arXiv:2304.02282v1 [math.NA] 5 Apr 2023

# 关于遵循因果律的物理信息神经网络训练最优损失函数

**瓦西里·A·埃斯金***下诺夫哥罗德大学无线电物理系，俄罗斯下诺夫哥罗德，603950；Manpower IT Solutions，俄罗斯下诺夫哥罗德，603140
vasiliy.eskin@gmail.com

**丹尼尔·V·达维多夫**
俄罗斯科学院机械工程研究所，俄罗斯下诺夫哥罗德，603155；
Manpower IT Solutions，俄罗斯下诺夫哥罗德，603140
davidovdan27@yandex.ru

**叶卡捷琳娜·D·叶戈罗娃**
华为下诺夫哥罗德研究中心，俄罗斯下诺夫哥罗德；俄罗斯科学院应用物理研究所，俄罗斯下诺夫哥罗德，603155
egorovaed@yandex.ru

**阿列克谢·O. 马尔哈诺夫**华为下诺夫哥罗德研究中心，俄罗斯下诺夫哥罗德 alexey.malkhanov@gmail.com

**米哈伊尔·A. 阿胡科夫**
Manpower IT Solutions，俄罗斯下诺夫哥罗德，603140
ma.akhukov@yandex.ru

**米哈伊尔·E. 斯莫尔卡洛夫**
斯科尔科沃科技研究院，俄罗斯莫斯科；
华为下诺夫哥罗德研究中心，俄罗斯下诺夫哥罗德 smorkalovme@gmail.com

2023年4月6日

**摘要**

本文提出一种方法，可将包含初始条件和边界条件的微分方程问题转化为仅由微分方程描述的问题。将这一修改后的问题用于物理信息神经网络（PINNs）方法，其优势在于可以将损失函数表示为仅与微分方程相关的单项形式，从而无需调整与边界条件和初始条件相关项的缩放系数。本文对遵循因果关系的加权损失函数进行了修改，并推导出基于广义函数的新型加权损失函数。针对若干问题开展了数值实验，结果表明所提方法具有较高精度。

关键词 深度学习 · 物理信息神经网络 · 偏微分方程 · 预测建模 · 计算物理学 · 非线性动力学

*通讯作者：托尔：Vasiliy Alekseevich Es'kin (vasiliy.eskin@gmail.com)

## 1 引言

过去十年以机器学习领域取得的重大成就而著称，例如深度学习 [1]。训练方法的发展和深度神经网络架构的改进，促成了许多应用的诞生；在许多过去只有人类能够完成的活动领域，这些应用已达到或超过普通人的水平 [2–4]。其中，可以区分出机器翻译系统 [5–7], 基于描述生成图像的系统 [8, 9], 游戏系统 [10, 11] 以及文本生成器 [12]。当前深度学习领域的革命也影响了与科学知识获取和使用相关的人类活动领域。用于解决科学问题的辅助系统可以分为以下几类：使用先前获得的数据（来自实验或计算）进行训练并据此作出预测的系统 [13–17], ；利用既有物理定律的系统 [18–20] ；以及同时采用这两种方法的系统 [21, 22]。第一类包括广为人知的AlphaFold [23] ，以及知名度稍低的激光辐射与介质相互作用分析系统 [24, 25]。数据驱动方法不在本文考察范围内，我们将专门关注第二类方法。利用物理定律约束神经网络的初步思想，早在上世纪的研究中就已得到探索 [26, 27] 。不过，近代该领域的研究是在几年前借助现代计算工具开展的，具体体现在物理信息神经网络（PINN）方面 [18] 。在PINN方法中，针对由物理方程以及初始条件和边界条件描述的特定问题，训练神经网络来近似物理量关于空间和时间坐标的依赖关系，有时还会使用来自计算或实验的数据。近年来，这种方法已被用于解决由常微分方程 [18, 21, 22, 28], 积分-微分方程 [29], 非线性偏微分方程 [18, 21, 22, 30–32], 含噪声数据的偏微分方程（PDE） [33], 等描述的广泛问题 [34, 35], , 涉及热力学 [32, 36], 流体力学 [30, 31, 37, 38], 力学 [39], 电动力学 [32, 40],地球物理学 [41], 金融 [29], 等不同知识领域。

尽管 PINN 已取得成功，但针对每个新问题，通常仍需调整学习超参数和神经网络架构。具体而言，神经网络的训练归结为最小化损失函数，其中包含与微分方程、初始条件和边界条件相关的项 [18, 19, 21]。在实践中，为了使训练过程收敛，必须调整上述各项的权重系数。这种调整要么依靠经验，要么通过求取最优值来完成 [42] ，而后者会带来显著的计算成本。在许多任务中，不仅要对这些不同的项进行加权，还要对感兴趣区域内选取的各个配置点进行加权 [43]。这种加权要么基于总体损失函数的最小化，要么依据某些行为特征，尤其是因果性原则 [19], ；该原则已被有效应用于多种任务，并被积极用于改进其他方法、加快其收敛速度 [44]。尽管取得了这些成功，PINN 方法仍在不断发展；其进一步发展需要针对具体问题，采用清晰、透明的方法来确定总体损失函数各组成项的权重系数以及神经网络的架构。

本文致力于开发并应用一系列技术，以提高基于PINN方法训练的神经网络的准确性。我们将简要列出本文的贡献：

1. 将原始问题重新表述为仅利用微分方程的形式；原始问题由微分方程以及初始条件和边界条件构成。
2. 基于重新表述的问题，提出一种仅含一个主要项的损失函数。 3. 此外，基于因果性原理推导出修正后的损失函数，以考虑将其用于大量时间切片的可能性。 4. 在广义函数形式体系（Heaviside阶跃函数和Dirac δ函数）下，提出基于因果性和时空局部性原理的损失函数。 5. 针对上述技术，使用具有齐次和非齐次激活函数的全连接神经网络开展了多项实验。

本文结构如下。第 2 节介绍原始 PINN 方法，并提出对问题和损失函数进行修改的建议。第 3 节在广义函数形式体系的框架下，介绍因果训练的原始描述、对因果训练的修改，以及将其改进为空间—时间训练的方法。第 4 节通过数值实验展示所提出方法的准确性和效率。最后，第 5 节给出总结性意见。

## 2 问题表述

### 2.1 物理信息神经网络（PINNs）

考虑非线性偏微分方程（PDE），它们依赖于时间 $t$ 和空间坐标 $\mathbf{x}$，通常具有以下形式

$$\mathbf{u}_t + \mathcal{N}[\mathbf{u}] = 0, \quad t \in [0,T], \quad \mathbf{x} \in \Omega, \tag{1}$$

满足初始条件和边界条件

$$\mathbf{u}(0,\mathbf{x}) = \mathbf{g}(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{2}$$

$$\mathcal{B}[\mathbf{u}] = 0, \quad t \in [0,T], \quad \mathbf{x} \in \partial\Omega, \tag{3}$$

其中，$\mathbf{u}(t,\mathbf{x})$ 表示由方程（1）的偏微分方程组控制的隐式解，$\mathbf{u}_t$ 表示时间导数，$\mathcal{N}$ 表示非线性微分算子，$\mathcal{B}$ 表示与边界条件（狄利克雷、诺伊曼、罗宾、周期性条件）相对应的边界算子，$\mathbf{g}(\mathbf{x})$ 表示$\mathbf{u}$的初始分布，$\Omega$ 和 $\partial\Omega$ 分别表示空间域及其边界。

根据 PINN 方法 [18]，（该方法建立在通用逼近定理 [45], 的基础上），其目标是利用深度神经网络$\mathbf{u}_{\boldsymbol{\theta}}(t,\mathbf{x})$逼近未知解$\mathbf{u}(t,\mathbf{x})$，其中 $\boldsymbol{\theta}$ 表示网络的所有可训练参数（例如权重和偏置）。寻找最优参数是一个优化问题，需要定义损失函数，使其最小值对应于偏微分方程的解。物理信息模型通过最小化复合损失函数进行训练；该函数由问题域及其边界上的微分方程局部残差构成，如下所示：

$$\mathcal{L}(\boldsymbol{\theta}) = \lambda_{ic}\mathcal{L}_{ic}(\boldsymbol{\theta}) + \lambda_{bc}\mathcal{L}_{bc}(\boldsymbol{\theta}) + \lambda_r\mathcal{L}_r(\boldsymbol{\theta}), \tag{4}$$

其中

$$\mathcal{L}_{ic}(\boldsymbol{\theta}) = \frac{1}{N_{ic}}\sum_{i=1}^{N_{ic}}\left|\mathbf{u}_{\boldsymbol{\theta}}\left(0,\mathbf{x}_i^{(ic)}\right) - \mathbf{g}\left(\mathbf{x}_i^{(ic)}\right)\right|^2, \tag{5}$$

$$\mathcal{L}_{bc}(\boldsymbol{\theta}) = \frac{1}{N_{bc}}\sum_{i=1}^{N_{bc}}\left|\mathcal{B}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]\left(t_i^{(bc)},\mathbf{x}_i^{(bc)}\right)\right|^2, \tag{6}$$

$$\mathcal{L}_r(\boldsymbol{\theta}) = \frac{1}{N_r}\sum_{i=1}^{N_r}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]\left(t_i^{(r)},\mathbf{x}_i^{(r)}\right)\right|^2, \tag{7}$$

$$\mathcal{R}[\mathbf{u}] := \mathbf{u}_t + \mathcal{N}[\mathbf{u}(t,\mathbf{x})] \tag{8}$$

这里，$\left\{\mathbf{x}_i^{(ic)}\right\}_{i=1}^{N_{ic}}$、$\left\{t_i^{(bc)},\ \mathbf{x}_i^{(bc)}\right\}_{i=1}^{N_{bc}}$ 和 $\left\{t_i^{(r)},\ \mathbf{x}_i^{(r)}\right\}_{i=1}^{N_r}$ 分别是与初始条件域、边界条件域和 PDE 域对应的点集。这些点可以是固定网格的顶点，也可以在每次梯度下降迭代时随机采样。输入变量（$t$ 和$\mathbf{x}$）及网络参数 $\theta$ 所需的所有梯度都可以通过自动微分 [46] 高效计算，其精度由计算系统的精度决定。请注意，超参数 $\lambda_{ic}$、$\lambda_{bc}$ 和 $\lambda_r$ 允许分别调整各损失项的学习率，以改善模型 [47,48]的收敛性。

优化问题可定义如下

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}), \tag{9}$$

其中，$\boldsymbol{\theta}^*$ 是使精确的未知解$\mathbf{u}$ 与近似解$\mathbf{u}_{\boldsymbol{\theta}^*}$之间的偏差最小的神经网络最优参数。本文所采用的通用 PINN 方法示意图如图 1 所示。

让我们重写损　失分量 (5)–(7) 的连续形式，以便进一步分析。这些表示为　e

$$\mathcal{L}_{ic}(\boldsymbol{\theta}) = \frac{1}{V_\Omega}\int_\Omega \left|\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x}) - \mathbf{g}(\mathbf{x})\right|^2 d\mathbf{x}, \tag{10}$$

$$\mathcal{L}_{bc}(\boldsymbol{\theta}) = \frac{1}{T}\int_0^T \left|\mathcal{B}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x})\right|^2 dt, \quad \mathbf{x} \in \partial\Omega, \tag{11}$$

$$\mathcal{L}_r(\boldsymbol{\theta}) = \frac{1}{T}\frac{1}{V_\Omega}\int_0^T dt\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x})\right|^2, \tag{12}$$

其中，$V_\Omega$ 是区域的“体积”。　Ω

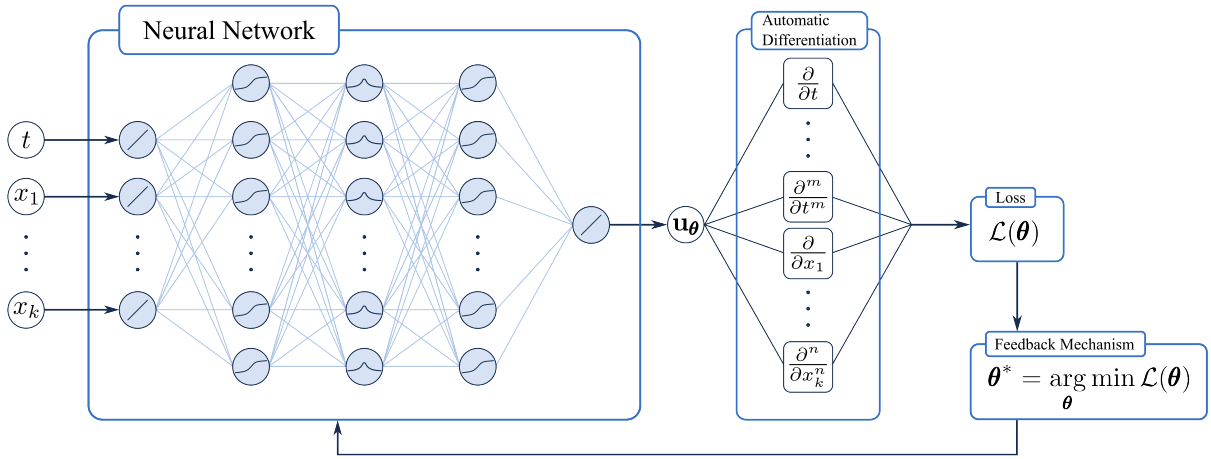


图 1：通用 PINN 方法示意图。

### 2.2 问题的重新表述

#### 2.2.1 时域问题

我们将问题 (1)–(3) 修改为一个不含初始条件的问题。为此，我们将时间区间扩展至从无限久远的过去到时刻 $T$。在时间区间（$-\infty$， 0] 内，假设给定过程 (U) 的值在时间上保持恒定，其空间分布由初始分布 (2) 描述。U的值具有以下形式：

$$\mathbf{U} = \mathbf{g}(\mathbf{x})\chi(-t) + \mathbf{u}(t,\mathbf{x})\chi(t), \quad t \in (-\infty, T], \tag{13}$$

