



同濟大學

TONGJI UNIVERSITY

硕士学位论文

(专业学位)

基于物理驱动深度学习的涡激振动 问题研究

上海市科技重大专项-人工智能基础理论与关键核心技术
(2021SHZDZX0100) 和中央高校基本科研业务费专项资金
资助

姓 名：杨 虎

学 号：1932456

所在院系：土木工程学院

学科门类：工 程

学科专业：建筑与土木工程

指导老师：唐和生

二〇二二年六月



同濟大學
TONGJI UNIVERSITY

A dissertation submitted to
Tongji University in conformity with the requirements for
the degree of Master of Engineering

Research on Vortex-induced Vibration Based on Physics-driven Deep Learning

Supported by the Shanghai Municipal Science and Technology
Major Project (2021SHZDZX0100) and the Fundamental
Research Funds for the Central Universities

Candidate: Yang Hu

Student Number: 1932456

School/Department: College of Civil Engineering

Discipline: Engineering

Major: Architecture and Civil Engineering

Supervisor: Tang Hesheng

June, 2022

基于物理驱动深度学习的涡激振动问题研究

杨

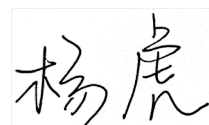
虎

同济大学

学位论文版权使用授权书

本人完全了解同济大学关于收集、保存、使用学位论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保存学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本学位论文全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制论文的部分或全部内容用于学术活动。

学位论文作者签名：

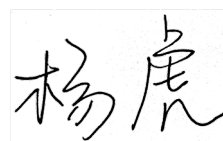


年 月 日

同济大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：



年 月 日

摘要

涡激振动 (Vortex-induced vibration, VIV) 广泛存在于海洋工程、桥梁工程以及结构工程等领域, 如柔性立管、大跨度桥梁以及高柔塔筒。VIV 是典型的非线性流固耦合问题, 计算效率是进行相关分析时遇到最大的挑战之一。传统有限元计算方法 (Finite element method, FEM) 和数据驱动深度神经网络 (Deep neural network, DNN) 在成本限制内所求得的 VIV 近似解精度亟需进一步提高。基于此, 本文基于物理驱动深度学习对二维圆柱 VIV 问题展开系统研究。通过优化训练方法和配置点采样策略以及引入迁移学习 (Transfer learning, TL) 策略, 进一步提高物理信息神经网络 (Physics informed neural network, PINN) 模型求解 VIV 正反问题的计算精度及效率, 实现“稀疏”数据条件下流场运动信息的时空域预测、结构振动信息的时域预测以及结构未知参数的识别。本文开展的相关研究对指导实际工程 VIV 预测具有重要意义, 主要内容如下所示:

(1) 针对传统 FEM 和数据驱动 DNN 计算精度及效率低下的问题, 本文基于物理驱动深度学习建立了强耦合 VIV 模型。同时利用计算流体理论和动网格技术对二维圆柱结构开展数值模拟, 获取标签数据构建训练样本集。基于 PINN 比较分析了强弱耦合 VIV 模型的表现性能, 结果表明强化物理信息对神经网络的约束后, 代理模型可以在较低的训练成本下具有更高的运行精度。

(2) 为了保证计算精度在容许情况下同时提高在“稀疏”数据条件下 PINN 求解 VIV 正反问题的计算效率, 本文提出分时段递推训练的方法, 采用分时段不同训练配置点采样策略, 仅利用有限时域长度信息来预测更长时间段的 VIV 信息。通过有效利用前一时段的网络训练特征以及预测信息, 后一时段逐渐减少训练配置点数, 从而减少训练成本。结果表明该方法可以在“稀疏”数据条件下实现流场运动信息的时空域预测和结构振动信息的时域预测, 同时准确识别出结构阻尼 c 和结构刚度 k 等未知参数。

(3) 为了进一步提高在“稀疏”数据条件下 PINN 方法的计算精度及效率, 本文提出 TL 增强的 PINN 模型。结合迁移策略, 源模型中的 VIV 公共特征知识可以加强目标任务的学习, 从而提高学习效率, 且无需大量标注样本集。结果表明, 尽管在训练过程中获得的数据量逐渐变小, 该方法与使用传统 PINN 算法的结果非常吻合。

关键词: 涡激振动, 计算流体力学, 物理信息神经网络, Navier-Stokes 方程, 迁移学习

ABSTRACT

Vortex-induced vibration (VIV) widely exists in the fields of ocean engineering, bridge engineering and structural engineering, such as flexible risers, long-span bridges and highly flexible towers. The VIV is a typical nonlinear fluid-structure interaction problem, and the traditional numerical methods, such as finite element method (FEM) and some data-driven methods faces great challenges in computational efficiency and accuracy. Because of its limitations, an improved physics informed neural network (PINN) model for solving the forward and inverse problem of VIV is proposed. In this model, in order to improve the computational efficiency and accuracy and solve the problem of sparse data, a new recursive training and configuration point sampling method and transfer learning (TL) strategy are adopted. The main contents are as follows:

(1) In order to improve the computational efficiency and accuracy of traditional FEM and data-driven deep neural network (DNN), a strong coupled VIV model based on physics-driven deep learning is proposed. The training dataset is provided by CFD simulation of VIV for a two-dimensional cylindrical structure. Through the comparative analysis of the strongly and weakly coupled VIV models based on PINN, the results show that the efficiency and accuracy of the strongly coupled VIV model are better.

(2) In order to improve the computational efficiency and accuracy of PINN model for solving the forward and inverse problem of VIV, a new recursive training and configuration point sampling method is proposed. Through taking advantage of the characteristic and predicted information, dataset with limited time-domain length can be used to predict information of VIV with longer time-domain length. The results show that the proposed method can accurately predict the information of VIV and identify the unknown structural parameters in sparse space.

(3) In order to further improve the computational efficiency and accuracy of PINN model and solve the problem of sparse data, a TL-enhanced PINN model is proposed. Combined with the transfer strategy, the common feature knowledge of VIV in the source model can enhance the computational efficiency and accuracy of the target task. The results show that although the training dataset is gradually getting smaller, the computational accuracy is still in good agreement compared with traditional PINN

algorithm.

Key words: Vortex-induced vibration, Computational fluid dynamics, Physics informed neural network, Navier-Stokes equation, Transfer learning

目录

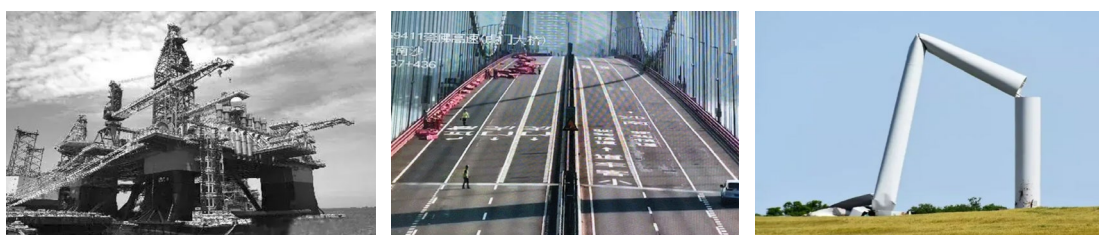
摘要.....	I
ABSTRACT.....	III
第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 研究问题.....	2
1.3 相关领域研究现状.....	2
1.3.1 涡激振动计算方法的研究现状.....	2
1.3.2 物理信息神经网络的研究现状.....	7
1.3.3 迁移学习的研究现状.....	10
1.4 本文基本研究内容介绍.....	13
第 2 章 涡激振动及物理信息神经网络理论简介.....	15
2.1 引言.....	15
2.2 涡激振动基本理论.....	15
2.2.1 产生机理及相关控制参数.....	15
2.2.2 基本控制方程.....	18
2.2.3 动网格技术.....	20
2.3 深度神经网络基本理论.....	23
2.3.1 深度神经网络简介.....	23
2.3.2 反向传播算法.....	24
2.3.3 损失函数的构建.....	26
2.3.4 激活函数的选择.....	28
2.3.5 神经网络训练参数.....	30
2.4 物理信息神经网络基本理论.....	32
2.4.1 物理信息神经网络简介.....	32
2.4.2 非线性偏微分方程正问题.....	34
2.4.3 非线性偏微分方程反问题.....	35
2.4.4 局部自适应激活函数.....	36
2.4.5 训练过程优化.....	38
2.5 本章小结.....	39
第 3 章 基于物理驱动深度学习的强耦合涡激振动模型.....	41
3.1 引言.....	41
3.2 涡激振动物理模型.....	41
3.3 强耦合物理信息神经网络模型.....	44
3.3.1 数据驱动深度神经网络拓扑结构.....	44

3.3.2	物理驱动深度神经网络拓扑结构.....	45
3.4	强弱耦合 PINN 模型表现性能比较	47
3.4.1	CFD 数值模拟.....	47
3.4.2	对比工况介绍.....	52
3.4.3	流场运动信息再现情况.....	53
3.4.4	结构振动信息再现情况.....	57
3.5	本章小结.....	59
第 4 章	基于物理驱动深度学习的涡激振动正反问题求解.....	61
4.1	引言.....	61
4.2	涡激振动正问题.....	61
4.3	正问题算例结果分析.....	64
4.3.1	流场运动信息预测情况.....	64
4.3.2	结构振动信息预测情况.....	67
4.4	涡激振动反问题.....	69
4.5	反问题算例结果分析.....	71
4.6	本章小结.....	72
第 5 章	基于迁移学习的涡激振动物理信息神经网络模型.....	75
5.1	引言.....	75
5.2	迁移学习概述.....	75
5.2.1	迁移学习背景介绍.....	75
5.2.2	迁移学习形式定义及分类.....	76
5.2.3	常见的迁移学习方法.....	77
5.3	涡激振动深度迁移学习模型.....	83
5.3.1	物理驱动深度迁移学习模型.....	83
5.3.2	算例工况介绍.....	84
5.4	算例结果分析.....	86
5.5	本章小结.....	105
第 6 章	结论与展望.....	107
6.1	主要工作及结论.....	107
6.2	课题展望.....	108
	参考文献.....	111
	致谢.....	119
	个人简历、在读期间发表的学术成果.....	121

第 1 章 绪论

1.1 研究背景及意义

VIV 在实际工程中发生频率极高，是具有强危害性的流固耦合现象^[1]。海洋工程中的柔性立管^[2]、桥梁工程中的大跨度桥梁^[3]以及结构工程中的高柔塔筒^[4]等深受其害，因 VIV 导致的疲劳损伤诱发了多起工程事故，如图 1.1 所示，给社会经济、人身安全以及自然环境等造成严重危害^[5]。



(a) Alexander Kierand 号平台 (b) 虎门大桥 (c) New Mexico 的 CE 风机

图 1.1 VIV 引起的工程事故

(1) 海洋荷载作用下的 VIV

随着国民经济开始进入快速发展的阶段，石油和天然气等自然资源逐渐成为保障社会经济持续增长、国民生活质量提升的关键性战略物资^[6]。如图 1.2 所示，在海洋工程中，立管结构、独立柱式平台以及锚缆等是连接海底油气资源开采设备和海平面工作平台的重要柔性结构，同时也是整个油气资源开发系统最薄弱的部分^[7]。在海洋荷载作用下，波浪力诱发的工作平台运动以及复杂洋流引起的 VIV 是导致柔性结构疲劳损伤的重要原因^[8]。



(a) 立管结构 (b) 独立柱式平台 (c) 锚缆

图 1.2 海洋荷载作用下的柔性结构

(2) 风荷载作用下的 VIV

在桥梁工程与结构工程中，风荷载是不可控的随机荷载^[9]。如图 1.3 所示，

大跨度桥梁和高柔塔筒都具有柔性较大、结构阻尼相对较小等特性，因此在与风荷载相互作用过程中极易发生结构损坏甚至完全倒塌^[10-11]。在风荷载作用下，静力与动力的双重作用引起的 VIV 是导致柔性结构产生疲劳损伤的重要原因^[12-13]。



(a) 大跨度桥梁



(b) 高柔塔筒结构

图 1.3 风荷载作用下的柔性结构

综上所述，在海洋工程、桥梁工程以及结构工程中，VIV 是导致柔性结构产生疲劳损伤的重要原因，严重影响了工程质量安全以及社会经济效益^[14]。为了增加柔性结构的服役周期，及时监测及预测流体运动信息和结构振动信息，则显得尤为重要和迫切。因此，本文开展对二维圆柱 VIV 的系统研究对指导实际工程 VIV 预测具有重要的理论意义及现实意义。

1.2 研究问题

进行 VIV 相关分析时，传统 FEM 经常受限于网格精度、复杂结构构型以及高雷诺数 (Reynolds number, Re) 等引起的计算效率问题。数据驱动 DNN 对训练成本的要求过高，且由于缺乏解释性，模型鲁棒性较弱。

基于此，本文引入 PINN 特性建立强耦合 VIV 模型，该模型具有“低成本、高精度、强鲁棒性”的特点；其次进一步优化训练方法和配置点采样策略，提高求解 VIV 正反问题的计算精度及效率，实现“稀疏”数据条件下流场运动信息的时空域预测、结构振动信息的时域预测以及结构未知参数的识别。

1.3 相关领域研究现状

1.3.1 涡激振动计算方法的研究现状

VIV 现象通常发生于大长径比柔性结构，多模态振动的复杂特性导致求解 VIV 问题相当困难^[15]。为了更清楚地解释 VIV 所包含的非线性机理，大量的学

者及研发团队尝试用不同的计算方法来研究相关问题。VIV 问题的计算方法共经历四个主要阶段，如图 1.4 所示，分别为经验及半经验模型阶段、有限元计算（FEM）阶段、纯数据驱动（Data-driven）的 DNN 阶段以及、物理驱动的 PINN 阶段。



图 1.4 VIV 计算方法的发展阶段

（1）第一阶段：经验及半经验模型

应用最为广泛的经验模型包括 DNV 模型^[16]、MARINTE 模型^[17]、LIC 模型^[18]以及 MIT-Vandiver 模型^[19]等。半经验模型中最具有代表性的是基于时域的 Van der Pol 尾流振子模型，这个概念模型是由 Bishop 和 Hasson 于 1964 年首次提出^[20]，他们把 VIV 的尾流作用比喻为一个非线性振子。随后尾流振子模型在此基础上被许多学者持续改进，逐渐发展出弹/刚性圆柱结构尾流振子模型^[21]、基于“能量流平衡”的尾流振子模型^[22]以及双耦合尾流振子模型等形式^[23]。

在该阶段中，基于经验及半经验模型提供了简化计算 VIV 问题的思路方法，可以有效避免直接求解流体运动的控制方程。但是由经验及半经验模型推导出的近似解误差较大，难以满足复杂 VIV 问题的精度要求。在计算机硬件工业快速发展的基础上，有限元数值模拟可以提供更为精确的求解方式，VIV 问题的计算方法开始向有限元计算阶段过渡。

（2）第二阶段：有限元计算

随着软件开发技术的高速发展，目前有多种仿真流体力学商用软件可以用于 VIV 有限元模拟计算，包括 ANSYS Fluent、ANSYS CFX、OpenFOAM、STAR-CCM、XFlow、Autodesk Simulation CFD、FloEFD 以及 Numeca 等。VIV 是具有代表性的计算流体力学（Computational fluid dynamics, CFD）问题，CFD 问题的数值模拟方法可以划分为两个大类^[24]：一是直接数值模拟（Direct numerical simulation, DNS），二是非直接数值模拟（Non-direct numerical simulation, NNS）。

DNS 方法是最传统且精准的数值模拟方法，可以直接求解每个时间步的流体运动方程（Navier-Stokes equations, N-S 方程）。NNS 方法是经简化后应用最为普遍的数值模拟方法。如图 1.5 所示，依据流体模型以及方程简化方式的差异，NNS 方法大致可以分为三类：大涡模拟（Large eddy simulation, LES）、统计平均

法 (Statistical average method, SAM) 以及 Reynolds 平均法 (Reynolds average Navier-Stokes, RANS)。在这三类方法中, 综合考虑计算成本及精度, RANS 方法是最常被采用的研究方法。RANS 方法通过将瞬态的脉动量以时均化方程体现, 来避免直接求解瞬态的 N-S 方程。RANS 方法包含两个主要的计算模型: Reynolds 应力模型和涡黏模型。其中 Reynolds 应力模型分为 Reynolds 应力模型和代数应力方程模型, 涡黏模型分为零方程模型、一方程模型 (RNG $k-\epsilon$ 模型、Realizable $k-\epsilon$ 模型、Standard $k-\omega$ 模型以及 SST $k-\omega$ 模型等) 以及两方程模型。

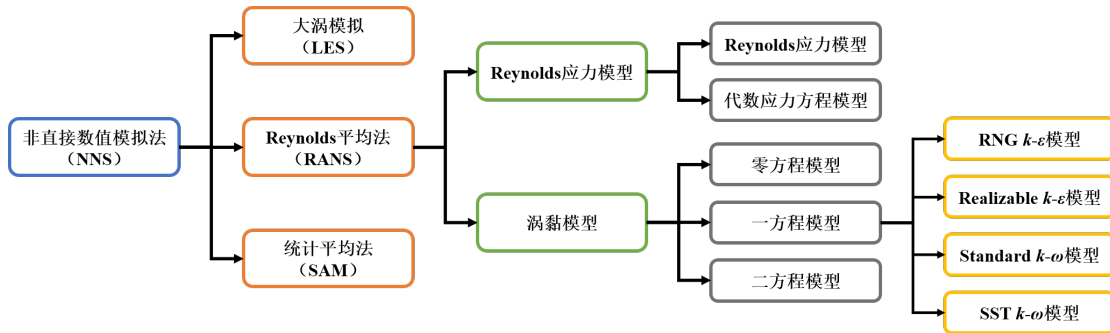


图 1.5 非直接数值模拟 (NNS) 方法分类

2001 年, Lucor 等^[25]基于 DNS 方法分别给出了长径比大于 500 的柔性圆柱结构在线性和指数剪切作用下的 VIV 数值模拟结果。结果显示 DNS 方法可以模拟出与实验中相同的漩涡错位, 同时得到了相应的力分布。

2003 年, Skaugset 等^[26]利用 DNS 方法研究了从振动圆柱结构圆孔中引入径向水射流来抑制 VIV 的方法。数值模拟了通过气缸壁上开口流出的射流, 并对不同的射流数量、射流位置和射流流量进行了参数研究。结果证明将开口设置在气缸圆周位置正负 120 度, 且沿两排径向水射流时, 可以最大程度地抑制 VIV 的振幅。

2004 年, Guilmineau 等^[27]采用不可压缩 RANS 方程对 VIV 过程中圆柱结构周围的漩涡脱落进行了数值研究。数值方法使用质量和动量通量的一致物理重建, 湍流模型采用 SST $k-\omega$ 模型。最后将数值模拟结果与 1996 年 Khalak 等人的实验结果进行对比, 结果证明依据数值模拟的初始条件, 可以正确地预测 VIV 过程中的最大振幅。

2012 年, 郭宇^[28]基于 LES 方法对三维柔性圆柱立管结构的 VIV 进行了数值模拟, 主要包括层流模型的小长径比圆柱立管结构、层流模型的大长径比圆柱立管结构以及湍流模型的大长径比圆柱立管结构三部分内容。并以此为基础对不同流场条件下圆柱立管结构的疲劳损伤进行研究。

2013 年, Muralidharan 等^[29]通过耦合 RANS 方程和结构的动力学方程, 数值模拟了光滑柔性圆柱结构的 VIV 过程。在研究中设计并实施了一种基于零剂

量注射剂的主动流量控制方法，以抑制 VIV 的发生。结果证明该方法通过闭环反馈系统实现精确控制尾迹控制的驱动量，从而有效地抑制甚至消除 VIV。

2015 年，Bargi 等^[30]采用 LES 方法对均匀圆柱结构和锥形圆柱结构的 VIV 实验研究进行了数值模拟。通过与实验结果进行对比，证明数值模拟结果可以较好地预测结构振动响应地初始分支和上分支，无法很好地预测较低的分支。

2015 年，杨志斌^[31]基于 RANS 方法分别对雷诺数 $Re=3900$ 的单圆柱结构绕流现象以及不同雷诺数下的串联双圆柱结构 VIV 进行数值模拟。通过与实验结果进行对比，证明 RANS 方法中 SST $k-\omega$ 模型的准确性最高。同时对于串联双圆柱结构 VIV，圆柱结构在不同方向的振动振幅与流体雷诺数、圆心间距比以及上下游圆柱位置的设置等均有关系。

2015 年，李骏等^[32]采用 RANS 方法，并基于 SST $k-\omega$ 模型，对低质量比弹性支承刚性圆柱结构的 VIV 进行数值模拟。以结构振动位移、频率响应以及尾迹形态等为指标，与 Williamson 的实验结果进行对比，结果证明 SST $k-\omega$ 模型可以有效完成上述实验内容的数值重构。

2017 年，曾攀^[33]基于 RANS 方法和有限体积法分别对柔性圆柱结构的单自由度和双自由度 VIV 进行数值模拟。结果证明质量比 m^* 和质量阻尼比 $m^*\zeta$ 分别是结构锁定区间长度和结构共振振幅的决定性因素。且当质量比 $m^*=3.5$ 时，双自由度 VIV 和单自由 VIV 的振动响应和尾迹形态开始出现明显差异。

2019 年，赵燕等^[34]采用 LES 方法对雷诺数 Re 在 (10000, 40000) 区间内的串联双圆柱结构 VIV 进行数值模拟。主要分析了折减风速对结构振动特性和流场的尾迹形态的影响，结果证明该数值模拟结果与风洞实验结果保持一致。

在该阶段中，计算 VIV 问题时经常受限于划分网格精度、复杂结构构型以及高雷诺数 Re 流体等所引起的计算效率问题。纯数据驱动的 DNN 可以有效规避上述问题，建立起具有非线性逼近性能的数据处理模型，高效率地计算 VIV 相关问题，VIV 的计算方法开始向纯数据驱动的 DNN 过渡。

(3) 第三阶段：纯数据驱动的深度神经网络

随着人工智能 (Artificial intelligence, AI) 的进一步发展，面向数据驱动的 DNN 可以不深究 VIV 的力学机理，仅依靠构建 VIV 样本集和相应的损失函数，即可高效地通过网络映射完成 VIV 预报等任务。

2019 年，翟立宾^[35]分别构建了以流体速度、轴向空间位置和轴向力为网络输入量的 BP 神经网络模型、支持向量机模型和 Gaussian 过程回归模型，来预报 VIV 过程中柔性圆柱结构在两个坐标方向上的振动位移以及频率响应。结果证明支持向量机模型具有更理想的鲁棒性和预报精度。

2019 年，高喜峰^[36]以相关实验数据作为训练样本集，构建了四个用来预报

VIV 振动响应的 BP 神经网络模型。同时将预报结果与实验数据进行对比，结果表明这四个模型在限定范围内可以准确预报柔性圆柱结构在横流向和顺流向上的振动位移以及频率响应。

2020 年，孟杨^[37]提出了用于预报 VIV 绕流力和流体雷诺数 Re 的卷积神经网络 (CNN) 模型。结果证明该模型可以准确预报 VIV 的绕流升力和绕流阻力，并且尤其适用于耦合边界维度较大的情况。与 OpenLB 物理模型相比，该模型预报流体雷诺数时，将预报误差缩小了一个数量级。

2021 年，廖洋洋^[38]以实验数据作为训练样本集，分别基于 BPNN、GA-BPNN、RBFNN 以及 GRNN 这四类深度神经网络对柔性立管的 VIV 疲劳损伤进行预报。同时将四个神经网络的预报结构与时域法计算结果进行比对，综合考虑训练成本和预报精度等因素，RBFNN 模型展现了更好的表现性能。

2021 年，Misaka^[39]提出了基于可视化实验流的 DNN 模型，来估计 VIV 过程中作用于圆柱结构的绕流升力。在该模型中，绕流力可以通过计算尾迹模式得出。将 DNN 模型预报出的绕流升力与实验值进行比对，结果证明在相同流量和振荡模式下，DNN 模型所预报的绕流升力系数均方根值为 0.82，与实验值 0.8 相当。

2021 年，Foster 等^[40]利用递归神经网络中的 LSTM 神经网络来描述非定常流体载荷。该模型以 VIV 速度场的离散时空测量数据为训练样本集，模拟了 VIV 过程中漩涡脱落作用下的绕流力以及相应的圆柱结构振动位移。

在该阶段中，计算 VIV 问题时也遇到了两大挑战：一是由于缺乏力学机理容易导致训练过拟合，二是受限于目前的 VIV 监测技术及设备，很难收集到足量的监测数据构建训练样本集。因此先验物理信息开始作为额外约束来解决上述问题，VIV 计算方法开始向物理信息神经网络过渡。

(4) 第四阶段：物理信息神经网络

VIV 的主要控制物理方程为流体 N-S 方程和固体动力学方程。为了进一步突破 VIV 问题的计算效率，N-S 方程等作为先验物理知识被嵌入至 DNN 结构来构建相应的 PINN 模型。通过数据和物理信息的双重驱动，PINN 模型可以凭借更低的训练成本来无限逼近 VIV 问题的真实解，同时保持有较好的鲁棒性和泛化能力。

2018 年，Raissi 等^[41]将描述不可压缩流体动量守恒的 N-S 方程作为先验物理信息嵌入至 DNN 结构中，建立了 VIV 弱耦合 PINN 模型。同时利用粒子图像测速法构建了雷诺数 $Re=100$ 的训练样本集，最终的计算结果表明该模型凭借较少的数据成本准确预报了流场运动信息以及结构振动信息。

2018 年，Wang 等^[42]利用 PINN 方法来学习 RANS 模拟中雷诺应力差异的函

数形式，从而预报不同 VIV 运动中的雷诺应力，并将预报出的雷诺应力传播到平均流场中。以方形管道内完全展开的紊流为例，在不变量的完整性基础上丰富输入特征，结果证明在雷诺应力及其传播速度下，该方法具有良好的预报性能。

2021 年，Cheng 等^[43]利用 PINN 方法求解圆柱结构的高雷诺数 VIV 和尾迹诱导振动问题。通过将 RANS 方程和附加湍流涡旋粘度系数嵌入到损失函数中，提出三种场景来验证 PINN 模型的表现性能，结果证明 PINN 模型对未知参数和高雷诺数流场的推断和重构更加有效。

