相对应的格点，以及选择应用该条件的场变量。由于采用双时间层结构，我们需要选取沿每个固体边界的两层单元中的场数据（图 7 左上部分）。首先，我们将与边界单元相对应的数据移至零振幅子空间，以便于操作。这可以借助辅助寄存器，并使用一系列算子 C_kX 来完成；每个算子都包含与格子维度 k 上的边界线相对应的多重受控 X 门，并作用于格子寄存器的一个子集 q，具体取决于所选的空间编码（见图 8）。

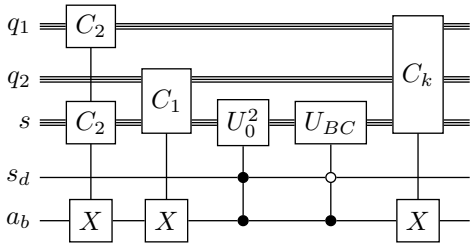

图 8. 用于在二维模型中施加均匀域边界条件的量子电路示意图。模块 C_k 表示与格子维度 k 的边界部分相匹配的控制，针对寄存器s上给定场振幅的两层单元层。量子门 U_0^2 将该边界层上的第二时间层振幅置零，量子门 U_{BC} 则是该条件本身在第一时间层边界振幅上的电路实现。作为最后一步，修改后的振幅借助 $a_b = 0$移回子态。此处我们假设使用单个辅助量子比特 a_b 来修改这些振幅；具体实现可能会使用更多。同样，某些量子门操作可以合并，并且根据边界条件，量子门序列的顺序可能略有不同。

作为第二步，我们施加边界算子 B_k 于已传输的数据上，指标 k 取决于维度。这些算子具体做什么取决于边界条件的类型。例如，在如图 7所示的二维情形中，两列包含两个垂直边界层（红色模块），我们将算子 B_y 施加于振幅子集 a_y（i，j）上，以匹配这些单元。对于零梯度边界条件，这将如下所示：

$$B_y : \begin{cases} a_y(0,j) = a_y(1,j) \\ a_y(N_x-1,j) = a_y(N_x-2,j), \end{cases} \tag{31}$$

也就是说，边界单元内层的数据会被复制到外层。类似地，对于零狄利克雷条件，我们会将外层的数据替换为零。随后，对水平边界层上的算子 B_x 进行类似处理。

最后，我们将两条边界线上的更新后宏观数据传回原始位置，然后对最终宏观变量求和（公式（18、19））），这对应于图7右下部分。为此，我们使用与第一步相同的算子集合 C_kX，并可能添加一些额外的多重受控 X 门，以正确匹配时间层。

3.6.1. 合并两个时间层。 根据式（18，19）对时间层进行的最终求和，原则上可用受控 H 门完成。由此得到最终态，其编码的演化后宏观场数据与区域边界处的边界条件相匹配。

不过，有两个细节需要处理。首先，上述边界处理应仅正确作用于第一时间层；对于第二时间层，需要确保边界数据不与所施加的条件相混合。这一点相对简单，可通过谨慎选择门控控制，并确保第二时间层的边界数据被置零来实现。

另一个细节涉及每个时间层 k 的权重 c_k：

$$c_1 = \frac{3-2\tau}{2-\tau}, \quad c_2 = \frac{\tau-1}{2-\tau}, \tag{32}$$

应用方式如方程（18、19）所示。由于振幅数据始终受到全局归一化的约束，且不存在简单的酉方式来缩放振幅，因此我们需要确保权重是相对于其余场数据施加的。对此有两种（可以同时采用的）处理方式：在构造碰撞矩阵中的系数时使用额外的缩放项，以及在边界电路中使用受控旋转门重新缩放振幅数据。请注意，这种旋转式“缩放”会导致部分振幅耗散。

没有一种方案能够适用于所有情况，因为可用的旋转取决于参数 τ：例如，不可能设计出一个能以系数 $c_k > 1$ 缩放振幅的单一旋转。在这种情况下，必须考虑使用诸如 c_j/c_k 之类的系数进行相对缩放。

3.7. **浸没物体**。为了模拟域中浸没物体的影响，我们可以按照第 2.4 节介绍的掩模方法，对这些格点施加零涨落条件。然而，由于双时间层结构，计算只会影响物体边界附近的两个单元层。因此，只需在这两层上施加零条件；这可能有助于降低门复杂度。

我们在物体子域内部边界单元的两层上各格点处引入多重受控 X 门，并以一个辅助量子比特为目标（第一层为外侧边界层，第二层为内部单元层）。这会将零子空间中的零振幅转移到物体所在位置，其方式与固体边界类似。

辅助量子比特后方仍会累积一些残余振幅，导致场数据在概率意义上发生一定程度的扩散。其影响程度取决于具体应用场景——通常，物体边界的尺寸相较于整个域较小。

4. 量子格子玻尔兹曼方法的适用范围与复杂性

4.1. **线性与非线性模型**。如第 2 节所述，格子玻尔兹曼方法可用于模拟各种物理过程。根据所模拟的物理特性，这些过程可分为两大类：线性动力学和非线性动力学。通常，会对非线性问题进行线性化，以便直接计算并采用成熟的线性技术。另一方面，量子力学的线性基础对我们在量子计算机上高效模拟复杂动力学构成了重大限制。

在格子玻尔兹曼模型中，非线性主要有两个来源。最明显的情形是，从宏观场变量推导平衡分布函数（式（6））的过程具有非线性。另一方面，控制离散方程（式（1））本身也可能具有非线性，例如受到具有非线性性质的周围介质影响时（例如电动力学中的磁性材料）。这里我们关注第一类情形。

在所述量子算法的背景下，线性情形很容易实现：首先定义分布函数 $\{f_\alpha\}$ 以匹配问题。由于该问题是线性的，得到的碰撞矩阵 C 可以直接使用，并按照第 3 节所述将其转化为量子算子。

对式 (6) 中的平衡分布方程进行线性化，是利用 LBM 量子算法处理非线性问题的一种常用方法 [6, 97–99]。例如，利用 Carleman 技术，首先将非线性映射线性化为一个无限的线性方程组，然后截断至特定阶数，并使用线性格子玻尔兹曼模型求解。此外，时间拼接（见下文讨论）在此直接嵌入该过程，因此无需采用某些特定的时间步进方法。然而，尽管该方法能够处理非线性，但截断阶数、碰撞算子的维数以及最终解出现概率较低等问题，使其目前难以用于实际应用。人们也探索了量子机器学习，以纳入平衡分布函数的非线性部分 [58,67]。尽管如此，目前的研究尚无确凿证据表明，这些模型有能力解决非线性部分，尤其是在从平衡分布方程中减去这些高阶项会导致较大相对误差的情况下。

量子-经典混合方法可以被视为解决非线性问题的另一种途径。这里保留了线性碰撞算子的局部结构，但纳入了更多宏观场变量。具体而言，不再仅对一组不同的宏观变量进行编码，而是同时纳入相关的乘积项。这不可避免地带来一项约束：每个时间步之后，都必须从量子系统中提取数据，并重新初始化解，从而限制了算法的效率。尽管如此，我们仍在第 6节中使用这一技术，给出非线性纳维-斯托克斯模型的实现示例。

4.2. 算法复杂度。假设格子构型固定（例如 D2Q17），门复杂度大致分为三部分：常数子程序（碰撞和积分）、仅依赖格子的子程序（传播、区域边界），以及依赖格子和物理模型的子程序（嵌入对象）。所谓常数子程序，是指无论格点数量多少、物理模型多么复杂，其算子的维度都保持不变。另一方面，传播的复杂度完全取决于格点数量；相对于格子量子比特数，其复杂度可能为二次（基于量子傅里叶变换的移位），也可能为线性（采用辅助量子比特的级联移位）。这同样主导了区域边界处理的复杂度，不过，实际的边界条件可能会略微影响这一点。最后，对象嵌入主要取决于嵌入的形状，并间接受晶格尺寸影响。

撇开复杂区域和物体边界可能带来的问题不谈，值得注意的是，核心算法——碰撞、传播和积分——可以被认为是非常高效的。所需量子门的数量取决于传播过程，在最坏情况下，其复杂度为 $\mathcal{O}\left(\log_2^2(N)\right)$ 量子门复杂度。这需要 $_2(N) + c$ 个量子比特，其中 c 是一个相对较小的常数因子，取决于格子构型以及电路优化中的选择；例如，在第 6 节中，我们构建了一个具有 $c = 8$的 D2Q9 模型。

上述内容都避开了态制备和最终测量。我们也将简要讨论这些方面，但需要指出的是，这对于任何数据密集型量子算法而言都是一个普遍问题。不过，我们有充分理由相信，可以利用计算流体力学的特性来缓解这两

个难点，因为在许多应用中，可能并不需要制备或读出非常精确的振幅数据。

4.3. 态制备与测量。 对于每个格点和每个场的振幅都各不相同的一般量子态，状态编码的计算复杂度为 $\mathcal{O}(N)$，其中 N 为格点数。这一步的计算复杂度高度依赖于初始态的纠缠程度。不过，对于纠缠程度适中的初始态，我们可以利用基于矩阵乘积态的方法进行态制备。结合张量交叉插值，这些方法可扩展至非常大的系统。

在 [62] 变分地最小化目标量子态的纠缠熵。根据晶格尺寸，我们既可以精确表示振幅，也可以使用张量交叉插值，按照某种解析分布载入系数。这需要适度的经典训练开销，具体取决于状态的精确表示方式以及变分电路的深度。

我们强调，初始态编码仍是一个活跃的研究领域。不过，已有一些具有合理扩展性的办法，可用于制备相对简单的初始态。许多实际应用自然可以基于这类状态构建，从而能够对更复杂的初始条件进行初始编码，或采用比现有方法更高效的状态制备电路。尤其是，对于编码计算流体力学相关数据的状态，我们通常可以假设其具有额外结构，例如对称性或光滑性，因此无需考虑完全一般的情形。

同样，即使量子比特数量不多，在电路执行结束时进行完整的量子态层析也不可行。与其追求完整的场数据，不如关注如何在整个模拟循环结束时提取有意义的全局信息。例如，在线性声学中，一个自然的选择是声能，它是格子场ρ 和$\mathbf{u}$的二次函数。声能对应一个对角可观测量，只需一种测量设置即可测得。具体而言，估算声能只需对 s 寄存器中的量子比特进行计算基测量。我们将在第 5 节更详细地分析这一步测量。

在第 6 节中，我们利用态制备和态层析（附录 B）中的若干简化方法，模拟一个小规模非线性问题。尽管其中并非所有方法都能扩展到实际场景，但这些方法仍然表明，通过针对具体情况调整通用设置，有望进一步提升计算流体力学中的优势。

4.4. 衔接时间步。由于格子玻尔兹曼方法涉及对含时方程进行离散化，因此需要纳入时间推进格式。在不于每个时间步后执行代价高昂的测量编码操作的情况下，按离散时间步推进计算，是实现任何实际量子优势的必要前提。通常，对变量进行测量并重新编码的成本呈指数级增长，尤其是因为测量需要对状态进行部分层析，以处理负振幅。

因此，我们需要一种合适的技术来连接不同的时间层，其中信息的提取仅在模拟结束时才需要。方法的选择取决于所模拟问题的类型（线性与非线性）。

同样，对于线性问题而言，这在某种程度上是较容易处理的任务。由于我们需要在每个时间步结束时获取宏观变量的信息，因此后续量子例程的应用

以进行下一步通常取决于是否确保了干净的辅助量子比特，以供碰撞和积分步骤中的非酉计算使用。要高效地清理每个时间步结束后的状态，最有前景的方法似乎是电路中测量和动态电路，例如 [125]。通过将残余垃圾态集中到一个标志辅助量子比特上，我们可以先测量标志量子比特，再根据测量结果为 0 有条件地执行下一步，从而推进到下一个时间步（见图 9，其中 a_s 充当标志量子比特）。这样，每个新的时间步都从一个干净的辅助寄存器开始 a，这足以继续计算。模拟完成后，通过后选择过程从系统中提取信息。

