arXiv:2602.05187v1 [cs.LG] 5 Feb 2026

SpectraKAN：条件化谱算子

Chun-Wun Cheng [1] **Carola-Bibiane Sch¨onlieb** [1] **Angelica I. Aviles-Rivero** [2]

摘要

谱神经算子，尤其是傅里叶神经算子（FNO），凭借其在频域中的高效全局混合，成为学习偏微分方程（PDE）解算子的强大框架。然而，现有谱算子依赖于对所有输入统一应用的静态傅里叶核，限制了其捕捉由系统全局状态支配的多尺度、状态依赖和各向异性动力学的能力。我们提出了**SpectraKAN**，这是一种根据输入本身对谱算子进行条件化的神经算子，将静态谱卷积转变为输入条件积分算子。具体而言，我们从时空历史中提取紧凑的全局表示，并利用单查询交叉注意力对多尺度傅里叶主干网络进行调制，使该算子能够调整自身行为，同时保留谱混合的效率。我们提供了理论依据，表明这种调制在网格细化下收敛于分辨率无关的连续算子，而 KAN 则实现了平滑且由利普希茨控制的全局调制。在多种偏微分方程基准测试中，SpectraKAN 达到了最先进的性能，相较于强基线方法，均方根误差最多降低了 49%；在具有挑战性的时空预测任务中，性能提升尤为显著。

1. 引言

偏微分方程（PDE）是大量科学与工程计算的基础，支配着从扩散–反应和波传播到地球物理流动与气候动力学等各种现象（Roub´ıˇcek，2005）。尽管经典数值求解器（例如有限差分、有限体积和有限元方法）具有较高精度且稳健，但在需要反复求解的情况下（例如设计优化、不确定性量化和逆问题），或者需要高分辨率和长时间跨度时，其计算成本可能高得令人难以承受。

神经算子（Lu 等人，2019；Li 等人，2020）近年来逐渐成为一种颇具吸引力的替代方案，通过从数据中学习无限维函数空间之间的映射，构建快速的偏微分方程代理求解器。训练完成后，神经算子能够以比传统求解器快几个数量级的速度计算解算子，同时保留离散化不变的泛化能力以及实现零样本超分辨率等关键优势。在这些方法中，傅里叶神经算子（Li 等人，2020）凭借通过谱卷积实现高效全局混合，已成为广泛采用的骨干架构。

尽管具有出色的实证表现，现有的基于傅里叶的算子在处理复杂时空系统时仍面临持续存在的挑战。首先，许多偏微分方程呈现多尺度结构和各向异性，而标准 FNO 模块依赖单分辨率谱表示和基本静态的滤波器，限制了其同时捕捉粗细尺度动力学及方向依赖行为的能力。其次，真实的偏微分方程动力学通常依赖于所处状态区间，全局状态和约束会影响局部演化；然而，大多数傅里叶算子在推理时采用相同的谱核，从而降低了适应性，并可能损害长时间自回归推演的稳定性。第三，由于非线性通常通过缺乏结构意义的固定逐点激活函数引入，可解释性和可控性仍然有限。

为缓解这些问题的部分症结，人们提出了众多扩展方案。多分辨率和多基算子学习器（Li 等人，2024a；Gupta 等人，2021；Roy 等人，2025）改善了尺度分离，而自适应／状态依赖核（Dang 等人）则通过注意力机制或超网络引入条件性。基于 Transformer 的神经算子（Wu 等人，2024；Hao 等人，2023）提供了灵活的非局部交互，而 Kolmogorov–Arnold 网络（KAN）（Liu 等人，2024）则以可学习的样条函数取代固定激活函数，从而有望提升可解释性。然而，这些方向往往各自独立发展：多尺度变体可能仍是全局静态的，自适应核可能只有单一尺度或计算开销较大，而

[1]剑桥大学应用数学与理论物理系，英国剑桥 [2]清华大学丘成桐数学科学中心，中国北京。通讯作者：Angelica I. Aviles-Rivero < aviles-rivero@tsinghua.edu.cn>。

预印本

KAN 分量常被用作局部非线性函数，却没有明确控制算子层面的谱行为或全局条件化。因此，仍缺少一种统一的算子，能够同时满足以下要求：(i) 具备多尺度特性并感知各向异性；(ii) 以依赖状态的方式进行全局条件化；(iii) 高效且稳定地用于时空偏微分方程预测。

我们通过**SpectraKAN**弥补这一不足。SpectraKAN 是一种**谱 KAN 注意力神经算子（SKAN）**，可将静态傅里叶算子转变为动态、具备上下文感知能力的谱算子。SpectraKAN 结合了三个分量：（*i*）自适应多尺度 *FNO*（*AMFNO*），通过融合粗、细谱分支（并采用方向门控），更好地捕捉长程耦合，同时减轻傅里叶截断偏差；（*ii*）全局条件 *KAN*（*G-KAN*），将时空历史（以及可选参数）压缩为紧凑且可由样条解释的全局令牌；以及（*iii*）全局调制层（*GML*），通过单查询交叉注意力机制注入该令牌。关键在于，SpectraKAN 并非仅仅改进 FNO 模块中的傅里叶卷积核，而是引入了一个额外的输入条件化核积分算子，该算子通过 KAN 令牌以整个输入为条件。SpectraKAN 可视为 $G(u)(x) = f(u)(x) + A_{\Phi_{\mathrm{KAN}}(u)}[f(u)](x)$，其中 $f(\cdot)$ 是（多尺度）傅里叶主干网络，而 $\mathrm{A}^{\Phi\mathrm{KAN}(u)}$ 是一个非局部积分算子，其核取决于 KAN 生成的全局上下文。这种分离改变了有效算子族，使其从纯粹平稳的谱卷积转变为全局条件化、非平稳的积分核算子，使模型能够在不同状态下调整其全局动力学，同时保持空间骨干网络的高效性。

我们的主要贡献如下。

- 我们发现，现有傅里叶神经算子存在一个根本性局限：使用静态谱核并将其统一应用于所有输入，从而限制了对依赖不同状态区域及多尺度偏微分方程动力学的建模能力。
- 我们提出了**SpectraKAN**，通过为傅里叶主干网络增添输入条件化核积分算子，重新构建谱算子学习框架，从而将有效算子类从平稳谱卷积转变为全局条件化的非平稳算子。
- 我们提出了一种通过 Kolmogorov–Arnold 网络实现全局条件化的原则性机制：从时空历史中提取紧凑且可用样条解释的词元，并通过单查询注意力调制谱算子；结果表明，KAN 能够提供平滑且受利普希茨约束的全局调制。
- 我们提供了理论分析，表明这种基于注意力的调制是一种一致的求积近似，可用于连续积分算子，并且在网格细化时收敛到一个与分辨率无关的算子 。
- 我们在各种偏微分方程基准测试中证明，对谱算子进行条件化，相较于强大的基于傅里叶、Transformer 和 KAN 的基线方法，可带来显著的性能提升，尤其是在具有挑战性的时空预测和多尺度预测任务上。

2. 相关工作

用于偏微分方程的深度学习 神经算子已成为偏微分方程代理模型的强大框架，能够学习函数空间之间的映射，并实现与离散化无关的泛化。DeepONet（Lu 等，2019）通过耦合的分支—主干网络对非线性算子进行参数化，而 FNO（Li 等，2020）则在傅里叶域中学习积分核，并在包括零样本超分辨率在内的经典基准任务上表现出色。在此基础上，研究者提出了许多扩展方法，以提升表达能力和效率，包括：(i) 多分辨率或多基算子，以增强多尺度建模（Kossaifi 等，2023；You 等，2024；Guo 和 Li，2024）；(ii) 因式分解架构和 U 形架构，以加深网络并提高计算效率（Wen 等，2022；Rahman 等，2022；Li 等，2023；Wang 等，2025）；以及 (iii) 物理信息神经算子，通过纳入偏微分方程残差约束来提升数据效率和分布外泛化能力（Li 等，2024b；Ding 等，2025）。

基于 Transformer 的神经算子。基于 Transformer 的神经算子（Vaswani 等人，2017）利用注意力机制对时变偏微分方程中的长程依赖关系进行建模，其变体包括 Galerkin Transformer（Cao，2021）通过线性复杂度注意力提高效率，OFormer（Li 等人，2022）和 GNOT（Hao 等人，2023）则将该框架扩展到采样函数、非规则网格和多输入设置，同时若干变体进一步提升了准确率或效率（Wu 等人，2024；Bryutkin 等人，2024；Wang 等人，2024）。近年来，Mamba（Cheng 等人，2025；Hu 等）已成为一种高效的替代方案，并被应用于算子学习。与依赖完整的逐 token 自注意力作为核心算子不同，我们的方法采用由基于 KAN 的全局 token 驱动的单查询交叉注意力机制，对多尺度门控傅里叶主干进行调制，从而得到一种输入条件化积分算子，以全局方式控制空间动力学。

KAN。Kolmogorov–Arnold 网络（KAN）采用基于 B-样条的可学习激活函数，以提升准确性和可解释性。KANO（Lee 等人，2025）提出了一种受可解释性启发、基于 KAN 的神经算子，而物理信息 KAN 则将 KAN 集成到物理信息神经网络中，以提

高准确性和泛化能力（Zhang 等人，2025）。其他研究将 KAN 引入偏微分方程求解器以提升效率（Yeo 等人，2024；Abueidda 等人，2025）。现有方法主要将 KAN 嵌入神经算子或替换多层感知机组件；相比之下，我们首次使用 KAN 学习全局条件令牌。

3. 方法

问题陈述。许多偏微分方程的解都是随时间变化、在空间中分布的场。我们将偏微分方程的解表示为向量值函数 $v : T \times S \times \Theta \to \mathbb{R}^d$，其中 $S \subset \mathbb{R}^p$ 是空间域，T 是时间索引集，Θ 表示编码物理系数或边界条件/初始条件的参数空间。对于每个时间 t，我们将 $v_\theta(t, \cdot)$ 视为定义在 S 上的函数空间 X 中的一个元素（例如，$X = L^2(S; \mathbb{R}^d)$）。对于固定参数 $\theta \in \Theta$，偏微分方程解的演化由前向传播算子 $F_\theta : X \to X$，$v_\theta(t+1, \cdot) = F_\theta(v_\theta(t, \cdot))$ 控制，它将当前状态映射到下一个状态。然而，由于偏微分方程的动力学取决于时间导数，单个状态快照通常不足以近似 F_θ。因此，我们使用由 ℓ 个先前状态组成的窗口。由此得到的离散时间算子为 $\hat{F}_\theta : (v_\theta(t-\ell+1, \cdot) \ldots, v_\theta(t, \cdot)) \mapsto v_\theta(t+1, \cdot)$。为方便起见，我们将输入窗口记为 $v_\theta([t-\ell+1 : t], \cdot)$。我们的目标是学习一个神经算子 $\widehat{\quad}$ $\widehat{\quad}$ 子 F，使得对于所有 $\theta \in \Theta$，$F \approx F_\theta$，从而能够根据过去的观测准确预测未来的偏微分方程状态，并推广到不同的物理配置。

3.1. 架构概述

我们的架构 SpectralKAN（SKAN）由三个关键分量组成。自适应多尺度 FNO（AMFNO）通过联合建模细尺度和粗尺度谱模态，捕捉从局部到全局的动态变化，在改善长程耦合的同时减少傅里叶截断偏差。全局条件 KAN（G-KAN）提取系统时间状态和参数状态的低维全局表示，作为空间算子的潜在控制器。全局调制层（GML）将 AMFNO 的空间特征与 G-KAN 的全局上下文融合，引入一种非局部、输入条件耦合，使谱主干转变为具备上下文感知能力的积分算子。图 1 展示了整体架构。数学上，我们将神经算子定义为：

$$G(u)(x) = \underbrace{f(u)}_{\text{multi-scale spectral operator}} + \underbrace{A_{\Phi_{\mathrm{KAN}}(u)}[f(u)](x)}_{\text{input-conditioned attention operator}} \tag{1}$$

其中，全局 KAN 令牌为 $\Phi_{\mathrm{KAN}}(u) = t$，而输入条件化的注意力算子定义为 $A_t[u](x) = \sum_{y \in D} \alpha(x, y; t)\, u(y)$，其注意力权重为 $\alpha(x, y; t) = \mathrm{softmax}(\frac{Q(t) \cdot K(f(y))}{d})$。

3.2. 自适应多尺度 FNO（AMFNO）

采用门控轴向频谱的谱卷积。每个傅里叶层都会沿两个空间轴分别执行可分离谱混合。设 $v \in \mathbb{R}^{X \times Y \times d}$ 是空间坐标域中的一个特征场，其中 X、Y 分别表示 x 轴和 y 轴上的空间网格点数，d 表示特征通道数。我们的目标是沿每个轴分别执行谱卷积，并使用门控权重动态调节频率响应。