2021 年，Kharazmi 等^[44]建立了一个稳健的 PINN 计算框架来推断柔性圆柱结构 VIV。通过给定时空中关于位移和水动力的分散数据，来重新还原整个 VIV 运动，同时通过精确估计结构参数来表征诱导振动。结果证明所开发的 PINN 框架具有同时分析不同时空子域数据的灵活性，从而产生更准确的估计。

2021 年，Li 等^[45]基于 Scanlan 半经验模型和桥梁涡激振动器，推导出描述涡激振动器动力学的时变常微分方程，并以此为物理信息构建 PINN 模型。同时对真实自然风场中的 VIV 响应进行长时间预报，以克服真实风的复杂性、高雷诺数效应以及桥梁结构不确定性等不利因素。

2022 年，Bai 等^[46]将基于参数化 N-S 方程的 PINN 模型从层流区域拓展至湍流区域，在 N-S 方程中引入一个人工黏度系数，并将其设定为隐藏的网络输出变量。同时以来自 CFD 数值模拟的散射速度和染料痕量浓度数据构建训练样本集，计算结果表明该模型在 VIV 环境下的湍流模拟和重建方面呈现上佳的表现性能。

1.3.2 物理信息神经网络的研究现状

随着机器学习模型变得越来越复杂，数据驱动的机器学习方法难以满足对模型可解释性的需求。主要原因有两个：一是缺乏足够的训练样本来维持模型的泛化能力与高性能表现，二是缺乏满足自然规律或客观物理准则等限制条件，也就是说缺乏可解释性。因此越来越多的学者开始研究如何将先验物理知识嵌入到学习过程中来改进机器学习模型，例如将逻辑规则、代数方程等嵌入至机器学习模型的损失函数中^[47-48]。

2017 年，Raissi 等^[49]人第一次推出“PINN”的概念，将潜在的物理定律编码为先验信息，由此产生的神经网络模型形成了一种新的高效普适函数近似器。从而替代传统的离散数值方法（Discrete numerical method, DNM）来无限逼近非线性偏微分方程（Nonlinear partial differential equation, NPDE）的正反问题真实解，同时有效避免传统数值方法所面临的维数灾难、反问题无效计算等问题。Raissi 团队分别基于离散时间模型和连续时间模型建立了 NPDE 的正反问题求解框架。对于 NPDE 的正问题，PINN 仅需少量的训练样本即可得到高精度的近似

解，同时神经网络模型保持有较高的鲁棒性和泛化能力。对于 NPDE 的反问题，NPDE 的未知物理参数可以直接嵌入网络结构中，直接参与神经网络模型的训练，从而求得未知物理参数的高精度近似解。

基于 Raissi 团队所建立的 PINN 框架，PINN 方法随后被广泛应用于各类学术领域来求解代表不同物理规律的 NPDE 正反问题。如图 1.6 所示，除了上文所提到的 VIV 相关问题，PINN 方法同样被用于尝试解决隐藏流体力学、NSFnets、特殊介质的地下水流、固体力学、高速空气动力学、心脏颤动诊断、超材料设计、电力系统动力学、图像重构以及金属裂缝检测等相关问题。

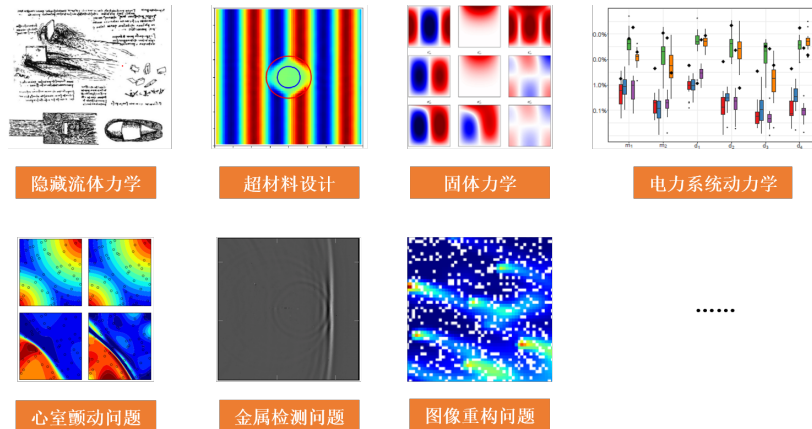


图 1.6 PINN 方法的应用领域

2019 年，Fang 等^[50]基于高频亥姆霍兹方程（Helmholtz equations）建立 PINN 模型，来解决超材料设计过程中存在的实际问题，例如材料隐身、旋转设计以及集中器处理等。结果证明该方法在处理连续参数和分段常数时具有更优越的灵活性，同时可以在数值实验中高效地解决相关基准问题。

2019 年，Mao 等^[51]将欧拉方程（Euler equations）嵌入深度神经网络结构中，建立 PINN 模型来近似模拟高速空气动力流动，解决一维区域和二维区域的正反问题。数值算例的结果表明，当相关的物理守恒定律被赋予于随机点集上，PINN 算法在面对无法用标准技术解决的正反问题时，比传统的数值方法更有优势。

2020 年，Raissi 等^[52]基于 N-S 方程开发了隐藏流体力学（Hidden fluid mechanics, HFM）PINN 模型框架，来实现在物理和生物系统中的流动可视化。HFM 框架不受几何形状、初始条件以及边界条件的影响，可以通过直接提取不可测量的数据样本，来快速解决物理或生物医学的相关问题。同时结果表明在该 PINN 模型中高频调频对观测样本中的低分辨率和大量噪声具有鲁棒性，这对流动可视化的潜在应用具有重要意义。

2020 年，Jin 等^[53]通过自动微分技术将 N-S 方程中的速度-压力（VP）公式和涡度-速度（VV）公式直接编码至 DNN 结构中，建立 N-S 流动网（NSFnets），

以克服流固耦合数值模拟过程中对于不可压缩层流和湍流的限制。最终以一些标准的基准问题验证了 NSFnets 在预报精度、收敛速度、计算代价以及灵活性上的优越性。

2020 年, Chen 等^[54]利用亥姆霍兹方程 (Helmholtz equations) 在 TM 极化激励下对弱非齐次二维介质的约束建立相应的 PINN 模型, 用于求解光子超材料中具有代表性的逆散射和纳米光学技术问题。结果证明 PINN 方法可以帮助设计出新颖的功能性纳米结构, 并显著拓宽设计超材料的空间。

2020 年, Guo 等^[55]提出一种改进的基于物理信息的神经结构搜索 (Neural architecture search, NAS) 方法, 建立求解多孔介质中地下水流动问题的模型。结果表明, 使用高斯相关函数的对数电导率场非常贴合基于 NAS 方法的 PINN 模型, 在拟合 NPDE 理论解的方面具有很大的潜力。

2020 年, Wang 等^[56]利用对流-扩散方程 (Advection-diffusion equation) 约束神经网络模型, 用于增强物理模型与卫星观测样本数据之间的一致性, 以及提高空间分辨率。由此发展出的超分辨率 (Super resolution, SR) 技术可以精确地将较低分辨率的图像重建为高分辨率图像, 从而完成图像重构。

2020 年, Misyris 等^[57]基于 PINN 方法建立了电力系统应用框架, 来描述电力系统中的稳态和动态行为。该模型可以以更低的成本确定电力系统的行为动态, 如转子角度、频率以及惯性和阻尼等不确定参数。最后以单机无限总线系统为例, 证明 PINN 方法可以精准确定转子的角度与频率, 同时比传统方法快约 87 倍。

2020 年, Shukla 等^[58]基于声波方程 (Acoustic wave equation) 建立 PINN 模型, 来解决识别金属板表面断裂裂纹的问题。同时引入一个可伸缩的参数来构建自适应激活函数, 以此实现 PINN 模型的最佳表现性能。该方法通过 PINN 模型来估计金属板的声速, 再结合声速与空间的关系, 从而识别并表征裂纹的位置 (即声速下降的位置)。

2020 年, Costabal 等^[59]在 PINN 模型中构建了心脏激活映射, 解释了潜在的波传播动力学原理, 同时量化了与预测相关的认知不确定性。最后以左心房的个性化电生理模型为例, 证明了 PINN 方法优于传统线性插值方法和高斯过程回归, 可以有效降低程序计算时间, 同时保持更低的错误水平。

2021 年, Stiasny 等^[60]利用广泛应用的相量测量单元 (PMUs), 以开发一种动态更新和参数估计工具为目标, 研究了用于监测未来电力系统频率动态的 PINN 模型的表现性能。最后与无味卡尔曼滤波 (Unscented kalman filter, UKF) 算法做比较, 结果证明 PINN 方法更有潜力应对非线性低惯性系统、测量噪声以及样本数据缺乏等挑战。

2021 年, 曾靖文^[61]尝试将伯格方程 (Burgers equation) 嵌入神经网络结构

中, 来更精确地模拟高速流场的湍流激波运动, 从而求解伯格方程有关的正反问题。同时对比了自适应激活函数与传统激活函数在优化神经网络训练过程中的差异, 发现自适应激活函数可以加速损失函数的收敛速度。

2021 年, Haghghat 等^[62]将 PINN 方法应用于固体力学的反演和代理建模, 把固体力学的动量平衡和本构关系纳入至 DNN 中。并以 von Mises 弹塑性固体材料为例说明了 PINN 方法在非线性问题中的扩展, 结果证明该模型可以准确预测相关参数, 同时将敏感性分析和代理模型纳入至模型框架内。

2021 年, 王伟杰^[63]基于粒数平衡方程 (Population balance equation) 建立溶液结晶过程的 PINN 模型, 来研究溶液结晶过程的机理表现。最后以谷氨酸为例进行溶液结晶过程的仿真设计, 结果证明在最优恒定过饱和度状态下该模型到达最佳性能表现。

2022 年, Zhang 等^[64]基于线弹性、超弹性以及塑性谱的本构关系建立 PINN 模型, 来识别材料内部未知的几何和材料参数, 从而描述材料内部结构及缺陷。结果证明对于含内部空洞或夹杂异物的材料, 该模型可以预测内部空洞的大小、形状和位置, 以及夹杂异物的弹性模量。

2022 年, Gokhale 等^[65]利用实测数据和建筑参数的先验知识, 实现了一个以建筑物理为指导的神经网络模型, 可以模拟室温、能耗以及建筑热质量温度等隐藏状态的时间演化过程。结果证明在更长的预测视野内, PINN 模型实现了比传统神经网络更准确的预测, 同时花费更少的样本数据成本。

PINN 方法凭借着“稀疏数据”实现了“快速且高精度的逼近”, 帮助许多学科领域突破了计算效率的限制, 以更科学的方式解决了本领域内的难题。PINN 方法目前处于不断探索的阶段, 所使用的建模以及训练优化方法相对单调。该方法仍需在多尺度物理系统、非牛顿流体以及多模态样本数据等方面不断优化改进, 以便于应用到更多的学科领域和实际工程项目中。

1.3.3 迁移学习的研究现状

目前机器学习 (Machine learning, ML) 已经取得了巨大进步, 但对于某些现实场景, 它仍然存在有一定的局限性^[66]。深度学习的理想场景是有大量的标记训练样本, 然而在许多场景中, 收集足量的训练数据是昂贵的、耗时的, 甚至是不现实的。**TL 方法专注于跨领域的知识迁移**, 是解决上述问题的一种很有前途的 ML 方法^[67]。

TL 方法的概念最初来自于教育心理学^[68]。心理学家 C.H. Judd 提出的迁移泛化理论指出, 学习迁移是经验泛化的必然结果。根据这一理论, 两个有联系的学习活动即可实现迁移。图 1.7 是现实生活中具有迁移潜力的客观示例^[69], 每两

个事物之间均存在有客观联系。迁移的目的是利用来自源域（Source domain）的知识来提高学习性能或者最小化在目标域（Target domain）所需的标记示例数量。目前 TL 方法已经被广泛应用于多种学科领域，如图 1.8 所示，主要包括建筑结构时变可靠性预测、自动化大脑图像分类、植物病害图像识别、轴承故障诊断、钢材表面缺陷检测、时间序列预测、小材料数据增强预测以及结构损伤识别等。



图 1.7 具有迁移潜力的客观示例

2020 年，Zafar 等^[70]在仅利用当前时间性能函数信息的基础上，对基于 TL 方法的结构或构件的时间依赖可靠性预测问题进行了初步研究，来解决结构失效信息缺乏的问题。利用转移分量分析的域自适应方法，在当前和未来的时间间隔内建立 Kriging 代理模型来生成随机过程的相关随机样本，以寻找共同的潜在空间。结果证明，该方法可以准确预测时变问题的失效概率，且具有较高的精度。

2020 年，Kaur 等^[71]基于具有迁移功能的深度 CNN 来研究病理脑图像分类问题。该研究使用了多种预训练的深度 CNN，如 Alexnet、Resnet50、VGG-16 等，在训练过程中通过替换模型的最后几层隐藏层，以适应应用程序的新图像识别。结果证明，经过迁移训练的 Alexnet 在更短的时间内表现出了更佳学习性能，三种测试数据集的准确率分别为 100%、94%和 95.52%。

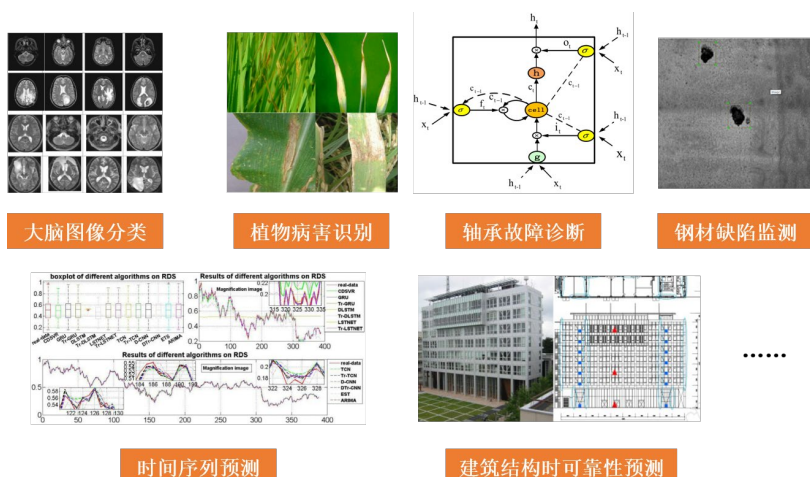


图 1.8 TL 方法的应用领域

2020 年，Chen 等^[72]研究了用于植物叶片危害识别的深度 CNN 的 TL 方法，并考虑使用从典型的海量数据集中学习的预训练模型，随后转移到由新训练数据

集的特定任务中。VGGNet 被选定为预训练模型，结果表明该方法在公共数据集的验证精度不低于 91.83%，在复杂背景条件下的分类预测平均精度达到 92.00%，有效地实现了植物病害的检测。

2020 年，Wu 等^[73]提出了一种自适应深度转移学习轴承故障诊断方法，来克服故障数据缺乏的不利状况。通过构建基于实例迁移学习的长短时记忆递归神经网络模型，同时采用特征 TL 方法联合分布自适应来缩小源域和目标域数据集在概率分布上的差异。结果表明所提方法在标注故障数据稀缺时具有令人满意的有效性和鲁棒性。

2021 年，Shu 等^[74]针对传统的带钢表面缺陷检测和识别方法不能适应实际检测环境的变化，提出了基于深度学习的 TL 方法。将基于 ImageNet 数据集训练的 ResNet 网络转移到 R-CNN 经典目标检测算法中，利用带钢表面缺陷数据集进行实验验证。结果证明该方法具有更高的检测精度，更适合于带钢表面缺陷的检测应用。

2021 年，Ye 等^[75]提出了一个具有深度 CNN 架构的迁移学习框架，来研究时间序列预测的问题。通过动态时间规整和 Jensen Shannon 发散相结合的方法来确定合适的源域，同时采用一组合成数据集和两组实际数据集的验证，验证了该方法在可用标记数据缺失的条件下仍具有有效性。

2021 年，Gupta 等^[76]提出了一个跨属性的深度迁移学习框架，该框架利用在大数据集上训练的模型来建立在不同属性的小数据集上的模型。通过在 39 个模拟数据集和 2 个实验数据集上测试所提出的框架。结果证明以元素部分作为网络输入量的迁移学习模型优于从头开始训练的机器学习模型，该框架可以广泛应用于解决材料科学中的机器学习应用中的小数据挑战。

2022 年，Lin 等^[77]通过设计一个特征提取器来生成损伤敏感和领域不变的特征，来解决跨域的结构损失检测（Structural damage detection, SDD）问题。根据数据模型和实际结构的数据来设计和训练自适应神经网络，同时不需要对实际结构进行损伤标记。数值实验结果表明该方法的性能表现优于传统深度 CNN 模型。

综上所述，TL 方法通过寻找不同领域之间的公共潜在空间，来“低成本”且“高精度”地解决各个学科领域中的机器学习难题。但是，TL 方法目前仍旧处于在不断探索的初级阶段，还有许多研究方向需要不断提升。首先，迁移技术可以进一步探索和应用于更广泛的应用领域，需要用新的方法来解决更复杂场景下的知识转移问题。其次，需要更好的衡量跨领域的可移动性，避免负迁移结果的出现。再次，TL 方法的可解释性仍需要深入研究。最后，可以逐渐展开 TL 方法的理论研究，为 TL 方法的有效性和适用性提供理论支持。

1.4 本文基本研究内容介绍

海洋立管、大跨度桥梁以及高柔塔筒等柔性结构极易发生 VIV，对结构本身造成严重的疲劳损伤，导致结构部分失效甚至完全破坏。传统 FEM 和数据驱动 DNN 在成本限制内所求得的 VIV 近似解精度亟需进一步提高。基于此，本文基于物理驱动深度学习对二维圆柱 VIV 问题展开系统研究，各章节的逻辑关系如图 1.9 所示，具体工作内容如下：

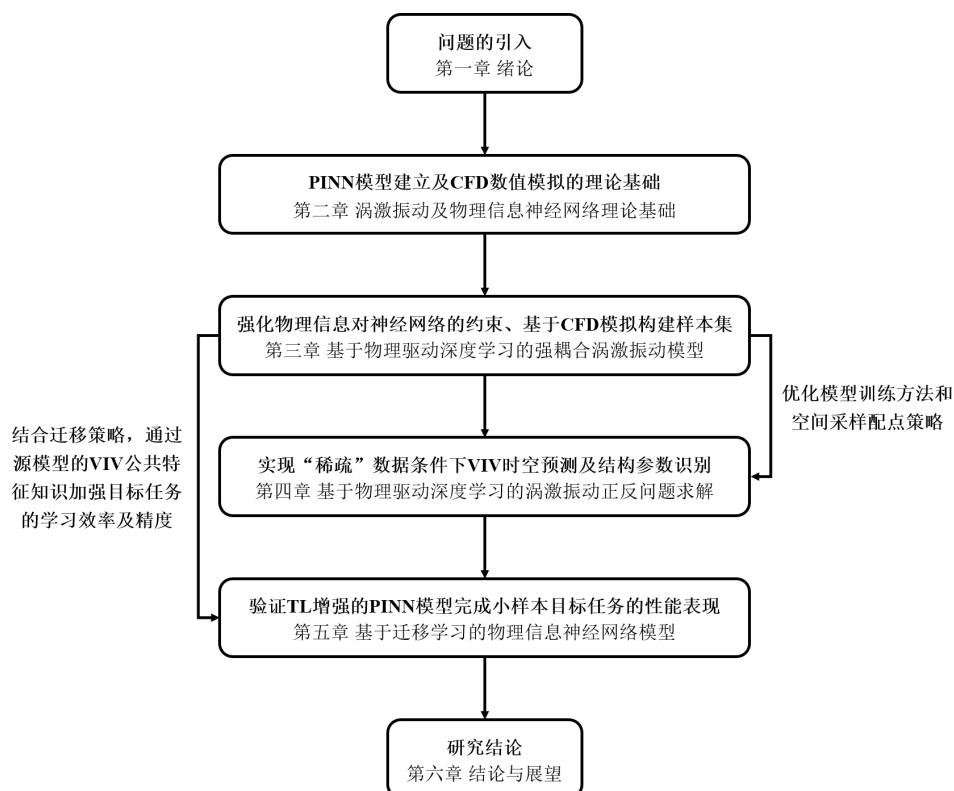


图 1.9 各章节逻辑关系

第一章为绪论。本章依次阐述了课题的研究背景、意义和主要问题，并详细梳理了 VIV 计算方法、PINN 以及 TL 的国内外研究现状。

第二章为涡激振动及物理信息神经网络理论简介。本章首先对 VIV 基本理论进行研究，为后续建立基于物理驱动深度学习的强耦合 VIV 模型以及 CFD 数值模拟提供理论基础。其次，本章将对数据驱动 DNN 基本理论进行详细介绍，主要包括：反向传播算法、损失函数的构建方法、对比不同激活函数以及神经网络训练参数的设置依据。最后，本章将对物理驱动 PINN 基本理论进行详细介绍，主要包括：NPDE 正反问题的算法逻辑、局部自适应激活函数的构建方法以及训练过程的优化方法。

第三章为基于物理驱动深度学习的强耦合 VIV 模型。本章将流体 N-S 方程

和流固耦合条件方程同时嵌入数据驱动 DNN 结构体系中，强化建立基于物理驱动深度学习的强耦合 VIV 模型。同时利用计算流体理论和动网格技术对二维圆柱结构开展数值模拟，获取标签数据构建训练样本集。最后以训练成本和计算精度为指标，基于 PINN 比较分析了强弱耦合 VIV 模型的表现性能。

第四章为基于物理驱动深度学习的 VIV 正反问题求解。本章提出分时段递推训练的方法，采用分时段不同训练配置点采样策略，仅利用有限时域长度信息来预测更长时间段的 VIV 信息。通过有效利用前一时段的网络训练特征以及预测信息，后一时段逐渐减少训练配置点数，从而减少训练成本，来提高“稀疏”数据条件下求解 VIV 正反问题的计算精度及效率。正问题为时空预测问题，包括：流场运动信息的时空域预测和结构振动信息的时域预测。反问题为参数识别问题，包括：结构阻尼 c 和结构刚度 k 等结构参数。

第五章为基于迁移学习的涡激振动物理信息神经网络模型。本章提出了 TL 增强的 PINN 模型，来进一步突破计算精度及效率的限制。结合迁移策略，源模型中的 VIV 公共特征知识可以加强目标任务的学习，从而提高学习效率，且无需大量标注样本集。通过三个具有不同数据来源、规模的数值算例，对比分析了深度迁移学习模型和传统 PINN 模型的表现性能，验证 TL 方法对强耦合 PINN 模型的增强作用。

第六章为结论与展望。本章主要归纳总结了全文的重要结论，同时反思了本课题存在的不足之处，对未来研究方向进行了展望。

第 2 章 涡激振动及物理信息神经网络理论简介

2.1 引言

传统 FEM 方法被用于分析 VIV 问题时，经常受限于网格精度、复杂结构构型以及高雷诺数 Re 等引起的计算效率问题。随着深度学习 (Deep learning, DL) 的进一步发展，PINN 方法被用于研究 VIV，该方法不仅可以满足样本数据的分布规律，而且也遵守由 NPDE 解释的物理规律。

首先，本章将对 VIV 基本理论进行研究，为后续建立基于物理驱动深度学习的强耦合 VIV 模型以及 CFD 数值模拟提供理论基础。其次，本章将对数据驱动 DNN 基本理论进行详细介绍，主要包括：反向传播算法、损失函数的构建方法、对比不同激活函数以及神经网络训练参数的设置依据。最后，本章将对物理驱动 PINN 基本理论进行详细介绍，主要包括：NPDE 正反问题的算法逻辑、局部自适应激活函数的构建方法以及训练过程的优化方法。

2.2 涡激振动基本理论

2.2.1 产生机理及相关控制参数

在流场中，当初始来流 U 经过结构表面时，其内部压力骤升，导致靠近结构表面的流场质点速度变小，在流固耦合面形成漩涡分离层。持续脱落的漩涡具有周期性，从而在垂直于初始来流 U 的方向生成绕流升力 F_l (Fluid lift force)，诱发结构产生横流向 (Cross-flow direction) 振动位移。同时也会在平行于初始来流 U 的方向生成绕流阻力 F_d (Fluid drag force)，诱发结构产生顺流向 (In-line direction) 振动位移，如图 2.1 所示。结构双向振动位移会反作用于流场，导致流场的速度场和压力场发生变化。绕流升力 F_l 和绕流阻力 F_d 的周期性存在较大差异，通常升力周期 T_l 和阻力周期 T_d 的关系为式 (2.1)。与横流向振动响应相比，顺流向振动响应非常微弱，甚至无法在同一数量级上比较，因此实际工程中往往不重视顺流向的振动响应^[78]。

$$T_d = \frac{T_l}{2} \quad (2.1)$$

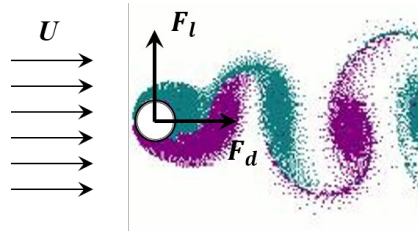


图 2.1 VIV 绕流力方向

初始来流 U 经过结构表面所产生的脱落漩涡具有不同形式，流体雷诺数 Re 大小是决定性因素^[79]。如图 2.2 所示，不同雷诺数 Re 范围内，会存在截然不同的漩涡脱落模式。表 2.1 梳理了不同的漩涡脱离模式。