其中，$\mathbf{u}(t,\mathbf{x})$ 是未知函数，预期为方程组（1）–（3）的解；$\chi(t)$ 是 Heaviside 阶跃函数 [49]，定义为

$$\chi(t) = \begin{cases} 0, & t < 0, \\ 1/2, & t = 0, \\ 1, & t > 0. \end{cases} \tag{14}$$

为了使新函数 $\mathbf{U}$ 成为时间区间（$-\infty, 0]$ 内的解，初始方程（1）应添加一个附加项。该项相当于一种“力”，会在时刻 $t = 0$瞬间消失。

为不失一般性并简化问题，我们考虑无界域中的问题，即不施加边界条件。因此，问题 (1)–(3) 可简化为以下微分方程

$$\mathbf{U}_t + \mathcal{N}[\mathbf{U}] = \mathcal{N}[\mathbf{g}(\mathbf{x})]\chi(-t), \quad t \in (-\infty, T], \quad \mathbf{x} \in \Omega, \tag{15}$$

其中，未知函数 $\mathbf{u}(t,\mathbf{x})$ 被封装在 $\mathbf{U}$ 中。与前文一样，我们用深度神经网络 $\mathbf{U}_{\boldsymbol{\theta}}(t,\mathbf{x}) = \mathbf{g}(\mathbf{x})\chi(-t) + \mathbf{u}_{\boldsymbol{\theta}}(t,\mathbf{x})\chi(t)$ 近似未知解 $\mathbf{U}(t,\mathbf{x})$。

在该问题表述中，问题 (15) 的损失函数仅包含与微分方程残差对应的项：

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_r(\boldsymbol{\theta}), \tag{16}$$

其中

$$\mathcal{L}_r(\boldsymbol{\theta}) = \frac{1}{T}\frac{1}{V_\Omega}\int_{-\infty}^{T} dt \int_\Omega d\mathbf{x} \left|\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t,\mathbf{x})\right|^2, \tag{17}$$

$$\mathcal{R}^*\left[\mathbf{u}, \mathbf{g}\right] := \mathbf{u}_t + \mathcal{N}\left[\mathbf{u}(t,\mathbf{x})\right] - \mathcal{N}[\mathbf{g}(\mathbf{x})]\chi(-t). \tag{18}$$

将（17）中的被积函数展开，可得

$$\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t,\mathbf{x}) = \begin{cases} \mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x}) + \left[\mathbf{u}_{\boldsymbol{\theta}}(t,\mathbf{x}) - \mathbf{g}(\mathbf{x})\right]\delta(t), & t \geq 0, \\ 0 & t < 0. \end{cases} \tag{19}$$

这里 $\delta(t)$ 是狄拉克 δ 一个函数 [50]。我们使用 δ函数和海维赛德阶跃函数的以下性质 [50 ]

$$\delta(x)=\frac{\partial\chi(x)}{\partial x},\quad \chi(-x)=1-\chi(x). \tag{20}$$

将（19）代入（17），可得到损失函数的如下表达式

$$\begin{aligned}\mathcal{L}_r(\boldsymbol{\theta})=&\frac{1}{T}\frac{1}{V_\Omega}\lim_{t\to 0}\delta(t)\int_\Omega d\mathbf{x}\left|\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right|^2\\&+2\frac{1}{T}\frac{1}{V_\Omega}\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](0,\mathbf{x})\left[\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right]\right|\\&+\frac{1}{T}\frac{1}{V_\Omega}\int_0^T dt\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x})\right|^2.\end{aligned} \tag{21}$$

这里我们考虑了 δ函数的一个性质：它在有限和中占主导地位，即 $|a\delta(x)+b|^2\simeq|a|^2\delta^2(x)+2|ab|\delta(x)+|b|^2$。

### 2.3 优化问题的重新表述

然后我们应用狄拉克 δ函数的众所周知的表示式 [50]

$$\delta(x)=\lim_{\sigma\to 0}\frac{\sigma}{x^2+\sigma^2}. \tag{22}$$

我们得到式（9）的损失函数如下形式

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta})=&\frac{1}{T}\frac{1}{V_\Omega}\int_0^T dt\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x})\right|^2\\&+\frac{1}{T}\frac{1}{V_\Omega}\left\{2\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](0,\mathbf{x})\left[\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right]\right|\right.\\&\left.+\lim_{\substack{\sigma\to 0\\ t\to 0}}\frac{\sigma}{t^2+\sigma^2}\int_\Omega d\mathbf{x}\left|\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right|^2\right\}.\end{aligned} \tag{23}$$

为了从连续时间转到离散时间表示，我们将时间区间 $[0,T]$ 划分为 $N_t$ 个区间，并对该网格采用步长为 $T/N_t$ 的矩形积分公式。我们取 $\sigma=T/N_t$ 作为时间域中δ函数的近似。由于这种δ函数表示保留了其基本性质 [50] $[\int_{-\infty}^{\infty}f(x)\delta(x-a)dx=f(a)]$，因此适用于数值积分。最终得到如下损失函数

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta})=&\frac{1}{N_t}\frac{1}{V_\Omega}\sum_{i=0}^{N_t}\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t_i,\mathbf{x})\right|^2\\&+\frac{1}{T}\frac{1}{V_\Omega}\left\{2\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t_0,\mathbf{x})\left[\mathbf{u}_{\boldsymbol{\theta}}(t_0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right]\right|\right.\\&\left.+\frac{N_t}{T}\int_\Omega d\mathbf{x}\left|\mathbf{u}_{\boldsymbol{\theta}}(t_0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right|^2\right\}.\end{aligned} \tag{24}$$

这里我们取 $t_0=0$。注意，式（24）第三项的乘数 $N_t$ 与在[19,20,51] 中针对 $N_t=100$情形通过经验找到的系数 $\lambda_{ic}$ 一致。

采用这种表示时，δ函数的近似与数值积分方案相关联。不过，对于许多任务，也可以构造不依赖积分规则的δ函数近似。为此，式（22）中的值 $\sigma$ 可取为初始条件趋于零的估计弛豫时间，也可取为将初始条件的值加倍或减半。请注意，所得的δ函数近似可作为权重 $\lambda_{ic}$，用于由损失项（5）–（7）描述的原始训练方案。第4.3.1节给出了针对特定问题推导此类近似的示例。

#### 2.3.1 MAE 损失

有趣的是，如果在优化问题（9）中不使用均方误差损失，而改用平均绝对误差损失，则无需对狄拉克δ函数作额外表示，并且我们得到的不是损失（23），而是如下函数

$$\mathcal{L}(\boldsymbol{\theta})=\frac{1}{T}\frac{1}{V_\Omega}\left(\int_0^T dt\int_\Omega d\mathbf{x}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,\mathbf{x})\right|+\int_\Omega d\mathbf{x}\left|\mathbf{u}_{\boldsymbol{\theta}}(0,\mathbf{x})-\mathbf{g}(\mathbf{x})\right|\right). \tag{25}$$

根据我们的实验，尽管方程（25）十分简单，但使用这种损失函数训练神经网络时，向所需解的收敛极其微弱。因此，在第4节给出的所有实验中，均使用了均方误差损失。

### 2.4 空间—时间区域中的问题

我们可以用同样的方法重新表述空间域中的问题。为使表述更加清晰简洁，且不失一般性，我们将考虑仅依赖一个空间坐标 $x$的问题。

设空间域由坐标 $x$ ($x \in [X_1, X_2]$) 确定，问题 (1)–(3) 满足以下边界条件

$$\mathbf{u}(t, X_1) = \mathbf{f_1}(t), \quad \mathbf{u}(t, X_2) = \mathbf{f_2}(t). \tag{26}$$

此时，(1)–(3) 的解经人为扩展后具有以下形式

$$\begin{aligned} \mathbf{U} = {} & \mathbf{g}(x)\chi(-t) + \mathbf{u}(t,x)\chi(t)\left[\chi(x - X_1) - \chi(x - X_2)\right] \\ & + \left[\mathbf{f_1}(t)\chi(X_1 - x) + \mathbf{f_2}(t)\chi(x - X_2)\right]\chi(t), \\ & \text{for the domain } t \in (-\infty, T], \quad x \in (-\infty, \infty). \end{aligned} \tag{27}$$

在这种情况下，问题 (1)–(3) 可化为以下微分方程

$$\begin{aligned} \mathbf{U}_t + \mathcal{N}[\mathbf{U}] = {} & \mathcal{N}[\mathbf{g}(x)]\chi(-t) \\ & + \left[\frac{\partial \mathbf{f_1}(t)}{\partial t}\chi(X_1 - x) + \frac{\partial \mathbf{f_2}(t)}{\partial t}\chi(x - X_2)\right]\chi(t) \\ & + \left\{\mathcal{N}\left[\mathbf{f_1}(t)\right]\chi(x - X_1) + \mathcal{N}\left[\mathbf{f_2}(t)\right]\chi(X_2 - x)\right\}\chi(t), \quad t \in (-\infty, T], \quad x \in (-\infty, \infty). \end{aligned} \tag{28}$$

值 $\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t, x)$ 的形式与 (19) 类似

$$\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t,x) = \begin{cases} \mathcal{R}\left[\mathbf{U}^*_{\boldsymbol{\theta}}\right](t,x) + \left[\mathbf{u}_{\boldsymbol{\theta}}(t,x) - \mathbf{g}(x)\right]\delta(t) \\ \quad - \left[\frac{\partial \mathbf{f_1}(t)}{\partial t}\chi(X_1 - x) + \frac{\partial \mathbf{f_2}(t)}{\partial t}\chi(x - X_2)\right] \\ \quad - \left\{\mathcal{N}\left[\mathbf{f_1}(t)\right]\chi(x - X_1) + \mathcal{N}\left[\mathbf{f_2}(t)\right]\chi(X_2 - x)\right\}, & t \geq 0, \quad x \in [X_1, X_2], \\ 0, & t < 0, \quad \text{or } x \notin [X_1, X_2], \end{cases} \tag{29}$$

其中

$$\mathbf{U}^*_{\boldsymbol{\theta}} = \mathbf{u}_{\boldsymbol{\theta}}(t,x)\left[\chi(x - X_1) - \chi(x - X_2)\right] + \mathbf{f_1}(t)\chi(X_1 - x) + \mathbf{f_2}(t)\chi(x - X_2). \tag{30}$$

$\mathcal{R}\left[\mathbf{U}^*_{\boldsymbol{\theta}}\right]$ 关于 $\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]$ 的表示形式与方程 (19) 类似，具体取决于微分方程的类型。这种表示形式的示例见第 4.2.1 节。

## 3 物理信息神经网络的因果训练

文献 [19] 提出，重新表述损失函数，使其在借助 PINN 求解 PDE 时遵循物理因果性：

$$\mathcal{L}_r(\boldsymbol{\theta}) = \frac{1}{N_t}\sum_{i=0}^{N_t} w_i \mathcal{L}_r(t_i, \boldsymbol{\theta}), \tag{31}$$

其中权重 $w_i$ 为

$$w_i = \exp\left[-\varepsilon \sum_{k=0}^{i-1} \mathcal{L}_r(t_k, \boldsymbol{\theta})\right], \quad \text{for } i = 1, 2, 3, \ldots, N_t, \text{and } w_0 = 1, \tag{32}$$

其中，$\varepsilon$ 是控制权重陡峭程度的因果参数 $w_i$。总损失（23）的时间残差损失 $[\mathcal{L}_r(t_i, \boldsymbol{\theta})]$为

$$\begin{aligned} \mathcal{L}_r(t_i, \boldsymbol{\theta}) = {} & \frac{1}{V_\Omega}\int_\Omega d\mathbf{x} \Big\{ \left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t_i, \mathbf{x})\right|^2 \\ & + 2\frac{N_t}{T}\delta_{0i}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t_0, \mathbf{x})\left[\mathbf{u}_{\boldsymbol{\theta}}(t_0, \mathbf{x}) - \mathbf{g}(\mathbf{x})\right]\right| \\ & + \frac{N_t^2}{T^2}\delta_{0i}\left|\mathbf{u}_{\boldsymbol{\theta}}(t_0, \mathbf{x}) - \mathbf{g}(\mathbf{x})\right|^2 \Big\}, \end{aligned} \tag{33}$$

$\delta_{0i}$ 是克罗内克符号。

当区域 Ω 由单个空间维度 $x\,(x \in [X_1, X_2],\, V_\Omega = X_2 - X_1)$ 构成时，在具有 $N_x$ 个节点的规则网格下，时间残差损失具有离散形式

$$\begin{aligned}\mathcal{L}_r\left(t_i, \boldsymbol{\theta}\right) =& \frac{1}{N_x}\sum_{n=0}^{N_x}\Big\{\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]\left(t_i, x_n\right)\right|^2 \\ &+ 2\frac{N_t}{T}\delta_{0i}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]\left(t_0, x_n\right)\left[\mathbf{u}_{\boldsymbol{\theta}}\left(t_0, x_n\right) - \mathbf{g}\left(x_n\right)\right]\right| \\ &+ \frac{N_t^2}{T^2}\delta_{0i}\left|\mathbf{u}_{\boldsymbol{\theta}}\left(t_0, x_n\right) - \mathbf{g}\left(x_n\right)\right|^2\Big\}. \end{aligned} \tag{34}$$