首先，我们应用升维映射和输入嵌入。然后执行可分离一维傅里叶变换。我们先考虑 y 轴。对于每个固定的 x_i，沿着 v 的 Y 个样本逐通道进行一维离散傅里叶变换（DFT）：$\hat{v}_y(x_i, k_2, c) = \sum_{j=0}^{Y-1} v(x_i, y_j, c)\, e^{-2\pi i j k_2 / Y}, k_2 = 0, 1, \ldots, Y-1$，对每个通道 $c = 1, \ldots, d$ 均如此。随后得到一个复值张量 $\hat{v}_y \in \mathbb{C}^{X \times Y \times d}$。不过，我们仅保留最低的 m 个傅里叶模态，即 $k_2 = 0, \ldots, m-1$。因此，截断表示变为 $\hat{v}_y \in \mathbb{C}^{X \times m \times d}$。谱卷积针对每个频率模态，在特征通道之间执行复矩阵乘法。令 $\hat{v}_y(x_i, k_2) \in \mathbb{C}^d$ 表示模态 k_2 处的特征向量。则谱混合定义为 $\tilde{v}_y(x_i, k_2) = W^{(y)}(k_2)\, \hat{v}_y(x_i, k_2)$ $W^{(y)}(k_2) \in \mathbb{C}^{d \times d}$。每个 $W^{(y)}(k_2)$ 都是一个复值线性算子，用于学习该频率下 d 个通道之间的交互方式。

我们采用门控机制，使谱响应具有自适应性并感知频率；具体而言，将每个模态乘以一个可学习实值标量门控 $\gamma_y(k_2) \in \mathbb{R}$。因此，沿 y 方向进行的门控谱卷积变为：$\hat{v}'_y(x_i, k_2) = \gamma_y(k_2)\, W^{(y)}(k_2)\, \hat{v}_y(x_i, k_2)$ $k_2 = 0, 1, \ldots, Y-1$。等价地，按通道展开的分量形式为：$\hat{v}'_{y,\ell}(x_i, k_2) = \gamma_y(k_2) \sum_{j=1}^{d} W^{(y)}_{\ell j}(k_2)\, \hat{v}_{y,j}(x_i, k_2)$ $\ell = 1, \ldots, d$。由此得到以下方程：$\hat{v}_y(k_2) \mapsto \sum_{j=1}^{d} \hat{v}_{y,j}(k_2)\, W^{(y)}{}_{j \to \cdot}(k_2)\, \gamma_y(k_2)$ $k_2 = 0, 1, \ldots, Y-1$。这里，$W^{(y)}{}_{j \to \cdot}(k_2)$ 表示一个权重向量，用于将傅里叶模态 k_2 处的第 j 个输入通道映射到所有输出通道。最后，应用逆傅里叶变换。随后，我们对 x 方向重复整个过程。

随后，挤压—激励模块根据当前特征图 $v_{\mathrm{spec}} \in \mathbb{R}^{X \times Y \times d}$ 的全局统计信息进行自适应通道重新缩放。首先，通过平均池化执行压缩操作，为每个通道计算全局描述符：

$$s_c = \frac{1}{XY} \sum_{i=1}^{X} \sum_{j=1}^{Y} v_{\mathrm{spec}}(x_i, y_j, c), \qquad c = 1, \ldots, d. \tag{2}$$

这些聚合描述符 $s = [s_1, \ldots, s_d] \in \mathbb{R}^d$ 概括了每个通道的全局激活。随后，激励操作通过学习逐通道调制权重

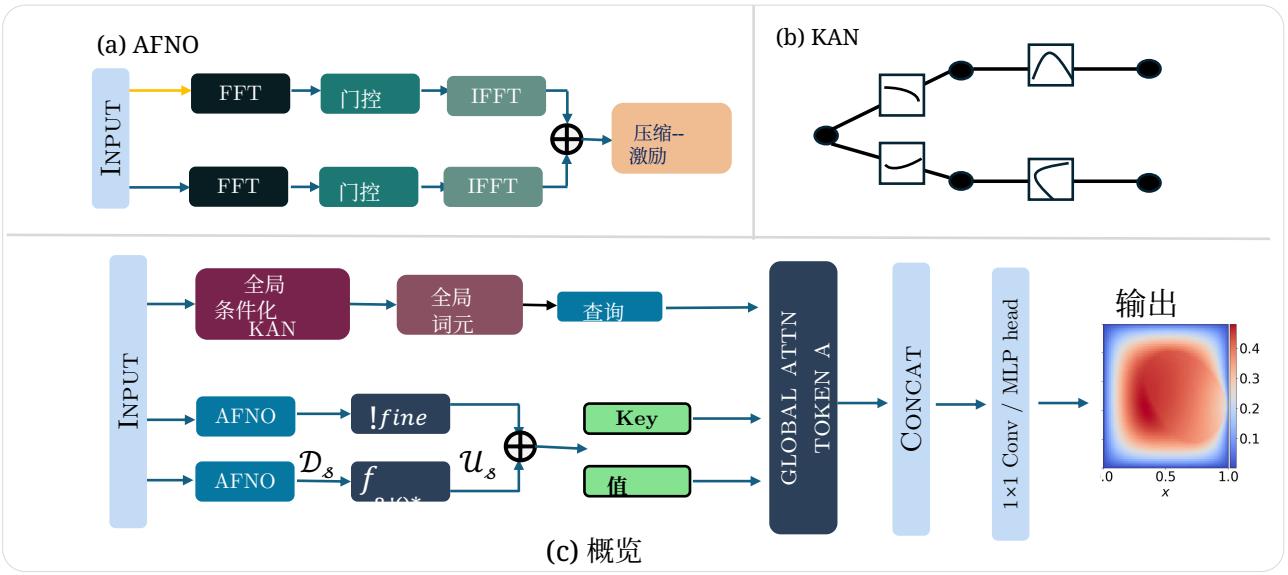

图 *1*。我们的 SpectraKAN架构概览。(a) AFNO 通过快速傅里叶变换–门控–快速傅里叶逆变换与挤压–激励实现谱混合。(b) KAN 使用可学习的样条函数对非线性交互进行建模。(c) 经全局条件化的 KAN 生成一个全局令牌（查询），并将其与多尺度AFNO特征（键/值）融合，随后通过轻量级投影得到预测结果。

由一个小型门控网络实现：

$$a = \sigma\big(W_2\,\delta(W_1\,s)\big), \qquad a \in (0,1)^d, \tag{3}$$

其中 $W_1 \in \mathbb{R}^{d\times\frac{d}{r}}$和 $W_2 \in \mathbb{R}^{\frac{d}{r}\times d}$是可学习参数， $\delta(\cdot)$ 是 ReLU激活函数， $\sigma(\cdot)$ 是 Sigmoid函数。最后，按通道对特征图重新加权：

$$\mathrm{SE}(v_{\mathrm{spec}})(x_i,y_j,c) = a_c \cdot v_{\mathrm{spec}}(x_i,y_j,c), \tag{4}$$

从而根据各通道的全局相关性，自适应地增强或抑制通道。

细尺度算子。为捕捉高分辨率局部动态，我们堆叠 L 个门控谱卷积层，每层由一个轴向傅里叶算子和一个逐点非线性函数组成。这定义了细尺度傅里叶神经算子 $\mathrm{F_{fine}} = \mathrm{K}_L \bigcirc \sigma \bigcirc \cdots \bigcirc \mathrm{K}_1$, ，其中每个 K_ℓ 表示一个作用于 $L^2(D;\mathbb{R}^d)$ 中场的门控谱积分算子。给定输入场 u 和空间网格 g，细尺度表示为

$$f_{\mathrm{fine}} = \mathcal{F}_{\mathrm{fine}}(u,g) \in \mathbb{R}^{X\times Y\times C_{\mathrm{fno}}}.$$

粗分支与融合。为了扩大感受野并捕获长程交互，我们引入第二个傅里叶算子，使其作用于粗粒度空间网格。设 D_s表示按因子 s 进行空间下采样，U_s 表示相应的上采样算子。粗尺度算子定义为如下

$$f_{\mathrm{coarse}} = \mathcal{U}_s(\mathcal{F}_{\mathrm{coarse}}(\mathcal{D}_s u, \mathcal{D}_s g)) \in \mathbb{R}^{X\times Y\times C_{\mathrm{fno}}}. \tag{5}$$

最终的多尺度表示通过加法融合得到：

$$f(u) = f_{\mathrm{fine}} + f_{\mathrm{coarse}}. \tag{6}$$

AMFNO 以沿各轴可分离的傅里叶算子取代联合谱混合，从而降低内存用量并提升计算效率。这种设计对各向异性偏微分方程（例如对流–扩散、纳维–斯托克斯方程、热输运）尤其有效，因为不同空间方向上的动态各不相同。在傅里叶域中，这种各向异性表现为能量分布随方向而异，分布在 $(k_x,\ k_y)$ 上。通过为每个轴分别学习谱算子，AMFNO 能够进行方向特定的频率建模，这对于捕捉各向异性湍流、剪切流和结构化扩散至关重要。

3.3. 全局条件 KAN

基于 KAN 的全局令牌条件化。 为了实现全局且由输入条件化的空间算子调制，我们引入一个基于 KAN 的全局令牌，将时间、参数和系统级上下文编码为紧凑的潜在表示。该令牌充当条件变量，通过注意力机制控制全局信息如何注入空间动力学。

全局上下文提取。令 $u_{0:\tau_{\text{in}}} \in \mathbb{R}^{X\times Y\times \tau_{\text{in}}C}$ 表示偏微分方程解近期的时空历史。我们首先提取一个紧凑的全局摘要 $z = \mathrm{P}(u_{0:\tau_{\text{in}}}) \in \mathbb{R}^{d_0}$, $d_0 = \tau_{\text{in}}C$，其中 P 表示池化或采样算子（例如中心像素提取或空间平均）。该向量捕捉粗粒度的时间趋势和全局状态信息，这些信息会影响整个定义域内解的演化。

Kolmogorov–Arnold 网络（KAN）编码器。我们使用 Kolmogorov–Arnold 网络（KAN）将全局摘要 z 映射为低维令牌 $t \in \mathbb{R}^{C_{\text{kan}}}$。该网络能够将多变量函数表示为单变量函数之和的复合。深度为 L、形状为$[d_0, d_1, \ldots, d_L]$的 KAN 定义了映射

$$t = \Phi_{\text{KAN}}(z) = (\Phi_{L-1} \circ \cdots \circ \Phi_0)(z), \qquad t \in \mathbb{R}^{d_L}.$$

每个 KAN 层 $\Phi_\ell : \mathbb{R}^{d_\ell} \to \mathbb{R}^{d_{\ell+1}}$ 按分量定义为

$$x_j^{(\ell+1)} = \sum_{i=1}^{d_\ell} \phi_{j,i}^{(\ell)}\left(x_i^{(\ell)}\right), \qquad j = 1, \ldots, d_{\ell+1}, \tag{7}$$

其中 $\phi_{j,i}^{(\ell)} : \mathbb{R} \to \mathbb{R}$ 是可学习的单变量边函数。

基于样条的边参数化。遵循标准 KAN 表述，每个边函数 $\phi_{j,i}^{(\ell)}$ 均参数化为残差 B 样条展开：

$$\phi_{j,i}^{(\ell)}(x) = w_{b,j,i}^{(\ell)}\, b(x) + w_{s,j,i}^{(\ell)} \sum_{r=1}^{G+k} c_{j,i,r}^{(\ell)}\, B_r^{(k)}(x; \Xi_i^{(\ell)}), \tag{8}$$

其中：$b(x) = \mathrm{SiLU}(x)$ 是固定的平滑基础函数，$B_r^{(k)}(\cdot; \Xi_i^{(\ell)})$ 是定义在可学习节点网格 $\Xi_i^{(\ell)}$ 上的 k 阶 B 样条基函数，$c_{j,i}^{(\ell)}$、、r 是样条系数，$w_{b,j,i}^{(\ell)}$，$w_{s,j,i}^{(\ell)}$ 是可学习的缩放参数。

这一构造得到了光滑的分段多项式表示，并能显式控制正则性和近似能力。

近似与可解释性。根据经典样条近似理论，式（8）中的 B 样条展开可以用误差 $f \in C^{k+1}([a,b])$ 近似任意单变量函数

$$\left\| f - \sum_r c_r B_r^{(k)}(\cdot; \Xi) \right\|_\infty = \mathcal{O}(h^{k+1}),$$

其中 h 表示节点间距。因此，Φ_{KAN}提供了一种平滑、可解释且低秩的全局动力学信息嵌入，而学习得到的节点位置 Ξ则能直观揭示输入中哪些状态最为相关。

全局令牌作为算子条件。词元 $t = \Phi_{\text{KAN}}(z)$不会直接预测空间场，而是充当空间算子的条件变量。具体而言，t 在交叉注意力机制模块中对查询进行参数化，从而定义一个由输入条件化的积分核

$$\kappa_t(x, y) = \mathrm{softmax}_y\left(\frac{Q(t)\, K(f(y))^\top}{\sqrt{d_a}}\right),$$

它作用于多尺度傅里叶特征。该机制将全局时间上下文和参数上下文注入所有空间位置，使算子能够适应系统状态。这种全局 KAN 条件化与多尺度 FNO 的平移不变谱核相辅相成：FNO 捕捉平稳空间结构，而 KAN 令牌则以数据依赖的方式调制算子，从而实现非平稳动态、状态变化以及仅靠谱卷积无法表达的全局约束。

3.4. 全局调制层

将全局调制层（GML）视为输入条件化积分算子。设 $D \subset \mathbb{R}^2$ 表示空间域，$f(u) : D \to \mathbb{R}^{C_{\text{fno}}}$ 表示 AMFNO 生成的多尺度傅里叶特征场，$t = \Phi_{\text{KAN}}(u) \in \mathbb{R}^{C_{\text{kan}}}$ 表示 G-KAN 生成的全局令牌。GML 的作用是通过构造一个作用于该特征场的输入条件化非局部算子，将全局上下文 t 与局部傅里叶特征 $f(u)$ 融合。我们将基于注意力机制的组件定义为核积分算子

$$(A_t[u])(x) = \int_D \kappa_t(x, y)\, \mathcal{V}\big(f(u)(y)\big)\, dy, \qquad x \in D, \tag{9}$$