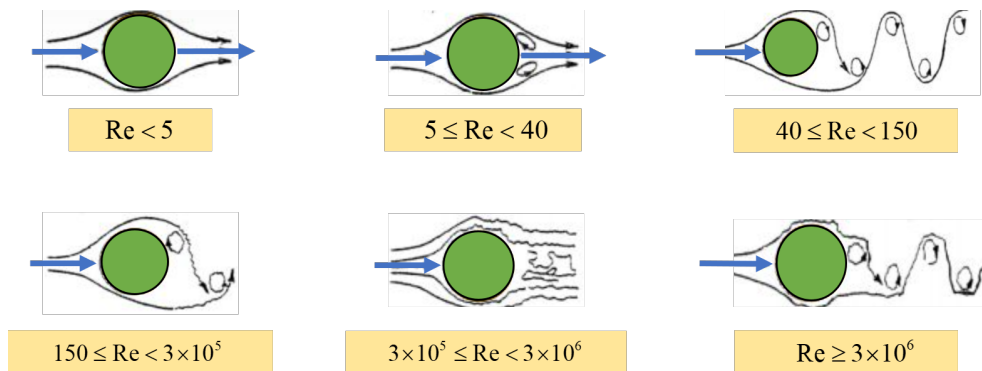


图 2.2 不同的漩涡脱落形式

表 2.1 不同雷诺数 Re 下的脱落模式

雷诺数 Re 范围	流体状态	脱落模式
$Re < 5$	层流	流固耦合面暂无明显分离情况产生
$5 \leq Re < 40$	层流	流固耦合面开始出现分离层，固体结构表面产生一对相对稳定的微小漩涡
$40 \leq Re < 150$	层流	漩涡开始呈周期性的脱落，固体结构表面交替生成层流状态下的漩涡
$150 \leq Re < 3 \times 10^5$	层流向湍流过渡状态	开始向亚临界范围过渡，固体结构表面交替生成湍流状态下的漩涡
$3 \times 10^5 \leq Re < 3 \times 10^6$	湍流	漩涡脱落无明显周期性
$Re \geq 3 \times 10^6$	湍流	开始进入超临界范围，“涡街”重新出现

VIV 的相关控制参数基本为无量纲参数，主要分为三类：**流体控制参数**、**结构控制参数**和**流固耦合控制参数**。流体控制参数包括**雷诺数 Re** 和**斯特劳哈尔数 St** ，结构控制参数包括**质量比 m^*** 、**长径比 λ** 和**阻尼比 ξ** ，流固耦合控制参数包括**绕流升力 F_l** 、**绕流阻力 F_d** 和**约化速度 Vr** 。

(1) 流体控制参数

1) 雷诺数 Re

雷诺数 Re 是流体黏性力和流体惯性力的比值, 用来描述流场域的流动状态, 如式 (2.2) 所示。其中 U 是流场的来流速度, 单位是 m/s ; D 是结构特征直径, 单位是 m ; ρ 是流体密度, 单位是 Kg/m^3 ; μ 是动力黏性系数, 单位是 $Pa \cdot s$ 。

雷诺数 Re 较小时, 流场黏性力呈主导作用, 流体呈稳定的层流状态; 雷诺数 Re 较大时, 流场惯例力呈主导作用, 流体呈紊乱无规则的湍流状态。

$$Re = \frac{UD\rho}{\mu} \quad (2.2)$$

2) 斯特劳哈尔数 (Strouhal number, St 数)

斯特劳哈尔数 St 是表示流体非定常性质的无量纲参数, 与漩涡脱落结构表面的频率 f 有关, 表达式为式 (2.3)。其中 f 是漩涡脱离结构表面的频率, 单位是 Hz 。

$$St = \frac{fD}{U} \quad (2.3)$$

国内外大量实验结果表明, 斯特劳哈尔数 St 与雷诺数 Re 存在一定的变化关系。图 2.3 是张杰团队^[80]通过实验研究所得到的圆柱绕流变化规律。

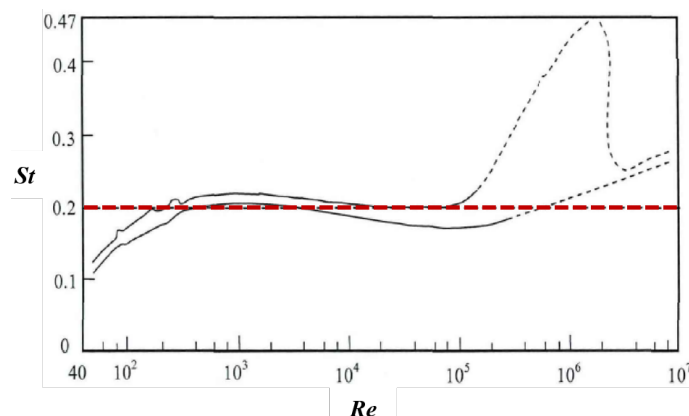


图 2.3 圆柱绕流中 St 随 Re 的变化关系

(2) 结构控制参数

1) 质量比 m^*

质量比 m^* 是单位长度固体结构的质量与单位长度固体结构所排出的流体质量的比值, 表达式为式 (2.4)。其中 m 是单位长度的结构质量, 单位是 Kg 。在 VIV 过程中, 质量比 m^* 是结构响应振幅的重要影响因素。

$$m^* = \frac{4m}{\pi\rho D^2} \quad (2.4)$$

2) 长径比 λ

长径比 λ 是表示结构整体柔性程度的无量纲参数，表达式为式 (2.5)。其中 L 是结构特征长度，单位是 m。长径比 λ 可以侧面反映柔性结构多模态振动特性之间的竞争关系。

$$\lambda = \frac{L}{D} \quad (2.5)$$

3) 阻尼比 ζ

阻尼比 ζ 是表示结构单振动周期内所消耗的能量大小的无量纲参数，表达式为式 (2.6)。其中 m 是结构质量，单位是 Kg； k 是结构刚度，单位是 N/m； c 是结构阻尼，单位是 N/(m/s)。结构阻尼比 ζ 一般通过自由衰减法获得。

$$\zeta = \frac{c}{2\sqrt{km}} \quad (2.6)$$

(3) 流固耦合控制参数

1) 绕流升力 F_l 和绕流阻力 F_d

参考 Van der Pol 尾流振子模型，结构所受的外部激励包括两部分：横流方向上的绕流升力 F_l 和顺流方向上的绕流阻力 F_d ，两个绕流力是流场运动和结构振动的重要条件。表达式分别为式 (2.7) 和式 (2.8)，其中 C_l 是绕流升力系数， C_d 是绕流阻力系数。

$$F_l = \frac{\rho U^2 D C_l}{2} \quad (2.7)$$

$$F_d = \frac{\rho U^2 D C_d}{2} \quad (2.8)$$

2) 约化速度 V_r

研究 VIV 相关规律时，经常将约化速度 V_r 作为自变量，将结构振动响应作为因变量。约化速度 V_r 的表达式为式 (2.9)，其中 f_n 是结构湿模态下的固有频率，单位是 Hz。

$$V_r = \frac{U}{f_n D} \quad (2.9)$$

2.2.2 基本控制方程

VIV 的基本控制方程共分为三类：**流体的质量守恒方程、流体的动量守恒方程以及固体结构的动力学方程**。若流场状态已经呈湍流，则需将湍流模型的基本控制方程考虑进来，本文重点关注**层流**状态下的基本控制方程。**研究 VIV 相关问题时，暂不考虑温度对于流体性质的影响**，因此基本控制方程中不需要添加流

体的能量守恒方程。固体结构的二维振动方程是基于 Van der Pol 尾流振子模型推导得出的，共包括横流向和顺流向两个方向。

(1) 流体的质量守恒方程

质量守恒方程也被称为连续性微分方程 (Continuity differential equation)。流体质量守恒方程的矢量形式为式 (2.10)，其中 t 代表时间， ρ 代表流体的密度， \mathbf{u} 代表流场质点的速度矢量。其中 div 是特殊的矢量符号，可以将其展开为式 (2.11)， u 、 v 和 w 分别为流场质点在 x 、 y 和 z 方向上的速度。

$$\frac{\partial \rho}{\partial t} + div(\rho \mathbf{u}) = 0 \quad (2.10)$$

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0 \quad (2.11)$$

(2) 流体的动量守恒方程

动量守恒方程也被称为运动微分方程 (Motion differential equation)。流体质点在 x 、 y 以及 z 方向上的动量守恒方程如式 (2.12) 至式 (2.14) 所示，其中 p 代表流场质点所受的压力， f_x 、 f_y 以及 f_z 代表流场质点所受的体积力， $\tau_{ij}(i, j=x, y, z)$ 代表流场质点表面所受的黏性应力 τ 在不同方向上的分量。

$$\frac{\partial \rho u}{\partial t} + \frac{\partial \rho u u}{\partial x} + \frac{\partial \rho u v}{\partial y} + \frac{\partial \rho u w}{\partial z} + \frac{\partial p}{\partial x} - \frac{\partial \tau_{xx}}{\partial x} - \frac{\partial \tau_{yx}}{\partial y} - \frac{\partial \tau_{zx}}{\partial z} = f_x \quad (2.12)$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial \rho v u}{\partial x} + \frac{\partial \rho v v}{\partial y} + \frac{\partial \rho v w}{\partial z} + \frac{\partial p}{\partial y} - \frac{\partial \tau_{xy}}{\partial x} - \frac{\partial \tau_{yy}}{\partial y} - \frac{\partial \tau_{zy}}{\partial z} = f_y \quad (2.13)$$

$$\frac{\partial \rho w}{\partial t} + \frac{\partial \rho w u}{\partial x} + \frac{\partial \rho w v}{\partial y} + \frac{\partial \rho w w}{\partial z} + \frac{\partial p}{\partial z} - \frac{\partial \tau_{xz}}{\partial x} - \frac{\partial \tau_{yz}}{\partial y} - \frac{\partial \tau_{zz}}{\partial z} = f_z \quad (2.14)$$

牛顿流体质点所受的黏性应力 τ 同剪切变形速率呈线性比例关系，其中不同方向的黏性应力 τ 可以表示为式 (2.15) 至式 (2.20)， μ 代表动力粘度系数， λ 代表第二黏度系数，一般取值为： $\lambda=-2/3$ 。

$$\tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \quad (2.15)$$

$$\tau_{yy} = 2\mu \frac{\partial v}{\partial y} + \lambda \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \quad (2.16)$$

$$\tau_{zz} = 2\mu \frac{\partial w}{\partial z} + \lambda \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \quad (2.17)$$

$$\tau_{xy} = \tau_{yx} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \quad (2.18)$$

$$\tau_{xz} = \tau_{zx} = \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \quad (2.19)$$

$$\tau_{yz} = \tau_{zy} = \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \quad (2.20)$$

(3) 结构二维双自由度振动方程

基于传统的 Van der Pol 尾流振子模型，建立结构二维双自由度弹性振动模型。顺流向上的振动方程可以表示为式 (2.21)，横流向上的振动方程可以表示为式 (2.22)。其中 x 代表结构的顺流向振动位移， y 代表结构的横流向振动位移， t 代表时间。

$$m \frac{\partial^2 x}{\partial t^2} + c \frac{\partial x}{\partial t} + kx = F_d \quad (2.21)$$

$$m \frac{\partial^2 y}{\partial t^2} + c \frac{\partial y}{\partial t} + ky = F_l \quad (2.22)$$

结合式 (2.6) 至式 (2.8)，可以将式 (2.21) 和式 (2.22) 转化为式 (2.23) 和式 (2.24)。其中 ω_0 代表结构固有频率， ξ 代表结构阻尼比。

$$\frac{\partial^2 x}{\partial t^2} + 2\xi\omega_0 \frac{\partial x}{\partial t} + \omega_0^2 x = \frac{\rho C_d U^2 D}{2m} \quad (2.23)$$

$$\frac{\partial^2 y}{\partial t^2} + 2\xi\omega_0 \frac{\partial y}{\partial t} + \omega_0^2 y = \frac{\rho C_l U^2 D}{2m} \quad (2.24)$$

2.2.3 动网格技术

与采用拉格朗日网格的固体有限元计算不同，流体有限元计算需要采用欧拉网格，这意味着在数值模拟中网格系统需要记录固定位置的物理量变化。因此在 VIV 数值模拟中，需要通过动网格技术更新网格来产生节点运动。动网格技术的最大挑战是避免网格更新后产生负体积网格。目前动网格技术中网格更新的方法主要有三类：网格光滑方法^[81] (Mesh smoothing method)、网格重构方法^[82] (Mesh reconstruction method) 以及动态层方法^[83] (Dynamic layer method)。

(1) 网格光滑方法

如图 2.4 所示，网格光滑方法主要包括两类：弹簧光滑和扩散光滑。其中弹簧光滑分为经典弹簧光滑、边界层光滑以及拉普拉斯光滑，扩散光滑分为基于壁

面距离式和基于单元体积式。

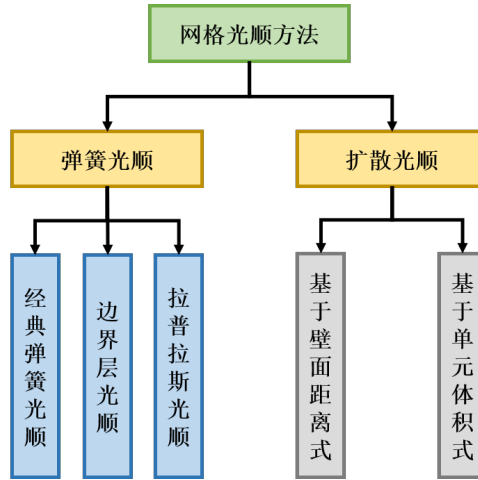


图 2.4 网格光滑方法分类

弹簧光滑需要重点关注四个参数：弹簧常数因子、边界节点松弛、收敛精度以及迭代次数。弹簧常数因子表示弹簧的刚性程度，表达式为式(2.25)，其中 m 、 n 是连接弹簧的两个节点， x_m 、 x_n 是两个节点在网格系统中的坐标位置。边界节点松弛参数的取值区间为 $(0, 1)$ ，这是网格更新时需要重点关注的参数，其中 0 代表网格节点的位置保持不变，1 代表无节点松弛处理。收敛精度是所求的流体运动控制方程所需的求解精度。迭代次数同样服务于收敛精度，网格节点 m 与 n 之间位移的迭代计算过程如式(2.26)所示，其中 Δx_m 代表节点 m 的位移， Δx_n 代表节点 n 的位移， a_m 代表与节点 m 相邻的网格节点总数， b 代表迭代次数。

$$k_{m,n} = \frac{1}{|x_m - x_n|} \quad (2.25)$$

$$\Delta x_m^{b+1} = \frac{\sum_n^{a_m} k_{m,n} \Delta x_n^b}{\sum_n^{a_m} k_{m,n}} \quad (2.26)$$

扩散光滑是基于扩散方程的求解结果来计算所求节点的位移，扩散方程可以表示为式(2.27)，其中 \vec{u} 代表网格系统的更新速度， γ 代表扩散系数。基于壁面距离式和基于单元体积式的扩散系数 γ 均需要通过扩散参数 α 计算得出，计算方程分别为式(2.28)和式(2.29)。其中 d 代表壁面正则距离， V 代表单元正则体积。

$$\nabla \cdot (\gamma \nabla \vec{u}) = 0 \quad (2.27)$$

$$\gamma = \frac{1}{d^\alpha} \quad (2.28)$$

$$\gamma = \frac{1}{V^\alpha} \quad (2.29)$$

网格光顺方法中，弹簧光顺和扩散光顺适用于不同的网格域，具体如表 2.2 所示。

表 2.2 弹簧光顺和扩散光顺的适用性

方法	适用性	
弹簧光顺	四面体网格域/三角形网格域	适用于任何变形及运动
	非四面体网格域/三角形网格域	运动边界集中于某一具体方向 主要顺着运动边界的垂直方向运动
扩散光顺	多面体网格域	适用于任何变形及运动

(2) 网格重构方法

动网格技术中，当系统更新后的网格超过规定的偏斜度或尺寸时，需要全面域或局部进行网格重构，直至新网格达到标准质量。网格重构方法主要分为六类：局部单元重构、局部面重构、区域重构、面域重构、cut-cell 域重构以及 2.5D 重构，每类重构方法的适用性如表 2.3 所示。

表 2.3 六类网格重构方法适用性

方法	适用性
局部单元重构	只适用于三角形网格域或四面体网格域
局部面重构	
区域重构	将其他网格类型转化为三角形网格或四面体网格
面域重构	2D 模型中适用于三角形网格域，3D 模型适用于四面体网格域
cut-cell 域重构	适用于所有网格类型
2.5D 重构	只适用于六面体网格域或棱柱形网格域

(3) 动态层方法

动态层方法主要分为两类：拆分网格和合并网格。这两类方法均适用于四面体网格域、六面体网格域以及棱柱形网格域。网格发生“拆分”的条件是：当网格边界开始运动时，紧邻运动边界的网格在运动边界法方向上的尺寸 a 大于规定值。同理，网格发生“合并”的条件是：当网格边界开始运动时，紧邻运动边界的网格在运动边界法方向上的尺寸 a 小于规定值。动态层方法更新网格的比对条件如图 2.5 所示。

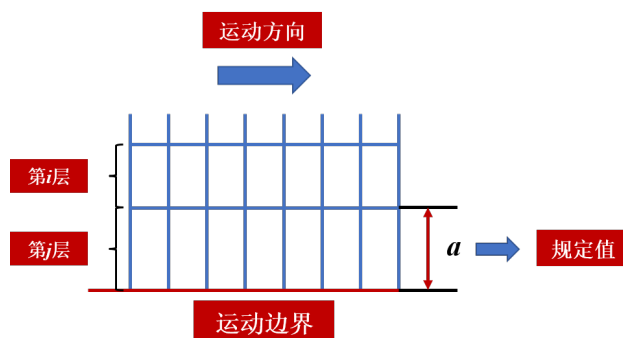


图 2.5 “拆分”与“合并”网格比对条件

本文将基于网格光滑方法和网格重构方法完成 VIV 的 CFD 数值模拟。

2.3 深度神经网络基本理论

2.3.1 深度神经网络简介

图 2.6 展示了 DNN 与 PINN 之间的关系。DNN 是 PINN 的基础网络拓扑结构。DNN 是数据驱动的神经网络学习，PINN 是数据以及物理信息双重驱动的神经网络学习。

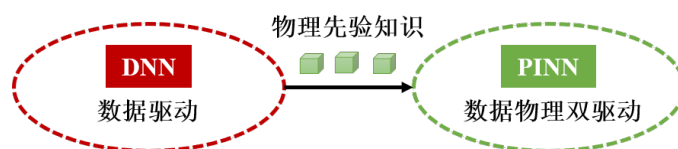


图 2.6 DNN 与 PINN 关系示意图

DNN 的发展主要分为三个阶段。第一阶段主要以阈值逻辑单元 (Threshold logic unit, TLU)、一代感知器 (Perceptron) 以及自适应线性神经元 (Adaptive linear neuron, ALN) 为主, 这一阶段 DNN 的学习能力在根本上受到了限制^[84]。随着反向传播算法的逐渐成熟, 第二代 DNN 开始通过获取超参数导数来进行学习。更高级的神经网络开始出现, 比如多层反向传播神经网络、长短记忆神经网络 (Long short-term memory, LSTM) 以及循环神经网络 (Recurrent neural networks, RNNs) 等^[85]。随着 GPU 和储存技术的进步, DNN 的发展进入到最成熟的第三阶段, 依据目前 DNN 的主要网络结构, 可以将 DNN 分为以下几类: 有限玻尔兹曼机^[86] (Restricted boltzmann machines, RBMs)、深度信念网络^[87] (Deep belief networks, DBNs)、全连接神经网络^[88] (Fully-connected neural networks, FCNNs)、卷积神经网络^[89] (Convolutional neural networks, CNNs)、概率神经网络^[90]

(Probabilistic neural networks, PNNs)、模糊神经网络^[91] (Fuzzy neural networks, FNNs) 以及生成对抗网络^[92] (Generative adversarial networks, GANs), 如图 2.7 所示。

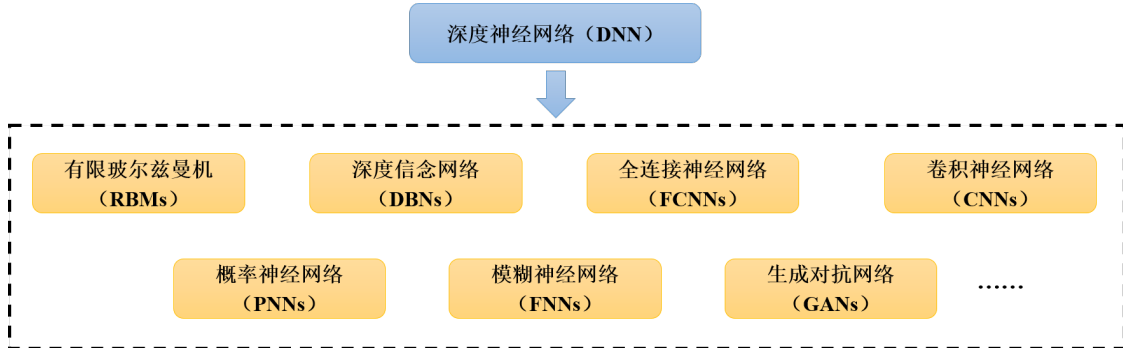


图 2.7 第三阶段 DNN 网络结构形式分类

2.3.2 反向传播算法

DNN 的结构体系是若干映射变换和非线性激活函数的重复组合。如图 2.8 所示, 假设 DNN 共含有 $L+1$ 个网络层, 同时将隐式解表示为 u_N 。第一个网络层 (t, x) 是输入层, 最后一个网络层 $u_N(t, x)$ 是输出层。输入层与输出层之间剩余的 $L-1$ 个网络层被称为隐藏层。常用的非线性激活函数也被应用于 DNN 中, 比如 sigmoid 函数、修正线性单元以及双曲正切函数等。除网络输入层的神经元外, DNN 中剩余的神经元均内置响应的偏置系数。在相邻的两层中, 神经元的连接关系被表示为权重矩阵。在 DNN 中, 第 $l-1$ 层和第 l 层之间的联系可以表示为式 (2.30)。

$$z_j^l = \sum_k^{N_{l-1}} \sigma_{l-1}(z_k^{l-1}) \omega_{jk}^l + b_j^l \quad (2.30)$$

其中 z_k^{l-1} 代表第 $l-1$ 个网络层中第 k 个神经元的赋值, N_{l-1} 代表第 $l-1$ 个网络层中神经元的数量, b_j^l 代表第 l 个网络层中第 j 个神经元的偏置系数, ω_{jk}^l 代表第 $l-1$ 个网络层中第 k 个神经元和第 l 个网络层中第 j 个神经元之间的权重, σ_{l-1} 代表第 $l-1$ 个网络层的非线性激活函数。

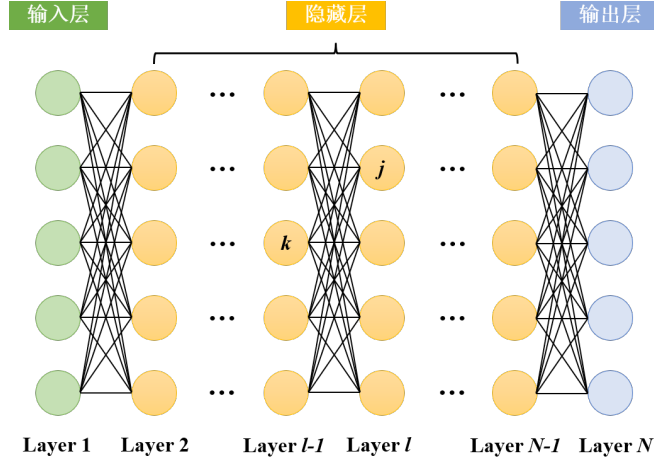


图 2.8 DNN 结构体系示意图

DNN 中的反向传播算法 (Back propagation algorithm, BPA) 是基于梯度下降法 (Gradient descent method, GDM) 优化损失函数至最小值, 不断更新网络隐藏层和网络输出层所对应的权重矩阵和偏置系数, 从而使样本集完成映射目标。BPA 共包含三个标准步骤: 前向传播网络信号、度量反向传播误差以及更新迭代网络权重和偏置。

(1) 前向传播网络信号

为了更清楚地阐述神经网络信号前向传播的过程, 以三层架构的 DNN 为例说明。其中网络输入层和网络隐藏层各含有 4 个神经元, 网络输出层包含 2 个神经元。依据式 (2.30), 网络输入层神经元的赋值分别是 z_1^1 、 z_2^1 、 z_3^1 以及 z_4^1 , 网络隐藏层神经元赋值的计算过程表示为式 (2.31) 至式 (2.34)。

$$z_1^2 = \sigma_1(\omega_{11}^2 z_1^1 + \omega_{12}^2 z_2^1 + \omega_{13}^2 z_3^1 + \omega_{14}^2 z_4^1 + b_1^2) \quad (2.31)$$

$$z_2^2 = \sigma_1(\omega_{21}^2 z_1^1 + \omega_{22}^2 z_2^1 + \omega_{23}^2 z_3^1 + \omega_{24}^2 z_4^1 + b_2^2) \quad (2.32)$$

$$z_3^2 = \sigma_1(\omega_{31}^2 z_1^1 + \omega_{32}^2 z_2^1 + \omega_{33}^2 z_3^1 + \omega_{34}^2 z_4^1 + b_3^2) \quad (2.33)$$

$$z_4^2 = \sigma_1(\omega_{41}^2 z_1^1 + \omega_{42}^2 z_2^1 + \omega_{43}^2 z_3^1 + \omega_{44}^2 z_4^1 + b_4^2) \quad (2.34)$$

网络输出层的输出变量为 z_1^3 和 z_2^3 , 相应的计算公式为式(2.35)和式(2.36)。

$$z_1^3 = \sigma_2(\omega_{11}^3 z_1^2 + \omega_{12}^3 z_2^2 + \omega_{13}^3 z_3^2 + \omega_{14}^3 z_4^2 + b_1^3) \quad (2.35)$$

$$z_2^3 = \sigma_2(\omega_{21}^3 z_1^2 + \omega_{22}^3 z_2^2 + \omega_{23}^3 z_3^2 + \omega_{24}^3 z_4^2 + b_2^3) \quad (2.36)$$

(2) 度量反向传播误差及更新迭代网络权重及偏置

神经网络反向传播过程中, 以标准差 (Standard deviation) 度量传播误差, 同时基于梯度下降法更新迭代网络权重及偏置。损失函数可以表示为式 (2.37),

其中 x 和 y 是目标映射关系, z^l 和 y 均为维度为 n 的特征向量, $\|A\|_2$ 代表变量 A 的 L2 范数, W 代表 DNN 的权重矩阵, b 代表 DNN 的偏置系数。

$$Loss(W, b, x, y) = \frac{\|z^l - y\|_2^2}{2} \quad (2.37)$$

依据式 (2.30), DNN 的网络输出层 (第 l 层) 也可以表示为式 (2.38)。因此式 (2.37) 可以转化为式 (2.39)。

$$z^l = \sigma(W^l z^{l-1} + b^l) \quad (2.38)$$

$$Loss(W, b, x, y) = \frac{\|\sigma(W^l z^{l-1} + b^l) - y\|_2^2}{2} \quad (2.39)$$

