

# 课程-迁移学习物理信息神经网络用于 曲面长时间对流扩散行为模拟\*

闵建, 傅卓佳, 郭远

(河海大学 力学与工程科学学院 工程与科学数值模拟软件中心, 南京 211100)

(我刊青年编委傅卓佳来稿)

**摘要:** 物理信息神经网络 (physics-informed neural networks, PINN) 将物理先验知识编码到神经网络中, 减少了神经网络对于数据量的需求。但是对于时间相关偏微分方程的长时间问题, 传统 PINN 稳定性差, 甚至难以求得有效解。针对此问题, 该文发展了一种基于课程学习和迁移学习的物理信息神经网络 (curriculum-transfer-learning-based physics-informed neural networks, CTL-PINN)。该方法的主要思想是: 将长时间历程模拟问题转化为该时间域内多个短时间历程模拟问题, 引入课程学习的思想, 由简到难, 通过 PINN 在小时间段区域内训练, 而后逐渐增大所求解的时域范围; 进而引入迁移学习方法, 在课程学习的基础上进行时域上的迁移, 逐步采用 PINN 进行求解, 从而实现曲面上对流扩散行为的长时间模拟。该文将此 CTL-PINN 与非本征的曲面算子处理技术相结合, 用于复杂曲面上长时间对流扩散行为的模拟, 并通过多个数值算例验证了 CTL-PINN 的有效性和鲁棒性。

**关键词:** 物理信息神经网络; 课程学习; 迁移学习; 对流扩散; 曲面; 长时间历程  
**中图分类号:** TP183; O34 **文献标志码:** A **DOI:** 10.21656/1000-0887.440320

## Curriculum-Transfer-Learning-Based Physics-Informed Neural Networks for Simulating Long-Term-Evolution Convection-Diffusion Behaviors on Curved Surfaces

MIN Jian, FU Zhoujia, GUO Yuan

(Center for Numerical Simulation Software in Engineering and Sciences,  
College of Mechanics and Engineering Science, Hohai University, Nanjing 211100, P.R.China)

(Contributed by FU Zhoujia, M.AMM Youth Editorial Board)

**Abstract:** Physics-informed neural networks (PINNs) encode prior physical knowledge into neural networks, alleviating the need for extensive data volume within the network. However, for long-term problems involving time-dependent partial differential equations, the traditional PINN exhibits poor stability and struggles to obtain effective solutions. To address this challenge, a novel physics-informed neural network based on curriculum learning and transfer learning (CTL-PINN) was introduced. The main idea of this method is to transform the problem of long-term course simulation into multiple short-term course simulation problems within this time do-

\* 收稿日期: 2023-10-25; 修订日期: 2024-02-22

基金项目: 国家自然科学基金 (12122205; 12372196)

作者简介: 闵建 (2001—), 男, 硕士生 (E-mail: 231308010033@hhu.edu.cn);

傅卓佳 (1985—), 男, 教授, 博士 (通讯作者, E-mail: paul212063@hhu.edu.cn).

引用格式: 闵建, 傅卓佳, 郭远. 课程-迁移学习物理信息神经网络用于曲面长时间对流扩散行为模拟[J]. 应用数学和力学, 2024, 45(9): 1212-1223.

main. Under the concept of curriculum learning, and step by step from simpleness to difficulty, the scope of the time domain to be solved was gradually expanded by training the PINN within small time quanta. Furthermore, the transfer learning method was adopted to transfer across the time domain based on the curriculum learning, and the PINN was gradually employed for solution, thus to achieve long-term simulation of convection-diffusion behaviors on curved surfaces. The CTL-PINN was combined with the extrinsic surface operator processing technology to simulate long-term convection-diffusion behaviors on complex surfaces, and the effectiveness and robustness of the improved physics-informed neural network were verified through multiple numerical examples.

**Key words:** physics-informed neural network; curriculum learning; transfer learning; convection-diffusion; surface; long-term evolution

## 0 引 言

近年来, 曲面上的扩散行为得到了越来越多国内外学者的关注, 其在水污染<sup>[1]</sup>、自然和工业上的颗粒沉积<sup>[2]</sup>以及生物表面的图案形成<sup>[3]</sup>等领域都有着广泛的应用前景. 描述这类曲面上物理力学行为的偏微分方程与平坦空间下不同, 此类偏微分方程定义在切线空间下, 并与曲面的曲率相关<sup>[4]</sup>.

数值求解方法, 如有限差分法 (finite difference method, FDM)<sup>[5]</sup>、有限单元法 (finite element method, FEM)<sup>[6-7]</sup>、有限体积法 (finite volume method, FVM)<sup>[8]</sup> 和无网格广义有限差分法 (generalized finite difference method, GFDM)<sup>[9]</sup> 等方法可用于求解曲面上定义的方程, 但是这些方法要求在曲面上构造高质量的网格或配点, 并且处理非线性问题时附加的非线性项也会影响其计算结果的精度和稳定性. 随着机器学习技术的发展, 一类基于物理信息的神经网络技术 (PINN)<sup>[10-11]</sup> 被广泛应用于求解各种不同领域的问题<sup>[12-15]</sup>. 该方法将反映物理信息的偏微分方程结合初始条件和边界条件作为额外的物理信息引入神经网络, 减少了神经网络对于数据量的需求. 相较于传统的网格法和配点法, PINN 通过神经网络的非线性映射能力, 可以灵活地逼近复杂的非线性函数关系, 在求解非线性偏微分方程和边值问题时表现出色; 结合物理方程和观测数据双驱动, 可得到符合物理方程和观测数据的解; 一定程度上避免了维度诅咒<sup>[16]</sup>. 其在求解曲面上偏微分方程已有了一些研究: Fang 等<sup>[17]</sup> 采用 PINN 求解了曲面上的时间无关问题; Tang 等<sup>[18-19]</sup> 通过 PINN 结合不同的曲面微分算子处理方法, 求解了曲面上的稳态扩散方程, 并验证了其在不同曲面和非线性问题上的适用性.

然而, PINN 也面临着一些挑战, 其中一项便是 PINN 难以进行长时间历程的模拟, 对于某一时间相关问题, 往往缺乏中间过程的数据, 并且大的时间域导致 PINN 更易陷入局部最优解. 对于这一问题, 已有了一定的研究: Krishnapriyan 等<sup>[20]</sup> 将求解的时间域切片, 将一个长时间问题转化为多个短时间问题, 并通过上一时域最终时刻的预测值作为下一时域的初始条件以逐步推进求解时域. Penwarden 等<sup>[21]</sup> 通过时域堆叠分解方法, 减小了单个 PINN 求解的时域, 并采用迁移学习使得神经网络模型在初始化时就满足堆叠域内数据, 提高了训练效率. Meng 等<sup>[22]</sup> 使用了一个串行的, 全求解域上的粗求解器和若干个并行的子域上的精细 PINN, 粗求解器的解用作每个子域中精细 PINN 的初始条件, 并通过精细 PINN 独立求解对应子域中的方程. Jagtap 等<sup>[23]</sup> 将求解域分解, 并在共享边界上施加适当的连续性条件, 使得各子域上的 PINN 在边界上满足连续性, 这种区域分解方法类似于有限元法. Guo 等<sup>[24]</sup> 提出了一种多重预训练方法逐步扩大单个 PINN 的求解域, 其本质上可视为一种课程学习策略.