其中，$\mathcal{V} : \mathbb{R}^{C_{\text{fno}}} \to \mathbb{R}^{d_a}$ 是值投影（由线性映射实现），而 $\kappa_t(x, y)$ 是由交叉注意力机制诱导的输入依赖核。注意力核定义为归一化指数相似度：

$$\kappa_t(x, y) = \frac{\exp\left(\frac{\langle q(\Phi_{\text{KAN}}(t)), k(y)\rangle}{\sqrt{d_a}}\right)}{\int_D \exp\left(\frac{\langle q(\Phi_{\text{KAN}}(t)), k(\xi)\rangle}{\sqrt{d_a}}\right) d\xi}. \tag{10}$$

在网格 $\{y_j\}_{j=1}^N$ 上，$N = XY$式（9）中的积分通过求积（黎曼和）近似为：

$$\begin{aligned}(A_t[u])(x_i) &\approx \sum_{j=1}^{N} \alpha_{ij}(t)\, \mathcal{V}\big(f(u)(y_j)\big),\\ \alpha_{ij}(t) &= \frac{\exp\left(\frac{\langle q(t), k(y_j)\rangle}{\sqrt{d_a}}\right)}{\sum_{\ell=1}^{N} \exp\left(\frac{\langle q(t), k(y_\ell)\rangle}{\sqrt{d_a}}\right)}.\end{aligned} \tag{11}$$

这与标准的 $\mathrm{softmax}(QK^\top)V$注意力机制相符。

广播融合与逐点读出。式（9）中的全局注意力输出会广播回所有空间位置，并通过逐点读出映射 $\mathcal{R}$ (a 1×1 卷积多层感知机）：$\hat{}u(x) = \mathcal{R}(f(u)(x) \quad A_t[u](x))$ $x \in D$融合局部傅里叶特征，从而得到最终的上下文感知预测场。

结合 AMFNO 和 GML，SKAN 神经算子可概括为

$$G(u)(x) = f(u)(x) + A_{\Phi_{\mathrm{KAN}}(u)}[f(u)](x), \qquad x \in D, \tag{12}$$

其中$A_{\Phi_{\mathrm{KAN}}(u)}$ 是由式（9）–（10）定义的输入条件化积分算子。

3.5. 理论基础

本节将介绍我们方法的理论基础，包括其分辨率不变性以及全局KAN令牌的选择。我们还将在附录A中证明，SpectraKAN 定义了函数空间之间的神经算子。

引理 *3.1*（样条 *KAN* 的边函数利普希茨界）。考虑形式为 $\phi(x) = w_b\, b(x) + w_s \sum_r c_r\, B_r^{(k)}(x;\Xi)$ 的边函数，其中 $b \in C^1$，且 $B_r^{(k)}$ 是定义在节点网格 $\Xi = \{t_j\}$ 上的 k 阶 B 样条，最小节点间距为 $h_{\min} := \min_j(t_{j+1} - t_j) > 0$。假设 $\sup_x |b'(x)| \le L_b$。那么，ϕ 是 *Lipschitz* 连续的，其利普希茨常数满足 $\mathrm{Lip}(\phi) \le |w_b| L_b + |w_s| \frac{2}{h_{\min}} \|c\|_1$。

完整证明见附录 A。我们强调，引入它并非 为了提升神经算子的空间表达能力。相反，KAN 将一个低维的全局调制 通路参数化，其敏感度可以得到显式控制。引理 3.1 给出了一个界，并提供了一种可检查的全局条件化正则化机制（例如，通过 $\|c\|_1$ 或节点间距），从而确保全局令牌随输入平滑变化。结合我们的单查询注意力形式，得到的调制在空间上是均匀的（广播至x），因此无法引入新的空间自由度。相反，它充当一个稳定的利普希茨受控控制器，可根据不同状态自适应地调整谱主干，从而提升长时域滚动稳定性。

随后，我们进一步证明，KAN 能够实现平滑且受利普希茨约束的全局调制。

定理 3.2（光滑、利普希茨控制的全局调制）。假设有界查询/键、*Lipschitz* 部分和 *KAN* 样条正则性。则全局调制映射$m : L^\infty(\mathcal{D}) \to \mathbb{R}^{d_m}$ 是 *Lipschitz* 连续的，满足$\|m(u) - m(\tilde{u})\|_2 \le L_m \|u - \tilde{u}\|_{L^\infty}$，其中 L_m 是一个显式常数，依赖于 L_p、L_{KAN}、L_k、L_v、M_q、M_k、$|\mathcal{D}|$，以及指数“温度”缩放 $1/\sqrt{d_k}$。特别地，减小 L_{KAN}（例如通过样条导数控制）会直接降低 L_m。

完整证明见附录，定理3.2表明 KAN 是一种光滑的全局控制器：它生成一个低维令牌，以连续且灵敏度有界的方式改变全局调制，这有利于稳定的自回归偏微分方程推演。

命题 3.3。（非正式表述）离散注意力输出A_h 收敛到连续输出 A，当 $h \to 0$时，对于足够小的 h，$\|A_h - A\| \le \frac{\sqrt{d}\,h}{Z}\big(L_g + \|A\| L_w\big)$。特别地，当网格逐渐细化时，$\|A_h - A\| = O(h)$。

L_g 衡量被积函数在空间中的变化速度，而L_w 衡量注意力权重在空间中的变化速度。离散 softmax 注意力聚合是连续归一化核积分算子的相容求积近似。命题3.3表明，随着网格细化，离散 GML 收敛到与分辨率无关的连续算子，且具有可证明的误差率。

4. 实验

本节介绍对我们方法的实验评估。我们首先介绍数据集和实现细节，随后与具有代表性的基线方法进行定量比较，并开展消融研究。

数据集、基线方法与训练细节。我们在多种不同的偏微分方程上评估所提出的方法。在自回归训练中，我们选用 PDEBench 中的数据集，涵盖一维 CNS、一维扩散反应、二维 Darcy 流和二维浅水。除 PDEBench（Takamoto 等，2022）外，我们还使用了 (Kissas 等，2022) 中的二维气候建模和浅水方程。数据集及预处理流程的详细说明见附录 B.1。我们将基线方法与以下几类方法进行了比较：(1) FNO 及其变体：FNO（Li 等，2020）、FFNO（Tran 等，2021）；(2) 基于 Transformer 的方法：Transovler（Wu 等，2024）、CViT（Wang 等，2024）；(3) 基于 KAN 的方法：DeepOKAN（Abueidda 等，2025）；(4) 其他方法：UNet（Ronneberger 等，2015）、LNO（Cao 等，2024）、局部傅里叶神经算子（Liu-Schiaffini 等，2024）。所有方法均在单张 NVIDIA A100 40GB GPU 上训练。

主要结果。我们将方法与具有代表性的基线方法进行了全面比较。结果汇总于表 1中，其中报告了均方根误差。此外，表 2报告了相对 L^2。

总体而言，我们的方法在表格中的全部五个基准测试上都取得了最佳表现1，表明其相较于近期实力强劲的基线方法，准确率始终有所提升。尤其是在一维可压缩纳维–斯托克斯（CNS）任务中，我们的方法将均方根误差从0.29493（FFNO）降至 0.*14544*，相当于，对

表1. 通过不同数据集比较我们的方法与其他有代表性的基线方法。性能最佳的结果以绿色标示，次佳结果以蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）	气候建模（2D）
U-Net（2015）	12.56934	0.03812	0.01315	0.11974	1.01260
FNO（2021）	0.34053	0.02884	0.02575	0.01448	0.01536
FFNO（2023）	0.29493	0.01409	0.02482	0.01191	–
LNO（2024）	8.97545	0.04703	0.03538	0.03938	-
Transolver（2024）	0.72094	0.00686	0.01987	0.00586	0.18575
LocalFNO（2024）	–	–	0.02051	0.00438	0.01508
DeepOKAN（2025）	9.21355	0.04387	0.01750	0.08618	0.036257
CViT（2025）	0.62194	0.00678	0.02287	0.01178	0.024066
SpectraKAN（本文方法）	**0.14544**	**0.00397**	**0.01260**	**0.0024**	**0.0096119**
提升（%）	49%	42%	4.18%	45.21%	36.26%

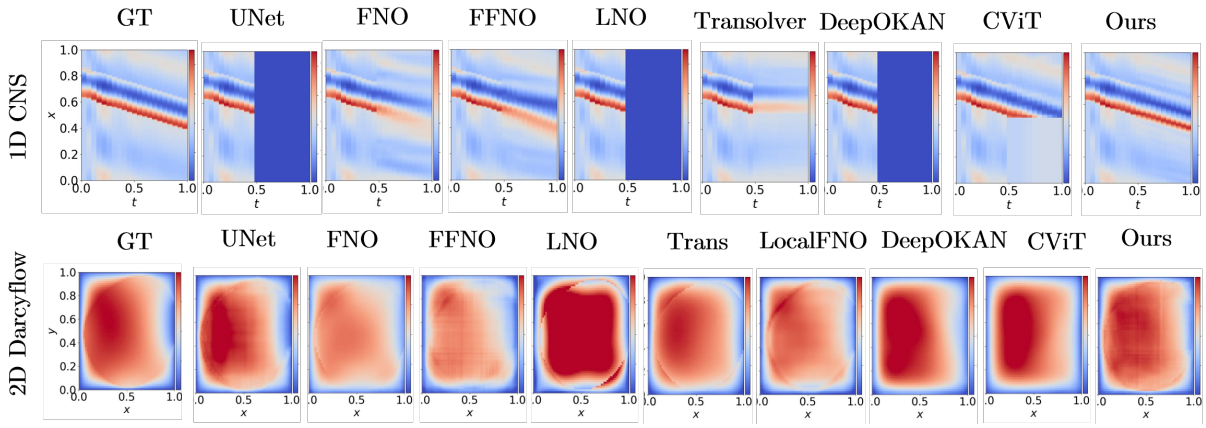

图2. 上：一维可压缩纳维-斯托克斯方程解的演化（时空图）。下：二维 Darcy 流解的快照。虽然几种基线方法能够恢复粗略结构，但往往会出现平滑处理或伪影；在这两种设置下，我们的方法所生成的预测都最接近真实值。

表2. 浅水方程 L^2 在不同条件下的相对误差。最佳结果以绿色突出显示

方法	ρ	u	v
U-Net	1.20×10^{-2}	3.62×10^{-1}	3.63×10^{-1}
FNO	1.15×10^{-3}	3.20×10^{-2}	3.33×10^{-2}
LNO	1.85×10^{-3}	6.77×10^{-2}	7.14×10^{-2}
Trasnolver	2.41×10^{-2}	1.00×10^{0}	1.00×10^{0}
DeepOKAN	2.35×10^{-2}	1.01×10^{0}	2.88×10^{-2}
CViT	2.38×10^{-2}	1.00×10^{0}	1.00×10^{0}
SpectraKAN（我们的方法）	$\mathbf{6.85 \times 10^{-4}}$	$\mathbf{1.91 \times 10^{-2}}$	$\mathbf{2.04 \times 10^{-2}}$

相对提升了 50.7%。对于一维扩散–反应系统，我们取得了 0.00397 的均方根误差，较此前最佳结果（0.00686）提升了 42.1%。在多种方法已达到较低误差的二维 Darcy 流基准测试中，我们的模型仍实现了稳定提升，误差达到 0.01260，而对比值为 0.01315。值得注意的是，我们取得的最大提升出现在具有挑战性的时空预测任务。在二维浅水基准上，我们的方法达到 0.0024 均方根误差，较表现最佳的竞争基线降低了 45.2% 。同样，对于二维气候建模，我们的方法取得了0.0096119 均方根误差，优于最强基线（0.01508），提升了36.3%。这些结果表明，我们的方法不仅在高精度领域（例如达西流）具有竞争力，而且尤其擅长捕捉时变系统中的复杂动态。更多数值结果见附录 B.2。

表 2 报告了其他浅水方程的结果。总体而言，我们的方法在所有分量上都取得了最低误差，始终优于其他所有基线方法。具体而言，我们的方法在 6.85×10^{-4} 上的 ρ误差为，比最强基线 FNO 提高了 $\approx 40.4\%$。对于速度场 u 和 v，也观察到了类似的提升。我们在 v 上取得了最佳性能，误差为 2.04×10^{-2}，相较于 $\approx 29.2\%$，在 v（DeepOKAN，2.88×10^{-2}）这一最强基线上的误差降低了。

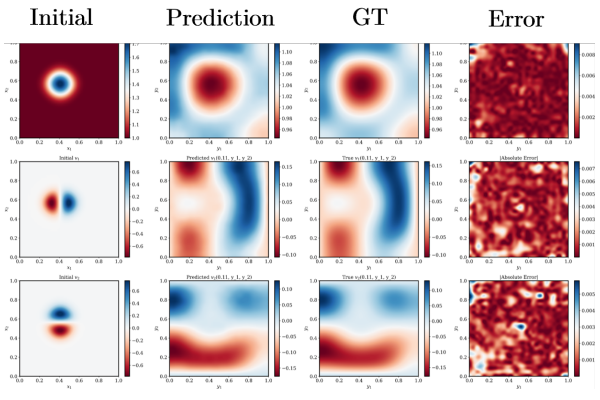

图 3。 浅水基准上的定性比较。展示了初始条件、预测解和真实解，以及 t=0.16s 时的绝对误差。

表 3。关于架构设计、词元构造和分辨率缩放的消融研究。最佳结果以绿色突出显示

方法	误差 64^2	128^2	32^2
（a）二维 Darcy 流上的架构消融			
AMFNO（基线）	0.01756	–	–
+ KAN（无 GML）	0.01327	–	–
+ KAN + GML（我们的方法）	**0.01260**	–	–
（b）二维 Darcy 流上的全局令牌条件化			
MLP	0.01639	–	–
数据（中心）	0.01616	–	–
数据（均值）	0.01693	–	–
KAN（我们的方法）	**0.0126**	–	–
（c）二维浅水超分辨率			
FNO	0.01448	0.14369	0.17709
Transformer	0.00586	0.08827	0.09834
局部傅里叶神经算子	0.00438	3527.224	0.04693
SpectraKAN（我们的方法）	**0.00240**	**0.00458**	**0.0107**