由此得到的加权残差损失为

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right) = \frac{1}{N_t}\sum_{i=0}^{N_t}\exp\left[-\varepsilon\sum_{k=0}^{i-1}\mathcal{L}_r\left(t_k, \boldsymbol{\theta}\right)\right]\mathcal{L}_r\left(t_i, \boldsymbol{\theta}\right). \tag{35}$$

[19] 的作者认识到，只有当所有先前的残差 $\{\mathcal{L}_r(t_k,\ \boldsymbol{\theta})\}_{k=0}^{i}$ 在 $t_i$ 之前都得到适当最小化时，权重 $w_i$ 才应当较大，从而允许最小化 $\mathcal{L}_r(t_i,\ \boldsymbol{\theta})$；反之亦然。因此，除非所有先前的残差 $\{\mathcal{L}_r(t_k,\ \boldsymbol{\theta})\}_{k=0}^{i-1}$ 都下降到足够小的值，使得 $w_i$ 足够大，否则不会最小化 $\mathcal{L}_r(t_i,\ \boldsymbol{\theta})$。

### 3.1 物理信息神经网络因果训练方法的修改

尽管因果训练方法显著改善了神经网络学习的收敛性，但随着时间网格节点数量的增加，这种方法难以扩展。这是因为加权函数（32）的指数项包含累积和。随着 $N_t$ 的增大，该累积和也随之增大，这种行为会阻碍神经网络在较大 $t$ 下的学习。请注意，在一系列情况下，与其他加权方法相比，这一特性可能使训练获得最佳收敛效果。

尝试将残差损失（35）写成连续形式时，这一缺陷就显而易见了。

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right) = \frac{1}{T}\int_0^T \lim_{N_t\to\infty}\left[\exp\left(-\varepsilon\frac{N_t}{t}\int_0^t \mathcal{L}_r\left(t', \boldsymbol{\theta}\right)dt'\right)\right]\mathcal{L}_r\left(t, \boldsymbol{\theta}\right)dt. \tag{36}$$

由于 $N_t$ 无限增大，该积分对任意 $\varepsilon$ 和时间都趋于零。这一缺陷使得无法使用基于积分和微分学的标准方法对损失函数进行数学分析。

为缓解上述缺陷，可以将残差损失（35）改写为以下形式

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right) = \frac{1}{N_t}\sum_{i=0}^{N_t} w_i\mathcal{L}_r\left(t_i, \boldsymbol{\theta}\right), \tag{37}$$

$$w_i = \exp\left[-\varepsilon s_i\right], \tag{38}$$

$$s_i = \frac{1}{i}\sum_{k=0}^{i-1}\mathcal{L}_r\left(t_k, \boldsymbol{\theta}\right). \tag{39}$$

残差损失（37）的连续形式不为零，其表达式如下

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right) = \frac{1}{T}\int_0^T dt\exp\left[-\frac{\varepsilon}{t}\int_0^t dt'\mathcal{L}_r\left(t', \boldsymbol{\theta}\right)\right]\mathcal{L}_r\left(t, \boldsymbol{\theta}\right). \tag{40}$$

### 3.2 物理信息神经网络因果训练的替代方法

### 3.3 Heaviside 加权残差损失

我们来考虑使用 Heaviside 阶跃函数加权的残差损失

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right) = \frac{1}{T}\int_0^T \left\{1 - \chi\left[S(t)\right]\right\}\mathcal{L}_r\left(t, \boldsymbol{\theta}\right)dt, \tag{41}$$

其中

$$S(t)=\frac{1}{t}\int_0^t \mathcal{L}_r\left(t',\boldsymbol{\theta}\right)dt'. \tag{42}$$

在这种情况下，方程 (41) 被积函数中的项 $1-\chi(x)$ 在条件 $x\leq 0$ 下并不趋于零。这意味着因果性条件得到满足。为计算损失，我们采用如下众所周知的 Heaviside 阶跃函数表示 [50]

$$\chi(x)=\lim_{\varepsilon\to\infty}\frac{1}{1+\exp\left(-2\varepsilon x\right)}. \tag{43}$$

方程 (43) 导出如下残差损失表示

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right)=\frac{1}{T}\int_0^T dt\frac{1}{1+\exp\left[2\varepsilon S(t)\right]}\mathcal{L}_r\left(t,\boldsymbol{\theta}\right), \tag{44}$$

其具有带权重的离散形式 (31)

$$w_i=\left[1+\exp\left(2\varepsilon s_i\right)\right]^{-1}. \tag{45}$$

### 3.3.1 Dirac 加权残差损失

让我们考虑加权残差损失的极端形式

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right)=\frac{1}{T}\int_0^T \delta\left[S(t)\right]\mathcal{L}_r\left(t,\boldsymbol{\theta}\right)dt. \tag{46}$$

这种损失函数有一个显著特性：若尚未先对较小的时间戳进行训练，就会阻止对较大时间戳进行训练。

为继续讨论该损失函数的离散形式，我们将使用以下广为人知的狄拉克δ函数表示 [50]

$$\delta(x)=\lim_{\sigma^2\to 0}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{47}$$

在这种情况下，残差损失的积分形式为

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right)=\frac{1}{T}\int_0^T dt\exp\left\{-\varepsilon\left[\frac{1}{t}\int_0^t dt't'\mathcal{L}_r\left(t',\boldsymbol{\theta}\right)\right]^2\right\}\mathcal{L}_r\left(t,\boldsymbol{\theta}\right). \tag{48}$$

请注意，在内积分中乘以 $t$，进一步增强了权重的因果性。对于等间距时间切片，式（48）的离散形式为

$$\mathcal{L}_r\left(\boldsymbol{\theta}\right)=\frac{1}{N_t}\sum_{i=0}^{N_t} w_i\mathcal{L}_r\left(t_i,\boldsymbol{\theta}\right), \tag{49}$$

$$w_i=\exp\left(-\varepsilon\tilde{s}_i^2\right), \tag{50}$$

$$\tilde{s}_i=\frac{1}{i}\sum_{k=0}^{i-1} k\mathcal{L}_r\left(t_k,\boldsymbol{\theta}\right). \tag{51}$$

推导方程（51）时，我们考虑到了 $t_k=kT/N_t$ 以及 $\tilde{s}_i$的乘数 $T/N_t$在式（50）中已隐含于因果参数 $\varepsilon$ 之中。

### 3.4 将因果训练方法扩展至时空领域

对于问题（28），损失函数可表示如下

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{\theta}\right)=&\frac{1}{T}\frac{1}{V_\Omega}\int_0^T dt\int_\Omega dx\left|\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}^*\right](t,x)\right|^2\\&+\frac{1}{T}\frac{1}{V_\Omega}\left\{2\int_\Omega dx\left|\mathcal{R}^*\left[\mathbf{U}_{\boldsymbol{\theta}}^*\right](0,x)\left[\mathbf{u}_{\boldsymbol{\theta}}\left(0,x\right)-\mathbf{g}\left(x\right)\right]\right|\right.\\&\left.+\lim_{\substack{\sigma\to 0\\ t\to 0}}\frac{\sigma}{t^2+\sigma^2}\int_\Omega dx\left|\mathbf{u}_{\boldsymbol{\theta}}\left(0,x\right)-\mathbf{g}\left(x\right)\right|^2\right\}.\end{aligned} \tag{52}$$

时空域中的加权残差损失可定义如下

$$\mathcal{L}_r^*(\boldsymbol{\theta}) = \frac{1}{N_t}\sum_{i=0}^{N_t} w_i \mathcal{L}_r^*(t_i, \boldsymbol{\theta}), \tag{53}$$

$$w_i = \exp\left[-\varepsilon s_i\right], \quad s_i = \frac{1}{i}\sum_{k=0}^{i-1} \mathcal{L}_r^*(t_k, \boldsymbol{\theta}), \tag{54}$$

其中

$$\begin{aligned}\mathcal{L}_r^*(t_i, \boldsymbol{\theta}) = \frac{1}{N_x}\sum_{n=0}^{N_x}\left(w_n^{(\rightarrow)} + w_n^{(\leftarrow)}\right)\Big\{&\left|\mathcal{R}\left[\mathbf{U}_{\boldsymbol{\theta}}^*\right](t_i, x_n)\right|^2 \\ &+ 2\frac{N_t}{T}\delta_{0i}\left|\mathcal{R}\left[\mathbf{U}_{\boldsymbol{\theta}}^*\right](t_0, x_n)\left[\mathbf{u}_{\boldsymbol{\theta}}(t_0, x_n) - \mathbf{g}(x_n)\right]\right| \\ &+ \frac{N_t^2}{T^2}\delta_{0i}\left|\mathbf{u}_{\boldsymbol{\theta}}(t_0, x_n) - \mathbf{g}(x_n)\right|^2\Big\},\end{aligned} \tag{55}$$

and

$$\begin{aligned}&w_n^{(\rightarrow)} = \exp\left[-\varepsilon s_n^{(\rightarrow)}\right], \quad w_n^{(\leftarrow)} = \exp\left[-\varepsilon s_n^{(\leftarrow)}\right], \\ &s_n^{(\rightarrow)} = \frac{1}{n}\sum_{k=0}^{n-1}\tilde{\mathcal{L}}_r(x_k, \boldsymbol{\theta}), \quad s_n^{(\leftarrow)} = \frac{1}{N_x - n}\sum_{k=0}^{N_x-n-1}\tilde{\mathcal{L}}_r(x_{N_x-k}, \boldsymbol{\theta}).\end{aligned} \tag{56}$$

空间残差损失为

$$\tilde{\mathcal{L}}_r(x_k, \boldsymbol{\theta}) = \frac{1}{N_t}\sum_{i=0}^{N_t}\left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t_i, x_k)\right|^2. \tag{57}$$

这里，上标（$\rightarrow$）和（$\leftarrow$）分别表示沿 $x$ 轴正向和反向的加权（即从 $X_1$ 到 $X_2$，以及从 $X_2$ 到 $X_1$）。

### 3.5 训练过程

基于对问题、损失函数和因果权重所做的上述修改，算法 1 和算法 2 分别给出了 PINN 的因果局部训练算法和时空局部训练算法。

**算法 1：** 在时间域训练物理信息神经网络**数据：** 坐标集合 $(t,\ \mathbf{x})$ 以及分别对应初始条件域和边界条件域中各点的值 $\mathbf{u}$，以及配点的坐标集合**结果：** 训练好的 NN$\mathbf{u}_{\boldsymbol{\theta}}$，用于近似求解 PDE。将时间权重初始化为 $w_i$（取值为 1），因果参数 $\varepsilon$ 和阈值参数 $\delta_w$ 设置为选定值。使用梯度下降算法进行 $N$ 轮迭代，并按如下方式更新参数 $\boldsymbol{\theta}$：**对于** $n = 1,\ \ldots\ldots,\ N$ **执行**1. 使用时间残差损失 (34)，根据公式 (38)、(45) 或 (50) 计算并更新时间权重 $w_i$ **如果** $\min_i\left(\exp\left[-\varepsilon s_i\right]\right) > \delta_w$ **则**$\varepsilon \leftarrow m_\varepsilon \times \varepsilon$2. 根据公式 (16) 和 (31) 计算损失。3. 通过梯度下降更新参数 $\theta$。通过数值实验，我们发现最有前景的超参数为 $\delta_w = 0.99$，$\varepsilon$=$10^{-3}$的初始值

$$\boldsymbol{\theta}_n \leftarrow \boldsymbol{\theta}_{n-1} - \eta\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_{n-1}). \tag{58}$$

，乘法因子 $m_\varepsilon = 2$ · 这里 $\eta$ 是梯度下降的学习率 ·

**算法 2：** T raining of the physics-informed 神经 l networks at spatio-t时间 l domai

**n数据：** 初始条件域和边界条件域中各点的坐标（$t$，$\mathbf{x}$）及其对应的值$\mathbf{u}$，以及配置点的坐标集合

**结果：** 训练好的神经网络$\mathbf{u}_{\boldsymbol{\theta}}$，用于近似求解偏微分方程的解。将时间权重和空间权重 $w_i$、$w_i^{(\rightarrow)}$ 以及 $w_i^{(\leftarrow)}$ 初始化为 1，将因果参数 $\varepsilon$ 和阈值参数 $\delta_w$ 设置为选定值。使用梯度下降算法进行 $N$ 轮迭代，并按如下方式更新参数 $\boldsymbol{\theta}$ **对于** $n = 1, \ldots\ldots, N$ **执行**1. 使用空间残差损失（57），根据方程（56）计算并更新空间权重 $w_i^{(\rightarrow)}$ 和 $w_i^{(\leftarrow)}$。2. 使用时间残差损失（55），根据方程（38）、（45）或（50）计算并更新时间权重 $w_i$ **如果** $\min_i(\exp[-\varepsilon s_i]) > \delta_w$ & $\min_i\left(\exp\left[-\varepsilon s_i^{(\rightarrow)}\right]\right) > \delta_w$ & $\min_i\left(\exp\left[-\varepsilon s_i^{(\leftarrow)}\right]\right) > \delta_w$ **则**$\varepsilon \leftarrow m_\varepsilon \times \varepsilon$3. 根据方程（16）和（54）计算损失。4. 通过梯度下降更新参数 $\theta$

$$\boldsymbol{\theta}_n \leftarrow \boldsymbol{\theta}_{n-1} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}\left(\boldsymbol{\theta}_{n-1}\right). \tag{59}$$