本文采用 Penwarden 等<sup>[21]</sup> 的时域堆叠分解思想, 结合迁移学习<sup>[25]</sup> 策略, 在 Guo 等<sup>[24]</sup> 方法的基础上进行了改进和发展, 提出了一种将 PINN 结合课程学习<sup>[26]</sup> 和迁移学习的课程-迁移学习物理信息神经网络 (CTL-PINN), 并将其用于模拟曲面长时间对流扩散行为.

## 1 方 法

本节中将介绍非本征方式表示的曲面微分算子、CTL-PINN 的原理及流程.

## 1.1 非本征技术表示的曲面微分算子

从平坦空间到卷曲空间,最根本的差异就在于微分算子<sup>[4]</sup>.此处非本征技术是一种曲面微分算子的处理技术,即根据切线空间算子与标准空间算子的映射关系,将切线空间的偏微分方程转化为 Euclid 空间的偏微分方程.定义在  $d$  维空间下的、光滑且封闭的曲面  $S \in \mathbb{R}^d$  的维度等于  $d - 1$ .本文讨论的是  $d = 3$  的三维空间,在三维空间中定义的瞬态对流扩散方程可以写成以下形式:

$$\frac{\partial u}{\partial t} = (\nabla_S \cdot (D \nabla_S) - \mathbf{v} \cdot \nabla_S + c)u(x, y, z, t) + f, \quad (1)$$

$$u(x, y, z, 0) = I(x, y, z), \quad (2)$$

其中  $D$  为扩散系数;  $\mathbf{v}$  为速度场;  $c$  为反应系数;  $f$  为源项;  $u(x, y, z, t)$  为待求物理量;  $\nabla_S$  为曲面梯度算子,当  $D$  为常数  $a$  时,  $\nabla_S \cdot (D \nabla_S) = a \Delta_S$ , 其中  $\Delta_S$  为曲面 Laplace 算子;  $I$  为初始条件,由于本文讨论的是封闭曲面,所以不存在边界条件.

已知曲面光滑,为了表示曲面微分算子与标准 Euclid 微分算子之间的关系,我们将任意  $(x, y, z) \in S$  处的单位外法向量表示为  $\mathbf{n} = (n_x, n_y, n_z)$  并将相应的投影矩阵在切线空间表示为

$$\mathbf{P} = (\mathbf{I}_3 - \mathbf{n}\mathbf{n}^T) \in \mathbb{R}^{3 \times 3}, \quad (3)$$

其中  $\mathbf{I}_3$  是  $3 \times 3$  的单位矩阵,曲面梯度算子  $\nabla_S$  可以根据标准 Euclid 梯度算子  $\nabla$  定义为

$$\nabla_S := \mathbf{P} \nabla, \quad (4)$$

类似地,曲面 Laplace 算子  $\Delta_S$  可以定义为

$$\Delta_S := \nabla_S \cdot \nabla_S, \quad (5)$$

其中曲面 Laplace 算子可视为散度梯度算子,通过引入非本征技术并将式(4)代入式(5).曲面梯度算子和曲面 Laplace 算子作用于任何足够光滑的函数可以表示为

$$\nabla_S u = \nabla u - \mathbf{n} \partial_n u, \quad (6)$$

$$\Delta_S u = \Delta u - H_S \partial_n u - \partial_n^{(2)} u, \quad (7)$$

其中  $\partial_n u = \mathbf{n}^T \nabla u$ ;  $\partial_n^{(2)} u = \mathbf{n}^T \mathbf{J}(\nabla u) \mathbf{n}$  并且  $H_S = \text{tr}(\mathbf{J}(\mathbf{n})(\mathbf{I} - \mathbf{n}\mathbf{n}^T))$ ,  $\mathbf{J}$  表示 Euclid 空间中的 Jacobi 算子. Euclid 空间是我们最熟悉的,并且大多数算法也是在 Euclid 空间中发展起来的,一旦获得非本征的 Euclid 空间下的形式,曲面算子就自然得到了.例如,对于曲面  $S = x^2 + y^2 + z^2 - 1$  表示的球面,可以得到单位法向量  $[x, y, z]^T$ . 将其代入式(6)和(7),曲面微分算子可以表示为

$$\nabla_S = \begin{bmatrix} 1 - x^2 & -xy & -xz \\ -xy & 1 - y^2 & -yz \\ -xz & -yz & 1 - z^2 \end{bmatrix} \begin{bmatrix} \partial_x \\ \partial_y \\ \partial_z \end{bmatrix} = \begin{bmatrix} (1 - x^2) \partial_x - xy \partial_y - xz \partial_z \\ -xy \partial_x + (1 - y^2) \partial_y - yz \partial_z \\ -xz \partial_x - yz \partial_y + (1 - z^2) \partial_z \end{bmatrix}, \quad (8)$$

$$\Delta_S = (1 - x^2) \partial_{xx} + (1 - y^2) \partial_{yy} + (1 - z^2) \partial_{zz} - 2xy \partial_{xy} - 2xz \partial_{xz} - 2yz \partial_{yz} - 2x \partial_x - 2y \partial_y - 2z \partial_z. \quad (9)$$

一旦获得了式(8)和(9),就可以使用一些已有的方法来表示在光滑曲面上定义的曲面算子.对于不同的曲面,由于其法向量不同,因此式(8)和(9)也不同.非本征方法简单高效、精度高,但是需要曲面的解析法向量.

## 1.2 CTL-PINN

CTL-PINN 的训练过程如图 1 所示,可分为三个阶段:标准 PINN、课程学习和迁移学习.第一阶段使用标准 PINN 在短时间内进行求解;第二阶段采用课程学习扩大 PINN 训练的时域;第三阶段采用迁移学习对第二阶段后的模型进行时域上的迁移.具体过程可参考算法 1.标准 PINN 可以很好地求解短时间问题,这也是 CTL-PINN 得以实现的基础,第二阶段扩大了单个神经网络的可求解时域,并且提高了后续迁移的效率.第三阶段本质上是一种堆叠分解方法<sup>[21]</sup>,其在第二阶段后进一步扩大了可求解域.每一步的训练过程可视为一个单独的 PINN 求解过程,某一步课程学习与迁移学习下的神经网络结构如图 2 所示.课程学习和迁移学习阶段的原理详见 1.2.2 和 1.2.3 小节.