图2比较了在一维 CNS 和二维 Darcy 流任务上的定性结果。虽然基线方法能够捕捉粗粒度动力学，但会出现平滑化或空间伪影。我们的方法生成的预测最接近真实值，保留了更清晰的结构和物理上一致的空间模式。图 3 展示了一个具有代表性的二维浅水结果，其中 $t = 0.11$ 处的预测准确匹配真实值的高度场和速度场，捕捉到了具有平滑空间变化的全局流动结构。更多定性结果见附录B.3。

消融研究。表3 探究了我们关键设计选择的作用及其在分辨率不变性方面的表现，包括：（i）整体架构，（ii）令牌构建模块，以及（iii）分辨率变化下的鲁棒性。

（a）架构消融。

从 AMFNO 基线出发，引入 KAN-基于全局条件化将误差从 0.01756 降至 0.01327（相对提升 24.4%）。将非 GML 混合替换为 GML 混合后，性能进一步提升至 0.0126（相较基线提升 28.2%）。这些消融结果表明二者角色分明：KAN 词元提供全局非平稳调制，而 GML 混合则增强场内的长程交互。

（b）全局令牌条件化。接下来，我们在固定其他所有组成部分的情况下，单独考察全局令牌条件化的效果。多层感知机令牌与简单的基于统计量的令牌（中心/均值）取得了相近的误差，这表明这些选择在捕捉物理场全局状态方面提供的归纳偏置有限。相比之下，基于 KAN 的令牌化显著提升了准确率，将误差降至 0.0126。这一差距表明，KAN 令牌的函数结构能够提供更有效的全局表示——实现更平滑且经过利普希茨正则化的调制——从而增强算子对全局上下文和场依赖动力学的建模能力。

（c）分辨率不变性。我们通过在 64^2 下训练，并分别在 128^2 下进行超分辨率评估、在 32^2下进行下采样评估，来考察跨分辨率泛化。尽管 FNO 和 Transovler 在训练分辨率下表现出色，但分辨率变化会使其性能显著下降；例如，在 128^2下，FNO 的误差从 0.01448 增加了近一个数量级，达到 0.14369。LocalFNO 在 64^2 下表现良好，但在放大时会变得数值不稳定，误差飙升至 3527.224，出现在 128^2 下。相比之下，我们的方法在 64^2下取得最低误差，并且在放大和缩小时均保持稳健，在 128^2 下仍能维持较高精度（0.00458），在 32^2 下亦然（0.0107）。

5. 结论

我们提出了 SpectraKAN，这是一种神经算子，通过显式的输入条件化积分算子扩展基于傅里叶的模型，将有效算子类从静态谱卷积转变为全局条件化的非平稳算子。我们证明，基于注意力机制的调制可作求积解释，并收敛到分辨率无关的连续算子；同时，KAN 路径提供了一种受利普希茨约束的机制，以实现稳定的全局条件化。在实证评估中，SpectraKAN 在各类偏微分方程基准测试中取得了最先进的表现，尤其在时空复杂度较高以及分辨率变化的情况下优势显著。这些结果表明，通过全局积分调制对谱算子进行条件化，是推动神经算子设计超越静态傅里叶核的一条有原则的路径。

致谢

CWC 的经费由瑞士国家科学基金会（SNSF）通过资助编号 20HW-1 220785 提供。CBS 感谢 Philip Leverhulme Prize、英国皇家学会沃尔夫森奖学金、EPSRC 高级职业生涯奖学金 EP/V029428/1、EPSRC 项目 EP/S026045/1 和 EP/T003553/1、EP/N014588/1、EP/T017961/1、Wellcome 创新者奖 215733/Z/19/Z 和 221633/Z/20/Z、欧盟地平线2020研究与创新计划通过玛丽·斯克沃多夫斯卡-居里资助协议 No. 777826（NoMADS）提供的支持，以及 Cantab Capital信息数学研究所和艾伦·图灵研究所的支持。AIAR 衷心感谢丘成桐数学科学中心和清华大学的支持。本研究还得到清华大学都市计划的资助。

影响声明

本文旨在推动机器学习领域的发展。我们的工作可能会带来许多社会影响，但我们认为没有哪项影响需要在此特别强调。

参考文献

Abueidda, D. W., Pantidis, P., and Mobasher, M. E. Deepokan：基于 Kolmogorov-Arnold 网络、用于力学问题的深度算子网络。《计算机在应用力学与工程中的方法》, 436: 117699, 2025. Bryutkin, A., Huang, J., Deng, Z., Yang, G., Sch¨onlieb, C.-B., and Aviles-Rivero, A. Hamlet：用于偏微分方程的图变换器神经算子。*arXiv* 预印本 *arXiv:2402.03541*, 2024. Cao, Q., Goswami, S., and Karniadakis, G. E. 用于求解微分方程的拉普拉斯神经算子。《自然·机器智能》, 6(6):631–640, 2024. Cao, S. 选择一种 Transformer：傅里叶还是 Galerkin。神经信息处理系统进展, 34: 24924–24940, 2021. Cheng, C.-W., Huang, J., Zhang, Y., Yang, G., Sch¨onlieb, C.-B., and Aviles-Rivero, A. Mamba 神经算子：谁更胜一筹？用于偏微分方程的 Transformer 与状态空间模型之比较。计算物理学杂志, pp. 114567, 2025. Dang, H. V., Nguyen, B. D., Nguyen, P. C., and Hy, T.-S. Adaptfno：采用动态谱模态和多尺度学习的自适应傅里叶神经算子，用于气候建模。Ding, S., Zhang, Z., Shi, G., Li, X., Gu, X., Xu, Y., Xie, H., Zhao, H., Shi, Y., and Liu, T. 面向非线性 Grad-Shafranov方程的物理信息神经算子学习。*arXiv*预印本 *arXiv:2511.19114*，2025。Guo, Z.-H. and Li, H.-B. MGFNO：用于参数化偏微分方程的多网格架构傅里叶神经算子。*arXiv*预印本 *arXiv:2407.08615*，2024。Gupta, G., Xiao, X., and Bogdan, P. 基于多小波的微分方程算子学习。《神经信息处理系统进展》，34：24048–24062, 2021。Hao, Z., Wang, Z., Su, H., Ying, C., Dong, Y., Liu, S., Cheng, Z., Song, J., and Zhu, J. GNOT：用于算子学习的通用神经算子Transformer。载于《国际机器学习会议》，第12556–12569页。PMLR，2023。Hu, Z., Cao, Q., Kawaguchi, K., and Karniadakis, G. E. DeepoMamba：用于时空PDE神经算子学习的状态空间模型。*SSRN 5149007*。Kissas, G., Seidman, J. H., Guilhoto, L. F., Preciado, V. M., Pappas, G. J., and Perdikaris, P. 利用耦合注意力学习算子。《机器学习研究期刊》，23(215)：1–63，2022。Kossaifi, J., Kovachki, N., Azizzadenesheli, K., and Anandkumar, A. 面向高分辨率偏微分方程的多网格张量化傅里叶神经算子。*arXiv*预印本 *arXiv:2310.00120*，2023。Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A. 神经算子：学习函数空间之间的映射及其在偏微分方程中的应用。《机器学习研究期刊》，24(89)：1–97，2023。Langley, P. 撰写机器学习论文。载于 Langley, P.（编），《第*17*届国际机器学习会议论文集（*ICML 2000*）》，第1207–1216页，Stanford, CA，2000。Morgan Kaufmann。Lee, J., Liu, Z., Yu, X., Wang, Y., Jeong, H., Niu, M. Y., and Zhang, Z. KANO：Kolmogorov-Arnold神经算子。*arXiv*预印本 *arXiv:2509.16825*，2025。Li, S., Yu, X., Xing, W., Kirby, R., Narayan, A., and Zhe, S. 傅里叶神经算子的多分辨率主动学习。载于国际人工智能与统计学会议，第2440–2448页。PMLR，2024a。Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. 用于参数化偏微分方程的傅里叶神经算子。*arXiv* 预印本 *arXiv:2010.08895*，2020。

Li, Z., Meidani, K., and Farimani, A. B. 用于偏微分方程算子学习的Transformer。*arXiv*预印本 *arXiv:2205.13671*，2022。

Li, Z., Peng, W., Yuan, Z., and Wang, J. 通过隐式 U-Net 增强型傅里叶神经算子对湍流进行长期预测。《流体物理》, 35(7), 2023. Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., Azizzadenesheli, K., and Anandkumar, A. 用于学习偏微分方程的物理信息神经算子。*ACM/IMS* 数据科学期刊, 1(3):1–27, 2024b.

Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halver- son, J., Solja˘ci´c, M., Hou, T. Y., and Tegmark, M. KAN：Kolmogorov-Arnold 网络。*arXiv* 预印本 *arXiv:2404.19756*, 2024.

Liu-Schiaffini, M., Berner, J., Bonev, B., Kurth, T., Aziz- zadenesheli, K., 和 Anandkumar, A. 采用局部积分核与微分核的神经算子。*arXiv* 预印本 *arXiv:2402.16845*，2024。Lu, L., Jin, P., 和 Karniadakis, G. E. DeepONet：基于算子通用逼近定理学习用于识别微分方程的非线性算子。*arXiv* 预印本 *arXiv:1910.03193*，2019。Rahman, M. A., Ross, Z. E., 和 Azizzadenesheli, K. U-NO：U 形神经算子。*arXiv* 预印本 *arXiv:2204.11127*，2022。Ronneberger, O., Fischer, P., 和 Brox, T. U-Net：用于生物医学图像分割的卷积网络。载于国际医学图像计算与计算机辅助干预会议，第 234–241 页。Springer，2015。Roub´ı˘cek, T. 带应用的非线性偏微分方程。Springer，2005。Roy, S., Bahmani, B., Kevrekidis, I. G., 和 Shields, M. D. 一种基于物理信息的多分辨率神经算子。*arXiv* 预印本 *arXiv:2510.23810*，2025。Takamoto, M., Praditia, T., Leiteritz, R., MacKinlay, D., Alesiani, F., Pfl¨uger, D., 和 Niepert, M. PDEBench：科学机器学习的综合基准。《神经信息处理系统进展》，35：1596–1611，2022。Tran, A., Mathews, A., Xie, L., 和 Ong, C. S. 因式分解傅里叶神经算子。*arXiv* 预印本 *arXiv:2111.13802*，2021。Vas wani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., 和 Polosukhin, I. 注意力就是你所需要的一切。《神经信息处理系统进展》，30，2017。

Wang, S., Seidman, J. H., Sankaran, S., Wang, H., Pappas, G. J., 和 Perdikaris, P. CViT：用于算子学习的连续视觉 Transformer。*arXiv* 预印本 *arXiv:2405.13998*，2024。

Wang, Y., Yang, H., Yuan, Z., Li, Z., Peng, W., and Wang, J. 一种结合 U-Net 与傅里叶神经算子的混合框架，用于快速预测具有混合周期性和非周期性边界条件的湍流。*arXiv* 预印本 *arXiv:2504.13126*，2025。

Wen, G., Li, Z., Azizzadenesheli, K., Anandkumar, A., and Benson, S. M. U-FNO——一种基于傅里叶神经算子的增强型深度学习模型，用于多相流。水资源进展，163:104180，2022。

Wu, H., Luo, H., Wang, H., Wang, J., and Long, M. Transolver：一种用于一般几何形状上偏微分方程的快速 Transformer 求解器。*arXiv* 预印本 *arXiv:2402.02366*，2024。

Yeo, S., Nguyen, P. A., Le, A. N., and Mishra, S. KAN-PDEs：一种使用 Kolmogorov-Arnold 网络求解偏微分方程的新方法——精度与效率均有所提升。载于国际电气与电子工程会议，第 43–62 页。Springer，2024。

You, Z.、Xu, Z.与Cai, W. MSCaleFNO：面向振荡函数空间的多尺度傅里叶神经算子学习。*arXiv*预印本 *arXiv:2412.20183*，2024。

Zhang, Z., Wang, Q., Zhang, Y., Shen, T., and Zhang, W. 采用混合 Kolmogorov-Arnold 网络和增广拉格朗日函数求解偏微分方程的物理信息神经网络。科学报告，15(1):10523，2025。