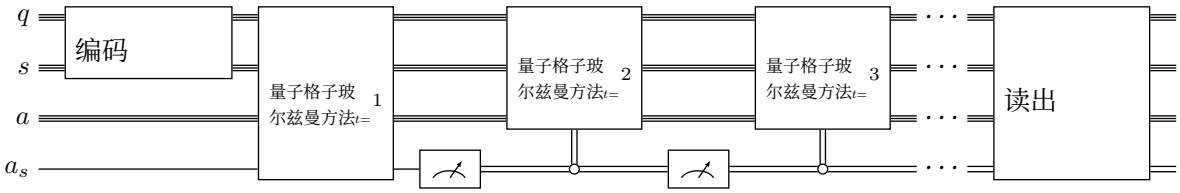

图 9. 时间步进过程的动态量子电路示意图。此处，量子比特a_s 用作时间步进的标志量子比特，所有残余垃圾态都集中在满足 $a_s = 1$的状态上。因此，通过测量 $a_s = 0$，我们可以进入下一步，并重复执行格子玻尔兹曼方法的量子线路。

非线性情形的时间拼接主要取决于所采用的线性化类型。对于 Carleman线性化，碰撞和传播分布函数（以及其他卡尔曼变量）最初编码在不同的时间层上。随后，时间步进便自然地编码到这一过程中，无需进行特殊处理。然而，由于前文提及的问题，Carleman线性化在实际应用中仍遥不可及。

5. 求解二维线性声学

作为第一个示例，我们来研究线性声学。本节将一个简单的状态向量模拟模型与解析解进行比较，然后更详细地讨论如何从测量中提取有意义的信息。具体而言，我们将探讨一种声能表达方式，该方式使我们能够为量子算法设计一个简单的测量方案。

5.1. **高斯脉冲的演化**。

线性声学研究振幅相对较小的声波。我们可以利用流体力学的基本原理对其进行建模，并忽略黏度和热导率的影响。我们可以基于流体动力学方程构建格子玻尔兹曼方法（见附录 A）：

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{33}$$

$$\partial_t \rho \mathbf{u} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p\boldsymbol{I}) = \nu \nabla^2 \mathbf{u}, \tag{34}$$

其中，我们讨论声学密度 ρ、声学质点速度 $\mathbf{u}$ 和声压 $p = c_s^2 \rho$，以及运动黏度 ν 和特征格子速度 c_s。

模型未知量（ρ、$\mathbf{u}$）定义为上述场的扰动分量（为简洁起见，此处使用相同符号表示）。分布函数可按照式（54）定义，据此我们可以为量子电路中的碰撞子程序构造一个线性矩阵 C 。

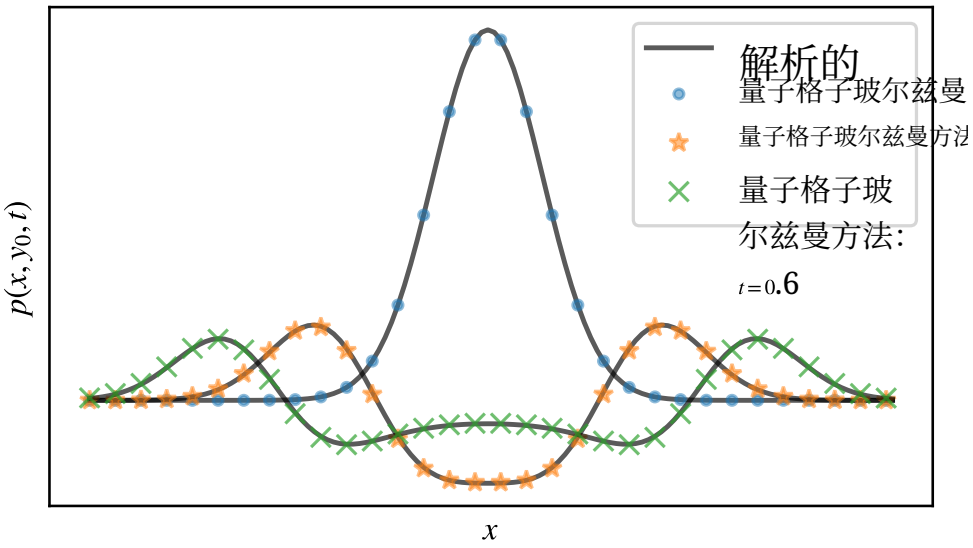

图10. 沿 x方向的声压波动 p 演化，并对量子模拟与解析解进行定性比较。初始条件对应于高斯分布 $p(x,\ y) = e^{-\beta(x^2+y^2)}$。我们得到的平均相对误差为 $\bar{L}_2^{rel} = 0.03$，这意味着在 $t \in [0, 0.8]$的演化过程中，解析解与量子算法之间的总偏差低于3%。

作为一个示例，我们采用二维声学中一个广为人知的解析解：高斯声脉冲演化，如图10 所示，网格包含 128×128 个点。该量子模型共使用 21 个量子比特，其中 14 个用于格子寄存器 q。态矢量模拟的结果与解析解高度吻合。

5.2. **能量测量**。 算法在经过 T 个时间步后的最后一部分是测量。当底层量子态包含 N 个非零振幅时，完整态层析成像的复杂度为$\Omega(N)$。因此，除少数情况外，完整态层析成像并不可行。不过，即使不进行完整态层析成像，我们仍能提取有意义的信息。对于多物理场模拟，我们可以尝试提取能量、场平均值或极值（例如速度或压力）、梯度，或其他更容易获取的数据。我们将说明如何通过简单的测量设置来获取声能。

对于线性声学，一个自然的选择是声能，它是宏观变量的二次函数。声能对应于一个对角可观测量，只需一种测量设置即可测量。具体而言，估算声能只需对 s 寄存器中的量子比特进行计算基测量。

根据 [95]，声能可以表示为

$$E(S) = \frac{1}{2}\sum_{x\in S} c_{phys}^2 \rho^2(x) + (u_1)^2(x) + (u_2)^2(x), \tag{35}$$

其中， $\rho(x)$ 和 $u_i(x)$ 分别表示格点上的波动密度和速度， c_{phys}^2 是与声速相关的换算因子，而 x 是在格子的子域 S 上求和。

模拟结束时的完整量子态包含未清理的辅助量子比特，因此在访问包含宏观变量的部分之前，必须先清理该状态。可以通过测量每个辅助量子比特是否处于零态来完成清理。此外，宏观变量还存在一个范数因子，需要对其进行跟踪。在分析测量设置后，我们会再讨论这一点。目前，我们假设可以访问量子寄存器中包含宏观变量的部分，并且范数因子是预先已知的。

我们将用 $\hat{E}(S)$ 表示对应于域S上声能的算子。该算子有两个非零本征值 $c_{phys}^2/2$ 和 $1/2$ ，分别对应寄存器中的密度部分和速度部分，因此，对于对应于某种宏观变量编码的状态 ψ，$\langle \hat{E}(S)\rangle_\psi = \langle \psi|\hat{E}(S)\psi\rangle$ 等于声能。该可观测量的方差为

$$\Delta_\psi(\hat{E}(S))^2 = \left(\frac{c_{phys}^2}{2} - \langle \hat{E}(S)\rangle_\psi\right)^2 \mathrm{Tr}\left[\rho\right] + \left(\frac{1}{2} - \langle \hat{E}(S)\rangle_\psi\right)^2 \mathrm{Tr}\left[\mathbf{u}\right], \tag{36}$$

其中，$\mathrm{Tr}\left[\rho\right]$ 和 $\mathrm{Tr}\left[\mathbf{u}\right]$ 表示在子域S 上，对状态 ψ 进行计算基测量时，分别得到对应于寄存器中编码密度和速度的部分的测量结果的概率。我们注意到，当 S 等于整个域时，测量可简化为对 s 寄存器进行计算基测量，因此这种测量的实现特别简单。此时，我们将相应的可观测量记为 $\hat{E}$。

对于固定的输入态 ψ，现在我们可以估算需要多少次采样才能估计声能。由于方差有限，我们可以计算标准误差:

$$\sigma_{\langle \hat{E}\rangle_\psi} = \frac{\sqrt{\Delta_\psi(\hat{E})^2}}{\sqrt{n}}, \tag{37}$$

其中 n 等于样本数。要简单估算达到特定标准误差所需的样本数，我们首先要求

$$\sigma_{\langle \hat{E}\rangle_\psi} \leq \epsilon \langle \hat{E}\rangle_\psi, \tag{38}$$

其中 ϵ 表示期望的相对精度。对 n 求解，得到

$$n \geq \frac{\Delta_\psi(\hat{E})^2}{\epsilon^2 \langle \hat{E}\rangle_\psi^2}. \tag{39}$$

为了进一步简化，我们注意到，虽然方差取决于编码态 ψ，但它总是有声能平方作为上界。因此，我们得到所需样本数 $n \geq \epsilon^{-2}$ 的简单上界，以使声能的相对精度达到 ϵ 。

为验证这一缩放关系，我们从一个简单的初始态出发，将其演化若干时间步，并对声学能量测量进行 1000 次模拟。精确的声能为 $\langle \hat{E}\rangle_\psi = 1.9\times10^{-6}$，而方差计算得为 $\Delta_\psi(\hat{E})^2 = 1.9\times10^{-8}$。要达到 $\epsilon = 0.01$ 的相对误差，每次实验包含 3146 个样本。正如预期的那样，各次测量结果服从高斯分布，所有实验的均值与精确能量非常接近。测量结果如图 11 所示。

还有两个额外因素会增加声学能量测量的扩展开销。第一个与被测子域 S 有关。当 S 并非整个域时，我们还需要测量 q 寄存器，以获取能量在各个

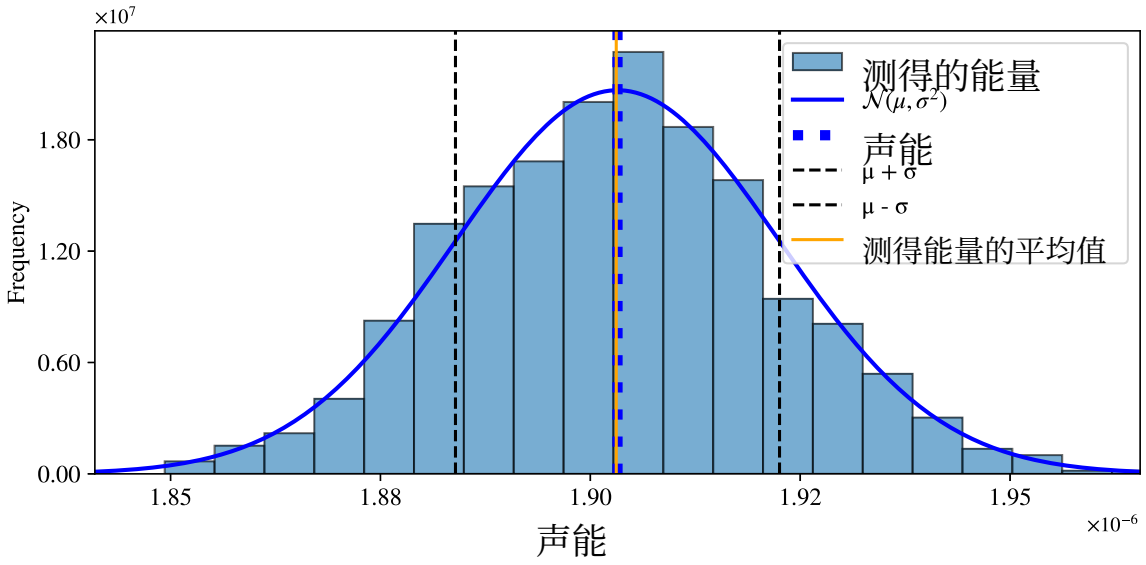

图 11。模拟声学能量测量重复 1000 次所得的直方图。

格点上的分布信息。这当然可行，但我们将重点讨论测量整个域这一更简单的情况。

第二个因子来自对编码态 ψ 范数的估计，因为该状态通常在初始时并未归一化。也就是说， ρ 和 u_i 的宏观值编码在一个次归一化状态中，但每个时间步对辅助量子比特进行测量，都会使状态归一化。嵌入的非幺正操作也会导致状态扩散到辅助空间中，因此必须将其纳入考量。态扩散的幅度取决于初始态和碰撞矩阵的结构。实际上，经过几个时间步后，扩散往往会趋于稳定。通过统计辅助量子比特测得为零的次数，可以估计归一化因子和态扩散。随后，零值与非零值辅助量子比特计数之比可直接给出状态所需的范数因子。利用霍夫丁不等式可以推导出严格的精度界限和置信区间。由于在声学能量测量中会舍弃辅助量子比特测得为非零的样本，因此能够以相对较高的精度估计状态的范数。因此，我们不再进一步分析这一因子，并在后续分析中将其省略。