求解权重矩阵 W 和偏置系数 b 的梯度时, 为了简化表示, 先计算损失函数对 z^l 的偏导, 将其定义为 θ^l , 如式 (2.40) 所示。其中 Δ 代表哈达玛 (Hadamard) 积, 对于维度相同的特征向量 $X=(x_1, x_2, x_3, \dots, x_n)^T$ 和 $Y=(y_1, y_2, y_3, \dots, y_n)^T$, $X \Delta Y = (x_1 y_1, x_2 y_2, x_3 y_3, \dots, x_n y_n)^T$ 。

权重矩阵 W 和偏置系数 b 的梯度计算公式分别为式 (2.41) 和式 (2.42)。

$$\theta^l = \frac{\partial Loss(W, b, x, y)}{\partial z^l} = (z^l - y) \Delta \sigma'(z^l) \quad (2.40)$$

$$\frac{\partial Loss(W, b, x, y)}{\partial W^l} = \frac{\partial Loss(W, b, x, y)}{\partial z^l} \frac{\partial z^l}{\partial W^l} = \theta^l (z^{l-1})^T \quad (2.41)$$

$$\frac{\partial Loss(W, b, x, y)}{\partial b^l} = \frac{\partial Loss(W, b, x, y)}{\partial z^l} \frac{\partial z^l}{\partial b^l} = \theta^l \quad (2.42)$$

梯度的求解结果依赖于每层的 θ^l 。每个网络层的 θ^l 将基于数学归纳法进行求解, θ^l 和 θ^{l+1} 的基础关系表示为式 (2.43)。将 θ^l 的递推式依次带入式 (2.41) 和式 (2.42), 即可得到权重矩阵 W 和偏置系数 b 的梯度求解结果, 从而基于梯度下降法对神经网络进行迭代更新。

$$\begin{aligned} \theta^l &= \frac{\partial Loss(W, b, x, y)}{\partial z^l} = \frac{\partial Loss(W, b, x, y)}{\partial z^{l+1}} \frac{\partial z^{l+1}}{\partial z^l} = \theta^{l+1} \frac{\partial z^{l+1}}{\partial z^l} \\ &= \theta^{l+1} \frac{\partial (W^{l+1} \sigma(z^l) + b^{l+1})}{\partial (W^l \sigma(z^{l-1}) + b^l)} = \theta^{l+1} ((W^{l+1})^T \Delta (\underbrace{\sigma'(z^l), \dots, \sigma'(z^l)}_{n_{l+1}})) \\ &= (W^{l+1})^T \theta^{l+1} \Delta \sigma'(z^l) \end{aligned} \quad (2.43)$$

2.3.3 损失函数的构建

构建损失函数 (Loss function) 的目的是度量神经网络模型预报值与真实值

之间的误差。损失函数是结构经验函数的核心成分，结构经验函数 $\hat{\theta}$ 的表达式为式 (2.44)，其中 $Loss(y_i, f(x_i; \theta))$ 表示损失函数， $\lambda\Phi(\theta)$ 表示衡量神经网络模型复杂程度的参数。

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{N} \sum_{i=1}^N Loss(y_i, f(x_i; \theta)) + \lambda\Phi(\theta) \quad (2.44)$$

目前适用于回归类神经网络模型的损失函数有以下几种^[93]：平均绝对误差损失（Mean absolute error, MAE）、均方差损失（Mean squared error, MSE）以及 Huber 损失。

(1) MAE 损失

MAE 损失也被称为 L1 损失，表达式为式 (2.45)。其中 M 表示训练样本的总数目， a_i 表示真实数据， \hat{a}_i 表示预报数据。**MAE 损失服从于拉普拉斯分布 (Laplace distribution)**。因此相对于其他损失函数，MAE 损失对神经网络模型的异常样本点展示出更强的鲁棒性及泛化能力。

$$Loss_{MAE} = \frac{1}{M} \sum_{i=1}^M |a_i - \hat{a}_i| \quad (2.45)$$

(2) MSE 损失

MSE 损失也被称为 L2 损失，表达式为式 (2.46)。其中 M 表示训练样本的总数目， a_i 表示真实数据， \hat{a}_i 表示预报数据。MSE 损失服从于标准高斯分布，即 $\mu=0$ ， $\sigma=1$ 。相对于其他损失函数，MSE 损失的梯度方向对于误差改变的敏感性非常高，因此当使用梯度下降法优化损失函数时，MSE 损失更容易更新迭代至收敛阈值。

$$Loss_{MSE} = \frac{1}{M} \sum_{i=1}^M (a_i - \hat{a}_i)^2 \quad (2.46)$$

(3) Huber 损失

Huber 损失也被称为 Smooth L1 损失，是将 L1 损失和 L2 损失相结合的损失函数，表达式为式 (2.47)。其中 M 表示训练样本的总数目， a_i 表示真实数据， \hat{a}_i 表示预报数据，超参数 δ 代表 L1 损失和 L2 损失的连接位置。 $(-\delta, \delta)$ 区间内，Huber 损失保持可导，同时梯度方向相对稳定。 **$(-\infty, -\delta)$ 和 (δ, ∞) 区间内，Huber 损失也具有较高的鲁棒性，极大地降低了异常样本点对结果的影响。**

$$Loss_{Huber} = \frac{1}{M} \sum_{i=1}^M \Pi_{|a_i - \hat{a}_i| \leq \delta} \frac{(a_i - \hat{a}_i)^2}{2} + \Pi_{|a_i - \hat{a}_i| > \delta} (\delta |a_i - \hat{a}_i| - \frac{\delta^2}{2}) \quad (2.47)$$

2.3.4 激活函数的选择

激活函数通常处于上层神经元输出节点与下层神经元输入节点之间，如图 2.9 所示。激活函数可以将神经网络的样本集规范化至相应区间内，从而实现 DNN 的非线性映射^[94]。通过这种非线性映射，DNN 几乎可以逼近任何函数，大大提高了 DNN 的网络表达能力。目前常用于 DNN 的激活函数有以下几种：Sigmoid 函数、Tanh 函数、ReLU 函数、Leaky-ReLU 函数以及 Sin 函数。

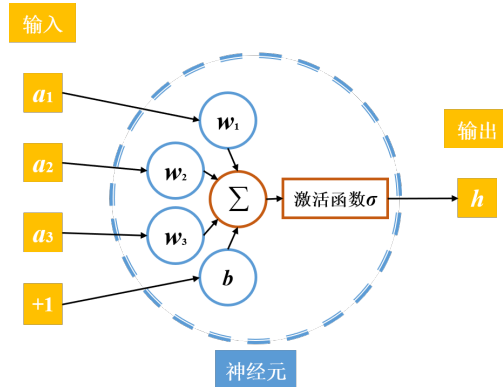


图 2.9 激活函数示意图

(1) Sigmoid 函数

Sigmoid 函数的表达式为式 (2.48)，图 2.10 是 Sigmoid 函数的几何图像。Sigmoid 函数可以将连续的实数集规范化至 $(0, 1)$ 的区间，但是存在有三个不可避免的缺点：一是信号在 DNN 反向传播过程中容易发生梯度消失和梯度爆炸；二是网络输出层不是 0 均值 (Zero-centered)；三是函数本身包含的幂运算增加了更多的计算成本。

$$f(a) = \frac{1}{1 + e^{-a}} \quad (2.48)$$

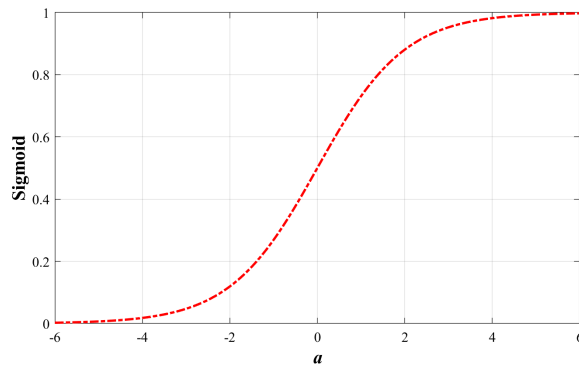


图 2.10 Sigmoid 函数几何图像

(2) Tanh 函数

Tanh 函数的表达式为式 (2.49)，图 2.11 是 Tanh 函数的几何图像。Tanh 函数的全称是 Hyperbolic tangent，其进一步改进了 Sigmoid 函数存在的缺陷。Tanh 函数有效解决了网络输出层非 0 均值的问题，但梯度问题以及计算成本问题仍旧十分棘手。

$$f(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \quad (2.49)$$

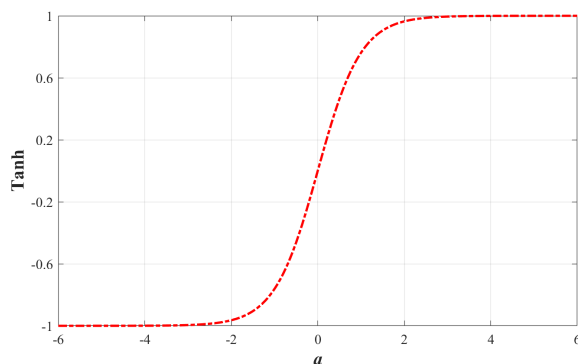


图 2.11 Tanh 函数几何图像

(3) ReLU 函数

ReLU 函数的表达式为式 (2.50)，图 2.12 是 ReLU 函数的几何图像。ReLU 函数的本质是取极大值的函数，是目前使用最广泛的非线性激活函数。ReLU 函数成功解决了梯度消失的问题，同时通过简化网络输入层的判断规则，降低了计算成本，有效提高了 DNN 的收敛速度。但是，ReLU 函数也存在有以下缺点：一是网络输出层是非 0 均值；二是容易出现部分神经元坏死问题，即部分始终未被激活的神经元的超参数无法被更新。

$$f(a) = \max(0, a) \quad (2.50)$$

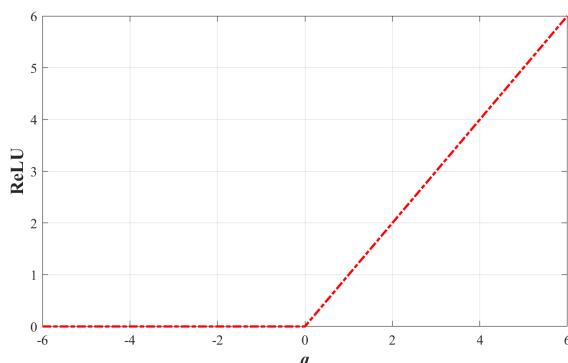


图 2.12 ReLU 函数几何图像

(4) Leaky-ReLU 函数

Leaky-ReLU 函数的表达式为式 (2.51)，图 2.13 是 Leaky-ReLU 函数的几何图像，参数 $\alpha=0.01$ 。Leaky-ReLU 函数在表达式中引入参数 α ，以此来解决 ReLU 函数存在的部分神经元坏死问题，通常 α 的取值是 0.01。Leaky-ReLU 函数是对 ReLU 函数的进一步优化。

$$f(a) = \max(\alpha a, a) \quad (2.51)$$

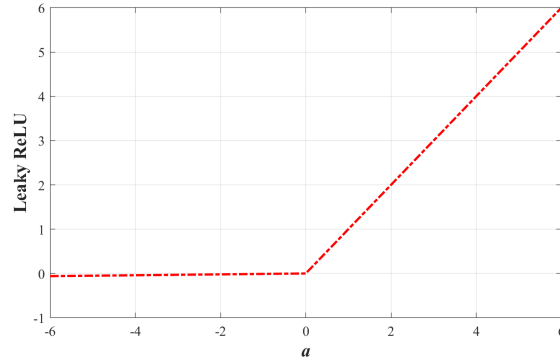


图 2.13 Leaky-ReLU 函数几何图像

(5) Sin 函数

Sin 函数的表达式为式 (2.52)，图 2.14 是 Sin 函数的几何图像。Sin 函数为神经网络引入了周期性，可以将实数轴分割成多个等长区间来进行分类及回归。因此 Sin 函数的 VC 维 (Vapnik-Chervonenkis dimension) 具有无穷大，对函数集的学习能力非常强，但容易造成过拟合。本文训练神经网络模型时所选取的激活函数是 Sin 函数。

$$f(a) = \sin a \quad (2.52)$$

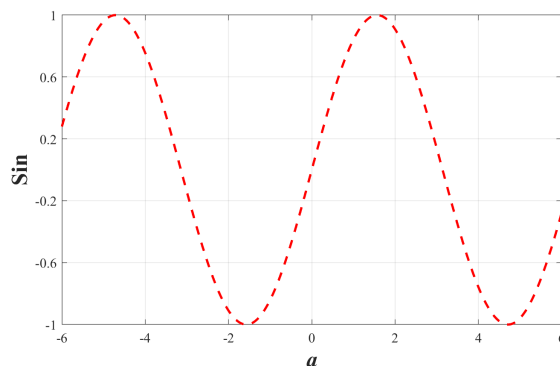


图 2.14 Sin 函数几何图像

2.3.5 神经网络训练参数

训练神经网络时，需要重点关注三个训练参数：迭代次数 (Epoch)、批次大小 (Batch size) 和学习速率 (Learning rate)。合理的训练参数设置是神经网络模

型获得高精度预报结果的关键^[95]。

(1) 迭代次数

Epoch 被称为“一代训练”，即神经网络模型对样本集进行一次完整训练。一个完整 Epoch 中，所有样本数据需要在 DNN 中分别进行一次“正向传播”和“反向传播”。Epoch 决定了神经网络的迭代次数，这直接影响到样本集的最终拟合情况。图 2.15 分别为样本集过拟合 (Overfitting)、正确拟合 (Optimum) 以及欠拟合 (Underfitting) 的情况。

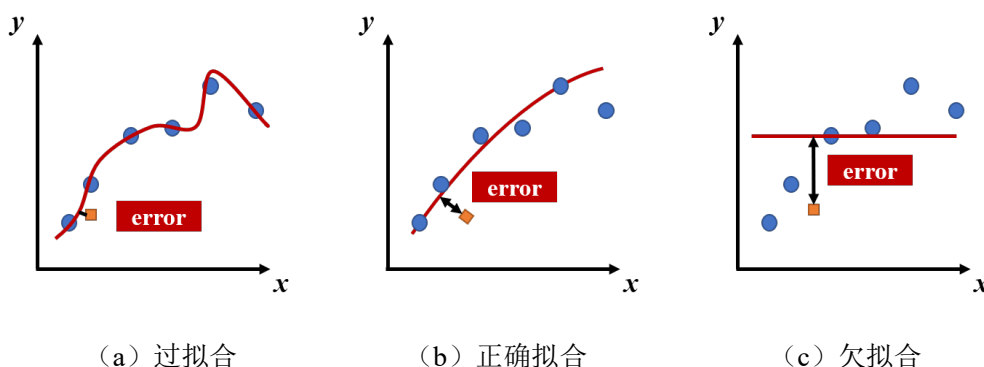


图 2.15 样本数据集拟合情况

(2) 批次大小

批次大小是指进行一次网络训练所需的样本数量，正确的 Batch size 可以最优化计算效率与内存容量之间的关系。Batch size 的选择一般有三种情况：全部批次 (Batch)、迷你批次 (Mini-batch) 以及随机批次 (Stochastic)，具体如表 2.4 所示。

表 2.4 Batch size 的三类取值情况

Batch size 类型	特点
全部批次	适用于训练样本集总数比较小的情况；可以准确地代表训练样本的整体分布，快速确定正确的梯度方向
迷你批次	以迷你批次的数量维度将所有的训练样本持续输入进 DNN 中；迭代过程中的损失函数是总体样本的平均值
随机批次	随机批次意味着 batch size=1；随机样本包含多个梯度方向，修正时难以确定正确方向，容易导致计算无法收敛

总体来看，当 Batch size 的取值过小时，难以确定正确的梯度方向，导致计算难以收敛。当增大 Batch size 的取值时，在内存容量并行式的前提下，可以极大地提高确定梯度方向的准确性，减小网络训练过程中的损失函数的振动幅值。但是，当 Batch size 的取值过大时，需要增加 Epoch 的次数来匹配训练，从而增加内存负担，影响计算效率。因此在训练过程中，需要寻找合适的 Batch size 来

平衡计算效率和内存容量的矛盾。

(3) 学习速率

学习速率也被称为“学习步长”，是用来描述优化过程中的误差尺度。当学习速率较大时，网络模型更新的速度更快，但容易出现损失函数爆炸或振荡等问题。当学习速率较小时，网络模型更新的速度比较慢，收敛时间过长，且容易出现过拟合的问题。在网络训练过程中，学习速率往往不是固定值，而是基于衰减机制逐渐减小，表 2.5 是目前比较常用的学习速率衰减机制，分别是轮数衰减 (Step decay)、指数衰减 (Exponential decay) 以及分数衰减 (1/t decay)。

表 2.5 常用的学习速率衰减机制

衰减机制类型	具体方法
轮数衰减	规定每经过 N 轮后，学习速率减半
指数衰减	学习速率基于训练轮数的增长指数进行插值递减
分数衰减	衰减公式为： $lr_i = \frac{lr_0}{1+kN}$ ，其中 k 是控制减缓幅度， N 是训练轮数

2.4 物理信息神经网络基本理论

2.4.1 物理信息神经网络简介

在数据驱动 DNN 的基础上，PINN 将代表客观物理规律的先验知识（比如流固耦合问题中的 Navier-Stokes 方程、电磁学问题中的 Maxwell 方程等）嵌入网络结构中。基于数据及物理信息的双重驱动，PINN 可以代替传统的离散数值计算方法，高效率地逼近 NPDE 的真实解，处理非线性正反问题^[96]。

神经网络学习以一个特定映射问题为出发点，把已知数据信息输入学习管道中，来完成目标回归任务：将输入 X 映射至输出 Y 。学习管道主要由四部分组成，分别是训练样本、假设集、算法逻辑和回归任务。传统 DNN 仅以样本数据作为已知信息，但是 PINN 的学习管道依托于一个混合信息源，包括两部分：样本数据和先验知识，如图 2.16 所示。先验知识独立于学习任务存在，以逻辑规则、仿真结果、知识图、初始边界条件等形式提供，通常表示为显示方程。先验知识作为额外存在的约束条件，一定程度上避免了样本数据的浪费。

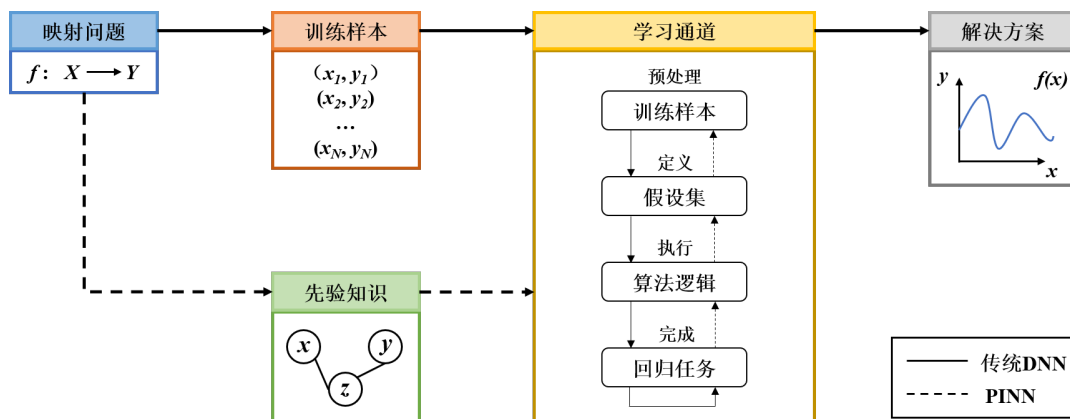


图 2.16 传统 DNN 与 PINN 逻辑框架对比

嵌入网络结构的先验知识往往具有不同的表达形式，为了保证 PINN 的合理性和可延展性，不同形式的 NPDE 需要与特定的网络结构设计相匹配，以适应“黑盒工具”的计算逻辑，保证网络模型的训练精度。随着应用领域的多样化，PINN 也演变出多种网络结构形式，主要包括非定域型^[97]（Nonlocal PINNs, nPINNs）、分式型^[98]（Fractional PINNs, fPINNs）、并行式^[99]（Parareal PINN, PPINN）、守恒式^[100]（Conservative PINNs, cPINNs）、分布式^[101]（Distributed PINN, DPINN）、贝叶斯式^[102]（Bayesian PINNs, BPINNs）以及变分式^[103]（Variational PINNs, vPINNs）等。PINN 的网络结构形式分类如图 2.17 所示。

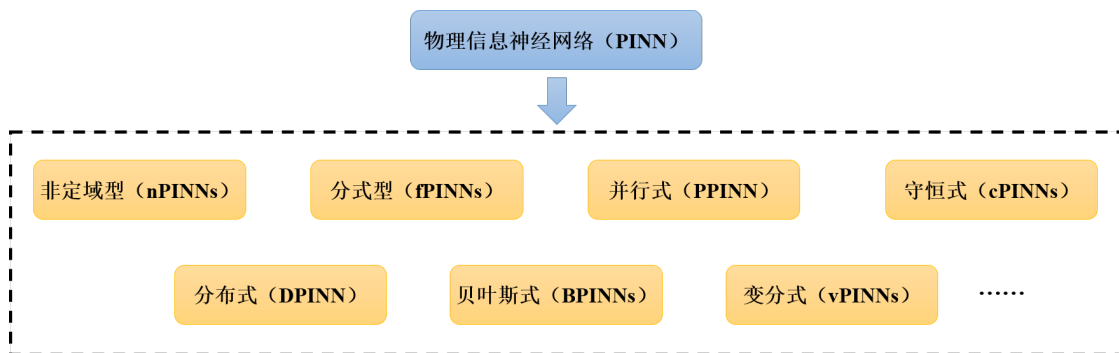


图 2.17 PINN 网络结构形式分类

在处理非线性问题上，PINN 可以作为求解 NPDE 的通用函数逼近器，而不需要进行任何预先假设、线性化或局部时域调整。这很大程度依赖于算法微分（Algorithmic differentiation, AD）技术的进步^[104]，可以自动求解网络输出层参数对于网络输入层参数的偏导，目前在 Tensorflow 中可以直接调用相关工具包用于计算。考虑到神经网络是一个复杂的复合函数，AD 应该使用链式法则来计算偏导数，向前传递所有变量的数值，向后传递偏导数的数值。

2.4.2 非线性偏微分方程正问题

正问题是通过 PINN 计算被 λ 参数化的 NPDE 的隐式解析解 $u(x)$ [105-106]，这里 $x=(x_1, x_2, x_3, \dots, x_m)$ 的定义域 Ω 属于实数集 R^D 的一个子集，并包含合适的初始条件 (Initial conditions, IC) 与边界条件 (Boundary conditions, BC)，如式 (2.53) 至式 (2.55) 所示。其中 I 是初始算子， f 表示初始分布； B 是边界算子， g 表示边界集合。将微分算子定义为 L_p 。

BC 类型主要包括狄利克雷 (Dirichlet)、诺伊曼 (Neumann)、罗宾 (Robin) 以及周期性 (Periodic) 等。IC 是时空域上一类特殊的狄利克雷边界条件，通常被认为自动满足计算要求。

$$f(x; \frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2}, \dots, \frac{\partial u}{\partial x_m}; \frac{\partial^2 u}{\partial x_1 \partial x_1}, \frac{\partial^2 u}{\partial x_1 \partial x_2}, \dots, \frac{\partial^2 u}{\partial x_1 \partial x_m}, \dots; \lambda) = 0, x \in \Omega \quad (2.53)$$

$$Iu = f, x \in \Omega \quad (2.54)$$

$$Bu = g, x \in \partial\Omega \quad (2.55)$$

求解正问题的算法逻辑如表 2.6 所示。首先，构造数据驱动 DNN，用含未知参数 θ 的网络输出 $\tilde{u}(x; \theta)$ 代替 NPDE 的隐藏解 $u(x)$ ，其中 θ 是包含权重矩阵和偏置系数的向量。再次，为样本数据、偏微分方程 (Partial differential equation, PDE) 以及 BC/IC 指定训练集，分别为 Γ_d 、 Γ_f 和 Γ_b ，将其统称为 Γ 。然后，构建损失函数 $Loss(\theta; \Gamma)$ ，从而约束 PINN 的输出以满足样本数据的回归映射以及先验知识规律。损失函数 $Loss$ 包含两部分：数据驱动损失函数 $Loss_{data}$ 和物理驱动损失函数 $Loss_{phy}$ 。最后，训练损失函数 $Loss(\theta; \Gamma)$ 至指定阈值，从而寻得最优参数 $\hat{\theta}$ 。

表 2.6 求解正问题的算法逻辑

步骤	算法内容
1	建立含未知参数 θ 的 PINN 模型
2	指定样本数据、PDE 和 BC/IC 的训练集 Γ_d 、 Γ_f 和 Γ_b
3	构造包含数据驱动和物理驱动的损失函数 $Loss(\theta; \Gamma)$
4	最小化损失函数 $Loss(\theta; \Gamma)$

将以上算法逻辑可视化为图 2.18，微分算子 L_p 通过 AD 技术将物理信息嵌入 DNN 中，形成 PINN 框架结构。PINN 的损失函数可以表示为式 (2.56) 至式 (2.57)，其中 ω_d 、 ω_f 和 ω_b 分别表示样本数据、PDE 和 BC/IC 在损失函数中所占权重。

$$Loss(\theta; \Gamma) = Loss_{data} + Loss_{phy} \quad (2.56)$$

$$Loss_{data} = \omega_d Loss(\theta; \Gamma_d) \quad (2.57)$$

$$Loss_{phy} = \omega_f Loss(\theta; \Gamma_f) + \omega_b Loss(\theta; \Gamma_b) \quad (2.58)$$

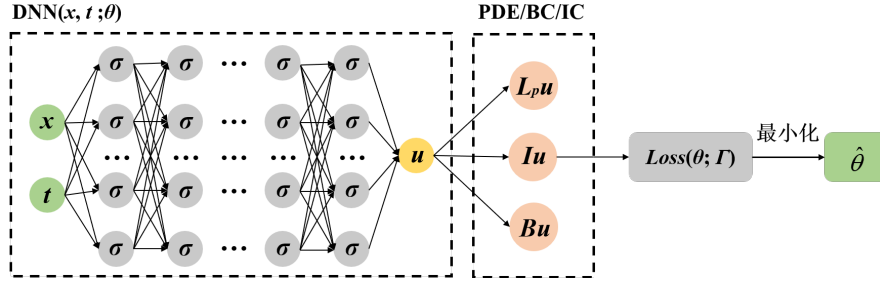


图 2.18 正问题求解框架

与传统数值计算方法不同，PINN 求解 NPDE 正问题的本质是在求解非凸优化问题（Non-convex optimization problem, NOP）。非凸优化问题的解通常没有唯一性，因此 PINN 所求的隐式解也无法保证是唯一解。基于这样的情况，为了获得更为精确的隐式解，需要根据 NPDE 的形式以及应用问题等，不断调整 PINN 的网络结构及训练参数等，以确保获得最优方案。