下面说明训练过程的一些细节。

为了节省计算资源并加快沿 $x$轴具有周期边界条件的问题的收敛，我们构造如下形式的输入坐标傅里叶特征嵌入

$$v(x) = [1, \cos(k_x x), \sin(k_x x), \cos(2k_x x), \sin(2k_x x), ..., \cos(mk_x x), \sin(mk_x x)], \tag{60}$$

其中，$k_x = 2\pi/L$、$L$ 是沿 $x$ 方向的最小空间周期，$m$ 是某个非负整数。采用这种嵌入后，任何经过训练的问题解近似函数 $\boldsymbol{u_\theta}[v(x)]$ 都能严格满足周期条件（详情见 [19,52]）。

这些网络通过随机梯度下降进行训练，使用 Adam 优化器 [53]和不同的调度器（详见表 5）。

## 4 数值实验

本节通过若干数值实验展示上述方法的准确性和性能。

为了评估借助 PINN 方法获得的近似解的精度，将神经网络在给定点预测的方程 (1) 的解值，与基于经典高精度数值方法计算的解值进行比较。采用相对总 $\mathbb{L}_2$ 预测误差作为精度度量，可表示为以下关系式

$$\epsilon_{\text{error}} = \left\{\frac{1}{N_e}\sum_{i=1}^{N_e}\left[\mathbf{u}_{\boldsymbol{\theta}}(t_i, \mathbf{x}_i) - \mathbf{u}(t_i, \mathbf{x}_i)\right]^2\right\}^{1/2} \times \left\{\frac{1}{N_e}\sum_{i=1}^{N_e}\left[\mathbf{u}(t_i, \mathbf{x}_i)\right]^2\right\}^{-1/2}, \tag{61}$$

其中，$\{t_i, \mathbf{x}_i\}_{i=1}^{N_e}$ 是从区域 $[0, T] \times \Omega$ 中选取的评估点集合，$\mathbf{u}_{\boldsymbol{\theta}}$ 和 $\mathbf{u}$ 分别表示预测解和参考解。

在我们的实验中，我们使用了 Pytorch [54] 版本 1.12.1，并在配备 Nvidia Tesla V100 GPU 的节点上进行训练。

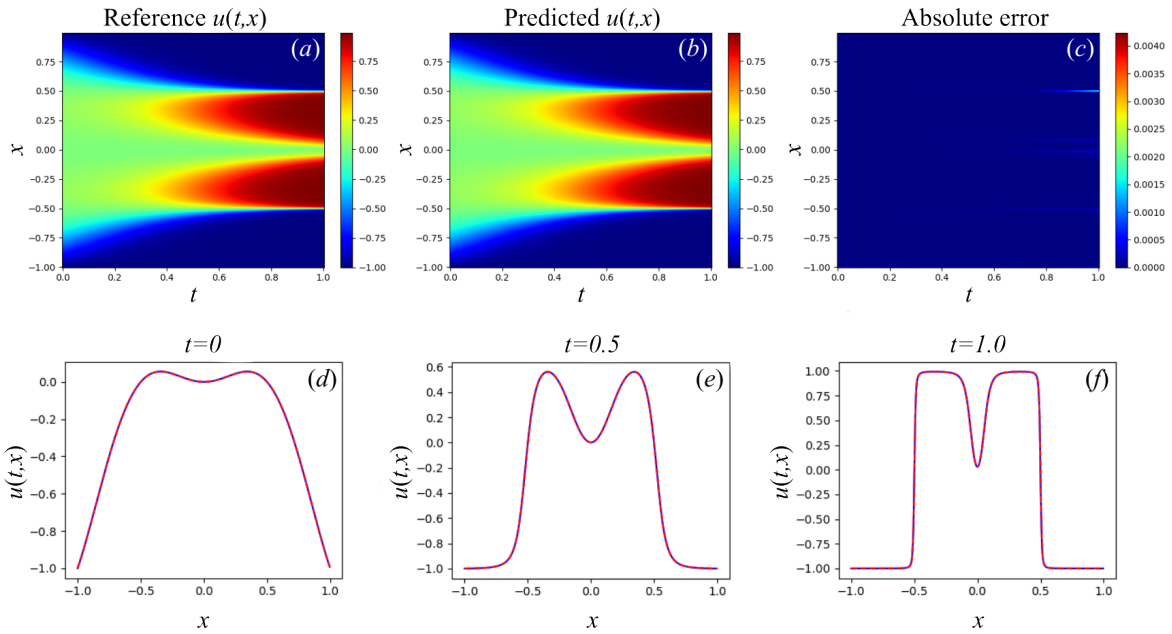


图2：Allen–Cahn方程。(a)为参考解，(b)为基于算法1训练的物理信息神经网络的预测结果，(c)为参考解与预测解之间的绝对差。相对误差$\epsilon_{\text{error}}$为 $6.29 \times 10^{-5}$。(d)、(e)和(f)分别为在 $t = 0.0$、 $t = 0.5$ 和 $t = 1.0$三个时刻快照处预测解（红色虚线）与参考解（蓝色实线）的对比。

### 4.1 Allen–Cahn方程

与 [18, 19] 一样，我们考虑一个由一维 Allen–Cahn 方程描述、具有周期性边界条件的经典相场。

$$
\begin{aligned}
&u_t - 0.0001u_{xx} + 5u^3 - 5u = 0, \quad t \in [0,1], \quad x \in [-1,1],\\
&u(x,0) = x^2\cos(\pi x),\\
&u(t,-1) = u(t,1), \quad u_x(t,-1) = u_x(t,1).
\end{aligned} \tag{62}
$$

我们用一个全连接神经网络 $u_{\boldsymbol{\theta}}$ 表示潜变量 $u$ ，该网络采用双曲正切激活函数，包含4个隐藏层，每层128个神经元。为简便起见，在计算域 $[0,1] \times [-1,1]$中构建尺寸为 $100 \times 256$ 的均匀网格，从而得到 $N_{ic} = 256$ 个初始点和 $N_r = 25600$ 个用于施加PDE残差的配点。嵌入中输入坐标的傅里叶展开由前20个空间谐波（ $m = 20$）确定。对于该问题，采用(16)的对数作为训练 $u_{\boldsymbol{\theta}}$的损失函数。我们选择这种方法，是因为根据我们的实验，它表现出更快、更稳定的收敛。在这种损失函数情形下，优化问题为

$$
\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \log\left[\mathcal{L}(\boldsymbol{\theta})\right]. \tag{63}
$$

参考解可以使用 Chebfun 软件包 [55], 生成，也可以使用[19] 补充材料中的数据集。

实验针对由方程(38)、(45)和(50)所描述的各种权重进行。训练所用的超参数见表5和表6。在表1中，我们还报告了文献 [18–20, 43, 51, 56, 57]中其他方法在Allen–Cahn问题上取得的精度。从表中可以看出，最佳结果由狄拉克δ函数和多层感知器取得，相对 $\mathbb{L}_2$ 误差为 $6.29 \times 10^{-5}$。该实验的结果如图2所示。预测解与真实解几乎完全吻合。

| 方法 | 相对 $\mathbb{L}_2$ 误差 |
|---|---|
| Raissi 等人提出的原始形式 [18] | $4.98 \times 10^{-1}$ |
| 自适应时间采样 [43] | $2.33 \times 10^{-2}$ |
| 自注意力 [56] | $2.10 \times 10^{-2}$ |
| 时间推进 [57] | $1.68 \times 10^{-2}$ |
| 因果训练（多层感知器） [19] | $1.43 \times 10^{-3}$ |
| 带守恒定律的改进多层感知器 [20] | $5.42 \times 10^{-4}$ |
| **因果训练（权重 (38)）** | $\mathbf{5.08 \times 10^{-4}}$ |
| **Heaviside 函数因果训练（权重（45））** | $\mathbf{3.34 \times 10^{-4}}$ |
| 因果训练（修改后的多层感知器） [19] | $1.39 \times 10^{-4}$ |
| DASA-PINN [51] | $8.57 \times 10^{-5}$ |
| **狄拉克δ函数因果训练（权重（50））** | $\mathbf{6.29 \times 10^{-5}}$ |

表1：Allen–Cahn方程：不同方法获得的相对 $\mathbb{L}_2$ 误差

### 4.2 科特韦格-德弗里斯方程

考虑一维科特韦格-德弗里斯方程

$$u_t + \eta u u_x + \mu^2 u_{xxx} = 0, \quad t \in [0,1], \quad x \in [-1,1], \tag{64}$$
$$u(x,\ 0)\ = \cos(\pi x), \tag{65}$$
$$u(t,\ -1)\ = u(t,\ 1)。 \tag{66}$$

我们采用了该问题的经典参数 $\eta = 1$ 和 $\mu = 0.022$ [58]。我们使用具有 $tanh$ 激活函数的全连接神经网络 $u_{\boldsymbol{\theta}}$ 来表示潜变量 $u$，该网络包含 3 个隐藏层，每个隐藏层有 128 个神经元。在计算域 $[0,1] \times [-1,1]$上构建了大小为 $100 \times 512$ 的均匀网格，得到 $N_{ic} = 512$ 个初始点和 $N_r = 51200$ 个配置点，用于算法 1 中施加 PDE 残差，权重由式（38）和基于[19]的权重给出。对于采用权重（50）的算法 2，网格大小为 $100 \times 256$，包含 $N_r = 25600$个配置点。 $\lambda_{ic} = 200$、$\lambda_r = 5$ 用于原始因果训练 [19]。所有场景的训练轮数均为 300000。参考解由伪谱方法 [58]生成。

要使用带有权重 (50) 的算法 2，我们需要针对当前问题推导函数 $\mathcal{R}\left[\mathbf{U}^*_{\boldsymbol{\theta}}\right]$。这一推导将在下一小节中给出。

#### 4.2.1 时空局部训练

为了使用算法 2，我们需要针对 $t \geq 0$，用 $\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right]$ 表示 $\mathcal{R}\left[\mathbf{U}^*_{\boldsymbol{\theta}}\right]$。为此，使用表达式（30）。

$$\mathcal{R}\left[\mathbf{U}^*_{\boldsymbol{\theta}}\right] = \frac{\partial \mathbf{U}^*_{\boldsymbol{\theta}}}{\partial t} + \eta \mathbf{U}^*_{\boldsymbol{\theta}} \frac{\partial \mathbf{U}^*_{\boldsymbol{\theta}}}{\partial x} + \mu^2 \frac{\partial^3 \mathbf{U}^*_{\boldsymbol{\theta}}}{\partial x^3}. \tag{67}$$

$$\begin{aligned}\mathcal{R}\left[\mathbf{U}_{\boldsymbol{\theta}}^{*}\right]=&\left(\frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial t}+\eta \mathbf{u}_{\boldsymbol{\theta}} \frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial x}+\mu^{2} \frac{\partial^{3} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{3}}\right)\left[\chi\left(x-X_{1}\right)-\chi\left(x-X_{2}\right)\right] \\ &+\left[\frac{\partial \mathbf{f}_{\mathbf{1}}(t)}{\partial t} \chi\left(X_{1}-x\right)+\frac{\partial \mathbf{f}_{\mathbf{2}}(t)}{\partial t} \chi\left(x-X_{2}\right)\right] \\ &+\eta \mathbf{f}_{\mathbf{1}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{1}}(t)\right] \delta\left(x-X_{1}\right) \\ &+\eta \mathbf{f}_{\mathbf{2}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{2}}(t)\right] \delta\left(x-X_{2}\right) \\ &+3 \mu^{2}\left[\frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial x} \frac{\partial \delta\left(x-X_{1}\right)}{\partial x}+\frac{\partial^{2} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{2}} \delta\left(x-X_{1}\right)\right] \\ &-3 \mu^{2}\left[\frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial x} \frac{\partial \delta\left(x-X_{2}\right)}{\partial x}+\frac{\partial^{2} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{2}} \delta\left(x-X_{2}\right)\right] \\ &+\mu^{2}\left[\mathbf{u}_{\boldsymbol{\theta}}-\mathbf{f}_{\mathbf{1}}(t)\right] \frac{\partial^{2} \delta\left(x-X_{1}\right)}{\partial x^{2}}+\mu^{2}\left[\mathbf{f}_{\mathbf{2}}(t)-\mathbf{u}_{\boldsymbol{\theta}}\right] \frac{\partial^{2} \delta\left(x-X_{2}\right)}{\partial x^{2}}.\end{aligned} \tag{68}$$

通过应用定义 δ函数导数的基本方程 [50]

$$\int_{-\infty}^{\infty} f(x) \frac{\partial^{n} \delta(x)}{\partial x^{n}} d x=-\int_{-\infty}^{\infty} \frac{\partial f(x)}{\partial x} \frac{\partial^{n-1} \delta(x)}{\partial x^{n-1}} d x, \tag{69}$$

我们可以将方程 (68) 简化为以下形式

$$\begin{aligned}\mathcal{R}\left[\mathbf{U}_{\boldsymbol{\theta}}^{*}\right]=&\left(\frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial t}+\eta \mathbf{u}_{\boldsymbol{\theta}} \frac{\partial \mathbf{u}_{\boldsymbol{\theta}}}{\partial x}+\mu^{2} \frac{\partial^{3} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{3}}\right)\left[\chi\left(x-X_{1}\right)-\chi\left(x-X_{2}\right)\right] \\ &+\left[\frac{\partial \mathbf{f}_{\mathbf{1}}(t)}{\partial t} \chi\left(X_{1}-x\right)+\frac{\partial \mathbf{f}_{\mathbf{2}}(t)}{\partial t} \chi\left(x-X_{2}\right)\right] \\ &+\eta \mathbf{f}_{\mathbf{1}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{1}}(t)\right] \delta\left(x-X_{1}\right) \\ &+\eta \mathbf{f}_{\mathbf{2}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{2}}(t)\right] \delta\left(x-X_{2}\right) \\ &+\mu^{2} \frac{\partial^{2} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{2}}\left[\delta\left(x-X_{1}\right)-\delta\left(x-X_{2}\right)\right].\end{aligned} \tag{70}$$