附录路线图

本附录按自底向上的方式组织，以支持正文：我们首先为 §3.5 中的论断提供完整的理论证明，然后报告更多实验结果和定性可视化。

A. 理论理解（证明与算子视角）。

- **A.1 引理 3.1 的证明（引理 A.1）**。我们证明了基于样条的 KAN 边函数的边函数利普希茨界，并通过节点间距和样条系数明确控制 Lip(ϕ)。
- **A.2 定理 3.2 的证明（定理 A.2）**。我们证明由 KAN 令牌诱导的全局调制映射具有利普希茨连续性，并说明该常数对 KAN 路径的依赖仅通过 L_{KAN}体现，因此 KAN 可作为可控的全局光滑度控制器。
- **A.3 命题 3.3 的证明（命题 A.3）**。我们证明，GML 中使用的离散单查询注意力机制是连续归一化积分算子的一致求积近似，且在网格细化下满足 $\|A_h - A\| = O(h)$。
- **A.4 神经算子解释（定理 A.4）**。我们将注意力核 $\kappa_u(x, y)$ 形式化，并证明 GML 模块定义了一个（非线性）积分核神经算子层；离散化后得到标准的 softmax 注意力。

B. 补充实验结果（更多指标与可视化证据）。

- B.1 **数据集描述**。我们提供该数据集的更多细节。
- B.2 **补充 PDEBench 结果**。我们报告主均方根误差表之外的扩展指标，包括归一化均方根误差、最大均方根误差、守恒变量均方根误差、边界均方根误差，以及按频带划分的傅里叶空间均方根误差。
- B.3 **补充可视化结果**。我们提供补充性定性比较（多个时间快照/面板），以突出在具有挑战性的时空动力学中的稳定性与细尺度保真度。

A. 理论理解

本节中，我们首先给出引理 3.1 的证明，然后给出定理3.2 和命题 3.3 的证明。最后，我们证明了我们的方法是一个神经算子。

我们首先给出引理 3.1 的证明。

引理 A.1 (样条 KAN 的边函数利普希茨界**)**。 考虑如下形式的边函数 $\phi(x) = w_b\, b(x) + w_s \sum_r c_r\, B_r^{(k)}(x;\Xi)$，其中 $b \in C^1$，且 $B_r^{(k)}$ 是定义在节点网格 $\Xi = \{t_j\}$ 上的 k 阶 B 样条，其最小节点间距为 $h_{\min} := \min_j (t_{j+1} - t_j) > 0$。假设 $\sup_x |b'(x)| \le L_b$。则 ϕ 是 *Lipschitz* 连续的，且其利普希茨常数满足

$$\mathrm{Lip}(\phi) \le |w_b|\, L_b + |w_s|\, \frac{2}{h_{\min}}\, \|c\|_1. \tag{13}$$

证明。 B 样条满足导数恒等式

$$\frac{d}{dx} B_r^{(k)}(x) = \frac{k}{t_{r+k} - t_r} \left(B_r^{(k-1)}(x) - B_{r+1}^{(k-1)}(x) \right).$$

由于 $B_r^{(k-1)} \in [0,1]$ 和 $t_{r+k} - t_r \ge k h_{\min}$，我们得到

$$\left| \frac{d}{dx} B_r^{(k)}(x) \right| \le \frac{k}{k h_{\min}} \left(|B_r^{(k-1)}(x)| + |B_{r+1}^{(k-1)}(x)| \right) \le \frac{2}{h_{\min}}.$$

因此，

$$|\phi'(x)| \le |w_b||b'(x)| + |w_s| \sum_r |c_r| \left| \frac{d}{dx} B_r^{(k)}(x) \right| \le |w_b| L_b + |w_s| \frac{2}{h_{\min}} \|c\|_1.$$

根据中值定理，$\mathrm{Lip}(\phi) \le \sup_x |\phi'(x)|$，得证。 □

然后我们给出定理3.2的证明。

定理 A.2. 假设如下：

1. （有界查询/键）存在常数 M_q、M_k，使得对所有可容许的 u，

$$\|q(\Phi_{\mathrm{KAN}}(P(u)))\|_2 \le M_q, \qquad \|k(u)\|_\infty \le M_k.$$

2. （***Lipschitz*** 部分）存在常数 L_P、L_{KAN}、L_k、L_v，使得

$$\|P(u) - P(\tilde{u})\|_2 \le L_P \|u - \tilde{u}\|_\infty,$$

$$\|\Phi_{\mathrm{KAN}}(z) - \Phi_{\mathrm{KAN}}(\tilde{z})\|_2 \le L_{\mathrm{KAN}} \|z - \tilde{z}\|_2,$$

$$\|k(u) - k(\tilde{u})\|_\infty \le L_k \|u - \tilde{u}\|_\infty, \qquad \|v(u) - v(\tilde{u})\|_\infty \le L_v \|u - \tilde{u}\|_\infty.$$

3. （***KAN*** 样条正则性）*KAN* 层中的每个边函数是阶数为 k 的样条，并且是 C^1，其导数一致有界（这由具有非退化节点间距的 *B* 样条满足；证明见下文）。

于是，全局调制映射 $m : L^\infty(\mathcal{D}) \to \mathbb{R}^{d_t}$ 是利普希茨的：

$$\|m(u) - m(\tilde{u})\|_2 \le L_m \|u - \tilde{u}\|_\infty,$$

其中 L_m 是一个取决于以下因素的显式常数

$$L_P,\ L_{\mathrm{KAN}},\ L_k,\ L_v,\ M_q,\ M_k,\ |\mathcal{D}|,$$

以及指数“温度”缩放$1/\sqrt{d_a}$。特别地，减小 $\overline{L_{\mathrm{KAN}}}$ （通过样条导数控制）会直接减小 L_m。

证明。 利普希茨界，其中 KAN 仅作为全局光滑度控制器出现。

Let

$$s_u(y) = \frac{1}{\sqrt{d_a}} \langle q_u, k_u(y) \rangle, \qquad q_u := q(\Phi_{\mathrm{KAN}}(P(u))), \qquad k_u(\cdot) := k(u)(\cdot),$$

并定义

$$w_u(y) = e^{s_u(y)}, \qquad Z_u = \int_{\mathcal{D}} w_u(y)\, dy, \qquad \pi_u(y) = \frac{w_u(y)}{Z_u}.$$

另外定义分子

$$N_u = \int_{\mathcal{D}} w_u(y)\, v_u(y)\, dy, \qquad v_u(\cdot) := v(u)(\cdot).$$

那么全局调制映射为

$$m(u) = \frac{N_u}{Z_u}.$$

步骤 1： s_u、w_u **和** Z_u **的一致有界性**。根据假设 1，

$$|s_u(y)| \le \frac{1}{\sqrt{d_a}} \|q_u\|_2 \, \|k_u(y)\|_2 \le \frac{M_q M_k}{\sqrt{d_a}} =: M_s.$$

因此

$$w_u(y) \in [e^{-M_s}, e^{M_s}], \qquad Z_u = \int_{\mathcal{D}} w_u(y)\, dy \ge |\mathcal{D}| e^{-M_s} =: Z_{\min} > 0.$$

步骤 2：界定 $\mathbf{g}\|w_u - w_{\tilde{u}}\|_{L^\infty}$**.** 在区间 $[-M_s, M_s]$ 上，指数函数是利普希茨连续的，其常数为 e^{M_s} ：

$$|e^a - e^b| \le e^{M_s}|a-b|, \qquad \forall a, b \in [-M_s, M_s].$$

因此,

$$\|w_u - w_{\tilde{u}}\|_{L^\infty} \le e^{M_s}\|s_u - s_{\tilde{u}}\|_{L^\infty}.$$

步骤 3：界定 $\|s_u - s_{\tilde{u}}\|_{L^\infty}$ **并揭示** L_{KAN}。对于任意 y,

$$s_u(y) - s_{\tilde{u}}(y) = \frac{1}{\sqrt{d_a}}\Big(\langle q_u - q_{\tilde{u}}, k_u(y)\rangle + \langle q_{\tilde{u}}, k_u(y) - k_{\tilde{u}}(y)\rangle\Big).$$

因此,

$$|s_u(y) - s_{\tilde{u}}(y)| \le \frac{1}{\sqrt{d_a}}\Big(\|q_u - q_{\tilde{u}}\|_2\|k_u(y)\|_2 + \|q_{\tilde{u}}\|_2\|k_u(y) - k_{\tilde{u}}(y)\|_2\Big).$$

对 y 取上确界并利用有界性,

$$\|s_u - s_{\tilde{u}}\|_{L^\infty} \le \frac{1}{\sqrt{d_a}}\left(M_k\|q_u - q_{\tilde{u}}\|_2 + M_q\|k_u - k_{\tilde{u}}\|_{L^\infty}\right).$$

Now,

$$\|q_u - q_{\tilde{u}}\|_2 \le \mathrm{Lip}(q)\|\Phi_{\mathrm{KAN}}(P(u)) - \Phi_{\mathrm{KAN}}(P(\tilde{u}))\|_2 \le \mathrm{Lip}(q)\, L_{\mathrm{KAN}}\, L_P\|u - \tilde{u}\|_{L^\infty},$$

and

$$\|k_u - k_{\tilde{u}}\|_{L^\infty} \le L_k\|u - \tilde{u}\|_{L^\infty}.$$

因此,

$$\|s_u - s_{\tilde{u}}\|_{L^\infty} \le \frac{1}{\sqrt{d_a}}\left(M_k\mathrm{Lip}(q)L_{\mathrm{KAN}}L_P + M_qL_k\right)\|u - \tilde{u}\|_{L^\infty}.$$

结合步骤 2,

$$\|w_u - w_{\tilde{u}}\|_{L^\infty} \le \underbrace{\frac{e^{M_s}}{\sqrt{d_a}}\left(M_k\mathrm{Lip}(q)L_{\mathrm{KAN}}L_P + M_qL_k\right)}_{=:C_w}\|u - \tilde{u}\|_{L^\infty}.$$

步骤 4：界定分子和分母之差。对于分母,

$$|Z_u - Z_{\tilde{u}}| = \left|\int_{\mathcal{D}}(w_u - w_{\tilde{u}})\,dy\right| \le |\mathcal{D}|\,C_w\|u - \tilde{u}\|_{L^\infty}.$$

对于分子,

$$N_u - N_{\tilde{u}} = \int_{\mathcal{D}}\Big((w_u - w_{\tilde{u}})v_u + w_{\tilde{u}}(v_u - v_{\tilde{u}})\Big)dy.$$

因此,

$$\|N_u - N_{\tilde{u}}\|_2 \le |\mathcal{D}|\Big(M_vC_w + e^{M_s}L_v\Big)\|u - \tilde{u}\|_{L^\infty},$$

其中 $M_v :=$ 上确界$_u\,\|v_u\|_{L^\infty}$。此外,

$$\|N_{\tilde{u}}\|_2 \le |\mathcal{D}|e^{M_s}M_v.$$

步骤 5：比值之差。使用

$$\frac{N_u}{Z_u} - \frac{N_{\tilde{u}}}{Z_{\tilde{u}}} = \frac{N_u - N_{\tilde{u}}}{Z_u} + N_{\tilde{u}}\left(\frac{1}{Z_u} - \frac{1}{Z_{\tilde{u}}}\right), \qquad \left|\frac{1}{Z_u} - \frac{1}{Z_{\tilde{u}}}\right| = \frac{|Z_u - Z_{\tilde{u}}|}{Z_u Z_{\tilde{u}}},$$

并注意到 Z_u， $Z_{\tilde{u}} \geq Z_{\min}$，我们得到

$$\|m(u) - m(\tilde{u})\|_2 \leq \frac{1}{Z_{\min}}\|N_u - N_{\tilde{u}}\|_2 + \frac{1}{Z_{\min}^2}\|N_{\tilde{u}}\|_2 |Z_u - Z_{\tilde{u}}|.$$

代入上述界以及 $Z_{\min} = |\mathbb{D}|e^{-M_s}$，可得到一个有限常数 L_m，使得

$$\|m(u) - m(\tilde{u})\|_2 \leq L_m \|u - \tilde{u}\|_{L^\infty}.$$

□

结论。关键在于，利普希茨常数 L_m 仅通过 L_{KAN} （以及查询的利普希茨常数）依赖于词元网络。因此，直接对 KAN 样条导数进行正则化，就能正则化全局调制的敏感性。

我们首先定义连续（极限）GML算子：设 $D \subset \mathbb{R}^d$ 为有界域， $w : D \to \mathbb{R}_+$ 为非负权重函数。连续归一化的 GML注意力输出 $A \in \mathbb{R}^{d_v}$ 定义为

$$A = \frac{\int_D w(y)\, V(f(y))\, dy}{\int_D w(y)\, dy}. \tag{14}$$

这个表达式是一个归一化核积分，可解释为在概率密度正比于 $w(y)$ 的条件下， $V(f(y))$ 的期望。接下来，我们可以定义离散 GML算子：将 D 划分为 n^d 个边长为 $h = 1/n$ 的均匀单元，并令 $\{y_j\}_{j=1}^N$ 表示单元中心，其中 $N = n^d$。离散 GML算子定义为

$$A_h = \frac{\sum_{j=1}^N w(y_j)\, V(f(y_j))}{\sum_{j=1}^N w(y_j)}. \tag{15}$$

命题 A.3. 假设：

1. w 在 D上满足利普希茨条件，利普希茨常数为 L_w，且有界：

$$0 < w(y) \leq M_w.$$

2. $g(y) := w(y)V(f(y))$ 在 D 上（向量值）满足利普希茨条件，且在欧几里得范数下的利普希茨常数为 L_g ：

$$\|g(y) - g(y')\| \leq L_g \|y - y'\|.$$

3. 归一化常数

$$Z = \int_D w(y)\, dy$$

为正（由于 $w > 0$，这一点始终成立；我们仍明确写出）。

那么，离散注意力输出 A_h 收敛到连续输出 A，当 $h \to 0$时成立；并且对于充分小的 h，

$$\|A_h - A\| \leq \frac{\sqrt{d}\, h}{Z}\big(L_g + \|A\| L_w\big).$$

特别是，

$$\|A_h - A\| = O(h)$$

随着网格细化。

证明。**步骤 1：利普希茨函数的黎曼和误差界**

将定义域 $D = [0,1]^d$ 划分为 $N = n^d$ 个边长为 $h = 1/n$ 的超立方体 $\{C_j\}_{j=1}^N$。令 y_j 表示 C_j的中心。对于任意具有利普希茨常数 L_ϕ的利普希茨标量函数 $\phi : D \to \mathbb{R}$，

$$\left| \int_D \phi(y)\, dy - \sum_{j=1}^{N} \phi(y_j)\, |C_j| \right| \le \sum_{j=1}^{N} \int_{C_j} |\phi(y) - \phi(y_j)|\, dy. \tag{16}$$