5.2.1. 示例：能量耗散。 在此示例中，我们将一个高斯脉冲置于 64×64 区域内，而该区域本身又处于一个更大的 128×128 格子内，以便观察能量在初始区域之外耗散的过程。这种设置模拟了开放边界，无需通过任何明确的边界条件来描述它们。初始态如图12所示。

在图13中，我们将能量测量结果与已知的状态向量数据进行比较。这里，我们采用两种不同的测量次数，以展示测量中散粒噪声的影响。由于所有计算均以晶格单位进行，我们使用归一化能量对测量结果进行定性分析——归一化是相对于状态向量数据给出的初始能量进行的。

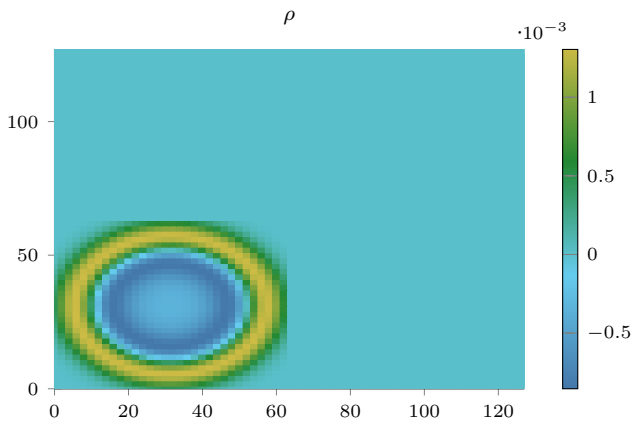

图 12。用于声能耗散测量的初始态。这里，高斯脉冲置于 128×128 格子的左下象限，并在此 64×64 单元上进行测量。

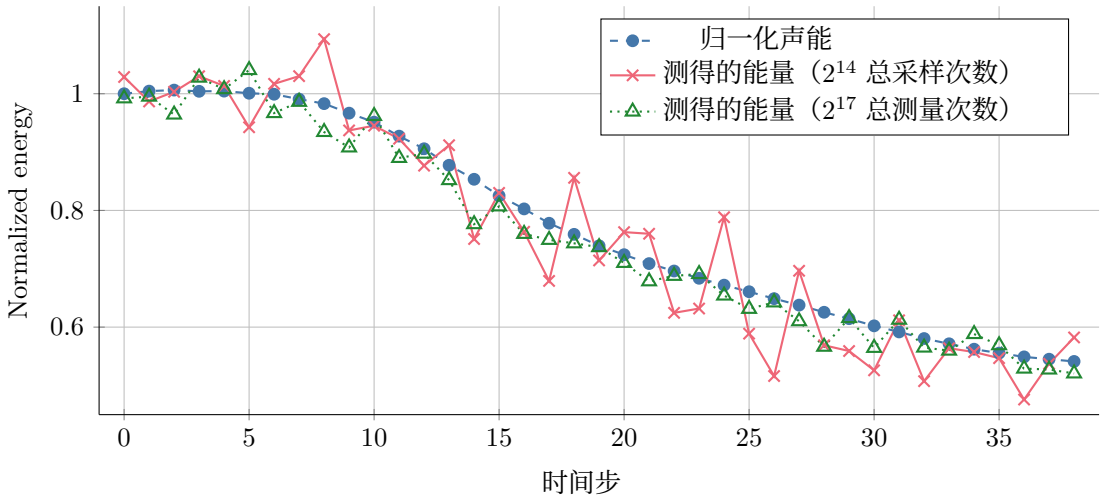

图13。 2^{14} 和 2^{17}总测量次数下测得的声能（归一化）演化与理想状态向量数据的对比。测量次数对准确性的影响清晰可见。

6. 在量子计算机上求解二维非线性翼型问题

本节中，我们验证 OSSLBM量子算法在描述浸没物体周围二维流动的非线性问题上的实际性能。我们在含噪量子处理器上运行量子线路，并将结果与理想态矢量模拟的预期结果进行比较。

为此，我们针对不可压缩纳维–斯托克斯方程定义了一个简单的二维翼型问题，其中计算域内包含一个固体物体。场演化遵循以下方程

$$\partial_{x_i} \rho u_i = 0 \tag{40}$$

$$\partial_t u_i + \partial_{x_j} \left(u_i u_j + P \delta_{ij} \right) = \nu \, \partial^2_{x_i, x_j} u_i, \tag{41}$$

其中， ν 是黏度， u_k 是速度场的分量（抽象模型中的动量$\mathbf{u}$ ）， $P = p/\rho_0$ 是归一化压力， ρ_0 是密度 $\rho = \rho_0 + \rho'$的参考基流值。

我们假设 ρ_0 均匀且稳态，因此可以用密度涨落 ρ'来表述这些方程。为简化记号，省略撇号后，(ρ, u_k)便表示计算中要求解的未知变量。

使用当前的 OSSLBM 模型，我们可以在层流状态下模拟这些方程。这里，我们在一个 8×8 网格上设置问题，网格中有一个 2×2 浸没方形物体，如图 14所示。我们假设网格顶部和底部的边界条件严格为零，并设置固定入口

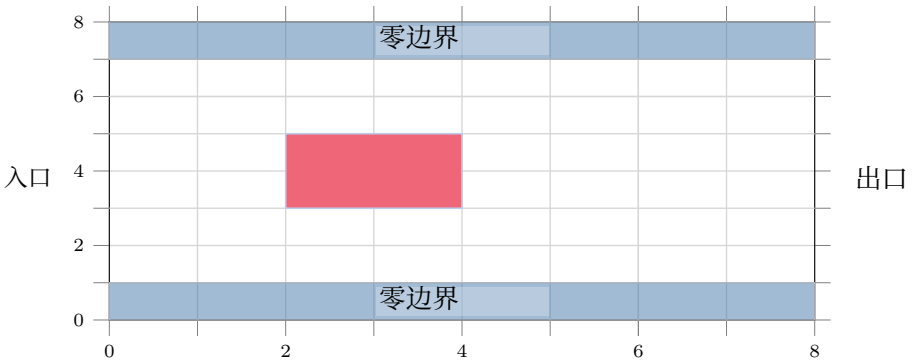

图 14. 翼型模型设置。

左侧边界层处指定的速度。入口速度以无量纲格子单位表示为 $u_x = 0.02$, $u_y = 0$ ——由于实验旨在将理想结果与含噪量子处理器模拟进行比较，因此我们不尝试使模型匹配任何实际物理数据。特别是，边界条件和物体形状的选择旨在最大限度地降低电路复杂度。我们还采用遵循 Tau1 简化的 D2Q9 格子配置（其格子黏度为 $1/6$）。图 15展示了该模型经过 15 个时间步后的状态，该结果由理想状态矢量模拟器生成，初始态为均匀速度，且与入口条件相匹配。

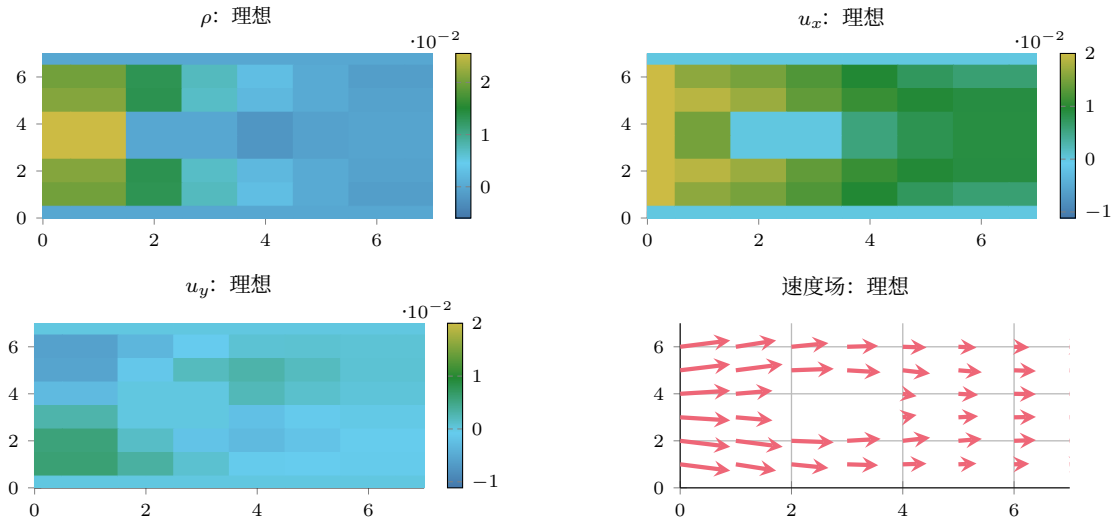

图15. 理想（状态向量）模拟的结果，以及经过15个时间步后重建的流场。该模拟尚未达到稳态，但已符合具有物理意义的流动模式。

相应的量子电路需要六个量子比特来编码格子，并需要八个辅助量子比特来定义问题。与密度场和速度场对应的三个变量（及其非线性组合）被映射为按格子寄存器索引的振幅，每个场在叠加寄存器 s 中占据一个子态。

6.1. **实验设置**。 对于所考虑的示例，量子-经典混合算法流程具有以下结构（另见图 2）：

（1）密度和速度（沿 x方向和 y方向）场的初始条件与入口速度一同定义。此处我们将密度和 y 方向速度设为0，并将x方向速度处处设为等于入口速度0.02。（2）非线性项数组（27）以经典方式制备，并在态制备期间加载到量子计算机上。这需要寄存器 s 中的三个叠加量子比特来标记这六个非线性项。（3）在硬件上执行实现单次格子玻尔兹曼方法步骤的量子电路。（4）执行带后选择的测量，以计入量子格子玻尔兹曼方法中的非线性演化。根据收集到的采样次数进行层析成像（见下文及附录B），以恢复密度场和速度场。（5）依据非线性数组与碰撞算子的原始范数，对每个场额外进行重新归一化。此外，将格子上的左边界层设为入口速度 $u_x = 0.02$。（6）从步骤2 起重复该过程，直至解收敛，或达到设定的步数。

为通过测量恢复各个场，我们引入并执行函数层析，将场表示为若干特定基函数的有限和。这里，我们选择对每个坐标使用最高二次的切比雪夫多项式。我们还执行网格变换，将浸没物体映射到单个点，以提升层析性能；这使我们能够使用次数更低的多项式。有关层析方法的更多详情，请参见附录B。

