

文章编号: 1000-8055(2026)08-20250324-21

doi: 10.13224/j.cnki.jasp.20250324

融合残差学习与物理信息神经网络的 流场预测方法

由儒全^{1,2}, 陆淳源¹, 刘润洲³, 施锦程³, 车俊新¹, 李海旺^{1,2,3}

- (1. 北京航空航天大学 航空发动机研究院, 北京 100191;
2. 北京航空航天大学 航空发动机气动热力国家级重点实验室, 北京 100191;
3. 太行国家实验室 数理与基础理论研究中心, 成都 610000)

摘 要: 研究了基于残差神经网络(ResNet)与物理信息神经网络(PINN)融合的高精度流场预测方法。基于数值模拟数据构建训练集, 建立融合残差神经网络的PINN模型(Res-PINN)用于求解圆柱绕流问题。该模型利用流场采样点的速度和压强信息进行训练, 以预测不同区域内的流场和压强分布。同时探究了网络结构类型、激活函数、训练集规模、网络层数、神经元数量及损失函数权重对预测结果的影响, 寻求网络的优化方法。结果表明: Res-PINN模型能有效重构圆柱绕流的流速和压强分布, 预测精度与直接数值模拟结果高度吻合, 物理量相对误差均低于5%; 对比分析显示, Res-PINN相较于传统PINN, 预测误差均下降超过45%, 显著提升了预测精度与稳定性; 多种网络超参数对预测效果均有明显影响, 优化时需综合考虑并平衡这些参数。

关键词: 物理信息神经网络; 圆柱绕流; 流场重构; 残差神经网络; 网络优化

中图分类号: V211.3

文献标志码: A

Flow field prediction method integrating residual learning and physics-informed neural network

YOU Ruquan^{1,2}, LU Chunyuan¹, LIU Runzhou³, SHI Jincheng³,
CHE Junxin¹, LI Haiwang^{1,2,3}

- (1. Research Institute of Aero-Engine, Beihang University, Beijing 100191, China;
2. National Key Laboratory of Science and Technology on Aero Engines Aero-thermodynamics,
Beihang University, Beijing 100191, China;
3. Mathematical and Basic Theory Research Center, Taihang Laboratory, Chengdu 610000, China)

Abstract: A high-precision prediction method for flow field was developed by integrating residual neural network (ResNet) with physics-informed neural network (PINN). Based on numerical simulation data, a PINN model integrating residual neural network (Res-PINN) solved the cylinder flow problem. The model used velocity and pressure data from sampled flow-field points to predict flow-field and pressure distributions in different regions. The influences of network type, activation function, dataset size, network layers, number of neurons and loss function weight on the prediction results were also explored. Results demonstrated that the Res-PINN effectively reconstructed flow-field velocity and pressure distributions. Its

收稿日期: 2025-07-11

作者简介: 由儒全(1991—), 男, 副研究员、博士生导师, 博士, 主要从事高温旋转部件的流动与换热测试、高效冷却技术等方面的研究。

E-mail: youruquan10353@buaa.edu.cn

通信作者: 李海旺(1980—), 男, 教授、博士生导师, 博士, 主要从事高温旋转部件的流动与换热、高效冷却技术、微尺度动力系统等方面的研究。E-mail: 09620@buaa.edu.cn

引用格式: 由儒全, 陆淳源, 刘润洲, 等. 融合残差学习与物理信息神经网络的流场预测方法[J]. 航空动力学报, 2026, 41(8): 20250324. YOU Ruquan, LU Chunyuan, LIU Runzhou, et al. Flow field prediction method integrating residual learning and physics-informed neural network[J]. Journal of Aerospace Power, 2026, 41(8): 20250324.

prediction accuracy was highly consistent with the direct numerical simulation results. The relative error of physical quantities was less than 5%. Comparative analysis showed that compared with traditional PINN, Res-PINN reduced prediction errors by over 45%, significantly improving the prediction accuracy and stability. Furthermore, multiple network hyperparameters had a significant impact on prediction performance, and optimization should be made to comprehensively consider and balance these parameters.

Keywords: physics-informed neural network; cylinder flow;
flow field reconstruction; residual neural network; network optimization

作为现代工业中“皇冠上的明珠”，航空发动机机械结构复杂精密，研制难度高，在技术领域上涵盖了工程热力学、机械、自动控制、材料科学、电子技术等多个学科领域，直接反映出一个国家综合的科学技术水平与发展程度。航空发动机中的旋转部件工作在高温、高压、高转速和高载荷的极端条件下，同时还需要在高紧凑的发动机中保证高寿命和高可靠地工作，为了其长期可靠工作，必须进行冷却。冷却结构设计是否合理直接决定了高温旋转部件工作的可靠性，进而影响着发动机的整机安全性和寿命。随着航空发动机技术的不断发展，对冷却结构设计要求不断提升，同时也对旋转部件的性能评估方法的准确性有了更高的要求。

计算流体动力学(computational fluid dynamics, CFD)自 20 世纪 60 年代伴随计算机技术兴起以来，历经半个多世纪的发展，已逐步迈向成熟阶段，并在空气动力学求解等诸多领域实现了广泛应用^[1]。在航空发动机冷却结构设计流程中，数值模拟技术凭借其在提升研制效率与质量、减少实物实验次数、降低研制风险和成本，以及加速研制进程等方面的显著优势，始终发挥着关键作用。在工程实际应用中，特别是航空发动机领域，雷诺平均应力模型(Reynolds-averaged Navier-Stokes, RANS)方法长期处于主导地位。然而，当处理复杂流动现象时，雷诺应力张量与应变率张量之间的本构关系不再遵循经典的 Boussinesq 假设下的线性对应法则，这种各向异性特征使得基于 RANS 方程的湍流模型封闭问题成为流体力学中的经典难题^[2]，结果会出现较大的偏差。此外，湍流模型中引入的经验参数通常基于特定工况的实验数据进行标定，这种数据依赖性严重制约了模型在不同流动条件下的泛化能力，显著限制了其工程应用的普适性^[3]。目前常用的 RANS 模型基本都是 20 世纪提出，主要有 Baldwin-Lomax (BL)模型^[4]、 $k-\varepsilon$ 模型^[5]、 $k-\omega$ 模型^[6]和 Spalart-All-

maras(SA)模型^[7]等。因此，CFD 亟需发展新方法。

随着计算机技术的发展与人工智能的广泛应用，能够结合机器学习与流体力学，给 CFD 方法带来新的突破。深度学习作为一种重要的机器学习改进方法^[8]，在湍流建模方面得到迅速发展与应用。在传统深度学习网络应用于方程求解时，其核心机制是通过对大规模训练集的采样数据执行多次迭代训练过程，促使网络学习并捕捉数据中的潜在模式与内在规律，进而实现对微分方程的高精度逼近与拟合^[9]。Wu 等^[10]利用卷积神经网络(CNN)实现 porous 介质渗透性的快速预测，该方法通过生成各种参数化的二维合成孔隙图像，并使用有限元模拟计算其渗透率来构建训练数据库。随后使用这种训练数据对 CNN 进行训练，使其能够从孔隙图像直接预测渗透性，显著提高了渗透性的预测精度，不仅大大减少了计算时间，而且对于复杂多变的孔隙结构具有良好的泛化能力。传统深度神经网络主要基于纯数据驱动范式构建，本质上是一个难以解读的“黑箱”问题^[11]。在流体力学这类数据采集困难、物理过程复杂的领域缺乏鲁棒性与可解释性。即使投入海量训练数据，由于未有效融入微分方程蕴含的先验物理约束，模型仅能机械地拟合输入与输出数据映射关系，导致物理机理的阐释能力存在显著局限。

近年来，物理信息神经网络^[12-13](physics informed neural network, PINN)得到了显著关注，该方法创新性地将偏微分方程的正向求解与参数反演问题转换为神经网络的优化训练任务，有效克服了传统数值方法在处理复杂反问题时的局限性。

为增强深度学习模型在偏微分方程系统求解中的可解释性与泛化能力，Raissi 等^[13]将偏微分方程嵌入深度学习架构，构建了基于物理信息的深度学习算法框架，并首次将其成功应用于偏微分方程正问题与反问题的数值求解。PINN 通过引入物理约束机制对神经网络输出进行正则化约束，利用稀疏测量数据与物理先验信息(如控制方

程及相关物理参数)实现物理过程的精准预测;同时,将偏微分方程中的偏导数直接通过自动微分算法进行数值近似^[14],从而实现数据驱动与物理模型的深度融合。这种结合了物理学知识的机器学习方法能够有效地求解各种物理问题,如流体力学、热传导和结构力学等^[15-17],在各种科学和工程领域都有着广泛的应用前景。

在流体力学研究范畴内,PINN展现出广阔的应用前景。作为描述流体运动的核心方程,Navier-Stokes(N-S)方程组存在高阶非线性耦合项,在数学建模过程中,此类非线性因素的有效处理始终是学术界与工程界面临的关键挑战;与此同时,复杂流动问题往往涉及高维空间的数学表达与数值求解,进一步加剧了建模的难度。PINN技术通过创新的架构设计与优化策略,突破了传统数值方法依赖网格划分的限制,能够在无网格框架下实现对非线性动力学特性的高效捕捉,并自然兼容高维空间的数学建模需求。该方法在流体力学问题的数值模拟与反问题求解中展现出优秀的适应性与可靠性,为复杂流动现象的理论研究与工程应用开辟了新的技术路径^[18]。Jin等^[19]开发了速度-压力(VP)和涡旋速度(VV)两种形式的NS-Fnets(Navier-Stokes flow nets),通过模拟二维稳态科瓦萨茨尼流等标准基准问题展示了其准确性和收敛性,系统研究了不同优化器,在湍流通道流动中用VP-NSFnet模拟了湍流雷诺数 Re_τ 约为1000的情况,借助动态权重策略等技术提高模拟精度,解决逆问题等传统CFD方法难以解决的问题,通过迁移学习减轻训练成本,对余差点添加进行了详细分析,为复杂流体力学问题模拟提供新方案。Rao等^[20]提出混合变量PINN方法,用于无测量数据下的低雷诺数可压缩层流模拟,该方法结合连续方程与物质法则,以流函数保证无旋性,提升了传统PINN的训练与求解精度。数值验证表明,相比传统方案,此方法对圆柱下游流场的速度和压力预测更准,适应性与鲁棒性佳。此外还探索了边界条件权重优化,并分析了其在高雷诺数湍流中的应用潜力与挑战。Sun等^[21]提出无数据的物理约束深度学习法用于流体动力学仿真。该方法利用深度神经网络近似N-S方程,实现无传统数据训练,确保预测符合物理定律。经圆管流动验证,对含不确定参数的流量预测准确,分析不同雷诺数下性能验证泛化能力;拓展至复杂血管流动仿真,能捕捉物理特性、预测不

确定性传播,为心血管疾病诊疗提供工具,且普适性与扩展性良好。Kharazmi等^[22]提出高阶变分物理信息网络(hp-VPINN)用于高效求解偏微分方程。该方法结合非线性函数逼近与域分解技术,以高阶多项式为测试函数优化投影残差,可处理复杂奇异解问题。研究阐述hp-VPINN理论框架,通过算例验证其灵活性,以泊松方程为例展示在复杂场景下的求解优势,并探讨网络设计与参数优化策略。相较于传统PINN,hp-VPINN在精度和泛化能力上表现更优。

本研究聚焦于圆柱绕流问题,运用PINN,融合数据驱动方法与流体物理约束条件,精准识别并重构流场涡旋结构,从而实现对圆柱绕流演化过程的高精度预测。本文介绍了PINN算法的原理,详细阐释其在偏微分方程数值求解、流场物理场动态预测以及方程待定系数确定的作用机制与可行性,同时探究神经网络参数对预测精度的影响规律,为进一步优化流场预测精度提供坚实的理论依据与数据支持。

1 PINN

神经网络(neural networks)是一类借鉴生物神经网络结构与运行机制构建的数学模型及计算架构,其核心功能在于实现函数的估计与逼近。如图1所示,一个简单的多层神经网络包含输入层、隐藏层和输出层,构建输出特征与输入特征的非线性函数关系,通过最小化网络预测值与参照值的残差来训练网络。

PINN是一种结合了深度学习和物理学知识的机器学习模型。PINN在模型训练过程中引入物理定律作为正则化约束,通过将偏微分方程形

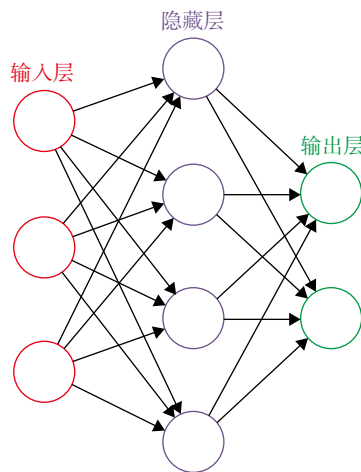


图1 简单神经网络结构

Fig. 1 Structure of simple network

式的物理规律嵌入损失函数构建机制,确保网络输出满足既定物理条件。该模型以深度神经网络为架构主体,通过在损失函数中显式引入物理信息正则项,将物理定律(通常以偏微分方程形式表征)作为约束条件。假设问题通过以下偏微分方程表示:

$$\frac{\partial U}{\partial t} + N(U) = 0 \quad x \in \Omega, t \in [0, t_{\max}] \quad (1)$$

$$U(x, t) = B(x, t) \quad x \in \partial\Omega, t \in [0, t_{\max}] \quad (2)$$

$$U(x, 0) = I(x) \quad x \in \Omega \quad (3)$$

其中 $U(x, t)$ 表示偏微分方程的待解函数, N 表示非线性微分算子, t 表示时间, $t_{\max}$ 表示时间范围最大值, Ω 表示空间坐标的定义域, $\partial\Omega$ 表示边界, $B(x, t)$ 表示边界条件, $I(x)$ 表示初始条件。