定义域 $t \geq 0$ 中的表达式 (29) 可表示为

$$\begin{aligned}\mathcal{R}^{*}\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t, x)=&\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t, x)+\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{g}(x)\right] \delta(t) \\ &+\eta \mathbf{f}_{\mathbf{1}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{1}}(t)\right] \delta\left(x-X_{1}\right) \\ &+\eta \mathbf{f}_{\mathbf{2}}(t)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{f}_{\mathbf{2}}(t)\right] \delta\left(x-X_{2}\right) \\ &+\mu^{2} \frac{\partial^{2} \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^{2}}\left[\delta\left(x-X_{1}\right)-\delta\left(x-X_{2}\right)\right].\end{aligned} \tag{71}$$

对于该　周期性边界条件下，最小空间周期为 $X_2 - X_1$，方程 (71) 可简化为　　to

$$\begin{aligned}\mathcal{R}^{*}\left[\mathbf{U}_{\boldsymbol{\theta}}, \mathbf{g}\right](t, x)=&\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t, x)+\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{g}(x)\right] \delta(t) \\ &+2 \eta \mathbf{u}_{\boldsymbol{\theta}}\left(t, X_{2}\right)\left[\mathbf{u}_{\boldsymbol{\theta}}(t, x)-\mathbf{u}_{\boldsymbol{\theta}}\left(t, X_{2}\right)\right] \delta\left(x-X_{1}\right).\end{aligned} \tag{72}$$

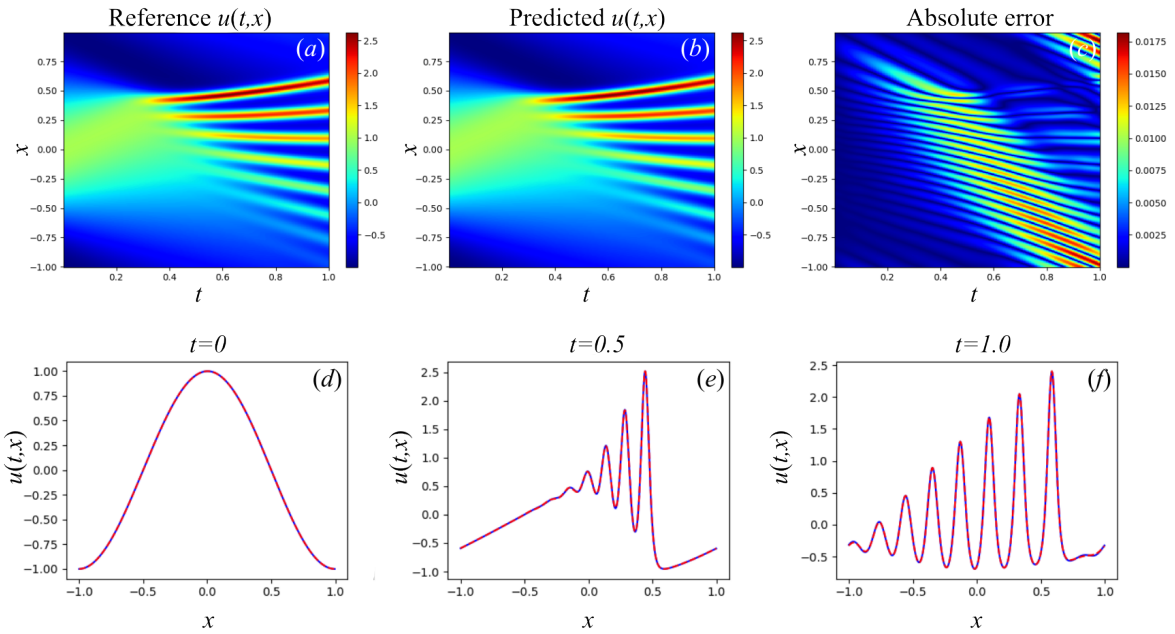


图 3：科特韦格-德弗里斯方程。(a) 为参考解，(b) 为基于算法 2 训练的物理信息神经网络的预测结果，(c) 为参考解与预测解之间的绝对差值。相对误差 $\epsilon_{\text{error}}$ 为 $6.84 \times 10^{-3}$。(d)、(e) 和 (f) 分别比较了对应于三个时间快照 $t = 0.0$、$t = 0.5$ 和 $t = 1.0$ 的预测解（红色虚线）与参考解（蓝色实线）。

此时，残差损失 (17) 为

$$
\begin{aligned}
\mathcal{L}_r(\boldsymbol{\theta}) = & \frac{1}{T}\frac{1}{V_\Omega}\int_0^T dt \int_{X_1}^{X_2} dx \left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,x)\right|^2 \\
& + \frac{1}{T}\frac{1}{V_\Omega}\left\{ 2\int_{X_1}^{X_2} dx \left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](0,x)\left[\mathbf{u}_{\boldsymbol{\theta}}(0,x) - \mathbf{g}(x)\right]\right| \right. \\
& \left. + \lim_{\substack{\sigma\to 0 \\ t\to 0}} \frac{\sigma}{t^2+\sigma^2} \int_{X_1}^{X_2} dx \left|\mathbf{u}_{\boldsymbol{\theta}}(0,x) - \mathbf{g}(x)\right|^2 \right\} \\
& + \frac{1}{T}\frac{1}{V_\Omega}\left\{ 4\eta \int_0^T dt \left|\mathcal{R}\left[\mathbf{u}_{\boldsymbol{\theta}}\right](t,X_1)\,\mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\left[\mathbf{u}_{\boldsymbol{\theta}}(t,X_1) - \mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\right]\right| \right. \\
& + \lim_{\substack{\sigma\to 0 \\ x\to X_1}} \frac{\sigma}{x^2+\sigma^2} 4\eta^2 \int_0^T dt \left|\mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\left[\mathbf{u}_{\boldsymbol{\theta}}(t,x) - \mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\right]\right|^2 \\
& \left. + 4\eta^2 \lim_{\substack{\sigma\to 0 \\ x\to X_1}} \frac{\sigma}{x^2+\sigma^2} \lim_{\substack{\tilde{\sigma}\to 0 \\ t\to 0}} \frac{\tilde{\sigma}}{t^2+\tilde{\sigma}^2} \left|\mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\left[\mathbf{u}_{\boldsymbol{\theta}}(t,x) - \mathbf{u}_{\boldsymbol{\theta}}(t,X_2)\right]\right| \left|\mathbf{u}_{\boldsymbol{\theta}}(t,x) - \mathbf{g}(x)\right| \right\}. \qquad (73)
\end{aligned}
$$

训练超参数见表 5 和表 6。表 2 展示了不同方法求解科特韦格-德弗里斯问题的精度。可以看出，对于多层感知器，使用狄拉克 δ函数和算法 2 得到了最佳结果，相对误差为 $\mathbb{L}_2$ $6.84 \times 10^{-3}$。该实验结果如图 3 所示。预测解与真实解几乎完全吻合。

### 4.2.2 具有异质激活函数的多层感知器

尽管借助前述全连接神经网络处理问题 (64)–(66) 取得了相当不错的结果，但上述架构并非解决这一问题的最佳选择。正如 [59]，所指出的，尽管通用

| 方法 | 相对 $\mathbb{L}_2$ 误差 |
|---|---|
| 因果训练（采用权重（38）的算法1） | $3.11 \times 10^{-2}$ |
| 基础因果训练（算法 1，权重来自 [19]） | $2.03 \times 10^{-2}$ |
| 狄拉克δ函数因果训练（算法 2，权重见（50）） | $6.84 \times 10^{-3}$ |
| 狄拉克 δ函数因果训练（用于 PAF多层感知机的算法 2，权重见（50）） | $2.45 \times 10^{-3}$ |

表 2：Korteweg–De Vries 方程：不同方法得到的相对 $\mathbb{L}_2$ 误差

逼近定理 [45] 保证了给定问题存在神经逼近器，但它并未提供任何机制来选择或构建最适合 PINN 训练的神经网络架构。

我们采用 [59], 中介绍的方法，该方法能够设计出在此问题上表现更佳的神经网络架构。这种方法称为物理激活函数 PINN（PAF PINN），其依据是所研究微分方程组的已知解或未知解的行为 $u$。

基本解　在无界介质中，微分方程（64）的解可以表示为孤子 [60,61]

$$u(x,t) = c\,\mathrm{sech}^2\left(ax - bt - d\right), \tag{74}$$

其中，$a$、$b$、$c$ 和 $d$ 为常数，它们取决于初始条件以及参数 $\eta$ 和 $\mu$；$\mathrm{sech}(\cdot) = \cosh^{-1}(\cdot)$ 为双曲正割。根据 PAF [59], 的概念，神经网络最后几层中的某一层的激活函数应为函数 $\mathrm{sech}^2$。

此外，我们还考虑到，由方程（64）–（66）描述的系统中可能存在多个孤子，其相互作用满足非线性叠加原理 [62]。孤子之间的相互作用由 Bäcklund 变换定义，其中包括 $tanh$ 函数，该函数适用于取值有限的孤子。因此，最后一层的激活函数必须采用双曲正切函数。

考虑到方程（64）已知解的上述特征，提出了以下神经网络架构：

1. 输入层由 2 个神经元组成（分别用于 $x$ 和 $t$ 数据）；2. 第二层由 120 个神经元组成，采用线性激活函数；3. 第三层由 128 个神经元组成，其中 10 个采用线性激活函数，118 个采用双曲正切函数作为激活函数；4. 第四层与第三层相同；5. 第五层由 128 个神经元组成，其中 10 个采用线性激活函数，118 个采用 $\mathrm{sech}^2$ 作为激活函数；6. 第六层与第三层相同；7. 第七层由一个线性神经元组成。

该神经网络的示意图如图 4(g) 所示。

该神经网络采用狄拉克 δ函数因果训练的超参数列于表 5 和表 6 中。训练共分为 3 次迭代，前两次迭代各进行 $3 \times 10^4$ 个训练轮次，最后一次进行 $9 \times 10^4$ 个训练轮次。第一次、第二次和第三次迭代训练后的相对误差 $\epsilon_{\text{error}}$ 分别为 $8.45 \times 10^{-3}$、$4.03 \times 10^{-3}$ 和 $2.45 \times 10^{-3}$。该实验结果如图 4 所示。可以看出，预测解与真实解高度吻合。

根据实验，我们得出结论：与采用同质激活函数的神经网络相比，采用异质激活函数的神经网络不仅准确度更高、收敛速度更快，而且还能在未进行训练的时空域中更好地外推解。图 5 展示了上一小节所述的 MLP（b）和 PAF多层感知机（e）在时间区间 $t \in [0,2]$ 内的解。可以看出，与 MLP 在同一区域中的表现相比，PAF多层感知机在区域 $t \in [1,2]$ 内展现出显著更好的定性解行为。相对误差定量地体现了这一结果：PAF多层感知机的相对误差为 $\epsilon_{\text{error}} = 3.28 \times 10^{-1}$，而 MLP 的相对误差为 $\epsilon_{\text{error}} = 6.61 \times 10^{-1}$。

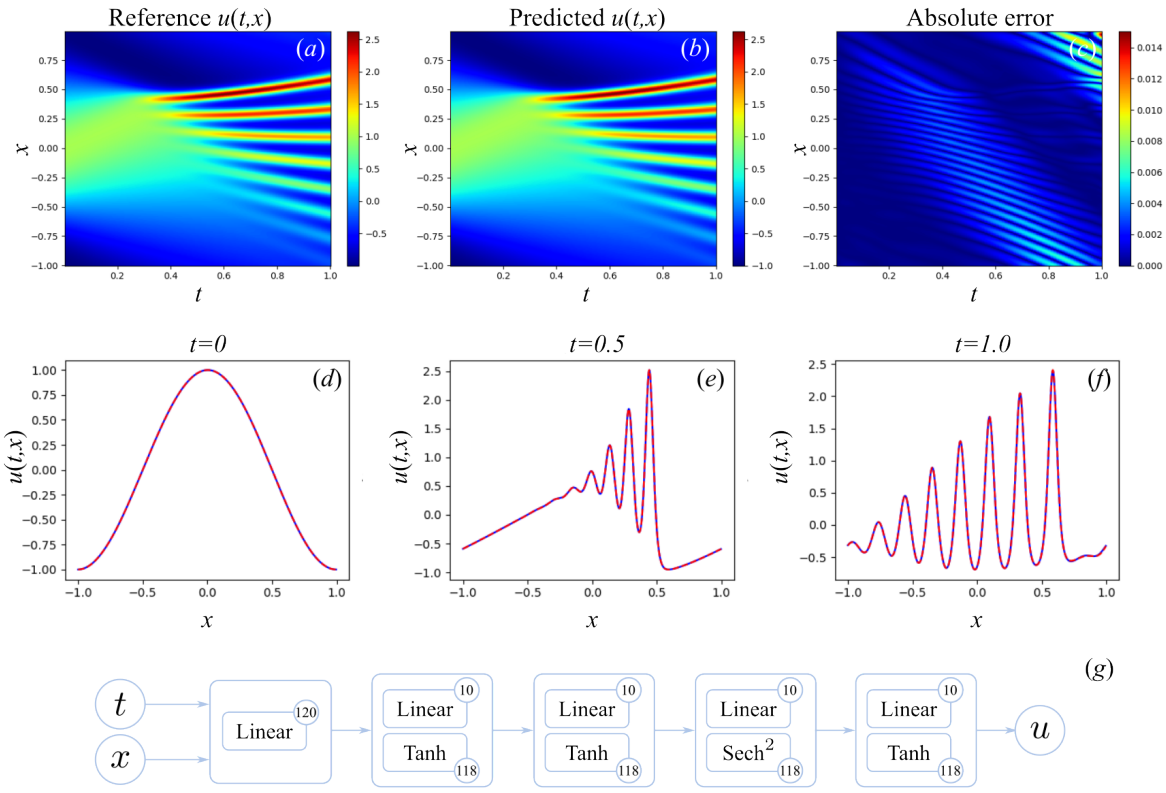


图 4：科特韦格-德弗里斯方程。(a) 为参考解，(b) 为基于算法 2 训练的物理信息神经网络的预测结果，(c) 为参考解与预测解之间的绝对差值。相对误差 $\epsilon_{\text{error}}$ 为 $2.45 \times 10^{-3}$。(d)、(e) 和 (f) 分别对比了对应于 $t = 0.0$、$t = 0.5$ 和 $t = 1.0$ 三个时间快照的预测结果（红色虚线）与参考解（蓝色实线）。(g) 为神经网络架构示意图。