各个场的范数可根据相应的可观测量恢复。具体而言，在辅助量子比特上测得标记各个场的比特串的概率，等于相应场的范数平方。计算这些可观测量只需从完整测量结果中对格子寄存器进行偏迹。

式（27）中非线性项在步骤2 的制备可以直接完成，因为可以对层析恢复函数显式执行经典计算。对于大规模实验，当格子寄存器较大时，在合成态制备线路之前，可能还需要将数组加载到矩阵乘积态（MPS）等高效中间表示中。这可以通过张量交叉插值完成 [34], ，它能够将给定函数表示为矩阵乘积态或张量网络，再据此高效制备量子态，如 [4, 12]所示。

6.2. **量子电路编译与执行细节**。 我们在超导 IBM Heron R3 设备上进行实验 `ibm boston`，该设备有 156 个量子比特，并采用重六边形连接。最终的电路结构如图16 所示（另见图2）。然而，直接对量子电路进行未优化编译会导致电路深度很大。为提升其在噪声硬件上的性能，我们实施了多项编译改进。

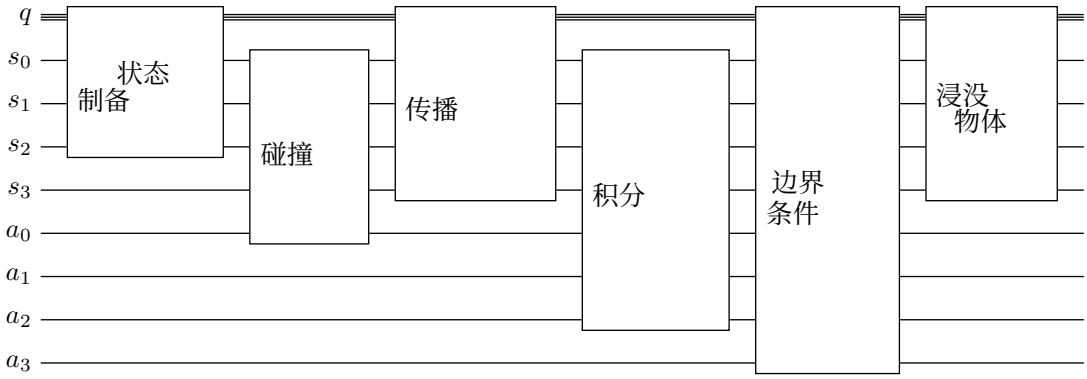

图 16. 用于硬件实验的完整 LBM量子电路结构，展示了功能模块及受影响的寄存器。有关各步骤的更多详情，请参阅第3节。

首先请注意，碰撞算子是在 s寄存器的四个量子比特上制备的，且与晶格尺寸无关（见第3.3节）。我们没有直接加载式（27）中的输入变量，并将碰撞步实现为幺正算子，而是将碰撞算子并入态制备中。这始终可以通过经典方式完成，但会使态制备所需的量子比特数比原来多两个。我们测试了 MPS [12, 62]以及更一般的张量网络 [4,75] 态制备技术，发现后者在处理多维数据和多个变量时性能更好，但基于张量网络的态制备会产生具有长程连通性的量子电路。转译到近线性拓扑的设备后，由于引入了额外的交换门，其性能优势通常会消失。因此，本实验采用了更简单的线性连通 MPS 态制备。

第二项改进涉及边界。量子电路末端部分（部分实现边界条件，完整实现浸没物体）仅由 X 门、CX门及其多重受控版本组成（见第3.6 节和第 3.7节）。这些量子门与 Z 基测量可交换，因此可以反向传播 [38] ，并在已测得的比特串上以经典方式执行。反向传播大幅降低了电路复杂度，但也限制了测量只能在计算基中进行。因此，本实验在硬件上执行量子电路时，仅进行绝对值函数层析（见附录 B）。由于所关注的问题是具有对称物体的层流，我们可以根据其对称性恢复解的相对符号。

最后，我们在量子电路的其余部分采用经过优化的量子门实现。对于托福利门和多重控制 X 门，只要条件允许，我们就使用相对相位分解 [77] 。对于多重控制旋转门，我们采用高效的多重控制特殊酉实现 [121]。

这些编译步骤结合起来，使我们能够将全连接情况下的 CX门数量从直接实现的 800 个减少到约 300 个。转译到 `ibm boston` 量子处理器后，双量子比特门数量从 1825 个减少到 540 个。

在硬件上执行量子电路时，我们还采用了额外的误差缓解技术。我们使用带有 XX 序列的动态解耦（DD），以应对量子电路中的较长空闲时间 [33]。我们还通过 16 个随机实例 [124] 执行泡利随机化，以使噪声效应对称化。此外

采用算子退相干重整化（ODR）来校正可观测量期望值，这些期望值定义了各个场的范数 [120]。

总共执行了 14 条量子线路，以迭代方式完成从时间步 1 到 15 的演化，并额外使用一条 ODR 量子电路，因为电路结构不会改变。对于每条量子电路，我们进行了 30,000 次采样。采样次数必须考虑到比特串必须经过后选择这一事实；在理想模拟中，仅有 3% 到 6% 的采样（取决于具体步骤）能通过后选择，并分配到其对应的场。实际层析成像仅使用数百次采样来处理密度和 x 速度场，并使用数十次采样来处理 y 速度场。

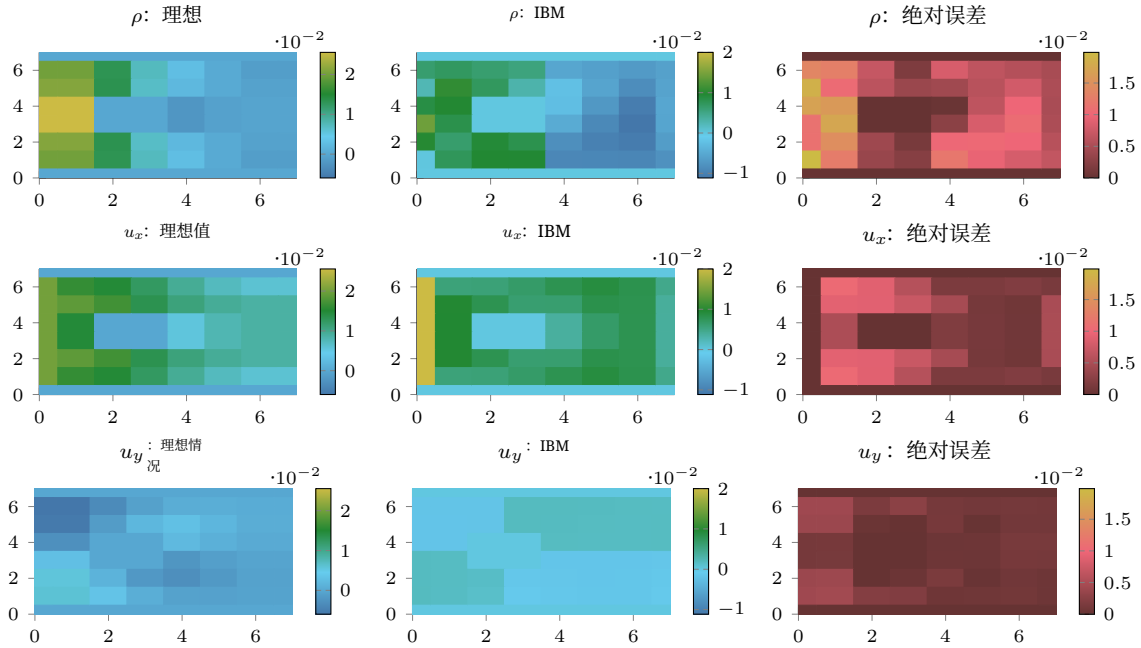

图17. 理想（状态向量）与硬件执行（ibm boston）的场数据结果：经过15个时间步后的结果及绝对误差。上：密度 ρ；中： x-速度 u_x；下： y-速度 u_y。每列使用相同的色标。模型假设$\tau = 1$，并在左侧设定固定入口速度， $u_x = 0.02$， $u_y = 0$ 以无量纲格子单位表示。底部和顶部边界均采用零狄利克雷边界条件。

6.3. **实验结果**。 量子计算机上的执行结果如图 17所示，其中我们将各个场与理想数据进行比较，并展示每个格点处的绝对误差。得到的场正确再现了目标状态的定性特征，不过y-速度因量子噪声而有些“扁平化”。理想解的完整细节见图15。

在图18 中，我们进一步比较了三个场向模型稳态收敛的情况。该稳态是通过在状态向量模拟器上执行算法足够长时间而得到的。图18中以蓝色显示状态向量模拟结果，以红色显示包含散粒噪声的理想模拟结果，以绿色显示在含噪 IBM 量子处理器上执行量子电路的结果。尽管结果受到噪声影响，该算法仍持续收敛，均方误差随着后续步骤不断降低。

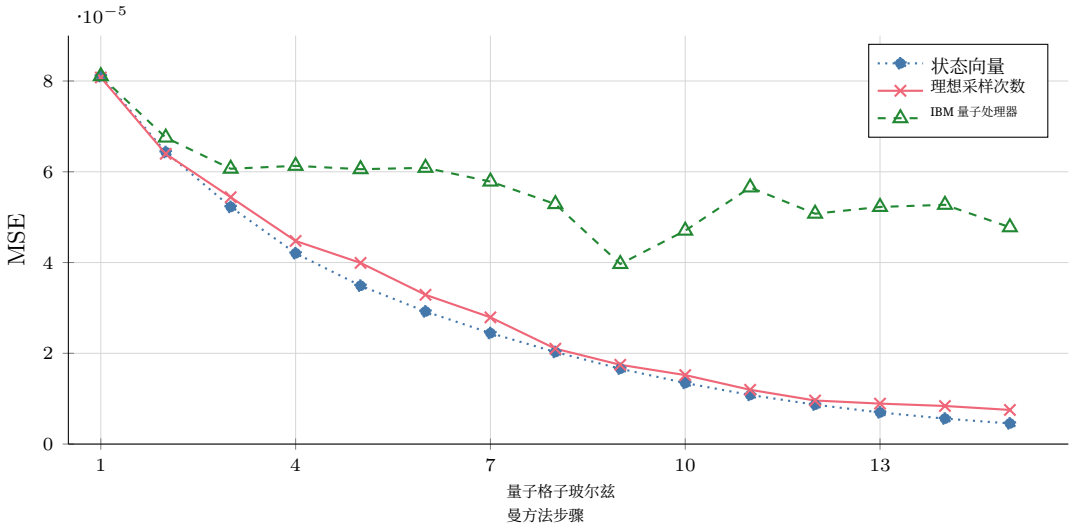

图18. 非线性翼型模拟收敛至稳态的完整状态，以均方误差（MSE）计算；比较了三种不同情形：状态向量模拟、采用固定采样次数并通过层析成像恢复结果的理想模拟，以及在 `ibm boston` 量子计算机上的执行。

7. 展望未来

我们提出了一套全面的模型，用于设计灵活的格子玻尔兹曼方法量子算法。其核心模型——碰撞、传播和积分——门复杂度较低，且设置相对简单。对于一部分适用于格子玻尔兹曼方法处理的线性物理问题，采用完全酉、端到端的完整量子算法甚至很有可行性。