PINN 的结构如图 2 所示,主要由神经网络和损失函数组成。神经网络部分输入空间坐标 (x, y) 与时间坐标 (t) , 并输出预测值 u_θ (目标物理量, 包括速度 u 、 v 和压力 p), 完成优化训练之后, u_θ 已逼近偏微分方程的解, 可以近似为真值 u 。

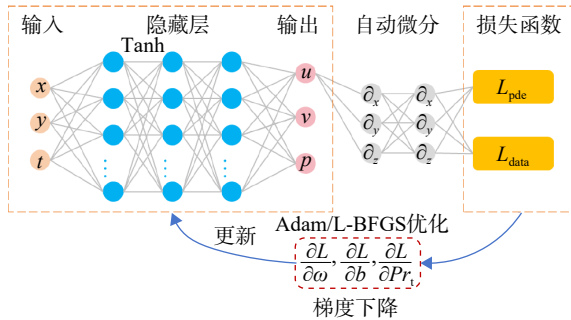


图 2 PINN 模型

Fig. 2 PINN model

损失函数部分融合了物理信息,通过最小化损失函数,能够使得网络的输出满足物理信息的结果^[23]。训练过程中,网络能够自动计算损失函数对神经元权重 ω 、偏差 b 和方程中参数(例如普朗特数 Pr)的偏导并进行优化。损失函数(L)由偏微分方程损失(L_{pde})、初始条件损失(L_{bc})、边界条件损失(L_{ic})和参照数据损失(L_{data}), 可以表示为

$$L = \omega_{pde} L_{pde} + \omega_{bc} L_{bc} + \omega_{ic} L_{ic} + \omega_{data} L_{data} \quad (4)$$

其中:

$$L_{pde} = \frac{1}{N_{pde}} \sum_{i=1}^{N_{pde}} \left| \frac{\partial u_\theta}{\partial t}(x_i, t_i) + N[u_\theta(x_i, t_i)] \right|^2 \quad (5)$$

$$L_{bc} = \frac{1}{N_{bc}} \sum_{i=1}^{N_{bc}} |u_\theta(x_i, t_i) - B(x_i, t_i)|^2 \quad (6)$$

$$L_{ic} = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} |u_\theta(x_i, t_i) - I(x_i)|^2 \quad (7)$$

$$L_{data} = \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} |u_\theta(x_i, t_i) - u(x_i, t_i)|^2 \quad (8)$$

其中 ω_{pde} 、 ω_{bc} 、 ω_{ic} 和 ω_{data} 为各项损失权重, N_{pde} 、 N_{bc} 、 N_{ic} 和 N_{data} 为各项残差点数。在训练过程中, L_{bc} 、 L_{ic} 和 L_{data} 保证网络预测解与边界条件、初始条件、参照数据对应采样点匹配, 而 L_{pde} 保证网络输出满足给定的物理信息。

PINN 既能求解正问题, 也就是通过训练使得网络输出逼近偏微分方程的解; 也能求解逆问题, 例如基于有限的参考数据推导模型中的未知参数数值或者源项等。通过反向传播, 最终确保所有域内的数据点都满足所表示的物理定律。Sirignano 等^[24]证明了当神经网络的各层神经元个数足够多时, PINN 能够以任意精度逼近偏微分方程的解。基于这一特性, 通过训练样本对所构建的 PINN 模型进行训练, 实现参数迭代优化, 可驱动模型输出向偏微分方程的精确解收敛, 最终实现对偏微分方程的高效数值求解与高精度建模。

损失函数中, 偏微分方程损失项使用自动微分或其他数值微分法计算微分方程在域中各点的残差。由于物理信息项并不计算预测值与参照值之间的误差, 因此可以将此项视为无监督损失项, 也就是说, 可以使用域中的任何点对网络进行训练, 即使没有在这些点上的测量值也是如此, 由于没有明确的标签, 无监督的损失函数能够更多反映数据本身的结构或特征。与仅从输入和输出数据中学习数学关系的纯粹数据驱动方法相比, PINN 使用先验物理知识, 能够在训练数据集之外做出更准确的预测, 在训练数据有限或含噪的情况下更有效。

2 PINN 的验证

求解一个非线性瞬态二维热传导问题, 几何模型为一个方形平板, 边长 $d = \pi$ m, 密度 $\rho = 1$ kg/m³, 比热容 $c = 1$ J/(kg·K), κ 为导热系数, 与温度 T 相关, $\kappa(T) = T^2 + T + 1$ 。

问题的控制方程为

$$\frac{\partial}{\partial x_i} \left[\kappa(T) \frac{\partial T}{\partial x_i} \right] = \rho c \frac{\partial T}{\partial t_T} \quad (9)$$

$$0 \leq x_i \leq d \quad 0 \leq t_T \leq 1$$

边界条件为

$$\begin{aligned} T(x_1, 0, t_T) &= T(x_2, d, t_T) = \\ T(0, y_1, t_T) &= T(d, y_2, t_T) = 0 \end{aligned} \quad (10)$$

初始条件为

$$T(x, y, 0) = \sin x \sin y \quad (11)$$

网络的输入为时空坐标 (x, y, t_T) , 其中 x 为横向坐标, y 为纵向坐标, t_T 为时间; 输出为输入坐标对应的温度场 T 。网络的主体结构为一个全连接神经网络(FCNN), 包含 4 个隐藏层, 每层的神经元数量均为 40, 激活函数选用 Tanh 函数。优化方法采用 Adam 优化器。

训练时, 在各个边界上均匀选取 100 个点, 共 400 个点, 在平板内随机选取 5 000 个点作为初始条件损失计算点, 在平板内随机选取 9 000 个点作为方程损失计算点。训练时间步长 $\Delta t_1 = 0.01$ s, 时间步数为 101。训练步数为 30 000 步。初始学习率 $L_r = 0.001$ 。训练收敛的条件为网络残差小于 10^{-8} 或者训练步数达到给定数值。

图 3 给出了 $t_T = 0.05$ s 时 PINN 计算结果与 COMSOL 计算结果的云图。

图 4、图 5 给出了 $t_T = 0.1, 0.5, 1.0$ s 时, $x = \pi/2$ m 处的计算结果对比及相对误差。相对误差(e)计算公式如下, 其中 f_{PINN} 为 PINN 预测值, f_{data} 为参照数据值, 此处为 COMSOL 计算值:

$$e = \left(\frac{f_{\text{PINN}} - f_{\text{COMSOL}}}{f_{\text{COMSOL}}} \right) \times 100\% \quad (12)$$

观察到, PINN 计算得到的温度场与数值仿真

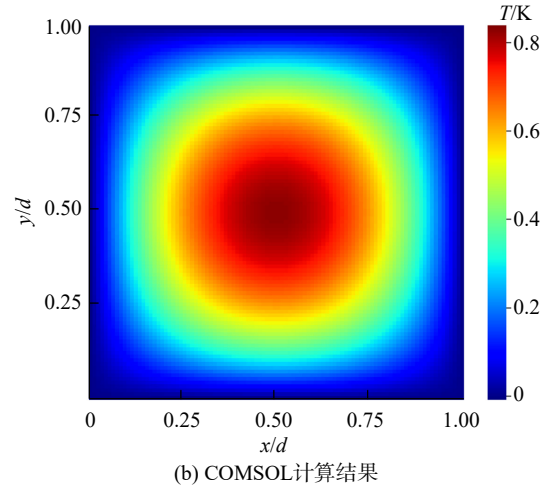
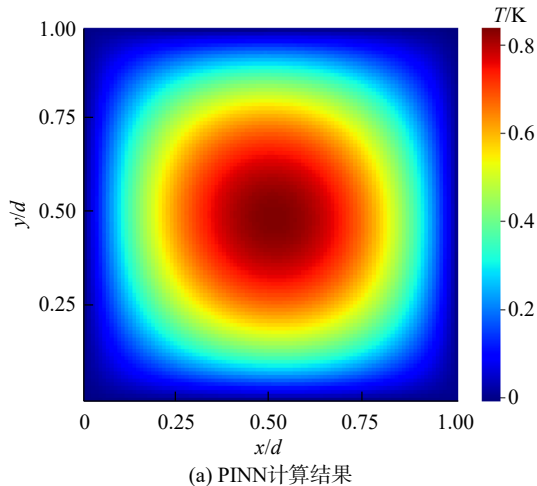
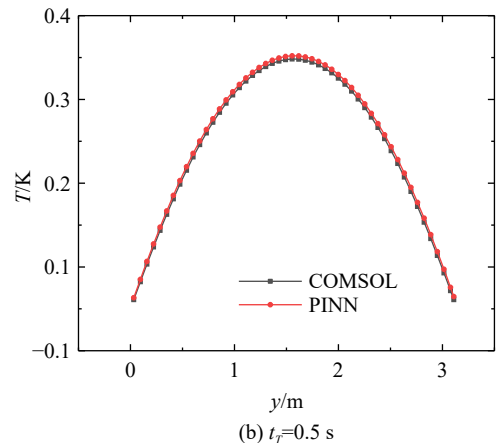
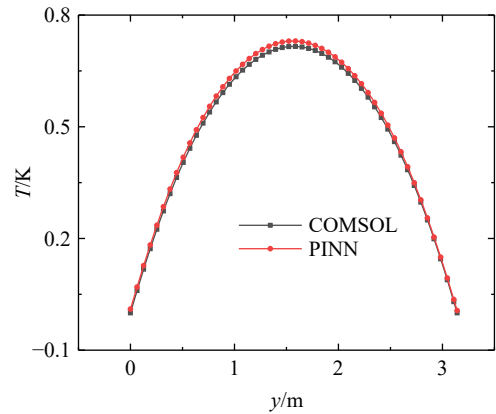
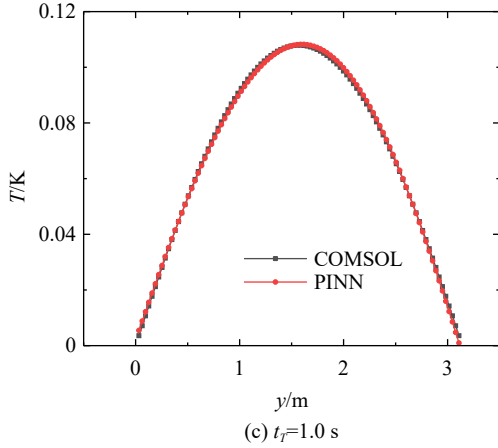
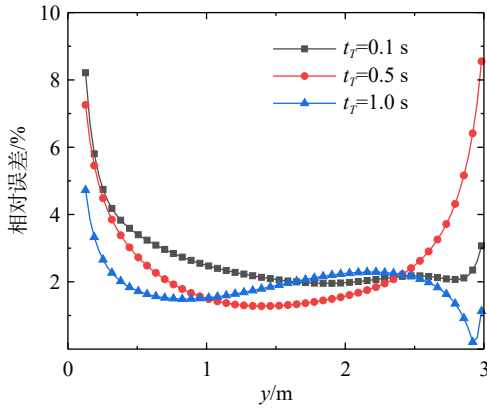


图 3 $t_T = 0.05$ s 时计算结果

Fig. 3 Calculation results at $t_T = 0.05$ s

的解较为吻合, 在不同时刻、不同位置均能得出较为准确的解; 不同时刻 $x = \pi/2$ m 处计算结果相对误差呈现两边大、中心小的趋势, 在中心区域相对误差均小于 5%。测试结果表明, PINN 构造了关于温度的先验信息, 并将物理定律和边界条件信息编码到神经网络中, 保证了计算结果的精度, 也保证了本研究设计的 PINN 结构的可行性。



图 4 不同时刻 $x=\pi/2$ m 处计算结果对比Fig. 4 Comparison of calculation results at $x=\pi/2$ m at different times图 5 不同时刻 $x=\pi/2$ m 处计算结果相对误差Fig. 5 Relative error of calculation results at $x=\pi/2$ m at different times

3 PINN 应用于圆柱绕流结构的流场预测

圆柱绕流是指流体在圆柱形物体表面发生的流动现象。当流体通过圆柱形物体时,物体表面会产生压力波动和速度波动。这些波动导致流体的旋涡出现,并伴随着能量的传递和损失。圆柱绕流是流体力学中经典的绕流问题之一,其研究揭示了流动分离、涡脱落等基础物理机制,成为理解复杂流动的基石。

$$L_{\text{pde}} = \frac{1}{N_{\text{pde}}} \left[\sum_{i=1}^{N_{\text{pde}}} |F_x(x, y, t)|^2 + \sum_{i=1}^{N_{\text{pde}}} |F_y(x, y, t)|^2 + \sum_{i=1}^{N_{\text{pde}}} |G(x, y, t)|^2 \right] \quad (19)$$

数据损失部分:

本文使用 Raissi 等^[13] 计算的不可压缩二维圆柱绕流直接数值模拟(direct numerical simulation, DNS)的结果作为训练和对照数据。计算条件为左边界添加均匀速度 $u_\infty = 1$ 的自由来流,二维圆柱绕流的雷诺数 $Re=100$,在距离圆柱中心 25 个直径的右边界添加零压力出口条件,在上下边界添加周期性边界条件;计算域为 $x \in [-15, 25]$, $y \in [-8, 8]$,计算时间步长 $\Delta t_2 = 0.1$ s,总时间步数为 100。考虑到计算机性能与训练需求,选取区域 $x \in [1, 8]$, $y \in [-2, 2]$ 的数据作为后续网络的训练数据。此时二维圆柱绕流会出现卡门涡街,即均匀来流条件下,圆柱后方的流场会出现周期性脱落、旋转方向相反、排列规则的双列线涡。

3.1 PINN 模型构建

针对上述问题构建 PINN 并进行求解。将二维不可压缩 N-S 方程与连续性方程作为控制方程加入神经网络的损失函数中。

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho_1} \cdot \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (13)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho_1} \cdot \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (14)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (15)$$