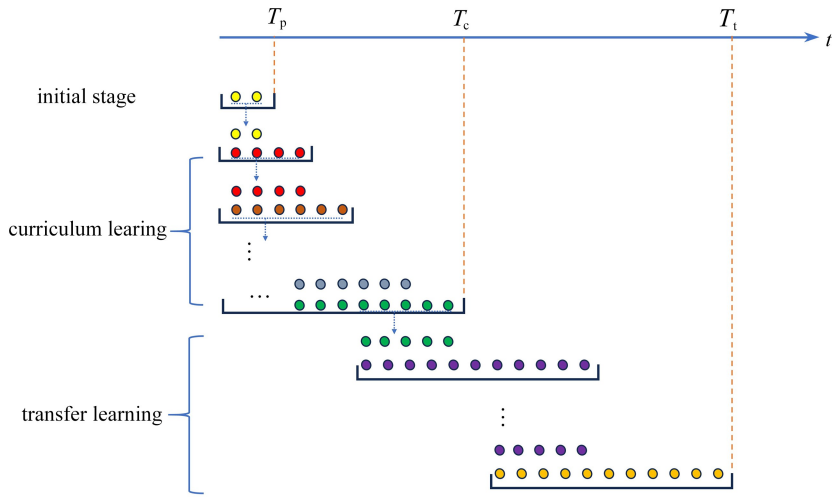


图 1 CTL-PINN 求解步骤示意图

Fig. 1 Steps of the CTL-PINN

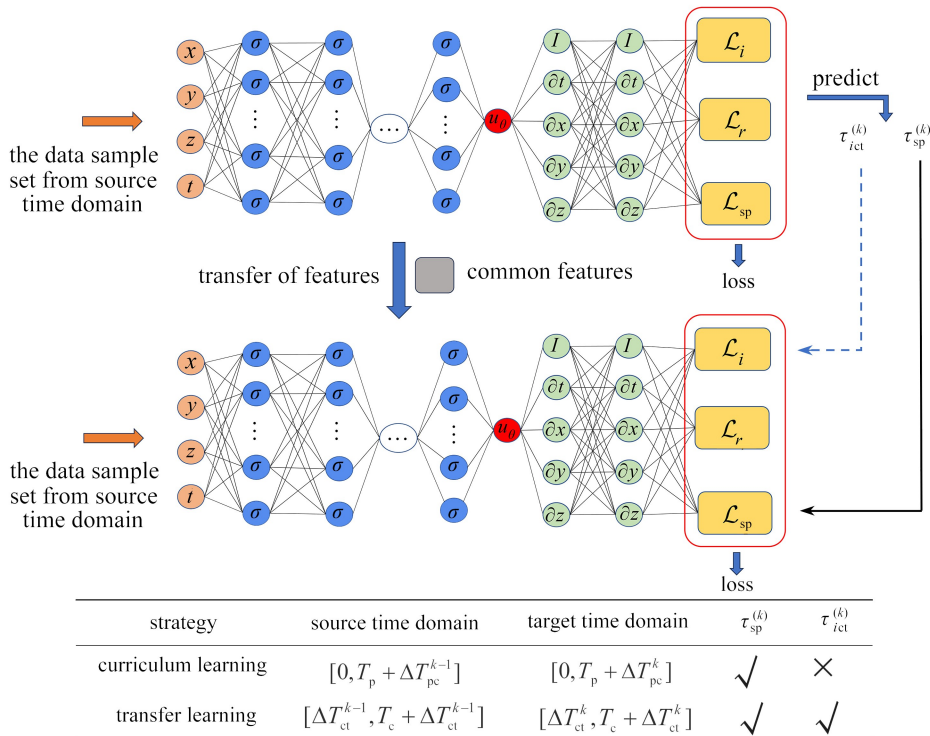


图 2 CTL-PINN 结构示意图

Fig. 2 Structure of the CTL-PINN

注 为了解释图中的颜色,读者可以参考本文的电子网页版本,后同。

### 1.2.1 标准物理神经网络基本理论 (PINN)

PINN 通过最小化损失函数  $\mathcal{L}(\theta; \tau_i, \tau_r)$  训练神经网络参数  $\theta$  以得到方程的近似解,  $\mathcal{L}(\theta; \tau_i, \tau_r)$  由初始条件以及配点处的偏微分方程残差项组成,其包含了方程的物理信息,其表达式如下所示:

$$\mathcal{L}(\theta; \tau_i, \tau_r) = w_i \mathcal{L}_i(\theta; \tau_i) + w_r \mathcal{L}_r(\theta; \tau_r), \tag{10}$$

其中

$$\mathcal{L}_i(\theta; \tau_i) = \frac{1}{N_i} \sum_{i=1}^{N_i} |u_\theta(x_i, y_i, z_i, 0) - I(x_i, y_i, z_i)|^2, \tag{11}$$

$$\mathcal{L}_r(\theta; \tau_r) = \frac{1}{N_r} \sum_{i=1}^{N_r} \left| \frac{\partial u_\theta(x_i, y_i, z_i, t_i)}{\partial t} - (D\Delta_s - \mathbf{v} \cdot \nabla_s + c)u_\theta(x_i, y_i, z_i, t_i) - f \right|^2, \quad (12)$$

其中  $w_i, w_r$  分别是损失函数组成部分的权重;  $\mathcal{L}_i$  表示初始条件的监督学习损失,  $\mathcal{L}_r$  表示控制方程的残差损失;  $N_i$  和  $N_r$  分别为初始条件和方程残差的标记点个数;  $u_\theta$  表示对应数据点上的预测值. 将初始条件数据集和方程残差数据集分别保存为

$$\tau_i = \{ (x_i, y_i, z_i, I(x_i, y_i, z_i)) \mid (x_i, y_i, z_i) \in \Omega \}_{i=1}^{N_i}, \quad (13)$$

$$\tau_r = \{ (x_i, y_i, z_i, t_i) \in \Omega \times (0, T] \}_{i=1}^{N_r}. \quad (14)$$

使用  $\Sigma$  表示数据集的总和, 即  $\Sigma = \{ \tau_i, \tau_r \}$ , 有  $\mathcal{L}(\theta; \Sigma) = \mathcal{L}(\theta; \tau_i, \tau_r)$ . 通过最小化  $\mathcal{L}(\theta; \Sigma)$ , PINN 可以在求解域中得出满足物理信息的预测. PINN 的训练就是一个求解损失函数最小值的优化过程, 即

$$\bar{\theta} = \arg \min_{\theta} \mathcal{L}(\theta; \Sigma). \quad (15)$$

本文以相对误差范数  $L_2$  衡量神经网络的性能好坏,  $L_2$  如下所示:

$$L_2 = \frac{\sqrt{\sum_{i=1}^N |u_\theta(x_i, y_i, z_i, t_i) - u(x_i, y_i, z_i, t_i)|^2}}{\sqrt{\sum_{i=1}^N |u(x_i, y_i, z_i, t_i)|^2}}, \quad (16)$$

其中  $u_\theta(x_i, y_i, z_i, t_i)$  为对应点的预测值;  $u(x_i, y_i, z_i, t_i)$  为对应点的真值;  $N$  为测试点的个数.

标准的 PINN 在短时间问题上求解效果较好, 但是难以求解长时间历程问题, 为了将 PINN 应用于曲面长时间对流扩散行为的模拟, CTL-PINN 基于 PINN 在短时段内的训练模型, 采用课程学习和迁移学习策略求解更长时间下的结果.

### 1.2.2 课程学习

课程学习, 其思想核心是通过简单的问题进行学习, 再逐步过渡到复杂问题. 本文中通过此策略扩大训练的时域, 若经过  $n$  次时域扩大, 训练的时域由  $[0, T_p]$  最终扩大至  $[0, T_c]$ , 其过程可描述为

$$[0, T_p] \rightarrow [0, T_p + \Delta T_{pc}^1] \rightarrow \cdots \rightarrow [0, T_p + \Delta T_{pc}^{m-1}] \rightarrow [0, T_c],$$

$$\Delta T_{pc}^k > \Delta T_{pc}^{k-1}, \Delta T_{pc}^m = T_c - T_p, k = 2, 3, \dots, n. \quad (17)$$