2.4.3 非线性偏微分方程反问题

在反问题中，存在着许多未知参数 λ 。因此需要利用除 BC/IC 之外的额外信息来寻求最佳参数 $\hat{\lambda}$ 。对未知参数 λ 的识别可以表示为以下最优化问题^[107-108]，如式 (2.59) 所示，其中 m 为未知参数的数量。

$$\{\lambda_1, \lambda_2, \dots, \lambda_m\} = \arg \min_{\lambda_1, \dots, \lambda_m} Loss(\theta, \lambda; \Gamma) \quad (2.59)$$

反问题仍旧基于正问题的物理驱动深度拓扑结构进行训练。唯一不同的是，待识别的未知参数 λ 与神经网络的权重矩阵 W 和偏置系数 b 同时进行迭代更新，损失函数可以写作 $Loss(W, b, \lambda)$ 。参考式 (2.40)，为了简化计算，将该损失函数对网络输出变量 z^l 的偏导记作 θ^l ，如式 (2.60) 所示。

$$\theta^l = \frac{\partial Loss(W, b, \lambda)}{\partial z^l} \quad (2.60)$$

神经信号反向传播过程中，将依次求解对权重矩阵 W 、偏置系数 b 以及未知参数 λ 的梯度，从而基于梯度下降算法对参数进行迭代更新。这些参数的梯度求解公式依次为式 (2.61) 至式 (2.63)。

$$\frac{\partial Loss(W, b, \lambda)}{\partial W^l} = \frac{\partial Loss(W, b, \lambda)}{\partial z^l} \frac{\partial z^l}{\partial W^l} = \theta^l \frac{\partial z^l}{\partial W^l} \quad (2.61)$$

$$\frac{\partial \text{Loss}(W, b, \lambda)}{\partial b^l} = \frac{\partial \text{Loss}(W, b, \lambda)}{\partial z^l} \frac{\partial z^l}{\partial b^l} = \theta^l \frac{\partial z^l}{\partial b^l} \quad (2.62)$$

$$\frac{\partial \text{Loss}(W, b, \lambda)}{\partial \lambda} = \frac{\partial \text{Loss}(W, b, \lambda)}{\partial z^l} \frac{\partial z^l}{\partial \lambda} = \theta^l \frac{\partial z^l}{\partial \lambda} \quad (2.63)$$

由上述推导结果可得，这些参数的梯度求解结果均与 θ^l 密切相关。 θ^l 与 θ^{l+1} 的递推式可以由数学归纳法推导得出，从而解得每个网络层的 θ^l 赋值。将其带入式 (2.61) 至式 (2.63)，即可获得权重矩阵 W 、偏置系数 b 以及未知参数 λ 的梯度求解结果，进而对神经网络进行迭代更新。将 $\text{Loss}(\theta, \lambda; \Gamma)$ 最小化至指定阈值后，即可输出所识别的最佳参数 $\hat{\lambda}$ 。

2.4.4 局部自适应激活函数

DNN 常用的 Sigmoid、Tanh 以及 ReLU 等激活函数应用于普通线性回归模型时，可以决定特定神经元是否应该激活，将非线性特性引入神经网络中。但面对较为复杂的 PINN 模型时，解决问题能力有限，容易发生梯度消失或梯度爆炸等问题^[109]。因此局部自适应激活函数（Locally adaptive activation function）被提出，以适应 PINN 的训练性能。

局部自适应激活函数是在每个隐藏层和神经元分别引入一个参数 k 来实现的，然后使用随机梯度下降算法的变体对其进行优化^[110]。为了进一步提高 PINN 的收敛速度，在损失函数中加入一个基于激活斜率的斜率恢复项。通过隐式将条件矩阵乘到基方法的梯度上，而不需要任何显式计算条件矩阵和矩阵向量积，从而降低了训练成本。

通过更新参数 k 以及神经网络的偏置和权重，来寻找神经网络损失函数 Loss 的最小值，这是局部自适应激活函数的优化问题。可伸缩参数 k 的最优解为 \hat{k} ，如式 (2.64) 所示。可伸缩参数 k 更新迭代的过程可以表示为式 (2.65)，其中 m 代表更新迭代的次数， η_l 代表神经网络的学习速率。

$$\hat{k} = \arg \min_{k \in \mathbb{R}^+} (\text{Loss}(k)) \quad (2.64)$$

$$k^{m+1} = k^m - \eta_l \nabla_{k^m} \text{Loss}^m(k) \quad (2.65)$$

在数学计算中，参数 k 可以改变函数的斜率，这是局部自适应激活函数在 PINN 模型训练中发挥作用的重要原因。图 2.19 至图 2.22 分别给出了参数 k 取不同值时 Sigmoid、Tanh、ReLU 以及 Leaky-ReLU 局部自适应激活函数的图形，由此可以看到函数斜率随参数 k 的变化情况。这些局部自适应激活函数对应的表达式依次为式 (2.66) 至式 (2.69)。

(1) Sigmoid 局部自适应激活函数

$$f(x)_{\text{Sigmoid}} = \frac{1}{1 + e^{-kx}} \quad (2.66)$$

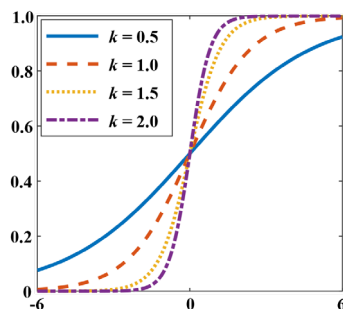


图 2.19 Sigmoid 局部自适应激活函数的几何图像

(2) Tanh 局部自适应激活函数

$$f(x)_{\text{Tanh}} = \frac{e^{kx} - e^{-kx}}{e^{kx} + e^{-kx}} \quad (2.67)$$

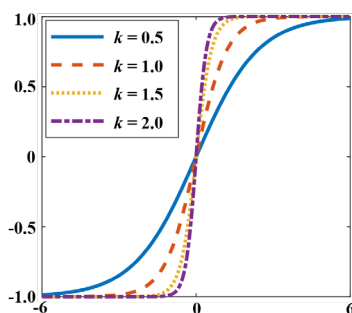


图 2.20 Tanh 局部自适应激活函数的几何图像

(3) ReLU 局部自适应激活函数

$$f(x)_{\text{ReLU}} = \max(0, kx) \quad (2.68)$$

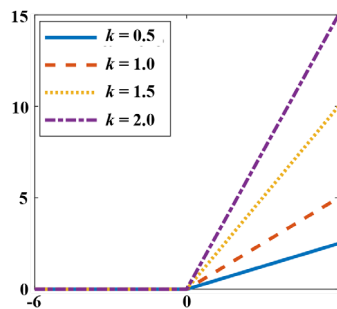


图 2.21 ReLU 局部自适应激活函数的几何图像

(4) Leaky-ReLU 局部自适应激活函数

$$f(x)_{\text{ReLU}} = \max(0, kx) \quad (2.69)$$

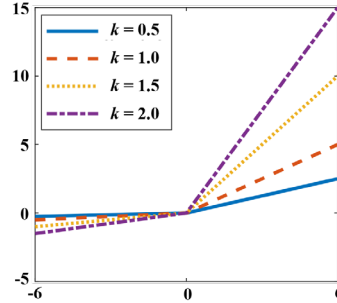


图 2.22 Leaky-ReLU 局部自适应激活函数的几何图像

2.4.5 训练过程优化

在 PINN 模型训练过程中，物理信息约束往往作为软惩罚，充当经验风险损失的领域特定正则化器。然而，对于某些呈现陡峭梯度隐式解的 NPDE，不平衡的反向传播梯度容易导致梯度病理^[111]。

一个好的残差点集 Γ 可以平衡反向传播梯度。因此，**基于残差的自适应细化 (Residual-based adaptive refinement, RAR)** 方法被提出来改善残差点集 Γ 的分布^[112]。**RAR 方法的主要思想是在 NPDE 残差较大的地方添加更多的残差点**，直到平均残差 ε_r 小于规定的残差阈值 ε_0 ，从而改善 PINN 的训练过程。平均残差 ε_r 可以表示为式 (2.70)，这里 V 是 Ω 的总体积。表 2.7 是 RAR 方法的算法逻辑。

$$\varepsilon_r = \frac{1}{V} \int_{\Omega} \left\| f(x; \frac{\partial \hat{u}}{\partial x_1}, \dots, \frac{\partial \hat{u}}{\partial x_m}; \frac{\partial^2 \hat{u}}{\partial x_1 \partial x_1}, \dots, \frac{\partial^2 \hat{u}}{\partial x_1 \partial x_m}; \dots; \lambda) \right\| dx \quad (2.70)$$

表 2.7 RAR 方法的算法逻辑

步骤	算法内容
1	选择初始残差点集 Γ ，对 PINN 进行有限次数的迭代训练
2	用蒙特卡洛积分法 (Monte Carlo integration) 估计 NPDE 的平均残差 ε_r ，即通过随机抽样一组残差点集 \mathcal{S} 来计算平均值， $\mathcal{S} = \{x_1, x_2, \dots, x_{ \mathcal{S} }\}$ ： $\varepsilon_r \approx \frac{1}{ \mathcal{S} } \sum_{x \in \mathcal{S}} \left\ f(x; \frac{\partial \hat{u}}{\partial x_1}, \dots, \frac{\partial \hat{u}}{\partial x_m}; \frac{\partial^2 \hat{u}}{\partial x_1 \partial x_1}, \dots, \frac{\partial^2 \hat{u}}{\partial x_1 \partial x_m}; \dots; \lambda) \right\ $
3	如果满足以下条件： $\varepsilon_r < \varepsilon_0$ ，则停止算法。否则 将残差点集 \mathcal{S} 中残差最大的 m 个新残差点添加至初始残差点集 Γ 中 ，继续重复步骤 2。

在 PINN 模型训练过程中，梯度的更新公式为式 (2.71)，其中 η 是学习效率

参数。基于 Adam 优化算法，学习速率退火方法被用于建立 PINN 模型自主选择 λ_i 权重的自适应规则，具体的算法逻辑如表 2.8 所示。该方法的主要思想是在 PINN 模型训练过程中跟踪反向传播梯度的一阶矩和二阶矩，并基于这些信息自适应缩放 θ 向量中每个参数的学习速率。同时利用反向传播的梯度统计量自动调整 λ_i 权重，使式 (2.72) 中各惩罚项的相互作用得到适当平衡。

$$\theta_{n+1} = \theta_n - \eta \nabla_{\theta} Loss(\theta_n) = \theta_n - \eta \nabla_{\theta} Loss_f(\theta_n) - \eta \sum_{i=1}^N \lambda_i \nabla_{\theta} Loss_i(\theta_n) \quad (2.71)$$

$$Loss(\theta) = Loss_f(\theta) + \sum_{i=1}^N \lambda_i Loss_i(\theta) \quad (2.72)$$

表 2.8 学习速率退火方法的算法逻辑

步骤	算法内容
1	建立 PINN 模型的损失函数: $Loss(\theta) = Loss_f(\theta) + \sum_{i=1}^N \lambda_i Loss_i(\theta)$
2	计算引入的权重常数 $\hat{\lambda}_i$: $\hat{\lambda}_i = \frac{\max_{\theta} \{ \nabla_{\theta} Loss_f(\theta_n) \}}{\overline{\nabla_{\theta} Loss_i(\theta_n)}}$, $i = 1, 2, 3, \dots, N$, 这里 $\overline{\nabla_{\theta} Loss_i(\theta_n)}$ 表示 $\nabla_{\theta} Loss_i(\theta_n)$ 关于参数 θ 的平均值
3	使用以下形式的移动均值来更新 λ_i : $\lambda_i = (1 - \beta)\lambda_i + \beta\hat{\lambda}_i$, $i = 1, 2, 3, \dots, N$
4	通过梯度下降法更新参数 θ : $\theta_{n+1} = \theta_n - \eta \nabla_{\theta} Loss_f(\theta_n) - \eta \sum_{i=1}^N \lambda_i \nabla_{\theta} Loss_i(\theta_n)$

2.5 本章小结

本章系统地梳理了 VIV、DNN 以及 PINN 的基本理论，为下文 CFD 数值模拟、建立强耦合 PINN 模型以及求解 VIV 正反问题提供理论支持。本章主要内容总结如下：

(1) 首先，对 VIV 的基本理论进行详细介绍。主要包括：VIV 的产生机理及相关控制参数、流体运动和固体振动的基本控制方程以及用于 CFD 模拟的动网格技术。

(2) 其次，对数据驱动 DNN 的基本理论进行介绍。主要包括：DNN 结构体系以及工作原理、以三层神经网络为例确定了反向传播算法的核心公式和计算步骤、损失函数的构建方法、对比不同激活函数以及网络训练参数的设置依据。

(3) 最后, 对物理驱动 PINN 的基本理论进行介绍。主要包括: PINN 结构体系以及工作原理、正问题的计算步骤、反问题的参数更新方法、局部自适应激活函数的构建方法以及 RAR 等训练过程优化方法。

第 3 章 基于物理驱动深度学习的强耦合涡激振动模型

3.1 引言

随着 DL 的进一步发展, PINN 开始被应用于解决 VIV 相关问题。但现有的研究侧重于弱耦合 PINN 模型, 仅将描述粘性不可压缩流体动量守恒的 N-S 方程嵌入至 DNN, 忽略了 VIV 过程中有关流体运动与固体振动耦合的先验规律。绕流升力 F_l 和绕流阻力 F_d 是 VIV 非常重要的耦合条件, 既可以基于流固耦合面的边界条件和流体的运动信息积分求得, 也可以通过固体振动的动力学方程计算求得。本章将 N-S 方程和流固耦合条件方程同时作为先验知识, 强化物理信息对神经网络的约束, 建立基于物理驱动深度学习的强耦合 VIV 模型。其次, 以 CFD 理论和动网格技术为指导, 数值模拟 $Re = 200$ 的二维圆柱结构 VIV, 提取 CFD 模拟数据构建训练样本集。最后, 以训练成本和计算精度为指标, 对比了强弱耦合 PINN 模型的表现性能。

3.2 涡激振动物理模型

在流场中, 初始来流 U 作用于固体结构表面, 脱离出大量漩涡。这些涡流反作用于固体结构上, 产生交替的时相关力 (绕流力), 致使固体结构产生振动响应。如图 2.1 所示, 绕流力作用在两个方向, 分别是平行于来流方向的绕流阻力 F_d 和垂直于来流方向的绕流升力 F_l 。平行于来流方向被称为顺流向, 垂直于来流方向被称为横流向。固体结构表面局部流体的交替旋转产生了循环来流 U_l 。如图 3.1 所示, 当初始来流 U 和循环来流 U_l 共同作用于固体结构表面时, 其上下表面的速度场分布不再保持对称, 分别呈 $U+U_l$ 和 $U-U_l$ 。固体结构上下表面的速度场分布差导致绕流升力 F_l 的存在。绕流阻力 F_d 是因为自然流体中存在的粘性所产生的。

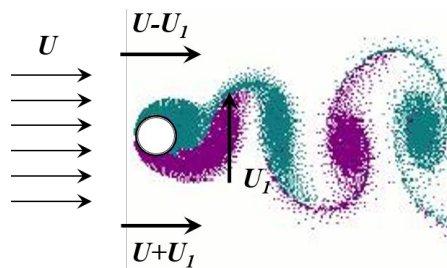


图 3.1 VIV 绕流力的产生机理

固体结构的顺流向振动位移与横流向振动位移相比非常微弱，目前大部分的研究仅考虑横流向的振动响应，因此本文在建立 VIV 物理模型时仅考虑横流向上绕流升力 F_l 的耦合。为了简化嵌入 DNN 中的物理方程，VIV 物理模型被简化成一个单向振动（横流向）的质量-弹簧-阻尼弹性系统，如图 3.2 所示。固体结构振动方程为式 (3.1)，其中 m 、 c 和 k 分别表示固体结构的质量、阻尼以及刚度， t 表示的是时间， $r(t)$ 表示固体结构在横流向的振动位移。

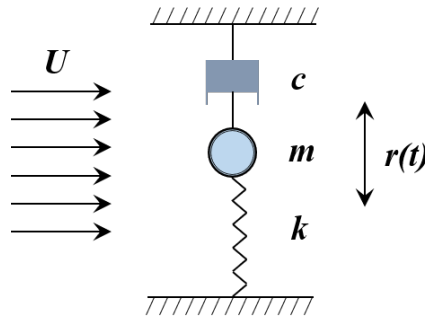


图 3.2 VIV 简化物理模型

$$m \frac{d^2 r}{dt^2} + c \frac{dr}{dt} + kr = F_l \quad (3.1)$$

对于雷诺数较小的流场，流体质点通常做有条不紊的规则运动，且呈现不相混掺的状态。同时流体被认为是不可压缩的，这意味着流体的密度可以一直保持恒定，一般满足以下基本假设：

- (1) 流体应符合理想牛顿液体的本构方程，即相邻层平行流动的质点所产生的剪切应力 τ 与横流向的速度梯度（剪切速率） L 呈正比关系；
- (2) 流场始终处于层流状态，流体质点微团的运动轨迹没有较明显的不规则脉动，并且相邻层平行流动的质点只有分子热运动引起的动能转换；
- (3) 除入流边界和出流边界外，其余的边界条件均基于无滑移假设。

本文仅考虑二维（Two-dimensional, 2D）平面 VIV 的相关计算，并使用 N-S 方程来描述粘性不可压缩流体的流场运动。在二维直角坐标系中，N-S 方程的分量形式如式 (3.2) 至式 (3.4) 所示，包括连续性微分方程（Continuity differential equation）和双向运动微分方程（Motion differential equation）。这里 (x, y) 表示二维流场的直角坐标， $u(t, x, y)$ 、 $v(t, x, y)$ 和 $p(t, x, y)$ 分别表示流场各质点的顺流向速度、横流向速度以及压力， Re 表示的是描述粘性流体流动状态的雷诺数。

连续性微分方程:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3.2)$$

双向运动微分方程:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + \frac{\partial p}{\partial x} - \frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = 0 \quad (3.3)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + \frac{\partial p}{\partial y} + \frac{\partial^2 r}{\partial t^2} - \frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) = 0 \quad (3.4)$$

在 VIV 过程中, 流场与固体结构之间存在着很强的耦合作用, 这主要是通过耦合条件绕流升力 F_l 和绕流阻力 F_d 所传递的。流固耦合面的 BC/IC 以及周围流体质点的速度、压力分布, 均会影响绕流力的大小, 从而决定固体结构的振动响应。同时固体结构的振动也反过来影响流场速度场和压力场的再分布。如图 3.3 所示, 绕流力可以通过两种途径求得, 分别是流场运动信息结合 BC/IC 积分计算和固体结构动力学方程微分计算。

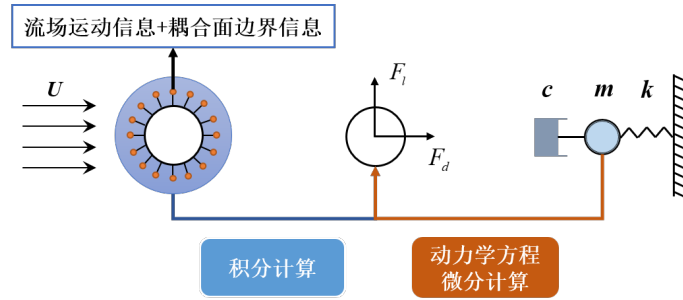


图 3.3 耦合条件求取途径

式 (3.5) 和式 (3.6) 表示基于流固耦合面的边界信息以及流场质点运动信息 (u 、 v 、 p) 积分求得绕流力的方法, 其中 (n_x, n_y) 表示流固耦合面的法向量, ds 表示积分弧长。式 (3.1) 和式 (3.5) 的结合代表了本文所建立的 VIV 物理模型在横流向的强耦合, 是嵌入 DNN 的重要先验知识。

$$F_l = \oint [-pn_y + 2\text{Re}^{-1}v_y n_y + \text{Re}^{-1}(u_y + v_x)n_x] ds \quad (3.5)$$

$$F_d = \oint [-pn_x + 2\text{Re}^{-1}u_x n_x + \text{Re}^{-1}(u_y + v_x)n_y] ds \quad (3.6)$$

3.3 强耦合物理信息神经网络模型

3.3.1 数据驱动深度神经网络拓扑结构

DNN 拓扑结构的设计是嵌入物理先验知识的基础，对所建立的 PINN 模型的性能表现影响非常大。其结构体系是由多个单层非线性网络组成，每层网络由大量神经元连接而成。确定好网络输入变量和输出变量后，**隐藏层和每层神经元个数的选择十分关键。选择的基本原则是：在满足训练精度的前提下构造尽可能紧凑的拓扑结构，避免“训练过拟合”。**同时需要根据解决问题的复杂程度、样本数据的分布特性、激活函数的选择以及 NPDE 的形式等对网络结构进行适当调整。

本文所建立的数据驱动 DNN 结构形式为全连接神经网络（Fully-connected neural network, FCNN），由网络输入层、网络输出层和 8 个隐藏层组成，其中每个隐藏层设置 30 个神经元，如图 3.4 所示。

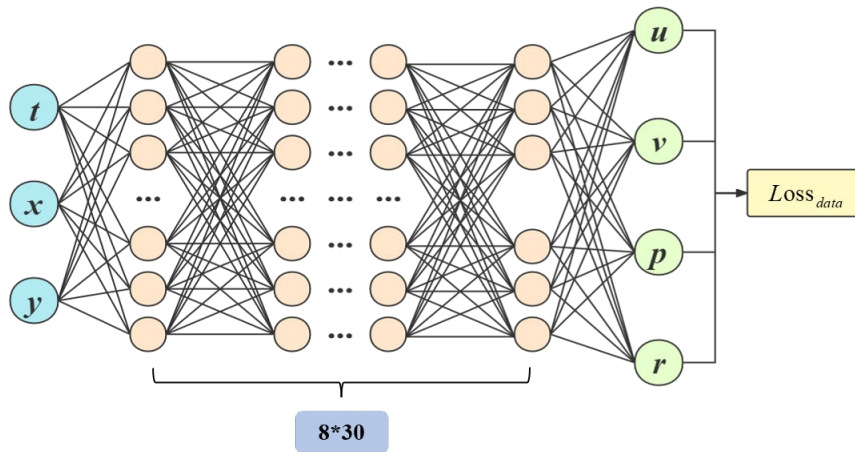


图 3.4 数据驱动 DNN 拓扑结构

网络输入层的变量分别是时间 t 和流场质点空间坐标 (x, y) 。网络输出层的变量分别是流场质点顺流向速度 $u(t, x, y)$ 、流场质点横流向速度 $v(t, x, y)$ 、流场质点压力 $p(t, x, y)$ 和固体结构振动位移 $r(t)$ 。在网络信号正向传播过程中， $f(x) = \sin x$ 作为激活函数，将所有样本数据归一化至 $(0, 1)$ ，实现神经网络的非线性映射。同时 Adam 优化器用于实现网络训练过程中梯度自动下降的目标，快速更新网络权重矩阵和偏置系数。在网络信号反向传播过程中，数据驱动损失函数 $Loss_{data}$ 逐步迭代至最小阈值， $Loss_{data}$ 被表示为式 (3.7)。

$$Loss_{data} = \sum_{n=1}^N (|u(t^n, x^n, y^n) - u^n|^2 + |v(t^n, x^n, y^n) - v^n|^2 + |p(t^n, x^n, y^n) - p^n|^2 + |r(t^n) - r^n|^2) \quad (3.7)$$

这里 N 是空间域的配点数量, u^n 、 v^n 、 p^n 和 r^n 分别表示流场质点顺流向速度、流场质点横流向速度、流场质点压力以及固体振动位移的真实值, $u(t^n, x^n, y^n)$ 、 $v(t^n, x^n, y^n)$ 、 $p(t^n, x^n, y^n)$ 和 $r(t^n)$ 分别表示流场质点顺流向速度、流场质点横流向速度、流场质点压力以及固体振动位移的网络预报值。

3.3.2 物理驱动深度神经网络拓扑结构

与传统纯数据驱动 DNN 相比, PINN 将物理先验知识和样本数据集作为共同约束, 快速逼近 NPDE 的真实解。随着 AD 技术的普及, 代表先验物理规律的 NPDE 被嵌入至 DNN 结构中, 以物理驱动损失函数 $Loss_{phy}$ 作为额外约束。如图 3.5 所示, Raissi 等^[41]人提出弱耦合 PINN 模型来研究二维圆柱 VIV, 将流场质点的连续性微分方程和双向运动微分方程添加至网络结构中。**此模型忽略了关键的流固耦合条件: 流场运动和固体振动所传递的耦合绕流力**。因此, 在避免“过拟合”的前提下, 需要较多的训练成本来维持代理模型的精确度。用于训练的 VIV 监测数据往往具有“小样本、稀疏”等特征, 这导致弱耦合 PINN 模型的普适性较差。