### 4.3 Petrov–Kudrin 方程

考虑 Petrov–Kudrin 方程组 [63], , 它描述了强电磁波在非线性、无色散介质中的传播,

$$\begin{aligned} &E_\rho - \varepsilon_1^{-1/2} H_\tau = 0, \\ &H_\rho + \rho^{-1} H - \varepsilon_1^{1/2} \exp(\alpha E)\, E_\tau = 0, \end{aligned} \tag{75}$$

其中 $\rho \in [0,5]$、$\tau \in [0,4.75]$、$\alpha = 1$、$\varepsilon_1 = 2$。我们将考虑具有以下初始值的初值问题

$$E(0,\rho) = \left[1 + \rho^2 \exp(\alpha E)\right]^{-3/2}, \quad H(0,\rho) = 0. \tag{76}$$

该问题的精确解 [63]形式如下

$$\begin{aligned} E(\tau,\rho) &= \operatorname{Re}\left\{\left[(1-i\theta)^2 + \rho^2 \exp(\alpha E)\right]^{-3/2} (1-i\theta)\right\}, \\ H(\tau,\rho) &= \varepsilon_1^{1/2} \rho \exp(\alpha E) \operatorname{Re}\left\{i\left[(1-i\theta)^2 + \rho^2 \exp(\alpha E)\right]^{-3/2}\right\}. \end{aligned} \tag{77}$$

此处 $\theta = \tau + \alpha\rho H/(2\varepsilon_1^{1/2})$。

我们借助一个全连接神经网络 $u_{\boldsymbol{\theta}}$ 来表示潜变量 $E$ 和 $H$，该网络采用 $tanh$ 激活函数，包含 6 个隐藏层，每个隐藏层有 64 个神经元。为简化起见，在计算域 $[0,4.75]\times[0,5]$中构造了尺寸为 $100 \times 256$ 的均匀网格，生成了 $N_{ic} = 512$ 个初始点和 $N_r = 25600$ 个配置点

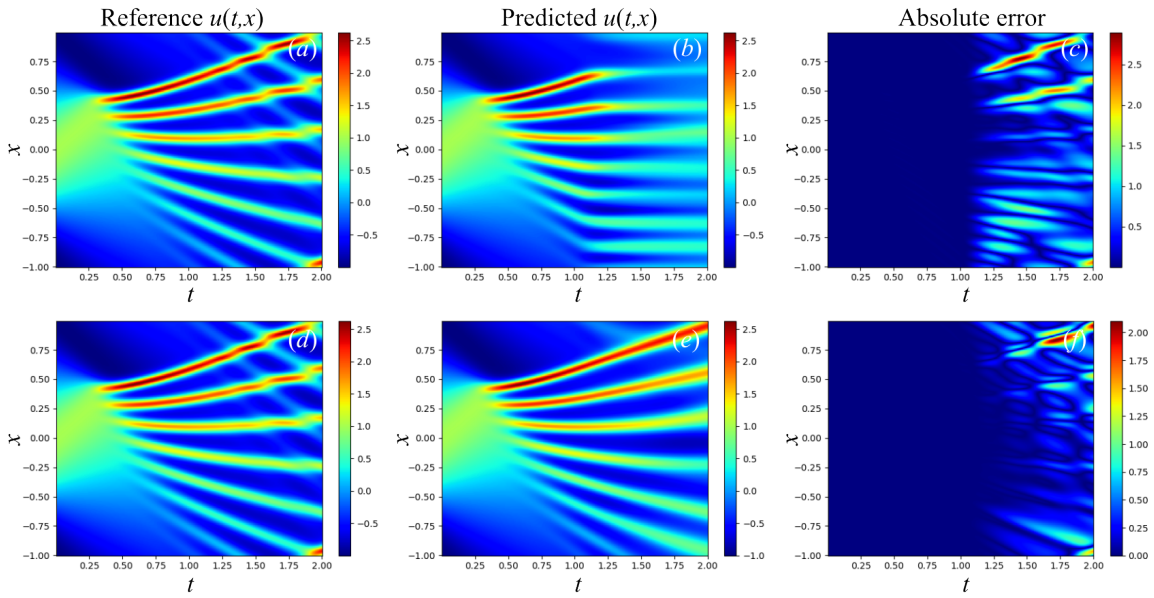


图5：科特韦格-德弗里斯方程。(a) 为参考解，(b) 为基于算法1训练的物理信息神经网络的预测（多层感知器的最佳结果，$\epsilon_{\text{error}} = 6.84 \times 10^{-3}$ 用于 $t \in [0,1]$ 且 $\epsilon_{\text{error}} = 6.61 \times 10^{-1}$ 用于$t \in [0,2]$），(c) 为参考解与预测解之差的绝对值。(d)–(f) 对PAF多层感知机相同（$\epsilon_{\text{error}} = 2.45 \times 10^{-3}$ 用于 $t \in [0,1]$ 且 $\epsilon_{\text{error}} = 3.28 \times 10^{-1}$ 用于 $t \in [0,2]$）。

| 方法 | 相对 $\mathbb{L}_2$ 误差 |
|---|---|
| 普通因果训练（算法1，权重来自 [19]） | $8.01 \times 10^{-3}$ |
| 因果训练（算法1，权重为（38））| $7.82 \times 10^{-3}$ |
| 狄拉克δ函数因果训练（算法1，权重为（50））| $7.69 \times 10^{-3}$ |

表3：Petrov–Kudrin方程：不同方法得到的相对 $\mathbb{L}_2$ 误差

，用于在算法 1 中通过权重 (38)、(50) 或来自 [19]的权重来约束 PDE 残差。所有情形下，训练轮数均设为 300000。与 Allen–Cahn 问题的情况相同，本问题使用 (16) 的对数形式和优化问题 63 进行训练 $u_{\boldsymbol{\theta}}$。

要使用算法1，必须将δ函数的极限表示转变为离散形式。所需的变换将在下一小节中介绍。

### 4.3.1 δ函数的近似

根据方程（22），用函数 $1/\sigma$ 近似式（23）中的 δ函数。假设 $\sigma$ 值为初始场分布增大或减小一倍所需的时间。对于问题 ˜（75）、（76），我们需要求出场 $E$ 在 $\rho = 0$ 处对 $\tau$ 的依赖关系。该值以下记为 $E$。此外，我们假设场 $E$ 对坐标 $\rho$ 的依赖方式与（76）中的 $E$ 类似。考虑到后一种情况，˜我们从方程（75）得到关于 $E$ 的以下方程

$$\frac{\partial}{\partial \tau}\left(e^{\alpha \tilde{E}} \frac{\partial \tilde{E}}{\partial \tau}\right) = -6 e^{\alpha \tilde{E}}. \tag{78}$$

方程（78）的解是

$$\tilde{E}(\tau) = \frac{1}{\alpha} \log\left[e^{\alpha} \cos\left(\tau \sqrt{6\alpha}\right)\right]. \tag{79}$$

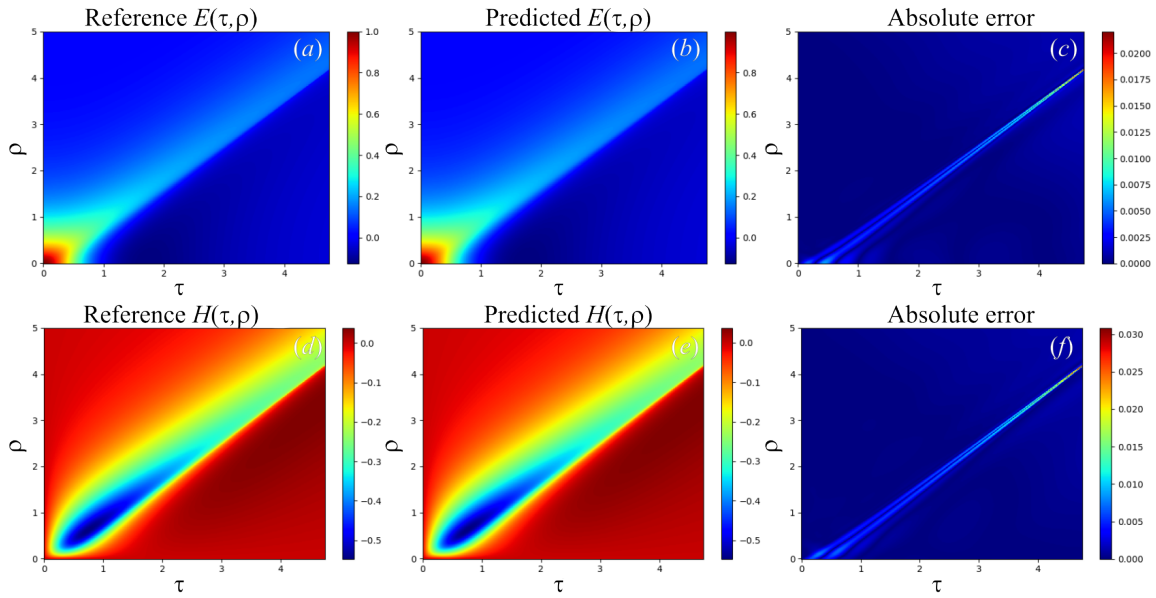


图6：Petrov–Kudrin方程：第一行是 $E$，第二行是 $H$。(a)和(d)为参考解，(b)和(e)为基于算法1训练的物理信息神经网络的预测结果，(c)和(f)为参考解与预测解之差的绝对值。总相对误差 $\epsilon_{\text{error}}$ 为 $7.69 \times 10^{-3}$。

这里我们使用初始 ˜ ˜1值 $E = 1$。由方程（79）可知，场 $E$ 的衰减时间是两倍

$$\tilde{\tau} = \arccos\left(e^{-\alpha/2}\right)/\sqrt{6\alpha}. \tag{80}$$

由此方程可知

$$\delta(\tau) \simeq 1/(2\tilde{\tau}). \tag{81}$$

表5和表6分别列出了算法1在权重 [19], 、权重(38)和权重(50)下的训练超参数。表3展示了使用这些方法求解Petrov–Kudrin问题的精度。对于多层感知器，使用Dirac δ函数和算法1取得了最佳结果，相对误差为 $\mathbb{L}_2$ $7.69 \times 10^{-3}$。该实验结果见图6和图7。预测解与真实解高度吻合。

## 5 结论

本文在物理信息神经网络（PINN）框架下介绍了几种神经网络训练方法。这些方法简化了损失函数的构建与分析，并改善了训练收敛性。我们还提出了更适用于所研究问题的神经网络架构。

我们将原先由微分方程以及初始条件和边界条件描述的问题，重新表述为仅由微分方程描述的问题。这种处理的主要优点是可以将损失函数简化为仅包含与微分方程相关的单项，从而无需调整与边界条件和初始条件相关各项的缩放系数。基于这一方法，我们推导出了 MAE 和均方误差损失的闭式表达式。我们还提出了狄拉克 δ函数的近似形式，该函数是均方误差损失表达式中的一部分。这些近似形式由定义域的离散化方法，或预期解在初始时刻及定义域边界处的行为决定。