其中 u 、 v 分别为 x 、 y 方向的流动速度, ρ_1 为流体密度, p 为当地压强, ν 为流体的运动黏度。

根据控制方程构建函数。

$$F_x = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + \frac{1}{\rho_1} \cdot \frac{\partial p}{\partial x} - \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (16)$$

$$F_y = \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + \frac{1}{\rho_1} \cdot \frac{\partial p}{\partial y} - \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (17)$$

$$G = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (18)$$

利用上述函数构建损失函数。其中方程损失部分:

$$L_{\text{data}} = \frac{1}{N_{\text{data}}} \left[\sum_{i=1}^{N_{\text{data}}} |u_{\text{pred}}(x, y, t) - u_{\text{train}}(x, y, t)|^2 + \sum_{i=1}^{N_{\text{data}}} |v_{\text{pred}}(x, y, t) - v_{\text{train}}(x, y, t)|^2 + \sum_{i=1}^{N_{\text{data}}} |p_{\text{pred}}(x, y, t) - p_{\text{train}}(x, y, t)|^2 \right] \quad (20)$$

其中方程损失中的计算点为计算域内随机选取, 数据损失中的 $u_{\text{pred}}(x, y, t)$ 代表该处网络预测的结果, $u_{\text{train}}(x, y, t)$ 代表上述的直接数值模拟结果构成的训练集中对应点的数据, 因此计算点需要选取训练集中存在的坐标对应的数据点。考虑到数据点较为稠密, 且包含初始时间步的数据, 因此认为数据损失部分包含了边界与初始情况的信息, 不需要再额外添加边界条件与初始条件的损失函数。

因此总的损失函数为:

$$L = \omega_{\text{pde}} L_{\text{pde}} + \omega_{\text{data}} L_{\text{data}} \quad (21)$$

考虑到圆柱绕流问题中物理方程所具有的非线性、多尺度结构以及长时间演化特性, 其求解对神经网络的精度要求极高。多种网络模型的出现使得神经网络的发展进入了几十层规模的阶段, 而且网络的层数越深, 越有可能获得优异的泛化能力。然而, 传统的 FCNN 在构建深层网络以提升表达能力时, 普遍面临梯度消失/爆炸问题, 导致训练困难、收敛缓慢, 且难以捕捉流场中的复杂细节, 这已成为制约 PINN 在复杂流动问题上性能的关键瓶颈。

残差神经网络属于深度学习模型的一种, 如图 6 所示。其核心在于让网络的每一层不直接学习预期输出, 而是学习与输入之间的残差关系。这种结构通过添加“跳跃连接”, 即跳过某些网络层的连接来实现身份映射, 再与网络层的输出相加合并。残差使得拥有几十上百层的深度学习模型可以更易于训练, 增加模型深度时还能保持甚至提高准确度。在残差网络架构中, 多个残差块串联在一起。

一个简单的残差块结构如图 7 所示。相比

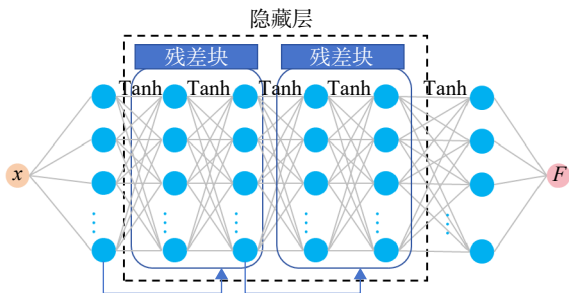


图 6 残差神经网络结构

Fig. 6 Structure of residual neural network

于 FCNN 的单个隐藏层, 残差块中增加了跳跃连接。假设一个包含若干层的残差块, 其中输入为 x , 原本线性层表达的输出为 $F(x)$, 在此基础上添加 x , 也就是将整个残差块的输出表达为 $E(x) = F(x) + x$ 。

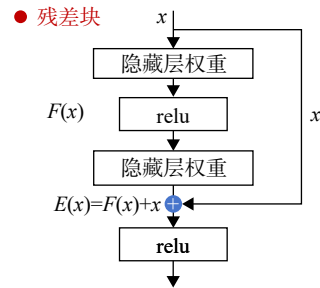


图 7 单个残差块模型

Fig. 7 Model of single residual block

除此之外, 残差网络并没有增加新的参数, 计算量相对而言也没有增加。通过残差结构, 网络允许梯度直接通过恒等映射回传, 避免因链式求导导致梯度衰减, 反向传播总能传输有效的梯度数值, 使得神经网络能够在层数较深时准确率不会因为网络深度增加而降低。

为了提升网络在深层结构中的性能, 本研究创新性地将残差神经网络^[25](residual neural network, ResNet)的核心思想深度融入 PINN 的架构设计中, 构建了融合残差的物理信息神经网络 (Res-PINN)。

传统基于 FCNN 的 PINN 因其链式求导机制, 在反向传播计算高阶物理约束 (如 N-S 方程中的 2 阶导数) 时, 梯度极易衰减或爆炸。本研究引入的残差跳跃连接构造了恒等映射路径, 确保了误差梯度能够直接、有效地在训练过程中传播。这不仅从根本上缓解了梯度消失问题, 使构建更深、表达能力更强的网络成为可能, 还大幅提升了训练的稳定性与收敛效率, 为成功求解复杂流场奠定基础; 此外, 残差模型的计算公式 $E(x) = F(x) + x$ 与物理问题的求解思维存在内在契合。 x 作为网络的初始、粗略的流场输出在网络中传播, 而残差函数 $F(x)$ 则代表了网络需要学习的、由物理效应所引起的修正量。因此, Res-PINN 不仅仅是一个黑箱模型, 其架构本身就在隐式地引导网络学习

物理规律的增量修正,在一定程度上增强了模型的可解释性,并使其在数据稀缺区域表现出更好的物理一致性和泛化能力。

基于以上思想,本文构建的 Res-PINN 结构如图 8 所示。残差神经网络包含输入层、输出层与

4 个残差块,输入时空坐标 (x, y, t) ,输出目标物理量 (u, v, p) ,每个残差块包含 2 个隐藏层与 2 个激活函数交错排列,串联组成残差神经网络,每个隐藏层包含 64 个神经元,激活函数均选取 Tanh 函数。

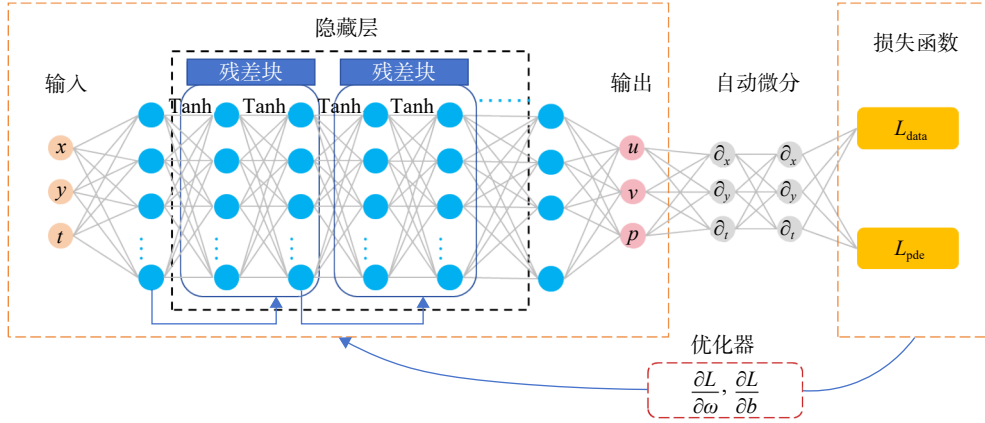


图 8 Res-PINN 模型

Fig. 8 Res-PINN model

训练之前,对各层网络中神经元权重 ω 和偏差 b 进行 Kaiming 初始化,减少可能存在的梯度消失或梯度爆炸问题。训练过程中,网络优化选用 Adam(adaptive moment estimation)优化器和 L-BFGS(limited-memory BFGS)优化器,在训练前期,先使用 Adam 优化器训练 40 000 步,并采用动态缩小的学习率,初始学习率为 0.001,每训练 800 步学习率变为之前的 80%,能够使得大规模数据高效快速收敛,后期再通过 L-BFGS 优化器更加精细地调整参数,预防过拟合,提高模型性能。

考虑到网络的输入、输出范围,参数存在量纲差异性,不同条件参数之间的数量级相差较大,为加快网络收敛速度,消除尺度效应,因此网络的输入、输出均为标准化后数据。将 x, y 坐标均匀缩小至 $[-1, 1]$,作为网络的输入,并在后续计算偏导时重新乘上放缩系数:

$$x_{nn} = \frac{x - x_{mid}}{x_{max} - x_{min}} \times 2 \quad (22)$$

$$\frac{\partial F}{\partial x} = \frac{\partial F}{\partial x_{nn}} \cdot \frac{\partial x_{nn}}{\partial x} \quad (23)$$

式中下标 max、min、mid 分别表示最大值、最小值、最大最小值的平均值。

同时网络的输出范围也受到激活函数的限制,范围为 $[-1, 1]$,需要根据数据集的极值将输出还原到真实值,再代入损失函数方程进行计算:

$$F = \frac{F_{nn}(F_{max} - F_{min})}{2} + F_{mid} \quad (24)$$

3.2 Res-PINN 在流场重构的应用

实验测量技术虽能提供高精度的流体力学数据,但受限于测量设备与实验条件,通常仅能获取有限区域或离散点的流场样本。在此背景下,为完整解析全域流场结构并开展深层次流动特性研究,流场重构技术成为不可或缺的研究手段。PINN 依托深度学习框架,融合数据驱动与物理约束双重优势,通过将流体力学控制方程(如 N-S 方程)以正则化形式嵌入神经网络损失函数,建立流场变量与时空坐标间的高阶映射关系,为复杂流场的高精度重构提供了理论完备且计算高效的解决方案。

考虑到计算机性能,从总计算结果中选取 3 个均匀时间步数据形成数据集,输入总数据量达到了 45 000;再随机采样 50% 的数据集作为训练集,用于计算数据残差部分。在训练时,随机选取计算域内 2 500 个点用于计算方程残差;按照训练集的数据点坐标生成 2 500 个点用于计算数据残差。

训练的残差曲线如图 9 所示。从图中可知,网络的收敛速度较快,在前半部分经历一些小幅波动,整体仍在平稳下降,最终趋于平稳;数据残差项与方程残差项的变化趋势类似,数值大小接近,这表示网络训练结果同时受到了数据集与物

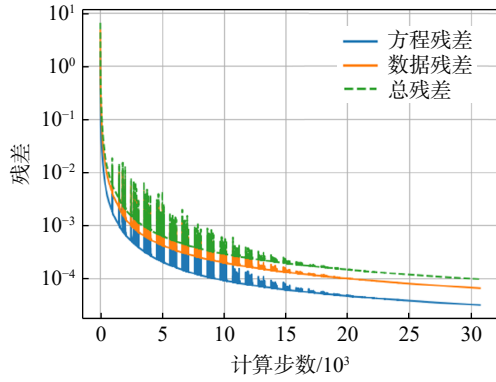


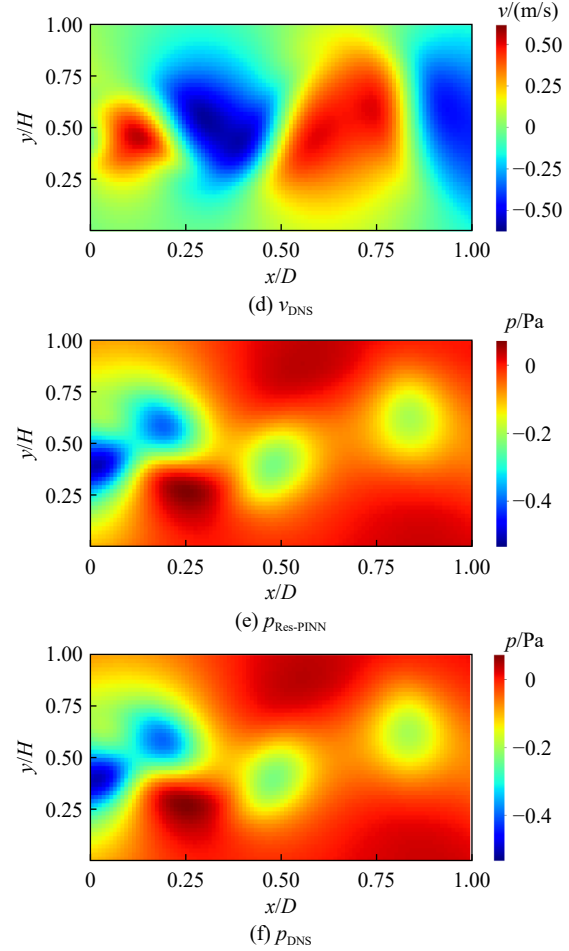
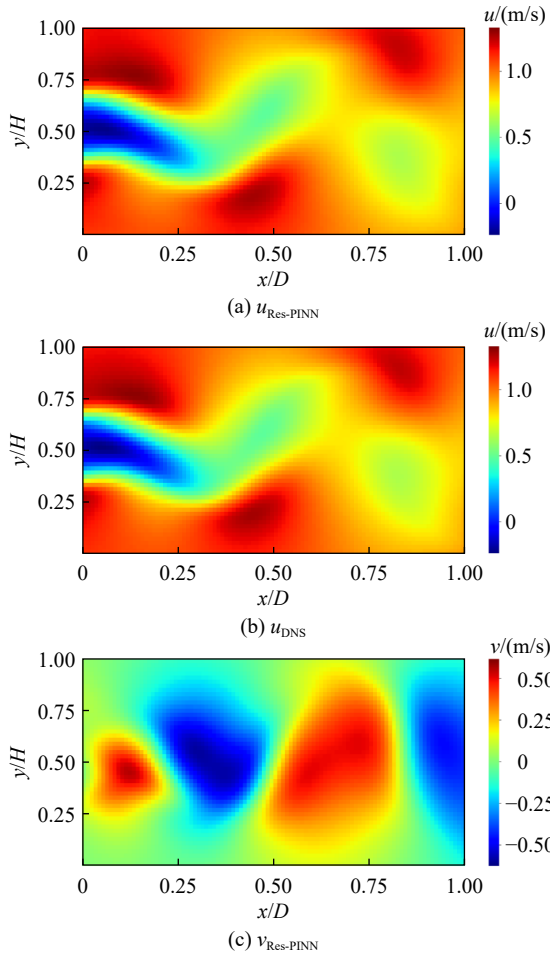
图 9 Res-PINN 训练残差