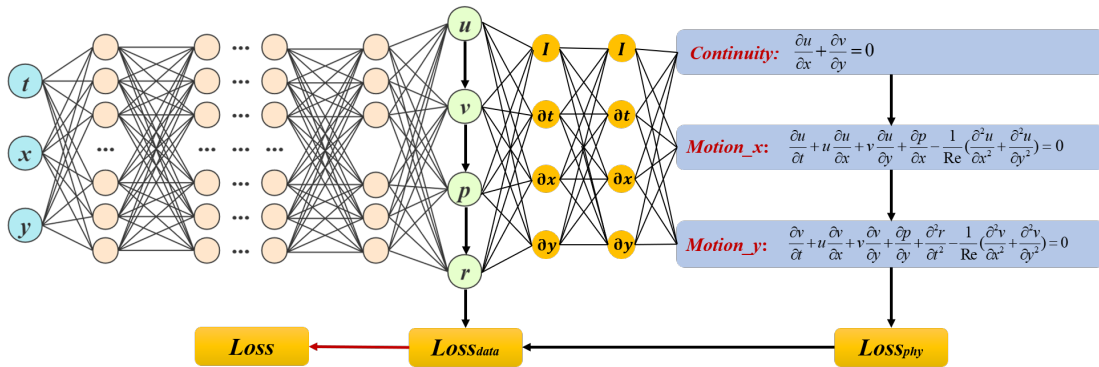


图 3.5 二维 VIV 弱耦合 PINN 模型

本文将绕流升力 F_l 的耦合条件方程添加至深度网络结构中, 建立强耦合 PINN 模型。如图 3.6 所示, F_l^1 表示基于流场运动信息和耦合面 BC/IC 积分计算出的绕流升力, F_l^2 表示基于固体结构动力学方程微分计算出的绕流升力, F_l^1 和 F_l^2 将以差值的形式存在于 $Loss_{phy}$, 确保训练样本满足 VIV 在横流向的耦合条件。为了方便地表达 $Loss_{phy}$, a_1 、 a_2 和 a_3 分别用来替换式 (3.2) 至式 (3.4), 依次表示为式 (3.8) 至式 (3.10)。 $Loss_{phy}$ 用于检验嵌入网络结构的 NPDE 拟合程度, 将其表示为式 (3.11)。

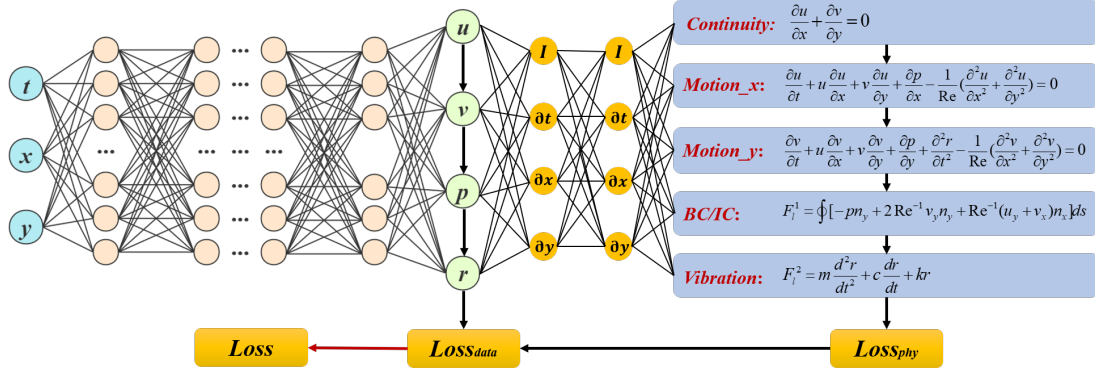


图 3.6 二维 VIV 强耦合 PINN 模型

$$a_1 = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (3.8)$$

$$a_2 = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + \frac{\partial p}{\partial x} - \frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (3.9)$$

$$a_3 = \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + \frac{\partial p}{\partial y} + \frac{\partial^2 r}{\partial t^2} - \frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3.10)$$

$$\text{Loss}_{phy} = \sum_{i=1}^3 \sum_{n=1}^N (|a_i(t^n, x^n, y^n)|^2) + \sum_{n=1}^N |F_1^1(t^n) - F_1^2(t^n)|^2 \quad (3.11)$$

强耦合 PINN 模型的损失函数 $Loss$ 共包含两部分：**数据驱动损失函数 $Loss_{data}$** 和 **物理驱动损失函数 $Loss_{phy}$** ，如式 (3.12) 所示。其中 N 表示空间域的配点数量， $a_i(t^n, x^n, y^n)$ 表示将网络输出变量 (u, v, p, r) 带入式 (3.8) 至式 (3.10) 所得到的计算结果， $F_1^1(t^n)$ 表示将 (u, v, p) 带入式 (3.5) 所得到的计算结果， $F_1^2(t^n)$ 表示将 $r(t)$ 带入式 (3.1) 所得到的计算结果。 **$F_1^1(t^n) - F_1^2(t^n)$ 是用来检验是否满足流体运动和固体振动在横流向上的耦合作用。**

$$\begin{aligned} \text{Loss} &= \text{Loss}_{data} + \text{Loss}_{phy} \\ &= \sum_{n=1}^N (|u(t^n, x^n, y^n) - u^n|^2 + |v(t^n, x^n, y^n) - v^n|^2 \\ &\quad + |p(t^n, x^n, y^n) - p^n|^2 + |r(t^n) - r^n|^2) \\ &\quad + \sum_{i=1}^3 \sum_{n=1}^N (|a_i(t^n, x^n, y^n)|^2) + \sum_{n=1}^N |F_1^1(t^n) - F_1^2(t^n)|^2 \end{aligned} \quad (3.12)$$

3.4 强弱耦合 PINN 模型表现性能比较

3.4.1 CFD 数值模拟

Fluent 软件是用来模拟各类流场运动的商业软件，包括从不可压缩流场到高超音速范围内的流场。其成熟的动网格技术为解决 VIV 这类边界运动问题提供了强大的支持，同时包含多种算法进行模拟求解，包括：**非耦合隐式算法 (Uncoupled implicit algorithm, UIA)**、**耦合隐式算法 (Coupled implicit algorithm, CIA)** 和 **耦合显式算法 (Coupled explicit algorithm, CEA)**。本文基于 Fluent 软件模拟了 $Re=200$ 的二维圆柱 VIV，并提取流场运动信息和圆柱固体振动位移，构建比较强弱耦合 PINN 模型表现性能的训练样本集。

(1) 二维圆柱结构 VIV 模型数值仿真流程

本文基于 Fluent 软件模拟 $Re=200$ 的二维圆柱 VIV，同时提取 CFD 模拟数据构建训练样本集。**首先利用 Geometry 软件建立 VIV 的二维尺寸模型**。圆柱结构的特征长度（管道直径） D 为 0.01m ，位于直角坐标系的中心。流场的顺流向（ x 方向）坐标范围为 $-5D$ 至 $10D$ ，横流向（ y 方向）坐标范围为 $-5D$ 至 $5D$ ，如图 3.7 所示。结构参数定义如下：质量 m 为 2.7325 ，固有频率 ω 为 2.5133 ，阻尼比 ζ 为 0.00383 。

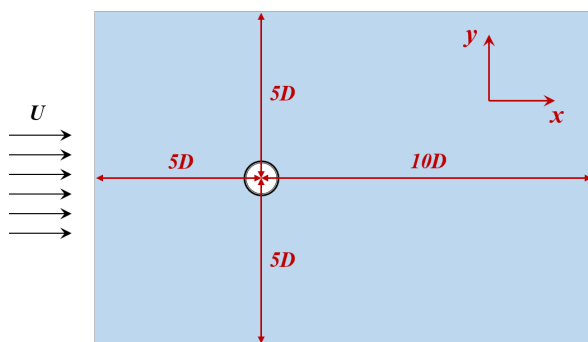


图 3.7 二维圆柱 VIV 的基础尺寸模型

接着利用 ICEM 软件对基础尺寸模型划分网格，使用的方法是动网格技术中的嵌套网格 (Overset grids) 方法，可以有效避免网格畸变、负网格等不利问题。本文所建立的嵌套网格共包含两套网格：背景网格 (Background grid) 和前景网格 (Foreground grid)，如图 3.8 所示，其中越靠近坐标中心网格划分越密。在 Fluent 软件进行后续计算时，求解器可以自动判别嵌套边界，并对网格重叠的地方进行挖洞插值计算，最终划分的整体网格如图 3.9 所示。

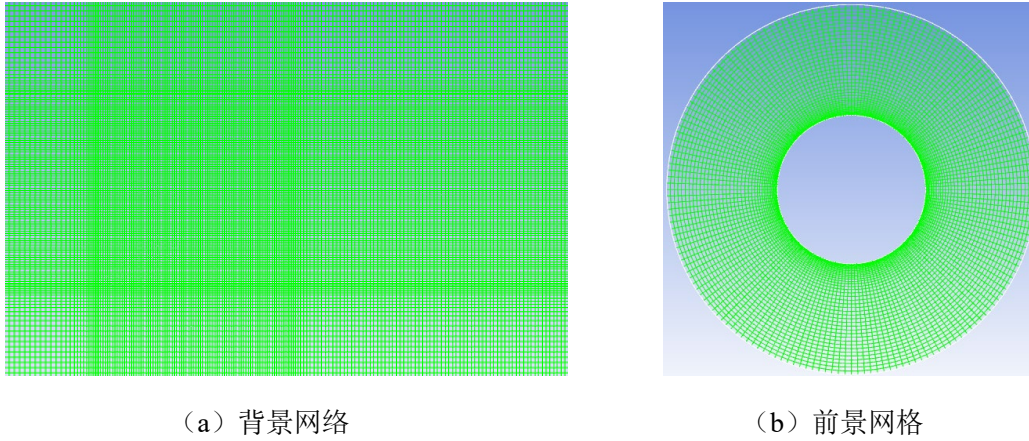


图 3.8 嵌套网格划分

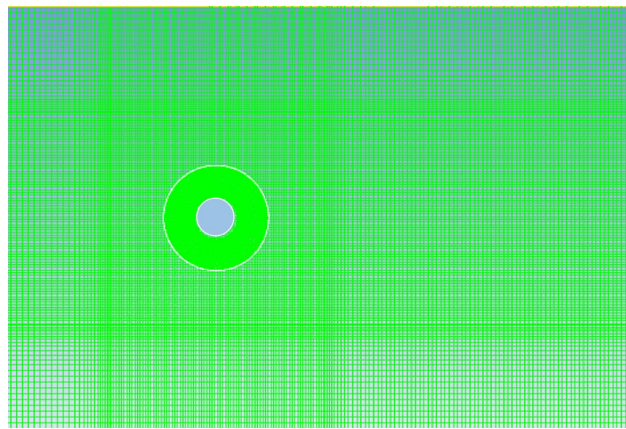


图 3.9 整体网格划分（中间为插值挖洞部分）

最后将划分完网格的 VIV 模型导入 Fluent 软件，计算进程为 6 核并行，计算精度为双倍精度。流场域的介质为液态水，液态水的密度 $\rho=998.2 \text{ kg/m}^3$ ，动力粘性系数 $\mu=0.001003 \text{ kg/m}\cdot\text{s}$ ，来流速度 $U=0.02 \text{ m/s}$ ，带入雷诺数计算公式结果是 200，如式 (3.13) 所示。入流边界的类型为速度入口 (Velocity-inlet)，出流边界的类型为压力出口 (Pressure-outlet)，剩余均为对称无滑移边界 (Symmetry no-slip boundary)。本文建立 VIV 模型选取的是两方程模型中的 SST $k-\omega$ 模型，使用全隐式积分方法计算时间项，使用二阶迎风离散式计算对流项。

$$\text{Re} = \frac{\rho U D}{\mu} = \frac{998.2 * 0.02 * 0.01}{0.001003} \approx 200 \quad (3.13)$$

(2) 用户自定义函数在 CFD 数值模拟软件的应用

CFD 数值模拟软件将基于 C 语言的用户自定义函数 (User defined function, UDF) 嵌入耦合计算模块，从而实现 VIV 的双向耦合运动仿真。在数值模拟过程中，软件自带的工具宏自动提取圆柱结构表面的绕流升力 F_l 和绕流阻力 F_d ，结合原结构参数 (m 、 c 和 k) 和初始振动条件，可以通过经典四阶龙格-库塔法

(Runge-Kutta methods) 求解圆柱结构的结构动力方程。求解所得的结构中心位移、速度和加速度等信息被调运至 CFD 求解器，同时利用嵌套网格方法更新流场域网格，进入下一个时间步长的耦合计算。UDF 程序的计算流程如图 3.10 所示。

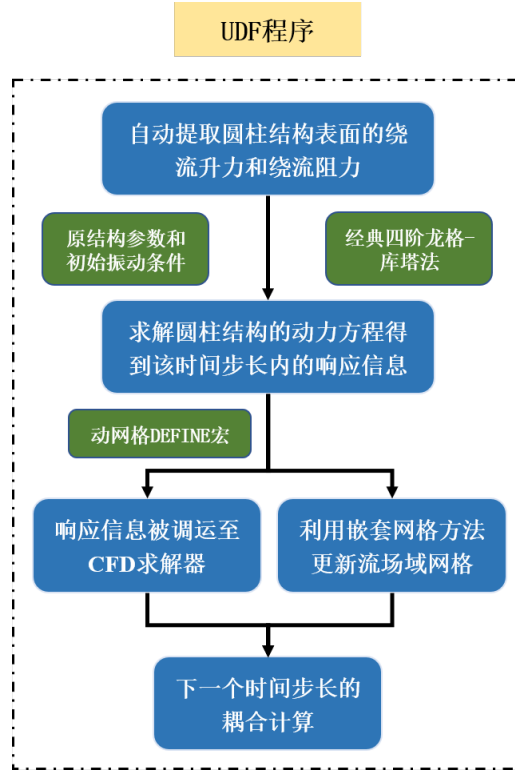


图 3.10 UDF 程序的计算流程

经典四阶龙格-库塔法通过显式或隐式迭代来求解 NPDE，简化求解过程的同时还保持有较高的计算精度，其原理如下：

需要求解的 NPDE 的一阶导数以及初值信息分别表示为式(3.14)和式(3.15)。

$$g' = f(t, g) \quad (3.14)$$

$$g(t_0) = g_0 \quad (3.15)$$

针对上述给出的初始条件，经典四阶龙格-库塔法给出的迭代方法为式(3.16)，其中 b_1 、 b_2 、 b_3 和 b_4 表示为式(3.17)至式(3.20)， h 表示计算的时间步长， n 表示迭代次数。

$$g_{n+1} = g_n + \frac{h}{6}(b_1 + 2b_2 + 2b_3 + b_4) \quad (3.16)$$

$$b_1 = f(t_n, g_n) \quad (3.17)$$

$$b_2 = f(t_n + \frac{h}{2}, g_n + \frac{h}{2}b_1) \quad (3.18)$$

$$b_3 = f\left(t_n + \frac{h}{2}, g_n + \frac{h}{2}b_2\right) \quad (3.19)$$

$$b_4 = f(t_n + h, g_n + hb_3) \quad (3.20)$$

(3) 网格独立性验证

网格独立性验证是数值模拟前一项非常重要的准备工作, 网格的整体质量直接影响到后续数值模拟计算的精度。较差的网格质量难以满足 PINN 训练样本集对于数值精度的要求, 过密的网格划分对计算成本是非常大的挑战。因此需要寻找合理的网格划分方法, 同时保证数值精度和计算成本的要求。

本文共使用 5 种网格划分方法进行网格独立性验证, 具体参数如表 3.1 所示。流体升力系数幅值 $|C_l|_{max}$ 、流体阻力系数均值 $\overline{C_d}$ 和斯特劳哈尔数 St 作为对比指标, 具体结果如表 3.2 所示。

表 3.1 VIV 模型网格划分参数

网格编号	前景网格圆周节点数量	整体网格单元数量
1	70	6877
2	90	12087
3	110	17126
4	125	20539
5	140	24506

表 3.2 不同网格数值模拟结果对比

网格编号	升力系数幅值 $ C_l _{max}$	阻力系数均值 $\overline{C_d}$	斯特劳哈尔数 St
1	0.42	1.08	0.179
2	0.51	1.15	0.185
3	0.59	1.24	0.194
4	0.66	1.34	0.208
5	0.68	1.35	0.211

由表 3.2 可得, 随着网格单元数的增加, 流体升力系数幅值 $|C_l|_{max}$ 、流体阻力系数均值 $\overline{C_d}$ 和斯特劳哈尔数 St 这三个指标逐渐增长至稳定水平。网格 2 相较于网格 1, 三个指标的相对增幅比分别为 21.4%、6.5% 和 3.4%。网格 3 相较于网格 2, 三个指标的相对增幅比分别为 15.7%、7.8% 和 4.9%。网格 4 相较于网格 3, 三个指标的相对增幅比分别为 11.9%、8.1% 和 7.2%。网格 5 相较于网格 4, 三个指标的相对增幅比分别为 3.0%、0.75% 和 1.4%。由此可见网格 4 和网格 5 的精密程度已经可以确保三个指标保持在稳定的水平。但考虑到网格单元数量对计算成本的影响, 最终本文选择基于网格 4 的划分方式建立相应的二维 VIV 模型。

(4) CFD 数值模拟结果准确性验证

为了确保 PINN 训练样本集的合理性, 本文将基于网格 4 所建立的二维 VIV 模型 ($Re=200$) 的数值模拟结果与其他学术论文中的结果进行比对, 如表 3.3 所示。本文结果与 Braza M^[113]论文结果相比, 流体升力系数幅值 $|C_l|_{max}$ 、流体阻力系数均值 $\overline{C_d}$ 和斯特劳哈尔数 St 这三个指标的相对误差分别为 1.5%、0.8%和 9.5%。与 Zhang J F^[114]论文结果相比, 三个指标的相对误差分别为 17.9%、5.5%和 4%。与魏志理^[115]论文结果相比, 三个指标的相对误差分别为 9.6%、3.9%和 9.5%。与常书平^[116]论文结果相比, 三个指标的相对误差分别为 1.5%、0.7%和 1.0%。以上误差结果均在合理区间范围内, 因此基于网格 4 所建立的二维 VIV 模型满足 PINN 训练样本集对于数据精度的要求。

表 3.3 本文 VIV 模型 ($Re=200$) 与前人文献模拟结果对比

作者	论文年份	升力系数幅值 $ C_l _{max}$	阻力系数均值 $\overline{C_d}$	斯特劳哈尔数 St
Braza M	1989	0.65	1.33	0.19
Zhang J F	1997	0.56	1.27	0.20
魏志理	2006	0.73	1.29	0.19
常书平	2008	0.67	1.35	0.21
本文	2022	0.66	1.34	0.208

基于网格 4 所建立的二维 VIV 模型 ($Re=200$) 共进行 120 s 的数值模拟计算。数值模拟计算结束时, 圆柱结构周围脱落形成的漩涡已经非常充分, 图 3.11 和图 3.12 分别为数值模拟过程中升力系数 C_l 和阻力系数 C_d 的时程变化。

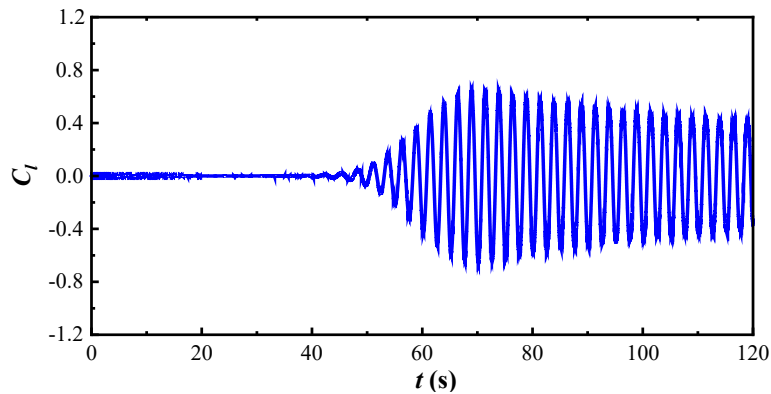
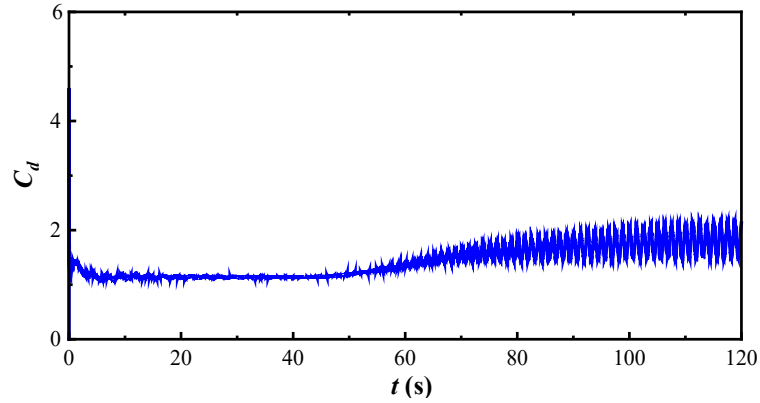


图 3.11 升力系数 C_l 时程变化图


 图 3.12 阻力系数 C_d 时程变化图

第 80 s 至第 90 s 的 CFD 模拟数据被用于构建训练样本集，包括流场域各质点的顺流向速度 u 、横流向速度 v 、压力 p 以及圆柱结构的振动位移 r 。其中流场质点的时空域分布如图 3.13 所示。训练样本集的总时长为 10 s，提取时间间隔为 0.05 s，共包含 200 个时间步长。为了保证 PINN 模型的训练效果，样本点位的选择范围为：流场顺流向（ x 方向）坐标是 $-2D$ 至 $7D$ ，流场横流向（ y 方向）坐标是 $-2.5D$ 至 $2.5D$ 。

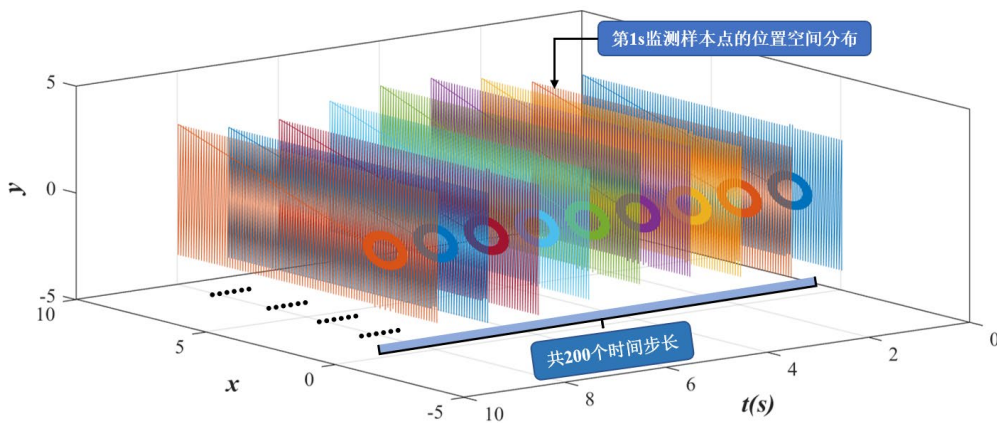


图 3.13 流场质点的时空域分布

3.4.2 对比工况介绍

为了比较强弱耦合 PINN 模型的表现性能，强耦合 PINN 模型的数据成本取为弱耦合 PINN 模型的一半，具体如表 3.4 所示。弱耦合 PINN 模型的空间域配点数量为 12000，其中前景网格范围内选择 7000 个流场质点，背景网格范围内选择 5000 个流场质点。强耦合 PINN 模型的空间域配点数量为 6000，其中前景网格范围内选择 3000 个流场质点，背景网格范围内选择 3000 个流场质点。

表 3.4 强弱耦合 PINN 模型的工况介绍

耦合类型	空间域配点数量	训练时间步数	训练方法	PINN 中添加的先验知识
弱耦合	12000	200	PINN	N-S 方程
强耦合	6000	200	PINN	N-S 方程+流固耦合条件方程

训练神经网络时，合理的网络训练参数设置是取得高精度训练结果的关键。训练强弱耦合 PINN 模型时采用一致的参数设置，具体如表 3.5 所示，以保证性能比较的公平性。以强耦合 PINN 模型为例，流场域的空间配点数量为 6000，训练时间步数为 200，因此在一个完整 Epoch 中共包含 $6000 \times 200 = 1.20E+06$ 个待更新样本点。Batch size 为 10000，意味着每批次取 10000 个样本点进行训练，直至更新完全部点集。

表 3.5 强弱耦合 PINN 模型的网络训练参数设置

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	10000	1.00E-03	$f(x) = \sin x$
第 2 阶段	300	10000	1.00E-04	
第 3 阶段	200	10000	1.00E-05	

3.4.3 流场运动信息再现情况

本文基于 Pycharm 平台训练强弱耦合 PINN 模型，运行 Pycharm 的硬件参数信息对训练的时间成本有着直接的影响。本文所使用的硬件设备为 HP Zhan 86 Pro G1 MT，处理器为 Intel(R) Core(TM) i7-7700，内存为 8 GB，操作系统为 64 位。表 3.6 是强弱 PINN 模型完成 700 次完整迭代后花费的时间成本，弱耦合 PINN 模型的训练时间为 46.7 h，强耦合 PINN 模型的训练时间为弱耦合 PINN 模型的 3/5，仅需 27.6 h。由此可见，流固耦合条件方程作为新增的物理约束，有效地降低了训练 PINN 模型所需的时间成本。

表 3.6 强弱耦合 PINN 模型的时间成本

耦合类型	迭代次数 (Epoch)	时间成本 (h)
强耦合	700	27.6
弱耦合	700	46.7

图 3.14 为强弱耦合 PINN 模型在训练迭代过程中损失函数 Loss 的变化曲线，损失函数 Loss 的表达式为式 (3.12)。由图可知，大致在 200 次迭代训练后，两个 PINN 模型的损失函数 Loss 开始趋于稳定，逐渐收敛至阈值范围内。由于损

失函数 $Loss$ 中增加了绕流升力 F_l (耦合条件) 的差值项, 强耦合 PINN 模型的 $Loss$ 曲线下落地更快, 并且可以迭代至更小误差, 从而获得高精度的计算结果。