受 [19] 启发，我们在广义函数形式体系（如赫维赛德阶跃函数和狄拉克δ函数）框架内，依据因果性和时空局部性原则提出了损失函数。研究还表明，

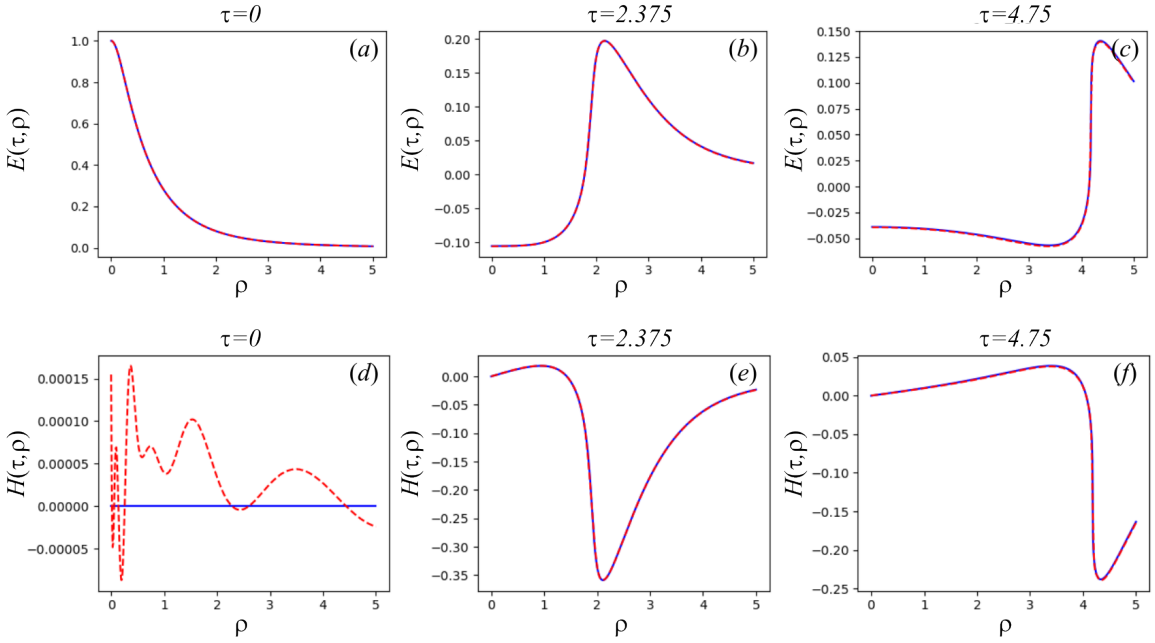


图 7：Petrov–Kudrin 方程：(a)、(b) 和 (c) 分别比较了 $\tau = 0.0$、 $\tau = 2.375$ 和$\tau = 4.75$三个时间快照下电场 $E$ 的预测解（红色虚线）与参考解（蓝色实线）。(d)、(e) 和 (f) 展示了磁场 $H$的相同关系

因果性训练可以扩展到空间域，并转化为时空训练。

最后，本文基于 [59]提出了针对科特韦格-德弗里斯问题的多层感知器模型改进方案。该改进基于科特韦格-德弗里斯方程的基本解和孤子的非线性叠加原理。研究表明，这种采用异质激活函数的神经网络训练速度快得多，且与采用同质激活函数的多层感知器相比，所得结果的精度显著更高。此外，这种神经网络对新领域的外推能力也强得多。
数值实验已针对若干问题开展，并给出了所提方法的精度。基于狄拉克δ函数权重的因果训练在所有开展的数值实验中均取得了最佳结果。

## 术语表

主要记号、缩写和符号的汇总见表4。

| 符号说明 | 描述 |
|---|---|
| MLP | 多层感知机 |
| PDE | 偏微分方程 |
| PINN | 物理信息神经网络 |
| MAE | 平均绝对误差 |
| MSE | 均方误差 |
| DASA-PINN | 用于 PINN 的可微对抗式自适应逐点加权方案 |
| PAF | 物理激活函数 |
| $\mathbf{u}$ | 偏微分方程的解 |
| $\mathbf{U}$ | 偏微分方程的人工扩展解 |
| $\mathcal{N}$ | 非线性微分算子 |
| $\mathcal{B}$ | 边界算子 |
| $\mathcal{R}$ | 偏微分方程残差 |
| $\mathbf{u}_{\boldsymbol{\theta}}$ | 偏微分方程解的神经网络表示 |
| $\boldsymbol{\theta}$ | 神经网络可训练参数的向量 |
| $N_t$ | 时间轴上的网格间隔数 |
| $N_x$ | 沿 $x$ 轴的网格间隔数 |
| $w_i$ | 时间 $t_i$处的残差权重 |
| $w_n^{(\rightarrow)}$ | 在下边界处开始计算权重时，坐标 $x_n$ 处的残差权重 |
| $w_n^{(\leftarrow)}$ | 上边界处权重初始计算情况下坐标 $x_n$ 处的残差权重 |
| $\varepsilon$ | 因果参数 |
| $\delta_w$ | 因果参数递增阈值 |
| $\mathcal{L}(t,\ \boldsymbol{\theta})$ | 时间残差损失 |
| $\mathcal{L}^*(t,\ \boldsymbol{\theta})$ | 时空域中的时间残差损失 |
| $\mathcal{L}(\boldsymbol{\theta})$ | 总训练损失 |

表 4：术语表

## B 方法的超参数

表 5 和表 6 分别汇总了所有数值实验的训练超参数和网络超参数。所有实验均使用 Adam 优化器，参数 $\theta$ 采用 Xavier 方案初始化 [64]。

| 偏微分方程问题 | 权重 | $N$ | $\eta_s$ | $\eta_{\min}$ | $\varepsilon_{\text{int}}$ | $\delta_w$ | 调度器 |
|---|---|---|---|---|---|---|---|
| 艾伦–卡恩 | (38)<br>(51)<br>(50) | $3\times10^5$<br>$6\times10^5$ | $3\times10^{-3}$<br>$10^{-3}$ | $10^{-12}$<br>$10^{-5}$<br>— | $10^{-3}$<br>$10^4$ | 0.99<br>0.95<br>0.99 | ExponentialLR<br>$\text{StepLR}^{0.95}_{5000}$ |
| 科特韦格-德弗里斯 | [19]<br>(38)<br>(50) | $3\times10^5$ | $3\times10^{-3}$ | — | $10^{-2}$ | 0.99 | $\text{StepLR}^{0.9}_{5000}$ |
| 科特韦格-德弗里斯 (PAF) | (50) | $3\times10^4$<br>$9\times10^4$ | $3\times10^{-3}$<br>$10^{-4}$<br>$3\times10^{-5}$ | $10^{-8}$ | $10^{-5}$<br>0.16<br>0.33 | 0.99 | 余弦退火学习率 |
| Petrov–Kudrin | [19]<br>(38)<br>(50) | $3\times10^5$ | $10^{-3}$ | — | $10^{-2}$ | 0.99 | $\text{StepLR}^{0.9}_{5000}$ |

表5：训练超参数

这里 $N$ 是轮次数， $\eta_s$ 是学习率的初始值， $\eta_{\min}$ 是学习率的最小值， $\varepsilon_{\text{int}}$ 是因果参数的初始值。对于ExponentialLR调度器，衰减率按以下公式计算

$$\gamma = (\eta_{\min}/\eta_s)^{1/N}\,.$$

$\text{StepLR}^{\gamma}_{E}$ 表示每隔 $E$ 个训练轮次，衰减率为 $\gamma$。

| PDE问题 | 架构 | 隐藏层数 | 神经元每层 | 激活函数 | $N_t$ | $N_x$ |
|---|---|---|---|---|---|---|
| 艾伦-卡恩 | MLP | 4 | 128 | tanh | 100 | 256 |
| 科特韦格-德弗里斯 | MLP | 3 | 128 | tanh | 100 | [19] 和（38）为 512<br>（50）为 256 |
| 科特韦格-德弗里斯 | PAF多层感知机 | 5 | 120 或 128 | 线性、双曲正切或双曲正割[2] | 100 | 512 对于 (50) |
| Petrov–Kudrin | MLP | 6 | 64 | tanh | 100 | 256 |

表 6：模型和网格参数

此处， $N_t$ 和 $N_x$ 分别表示时间轴和 $x$ 坐标轴方向上的数值数量。

## 参考文献

[1]L. Alzubaidi、J. Zhang、A. J. Humaidi、A. A. Dujaili、Y. Duan、O. A. Shamma、J. Santamaría、M. A. Fadhel、M. A. Amidie 和 L. Farhan，《深度学习综述：概念、CNN 架构、挑战、应用及未来方向》。Springer International Publishing，2021 年。 [在线]。获取地址：https://doi.org/10.1186/s40537-021-00444-8[2] Y. LeCun、Y. Bengio 和 G. Hinton，《深度学习》，Nature，第 521 卷，第 7553 期，第 436–444 页，2015 年 5 月。[在线]。获取地址：http://www.nature.com/articles/nature14539[3]P. Linardatos、V. Papastefanopoulos 和 S. Kotsiantis，《可解释人工智能：机器学习可解释性方法综述》，Entropy，第 23 卷，第 1 期，2021 年。 [在线]。获取地址：https: //www.mdpi.com/1099-4300/23/1/18[4] S. Khan、M. Naseer、M. Hayat、S. W. Zamir、F. S. Khan 和 M. Shah，《视觉领域中的 Transformer：综述》，ACM Comput. Surv.，第 54 卷，第 10s 期，2022 年 9 月。[在线]。获取地址：https://doi.org/10.1145/3505244[5] A. Vaswani、S. Bengio、E. Brevdo、F. Chollet、A. N. Gomez、S. Gouws、L. Jones、L. Kaiser、N. Kalchbrenner、N. Parmar、R. Sepassi、N. Shazeer 和 J. Uszkoreit，《用于神经机器翻译的 Tensor2Tensor》，2018 年。[在线]。获取地址：https://arxiv.org/abs/1803.07416[6] Q. Wang、B. Li、T. Xiao、J. Zhu、C. Li、D. F. Wong 和 L. S. Chao，《学习用于机器翻译的深度 Transformer 模型》，2019 年。 [在线]。获取地址：https://arxiv.org/abs/1906.01787[7] S. Yao 和 X. Wan，《用于多模态机器翻译的多模态 Transformer》，载于《第 58 届计算语言学协会年会论文集》。在线：计算语言学协会，2020 年 7 月，第 4346–4350 页。 [在线]。获取地址：https://aclanthology.org/2020.acl-main.400[8] A. Ramesh、M. Pavlov、G. Goh、S. Gray、C. Voss、A. Radford、M. Chen 和 I. Sutskever，《零样本文本到图像生成》，2021 年。 [在线]。获取地址：https://arxiv.org/abs/2102.12092[9] S. Frolov、T. Hinz、F. Raue、J. Hees 和 A. Dengel，《对抗式文本到图像合成：综述》，Neural Networks，第 144 卷，第 187–209 页，2021 年。 [在线]。获取地址：https://www.sciencedirect.com/science/article/pii/ S0893608021002823[10] D. Silver、T. Hubert、J. Schrittwieser、I. Antonoglou、M. Lai、A. Guez、M. Lanctot、L. Sifre、D. Kumaran、T. Graepel、T. Lillicrap、K. Simonyan 和 D. Hassabis，《一种通过自我对弈掌握国际象棋、日本将棋和围棋的通用强化学习算法》，Science，第 362 卷，第 6419 期，第 1140–1144 页，2018 年。 [在线]。获取地址：https://www.science.org/doi/abs/10.1126/science.aar6404[11]J. Schrittwieser、I. Antonoglou、T. Hubert、K. Simonyan、L. Sifre、S. Schmitt、A. Guez、E. Lockhart、D. Hassabis、T. Graepel、T. Lillicrap 和 D. Silver，《通过基于学习模型的规划掌握 Atari、围棋、国际象棋和日本将棋》，Nature，第 588 卷，第 7839 期，第 604–609 页，2020 年 12 月。 [在线]。获取地址：http://www.nature.com/articles/s41586-020-03051-4[12] L. Ouyang、J. Wu、X. Jiang、D. Almeida、C. L. Wainwright、P. Mishkin、C. Zhang、S. Agarwal、K. Slama、A. Ray、J. Schulman、J. Hilton、F. Kelton、L. Miller、M. Simens、A. Askell、P. Welinder、P. Christiano、J. Leike 和 R. Lowe，《训练语言模型以遵循人类反馈指令》，2022 年。 [在线]。获取地址：https://arxiv.org/abs/2203.02155[13] L. Lu、P. Jin、G. Pang、Z. Zhang 和 G. E. Karniadakis，《基于算子通用逼近定理，通过 DeepONet 学习非线性算子》，Nature Machine Intelligence，第 3 卷，第 3 期，第 218–229 页，2021 年 3 月。 [在线]。获取地址：https://doi.org/10.1038%2Fs42256-021-00302-5[14] Z. Li、D. Z. Huang、B. Liu 和 A. Anandkumar，《用于一般几何形状上偏微分方程的带学习变形 Fourier 神经算子》，2022 年。 [在线]。获取地址：https://arxiv.org/abs/2207.05209[15]M. A. Krinitskiy、V. M. Stepanenko、A. O. Malkhanov 和 M. E. Smorkalov，《一种基于神经网络的通用偏微分方程识别方法：在新型 AI 加速器上的实现》，Supercomputing Frontiers and Innovations，第 9 卷，第 3 期，2022 年 9 月。 [在线]。获取地址：https://superfri.org/index.php/superfri/article/view/439[16]V. Fanaskov 和 I. Oseledets，《谱神经算子》，2022 年。 [在线]。获取地址：https: //arxiv.org/abs/2205.10573[17] O. Ovadia、A. Kahana、P. Stinis、E. Turkel 和 G. E. Karniadakis，《ViTO：视觉 Transformer 算子》，2023 年 3 月。 [在线]。获取地址：http://arxiv.org/abs/2303.08891[18]M. Raissi、P. Perdikaris 和 G. E. Karniadakis，《物理信息神经网络：一种用于求解涉及非线性偏微分方程的正问题和逆问题的深度学习框架

方程，” 计算物理学杂志，第 378 卷，第 686–707 页，2019 年。 [在线]。可访问：https://linkinghub.elsevier.com/retrieve/pii/S0021999118307125[19] S. Wang、S. Sankaran 和 P. Perdikaris，“尊重因果关系是训练物理信息神经网络所需的一切”，2022 年。 [在线]。可访问：http://arxiv.org/abs/2203.07404[20] H. Wang、X. Qian、Y. Sun 和 S. Song，“一种用于求解具有守恒律的偏微分方程的改进型物理信息神经网络”，—。 [在线]。可访问：https://ssrn.com/abstract=4274376

[21]M. Raissi、P. Perdikaris 和 G. E. Karniadakis，“物理信息深度学习（第一部分）：非线性偏微分方程的数据驱动求解方案”，第 I 部分，第 1–22 页。 [在线]。可访问：https://arxiv.org/abs/1711.10561