不过，仍有一些难点尚待解决。通用情形下，部分算子的非酉性，以及非线性物理建模的困难，是量子格子玻尔兹曼模拟走向工业规模应用的两大障碍。

非酉性会导致时间步进过程中振幅耗散，从而降低一次模拟结束时测得有意义数据的概率。对此有多种选择：首先，或许可以将至少部分算子酉化。这既可以直接实现，也可以借助采用酉拟设的量子机器学习来实现 [58, 67]。近期研究表明，这对于线性情形是可行的：忽略非线性项 $\mathcal{O}(u^2)$ 后，可以近似得到 LBM 的平衡分布函数。这在低雷诺数和低速度下尤其成立。利用该技术生成的量子线路更浅、具有酉性且错误率较低。尽管取得了这些成果，但所学习算子的一般性，以及其在涉及多个时间步的模拟中的效率，仍有待验证。第二种选择是通过振幅放大来增强数据。近期工作 [138] 研究了利用分块编码或线性组合酉算子，对未知初始状态向量的非酉算子应用无意识振幅放大。该研究表明，尽管应用振幅放大会在状态向量的相对振幅中引入误差，但其近似版本仍可能得到可行的解。最后，也可以考虑设计替代的数据编码方法，而非仅采用纯振幅编码。这些方法并不互斥，我们最终或许会以某种形式结合使用它们。

非线性也与振幅编码相关，因为能够直接操纵振幅的算子必然是酉算子，因此也是线性的。然而，越来越多的文献开始探讨如何处理非线性算子，和／或如何改变编码方式，以便在酉算子之外获得更大的灵活性。

参考文献

[1] T. Abe. “通过离散纵标法由玻尔兹曼方程推导格子玻尔兹曼方法”. 载于：计算物理学杂志 131.1 (1997).[2] P. Angot、C.-H. Bruneau与P. Fabrie. “一种在不可压缩黏性流中考虑障碍物的惩罚法”. 载于：数值数学 81.4 (1999)，第497–520页。[3] M. Bacza、T. Tomczak与M. Matyka. “弛豫时间等于1时用于流体流动的格子玻尔兹曼方法：性能研究”. 2023. doi: 10.48550/arxiv.2312.14742. arXiv:2312.14742 [physics.comp-ph].[4] M. Ballarin等人. “利用张量网络高效制备多元函数的量子态”. 2025. arXiv: 2511.15674 [quant-ph]. url: https://arxiv.org/abs/2511.15674.[5] A. D. Bastida Zamora等人. “高效的量子格子气自动机”. 载于：Computers & Fluids 286 (2025年1月)，第106476页. doi: 10.1016/j.compfluid.2024.106476.[6] A. D. Bastida-Zamora等人. “多时间步的量子格子玻尔兹曼方法：一种局部卡尔曼线性化算法”. 载于：《物理评论E》(2026年2月)，第–页. doi: 10.1103/kbx3-c35d. url: https://link.aps.org/doi/10.1103/kbx3-c35d.[7] M. C. Bedrunka等人. “基于不变网络的机器学习增强型格子玻尔兹曼方法碰撞算子”. 载于：《物理评论E》112 (2025年11月5日)，第055308页. doi:10.1103/rbf2-p8tf. url: https://link.aps.org/doi/10.1103/rbf2-p8tf.[8] H. Bethe等人. IBM内爆流体力学计算. 技术报告. 洛斯阿拉莫斯国家实验室报告，LA-94，1944.[9] P. L. Bhatnagar、E. P. Gross与M. Krook. “气体中碰撞过程的模型。I. 带电和中性单组分系统中的小振幅过程”. 载于：《物理评论》94 (1954年5月3日)，第511–525页. doi: 10.1103/PhysRev.94.511.[10] X. Bi等人. “带修正函数的格子玻尔兹曼BGK模型，用于二维二阶非线性偏微分方程”. 载于：Entropy 27.7 (2025)，第717页。[11] V. Bohun、A. Kuzmak与M. Koch-Janusz. “复杂几何中线性化欧拉方程的量子哈密顿量模拟”. arXiv 预印本 arXiv:2510.17978. 2025.[12] V. Bohun等人. “光滑函数的矩阵乘积态表示中的纠缠标度及其浅层量子电路近似”. 2025. arXiv: 2412.05202 [quant-ph]. url: https://arxiv.org/abs/2412.05202.[13] P. Boivin、M. Tayyab与S. Zhao. “反应流格子玻尔兹曼求解器的基准测试：该方法对燃烧是否值得投入？”载于：《流体物理学》33.7 (2021).[14] O. Boolakee、M. Geier与L. De Lorenzis. “用于线性弹性动力学的格子玻尔兹曼方法：周期性问题与狄利克雷边界条件”. 载于：《计算机方法在应用力学与工程中的应用》433 (2025)，第117469页。

[15] Lj. Budinski. “用格子玻尔兹曼方法模拟对流–扩散方程的量子算法”. 载于：《量子信息处理》20.2 (2021). doi: 10.1007/s11128-021-02996-3.[16] Lj. Budinski. “基于流函数-涡量表述与格子玻尔兹曼方法的纳维–斯托克斯方程量子算法”. 载于：《国际量子信息杂志》20.02 (2022)，第2150039页. doi: 10.1142/S0219749921500398. url: ttps://doi.org/10.1142/S0219749921500398.[17] M. Cerezo等人. “变分量子算法”. 载于：《自然综述：物理》3 (2021年8月)，第625–644页. doi: 10.1038/s42254-021-00348-9.[18] Z. Chai等人. “高阶非线性偏微分方程的格子玻尔兹曼模型”. 载于：《物理评论E》97.1 (2018)，第013304页。[19] R. Chen等人. “用于模拟声传播问题的声学多极源–简化格子玻尔兹曼方法”. 载于：《国际流体数值方法杂志》95.7 (2023)，第1174–1196页。[20] Z. Chen、C. Shu与D. Tan. “一种无需演化分布函数的简化热格子玻尔兹曼方法”. 载于：International Journal of Heat and Mass Transfer 105 (2017)，第741–757页。[21] Z. Chen等人. “一种无需演化分布函数的简化格子玻尔兹曼方法”. 载于：《应用数学与力学进展》9.1 (2017)，第1–22页。[22] Z. Chen等人. “关于简化且高度稳定的格子玻尔兹曼方法的改进：公式表述、边界处理与稳定性分析”. 载于：《国际流体数值方法杂志》87.4 (2018)，第161–179页。[23] Z. Chen与C. Shu. 简化且高度稳定的格子玻尔兹曼方法：理论与应用. 第5卷. 世界科技出版公司，2020.[24] Z. Chen、C. Shu与D. S. Tan. “一种真正二阶且无条件稳定的热格子玻尔兹曼方法”. 载于：《应用科学》 7.3 (2017)，第 277 页。[25] A. Corbetta 、 A. Gabbana 、 V. Gyrya 等人 . “迈向学习格子玻尔兹曼碰撞算子”. 载于：《欧洲物理学杂志E》46 (2023)，第10页. doi: 10.1140/epje/s10189-023-00267-w. url: https://doi.org/10.1140/epje/s10189-023-00267-w.[26] A. De Rosis. “浅水方程的基于中心矩的格子玻尔兹曼格式”. 载于：《计算机方法在应用力学与工程中的应用》319 (2017)，第379–392页。[27] A. De Rosis. “亚临界浅水流的格子 Boltzmann 格式比较”. 载于：《流体物理学》35.4 (2023).[28] A. De Rosis. “用格子玻尔兹曼方法模拟流行病”. 载于：《物理评论E》102.2 (2020)，第023301页。[29] A. Delgado-Guti´errez等人. “用于高湍流的单步简化图形处理器格子玻尔兹曼方法”. 载于：《国际流体数值方法杂志》93.7 (2021)，第2339–2361页。[30] A. Delgado-Guti´errez等人. “不规则流体区域中图形处理器加速的单步简化格子玻尔兹曼方法的高效实现”. 载于：《流体物理学》34.12 (2022).

[31] M. Escande 等. “采用规则格子的弹性固体波传播格子玻尔兹曼方法：理论分析与验证”。2020。[32] O. Ezratty。“量子计算是否存在摩尔定律？”2023。[33] N. Ezzell 等.“超导量子比特的动态解耦：性能综述”。载于：Physical Review Applied 20.6（2023年12月）。issn：2331-7019。doi：10.1103/physrevapplied.20.064027。url：http://dx.doi.org/10.1103/PhysRevApplied.20.064027。[34] Y. N. Fern´andez 等.“利用张量交叉插值学习张量网络：新算法与库”。载于：SciPost Phys. 18（2025），第104页。doi：10.21468/SciPostPhys.18.3.104。url：https://scipost.org/10.21468/SciPostPhys.18.3.104。[35] J. Floryan。“用于周期构型流动的保角映射坐标生成方法”。载于：计算物理学杂志 62.1（1986），第221–247页。issn：0021-9991。doi：https://doi.org/10.1016/0021-9991(86)90108-7。url：https://www.sciencedirect.com/science/article/pii/0021999186901087。[36] N. Fonio、P. Sagaut 和 G. Di Molfetta。“用于流体动力学格子气自动机的量子碰撞电路、量子不变量和量子相位估计过程”。载于：Computers & Fluids 299（2025年8月），第106688页。issn：0045-7930。doi：10.1016/j.compfluid.2025.106688。[37] N. Fonio、P. Sagaut 和 G. Di Molfetta。“用于各向异性类 Burgers 方程的二维量子格子气算法”。2026。doi：10.48550/arXiv.2602.17303。arXiv：2602.17303[quant-ph]。url：https://arxiv.org/abs/2602.17303。[38] B. Fuller 等.“通过算子反向传播改进量子计算”。载于：npj Quantum Information（2026）。issn：2056-6387。doi：10.1038/s41534-026-01196-0。url：https://doi.org/10.1038/s41534-026-01196-0。[39] C. A. Georgescu、M. A. Schalkers 和M. M¨oller。“QLBM——量子格子玻尔兹曼软件框架”。载于：Computer Physics Co mmunications 315（2025），第109699页。issn：0010-4655。doi：https://doi.org/10.1016/j.cpc.2025.109699。url：https://www.sciencedirect.com/science/article/pii/S0010465525002012。[40] M. Geveler 等.“多核与众核处理器上浅水方程的格子玻尔兹曼模拟及流固耦合”。载于：《面向多核挑战：并行计算中新范式与技术的若干方面》。Springer，2010，第92–104页。[41] R. Glowinski 等.“采用虚拟区域方法对运动刚体周围的不可压缩黏性流进行直接数值模拟：颗粒流应用”。载于：计算物理学杂志 169.2（2001），第363–426页。[42] Z. Guo、B. Shi 和 N. Wang。“不可压缩纳维–斯托克斯方程的格子 BGK 模型”。载于：计算物理学杂志 165.1（2000），第288–306页。[43] S. M. Hanasoge、S. Succi 和 S. A. Orszag。“用于电磁波传播的格子玻尔兹曼方法”。载于：《欧洲物理快报》96.1（2011），第14002页。[44] L. Hantzko、L. Binkowski 和 S. Gupta。“张量化泡利分解算法”。载于：Physica Scripta 99.8（2024年7月），第085128页。doi：10.1088/1402-4896/ad6499。[45] A. W. Harrow、A. Hassidim 和 S. Lloyd。“线性方程组量子算法”。载于：Phys. Rev. Lett. 103（2009年10月15日，第15期），第150502页。doi：10.1103/PhysRevLett.103.150502。[46] A. Hauser 和 J. Verhey。“适用于介质中麦克斯韦方程的稳定格子玻尔兹曼模型”。载于：Physical Review E 96.6（2017），第063306页。