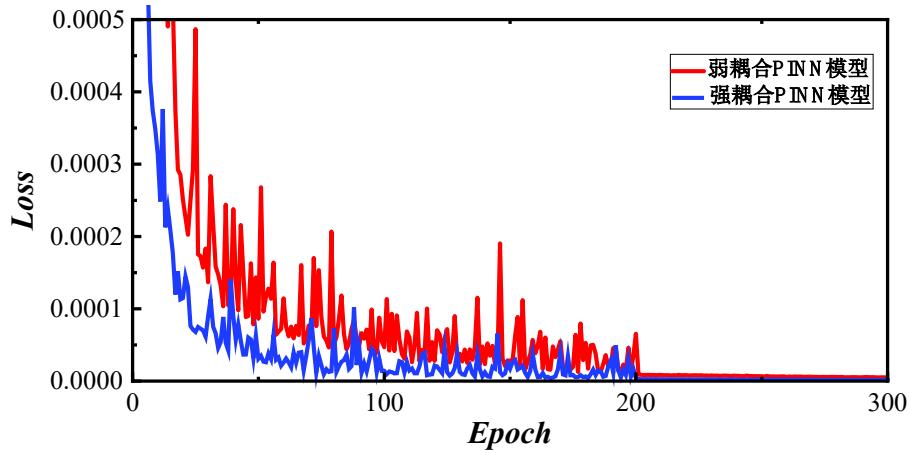
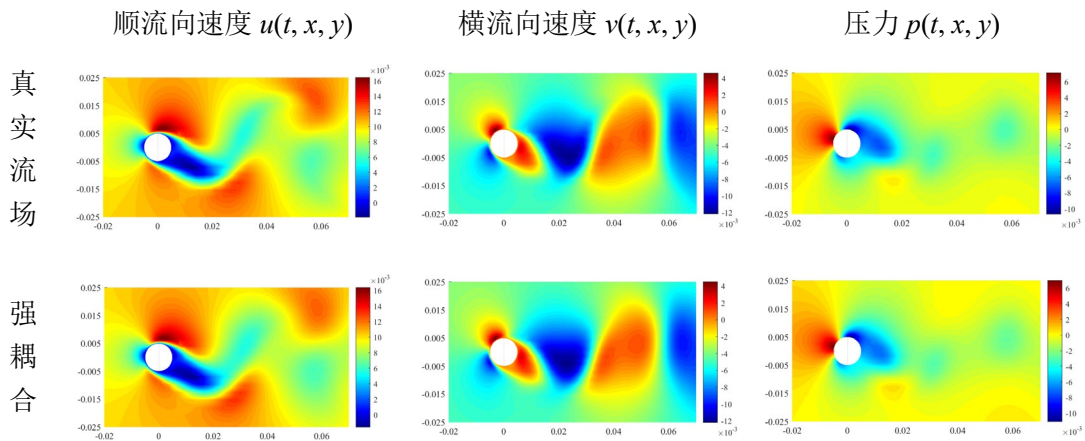


图 3.14 强弱耦合 PINN 模型损失函数 $Loss$ 的变化曲线

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (X_{fore,i} - X_{true,i})^2}{N}} \quad (3.21)$$

强弱耦合 PINN 模型均可以准确再现第 0-10 s 内流场域的运动信息, 包括各流场质点的顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 以及压力 $p(t, x, y)$ 。图 3.15 至图 3.17 分别为强弱耦合 PINN 模型第 1 s、第 3 s 和第 8 s 流场运动信息的再现结果。均方根误差 (Root mean squared error, RMSE) 被用来量化强弱耦合 PINN 模型再现结果与真实值的误差, 从而比较两个模型的表现性能。RMSE 的表示式为式 (3.21), 其中 N 表示空间域的配点数量, $X_{true,i}$ 表示真实值, $X_{fore,i}$ 表示模型输出值。RMSE 的值越小, 表示代理模型的运行精度越高。图 3.18 是强弱耦合 PINN 模型第 0-10 s 内流场运动信息再现结果的 RMSE 对比。下文图表里的所有变量均采用国际标准单位。



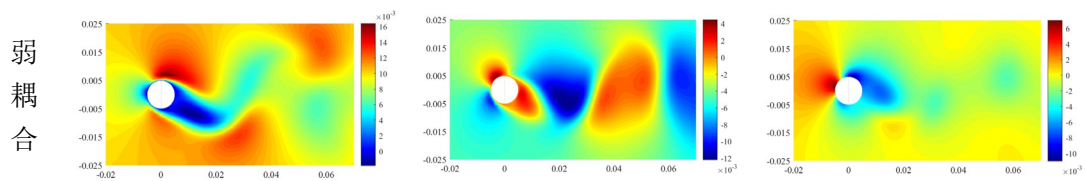


图 3.15 强弱耦合 PINN 模型第 1 s 流场运动信息的再现结果

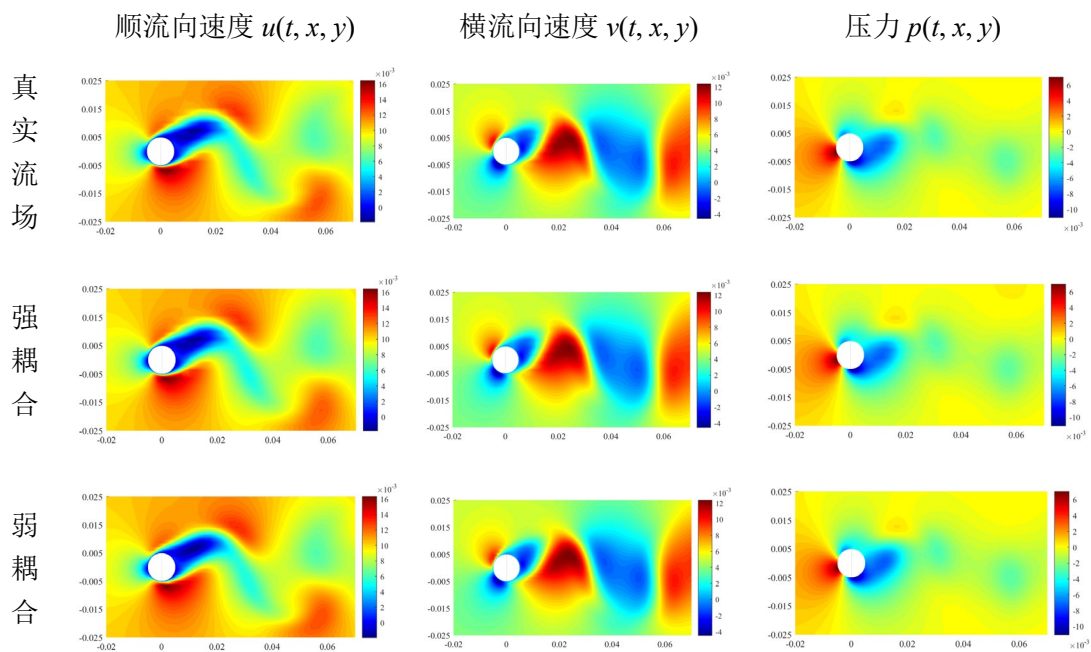


图 3.16 强弱耦合 PINN 模型第 3 s 流场运动信息的再现结果

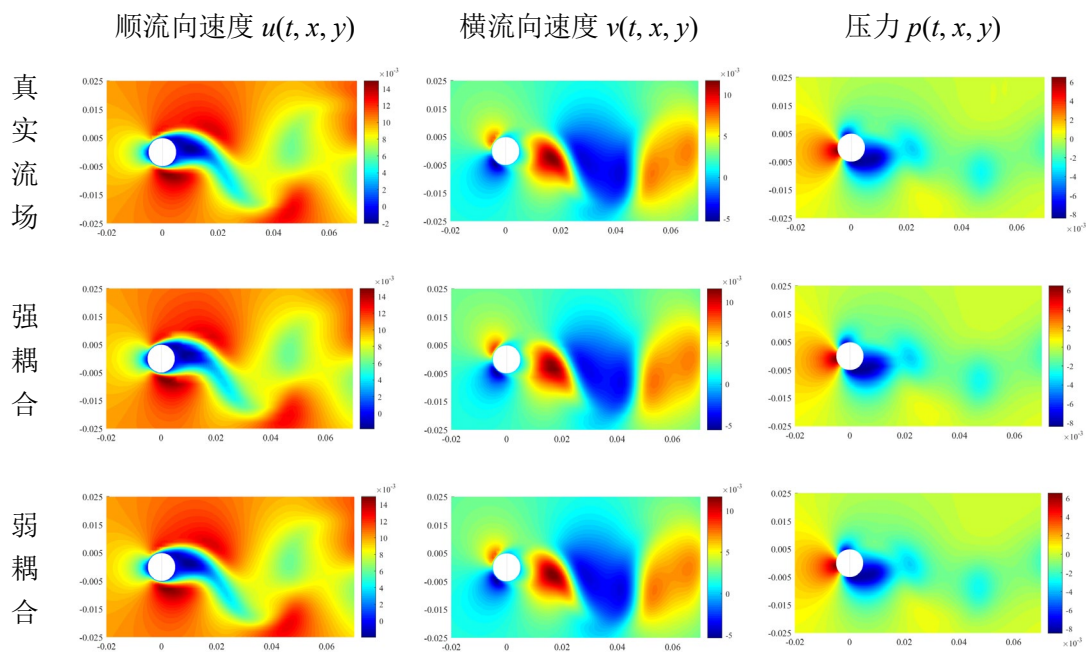


图 3.17 强弱耦合 PINN 模型第 8 s 流场运动信息的再现结果

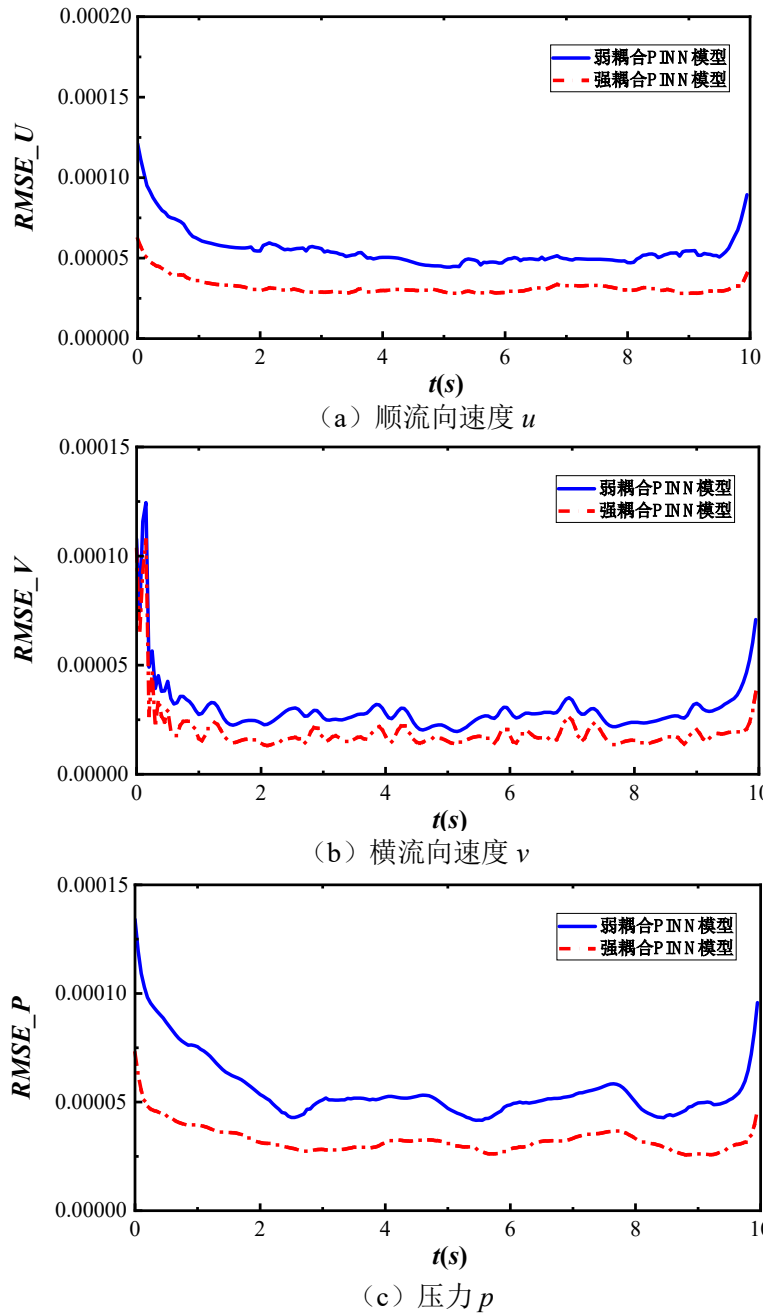


图 3.18 强弱耦合 PINN 模型流场运动信息再现结果的 RMSE 对比

对于流场质点顺流向速度 u ，弱耦合 PINN 模型再现结果的 RMSE 在 $4.4\text{E-}05$ 至 $1.21\text{E-}04$ 范围内浮动，强耦合 PINN 模型再现结果的 RMSE 在 $2.8\text{E-}05$ 至 $6.2\text{E-}05$ 范围内浮动，单个时间步长内 RMSE 相对降低 45.2%至 51.3%。

对于流场质点横流向速度 v ，弱耦合 PINN 模型再现结果的 RMSE 在 $4.2\text{E-}05$ 至 $1.34\text{E-}04$ 范围内浮动，强耦合 PINN 模型再现结果的 RMSE 在 $2.6\text{E-}05$ 至 $7.3\text{E-}05$ 范围内浮动，单个时间步长内 RMSE 相对降低 48.8%至 54.1%。

对于流场质点压力 p ，弱耦合 PINN 模型再现结果的 RMSE 在 $2.0\text{E-}05$ 至 $1.25\text{E-}04$ 范围内浮动，强耦合 PINN 模型再现结果的 RMSE 在 $1.3\text{E-}05$ 至 $1.07\text{E-}04$ 范围内浮动，单个时间步长内 RMSE 相对降低 44.3%至 80.2%。

图 3.18 量化地比较了强弱耦合 PINN 模型再现流场运动信息的性能表现。综合来看，与弱耦合 PINN 模型相比，本文所提出的强耦合 PINN 模型再现流场运动信息时表现出更高的运行精度。

3.4.4 结构振动信息再现情况

除流场域的运动信息外，结构振动信息也是 PINN 模型需要准确再现的 VIV 数据，包括固体结构振动位移 r 、绕流升力 F_l 以及绕流阻力 F_d 。

横流向运动微分方程和固体结构动力学方程均包含位移参数 r ，这与 PINN 模型损失函数 $Loss$ 的收敛水平密切相关。图 3.19 是强弱耦合 PINN 模型对 0-10 s 内固体结构振动位移 $r(t)$ 的再现结果，图 3.20 是再现结果的 RMSE 对比。

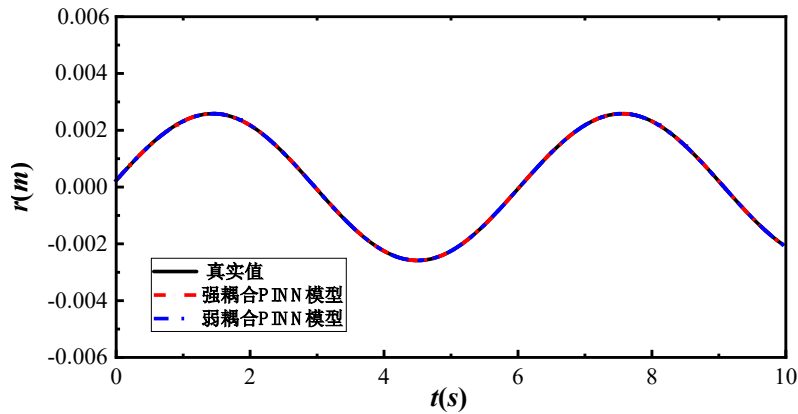


图 3.19 强弱耦合 PINN 模型 0-10 s 内固体结构振动位移 r 的再现结果

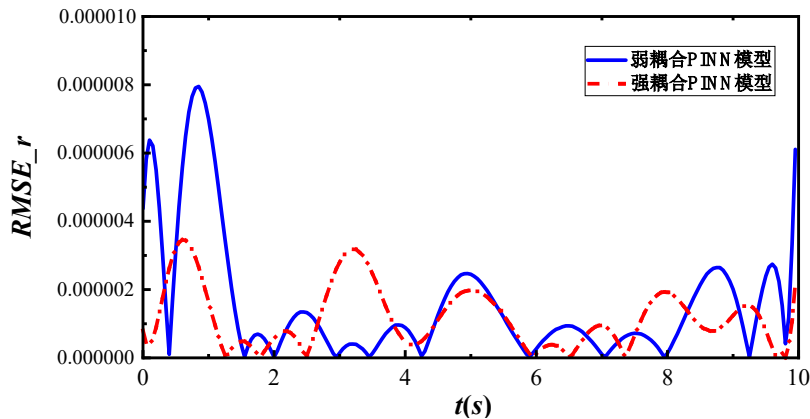


图 3.20 固体结构振动位移 r 再现结果的 RMSE 对比

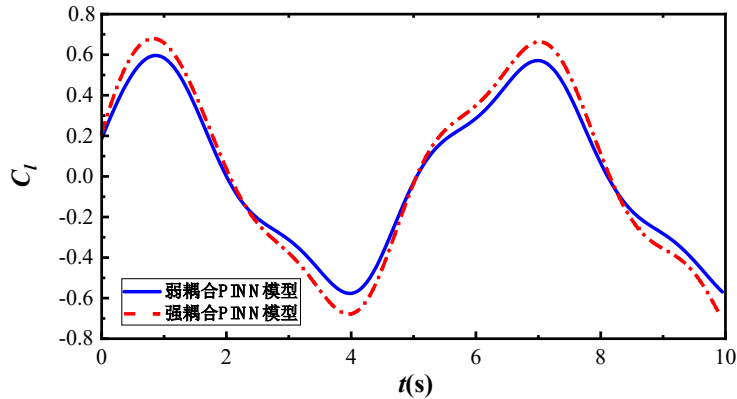
对于固体结构振动位移 r ，弱耦合 PINN 模型再现结果的 RMSE 在 $1.1\text{E-}07$

至 $7.9\text{E-}06$ 范围内浮动, 强耦合 PINN 模型再现结果的 RMSE 在 $1.0\text{E-}07$ 至 $3.4\text{E-}06$ 范围内浮动。在大多数时间步长内, 强耦合 PINN 模型再现结果的 RMSE 小于弱耦合 PINN 模型, 且 RMSE 相对降低 21.7%至 90.3%。同时强耦合 PINN 模型可以改善部分时间步长内 RMSE 峰值相对过高的情况, 将整体 RMSE 控制在更低水平。

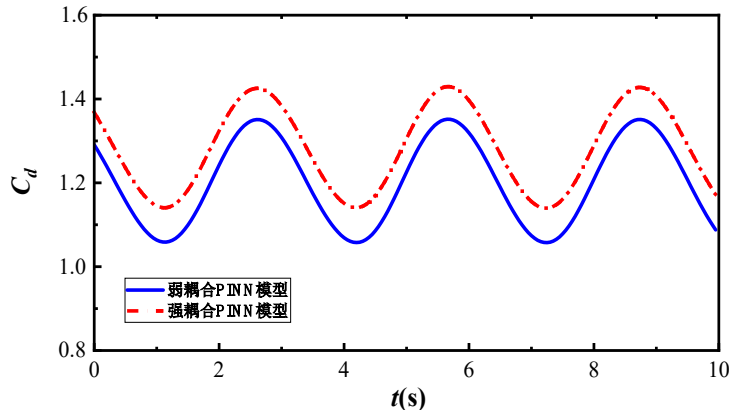
基于流场运动信息的输出结果和流固耦合面的边界条件, 可以积分计算出固体结构所受的绕流升力 F_l 和绕流阻力 F_d , 积分表达式参考式 (3.5) 和式 (3.6)。为了更直观地比较强弱耦合 PINN 模型对绕流力的再现结果, 通过式 (3.22) 和式 (3.23) 将绕流升力 F_l 和绕流阻力 F_d 分别转化为流体升力系数 C_l 和流体阻力系数 C_d , 如图 3.21 所示。最后以流体升力系数幅值 $|C_l|_{max}$ 和流体阻力系数均值 \bar{C}_d 为指标, 将两个 PINN 模型的再现结果与 CFD 模拟结果做比较, 来验证再现结果的准确性, 如表 3.7 所示。

$$C_l = \frac{2F_l}{\rho DU^2} \quad (3.22)$$

$$C_d = \frac{2F_d}{\rho DU^2} \quad (3.23)$$



(a) 流体升力系数 C_l



(b) 流体阻力系数 C_d

图 3.21 强弱耦合 PINN 模型对流体力系数的再现结果

表 3.7 流体力系数再现结果正确性验证

耦合类型	输出结果 $ C_l _{max}$	数值模拟 $ C_l _{max}$	误差	输出结果 $\overline{C_d}$	数值模拟 $\overline{C_d}$	误差
强耦合	0.68	0.66	0.02	1.28	1.34	0.06
弱耦合	0.59	0.66	0.07	1.21	1.34	0.13

CFD 模拟方法或者实验方法的不同,会导致流体力系数的结果存在差异。对于 $Re=200$ 的层流流场,流体升力系数幅值 $|C_l|_{max}$ 通常在(0.56, 0.74)区间内浮动,流体阻力系数均值 $\overline{C_d}$ 通常在(1.19, 1.36)区间内浮动,本文强弱耦合 PINN 模型对流体力系数的再现结果均在合理范围内。参考表 3.3, CFD 模拟结果如下:流体升力系数幅值 $|C_l|_{max}$ 为 0.66,流体阻力系数均值 $\overline{C_d}$ 为 1.34。

对于流体升力系数幅值 $|C_l|_{max}$,弱耦合 PINN 模型的再现结果为 0.59,强耦合 PINN 模型的再现结果为 0.68,再现结果的相对误差由 10.7%降低至 3.0%。

对于流体阻力系数均值 $\overline{C_d}$,弱耦合 PINN 模型的再现结果为 1.21,强耦合 PINN 模型的再现结果为 1.28,输出结果的再现误差由 9.7%降低至 4.5%。

图 3.20 和图 3.21 量化地比较了强弱耦合 PINN 模型再现固体振动信息的性能表现。综合来看,与弱耦合 PINN 模型相比,本文所提出的强耦合 PINN 模型再现固体振动信息时表现出更高的运行精度。

3.5 本章小结

本章将描述粘性不可压缩流体动量守恒的 N-S 方程和流固耦合条件方程同时作为先验物理信息嵌入到传统 DNN 中,建立了强耦合 PINN 模型。同时基于 CFD 理论和动网格技术,对 $Re=200$ 的 VIV 进行 CFD 数值模拟,提取流场运动信息和固体结构振动位移来构建训练样本集。最后以流场运动信息和结构振动信息的再现结果作为指标,对比了强弱耦合 PINN 模型的性能表现。本章的主要结论如下所示:

(1) 与弱耦合 PINN 模型相比,强耦合 PINN 模型在先验知识中添加了流固耦合条件方程,强化了物理信息对神经网络的约束,凭借更少的训练成本获得更高精度的运行结果。

(2) 强耦合 PINN 模型可以更为精准地再现流场运动信息。对于顺流向速度 u 、横流向速度 v 以及压力 p ,单个时间步的 RMSE 的相对降低比例分别在 (45.2%, 51.3%)、(48.8%, 54.1%) 和 (44.3%, 80.2%) 区间内浮动。

(3) 强耦合 PINN 模型可以更为精准地再现结构振动信息。对于固体结构

振动位移 r ，单个时间步的 RMSE 的相对降低比例在 (21.7%, 90.3%) 区间内浮动。流体升力系数幅值 $|C_l|_{max}$ 的相对误差由 10.7% 降低到 3.0%，流体阻力系数均值 $\overline{C_d}$ 的相对误差由 9.7% 降低到 4.5%。

(4) 完成相同的训练任务时，强耦合 PINN 模型可以节省大量的数据成本和时间成本。本章的计算工况中，强耦合 PINN 模型所需的数据成本仅为弱耦合 PINN 模型的 1/2，计算时间成本仅为弱耦合 PINN 模型的 3/5。

第 4 章 基于物理驱动深度学习的涡激振动正反问题求解

4.1 引言

VIV 是由流体质量守恒方程、流体动量守恒方程以及结构动力学方程等偏微分方程控制的非线性流固耦合问题，PINN 可以作为求解这类方程的通用函数求解器，同时逼近 NPDE 的理论解析解以及模型的未知参数。本章将基于上文所建立的强耦合 PINN 模型对 VIV 时空预测（正问题）和结构参数识别（反问题）的目标任务进行验证。在正问题中，提出分时段递推训练的方法，采用分时段不同训练配置点采样策略，仅利用有限时域长度信息来预测更长时间段的 VIV 信息，实现“稀疏”数据条件下流场运动信息的时空域预测以及结构振动信息的时域预测。在反问题中，通过训练样本数据，反向识别结构的未知参数。

4.2 涡激振动正问题

正问题是通过改进训练方法来实现 VIV 时空序列的预测，包括流场运动信息的时空域预测和结构振动信息的时域预测。训练模型为上一章所建立的强耦合 PINN 模型：输入层变量为时间 t 和空间坐标 (x, y) ，输出层变量为流场质点的顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 、压力 $p(t, x, y)$ 以及结构振动位移 $r(t)$ ，添加的物理信息为二维 N-S 方程和流固耦合条件方程。

在实际应用中，受限于监测技术及设备，VIV 监测数据往往具有“小样本、稀疏”等特征；同时，VIV 是典型的非线性问题，由于误差累积，长时空距的预测精度会大大降低。为了提高在“稀疏”数据条件下求解正问题的计算效率和精度，本章提出了分时段递推训练的方法来优化模型训练过程。

所谓分时段递推训练，首先利用已知测量的有限时域长度的数据来训练神经网络，然后基于此网络进行有限时间步长的预测，完成第一个时间段的训练和预测；然后把上一次预测的时段信息保留到下一个时段（由已知有限长度时段加上所有预测时段）进行训练和预测。同时在不同的训练时段采用数量不同的空间配置点（本文采样递减方式）。依此类推，参与训练数据的时域越来越长，而空间配置点越来越少。采用此策略，有效利用预测信息，减少空间配置点的要求，提高了计算效率，实现了仅利用有限时域长度信息（初始训练集）来预测更长时间

段的 VIV 信息。

接下来将以正问题数值算例为例具体说明训练集的递推训练和预测流程，如图 4.1 所示。第三章的 CFD 模拟数据被用来构建正问题数值算例的训练集和测试集。初始训练集是第 0-2 s 的 VIV 信息，时间间隔为 0.05 s，时间步数为 40。该算例将基于此，对后续第 2.05-5 s 的 VIV 信息进行预测，共包含 60 个时间步。

首先，基于强耦合 PINN 模型对初始训练集进行训练，预测第 2.05 和 2.10 s 的 VIV 信息，完成第 1 次训练和预测。

然后，更新训练集为第 2 次预测做准备，将第一次预测的第 2.05 和 2.10 s 的 VIV 信息添加至训练集中，延长训练集的时域长度至 0-2.10 s。然后重复训练 PINN 模型来预测第 2.15 和 2.20 s 的 VIV 信息，完成第 2 次训练和预测。