课程学习某一训练步中, 源时域  $[0, T_p + \Delta T_{pc}^{k-1}]$  的模型与目标时域  $[0, T_p + \Delta T_{pc}^k]$  模型具有相同控制方程与初始条件. 为提高训练效率, 可将源时域训练完成的模型参数  $\theta_{pc}^{(k-1)}$  用于初始化第  $k$  步模型参数, 并采用源时域已训练成功的模型在两时域交集  $[0, T_p + \Delta T_{pc}^{k-1}]$  内进行预测得到  $u_{\theta_{pc}^{(k-1)}}^{k-1}$ , 将预测结果数据保存为  $\tau_{sp}^{(k)}$ , 用于目标时域模型训练.  $\tau_{sp}^{(k)}$  可表示为

$$\tau_{sp}^{(k)} = \{ (x_i, y_i, z_i, t_i, u_{\theta_{pc}^{(k-1)}}^{k-1}) \mid (x_i, y_i, z_i, t_i) \in \Omega \times [0, T_p + \Delta T_{pc}^{k-1}] \}_{i=1}^{N_{sp}^{(k)}}. \quad (18)$$

使用  $\Sigma_{pc}^{(k)} = \{ \tau_{ipc}^{(k)}, \tau_{rpe}^{(k)}, \tau_{sp}^{(k)} \}$  表示时域  $[0, T_p + \Delta T_{pc}^k]$  的训练数据集, 则时域  $[0, T_p + \Delta T_{pc}^k]$  内神经网络的训练过程可表示为如下的优化问题:

$$\theta_{pc}^{(k)} = \arg \min_{\theta} \mathcal{L}(\theta; \Sigma_{pc}^{(k)}, \theta_{pc}^{(k-1)}). \quad (19)$$

损失函数如下:

$$\mathcal{L}(\theta; \Sigma_{pc}^{(k)}, \theta_{pc}^{(k-1)}) = w_i \mathcal{L}_i(\theta; \tau_{ipc}^{(k)}) + w_r \mathcal{L}_r(\theta; \tau_{rpe}^{(k)}) + w_{sp} \mathcal{L}_{sp}(\theta; \tau_{sp}^{(k)}), \quad (20)$$

其中  $w_{sp}$  为额外的监督学习部分的权重,  $\mathcal{L}_{sp}$  为额外监督学习的损失,

$$\mathcal{L}_{sp}(\theta; \tau_{sp}^{(k)}) = \frac{1}{N_{sp}^{(k)}} \sum_{i=1}^{N_{sp}^{(k)}} |u_\theta(x_i, y_i, z_i, t_i) - u_{\theta_{pc}^{(k-1)}}^{k-1}(x_i, y_i, z_i, t_i)|^2, \quad (21)$$

其中  $N_{sp}^{(k)}$  是额外监督学习点的数量,  $u_{\theta_{pc}^{(k-1)}}^{k-1}(x_i, y_i, z_i, t_i)$  是前一模型的预测值.

#### 算法 1 CTL-PINN 求解长时间历程问题步骤

Input:

1 神经网络结构;

时域扩大间隔:  $[0, T_p], [0, T_p + \Delta T_{pc}^1], \dots, [0, T_c]$



时域迁移间隔:  $[0, T_c], [\Delta T_{ct}^1, T_c + \Delta T_{ct}^1], \dots, [\Delta T_{ct}^n, T_t]$

2 课程学习部分:

3 初始化神经网络

4 标准 PINN 求解时域  $[0, T_p]$  内方程, 输出神经网络参数  $\theta_{pc}^{(1)}$

5 for  $k = 2, 3, \dots, n$  do:

6 生成监督学习数据集  $\tau_{pc}^{(k)}$  并构成训练数据集  $\Sigma_{pc}^{(k)} = \{\tau_{ipc}^{(k)}, \tau_{rpc}^{(k)}, \tau_{sp}^{(k)}\}$

7 最小化损失函数  $\theta_{pc}^{(k)} = \arg \min_{\theta} \mathcal{L}(\theta; \Sigma_{pc}^{(k)}, \theta_{pc}^{(k-1)})$

8 更新神经网络参数  $\theta_{pc}^{(k)}$  作为下一步的初始化参数

9 end

10 end

11 迁移学习部分:

12  $\theta_{pc}^{(n)}$  作为初始化参数

13 for  $k = 2, 3, \dots, m$  do:

14 生成监督学习数据集  $\tau_{ct}^{(k)}$  并构成训练数据集  $\Sigma_{ct}^{(k)} = \{\tau_{ict}^{(k)}, \tau_{ret}^{(k)}, \tau_{sp}^{(k)}\}$

15 最小化损失函数  $\theta_{ct}^{(k)} = \arg \min_{\theta} \mathcal{L}(\theta; \Sigma_{ct}^{(k)}, \theta_{ct}^{(k-1)})$

16 保存神经网络参数  $\theta_{ct}^{(k)}$  作为下一模型的初始化参数

17 end

18 end.

### 1.2.3 迁移学习

迁移学习,其思想是应用已有的知识学习新的知识,本文中以课程学习后的最终时域  $[0, T_c]$  的模型为基础,将其求解时域扩大至  $[0, T_t]$ 。值得一提的是,课程学习中最后模型可得到方程  $[0, T_c]$  内的解,而迁移学习下的  $[0, T_t]$  内的解是由多个模型得到的,若经  $m$  次时域迁移后,可求解域由时域  $[0, T_c]$  最终扩大至时域  $[0, T_t]$ , 迁移过程描述为

$$[0, T_c] \rightarrow [\Delta T_{ct}^1, T_c + \Delta T_{ct}^1] \rightarrow \dots \rightarrow [\Delta T_{ct}^{m-1}, T_c + \Delta T_{ct}^{m-1}] \rightarrow [\Delta T_{ct}^m, T_t],$$

$$\Delta T_{ct}^k > \Delta T_{ct}^{k-1}, \Delta T_{ct}^m = T_t - T_c, k = 2, 3, \dots, m. \quad (22)$$

与课程学习阶段相同,在从时域迁移学习阶段同样使用上一步训练结束的神经网络参数  $\theta_{ct}^{(k-1)}$  初始化下一步网络参数,并将上一步模型在两时域交集  $[\Delta T_{ct}^k, T_c + \Delta T_{ct}^{k-1}]$  内的预测值  $u_{\theta_{ct}^k}^{k-1}$  用作下一步的额外监督点,则额外的监督数据  $\tau_{sp}^{(k)}$  表示为

$$\tau_{sp}^{(k)} = \{(x_i, y_i, z_i, t_i, u_{\theta_{ct}^k}^{k-1}) \mid (x_i, y_i, z_i, t_i) \in \Omega \times [\Delta T_{ct}^k, T_c + \Delta T_{ct}^{k-1}]\}_{i=1}^{N_{sp}}. \quad (23)$$

对于下一模型,初始条件发生变化,使用上一模型的预测值作为初始条件,初始条件数据  $\tau_{ict}^{(k)}$  如下:

$$\tau_{ict}^{(k)} = \{(x_i, y_i, z_i, \Delta T_{ct}^k, u_{\theta_{ct}^k}^{k-1}(x_i, y_i, z_i, \Delta T_{ct}^k)) \mid (x_i, y_i, z_i, \Delta T_{ct}^k) \in \Omega\}_{i=1}^{N_i}. \quad (24)$$

使用  $\Sigma_{ct}^{(k)} = \{\tau_{ict}^{(k)}, \tau_{ret}^{(k)}, \tau_{sp}^{(k)}\}$  表示第  $k$  步神经网络的训练数据集,在时域  $[\Delta T_{ct}^k, T_c + \Delta T_{ct}^k]$  中存在以下优化问题:

$$\theta_{ct}^{(k)} = \arg \min_{\theta} \mathcal{L}(\theta; \Sigma_{ct}^{(k)}, \theta_{ct}^{(k-1)}), \quad (25)$$