[47] X. He、G. D. Doolen 和 T. Clark。“格子玻尔兹曼方法与人工可压缩性方法在纳维–斯托克斯方程中的比较”。载于：计算物理学杂志 179.2（2002），第439–451页。[48] X. He 和 L.-S. Luo。“不可压缩纳维–斯托克斯方程的格子玻尔兹曼模型”。载于：统计物理学杂志 88（1997），第927–944页。[49] X. He 和 L.-S. Luo。“格子玻尔兹曼方法理论：从玻尔兹曼方程到格子玻尔兹曼方程”。载于：《物理评论E》56.6（1997），第6811页。[50] F. Higuera 和 J. Jim´enez。“格子气体模拟的玻尔兹曼方法”。载于：《欧洲物理快报》9.7（1989），第663页。[51] F. Higuera、S. Succi 和 R. Benzi。“具有增强碰撞的格子气体动力学”。载于：《欧洲物理快报》9.4（1989），第345页。[52] C. Hirsch。“内流与外流的数值计算。第1卷——数值离散基础”。载于：奇切斯特（1990）。[53] C. Hirsch。“内流与外流的数值计算。第2卷——无黏与有黏流动的计算方法”。载于：奇切斯特（1990）。[54] S. A. Hosseini、N. Darabiha 和 D. Th´evenin。“湍流燃烧的低马赫数格子玻尔兹曼模型：受限几何中的流动”。载于：燃烧学会论文集 39.4（2023），第5357–5364页。[55] X. Huang 等。“实空间与傅里叶空间读出方法：复杂度比较及其在计算流体力学问题中的应用”。2025。arXiv：2511.20017 [quant-ph]。url：https://arxiv.org/abs/2511.20017。[56] T. Inamuro。“具有传热功能的不可压缩黏性流格子动力学格式”。载于：《伦敦皇家学会哲学汇刊》。A辑：数学、物理与工程科学 360.1792（2002），第477–484页。[57] S. Inui 和 M. Tsubota。“基于格子玻尔兹曼方法的超流氦-4中量子化涡旋与正常流体的耦合动力学”。载于：《物理评论B》104.21（2021），第214503页。[58] W. Itani 和 K. R. Sreenivasan。“面向流体动力学模拟的格子玻尔兹曼碰撞算子量子机器学习”。版本 v2，最后修订于2026年1月7日。2025。arXiv：2512.23991 [quant-ph]。url：https://doi.org/10.48550/arXiv.2512.23991。[59] W.-S. Jiaung、J.-R. Ho 和 C.-P. Kuo。“具有相变的热传导问题的格子玻尔兹曼方法”。载于：《数值传热：B辑：基础》39.2（2001），第167–187页。[60] S. Jin、N. Liu 和 Y. Yu。“线性高维多尺度偏微分方程量子差分方法的时间复杂度分析”。载于：计算物理学杂志 471（2022），第111641页。doi：10.1016/j.jcp.2022.111641。[61] B. Kadoch 等。“用于不可压缩流动及含运动障碍物的标量对流–扩散问题的体积惩罚法”。载于：计算物理学杂志 231.12（2012），第4365–4383页。[62] O. Kerppo 等。“通过最小化纠缠熵来改进量子态制备”。arXiv 预印本。2025。doi：10.48550/arxiv.2507.22562。arXiv：2507.22562 [quant-ph]。[63] K. Khadra 等。“用于纳维–斯托克斯方程数值建模的虚拟区域方法”。载于：《国际流体数值方法杂志》34.8（2000），第651–684页。

[64] S. Kocherla 等人。“一种减少量子格子玻尔兹曼方法量子资源的双电路方法”。2024年。[65] D. Kolomenskiy 与 K. Schneider。“一种用于纳维–斯托克斯方程的傅里叶谱方法，采用体积惩罚处理运动固体障碍物”。载于：计算物理学杂志 228.16（2009），第5687–5709页。[66] T. Kr¨ger 等人。《格子玻尔兹曼方法：原理与实践》。Springer，2016年。[67] M. L˘ac˘atu.s 与 M. M¨oller。“格子玻尔兹曼碰撞算子的代理量子线路设计”。2025年。网址：https://doi.org/10.48550/arXiv.2507.12256。[68] D. Lapitski 与 P. J. Dellar。“三维量子格子玻尔兹曼格式向狄拉克方程解的收敛”。载于：英国皇家学会哲学汇刊A：数学、物理与工程科学 369.1944（2011），第2155–2163页。[69] M. Larocca 等人。“变分量子计算中的贫瘠高原”。载于：《自然综述：物理》7（2025），第174–189页。doi：10.1038/s42254-025-00171-8。[70] C.-K. Li、K. Y. Wu 与 Z. Zhang。“通过稀疏项优化实现高效的基于电路的量子态层析”。2025年。arXiv：2407.20298 [quant-ph]。网址：https://arxiv.org/abs/2407.20298。[71] X. Li 等人。“流体动力学模拟的潜在量子优势”。载于：物理评论研究 7.1（2025），第013036页。[72] V. D. Liseikin。《网格生成方法》。第1卷。Springer，1999年。[73] X. Liu 等人。“基于介观正则化格子玻尔兹曼方法的宏观有限差分格式”。载于：《物理评论E》109.2（2024），第025301页。[74] V. J. Llorente。“湍流中体积惩罚浸入边界法的理论考量”。载于：《流体物理学》36.7（2024）。[75] H. Manabe 与 Y. Sano。“利用树张量网络制备多元正态分布的态”。载于：Quantum 9（2025年5月），第1755页。issn：2521-327X。doi：10.22331/q-2025-05-28-1755。网址：http://dx.doi.org/10.22331/q-2025-05-28-1755。[76] N. Maquignon 等人。“一种简化且稳定的格子玻尔兹曼浅水模型”。载于：物理学杂志：会议系列。第2202卷。第1期。IOP出版社。2022年，第012055页。[77] D. Maslov。“使用相对相位托佛利门的优势及其在多控制托佛利门优化中的应用”。载于：Physical Review A 93.2（2016），第022311页。[78] M. Matyka 与 M. Dzikowski。“适用于低雷诺数流动的内存高效格子玻尔兹曼方法”。载于：Computer Physics Communications 267（2021），第108044页。issn：0010-4655。doi：10.1016/j.cpc.2021.108044。网址：https://www.sciencedirect.com/science/article/pii/S0010465521001569。[79] G. G. McNamara 与 G. Zanetti。“利用玻尔兹曼方程模拟格子气自动机”。载于：物理评论快报 61.20（1988），第2332–2335页。[80] M. Mendoza 与 J. Munoz。“用于电动力学的三维格子玻尔兹曼模型”。载于：Physical Review E—Statistical, Nonlinear, and Soft Matter Physics 82.5（2010），第056708页。[81] J. Meng、Y. Qian 与 S. Dai。“用格子玻尔兹曼模型对城市交通网络建模”。载于：《欧洲物理快报》81.4（2008），第44003页。[82] J. Meng 等人。“交通流的格子玻尔兹曼模型”。载于：Physical Review E—Statistical, Nonlinear, and Soft Matter Physics 77.3（2008），第036108页。

[83] R. Moser。《湍流模拟中的数值方法》。爱思唯尔，2022年。[84] H. M¨uller、J. Panov 与 R. M¨uller。“弹性固体格子玻尔兹曼方法中采用TRT碰撞格式改善稳定性”。载于：PAMM 24.4（2024），e202400108。[85] J. S. N. Murthy 等人。“弹性固体中波传播的格子玻尔兹曼方法”。载于：《计算物理通讯》23.4（2018），第1223–1240页。[86] H. ng。“基于格子玻尔兹曼方法的Gross-Pitaevskii方程数值模拟”。载于：《空间研究进展》60.6（2017），第1261–1270页。[87] O. Niemim¨aki 等人。“量子基态的移位”。预印本。2026年。[88] G. S. O’Brien、T. Nissen-Meyer 与 C. Bean。“泊松固体中弹性波传播的格子玻尔兹曼方法”。载于：《美国地震学会通报》102.3（2012），第1224–1234页。[89] R. Peyret。《不可压缩黏性流的谱方法》。第148卷。Springer，2002年。[90] Y.-H. Qian、D. d’Humi`eres 与 P. Lallemand。“纳维–斯托克斯方程的格子BGK模型”。载于：《欧洲物理快报》17.6（1992），第479页。[91] S. Qin 等人。“一种新型单步简化格子玻尔兹曼方法及其在大密度比多相流中的应用”。载于：《流体物理学》35.5（2023年5月），第053318页。issn：1070-6631。doi：10.1063/5.0150935。[92] S. Qin 等人。“一种无需分布函数演化的单步简化格子玻尔兹曼方法”。载于：《国际流体数值方法杂志》94.7（2022），第1001–1025页。doi：10.1002/fld.5081。[93] S. Qin 等人。“布西内斯克近似下热流的单步简化格子玻尔兹曼方法”。载于：Phys. Rev. E 108（2023年10月4日），第045305页。doi：10.1103/PhysRevE.108.045305。[94] R. Richtmyer 与 N. C. Metropolis。“现代计算”。载于：《今日物理》2.10（1949），第8–15页。[95] S. Rienstra 与 A. Hirschberg。《声学导论》。英文。IWDE报告。埃因霍温理工大学，1992年。[96] K. Rupp 与 S. Selberherr。“摩尔定律的经济极限”。载于：《IEEE半导体制造汇刊》24.1（2010），第1–4页。[97] C. Sanavio 等人。“经典流体量子模拟的三条Carleman路径”。载于：《流体物理学》36（5 2024），第057143页。doi：10.1063/5.0204955。[98] C. Sanavio、E. Mauri 与 S. Succi。“流体量子模拟的Carleman-grad方法”。2024年。doi：10.48550/arXiv.2406.01118。arXiv：2406.01118 [quant-ph]。[99] C. Sanavio 等人。“带矩阵访问预言机的Carleman-格子玻尔兹曼量子电路”。载于：《流体物理学》37（3 2025），第037123页。doi：10.1063/5.0254588。[100] M. A. Schalkers 与 M. M¨oller。“量子玻尔兹曼方法的动量交换方法”。2024年。doi：10.48550/arXiv.2404.17618。arXiv：2404.17618 [quant-ph]。[101] T. K. Sengupta。《计算流体力学基础》。大学出版社，2004年。[102] A. Shakeel。“基于量子傅里叶变换的高效可扩展量子游走算法”。载于：《量子信息处理》19.323（2020年8月）。doi：10.1007/s11128-020-02834-y。[103] X. Shan、X.-F. Yuan 与 H. Chen。“流体动力学的动理学理论表述：超越纳维–斯托克斯方程的途径”。载于：《流体力学杂志》550（2006），第413–441页。