Fig. 9 Training residuals of Res-PINN

理方程信息的约束影响。

为了验证 Res-PINN 在流场重构中的效果, 使用训练完成的网络进行流场重构, 输入计算区域平面内坐标网格与时刻, 使得网络输出此时刻区域内的速度场与压力场。得到 $t=0$ s 时刻、计算区域平面的 Res-PINN 重构的目标流场, 包括 x 方向流动速度 u 、 y 方向流动速度 v 、压力 p , 并与 DNS 对照数据进行对比。其中 D 为计算域横向长度, H 为计算域纵向长度。

从图 10 及数据分析可以得出, 在不同时刻,

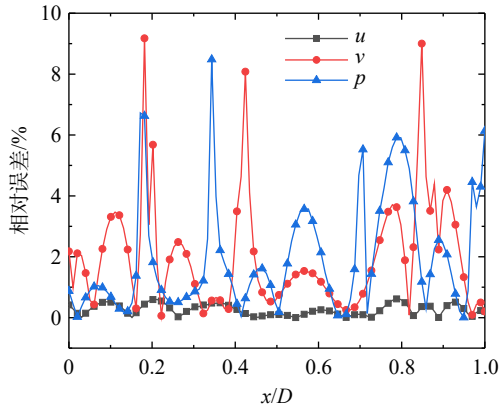
图 10 Res-PINN 与对照数据的计算结果对比 ($t=0$ s)Fig. 10 Comparison of calculation results between Res-PINN and reference data ($t=0$ s)

Res-PINN 的流场重构结果与 DNS 结果十分接近; 在未参与训练的时刻, Res-PINN 同样能够实现该时刻流场的预测, 证明添加了物理信息的神经网络学习到了流场及压力场的特征, 能够实现缺失数据的补全, 具有较好的重构能力。

选取 $t=0$ s 时刻平面内 $y/H=0.2, 0.8$ 处 Res-PINN 的预测结果与 DNS 结果对比, 计算相对误差, 如图 11 所示。

能够观察到, 在流场中的不同位置, u 、 v 、 p 的重构结果与 DNS 对照数据的相对误差均小于 10%, 其中速度 u 拟合得最好, 相对误差均小于 1%, 速度 v 和压力 p 在流向上存在个别点相对误差较大, 余下的大部分位置均小于 5%。其中速度 v 作为脉动速度具有随机性, 并且 v 和 p 的数量级较小, 因此预测的相对误差较大一些。

总的来说, 上述内容验证了 Res-PINN 重构流场的能力, Res-PINN 不仅能够根据先验物理信息学习到流场的流动特征, 同时结合数据训练拟合

图 11 Res-PINN 重构结果的相对误差 ($t=0$ s, $y/H=0.2$)Fig. 11 Relative error of Res-PINN reconstruction results ($t=0$ s, $y/H=0.2$)

的精度也较高。

Res-PINN 的流场重构与预测能力具有实际应用价值,能够利用实验的少部分精准流场数据,通过训练网络使得能够还原较为完整且精度较高的流场物理场数据,具有一定的工程意义。

3.3 Res-PINN 在参数反演的应用

模型方程的参数反演是众多工程问题的难点

之一。利用神经网络的逼近能力,可以进行基于标签数据的数据拟合。数据驱动下的参数反演可无须考虑物理过程。然而,将数据驱动与物理信息相结合,可能会有更好的反演效果。从另一个角度,将先验物理信息加入预测模型,能够根据容易获得的数据预测更难获得的数据。因此,不同于传统做法,Res-PINN 具备结合样本数据反推方程属性的能力,可以大幅降低标签数据的数据量,在参数重建领域发挥巨大的作用。

针对本研究问题,如图 12 所示,构建 Res-PINN 模型来实现参数反演。设置 N-S 约束方程初始系数 λ_1 、 λ_2 为可训练参数,在训练的反向传播过程中与网络的其他参数一样进行优化。

在训练 Res-PINN 前给予 λ_1 、 λ_2 不同的初始值,验证训练之后的 λ_1 、 λ_2 是否接近真实值,如果 λ_1 、 λ_2 最终的收敛值接近于真实的 λ_1 、 λ_2 ,则说明 Res-PINN 方法学习到了 N-S 方程的约束信息,并且具备参数矫正和辨识能力。给定不同的初始值进行训练,参数 λ_1 、 λ_2 的收敛曲线如图 13 所示。表 1 给出了最终收敛值和对应的相对误差。

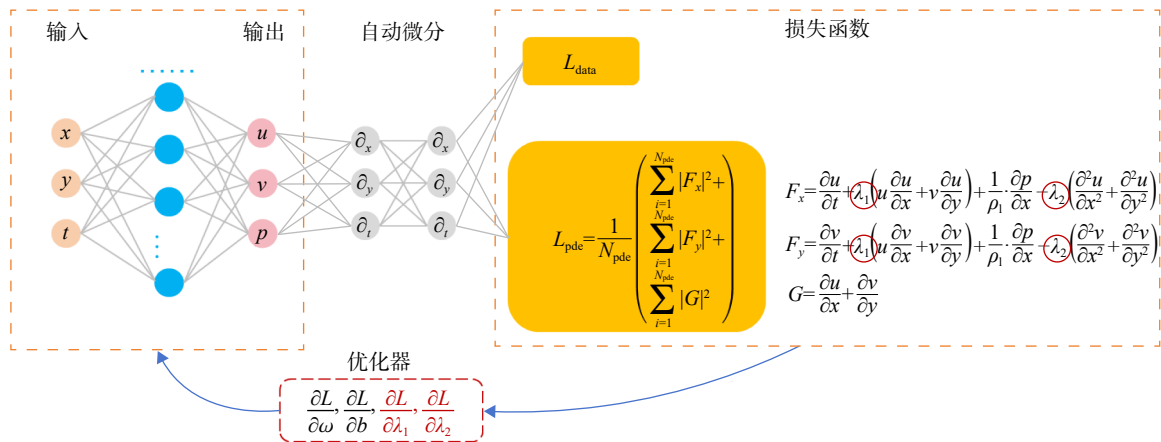
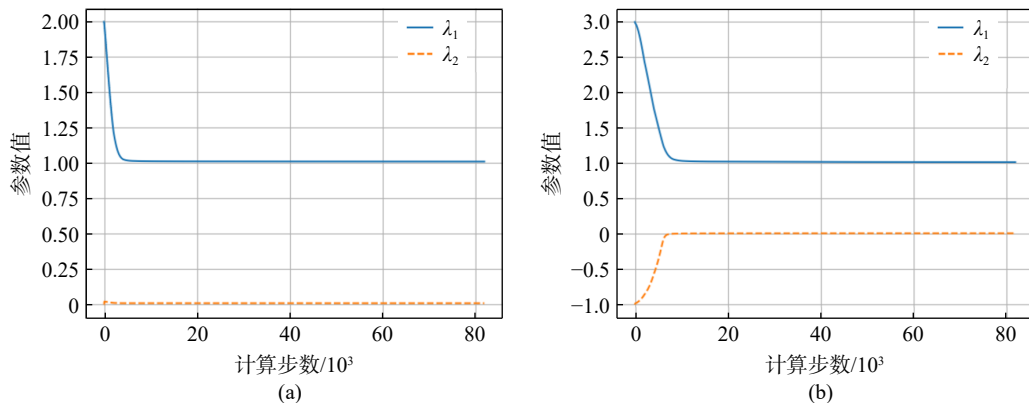


图 12 用于参数反演的 Res-PINN 模型

Fig. 12 Res-PINN model for parameter inversion



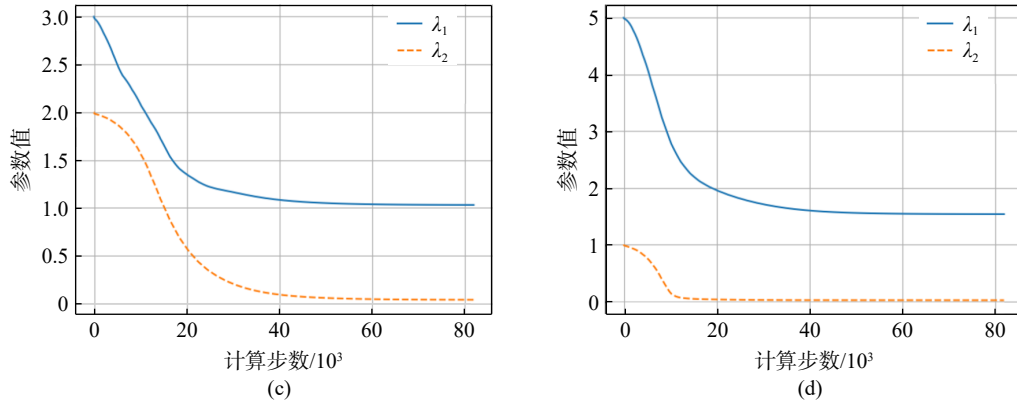


图 13 不同初始值方程参数随训练步数的变化

Fig. 13 Variation of equation parameters of different initial values with training steps

表 1 不同初始值方程参数最终收敛值及其误差

Table 1 Final convergence value and error of equation parameters with different initial values

参数	初始值	收敛值	误差/%
λ_1	2	1.012 4	1.24
	3	1.016 6	1.66
	3	1.029 7	2.97
	5	1.535 9	53.59
λ_2	0	0.010 2	2.00
	-1	0.009 8	2.00
	2	0.012 4	24.00
	1	0.010 7	7.00

观察发现,不同初始值下的 λ_1 、 λ_2 在训练过程中最终能够达到一个全局最优值,并且接近于真实值;参数的初始值偏离真实值越远,最终的收敛值误差也就会相应的偏大。在前两种收敛的较好的情况中,反演参数的相对误差均小于 5%,表明 Res-PINN 反演的参数数值存在较好的可信度。对于工程应用的方程中缺少精确数值的参数,能够利用类似的方法,结合 Res-PINN 与部分高保真数据,修正参数使其更为匹配高精度流场,提高工程计算的精度。

3.4 网络性能对比测试

本部分研究针对网络结构、激活函数、无监督训练集数量、网络层数、神经元数量、损失函数权重对网络预测效果的影响进行了测试,探究了如何构建网络能够达到最优的性能。

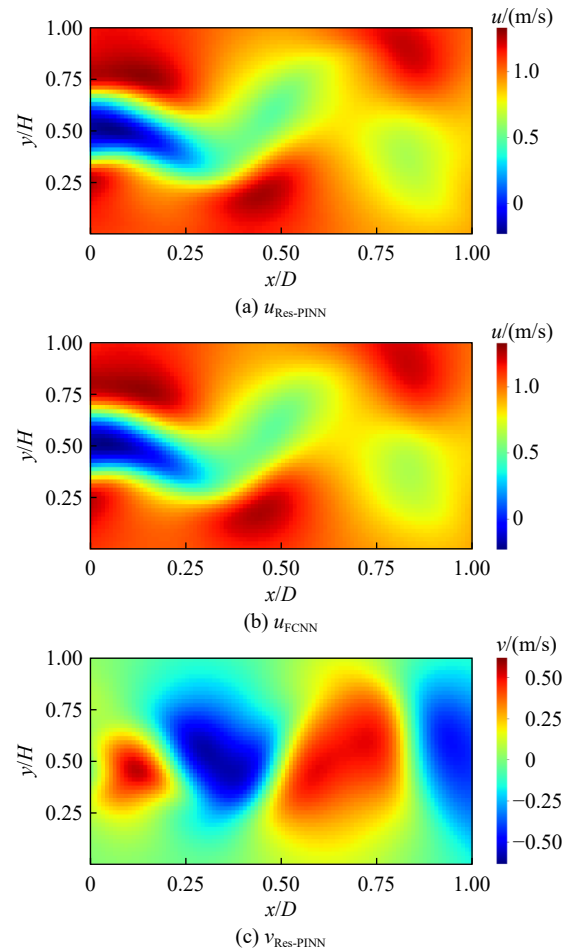
3.4.1 网络结构

本研究构建了 Res-PINN。为了验证所构建的网络效果,使用本研究构建的网络与基于 FCNN 的 PINN 分别对文中第 3.2 节的问题进行计算并对比性能。除了基础的网络架构分别为 Res-