由于 ϕ 是利普希茨的,

$$|\phi(y) - \phi(y_j)| \le L_\phi \|y - y_j\|.$$

在边长为 h的立方体中，到中心的最大距离满足 $\|y - y_j\| \le \frac{\sqrt{d}}{2} h$。因此，对于所有 $y \in C_j$,

$$|\phi(y) - \phi(y_j)| \le L_\phi \frac{\sqrt{d}}{2} h.$$

因此,

$$\int_{C_j} |\phi(y) - \phi(y_j)| dy \le L_\phi \frac{\sqrt{d}}{2} h\, |C_j| = L_\phi \frac{\sqrt{d}}{2} h^{d+1}. \tag{17}$$

对所有 j 求和并使用 $\sum_j |C_j| = |D| = 1$，得到

$$\left| \int_D \phi(y)\, dy - h^d \sum_{j=1}^{N} \phi(y_j) \right| \le L_\phi \frac{\sqrt{d}}{2} h. \tag{18}$$

同样的论证逐分量适用于向量值利普希茨函数 $\psi : D \to \mathbb{R}^m$，得到

$$\left\| \int_D \psi(y)\, dy - h^d \sum_{j=1}^{N} \psi(y_j) \right\| \le L_\psi \frac{\sqrt{d}}{2} h. \tag{19}$$

步骤 2：将界应用于注意力分子和分母 定义连续量 es

$$Z = \int_D w(y)\, dy, \qquad N = \int_D w(y) V(f(y))\, dy = \int_D g(y)\, dy,$$

及其黎曼和近似

$$Z_h = h^d \sum_{j=1}^{N} w(y_j), \qquad N_h = h^d \sum_{j=1}^{N} g(y_j).$$

将18应用于 w，并将 19 应用于 g 可得

$$|Z - Z_h| \le L_w \frac{\sqrt{d}}{2} h, \tag{20}$$

$$\|N - N_h\| \le L_g \frac{\sqrt{d}}{2} h. \tag{21}$$

离散注意力输出为

$$A_h = \frac{\sum_j w(y_j) V(f(y_j))}{\sum_j w(y_j)} = \frac{N_h}{Z_h},$$

因为 h^d 的因子会抵消。连续注意力输出为 $A = N/Z$。

步骤 3：界定比值之差

我们计算

$$A_h - A = \frac{N_h}{Z_h} - \frac{N}{Z} = \frac{Z(N_h - N) + N(Z - Z_h)}{ZZ_h}.$$

取范数,

$$\|A_h - A\| \le \frac{Z\|N_h - N\| + \|N\||Z - Z_h|}{ZZ_h}. \tag{22}$$

由于 $w > 0$，我们有 $Z > 0$。假设 h 足够小，使得

$$|Z - Z_h| \le \frac{Z}{2},$$

这在以下条件下成立:

$$h \le \frac{Z}{L_w\sqrt{d}}. \tag{23}$$

那么 $Z_h \ge Z/2$，并且

$$ZZ_h \ge \frac{Z^2}{2}. \tag{24}$$

代入24到 22中,

$$\|A_h - A\| \le \frac{2}{Z}\left(\|N_h - N\| + \frac{\|N\|}{Z}|Z - Z_h|\right). \tag{25}$$

由于 $A = N/Z$，我们有 $\|N\|/Z = \|A\|$。应用界 (1)–(2) 可得

$$\|A_h - A\| \le \frac{2}{Z}\left(L_g\frac{\sqrt{d}}{2}h + \|A\|L_w\frac{\sqrt{d}}{2}h\right) \tag{26}$$

$$= \frac{\sqrt{d}\,h}{Z}\left(L_g + \|A\|L_w\right). \tag{27}$$

□

离散注意力算子以如下速率收敛到其连续对应算子

$$\boxed{\|A_h - A\| = O(h)}$$

当 $h \to 0$ 时，其中的常数取决于 w 和 g 的利普希茨界，以及连续注意力输出的大小。

定理 A.4. 定义（单查询）注意力核

$$\kappa_u(x,y) = \frac{\exp\left(\frac{\langle q(\Phi_{\mathrm{KAN}}(u)), k(y)\rangle}{\sqrt{d_a}}\right)}{\int_D \exp\left(\frac{\langle q(\Phi_{\mathrm{KAN}}(u)), k(\xi)\rangle}{\sqrt{d_a}}\right) d\xi}, \qquad x, y \in D, \tag{28}$$

以及相应的算子

$$\left(A_{\Phi_{\mathrm{KAN}}}(u)\right)(x) = \int_D \kappa_u(x,y)\, V\left(f(u)(y)\right) dy, \qquad x \in D. \tag{29}$$

于是，下列陈述成立：*(1)*$A_{\Phi_{\mathrm{KAN}}}(u)$ 是一个非线性积分核算子，因此是神经算子层的一种特殊情形，其中允许核依赖于整个输入函数。

(2) 对积分进行离散化（例如通过求积或黎曼和），即可得到标准的 softmax 注意力 计算，即一个单查询交叉注意力块。

证明。 **步骤 1：回顾神经算子层模板（非线性核形式）**。 在神经算子框架中，一种可行的核积分 层是非线性积分核算子:

$$(\mathcal{K}(v))(x) = \int_D \kappa\big(x, y, v(\Pi(x)), v(y)\big)\, v(y)\, d\nu(y), \tag{30}$$

其中，核 κ可依赖于输入表示 v，从而引入非线性。

此外，在Kovachkietal.、2023年关于命题 6（Transformer 作为神经算子）的证明中，作者明确引入符号 κ_v，表示核可以依赖于整个函数 v，而不仅仅是逐点取值。文中将其描述为对（30）书面形式的“直接推广”。这种推广正是依赖于全局令牌的核所需要的

$$t = \Phi_{\mathrm{KAN}}(u).$$

步骤 2：将 SKAN 的 GML 注意力机制写作积分核算子。在我们的模型中，基于注意力机制的全局调制层（GML）定义为积分算子（式（16）：

$$(\mathcal{A}_t[u])(x) = \int_D \kappa_t(x, y)\, V\big(f(u)(y)\big)\, dy, \tag{31}$$

其中 $t = \Phi_{\mathrm{KAN}}(u)$，且 V是线性值投影。

由于 V是线性的，因此存在一个矩阵 W_V，使得

$$V(z) = W_V z.$$

Let

$$v(\cdot) := f(u)(\cdot) \in \mathbb{R}^{C_{\mathrm{fno}}}.$$

那么（31）可以改写为

$$(\mathcal{A}_{\Phi_{\mathrm{KAN}}(u)}[u])(x) = \int_D \kappa_u(x, y)\, W_V\, v(y)\, dy. \tag{32}$$

定义矩阵值核

$$\widetilde{\kappa}_u(x, y) := \kappa_u(x, y)\, W_V.$$

该算子变为

$$(\mathcal{A}_{\Phi_{\mathrm{KAN}}(u)}[u])(x) = \int_D \widetilde{\kappa}_u(x, y)\, v(y)\, dy, \tag{33}$$

其结构形式与神经算子核积分映射 K(·) 完全相同 .

Step 3：SKAN 注意力核的有效性。单查询交叉注意力核定义为 y

$$\kappa_u(x, y) = \frac{\exp\big(\langle q(\Phi_{\mathrm{KAN}}(u)),\, k(y)\rangle/\sqrt{d_a}\big)}{\int_D \exp\big(\langle q(\Phi_{\mathrm{KAN}}(u)),\, k(\xi)\rangle/\sqrt{d_a}\big)\, d\xi}, \tag{34}$$

with

$$q(\Phi_{\mathrm{KAN}}(u)) = Q(\Phi_{\mathrm{KAN}}(u)), \qquad k(y) = K(f(u)(y)) = K(v(y)).$$

因此，核 κ_u 取决于：(i) 通过全局令牌 $\Phi_{\mathrm{KAN}}(u)$作用的整个输入场 u；(ii) 通过键映射 $K(v(y))$作用的逐点特征值$v(y)$；(iii) 在单查询设置下，与 x 无关，但广播/读出除外。这种依赖关系正是（Kovachki等人，2023）的命题 6 所允许的广义形式 κ_v。因此，$\mathrm{A}_{\Phi_{\mathrm{KAN}}(u)}$ 是一个有效的非线性神经算子核映射。

步骤 4：离散化得到标准的 softmax 注意力。 设 $\{y_j\}_{j=1}^N \subset D$ 为求积点，其权重为 w_j。用求积近似积分可得

$$\int_D \kappa_u(x, y)\, V(v(y))\, dy \approx \sum_{j=1}^{N} w_j\, \kappa_u(x, y_j)\, V(v(y_j)), \tag{35}$$

$$\int_D \exp\Big(\langle q(t), k(\xi)\rangle/\sqrt{d_a}\Big)\, d\xi \approx \sum_{\ell=1}^{N} w_\ell\, \exp\Big(\langle q(t), k(y_\ell)\rangle/\sqrt{d_a}\Big). \tag{36}$$

对于均匀权重，常数项相互抵消，从而得到离散注意力系数

$$\alpha_j(t) = \frac{\exp\big(\langle q(t), k(y_j)\rangle/\sqrt{d_a}\big)}{\sum_{\ell=1}^{N}\exp\big(\langle q(t), k(y_\ell)\rangle/\sqrt{d_a}\big)}. \tag{37}$$

因此，

$$(\mathcal{A}_t[u])(x_i) \approx \sum_{j=1}^{N} \alpha_j(t)\, V\big(f(u)(y_j)\big), \tag{38}$$

这正是标准的 softmax($QK^\top$)V注意力计算。

步骤 5：逐点融合保留了神经算子结构。 最后，注意力输出通过以下方式进行广播，并与局部特征融合：

$$\hat{u}(x) = R\big(f(u)(x), \mathcal{A}_t[u](x)\big), \tag{39}$$

其中 R 是一个逐点（1×1 卷积/多层感知机）映射。此类局部读出是神经算子架构中的标准涅米茨基型算子。因此，完整的 GML 模块——先进行非局部核积分，再进行逐点融合——仍是一个有效的神经算子层。

□

因此，SKAN 的 Transformer 组件（GML 交叉注意力以及广播融合/读出）构成了一个神经算子风格的层。

B. 附加实验结果

本节介绍数据集，并展示 PDEBench 上的附加实验结果和更多可视化结果。

B.1. 数据集

对于 PDEBench 中的一维 CNS、一维扩散反应、二维 Darcy 流和二维浅水方程，我们在表4中汇总了相应的偏微分方程。此外，对于浅水方程和气候建模，我们采用（Kissas 等人，2022）中的数据。

对于气候建模，目标是近似地表气温与地表气压之间的关系。我们旨在学习一个黑箱算子，以便直接根据温度预测气压。具体而言，我们考虑如下映射

$$T(x) \longmapsto P(y),$$

其中 $x,\ y\in[-90,90]\times[0,360]$ 表示经纬度坐标。对于一年中的某个固定日期，未知的解算子将地表气温场映射为相应的地表气压场。

在此设定中，输入定义域与输出定义域相同，因此 $X=Y$。由于温度和压力都是标量场，我们有 $d_u=d_s=1$，且空间维度满足 $d_x=d_y=2$，对应于二维经纬度网格。因此，算子学习问题可表述为

$$G: C(X,\mathbb{R}) \to C(X,\mathbb{R}),$$

其中，G 将定义域 X 上的连续温度场映射为同一定义域上的连续压力场。

对于浅水方程，它描述薄流体层的演化过程：当水平长度尺度远大于垂直方向的长度尺度时，流动可视为深度平均。

设 $x=(x_1,x_2)\in(0,1)^2$ 且 $t\in(0,1]$。未知量为流体柱总高度 $\rho(t,x)$ 以及深度平均水平速度分量 $v_1(t,x)$ 和 $v_2(t,x)$。给定重力加速度 g，浅水系统可

写为

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v_1)}{\partial x_1} + \frac{\partial(\rho v_2)}{\partial x_2} = 0, \tag{40}$$

$$\frac{\partial(\rho v_1)}{\partial t} + \frac{\partial}{\partial x_1}\left(\rho v_1^2 + \frac{1}{2} g \rho^2\right) + \frac{\partial(\rho v_1 v_2)}{\partial x_2} = 0, \tag{41}$$

$$\frac{\partial(\rho v_2)}{\partial t} + \frac{\partial(\rho v_1 v_2)}{\partial x_1} + \frac{\partial}{\partial x_2}\left(\rho v_2^2 + \frac{1}{2} g \rho^2\right) = 0. \tag{42}$$