[104] W. Shi 等. “具有平衡交通压力的交通流修订格子玻尔兹曼模型”。载于：Physica A: Statistical Mechanics and its Applications 443 (2016)，第22–31页。[105] P. Sikdar、S. M. Dash 和 K. P. Sinhamahapatra. “一种基于灵活强迫浸入边界格式的单步简化格子玻尔兹曼方法，用于二维流固相互作用问题”。载于：Computers & Fluids 265 (2023)，第105996页。[106] Z. Song 等. 哈达玛随机森林：以指数级减少测量设置重建实值量子态。2025年。arXiv： 2505.06455 [quant-ph]。网址：https://arxiv.org/abs/2505.06455。[107] Z. Song 等. “通过量子–经典混合方案在含噪量子硬件上求解不可压缩纳维–斯托克斯方程”。载于：Computers & Fluids 288（2025年2月），第106507页。ISSN：0045-7930。DOI： 10.1016/j.compfluid.2024.106507。网址：http://dx.doi.org/10.1016/j.compfluid.2024.106507。[108] H. Su、S. Xiong 和 Y. Yang. “利用切比雪夫多项式实现高效量子态层析”。2025年。arXiv： 2509.02112 [quant-ph]。网址：https://arxiv.org/abs/2509.02112。[109] S. Succi.《格子玻尔兹曼方程：流体动力学及其他领域》。牛津大学出版社，2001年。[110] S. Succi 和 R. Benzi. “用于量子力学的格子玻尔兹曼方程”。载于：Physica D 69.3-4 (1993)，第327–332页。[111] S. Succi、F. Fillion-Gourdeau 和 S. Palpacelli. “量子格子玻尔兹曼方法是一种量子游走”。载于：EPJ Quantum Technology 2.1 (2015)，第1–17页。[112] S. Succi 等. “流体领域的量子计算：进展如何？”载于：《欧洲物理快报》144.1（2023年10月），第10001页。DOI： 10.1209/02955075/acfdc7。[113] K. Suzuki 和 T. Inamuro. “一种改进的格子动力学格式，用于不可压缩黏性流动”。载于：International Journal of Modern Physics C 25.01 (2014)，第1340017页。[114] K. Suzuki 等. “用于不可压缩黏性单相和两相流动的简易扩展格子玻尔兹曼方法”。载于：《流体物理学》33.3 (2021)。[115] S. Taileb 等. “氢气射流火焰的格子玻尔兹曼建模：一种危险点火预测新模型”。载于：Combustion and Flame 245 (2022)，第112317页。[116] A. Termanova 等. “张量量子编程”。载于：New Journal of Physics 26.12 (2024)，第123019页。DOI： 10.1088/1367-2630/ad985b。网址： https://doi.org/10.1088/1367-2630/ad985b。[117] T. N. Theis 和 H.-S. P. Wong. “摩尔定律的终结：信息技术的新开端”。载于：Computing in science & engineering 19.2 (2017)，第41–50页。[118] A. Tiwari 等. “迈向可实现量子格子玻尔兹曼方法的算法进展”。arXiv预印本。2025年。arXiv：2504.10870。[119] G. Torlai 等. “结合无监督学习和张量网络的量子过程层析”。载于：Nature Communications 14.1（2023年5月），第2858页。ISSN：2041-1723。DOI： 10.1038/s41467-023-38332-9。网址：https://doi.org/10.1038/s41467-023-38332-9。[120] M. Urbanek 等. “利用噪声估计电路缓解量子计算机中的退极化噪声”。载于：物理评论快报 127.27（2021年12月）。ISSN：1079-7114。DOI： 10.1103/physrevlett.127.270502。网址：http://dx.doi.org/10.1103/PhysRevLett.127.270502。

[121] R. Vale 等. 多重受控特殊酉单量子比特门的分解。2023年。arXiv：2302.06377 [quant-ph]。网址： https://arxiv.org/abs/2302.06377。[122] J. Von Neumann. 用于处理水动力激波问题的新数值方法的提案与分析。应用数学组，高等研究院，1944年。[123] J. Von Neumann 和 D. Richtmyer. “一种水动力激波数值计算方法”。载于：J. Appl. Phys 21 (1950)，第332页。[124] J. J. Wallman 和 J. Emerson. “通过随机化编译实现可扩展量子计算的噪声整形”。载于：Physical Review A 94.5（2016年11月）。ISSN：2469-9934。DOI： 10.1103/physreva.94.052325。网址： http://dx.doi.org/10.1103/PhysRevA.94.052325。[125] D. Wawrzyniak 等. “用于量子格子玻尔兹曼方法的动态电路”。arXiv预印本。2025年。[126] Y. Wei 等. “用于辐射-传导传热的统一格子玻尔兹曼框架”。载于：International Journal of Heat and Mass Transfer 200 (2023)，第123513页。[127] B. Wilfong 等. “模拟多发动机航天器：通过信息几何正则化和mfc流动求解器实现超过一千亿个网格点”。2025年。[128] G. Wissocq 等. “用于气相爆轰的混合格子玻尔兹曼方法”。载于：计算物理学杂志 494 (2023)，第112525页。[129] D. A. Wolf-Gladrow.《格子气元胞自动机与格子玻尔兹曼模型：导论》。Springer，2004年。[130] Y. Xiao 等. “用于求解二维和三维不可压缩流动的量子格子动力学格式”。2025年。DOI： 10.48550/arxiv.2505.10883。arXiv： 2505.10883 [quant-ph]。[131] T. Xin 等. “求解线性微分方程的量子算法：理论与实验”。载于：Phys. Rev. A 101 (3 Mar. 2020)，第032307页。DOI： 10.1103/PhysRevA.101.032307。[132] L. Xiu. “时间摩尔：从时间角度利用摩尔定律”。载于：IEEE Solid-State Circuits Magazine 11.1 (2019)，第39–55页。[133] L. Xu 等. “迈向众核CPU-加速器系统上的特大城市尺度风流模拟”。载于：SIAM Journal on Scientific Computing 47.2 (2025)，B402–B427。[134] Y. Xue 等. “基于目标驱动格子玻尔兹曼模型的密集人群视频异常预测”。载于：International Journal of Applied Mathematics and Computer Science 27.1 (2017)，第181–194页。[135] K. Yamamoto、X. He 和 G. D. Doolen. “利用格子玻尔兹曼方法模拟燃烧场”。载于：统计物理学杂志 107.1 (2002)，第367–383页。[136] W. Yang 和 C. Li. “一种用于变系数四阶偏微分方程的统一格子玻尔兹曼模型”。载于：Entropy 24.9 (2022)，第1176页。[137] A. D. B. Zamora 等. “采用浮点数的格子气自动机：量子计算机中分子动力学与格子玻尔兹曼方法之间的联系”。载于：《物理评论E》112.1 (2025)，第015305页。DOI： 10.1103/PhysRevE.112.015305。[138] A. A. Zecchi 等. “用于经典输运问题量子模拟的改进振幅放大策略”。载于：Quantum Science and Technology 10.3 (2025)。开放获取，由IOP Publishing Ltd出版，第035039页。DOI： 10.1088/2058-9565/addeea。

[139] N. S. Zhdanova、I. V. Abalakin 和 O. V. Vasilyev。《广义 Brinkman 体积惩罚法在运动障碍物周围可压缩流中的应用》。载于：*Mathematical Models and ComputerSimulations* 14.5 (2022)，第 716–726 页。[140] L. Zhong 等。《用于非线性薛定谔方程的格子 Boltzmann 格式》。载于：¨物理评论 E—统计、非线性与软物质物理 74.3 (2006)，第 036704 页。[141] J. G. Zhou。《浅水方程的格子玻尔兹曼模型》。载于：《计算机方法在应用力学与工程中的应用》191.32 (2002)，第 3527–3539 页。[142] J. G. Zhou。《宏观格子 Boltzmann 方法》。载于：*Water* 13.1 (2021)。ISSN：2073-4441。DOI：10.3390/w13010061。网址：https://www.mdpi.com/2073-4441/13/1/61。

附录 A. 流体动力学与声学相关的格子玻尔兹曼模型

A.1. **用于流体动力学和声学模型的格子玻尔兹曼方法**。对于 Navier–Stokes 流体动力学，平衡部分由精确麦克斯韦分布的二次展开给出，因此 [66, 第 4]

$$f_\alpha^{eq}(x,t) = \rho W_\alpha \left(1 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u})^2}{2c_s^4} - \frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2c_s^2}\right), \tag{42}$$

非平衡部分定义为 $f_\alpha^{neq} = f_\alpha - f_\alpha^{eq}$。宏观量，即密度 ρ 和动量 $\rho\boldsymbol{u}$，按如下方式计算：

$$\sum_i f_i^{eq} = \rho, \quad \sum_i \boldsymbol{c}_i f_i^{eq} = \rho\boldsymbol{u} \tag{43}$$

此表达式适用于弱可压缩非热流体流动，能够捕捉剪切扰动和声学扰动。相应的宏观流体动力学控制方程为

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{44}$$

$$\frac{\partial \rho\boldsymbol{u}}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u} + p\boldsymbol{I}) = \nu \nabla^2 \boldsymbol{u} \tag{45}$$

其中，压力 p 由格子 Boltzmann 方法原生状态方程 $p = \rho c_s^2$给出， c_s 为特征格子速度。流体的物理运动黏度 ν 与格子速度 c_s、松弛参数 τ、网格尺寸 Δx 和时间步 Δt 之间满足以下关系

$$\nu = \frac{(\tau - 1/2)}{c_s^2} \frac{\Delta x^2}{\Delta t} \tag{46}$$

该模型可以修改为处理（近似）不可压缩流 [48], ，其中密度拆分为$\rho = \rho_0 + \rho'$，其中 ρ_0 和 $\rho' \ll \rho_0$ 分别表示基流密度和密度涨落。将这种拆分代入式（42），可得

$$f_\alpha^{eq}(x,t) = W_\alpha \left(\rho' + \rho_0 \left[\frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u})^2}{2c_s^4} - \frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2c_s^2}\right]\right), \tag{47}$$

其宏观未知量如下

$$\sum_i f_i^{eq} = \rho', \quad \sum_i \boldsymbol{c}_i f_i^{eq} = \rho_0 \boldsymbol{u} \tag{48}$$

这里假设按 $\rho'\boldsymbol{u}$ 缩放的项可以忽略。在这种不可压缩近似下，密度涨落与速度相互解耦。平衡函数关于 ρ' 呈线性，而关于 $\mathbf{u}$ 呈二次。

相应的宏观方程为

$$\frac{1}{c_s^2}\frac{\partial P}{\partial t}+\nabla\cdot\boldsymbol{u} = 0 \tag{49}$$

$$\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla P+\nu\nabla^2\boldsymbol{u}+\nu\nabla(\nabla\cdot\boldsymbol{u}) \tag{50}$$

与 $P=p/\rho_0$。已证明该系统与人工可压缩性方法相似，并且近似于不可压缩纳维–斯托克斯方程 [47, 48]:

$$\nabla\cdot\boldsymbol{u} = 0+O(M^2) \tag{51}$$

$$\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla P+\nu\nabla^2\boldsymbol{u}+O(M^3), \tag{52}$$

其中误差以马赫数 $M=u/c_s$ 的形式出现。值得注意的是，对于 OSSLBM 的量子算法，这种平衡函数表达式比 [42] 中提出的表达式更合适。

考虑声学扰动，可进一步对平衡函数进行线性化；此时速度可分解为 $\boldsymbol{u}=\boldsymbol{u}_0+\boldsymbol{u}'$，其中 $\boldsymbol{u}_0$ 和 $\boldsymbol{u}'$ 分别为基流速度和声学扰动速度场。根据线性声学假设，有 $\rho'\ll\rho_0$。为简便起见，可将动量分解为 $\rho\boldsymbol{u}=(\rho\boldsymbol{u})_0+(\rho\boldsymbol{u})'$，其中 $(\rho\boldsymbol{u})'=\rho_0\boldsymbol{u}'+\rho'\boldsymbol{u}_0+\rho'\boldsymbol{u}'\simeq\rho_0\boldsymbol{u}'+\rho'\boldsymbol{u}_0$，从而得到

$$\sum_i f_i^{eq}=\rho',\quad \sum_i \boldsymbol{c}_i f_i^{eq}=(\rho\boldsymbol{u})' \tag{53}$$

忽略所有非线性项，以恢复经典线性声学方程，可得

$$\begin{aligned} f_\alpha^{eq}(x,t) &= W_\alpha\left[\rho'\left(1+\frac{\boldsymbol{c}_\alpha\cdot\boldsymbol{u}_0}{c_s^2}+\frac{(\boldsymbol{c}_\alpha\cdot\boldsymbol{u}_0)^2}{2c_s^4}-\frac{\boldsymbol{u}_0\cdot\boldsymbol{u}_0}{2c_s^2}\right)\right. \\ &\quad \left.+\rho_0\left(\frac{\boldsymbol{c}_\alpha\cdot\boldsymbol{u}'}{c_s^2}+\frac{(\boldsymbol{c}_\alpha\cdot\boldsymbol{u}_0)(\boldsymbol{c}_\alpha\cdot\boldsymbol{u}')}{c_s^4}-\frac{\boldsymbol{u}_0\cdot\boldsymbol{u}'}{c_s^2}\right)\right], \end{aligned} \tag{54}$$