Net 与 FCNN 外,两类 PINN 的其他部分(损失函数、优化器、训练集)完全一致。两种网络的激活函数均为 Tanh,其中每个隐藏层的神经元数量均相同。

图 14 给出了时刻 2($t=0.2$ s)两类 PINN 的流场计算结果。表 2 给出了不同物理量的计算结果全场平均相对误差。

观察到两类 PINN 均能够实现基础的功能,



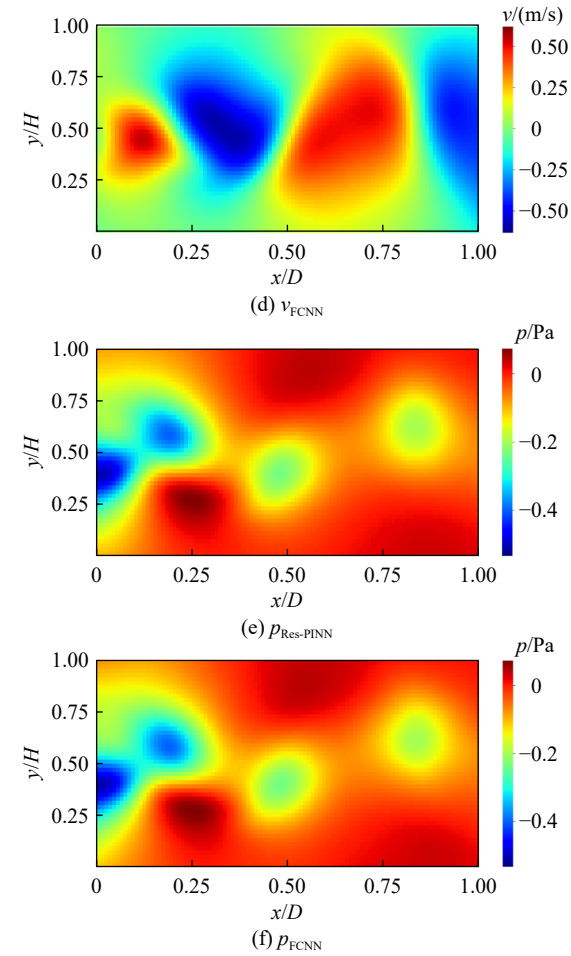


图 14 两类 PINN 的流场计算结果对比($t=0.2\text{ s}$)
Fig. 14 Comparison of flow field calculation results of two kinds of PINN ($t=0.2\text{ s}$)

表 2 两类 PINN 计算结果全场平均相对误差
Table 2 Average relative error of two kinds of PINN calculation results in the whole field

模型种类	平均相对误差/%		
	u	v	p
FCNN	1.222 56	5.421 28	4.919 65
Res-PINN	0.535 89	3.032 91	2.598 81

即流场的拟合与预测,但 Res-PINN 在预测精度上实现了提升,相比于传统的基于 FCNN 的 PINN,对速度 u 、速度 v 、压力 p 的预测误差下降了约 50%,预测精度实现了显著提升。

图 15 和图 16 给出了此时刻 $y/H=0.2, 0.5$ 处的计算结果对比。图 17 和图 18 给出了对应的相对误差。

观察到,在流场的不同位置,两类 PINN 的预测结果均与参照 DNS 数据较为接近,预测准确性较好;基于 FCNN 的 PINN 的预测误差在流场空间中波动剧烈,特别是在计算域进口、出口及流

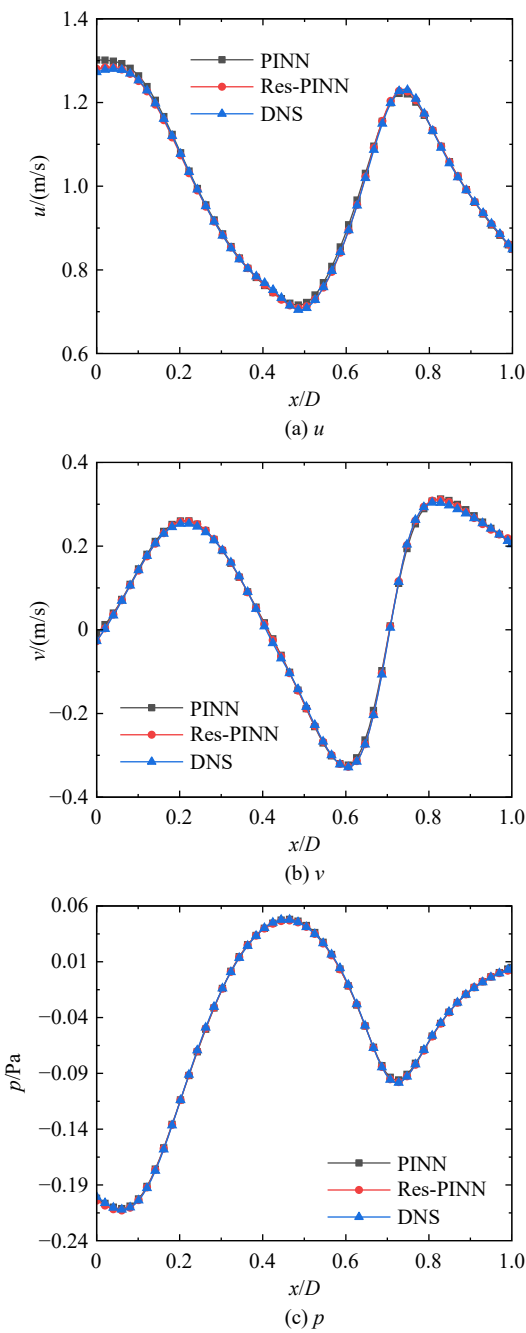
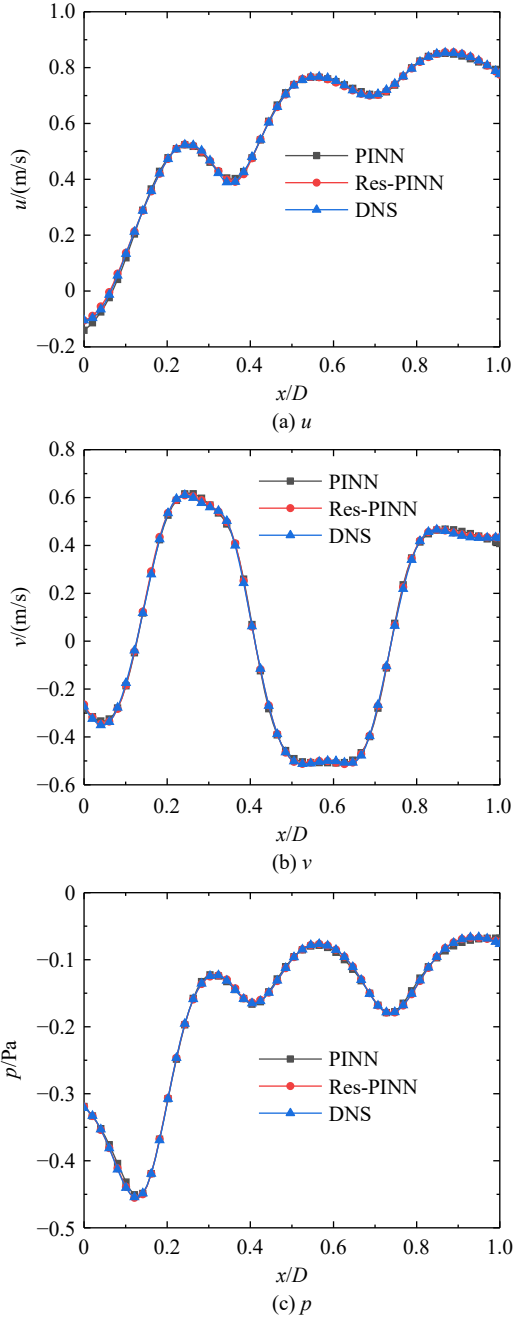


图 15 $y/H=0.2$ 处计算结果对比($t=0.2\text{ s}$)
Fig. 15 Comparison of calculation results at $y/H=0.2$ ($t=0.2\text{ s}$)

场中部等区域出现了明显的误差峰值,这表明其预测存在不稳定性;Res-PINN 则表现出卓越的稳定性与准确性。其误差曲线整体更为平缓,量值显著降低,且成功抑制了 FCNN 中出现的多数误差峰值,将各物理量的最大相对误差基本控制在 5% 以下。

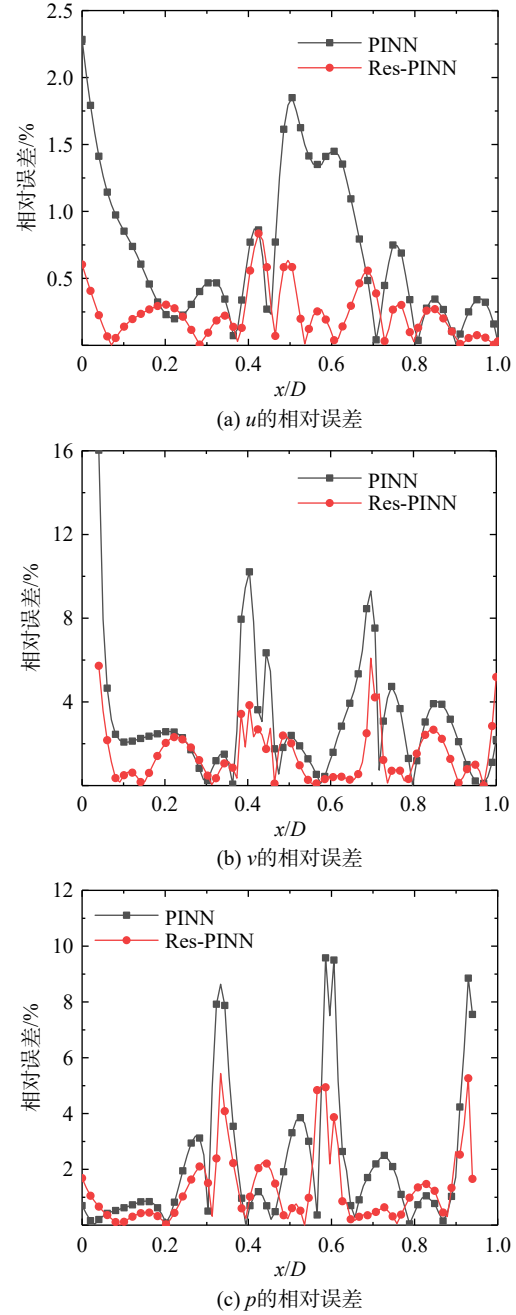
本研究构建的融合残差结构的 PINN 通过引入残差神经网络的核心思想,利用跳跃连接有效缓解了深层网络中的梯度消失问题,促进了梯度的有效传播,从而显著增强了模型的优化能力和

图 16 $y/H=0.5$ 处计算结果对比($t=0.2$ s)Fig. 16 Comparison of calculation results at $y/H=0.5$ ($t=0.2$ s)

表达能力。对比结果证明, 本研究的设计不仅全面提升了解场预测的精度, 更极大地增强了解的平滑性与稳定性, 为解决复杂流动问题提供了一个更可靠、更高效的深度学习框架。

3.4.2 激活函数

激活函数是神经网络中重要的一部分, 它直接决定了网络的表达能力、学习效率和最终性能。激活函数的核心作用是为神经网络引入非线性变换, 这使得神经网络能够逼近任何复杂函数, 实现特征转换。本部分研究使用结合不同激活函数

图 17 $y/H=0.2$ 处计算结果相对误差对比($t=0.2$ s)Fig. 17 Comparison of relative error of calculation results at $y/H=0.2$ ($t=0.2$ s)

的神经网络, 对比其计算结果以及误差, 结合激活函数本身分析其性能影响与具体的物理意义。分别使用了 Tanh、ReLU、Sigmoid 这 3 种激活函数。表 3 给出了不同层数网络的计算结果全场平均相对误差。图 19 给出了此时刻 $y/H=0.5$ 处的计算结果对比。图 20 给出了对应的相对误差。

观察到, 不同激活函数导致了显著的性能差异。其中使用 Tanh 函数的网络得到了最低的全场预测平均误差, 远低于其他两种函数, 展现了

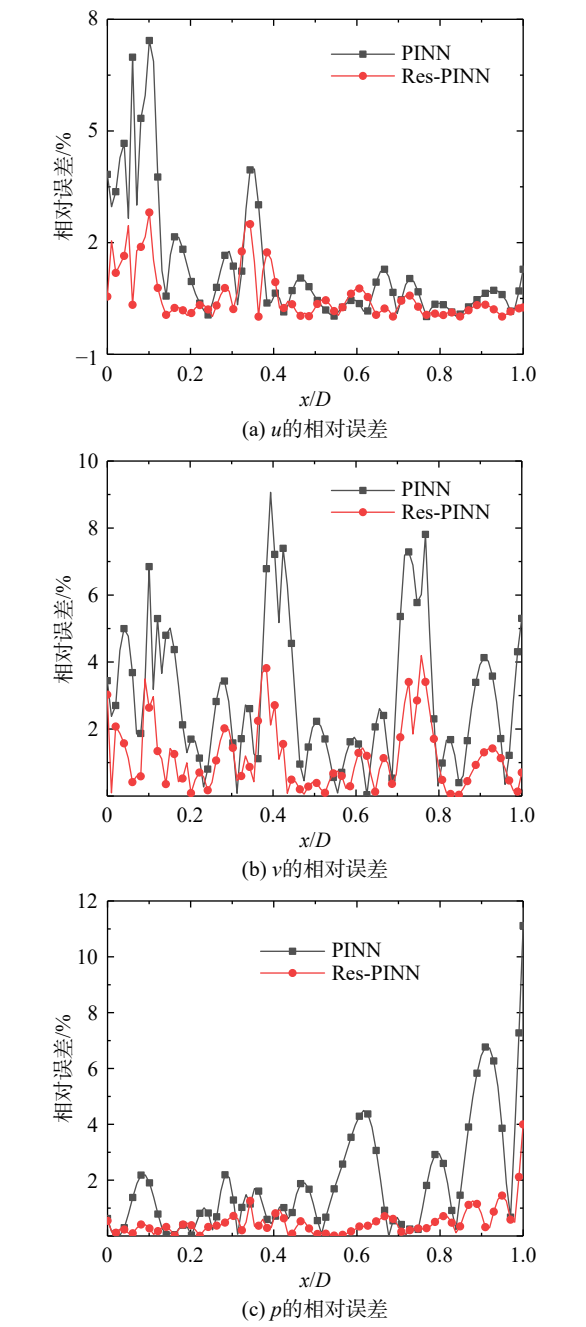


图 18 $y/H=0.5$ 处计算结果相对误差对比($t=0.2\text{ s}$)

Fig. 18 Comparison of relative error of calculation results at $y/H=0.5$ ($t=0.2\text{ s}$)

激活函数	平均相对误差/%		
	u	v	p
Tanh	0.535 89	3.032 91	2.598 81
ReLU	1.276 66	4.631 03	5.810 88
Sigmoid	1.316 49	4.594 53	4.447 88