[22] ——, “物理信息深度学习（第二部分）：数据驱动的非线性偏微分方程发现”，第二部分，第 1–19 页。 [在线]。可访问：https://arxiv.org/abs/1711.10566

[23] J. Jumper、R. Evans、A. Pritzel、T. Green、M. Figurnov、O. Ronneberger、K. Tunyasuvunakool、R. Bates、A. Žídek、A. Potapenko、A. Bridgland、C. Meyer、S. A. A. Kohl、A. J. Ballard、A. Cowie、B. Romera-Paredes、S. Nikolov、R. Jain、J. Adler、T. Back、S. Petersen、D. Reiman、E. Clancy、M. Zielinski、M. Steinegger、M. Pacholska、T. Berghammer、S. Bodenstein、D. Silver、O. Vinyals、A. W. Senior、K. Kavukcuoglu、P. Kohli 和 D. Hassabis，“利用 AlphaFold 进行高精度蛋白质结构预测”，*Nature*，第 596 卷，第 7873 期，第 583–589 页，2021 年 8 月。[在线]。可访问：https://www.nature.com/articles/s41586-021-03819-2

[24] J. Lin、F. Haberstroh、S. Karsch 和 D. Andreas，“高功率激光系统与实验中的目标检测网络应用”，第 1–10 页，2022 年。 [在线]。可访问：http://arxiv.org/abs/2210.02539[25]M. J. V. Streeter、C. Colgan、C. C. Cobo、C. Arran、E. E. Los、R. Watt、N. Bourgeois、L. Calvin、J. Carderelli、N. Cavanagh 等，“利用变分神经网络对激光尾场加速器进行建模”，高功率激光科学与工程，第 11 卷，文章 e9，2023 年。 [在线]。可访问：https://www.cambridge.org/core/journals/high-power-laser-science-and-engineering/article/laser-wakefield-accelerator-modelling-with-variational-neural-networks/272206304D99ABB614B6F5A57B389B29[26] D. C. Psichogios 和 L. H. Ungar，“用于过程建模的混合神经网络—第一性原理方法”，*AicheJournal*，第 38 卷，第 1499–1511 页，1992 年。[27] I. Lagaris、A. Likas 和 D. Fotiadis，“用于求解常微分方程和偏微分方程的人工神经网络”，*IEEE* 神经网络汇刊，第 9 卷，第 5 期，第 987–1000 页，1998 年。 [在线]。可访问：https://doi.org/10.1109%2F72.712178[28] C. Rackauckas、Y. Ma、J. Martensen、C. Warner、K. Zubov、R. Supekar、D. Skinner、A. Ramadhan 和 A. Edelman，“面向科学机器学习的通用微分方程”，第 1–55 页，2020 年 1 月。 [在线]。可访问：http://arxiv.org/abs/2001.04385[29]L. Yuan、Y.-Q. Ni、X.-Y. Deng 和 S. Hao，“A-PINN：用于非线性积分微分方程正问题和逆问题的辅助物理信息神经网络”，计算物理学杂志，第 462 卷，文章 111260，2022 年 8 月。 [在线]。可访问：https://doi.org/10.1016/j.jcp.2022.111260https: //linkinghub.elsevier.com/retrieve/pii/S0021999122003229[30] X. Jin、S. Cai、H. Li 和 G. E. Karniadakis，“NSFnets（Navier–Stokes 流网络）：用于不可压缩 Navier–Stokes 方程的物理信息神经网络”，计算物理学杂志，第 426 卷，文章 109951，2021 年。 [在线]。可访问：https://www.sciencedirect.com/science/article/pii/S0021999120307257[31] L. Zhao、Z. Li、Z. Wang、B. Caswell、J. Ouyang 和 G. E. Karniadakis，“将主动学习和迁移学习应用于微观—宏观耦合，以模拟黏弹性流动”，计算物理学杂志，第 427 卷，文章 110069，2021 年。 [在线]。可访问：https://doi.org/10.1016/j.jcp.2020.110069[32] E. Kharazmi、Z. Zhang 和 G. E. Karniadakis，“hp-VPINNs：采用域分解的变分物理信息神经网络”，计算力学与工程中的计算方法，第 374 卷，文章 113547，2021 年。[在线]。可访问：https://doi.org/10.1016/j.cma.2020.113547[33] L. Yang、X. Meng 和 G. E. Karniadakis，“B-PINNs：用于含噪数据下 PDE 正问题和逆问题的贝叶斯物理信息神经网络”，计算物理学杂志，第 425 卷，文章 109913，2021 年。[在线]。可访问：https://doi.org/10.1016/j.jcp.2020.109913[34] S. Cuomo、V. S. di Cola、F. Giampaolo、G. Rozza、M. Raissi 和 F. Piccialli，“通过物理信息神经网络实现科学机器学习：现状与展望”，2022 年 1 月。 [在线]。可访问：http://arxiv.org/abs/2201.05624

[35] G. Pang、M. D'Elia、M. Parks 和 G. E. Karniadakis，“nPINNs：用于参数化非局部通用拉普拉斯算子的非局部物理信息神经网络：算法与应用”，《计算物理学杂志》，第 422 卷，文章 109760，2020 年。 [在线]。可访问：https://doi.org/10.1016/j.jcp.2020.109760[36] R. G. Patel、I. Manickam、N. A. Trask、M. A. Wood、M. Lee、I. Tomas 和 E. C. Cyr，“用于双曲系统的热力学一致物理信息神经网络”，《计算物理学杂志》，第 449 卷，文章 110754，2022 年 1 月。 [在线]。可访问：https://doi.org/10.1016/j.jcp.2021.110754https: // linkinghub.elsevier.com/retrieve/pii/S0021999121006495[37] S. Cai、Z. Mao、Z. Wang、M. Yin 和 G. E. Karniadakis，“用于流体力学的物理信息神经网络（PINNs）：综述”，《力学学报》，第 37 卷，第 12 期，第 1727–1738 页，2021 年 12 月。 [在线]。可访问：https://link.springer.com/10.1007/s10409-021-01148-1[38] S. Thakur、M. Raissi、H. Mitra 和 A. Ardekani，“物理信息神经网络的时间一致性损失”，第 1–14 页，2023 年 1 月。 [在线]。可访问：http://arxiv.org/abs/2301.13262[39] B. Moseley、A. Markham 和 T. Nissen-Meyer，“利用物理信息深度学习求解波动方程”，2020 年 6 月。 [在线]。可访问：http://arxiv.org/abs/2006.11894[40]G. Lin、P. Hu、F. Chen、X. Chen、J. Chen、J. Wang 和 Z. Shi，“BI Net：利用边界积分网络学习求解偏微分方程”，第 1–27 页，2021 年 10 月。 [在线]。可访问：http://arxiv.org/abs/2110.00352[41] Q. He、D. Barajas-Solano、G. Tartakovsky 和 A. M. Tartakovsky，“用于多物理场数据同化的物理信息神经网络及其在地下传输中的应用”，《水资源进展》，第 141 卷，文章 103610，2020 年 7 月。 [在线]。可访问：https://linkinghub.elsevier.com/retrieve/pii/S0309170819311649[42] T. Qin、A. Beatson、D. Oktay、N. McGreivy 和 R. P. Adams，“Meta-PDE：无需网格即可快速学习求解偏微分方程”，2022 年 11 月。 [在线]。可访问：http://arxiv.org/abs/2211.01604[43] C. L. Wight 和 J. Zhao，“使用自适应物理信息神经网络求解 Allen–Cahn 方程和 Cahn–Hilliard 方程”，2020 年 7 月。 [在线]。可访问：http://arxiv.org/abs/2007.04542[44] A. Daw、J. Bu、S. Wang、P. Perdikaris 和 A. Karpatne，“利用进化采样缓解 PINN 中的传播失败”，2022 年 7 月。 [在线]。可访问：http://arxiv.org/abs/2207.02338[45] K. Hornik、M. Stinchcombe 和 H. White，“多层前馈网络是通用逼近器”，《神经网络》，第 2 卷，第 5 期，第 359–366 页，1989 年。 [在线]。可访问：https://www.sciencedirect.com/science/ article/pii/0893608089900208[46] A. Griewank 和 A. Walther，《导数求值》，第 2 版。工业与应用数学学会，2008 年。 [在线]。可访问：https://epubs.siam.org/doi/abs/10.1137/1.9780898717761[47]S. Wang、Y. Teng 和 P. Perdikaris，“理解并缓解物理信息神经网络中的梯度流病态问题”，《SIAM 科学计算杂志》，第 43 卷，第 5 期，第 A3055—A3081 页，2021 年。 [在线]。可访问：https://doi.org/10.1137/20M1318043[48]S. Wang、X. Yu 和 P. Perdikaris，“ PINN 何时以及为何无法训练：神经切线核视角”，《计算物理学杂志》，第 449 卷，文章 110768，2022 年。 [在线]。可访问：https://www.sciencedirect.com/science/article/pii/S002199912100663X[49] B. Davies，《积分变换及其应用》，第 3 版。Springer New York, NY，2008 年。 [在线]。可访问：https://link.springer.com/book/10.1007/978-1-4684-9283-5#bibliographic-information[50] A. I. Saichev 和 W. A. Woyczynski，《物理与工程科学中的广义函数》，第 1 卷：广义函数与分形微积分、积分变换和小波，第 1 版。Birkhäuser Cham，1996 年。 [在线]。可访问：https://link.springer.com/book/10.1007/978-3-319-97958-8[51] G. Zhang、H. Yang、F. Zhu、Y. Chen 和 X. Zheng，“Dasa-Pinns：用于物理信息神经网络的可微对抗式自适应逐点加权方案”，《SSRN 电子期刊》，2023 年。 [在线]。可访问：https://www.ssrn.com/abstract=4376049[52] S. Dong 和 N. Ni，“一种表示周期函数并利用深度神经网络严格施加周期性边界条件的方法”，《计算物理学杂志》，第 435 卷，文章 110242，2021 年。 [在线]。可访问：https://www.sciencedirect.com/science/article/pii/S0021999121001376[53]D. P. Kingma 和 J. Ba，“Adam：一种随机优化方法”，2014 年。 [在线]。可访问：https://arxiv.org/abs/1412.6980[54] A. Paszke、S. Gross、F. Massa、A. Lerer、J. Bradbury、G. Chanan、T. Killeen、Z. Lin、N. Gimelshein、L. Antiga、A. Desmaison、A. Köpf、E. Yang、Z. DeVito、M. Raison、A. Tejani、S. Chilamkurthy、B. Steiner、L. Fang、J. Bai,

以及 S. Chintala，《PyTorch：一种命令式风格的高性能深度学习库》。美国纽约州雷德胡克：Curran Associates Inc., 2019.[55] T. A. Driscoll、N. Hale 和 L. N. Trefethen，《Chebfun 指南》。牛津：Pafnuty Publications，2014.[56] L. McClenny 和 U. Braga-Neto，“使用软注意力机制的自适应物理信息神经网络”，2020 年 9 月。 [在线]。可访问：https://arxiv.org/abs/2009.04544[57] R. Mattey 和 S. Ghosh，“一种用于训练 Allen–Cahn 和 Cahn–Hilliard 方程物理信息神经网络的新型序贯方法”，《应用力学与工程中的计算方法》，第 390 卷，页码 114474，2022 年 2 月。 [在线]。可访问：https://linkinghub.elsevier.com/retrieve/pii/S0045782521006939[58]N. J. Zabusky 和 M. D. Kruskal，“无碰撞等离子体中‘孤子’的相互作用及初始状态的复现”，《物理评论快报》，第 15 卷，页码 240–243，1965 年 8 月。 [在线]。可访问：https://link.aps.org/doi/10.1103/PhysRevLett.15.240[59] J. Abbasi 和 P. Ø. Andersen，“物理激活函数（PAFs）：一种将物理学更高效地引入物理信息神经网络（PINNs）的方法。” [在线]。可访问：https://arxiv.org/abs/2205.14630[60] D. J. Korteweg 和 G. de Vries，“XLI. 关于长波在矩形运河中传播时形态的变化，以及一种新型长驻波”，《伦敦、爱丁堡和都柏林哲学杂志与科学杂志》，第 39 卷，第 240 期，页码 422–443，1895 年 5 月。 [在线]。可访问：https://www.tandfonline.com/doi/full/10.1080/14786449508620739[61] A. F. Vakakis 编，《非线性系统中的正规模态与局域化》。多德雷赫特：施普林格荷兰，2001. [在线]。可访问：http://link.springer.com/10.1007/978-94-017-2452-4[62] K. Brauer，“Korteweg-de Vries 方程：历史、精确解与图形表示”，2000 年。[在线]。可访问：https://www.math.arizona.edu/~gabitov/teaching/141/math_485/KDV.pdf[63]E. Y. Petrov 和 A. V. Kudrin，“非线性色散介质中麦克斯韦方程的精确轴对称解”，《物理评论快报》，第 104 卷，页码 190404，2010 年 5 月。 [在线]。可访问：https://link.aps.org/doi/10.1103/PhysRevLett.104.190404[64] X. Glorot 和 Y. Bengio，“理解深度前馈神经网络训练的难点”，载于《第十三届国际人工智能与统计学会议论文集》，机器学习研究论文集系列，Y. W. Teh 和 M. Titterington 编，第 9 卷。意大利撒丁岛基亚拉古纳度假村：PMLR，2010 年 5 月 13–15 日，页码 249–256。 [在线]。可访问：https://proceedings.mlr.press/v9/glorot10a.html