其中损失函数为

$$\mathcal{L}(\theta; \Sigma_{ct}^{(k)}, \theta_{ct}^{(k-1)}) = w_i \mathcal{L}_i(\theta; \tau_{ict}^{(k)}) + w_r \mathcal{L}_r(\theta; \tau_{ret}^{(k)}) + w_{sp} \mathcal{L}_{sp}(\theta; \tau_{sp}^{(k)}). \quad (26)$$

## 2 数值算例及讨论

本节采用以上方法,将 CTL-PINN 结合非本征技术应用于求解曲面上瞬态的对流扩散方程,并通过 3 个算例验证其有效性,本文中算例神经网络的激活函数均为 tanh 函数;为加快训练速度,采用 L-BFGS 优化器进行优化,L-BFGS 优化器是一种基于拟 Newton 方法的优化算法,在优化过程中会自适应地调整学习率;最大迭代次数为 50 000 次,损失函数权重  $w_i = w_r = w_{sp} = 1$ 。

## 2.1 球面线性对流扩散问题

将式(6)和(7)代入(1)中, 曲面上的瞬态对流扩散方程可写为

$$\frac{\partial u}{\partial t} = (a\Delta_s - \mathbf{v} \cdot \nabla_s + c)u(x, y, z, t) + f, \quad (27)$$

其中  $a = 1, \mathbf{v} = \mathbf{0}, c = 0, f$  为源项. 精确解为  $u = \sin(x + y + z)\sin(t/2)$ . 为了保证采样的随机性, 每次对 PINN 进行训练时, 采用拉丁超立方采样<sup>[27]</sup> 在对应求解的时空域内选取  $N_t = 1\,000$  个点作为训练点;  $N_i = 500$  个初始条件监督点; 并在时空域内随机取 1 000 个额外的监督学习点 (后续算例采样方式相同). 每次时域扩大的步长取 0.5 s. 对时域  $[0\text{ s}, 5\text{ s}]$  进行模拟, 测试在不同的神经网络结构下的训练结果, 其结果如表 1 所示. 对于 8 层神经网络增加每层神经元个数误差基本不变, 而 13 和 15 层随着每层神经元个数增加误差总体表现减小趋势, 17 层时误差先随着神经元个数减小而后增加; 并且对于不同单层神经元个数, 总体上均随着层数增加误差反而增大, 这是因为对于时域  $[0\text{ s}, 5\text{ s}]$  的问题较为简单, 而更多的神经元与隐藏层数目虽然增加了神经网络拟合能力, 也使得其更易陷入局部最优解. 考虑到需要计算更长时间下的问题, 为了保证神经网络有足够的拟合能力, 采用 15 层神经网络, 隐藏层每层都为 50 个神经元 (后续算例相同).

表 1 不同神经网络结构下的  $L_2$  误差

Table 1 Errors  $L_2$  under different neural network structures

	10 neurons	30 neurons	50 neurons
8 layers	$2.76 \times 10^{-4}$	$2.32 \times 10^{-4}$	$2.86 \times 10^{-4}$
13 layers	$7.53 \times 10^{-4}$	$2.85 \times 10^{-4}$	$3.58 \times 10^{-4}$
15 layers	$6.43 \times 10^{-4}$	$5.15 \times 10^{-4}$	$3.24 \times 10^{-4}$
17 layers	$4.33 \times 10^{-3}$	$3.48 \times 10^{-4}$	$6.71 \times 10^{-4}$

本算例对不同时间步长下 CTL-PINN 的训练结果进行了探究, 其结果如图 3 所示. 可以看出, 时间步长越短, 其训练的精度越高, 但是步长越短, 模型迭代的次数越多, 对于时间步长的选择需要综合考虑. 以  $\Delta t = 0.5\text{ s}$  为步长采用课程学习策略, 得到  $[0\text{ s}, 80\text{ s}]$  的模型, 输出其误差的变化, 与标准的 PINN 进行比较, 其结果如图 4 所示. 可以看出, 标准的 PINN 在短时域内可以得到好的结果, 这也是 CTL-PINN 得以实现的基础, 但是随着时域的增大, 其误差急剧增大, 而通过 CTL-PINN 训练的结果更为精确和稳定. CTL-PINN 前 8 步课程学习的损失函数随迭代次数收敛的曲线如图 5 所示. 可以看出, 监督学习损失 ( $\mathcal{L}_i + \mathcal{L}_{sp}$ ) 比方程残差损失 ( $\mathcal{L}_r$ ) 下降更快; 并且自第一步后损失函数下降速度明显加快, 这表明使用上一步的神经网络参数初始化下一步神经网络提高了训练效率.

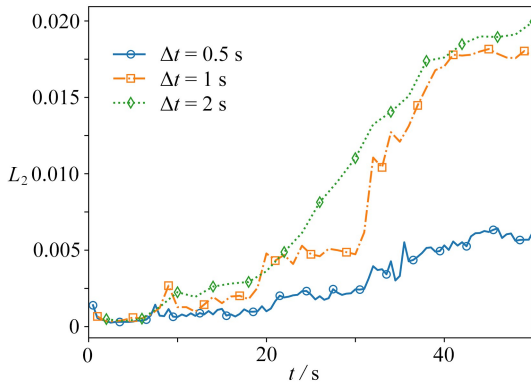


图 3 不同时间步长的训练结果

Fig. 3 Training results with different time steps

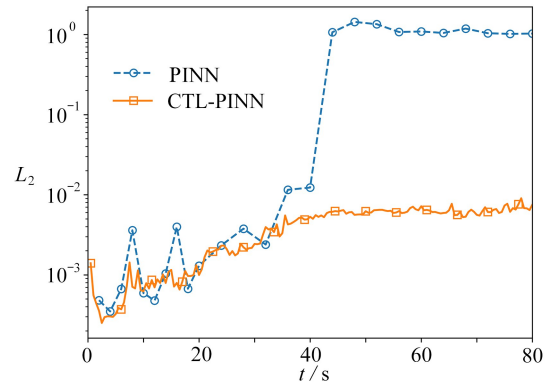


图 4 CTL-PINN 与 PINN 的误差随时域变化

Fig. 4 The errors of the CTL-PINN and the PINN changing with the time domain

当时域扩大到一定程度后, 为提高训练效率, 采用迁移学习策略对课程学习得到的时域  $[0\text{ s}, 80\text{ s}]$  模型进行时域迁移. 每次迁移使用上一模型预测初始点和额外监督学习点各 1 000, 迁移步长均为 10 s, 迁移 4 步, 其结果如表 2 所示. 其某一瞬时的绝对误差分布如图 6 所示, 其中前 80 s 由  $[0\text{ s}, 80\text{ s}]$  模型预测, 120 s 由

[40 s, 120 s] 模型预测.