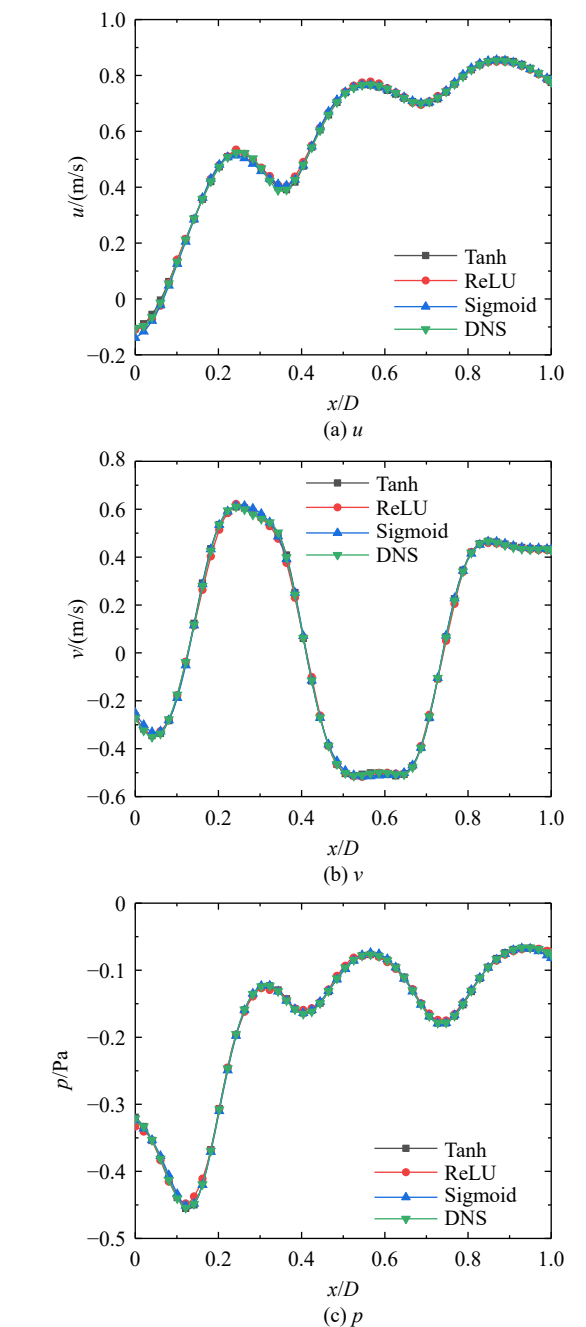


图 19 $y/H=0.5$ 处不同激活函数计算结果对比($t=0.2\text{ s}$)

Fig. 19 Comparison of calculation results for different activation functions at $y/H=0.5$ ($t=0.2\text{ s}$)

较好的预测精度和稳定性;使用 ReLU 和 Sigmoid 函数的网络表现不太理想,误差波动剧烈且峰值较高。结果表明,激活函数的选择不仅关乎数值稳定性,更影响着物理规律能否被正确嵌入和学习。Tanh 函数作为平滑且任意阶可微函数,非常适合计算并传递损失函数中的高阶偏导数项,同时 Tanh 函数的输出是以 0 为中心的,其梯度在 0 附近最大,有助于优化器更高效地更新权重,加速收敛并找到更优的解。而 ReLU 函数存在 2

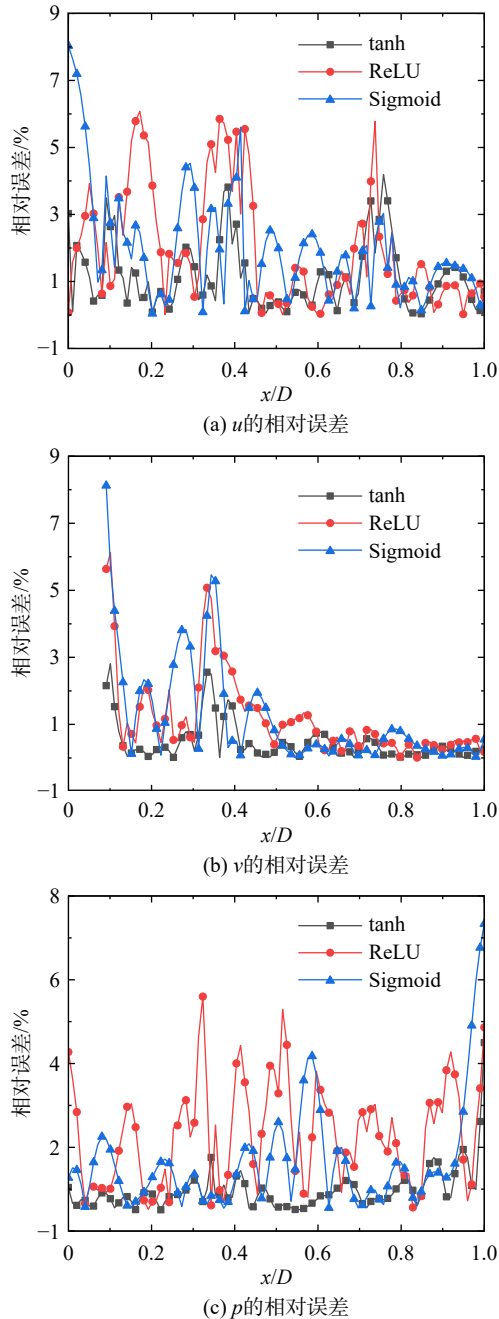


图 20 $y/H=0.5$ 处不同激活函数计算结果相对误差对比
($t=0.2$ s)

Fig. 20 Comparison of relative errors in calculation results for different activation functions at $y/H=0.5$ ($t=0.2$ s)

阶梯度消失问题, 导致物理约束(PDE 损失)无法被正确学习, 同时函数本身输出也导致一旦神经元输出为负, 其梯度将恒为 0, 会导致网络表达能力下降, 难以捕捉流场中的复杂特征。Sigmoid 函数则是两端的饱和区梯度接近于 0, 极易导致梯度消失问题, 在深度网络中, 会使得梯度值逐层急剧衰减, 最终导致网络底层权重几乎无法更新。因此尽管 ReLU 函数在传统图像深度学习中

广泛应用, 但其特性与 PINN 的原理存在冲突, 完全不适用。经过对比, 选择平滑、可微的 Tanh 函数作为网络隐藏层的激活函数, 保证了网络的性能可靠。

3.4.3 无监督训练集数量

本部分研究训练集的数量对网络计算结果的影响。在文中第 3.4.2 节的研究基础上, 本部分使用融合残差的 PINN, 网络共有 8 个隐藏层、4 个残差块, 损失函数权重均为 1。完整的数据集为文中第 3.2 节中包含 3 个时间步的数据集, 在此基础上分别随机选取完整数据集的 70%、50%、20%、5% 作为神经网络的训练集进行训练, 训练步数均为 30 000, 对比分析训练效果。

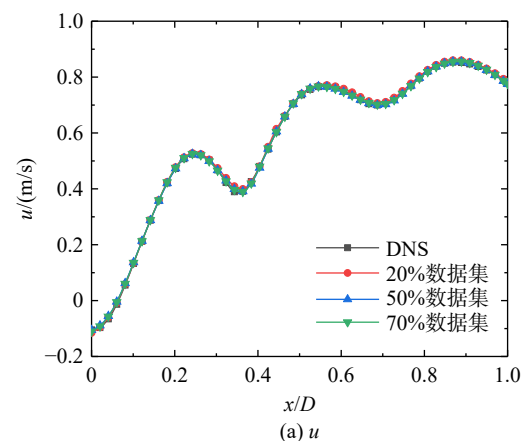
表 4 给出了不同训练集的计算结果全场平均相对误差。图 21 给出了此时刻 $y/H=0.5$ 处的计算结果对比。图 22 给出了对应的相对误差。

结果表明, 与预期的结果符合, 训练集的数量显著地影响了训练效果, 随着训练集的增大, 计算结果的误差呈现下降的趋势, 其中速度 u 误差变化最小, 表明该物理量学习难度最小, 压力 p 误差变化极大, 对数据量最为敏感。在 5% 数据集时误差急剧恶化, 尤其压力 p 误差达 23.31%, 说明此时模型已严重欠拟合, 过于稀缺的数据点不足以覆盖解空间, 物理场的关键特征可能未被采样。

表 4 不同训练集的网络计算结果全场平均相对误差

Table 4 Average relative error of the network calculation results in the whole field with different training sets

训练集	平均相对误差/%		
	u	v	p
70% 数据集	0.522 76	2.495 47	2.686 07
50% 数据集	0.535 49	3.037 86	2.582 73
20% 数据集	1.020 41	5.902 85	3.113 27
5% 数据集	1.388 52	12.060 42	23.311 84



(a) u

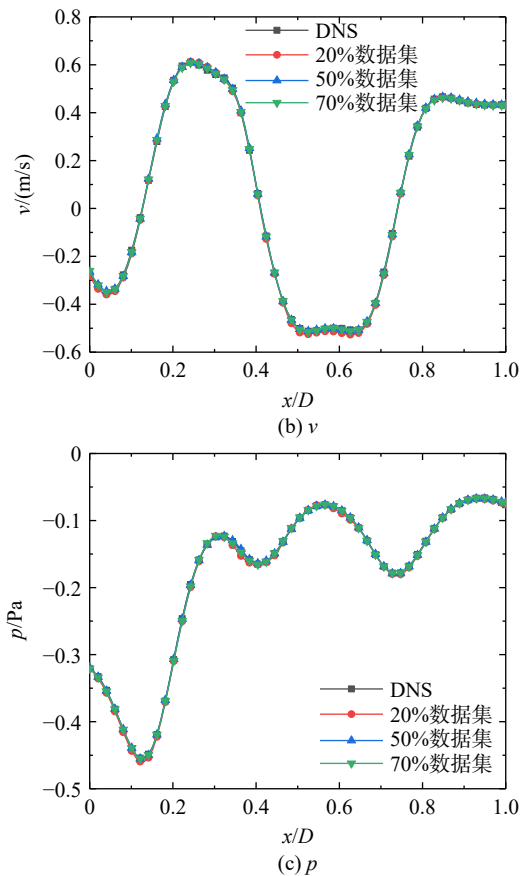


图 21 $y/H=0.5$ 处不同训练集计算结果对比($t=0.2\text{ s}$)

Fig. 21 Comparison of calculation results of different training sets at $y/H=0.5$ ($t=0.2\text{ s}$)

当训练集大于等于 50% 时,网络的训练效果已基本满足目标,此时再增大训练集的数量,效果并不是很显著,压力的全场平均相对误差由 2.58% 变为 2.69%,出现了负优化。这表明想要实现训练效果,需要一定的数据量,但并不能简单地通过增大训练集数量来提升训练效果,而是应该针对数据敏感的物理量进行加强。这也验证了 PINN 根据部分数据实现全流场重构的功能,应用时需要根据实际存有的数据量与计算机性能选择合适的训练集数量。后续研究均选取 50% 数据集作为训练集。

3.4.4 网络层数

一般来说,神经网络的隐藏层数越深,其拟合能力越强,结果也越精细。然而网络层数过多时,可能会导致过拟合。本部分研究神经网络的层数对网络计算结果的影响,探究本问题适合的网络结构。分别使用 net_4(4 个隐藏层、2 个残差块)、net_8(8 个隐藏层、4 个残差块)、net_16(16 个隐藏层、4 个残差块)进行训练计算,损失函数权重均为 1。表 5 给出了不同层数网络的计算结果全

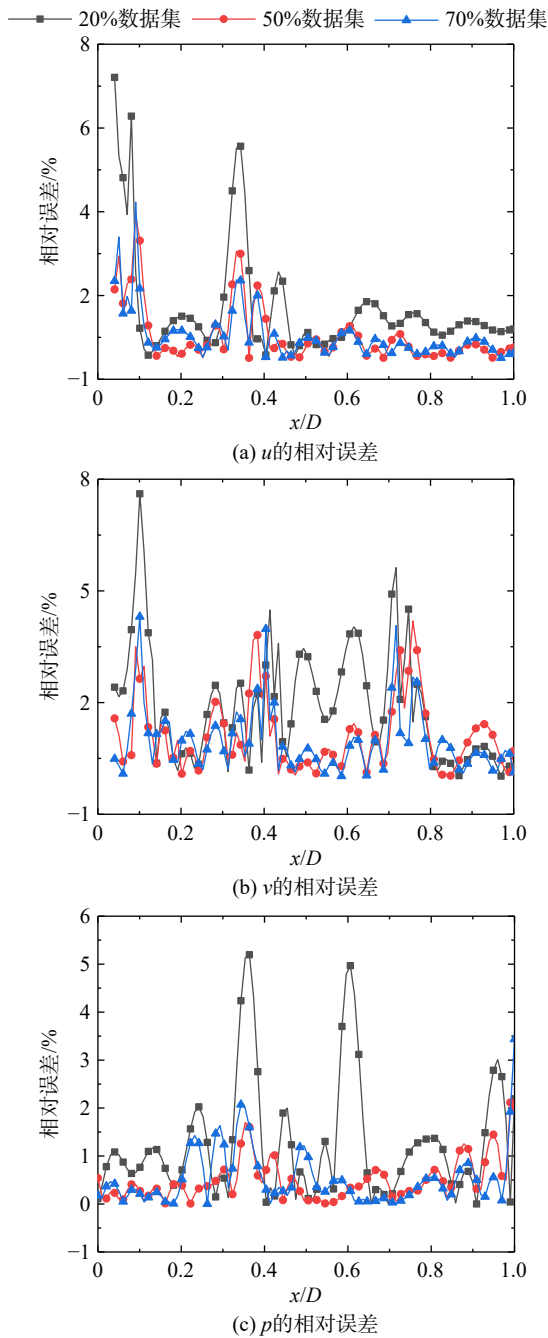


图 22 $y/H=0.5$ 处不同训练集计算结果相对误差对比($t=0.2\text{ s}$)