等价地，该系统可写成守恒形式：

$$\frac{\partial \tilde{U}}{\partial t} + \frac{\partial \tilde{F}(\tilde{U})}{\partial x_1} + \frac{\partial \tilde{G}(\tilde{U})}{\partial x_2} = 0,$$

其中，守恒变量和通量为

$$\tilde{U} = \begin{pmatrix} \rho \\ \rho v_1 \\ \rho v_2 \end{pmatrix}, \qquad \tilde{F} = \begin{pmatrix} \rho v_1 \\ \rho v_1^2 + \frac{1}{2} g \rho^2 \\ \rho v_1 v_2 \end{pmatrix}, \qquad \tilde{G} = \begin{pmatrix} \rho v_2 \\ \rho v_1 v_2 \\ \rho v_2^2 + \frac{1}{2} g \rho^2 \end{pmatrix}.$$

给定初始状态，相应的解算子将初始高度场和速度场映射为后续时刻的高度场和速度场。由于输入和输出定义在同一时空域上，我们取 $X = Y$ with $d_x = d_y = 3$ （时间加两个空间坐标）和 $d_u = d_s = 3$ （三个输出通道）。因此，算子学习任务为

$$G : C(X, \mathbb{R}^3) \to C(X, \mathbb{R}^3).$$

它施加反射（固壁）边界条件，即边界上没有法向流动：

$$v \cdot \hat{n} = v_1 n_{x_1} + v_2 n_{x_2} = 0 \qquad \text{on } \partial(0,1)^2,$$

^ ^ 其中，$\hat{n} = n_{x_1} i + n_{x_2} j$ 表示单位外法向量。

为生成训练数据，我们对对应于“下落液滴”扰动的初始条件进行随机采样，该扰动的初始速度为零：

$$\rho(0, x) = 1 + h \exp\left(-\frac{(x_1 - \xi)^2 + (x_2 - \zeta)^2}{w}\right), \tag{43}$$

$$v_1(0, x) = 0, \qquad v_2(0, x) = 0, \tag{44}$$

其中， h 控制液滴振幅， w 控制其宽度，(ξ, ζ)表示其初始中心。由于所有样本在 $t = 0$ 时的速度都为零，我们改为使用第一时间步 $t_0 = \Delta t = 0.002$s处的状态$(\rho(t_0, \cdot)\ v_1(t_0, \cdot)\ v_2(t_0, \cdot))$来定义算子输入。

随机参数从均匀分布中抽取：

$$h \sim \mathcal{U}(1.5, 2.5), \qquad w \sim \mathcal{U}(0.002, 0.008), \qquad \xi \sim \mathcal{U}(0.4, 0.6), \qquad \zeta \sim \mathcal{U}(0.4, 0.6).$$

通过在 32×32 网格上对这些初始条件进行采样，并使用 Lax–Friedrichs 格式求解正问题，构建了该数据集。由此得到的解覆盖完整的 32×32 网格，之后可对其进行子采样，形成仅使用部分可用空间观测值的训练集，同时仍能学习预测完整解。

B.2. 补充实验结果

本节提供 PDEBench 的补充实验结果，包括归一化均方根误差、均方根误差最大值、守恒变量的均方根误差、边界处的均方根误差、傅里叶空间低频段的均方根误差、傅里叶空间中频段的均方根误差和傅里叶空间高频段的均方根误差。

B.3. 额外可视化

本节提供二维浅水、气候建模和浅水方程的补充可视化结果。

PDE	控制方程	N_s	N_t	# 样本
一维 CNS	（一维可压缩纳维-斯托克斯方程） $\partial_t \rho + \nabla \cdot (\rho v) = 0,$ $\rho(\partial_t v + v \cdot \nabla v) = -\nabla p + \eta\, \partial_{xx} v + (\zeta + \eta/3)\, \nabla(\nabla \cdot v),$ $\partial_t(\epsilon + \frac{\rho v^2}{2}) + \nabla \cdot \left[(p + \epsilon + \frac{\rho v^2}{2})\, v - v \cdot \sigma'\right] = 0.$	128	20	10000
一维扩散-反应	$\partial_t u(t,x) - \nu\, \partial_{xx} u(t,x) - \rho\, u(t,x)\big(1 - u(t,x)\big) = 0.$	256	200	10000
二维 Darcy 流	（稳态达西流） $-\nabla \cdot (a(x) \nabla u(x)) = f(x), \quad x \in (0,1)^2,$ $u(x) = 0, \quad x \in \partial(0,1)^2.$	64×64	–	10000
二维浅水	$\partial_t h + \partial_x(hu) + \partial_y(hv) = 0,$ $\partial_t(hu) + \partial_x(u^2 h + \frac{1}{2} g_r h^2) + \partial_y(uvh) = -g_r h\, \partial_x b,$ $\partial_t(hv) + \partial_y(v^2 h + \frac{1}{2} g_r h^2) + \partial_x(uvh) = -g_r h\, \partial_y b.$	64×64	100	1000

表 *4.* 偏微分方程与数据集规模（N_s：空间分辨率，N_t：时间分辨率）。对于 DarcyFlow，N_t 不适用，因为该数据集为稳态数据集。

表 *5.* 通过不同数据集比较我们的方法与其他代表性基线方法的性能，指标为归一化均方根误差。性能最佳的结果以绿色标示，次佳结果以蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	0.8123	0.07399	**0.06215**	0.11552
FNO（2021）	0.23532	0.05484	0.13772	0.01393
FFNO（2023）	**0.16448**	0.02685	0.13121	0.01145
LNO（2024）	0.72711	0.09054	0.21483	0.03793
Transolver（2024）	0.35738	0.01347	0.10777	0.00565
LocalFNO（2024）	–	–	0.10662	0.00422
DeepOKAN（2025）	2.77435	0.08516	0.09196	0.08282
CViT（2025）	0.19233	0.01263	0.12313	0.01133
（本文方法）	0.17462	**0.00814**	0.06304	**0.00231**
提升（%）	-6.16%	35.55%	-1.43%	45.26%

表 6. 通过不同数据集比较我们的方法与其他具有代表性的基线方法，指标为均方根误差最大值。表现最佳的结果以绿色标出，表现次佳的结果以蓝色标出。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	50.83448	0.21527	**0.18478**	0.75481
FNO（2021）	4.78869	0.0521	0.22714	0.13625
FFNO（2023）	4.55828	0.0304	0.25059	0.12899
LNO（2024）	38.83067	0.25961	0.41448	0.26936
Transolver（2024）	8.22922	0.03871	0.26815	0.12001
LocalFNO (2024)	–	–	0.21291	0.09241
DeepOKAN (2025)	39.21762	0.14834	0.22246	0.63804
CViT (2025)	8.25447	0.04548	0.22511	0.15085
（我们的方法）	**2.89683**	**0.02429**	0.19617	**0.03033**
提升（%）	36.45%	18.09%	-6.16%	67.18%

表7。 我们的方法与其他具有代表性的基线方法在不同数据集上的性能比较，采用 守恒变量的均方根误差。最佳结果以 绿色标出，次佳结果以 蓝色 标出。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	15.34809	0.05359	0.01437	0.03397
FNO（2021）	0.0879	0.03036	0.02368	0.00058
FFNO（2023）	0.06288	0.01511	0.01976	0.00069
LNO (2024)	10.74895	0.07057	0.02867	0.00452
Transolver (2024)	0.22427	0.00963	0.0144	0.00066
LocalFNO (2024)	–	–	0.02042	0.00038
DeepOKAN (2025)	10.80045	0.03845	0.0149	0.01251
CViT (2025)	0.05702	0.01131	0.0166	0.00144
（本文）	**0.0433**	**0.00586**	**0.01183**	**0.00012**
提升（%）	24.37%	39.15%	17.85%	68.42%

表 8。 通过不同数据集比较我们的方法与其他代表性基线方法的性能，比较指标为 边界处的均方根误差。最佳结果以 绿色蓝色标示，次佳结果以蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	11.72038	0.03576	**0.00407**	0.01159
FNO（2021）	0.2955	0.02934	0.02069	0.00338
FFNO（2023）	0.22411	0.01408	0.0136	0.00364
LNO（2024）	8.82148	0.04704	0.02373	0.01552
Transolver（2024）	0.62781	0.00682	0.01248	0.00371
LocalFNO（2024）	–	–	0.00772	0.00201
DeepOKAN（2025）	9.05406	0.0409	0.01221	0.02932
CViT (2025)	0.74478	0.00675	0.01503	0.00572
（我们的方法）	**0.11007**	**0.00386**	0.00581	**0.00054**
提升（%）	50.89%	45.48%	-42.75%	73.13%

表9. 我们的方法与其他代表性基线方法在不同数据集上的性能比较，评估指标为 傅里叶空间低频段的均方根误差。性能最佳的结果以 绿色标示，次佳结果以 蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	4.42510	0.01579993	0.00676289	0.03163715
FNO（2021）	0.11258	0.0078190938	0.01132	0.00059789
FFNO（2023）	0.09931	0.0040068841	0.01056	0.00223955
LNO（2024）	3.01964	0.0183305	0.01507362	0.0067391
Transolver（2024）	0.27045	0.0026418727	0.00813	0.00073594
LocalFNO (2024)	–	–	0.02896	0.00041726
DeepOKAN (2025)	3.09761	0.01418	0.00787	0.02002841
CViT (2025)	0.22286	0.00306786	0.00936	0.00136
（我们的方法）	**0.04996734**	**0.0016653806**	**0.00590517**	**0.00012509**
提升（%）	49.69%	36.96%	12.68%	70.02%

表10. 我们的方法与其他代表性基线方法在不同数据集上的性能比较，指标为傅里叶空间中频段的均方根误差。最佳结果以绿色标出，次佳结果以蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	0.18985	0.00102379	**0.000643**	0.00874112
FNO（2021）	0.04998	0.000050198119	0.00104	0.0006694
FFNO（2023）	0.04275	0.000052739455	0.00152	0.00109296
LNO（2024）	0.06662	0.0000079677	0.00211222	0.00487229
Transolver（2024）	0.08406	0.000080886944	0.00128	0.00061484
LocalFNO（2024）	–	–	0.00297	0.00050069
DeepOKAN（2025）	0.08463	0.00321	0.00108	0.00839461
CViT（2025）	0.07991	0.0000476469	0.00115	0.00126
（我们的）	**0.0160807**	**0.000044599139**	0.00092448	**0.00026044**
提升 (%)	62.37%	6.40%	-43.78%	47.98%

表11. 我们的方法与其他代表性基线方法在不同数据集上的性能比较，指标为傅里叶空间高频段的均方根误差。表现最佳的结果以绿色突出显示，而次佳结果以蓝色标示。

方法	CNS（1D）	扩散–反应（1D）	达西流（2D）	浅水（2D）
U-Net（2015）	0.03155	0.00022391	**0.00012363**	0.00196839
FNO（2021）	0.00706	0.0000031962038	0.00026	0.00147641
FFNO（2023）	0.00686	0.0000038141336	0.00053	0.00042784
LNO（2024）	0.00583	**0.0000000697743**	0.00075919	0.00346834
Transolver（2024）	0.01283	0.000011153535	0.00041	0.00042714
LocalFNO（2024）	–	–	0.00118	0.00033259
DeepOKAN（2025）	0.00893	0.00019	0.00025	0.00229325
CViT（2025）	0.0156	0.000024853	0.0003	0.00087
（我们的方法）	**0.00570552**	0.0000041244448	0.00030422	**0.00019776**
提升（%）	2.14%	-5,810.6%	-1.46%	40.54%