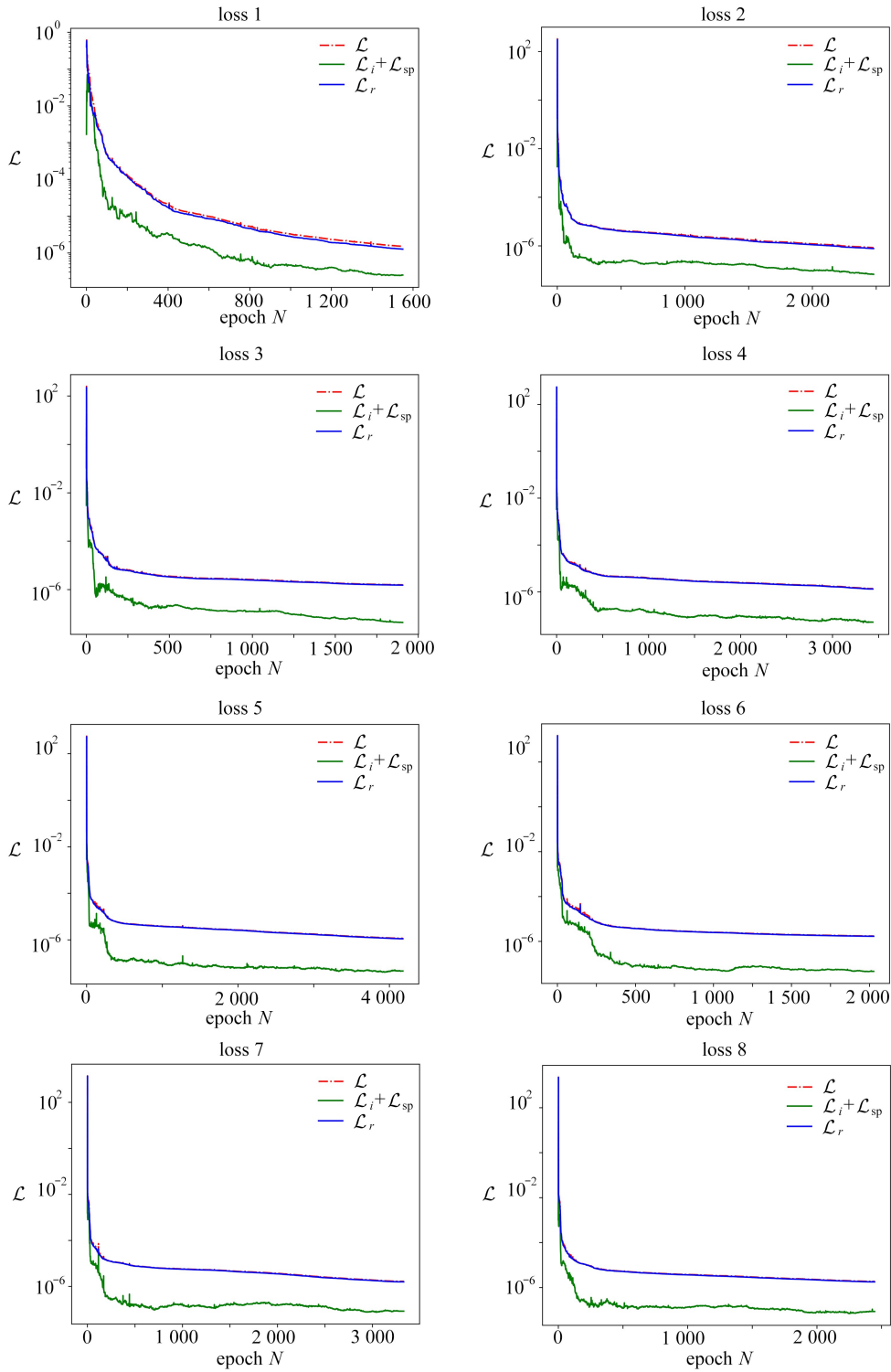


图 5 CTL-PINN 前 8 步损失函数收敛曲线

Fig. 5 Convergence curves of loss functions for the 1st 8 steps of the CTL-PINN

表 2 以步长 10 s 进行迁移学习的  $L_2$  误差

Table 2 Errors  $L_2$  of the transfer learning with a step size of 10 s

time domain $T/s$	[0, 80]	[10, 90]	[20, 100]	[30, 110]	[40, 120]
$L_2$	$7.01 \times 10^{-3}$	$7.61 \times 10^{-3}$	$9.67 \times 10^{-3}$	$1.18 \times 10^{-2}$	$1.05 \times 10^{-2}$



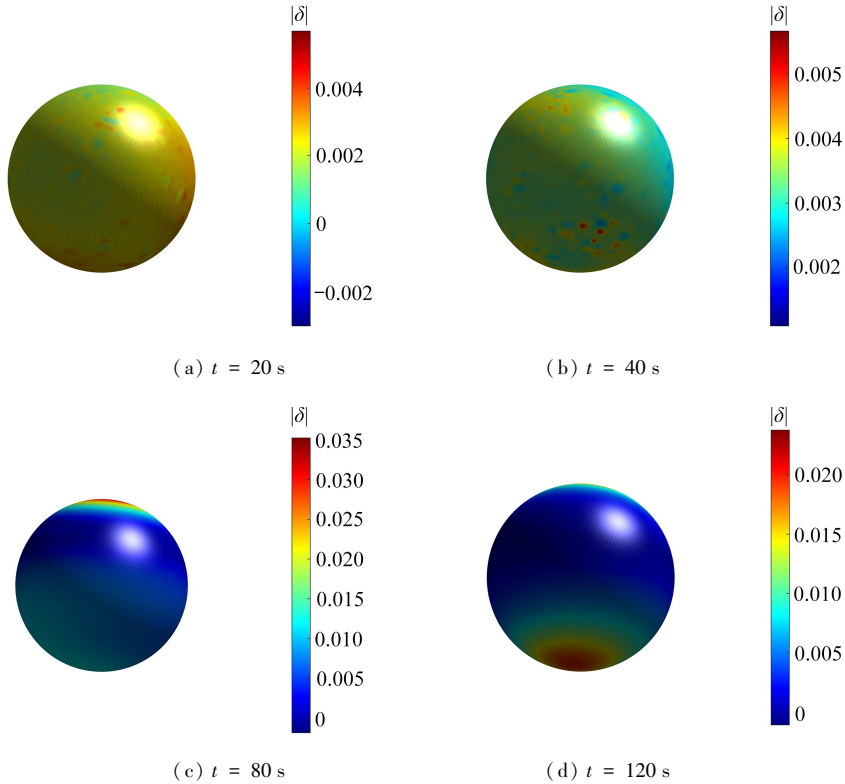


图6 不同时刻的绝对误差分布

Fig. 6 The distributions of absolute errors at different moments

### 2.2 球面非线性对流扩散问题

球面上的非线性对流扩散方程为

$$\frac{\partial u}{\partial t} = (a\Delta_S - \mathbf{v} \cdot \nabla_S + c)u(x, y, z, t) + g(u) + f, \tag{28}$$

其中  $g(u) = u^2$  并且  $a = 1, \mathbf{v} = [0, 0, 1]^T, c = 0$ . 精确解为  $\sin(x + y + z)\cos(t)$ . 时域扩大步长取  $0.5 \text{ s}$ , 取点方法同 2.1 小节, 其不同的额外监督学习点数  $N_{sp}$  下误差随时域  $[0, t]$  的变化如图 7 所示, 可以看出适量的额外的监督学习点可以提高精度, 但是由于这些点的预测值并非真实值, 过多的监督学习点反而会影响训练的精度. 分别将  $(1, 0, 0)$  和  $(0, 1, 0)$  两点在不同时间下的预测值和真值输出, 如图 8 所示, 可以看出, 无额外监督学习点时的效果最差.