Fig. 22 Comparison of relative error of calculation results of different training sets at $y/H=0.5$ ($t=0.2\text{ s}$)

表 5 不同结构的网络计算结果全场平均相对误差

Table 5 Average relative error of the calculation results of networks with different structures in the whole field

网络结构	平均相对误差/%		
	u	v	p
Net_4	0.815 08	3.325 81	3.414 53
Net_8	0.535 49	3.037 86	2.582 73
Net_16	0.582 97	3.011 36	3.148 37

场平均相对误差。图 23 给出了此时刻 $y/H=0.5$ 处的计算结果对比。图 24 给出了对应的相对误差。

计算结果表明,随着网络层数的增加,整体的计算损失逐渐降低,计算精度提升;然而在网络隐藏层数增加到 16 时,压力 p 的计算相对误差反而有所上升,相较 8 个隐藏层的网络效果变差,并且在流道中段出现了剧烈的波动与较大的误差峰值,这表明过深的网络结构会对 PINN 的部分性能造成负面的影响。此时速度 u 和速度 v 的计算

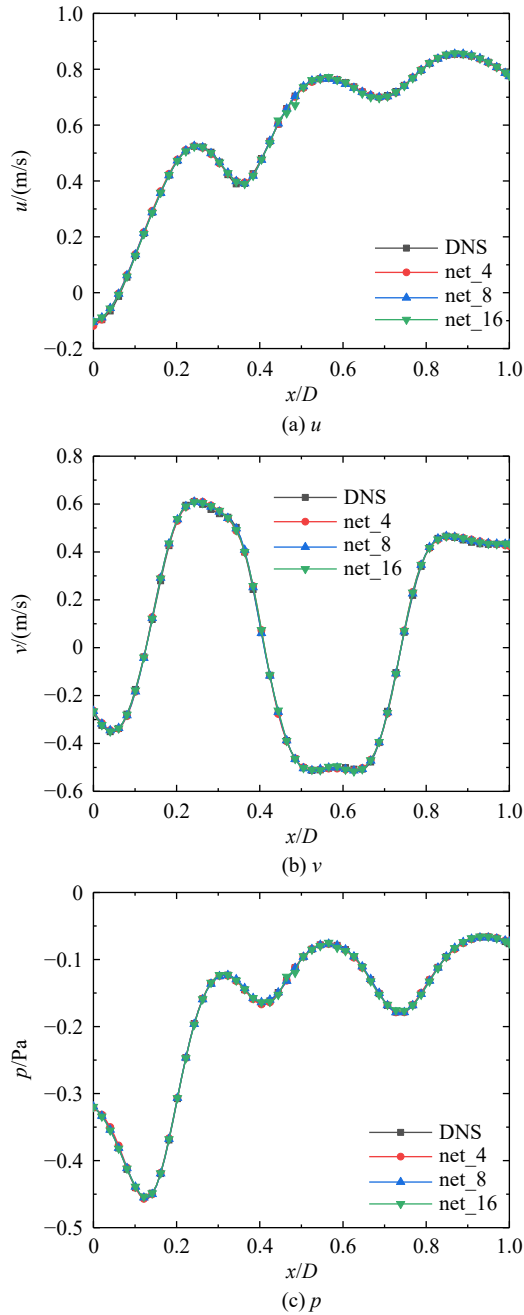


图 23 $y/H=0.5$ 处不同网络计算结果对比($t=0.2$ s)

Fig. 23 Comparison of calculation results of different networks at $y/H=0.5$ ($t=0.2$ s)

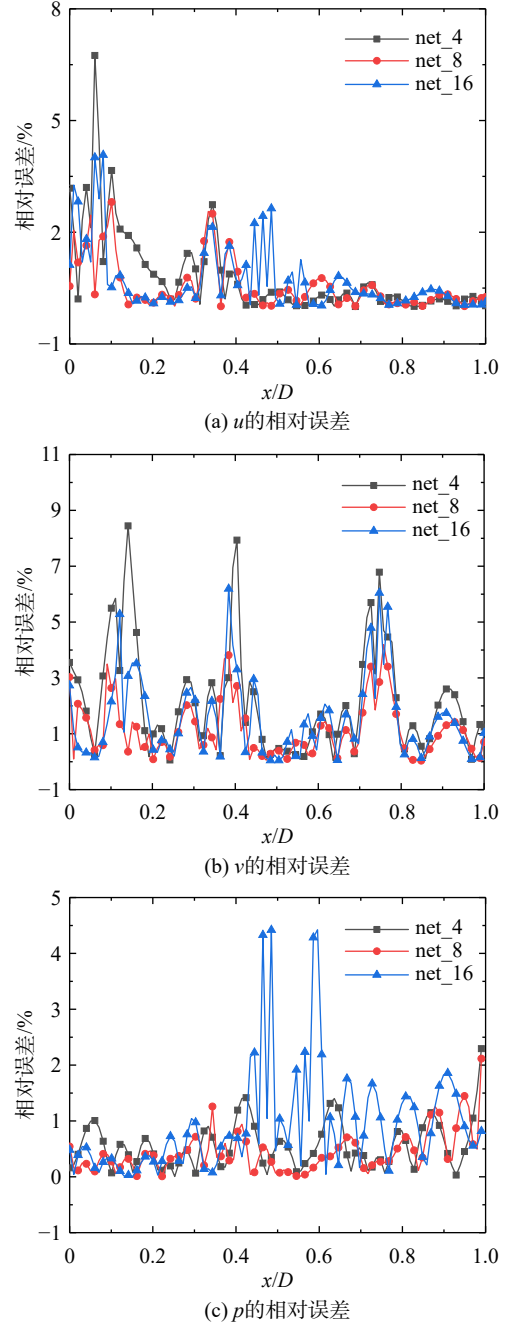


图 24 $y/H=0.5$ 处不同网络计算结果相对误差对比($t=0.2$ s)

Fig. 24 Comparison of relative error of calculation results of different networks at $y/H=0.5$ ($t=0.2$ s)

相对误差与 8 个隐藏层的网络计算结果接近,速度 u 的全场平均相对误差分别为 0.53% 与 0.58%,速度 v 的全场平均相对误差分别为 3.04% 与 3.01%,表示残差结构对深层神经网络性能产生了一定的正向效果,使得深层网络的部分性能不会随着网络深度增加而降低。因此,在构建神经网络时,需要根据具体问题权衡隐藏层数与网络稳定性,并通过一些优化设计增强深层神经网络的性能。

3.4.5 神经元数量

除了隐藏层层数之外,每层隐藏层中的神经元数量(N_{in})也是神经网络架构中关键的超参数之一,它直接影响着模型的表达能力、学习能力和泛化能力。网络性能并非与神经元数量有着简单的线性关系,而需要在一个平衡中寻找最优解。在前文研究的基础上,本研究分别选择每层 32、64、128、256 个神经元的神经网络进行训练,分析训练效果。表 6 给出了不同神经元数量的计算结果全场平均相对误差。图 25 给出了此时刻 $y/H=0.5$ 处的计算结果对比。图 26 给出了对应的相对误差。

计算结果表明,随着每层神经元数量的增加,

表 6 不同神经元数量的网络计算结果全场平均相对误差
Table 6 Average relative error of the networks calculation results in the whole field with different numbers of neurons

每层神经元数	平均相对误差/%		
	u	v	p
32	0.873 78	3.782 47	4.465 73
64	0.535 89	3.032 91	2.598 81
128	0.444 02	2.680 27	2.026 15
256	0.301 40	2.636 78	2.244 59

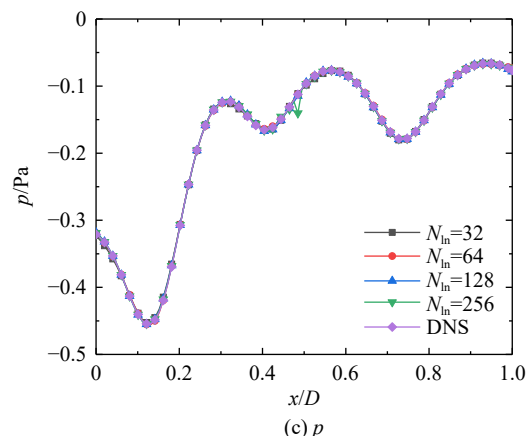
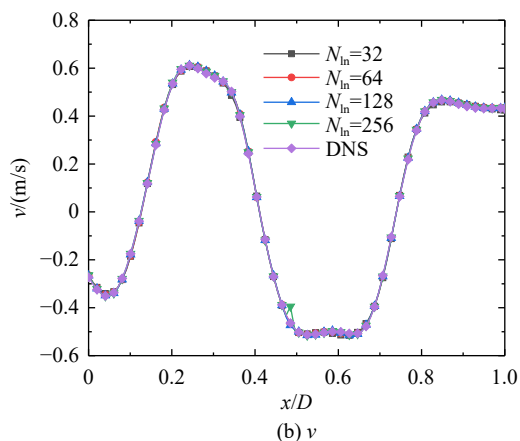
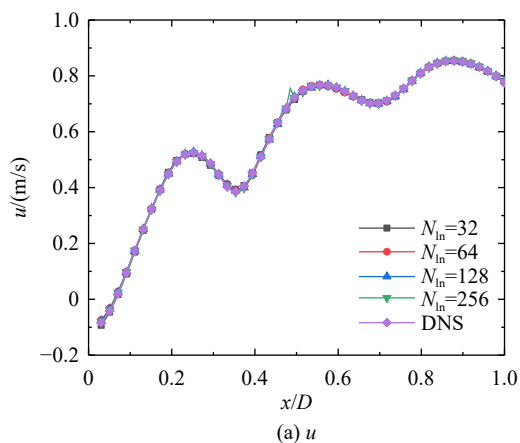


图 25 $y/H=0.5$ 处不同网络计算结果对比($t=0.2$ s)

Fig. 25 Comparison of calculation results of different networks at $y/H=0.5$ ($t=0.2$ s)

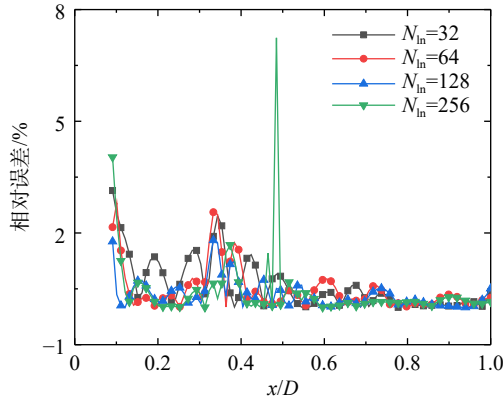
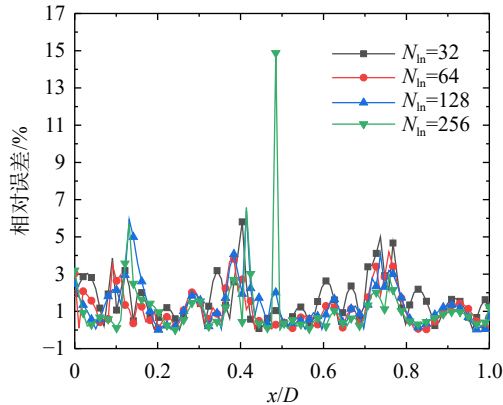
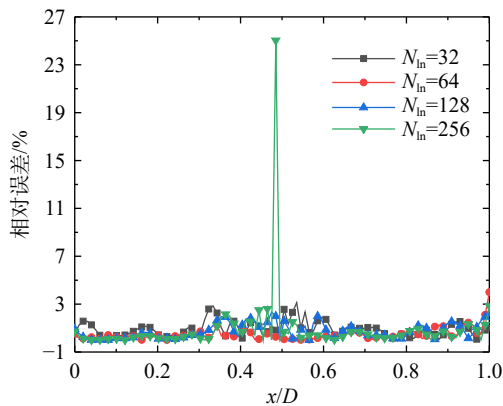
网络的计算误差出现了显著的降低,其中速度 u 和压力 p 的计算误差下降幅度超过 50%;但是随着每层神经元数量从 128 增加到 256,速度 v 的误差变化较小,压力 p 的误差甚至出现反弹,表明对于压力场求解,256 神经元的网络可能已逼近或达到过拟合的临界点;从相对误差对比图中也能观察到,随着神经元数量增加,连续位置的误差也会变得更加平稳,但是 256 神经元的网络在流场中段出现一个误差峰值,对于每个物理量的计算误差都远高于流场其他位置,表明在这个点可能出现过拟合,导致了局部恶化,未能在该处实现准确的流场预测。此外考虑到神经元增加所产生的资源消耗增大,神经元数量翻倍会导致网络计算所需显存与计算时间大幅增加,其中前两个网络在 NVIDIA Quadro P1000 4 GB 上训练,训练耗时在 1 h 之内;后两个网络由于消耗内存过大,选择在 NVIDIA RTX 5080 16 GB 上训练,耗时均超过了 2 h。因此,综合考虑网络性能与计算效率,本研究使用每层 64 神经元的网络,训练效果基本满足需求,能够准确拟合流场,同时能有较高的训练效率。

3.4.6 损失函数权重

本部分研究损失函数权重对网络计算结果的影响,使用融合残差的 PINN,选取有 8 个隐藏层、4 个残差块的神经网络。损失函数为:

$$L = \omega_{pde} L_{pde} + \omega_{data} L_{data} \quad (25)$$

在之前的研究中, $\omega_{pde} = \omega_{data} = 1$;在这部分中,通过改变方程项损失的权重 ω_{pde} ,训练网络并对比训练效果,探究 PINN 中物理方程损失项的权