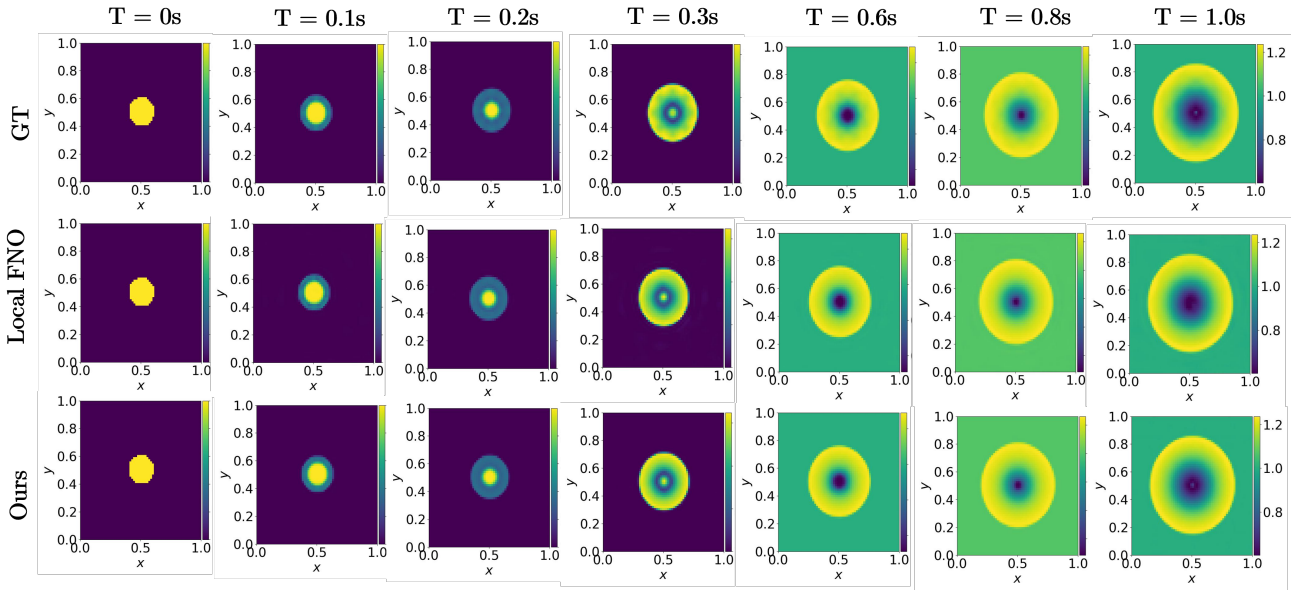

图 *4.* 二维浅水任务中局部傅里叶神经算子与我们的方法的定性比较。

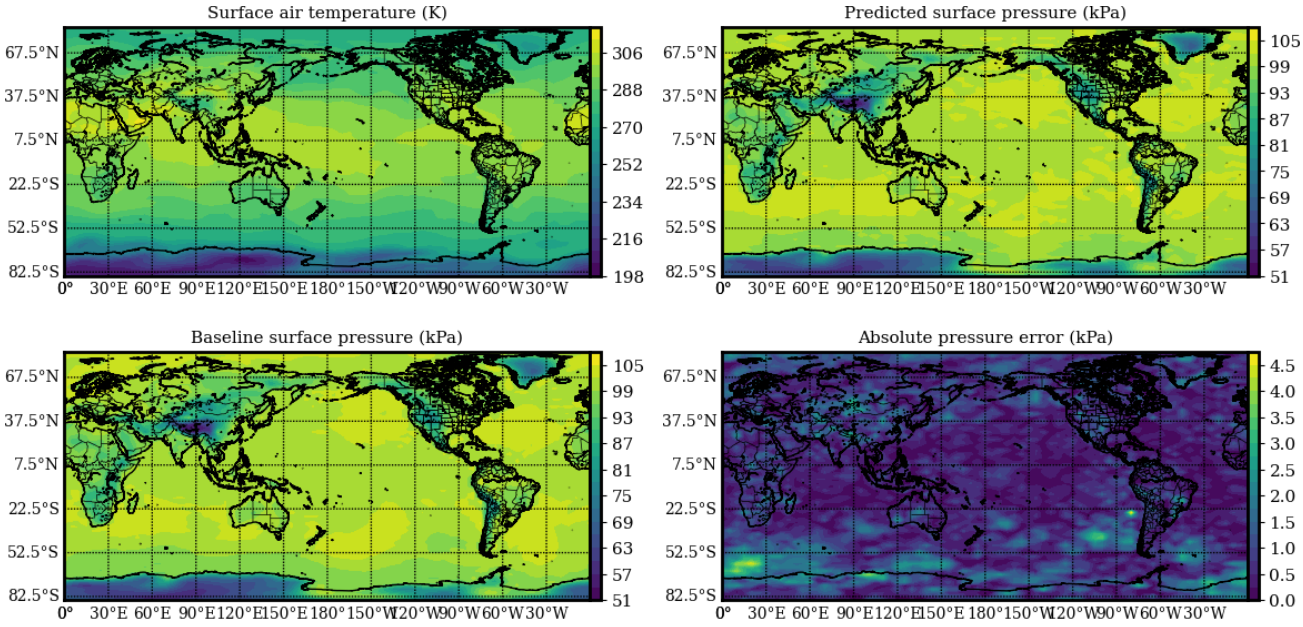

图 *5.*气候建模基准上的全分辨率比较。图中展示了输入温度场、模型预测、真实值，以及相对于参考解的绝对误差。

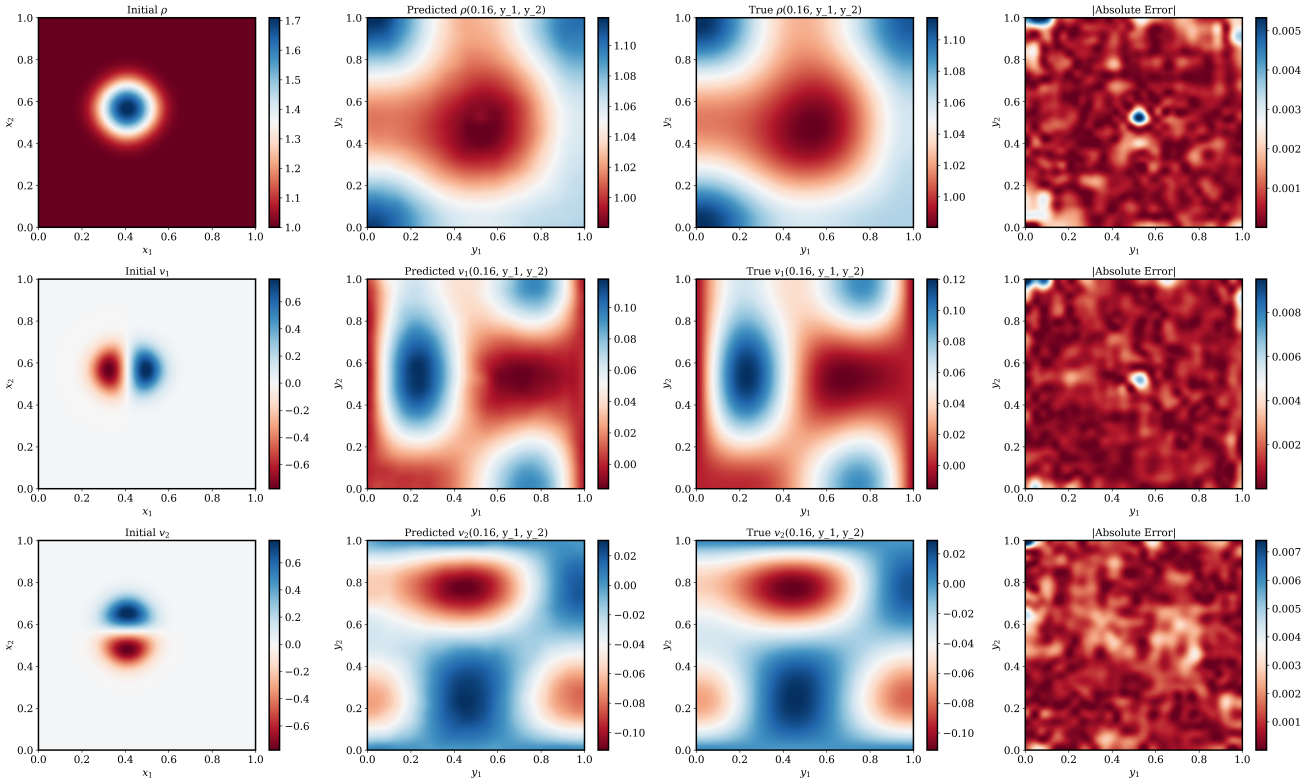

图 *6*。浅水基准上的定性比较。图中展示了初始条件、预测解和真实解，以及 t=0.16s 时的绝对误差。

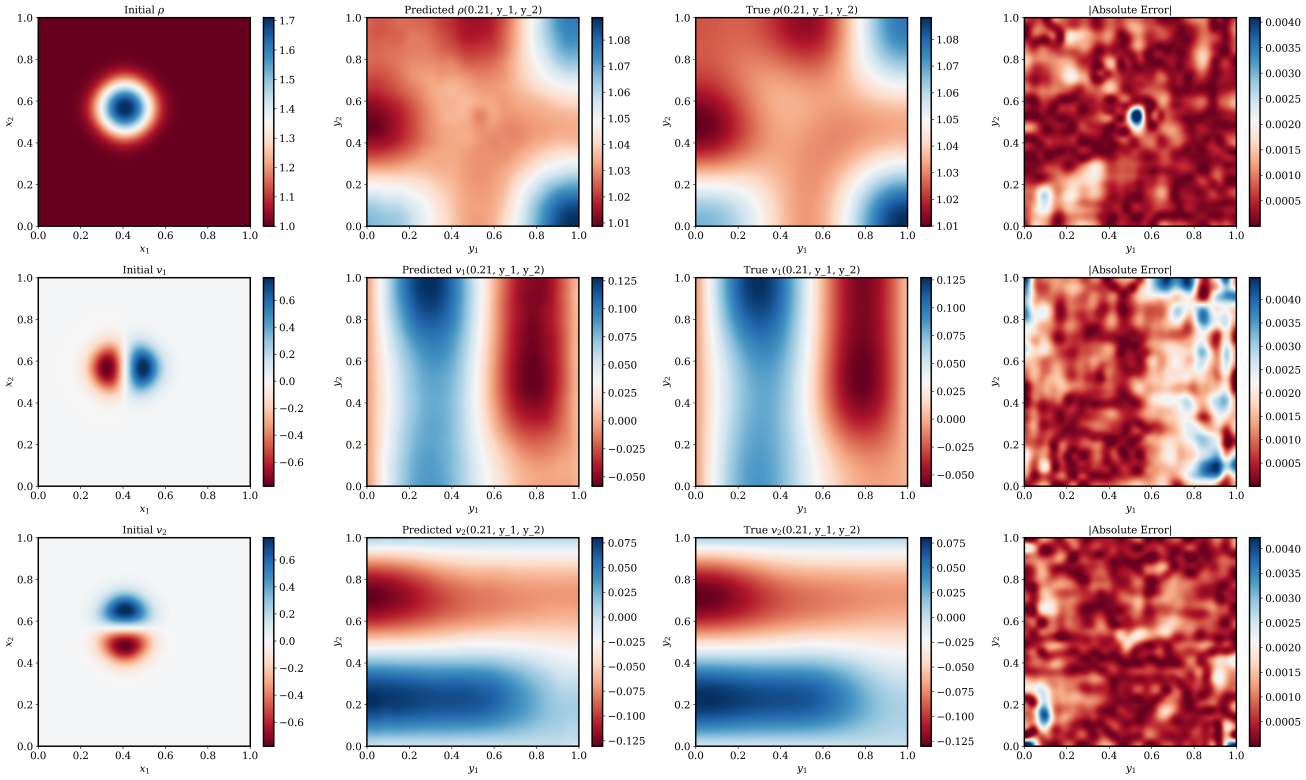

图 7。浅水基准上的定性比较。图中展示了初始条件、预测解和真实解，以及 t=0.21s 时的绝对误差。

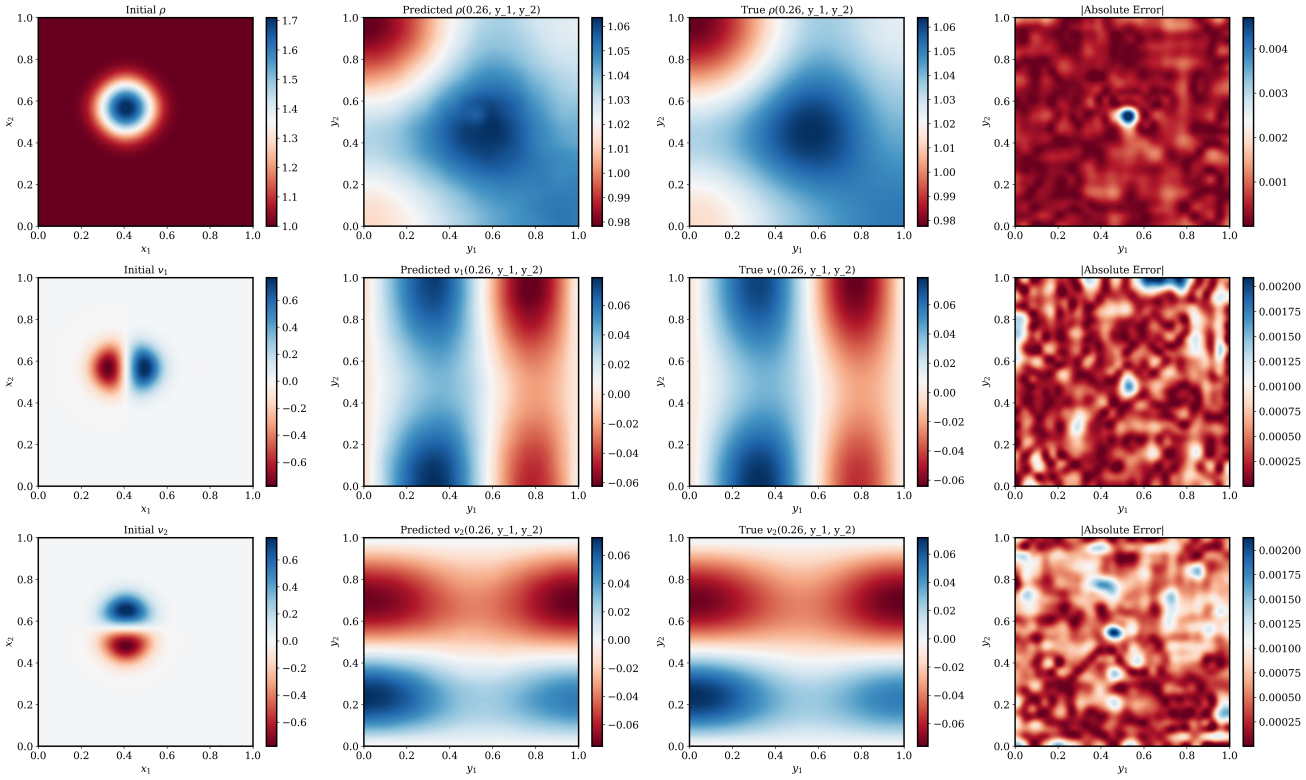

图 *8*。浅水基准上的定性比较。展示了初始条件、预测解和真实解，以及 t=0.26s 时的绝对误差。