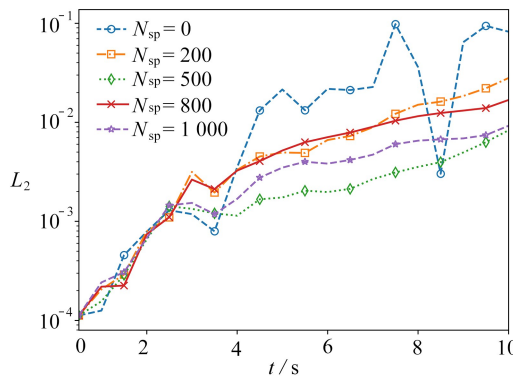


图7 不同额外监督学习点数量下的误差随时域  $[0, t]$  的变化

Fig. 7 The errors changing over time domain  $[0, t]$  with different numbers of extra supervised learning points

### 2.3 复杂曲面上的对流扩散问题

以上算例都是在简单的球面上进行的, 为探究该方法用于复杂曲面上的效果, 本算例针对复杂曲面上的

对流扩散行为进行探究,采用对流扩散方程(28),其中  $g(u) = u^2$  并且  $a = 1, v = \mathbf{0}, c = 0$ , 精确解取  $u = \sin(x + y + z)e^{-\nu t} \sin(3t) + 1$ , 此处考虑的曲面方程如下:

蝴蝶结曲面(bretzel 2)

$$S = (x^2(1 - x^2) - y^2)^2 + \frac{1}{2}z^2 - \frac{1}{40}.$$

等距积面(CDP)

$$s = \sqrt{(x - 1)^2 + y^2 + z^2} \sqrt{(x + 1)^2 + y^2 + z^2} \sqrt{x^2 + (y - 1)^2 + z^2} \sqrt{x^2 + (y + 1)^2 + z^2} - 1.1.$$

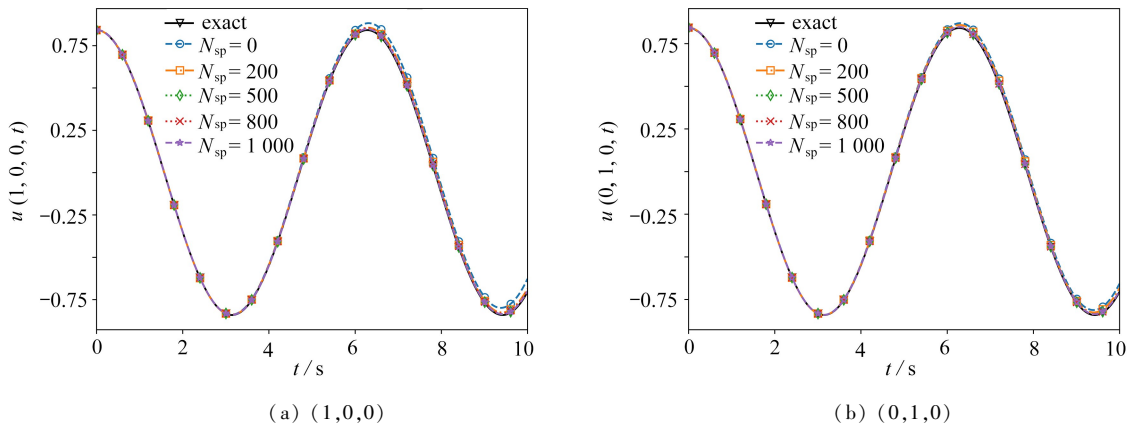


图 8 不同额外监督学习点数量下两点的预测值以及精确解

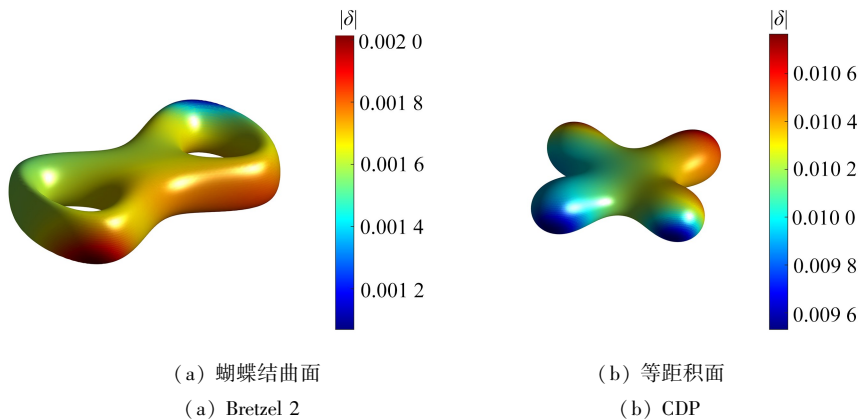
Fig. 8 The exact and predicted values under different numbers of extra supervised learning points

为描述更为复杂的曲面并保证采样的随机性,在 bretzel 2 和 CDP 两曲面上分别取得 368 458 和 399 310 个点,每次训练时在其与所求时域构成的时空域上均随机取  $N_t = 3\ 000$  个点作为训练点,  $N_i = 3\ 000$ . 以 0.5 s 为课程学习步长,由 [0 s, 0.5 s] 模型得到 [0 s, 4 s] 模型,其不同额外监督点数  $N_{sp}$  下的结果如表 3 所示.与图 7 结合分析,监督学习点在小时域内的效果并不明显,但从长远看,可使得模型在时域扩大下更为稳定,综合来看可选取训练点数的 50%.而后对 [0 s, 4 s] 模型以步长 3 s 进行迁移学习(bretzel 2 采用  $N_{sp} = 1\ 500$  模型; CDP 采用  $N_{sp} = 2\ 400$  模型),每次迁移使用上一模型预测初始点和额外监督学习点各 2 000 个,其结果如表 4 所示.对于该问题,19 s 时便已稳定,输出  $t = 19$  s 时的绝对误差如图 9 所示.

表 3 复杂曲面上不同  $N_{sp}$  下模型  $L_2$  误差

Table 3 The  $L_2$  errors of the model at different  $N_{sp}$  values on the complex surface

$N_{sp}$	0	600	1 200	1 500	1 800	2 400	3 000
bretzel 2	$1.04 \times 10^{-3}$	$1.14 \times 10^{-3}$	$1.23 \times 10^{-3}$	$4.96 \times 10^{-4}$	$2.62 \times 10^{-3}$	$3.61 \times 10^{-3}$	$3.48 \times 10^{-4}$
CDP	$5.27 \times 10^{-4}$	$8.83 \times 10^{-4}$	$1.41 \times 10^{-3}$	$1.68 \times 10^{-3}$	$3.54 \times 10^{-3}$	$4.93 \times 10^{-4}$	$7.46 \times 10^{-3}$



(a) 蝴蝶结曲面 (a) Bretzel 2 (b) 等距积面 (b) CDP

图 9  $t = 19$  s 时两个曲面的绝对误差分布

Fig. 9 The distributions of absolute errors on the 2 surfaces at  $t = 19$  s

表4 复杂曲面以步长3 s进行迁移学习的 $L_2$ 误差Table 4 The  $L_2$  errors of the transfer learning on the complex surface with a step size of 3 s

time domain $T/s$	[0,4]	[3,7]	[6,10]	[9,13]	[12,16]	[15,19]
bretzel 2	$1.04 \times 10^{-3}$	$1.14 \times 10^{-3}$	$1.23 \times 10^{-3}$	$4.96 \times 10^{-4}$	$2.62 \times 10^{-3}$	$3.61 \times 10^{-3}$
CDP	$5.27 \times 10^{-4}$	$8.83 \times 10^{-4}$	$1.41 \times 10^{-3}$	$1.68 \times 10^{-3}$	$3.54 \times 10^{-3}$	$4.93 \times 10^{-4}$

### 3 总结与展望

本文使用课程学习和迁移学习技术结合 PINN 用于求解曲面长时间对流扩散问题, PINN 避免了传统数值方法中网格划分、网格质量等问题, 并且在短时间问题上可以得到理想的效果, 但是随着时域的增大, 其求解效果越来越差. 本文通过课程学习和迁移学习的方法在长时间问题上得到了更为精确的结果, 并通过算例进行了验证. 我们可以做出以下的总结与展望:

1) 通过 CTL-PINN 可以得到比 PINN 方法更为精确的解, 但是由于需要反复导入模型训练, 其相比 PINN 更为繁琐, 必须要考虑如何在增大步长的同时保证模型精度.