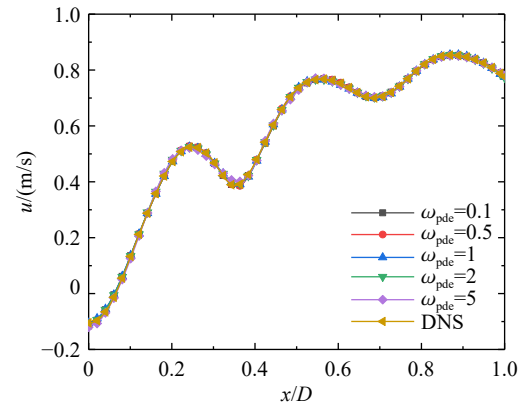
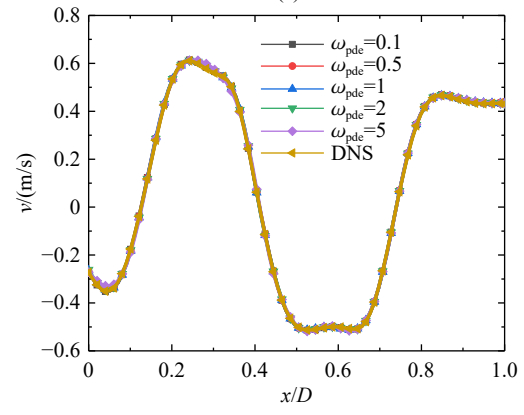
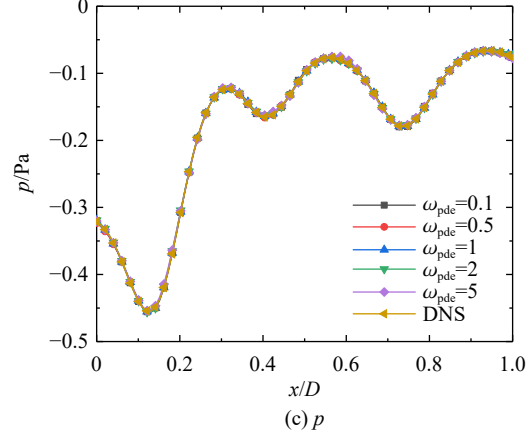
(a) u 的相对误差(b) v 的相对误差(c) p 的相对误差图 26 $y/H=0.5$ 处不同网络计算结果相对误差对比
($t=0.2$ s)Fig. 26 Comparison of relative error of calculation results of different networks at $y/H=0.5$ ($t=0.2$ s)

重对 PINN 性能的影响。表 7 给出了不同损失 ω_{pde} 的计算结果全场平均相对误差。图 27 给出了此时刻 $y/H=0.5$ 处的计算结果对比。图 28 给出了对应的相对误差。

结果显示, 当权重 ω_{pde} 增大时, 网络的计算误差反而出现了线性的增加, 精度有一定程度的下降; 当权重 ω_{pde} 减小时, 计算误差出现了下降, 但是在 ω_{pde} 减小到 0.1 时, 误差出现了小幅增加, 但

表 7 不同损失函数权重的网络计算结果全场平均相对误差
Table 7 Average relative error of the networks calculation results in the whole field with different the weights of loss functions

ω_{pde}	平均相对误差/%		
	u	v	p
0.1	0.426 87	2.411 45	2.239 12
0.5	0.364 08	2.772 76	2.159 01
1	0.535 49	3.037 86	2.582 73
2	0.619 21	3.235 86	3.262 56
5	0.997 28	4.178 62	3.484 49

(a) u (b) v (c) p 图 27 $y/H=0.5$ 处不同权重计算结果对比($t=0.2$ s)Fig. 27 Comparison of calculation results of different weights at $y/H=0.5$ ($t=0.2$ s)

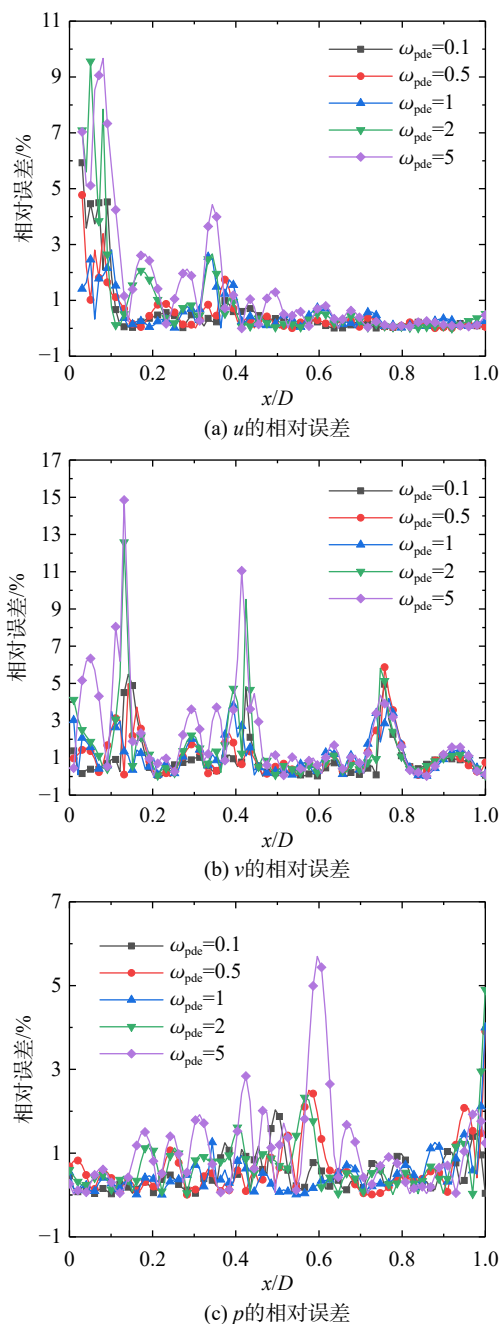


图 28 $y/H=0.5$ 处不同权重计算结果相对误差对比
($t=0.2\text{ s}$)

Fig. 28 Comparison of relative error of calculation results of different weights at $y/H=0.5$ ($t=0.2\text{ s}$)

仍低于 $\omega_{\text{pde}}=1$ 的情况。其中 $\omega_{\text{pde}}=2, 5$ 的情况下,整体的相对误差与峰值均较大,并且出现相比其他情况较为剧烈的波动。这表明方程损失项的权重如果过大,可能会使网络过于关注物理约束而忽视数据拟合,导致计算结果容易偏离参照数据,网络稳定性不足;方程损失项的权重能够通过适当减小,使得网络在满足方程的前提下更好地拟合数据,提升网络的精度。

4 结 论

本研究通过 PINN 开展流动特性研究,创新性构建融合残差的物理信息神经网络(Res-PINN),结合数据驱动方法,考虑流体的物理约束,针对圆柱绕流问题进行了研究。通过系统对比网络结构、激活函数、训练集规模、网络层数、神经元数量、损失函数权重等关键参数对模型性能的影响,深入探究网络优化路径,得到以下主要结论:

1) 流场重构应用验证。通过在损失函数中添加物理方程约束,结合部分 DNS 训练数据集,可高效实现流场的精准重构。经过计算结果的统计分析与对比验证,充分验证了该预测网络的可靠性与有效性。

2) 参数反演能力验证。结合不可压缩圆柱绕流的控制方程和 DNS 数据训练集,通过融合残差的 PINN 训练,能够准确求解方程中的待定系数。这一结果验证了该方法可以应用于模型参数反演场景。

3) 网络性能优势。与传统 PINN 相比,本研究构造的融合残差的 PINN 在预测误差上实现显著降低,同时有效提升计算精度与计算稳定性,尤其改善了网络在深层结构下的性能表现,解决了传统网络深层训练易出现的性能衰减问题。

4) 超参数影响规律。研究发现,激活函数的选择需匹配问题类型与网络特性,对网络性能起着关键作用;随着训练集数量的增加,网络的计算误差减小,预测精度提升,但同时也伴随着计算成本的增加;适当增加神经元数量、网络层数与减小方程损失权重能够提升网络的预测精度,但是存在一定的阈值,超过之后反而会实现负优化,影响网络稳定性。

5) 应用潜力展望。本研究构建的融合残差的 PINN 成功实现了 N-S 方程的求解与流体问题的计算,为复杂流体场景的数值模拟提供了有效工具,具备推广应用于涡轮叶片冷却结构扰流柱通道等工程场景的潜力。

综上,本研究构建的融合残差 PINN 已成功实现圆柱绕流的流场重构与参数反演,并通过对测试明确了网络性能优化措施,但研究仍存在改进空间:当前网络超参数优化均基于单参数独立变化的测试场景,而实际应用中如何在多参数耦合条件下实现超参数的平衡优化,是提升网络性能的关键问题。未来研究将进一步探索自适应

参数调节等更全面智能的网络优化方法, 以进一步提升 PINN 在复杂流体问题中的性能表现。

参考文献:

- [1] 张伟伟, 王旭, 寇家庆. 面向流体力学的多范式融合研究展望[J]. *力学进展*, 2023, 53(2): 433-467.
ZHANG Weiwei, WANG Xu, KOU Jiaqing. Prospects of multi-paradigm fusion methods for fluid mechanics research[J]. *Advances in Mechanics*, 2023, 53(2): 433-467. (in Chinese)
- [2] MOLLICONE J P, BATTISTA F, GUALTIERI P, et al. Effect of geometry and Reynolds number on the turbulent separated flow behind a bulge in a channel[J]. *Journal of Fluid Mechanics*, 2017, 823: 100-133.
- [3] 张伟伟, 朱林阳, 刘溢浪, 等. 机器学习在湍流模型构建中的应用进展[J]. *空气动力学学报*, 2019, 37(3): 444-454.
ZHANG Weiwei, ZHU Linyang, LIU Yilang, et al. Progresses in the application of machine learning in turbulence modeling[J]. *Acta Aerodynamica Sinica*, 2019, 37(3): 444-454. (in Chinese)
- [4] BALDWIN B, LOMAX H. Thin-layer approximation and algebraic model for separated turbulent flows[R]. AIAA-1978-0257, 1978.
- [5] LAUNDER B E, SPALDING D B. The numerical computation of turbulent flows[M]//*Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion*. Amsterdam: Elsevier, 1983: 96-116.
- [6] WILCOX D C. Reassessment of the scale-determining equation for advanced turbulence models[J]. *AIAA Journal*, 1988, 26(11): 1299-1310.
- [7] SPALART P, ALLMARAS S. A one-equation turbulence model for aerodynamic flows[R]. AIAA-1992-0439, 1992.
- [8] BISHOP C M. Neural networks and their applications[J]. *Review of Scientific Instruments*, 1994, 65(6): 1803-1832.
- [9] 苑光耀, 王俊淞, 赵玄烈, 等. 基于 PINN 的二维剪切流圆柱绕流场重构[J]. *力学学报*, 2025, 57(2): 436-452.
YUAN Guangyao, WANG Junsong, ZHAO Xuanlie, et al. Reconstruction of the flow field around a cylinder in a two-dimensional shear flow based on pinn[J]. *Chinese Journal of Theoretical and Applied Mechanics*, 2025, 57(2): 436-452. (in Chinese)
- [10] WU Jinlong, YIN Xiaolong, XIAO Heng. Seeing permeability from images: fast prediction with convolutional neural networks[J]. *Science Bulletin*, 2018, 63(18): 1215-1222.
- [11] XU Hao, CHANG Haibin, ZHANG Dongxiao. DL-PDE: deep-learning based data-driven discovery of partial differential equations from discrete and noisy data[J]. *Communications in Computational Physics*, 2025, 29(3): 698-728.
- [12] KARNIADAKIS G E, KEVREKIDIS I G, LU Lu, et al. Physics-informed machine learning[J]. *Nature Reviews Physics*, 2021, 3(6): 422-440.
- [13] 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.
- [14] BAYDIN A G, PEARLMUTTER B A, RADUL A A, et al. Automatic differentiation in machine learning: a survey[J]. *Journal of Machine Learning Research*, 2017, 18(1): 5595-5637.
- [15] AVOLEDO E, TOGNAN A, SALVATI E. Quantification of uncertainty in a defect-based physics-informed neural network for fatigue evaluation and insights on influencing factors[J]. *Engineering Fracture Mechanics*, 2023, 292: 109595.
- [16] HE Yichuan, WANG Zhicheng, XIANG Hui, et al. An artificial viscosity augmented physics-informed neural network for incompressible flow[J]. *Applied Mathematics and Mechanics (English Edition)*, 2023, 44(7): 1101-1110.
- [17] STRELOW E L, GERISCH A, LANG J, et al. Physics informed neural networks: a case study for gas transport problems[J]. *Journal of Computational Physics*, 2023, 481: 112041.
- [18] 刘肖廷, 闵建, 于相楠, 等. 物理信息神经网络的应用与研究进展[J]. *河南科学*, 2024, 42(7): 945-959.
LIU Xiaoting, MIN Jian, YU Xiangnan, et al. Applications and advancements of physics-informed neural networks: an overview[J]. *Henan Science*, 2024, 42(7): 945-959. (in Chinese)
- [19] JIN Xiaowei, CAI Shengze, LI Hui, et al. NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations[J]. *Journal of Computational Physics*, 2021, 426: 109951.
- [20] RAO Chengping, SUN Hao, LIU Yang. Physics-informed deep learning for incompressible laminar flows[J]. *Theoretical and Applied Mechanics Letters*, 2020, 10(3): 207-212.
- [21] SUN Luning, GAO Han, PAN Shaowu, et al. Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data[J]. *Computer Methods in Applied Mechanics and Engineering*, 2020, 361: 112732.
- [22] KHARAZMI E, ZHANG Zhongqiang, KARNIADAKIS G E M. hp-VPINNs: variational physics-informed neural networks with domain decomposition[J]. *Computer Methods in Applied Mechanics and Engineering*, 2021, 374: 113547.
- [23] 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.
- [24] SIRIGNANO J, SPILIOPOULOS K. DGM: a deep learning algorithm for solving partial differential equations[J]. *Journal of Computational Physics*, 2018, 375: 1339-1364.
- [25] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, US: IEEE, 2016: 770-778.

(编辑: 陈 越)