相应的宏观方程为

$$\frac{\partial\rho'}{\partial t}+\nabla\cdot(\rho\boldsymbol{u})' = 0 \tag{55}$$

$$\frac{\partial(\rho\boldsymbol{u})'}{\partial t}+\nabla\cdot((\rho\boldsymbol{u})_0\boldsymbol{u}'+(\rho\boldsymbol{u})'\boldsymbol{u}_0+p'\boldsymbol{I}) = \nu\nabla^2\boldsymbol{u}' \tag{56}$$

A.2. 用于浅水物理模型的格子玻尔兹曼方法。经典浅水方程由纳维-斯托克斯方程沿水床深度方向平均得到。引入水深 h 和深度平均流体速度二维向量 $\boldsymbol{u}$，平衡函数定义如下（例如参见 [26, 27, 40, 76, 141]）：

$$f_i^{eq}=\begin{cases} h-\dfrac{5gh^2}{2c_s^2}-\dfrac{2h}{c_s^2}(\boldsymbol{u}\cdot\boldsymbol{u}) & i=0 \\ \dfrac{gh^2}{2c_s^2}+\dfrac{h}{c_s^2}(\boldsymbol{c}_i\cdot\boldsymbol{u})+\dfrac{9h}{2c_s^4}(\boldsymbol{c}_i\cdot\boldsymbol{u})^2-\dfrac{h}{2c_s^2}(\boldsymbol{u}\cdot\boldsymbol{u}) & i=1,4 \\ \dfrac{1}{4}\left[\dfrac{gh^2}{2c_s^2}+\dfrac{h}{c_s^2}(\boldsymbol{c}_i\cdot\boldsymbol{u})+\dfrac{9h}{2c_s^4}(\boldsymbol{c}_i\cdot\boldsymbol{u})^2-\dfrac{h}{2c_s^2}(\boldsymbol{u}\cdot\boldsymbol{u})\right] & i=5,8 \end{cases} \tag{57}$$

其中 g 为重力加速度。重构得到的宏观量为水床高度和质量通量

$$\sum_i f_i^{eq} = h, \quad \sum_i \boldsymbol{c}_i f_i^{eq} = h\boldsymbol{u} \tag{58}$$

平床流动对应的宏观方程为

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\boldsymbol{u}) = 0 \tag{59}$$

$$\frac{\partial (h\boldsymbol{u})}{\partial t} + \nabla \cdot (h\boldsymbol{u}\boldsymbol{u}) = -g\nabla\left(\frac{h^2}{2}\right) + \nu\nabla \cdot (h\nabla\boldsymbol{u}) \tag{60}$$

可以添加额外的强迫项，以考虑床面高程的变化。

该系统可以线性化，以研究弱表面波动力学。引入以下脉动变量:

$$\sum_i f_i^{eq} = h', \quad \sum_i \boldsymbol{c}_i f_i^{eq} = (h\boldsymbol{u})' = h\boldsymbol{u} - h_0\boldsymbol{u}_0 \sim h_0\boldsymbol{u}' + h'\boldsymbol{u}_0 \tag{61}$$

其中， h_0 和 $\boldsymbol{u}_0$ 为基流物理量，有

$$f_i^{eq} = \begin{cases} \left(1 - \dfrac{5gh_0}{c_s^2} + \dfrac{2}{c_s^2}(\boldsymbol{u}_0 \cdot \boldsymbol{u}_0)\right) h' - \dfrac{4}{c_s^2}\boldsymbol{u}_0 \cdot (h\boldsymbol{u})' & i = 0 \\ \left(\dfrac{gh_0}{c_s^2} - \dfrac{9}{2c_s^4}(\boldsymbol{c}_i \cdot \boldsymbol{u}_0)^2 - \dfrac{1}{2c_s^2}(\boldsymbol{u}_0 \cdot \boldsymbol{u}_0)\right) h' & i = 1, 4 \\ + \left(\dfrac{\boldsymbol{c}_i + \boldsymbol{u}_0}{c_s^2} + \dfrac{9(\boldsymbol{c}_i \cdot \boldsymbol{u}_0)}{c_s^4}\boldsymbol{c}_i\right) \cdot (h\boldsymbol{u})' & \\ \left(\dfrac{gh_0}{c_s^2} - \dfrac{9}{2c_s^4}(\boldsymbol{c}_i \cdot \boldsymbol{u}_0)^2 - \dfrac{1}{2c_s^2}(\boldsymbol{u}_0 \cdot \boldsymbol{u}_0)\right) \dfrac{h'}{4} & i = 5, 8 \\ + \dfrac{1}{4}\left(\dfrac{\boldsymbol{c}_i + \boldsymbol{u}_0}{c_s^2} + \dfrac{9(\boldsymbol{c}_i \cdot \boldsymbol{u}_0)}{c_s^4}\boldsymbol{c}_i\right) \cdot (h\boldsymbol{u})' & \end{cases} \tag{62}$$

相应的宏观方程组为

$$\frac{\partial h'}{\partial t} + \nabla \cdot (h\boldsymbol{u})' = 0 \tag{63}$$

$$\frac{\partial (h\boldsymbol{u})'}{\partial t} + \nabla \cdot ((h\boldsymbol{u})'\boldsymbol{u}_0 + (h\boldsymbol{u})_0\boldsymbol{u}') = -g\nabla h' + \nu\nabla \cdot (h'\nabla\boldsymbol{u}_0 + h_0\nabla\boldsymbol{u}'). \tag{64}$$

附录 B. 量子态的函数层析

在此，我们介绍并描述一种执行量子态层析的方法，该方法是高效的，前

提是编码于量子振幅中的函数满足某些稀疏性假设。设 $|\psi\rangle$ 为在量子计算机上制备的 n 量子比特波函数。我们将其写为:

$$|\psi\rangle = \sum_{i=0}^{2^n-1} f(x_i)|i\rangle, \tag{65}$$

其中 $|i\rangle$ 是第 i 个计算基态，而 x_i 是某个函数 $f(x)$ 的离散化定义域中的第 i 个网格点。例如，如果 f 定义在区间 $[0,1]$ 上，我们可以令 $x_i = i/2^n$。这里，为简单起见，我们假设 f 是一个一维函数，但该方法可以直接推广到更高维。

现在我们假设 $f(x)$ 可以表示为以下有限和

$$f(x) = \sum_{j=1}^{m} a_j g_j(x), \tag{66}$$

其中， $g_j(x)$ 是实函数的某个预先确定的（不一定正交的）函数基， a_j 也为实数。关键是，我们进一步假设其具有稀疏性，即 $m \ll 2^n$。不对函数 $g_j(x)$ 施加其他假设。然而，为使量子态得到正确归一化，振幅向量 $f(x)$ 必须具有单位范数。该条件可写为：

$$W(a) = \sum_j \sum_k a_j G_{jk} a_k = 1, \tag{67}$$

其中，$G_{j,\,k}$ 是 g_j 和 g_k 在离散化定义域上的内积。选择正交归一函数 g_j 可将表达式简化为 $W(a) = \sum_{j=1}^{m} a_j^2$。

函数层析的目标是，仅通过对量子态 $|\psi\rangle$ 进行投影测量，高效地求出系数 a_j 。

B.1. 恢复编码函数的绝对值。 首先考虑一个更简单的问题：执行函数层析，仅提取编码函数的绝对值 $|f(x)|$ 。当已知 $f(x) \geq 0$，或之后可以通过其他方式恢复符号时，这种方法便可使用。

设 $P(i)$ 为在计算（Z）基下测量 $|\psi$ 所得到的量子态概率分布函数。 $P(i)$ 等于观测到状态 $|i$ 的次数除以采样次数。设 a 为变分系数 a_j（当前值）构成的向量，并将其指定的函数 $g_j(x)$ 的线性组合记为 $f_a(x)$。目标是确定 a，使得 $f_a(x) \approx f(x)$。为此，根据 [119],，我们可以用两个分布函数之间的 Kullback-Leibler散度来计算 $P(i)$ 与用 $f_a(x)$ 对状态 $|i$ 采样所得概率之间的距离；等价地，也可以计算负对数似然：

$$L(a) = -\sum_i P(i) \log\left(\frac{|f_a(x_i)|^2}{W(a)P(i)}\right). \tag{68}$$

可以通过基于梯度的方法高效地最小化该损失函数：

$$\frac{\partial L}{\partial a_j} = -2\sum_i P(i) g_j(x_i)/f_a(x_i) + 2\sum_{k=1}^{m} a_k G_{j,k}/W(a), \tag{69}$$

其中，优化需满足约束 $W(a) = 1$，以确保解具有正确的量子范数。

请注意， i 式（68）中对计算基态的求和仅针对已测量的状态，因此损失函数的项数不会超过采样次数。这确保了即使在量子比特数量较多的系统中，也始终可以计算代价函数 n。

B.2. **恢复完整的实值编码函数。** 为了进行层析成像，以重建完整的实函数 $f(x$，包括振幅中的相对符号，我们将上述方法（其本身采用了与矩阵乘积态（MPS）层析成像类似的代价函数[116, 119]）与实态层析成像的性质结合起来 [70, 106, 107]。

要恢复完整状态，所有层析成像方法都必须在多个基下进行测量。实态层析方法 [70, 106, 107] 基于以下观察：如果已知振幅中编码的函数为实值，那么只需在数量随量子比特数线性增长的多个基下进行测量即可 n （而一般情况下所需的基数呈指数增长）。具体而言，除 Z 基测量外，只需在 n 个额外的基下进行测量，这些基仅需将其中一个量子比特旋转到局部 X 基即可获得。请注意，无法恢复整体符号，因为函数 $f(x)$ 和 $-f(x)$ 仅相差一个全局相位。

现在设 $P_k^{(X)}(i)$ 为从量子态 $|\psi\rangle$ 测得的概率分布，其中第 k 个量子比特（从 0 开始计数）处于 X 基。随后，我们将上述实态层析成像方法调整为适用于函数层析的任务，即确定固定有限函数基 g_j 中的系数，并在代价函数（$0 \le k < n$）中引入 n 个附加项：

$$L_k^{(X)}(a) = -\sum_i P_k^{(X)}(i) \log\left(\frac{|f_a(x_i) \pm f_a(x_{i+2^k \mod 2^n})|^2}{2W(a)P_k^{(X)}(i)}\right), \tag{70}$$

其中，正负号分别由第 k 次测量结果为 0 或 1 决定。

值得注意的是，如上所述的函数层析（在函数基中确定系数）适用于某些特殊的基函数集合，但在技术和实践上存在一个关键差异。具体而言，当基 g_j 被假定为傅里叶 [55]，或切比雪夫多项式 [108] 时，已有方法可用于提取系数 a_j。然而，与我们的方法不同，这些方法是在量子设备上直接测量，并且需要添加一个复杂的量子电路，在测量前作用于状态 $|\psi\rangle$。因此，在含噪声中型量子设备上，该量子电路在实践中过深，难以使用。另一方面，理想模拟器使用上述方法的结果表明，原则上，这种函数层析能够相当准确地表示各种复杂的流体流动及其状态，尽管其结果只能表示为有限个函数之和。

最后，为了处理物体存在导致函数可能出现的不连续性，在对流场进行层析成像时，可以额外进行网格变换。这或许能减少层析成像所需的基函数数量，从而简化变分拟合。具体而言，在第 6 节的实验中，使用了一个尺寸为 $2a \times 2b$ 的矩形物体，其中心位于坐标系原点。通过使用映射 $(x,\ y) \mapsto (x,\ y) \cdot \max(r-1,\ 0)/r$，其中 $r = \max\{|x/a|,\ |y/b|\}$，物体周围的坐标系网格经过了变换，物体本身则被映射为一个点（另见 [72]）。数值验证表明，这种简单的变换能够减少层析成像所需的基函数数量。对于更复杂的物体和流场，则可以使用保角映射 [35]。