2) 额外监督学习点可以提高神经网络的训练效率, 但是由于额外监督学习的数据是由神经网络预测得到的, 其与精确值之间存在一定误差, 过多的额外监督学习点会导致过大的误差累积.

3) 本文只求解了曲面上的正问题, 后续可以更进一步地将此算法用于求解曲面上的反问题.

#### 参考文献 (References):

- [1] SAYEVAND K, MACHADO J T, MASTI I. Analysis of dual Bernstein operators in the solution of the fractional convection-diffusion equation arising in underground water pollution[J]. *Journal of Computational and Applied Mathematics*, 2022, **399**: 113729.
- [2] NAZEMIFARD N, MASLIYAH J H, BHATTACHARJEE S. Particle deposition onto charge heterogeneous surfaces: convection-diffusion-migration model[J]. *Langmuir*, 2006, **22**(24): 9879-9893.
- [3] BARREIRA R, ELLIOTT C M, MADZVAMUSE A. The surface finite element method for pattern formation on evolving biological surfaces[J]. *Journal of Mathematical Biology*, 2011, **63**(6): 1095-1119.
- [4] 殷雅俊. 曲面物理和力学[J]. 力学与实践, 2011, **33**(6): 1-8. (YIN Yajun. Physics and mechanics on curved surfaces[J]. *Mechanics in Engineering*, 2011, **33**(6): 1-8. (in Chinese))
- [5] CASULLI V. A semi-implicit finite difference method for non-hydrostatic, free-surface flows[J]. *International Journal for Numerical Methods in Fluids*, 1999, **30**(4): 425-440.
- [6] DZIUK G, ELLIOTT C M. Finite elements on evolving surfaces[J]. *IMA Journal of Numerical Analysis*, 2006, **27**(2): 262-292.
- [7] 张一鸣, 王雪雅, 冯春, 等. 连续体的有限单元法课程理论与上机课程建设[J]. 高教学刊, 2022, **8**(16): 88-91. (ZHANG Yiming, WANG Xueya, FENG Chun, et al. Continuum finite element method curriculum theory and computer-based curriculum construction[J]. *Journal of Higher Education*, 2022, **8**(16): 88-91. (in Chinese))
- [8] CALHOUN D A, HELZEL C. A finite volume method for solving parabolic equations on logically Cartesian curved surface meshes[J]. *SIAM Journal on Scientific Computing*, 2010, **31**(6): 4066-4099.
- [9] TANG Z, FU Z, CHEN M, et al. An efficient collocation method for long-time simulation of heat and mass transport on evolving surfaces[J]. *Journal of Computational Physics*, 2022, **463**: 111310.
- [10] KARNIADAKIS G E, KEVREKIDIS I G, LU L, et al. Physics-informed machine learning[J]. *Nature Reviews Physics*, 2021, **3**(6): 422-440.
- [11] RAISSI M, PERDIKARIS P, KARNIADAKIS G E. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations[J]. *Journal of Computational Physics*, 2019, **378**: 686-707.
- [12] YANG L, MENG X H, KARNIADAKIS G E. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data[J]. *Journal of Computational Physics*, 2021, **425**: 109913.

- [13] MAO Z P, MENG X H. Physics-informed neural networks with residual/gradient-based adaptive sampling methods for solving partial differential equations with sharp solutions[J]. *Applied Mathematics and Mechanics (English Edition)*, 2023, **44**(7): 1069-1084.
- [14] CAI S Z, MAO Z P, WANG Z C, et al. Physics-informed neural networks (PINNs) for fluid mechanics: a review[J]. *Acta Mechanica Sinica*, 2021, **37**(12): 1727-1738.
- [15] 林云云, 郑素佩, 封建湖, 等. 间断问题扩散正则化的 PINN 反问题求解算法[J]. *应用数学和力学*, 2023, **44**(1): 112-122. (LIN Y Yunyun, ZHENG Supei, FENG Jianhu, et al. Diffusive regularization inverse PINN solutions to discontinuous problems[J]. *Applied Mathematics and Mechanics*, 2023, **44**(1): 112-122. (in Chinese))
- [16] BELLMAN R. Dynamic programming[J]. *Science*, 1966, **153**(3731): 34-37.
- [17] FANG Z, ZHAN J. A physics-informed neural network framework for PDEs on 3D surfaces: time independent problems[J]. *IEEE Access*, 2020, **8**: 26328-26335.
- [18] TANG Z C, FU Z J, REUTSKIY S. An extrinsic approach based on physics-informed neural networks for PDEs on surfaces[J]. *Mathematics*, 2022, **10**(16): 2861.
- [19] 汤卓超, 傅卓佳. 基于物理信息的神经网络求解曲面上对流扩散方程[J]. *计算力学学报*, 2023, **40**(2): 216-222. (TANG Zhuochao, FU Zhuojia. Physics-informed neural networks for solving convection-diffusion equations on surfaces[J]. *Chinese Journal of Computational Mechanics*, 2023, **40**(2): 216-222. (in Chinese))
- [20] KRISHNAPRIYAN A, GHOLAMI A, ZHE S, et al. Characterizing possible failure modes in physics-informed neural networks[J]. *Advances in Neural Information Processing Systems*, 2021, **34**: 26548-26560.
- [21] PENWARDEN M, JAGTAP A D, ZHE S, et al. A unified scalable framework for causal sweeping strategies for physics-informed neural networks (PINNs) and their temporal decompositions[J]. *March Learning*, 2023, **493**: 112464..
- [22] MENG X H, LI Z, ZHANG D K, et al. PPINN: parareal physics-informed neural network for time-dependent PDEs[J]. *Computer Methods in Applied Mechanics and Engineering*, 2020, **370**: 113250.
- [23] JAGTAP A D, KARNIADAKIS G E. Extended physics-informed neural networks (XPINNs): a generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations [J]. *Communications in Computational Physics*, 2020, **28**(5): 2002-2041.
- [24] GUO J, YAO Y, WANG H, et al. Pre-training strategy for solving evolution equations based on physics-informed neural networks[J]. *Journal of Computational Physics*, 2023, **489**: 112258.
- [25] GOSWAMI S, ANITESCU C, CHAKRABORTY S, et al. Transfer learning enhanced physics informed neural network for phase-field modeling of fracture[J]. *Theoretical and Applied Fracture Mechanics*, 2020, **106**: 102447.
- [26] MÜNZER M, BARD C. A curriculum-training-based strategy for distributing collocation points during physics-informed neural network training[R/OL]. 2022[2024-02-22]. <https://arxiv.org/pdf/2211.11396>.
- [27] LOH W L. On Latin hypercube sampling[J]. *The Annals of Statistics*, 1996, **24**(5): 2058-2080.