接下来，再次更新训练集为第 3 次预测做准备，将第 2 次预测的第 2.15 和 2.20 s 的 VIV 信息添加至训练集中，继续延长训练集的时域长度至 0-2.20 s。然后重复训练 PINN 模型来预测第 2.25 和 2.30 s 的 VIV 信息，完成第 3 次训练和预测。

最后，依照上述步骤进行递推训练，将预测的 VIV 信息不断添加至初始训练集，持续延长训练集的时域长度，直至完成测试集所有时段的预测，具体的预测流程如图 4.1 所示。

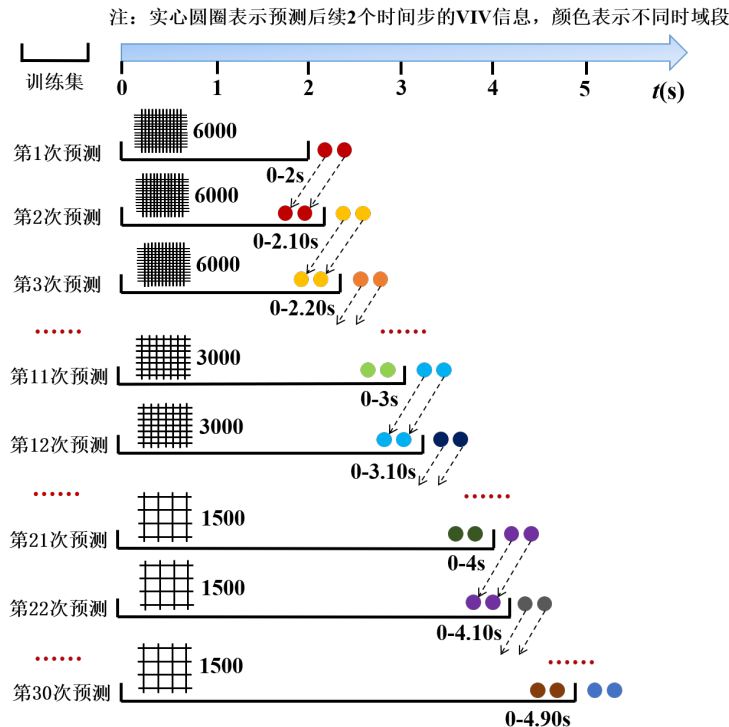


图 4.1 递推训练和预测流程

如图 4.1 所示，该算例中将采用分时段配置点采样策略来调整不同预测批次

中样本集的空间配点数量，以模拟“稀疏”空间环境，如表 4.1 所示。预测第 2.05 至 3 s 的 VIV 信息时，流场空间配点数量为 6000；预测第 3.05 至 4 s 的 VIV 信息时，流场空间配点数量为 3000；预测第 4.05 至 5 s 的 VIV 信息时，流场空间配点数量为 1500。

表 4.1 不同预测批次的空间配点数量

预测时段	流场空间配点数量
第 2.05-3 s	6000
第 3.05-4 s	3000
第 4.05-5 s	1500

针对不同的空间配点数量，神经网络的训练参数也随之调整，以确保更好的计算效率和精度。预测第 2.05-3 s 的 VIV 信息时，神经网络训练参数设置如表 4.2 所示；预测第 3.05-4 s 的 VIV 信息时，神经网络训练参数设置如表 4.3 所示；预测第 4.05-5 s 的 VIV 信息时，神经网络训练参数设置如表 4.4 所示。

表 4.2 神经网络训练参数设置（空间配点数量为 6000）

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	10000	1.00E-03	$f(x) = \sin x$
第 2 阶段	200	10000	1.00E-04	
第 3 阶段	200	10000	1.00E-05	
第 4 阶段	100	10000	1.00E-06	

表 4.3 神经网络训练参数设置（空间配点数量为 3000）

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	9000	1.00E-03	$f(x) = \sin x$
第 2 阶段	200	9000	1.00E-04	
第 3 阶段	200	9000	1.00E-05	

表 4.4 神经网络训练参数设置（空间配点数量为 1500）

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	8000	1.00E-03	$f(x) = \sin x$
第 2 阶段	200	8000	1.00E-04	
第 3 阶段	100	8000	1.00E-05	

4.3 正问题算例结果分析

4.3.1 流场运动信息预测情况

计算该算例所使用的硬件设备为 HP Zhan 86 Pro G1 MT，处理器为 Intel(R) Core(TM) i7-7700，内存为 8 GB，操作系统为 64 位。表 4.5 是在不同神经网络训练参数设置下，完成一次训练所花费的平均时间成本。预测第 2.05-3 s 的 VIV 信息时，一次训练所花费的平均时间成本为 5.43 h；预测第 3.05-4 s 的 VIV 信息时，一次训练所花费的平均时间成本为 4.12 h；预测第 4.05-5 s 的 VIV 信息时，一次训练所花费的平均时间成本为 2.96 h。由此可见，渐稀疏化空间配点采样策略可以节省大量的时间成本，有效提高 PINN 模型的计算效率。

表 4.5 不同预测批次完成一次训练所花费的时间成本

预测时段	流场空间配点数量	一次训练的 Epoch	平均时间成本 (h)
第 2.05-3 s	6000	700	5.43
第 3.05-4 s	3000	600	4.12
第 4.05-5 s	1500	500	2.96

图 4.2 为不同神经网络训练参数设置下神经网络损失函数 $Loss$ 的变化曲线，损失函数 $Loss$ 的表达式为式 (3.12)。由图可知，所有的训练轮次均在前 200 次 Epoch 内趋于稳定，逐渐收敛至阈值范围内，且最终的下降阈值基本相近。

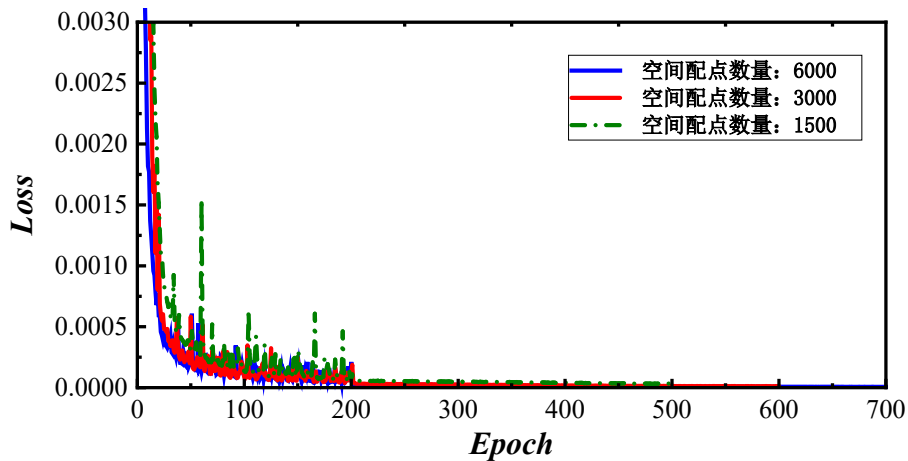


图 4.2 损失函数 $Loss$ 的变化曲线

该算例的测试集为第 2-5 s 的 VIV 模拟数据，共包含 60 个时间步。计算结果表明基于逐步迭代训练的方法可以准确完成流场运动信息的时空预测。图 4.3 至图 4.6 分别第 2.05 s、第 3 s、第 4 s 以及第 4.95 s 的流场运动信息时空预测结

果以及相应的预测残差。图 4.7 是流场运动信息时空预测结果的 RMSE。下文图表里的所有变量均采用国际标准单位。

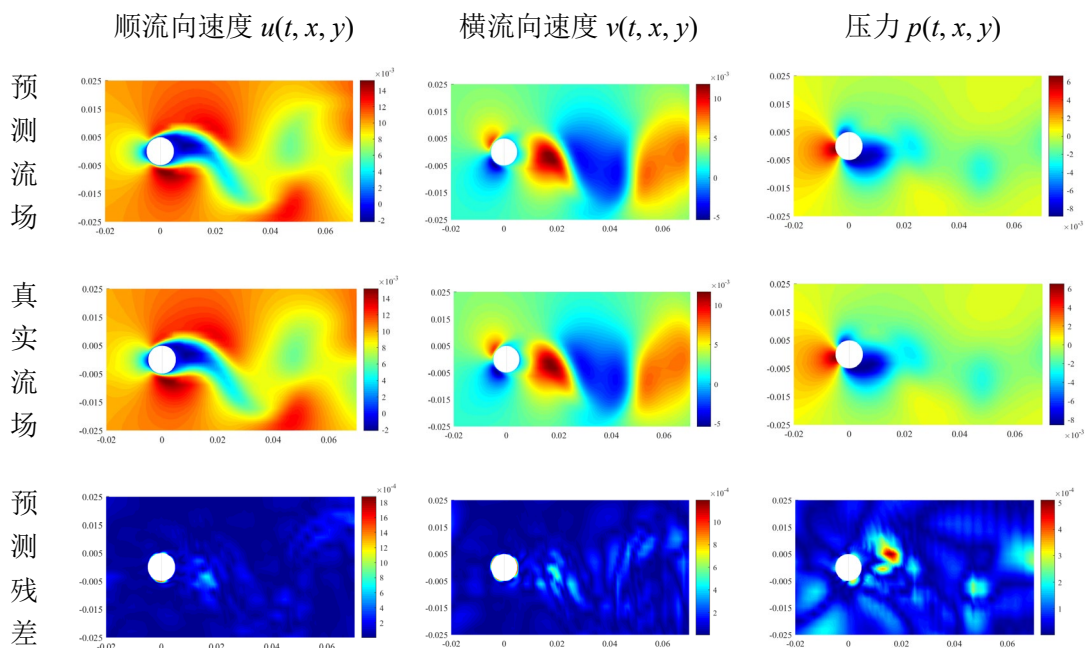


图 4.3 第 2.05 s 流场运动信息时空预测结果

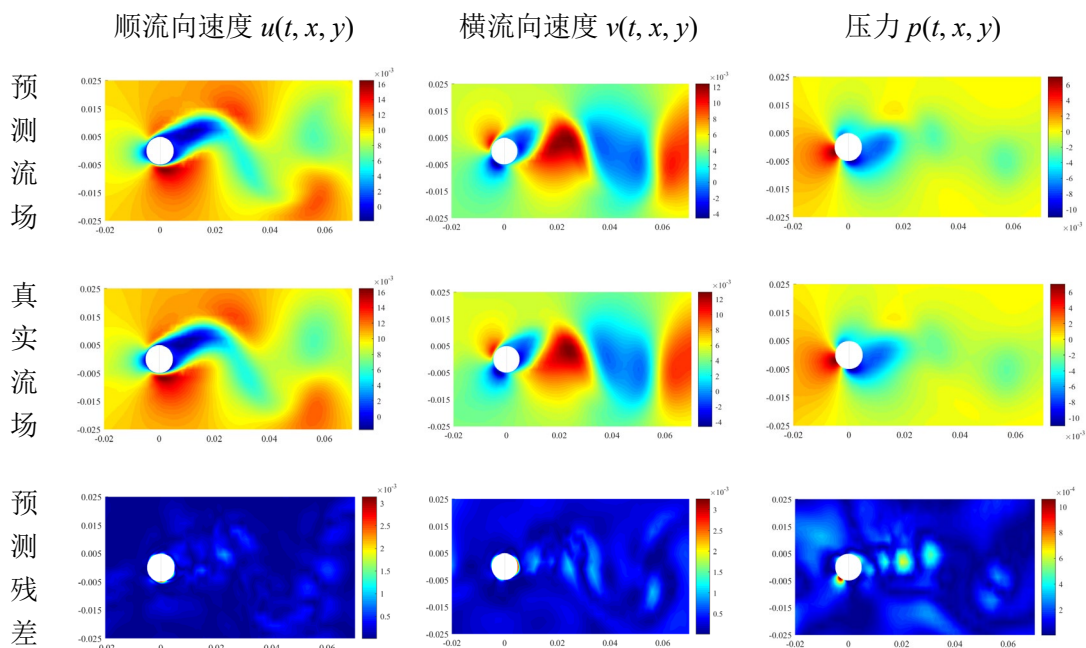


图 4.4 第 3 s 流场运动信息时空预测结果

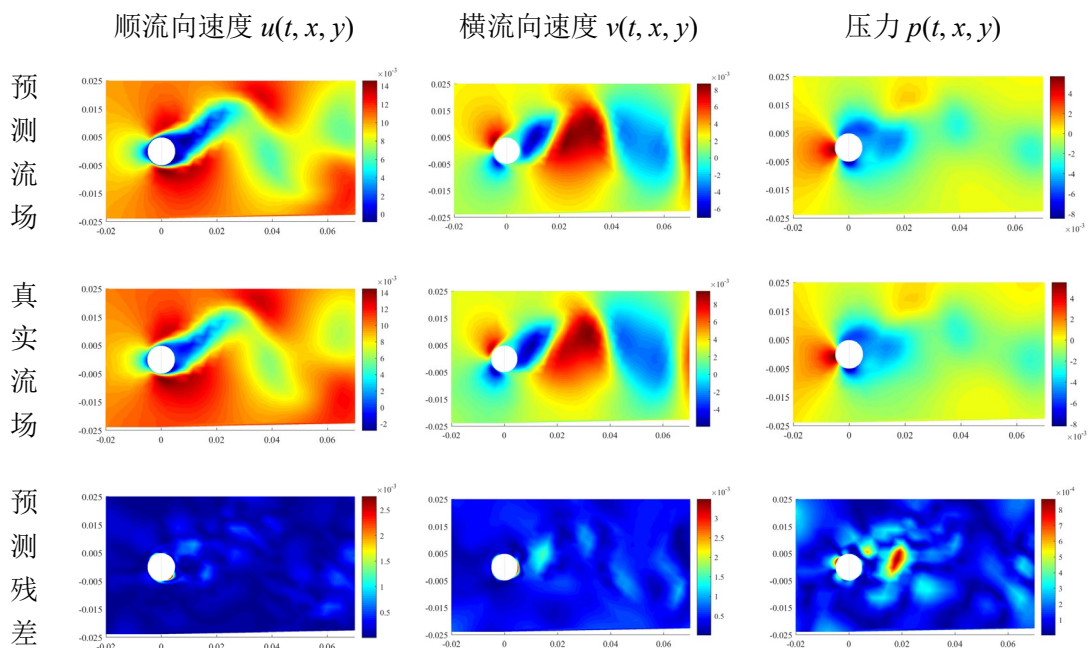


图 4.5 第 4 s 流场运动信息时空预测结果

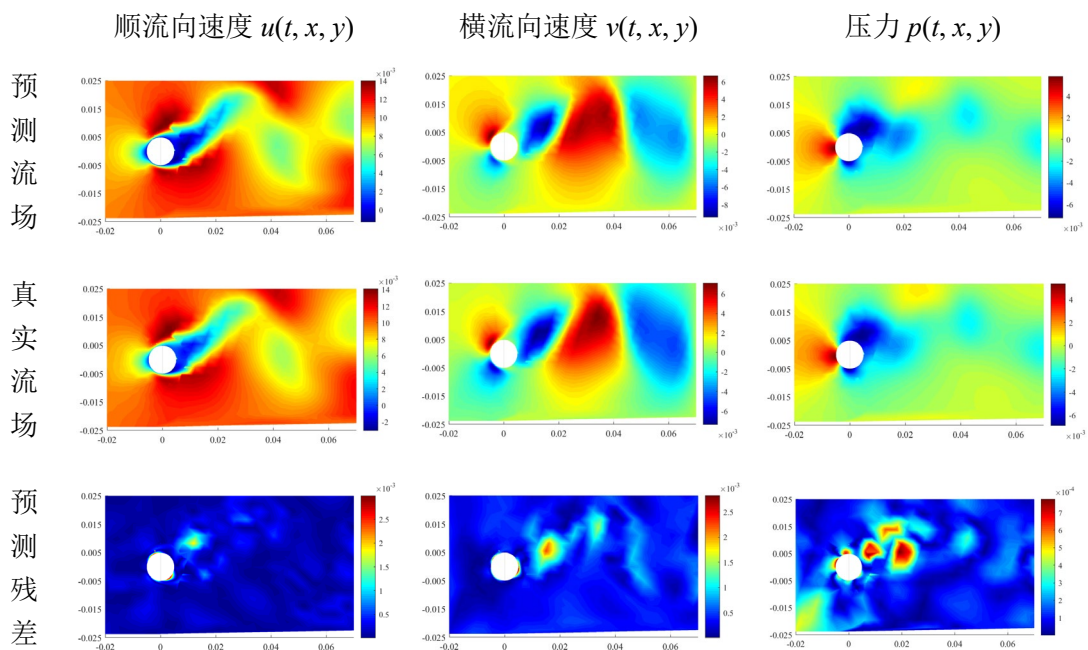


图 4.6 第 4.95 s 流场运动信息时空预测结果

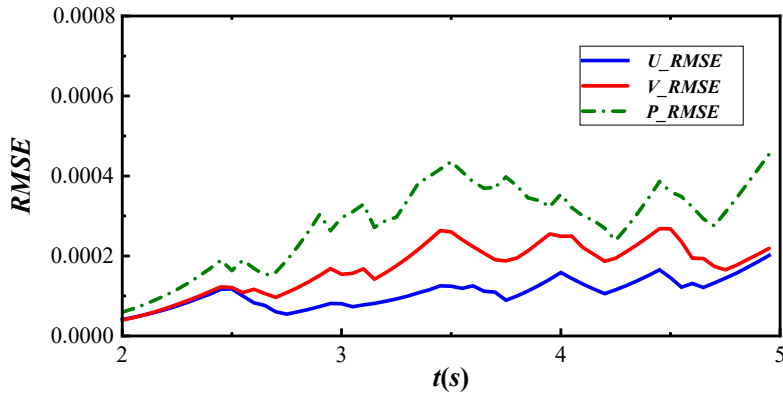


图 4.7 流场运动信息时空预测结果 RMSE

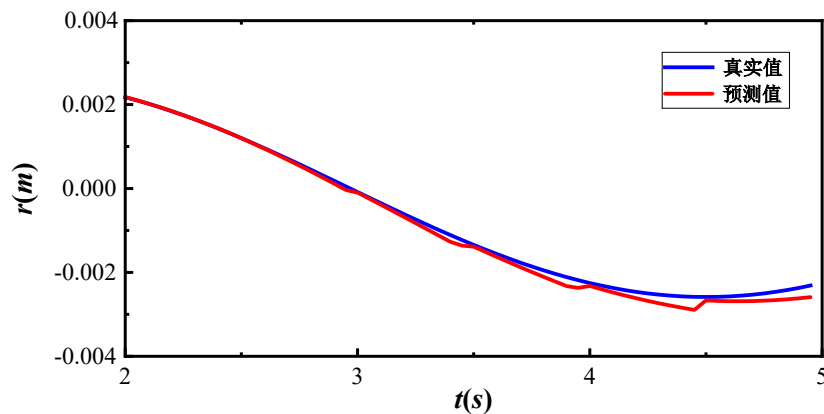
对于顺流向速度 u ，预测结果的 RMSE 浮动区间是 $(4.1\text{E-}05, 2.0\text{E-}04)$ ；对于横流向速度 v ，预测结果的 RMSE 浮动区间是 $(4.0\text{E-}05, 2.7\text{E-}04)$ ；对于压力 p ，预测结果的 RMSE 浮动区间是 $(5.9\text{E-}05, 4.6\text{E-}04)$ 。

综合来看，该算例将顺流向速度 u 、横流向速度 v 以及压力 p 预测结果的 RMSE 分别控制在 $2.0\text{E-}04$ 、 $3.0\text{E-}04$ 以及 $5.0\text{E-}04$ 以下。参考实际流场数据信息，基于渐稀疏化空间配点采样策略和逐步迭代训练的方法，该算例在“稀疏”数据条件下实现了流场运动信息的时空域预测，同时保持有良好的计算效率和精度。

4.3.2 结构振动信息预测情况

结构振动信息主要包括固体结构振动位移 r 、绕流升力 F_l 以及绕流阻力 F_d 。

该算例的测试集为第 2-5 s 的 VIV 信息，需要预测 60 个时间步的固体结构振动位移 r 。图 4.8 是该算例对固体结构振动位移 r 的预测结果，图 4.9 是预测结果的 RMSE。

图 4.8 固体结构振动位移 r 的预测结果

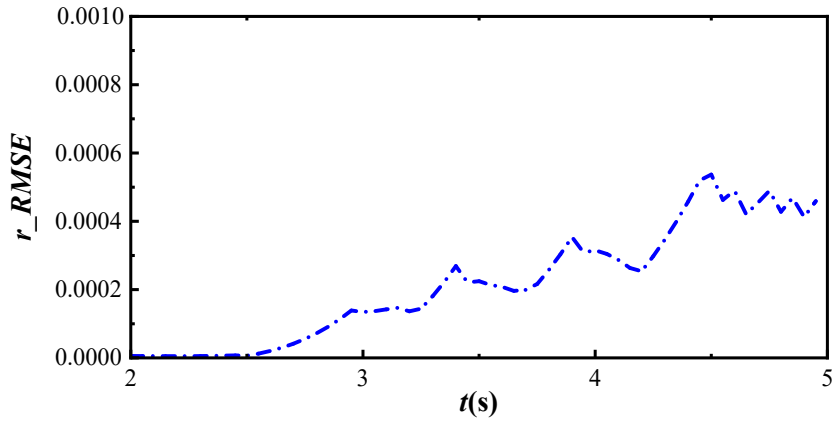
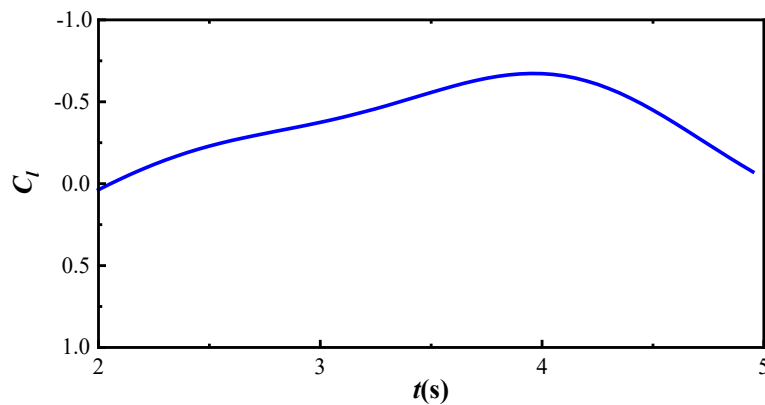


图 4.9 固体结果振动位移 r 预测结果 RMSE

对于固体结构振动位移 r , 预测结果的 RMSE 浮动区间为 $(5.0E-06, 5.4E-04)$ 幅值为 $5.4E-04$ 。在单个时间步内, 预测结果的最大相对误差为 9.73%。从图 4.9 可以看出, 随着时间序列的增长, 固体结构振动位移的预测误差在整体上呈上升趋势, 但仍控制在合理误差范围内。

基于流场运动信息的再现结果和流固耦合面的边界条件, 可以积分计算得到绕流升力 F_l 和绕流阻力 F_d , 积分表达式为式 (3.5) 和式 (3.6)。为了更直观地验证该算例预测绕流力的准确性, 通过式 (3.22) 和式 (3.23) 将绕流升力 F_l 和绕流阻力 F_d 转化为绕流升力系数 C_l 和绕流阻力系数 C_d , 如图 4.10 所示。最后以绕流升力系数幅值 $|C_{l,max}|$ 和绕流阻力系数均值 $\overline{C_d}$ 为指标, 将预测结果与第三章 CFD 模拟结果进行对比, 如表 4.6 所示。



(a) 流体升力系数 C_l

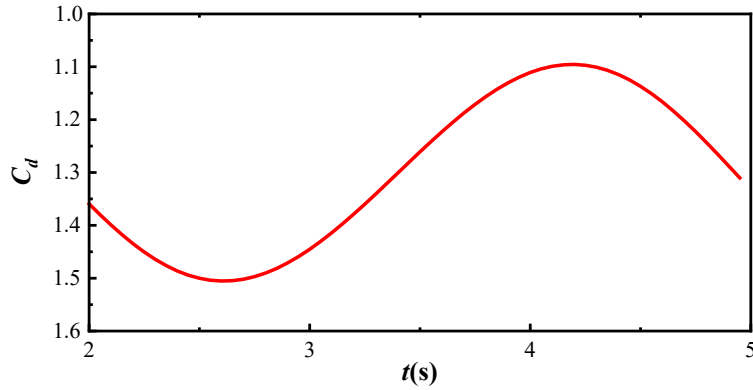
(b) 流体阻力系数 C_d

图 4.10 流体力系数的预测结果

表 4.6 流体力系数预测结果正确性验证

方法	预测结果 $ C_l _{max}$	数值模拟 $ C_l _{max}$	误差	预测结果 $\overline{C_d}$	数值模拟 $\overline{C_d}$	误差
逐步迭代 训练	0.69	0.66	0.03	1.30	1.34	0.04

对于流体升力系数幅值 $|C_l|_{max}$ ，该算例的预测结果为 0.69，CFD 数值模拟结果为 0.66，预测结果的相对误差为 4.5%。

对于流体阻力系数均值 $\overline{C_d}$ ，该算例的预测结果为 1.30，CFD 数值模拟结果为 1.34，预测结果的相对误差为 3.0%。

参考实际的结构振动信息以及 CFD 数值模拟结果，基于渐稀疏化空间配点采样策略和逐步迭代训练的方法，该算例在“稀疏”数据条件下实现了结构振动信息的时域预测，同时保持有良好的计算效率和精度。

4.4 涡激振动反问题

反问题的本质是寻找描述神经网络模型回归映射规律的最优未知参数。在 VIV 过程中，阻尼 c 、刚度 k 等结构参数的退化情况需要重点关注，这些结构参数直接影响柔性结构的服役状态。本文将基于第三章所建立的强耦合 PINN 模型训练样本数据，从而反向识别结构阻尼 c 和结构刚度 k 。

在反问题训练过程中，待识别的结构参数与神经网络的权重矩阵 W 和偏置系数 b 同时进行迭代更新，损失函数可以写作 $Loss(W, b, c, k)$ 。参考式 (2.40)，为了简化计算，将该损失函数对网络输出变量 z^l 的偏导记作 θ^l ，如式 (4.1) 所示。

$$\theta^l = \frac{\partial \text{Loss}(W, b, c, k)}{\partial z^l} \quad (4.1)$$

神经信号反向传播过程中，将依次求解对权重矩阵 W 、偏置系数 b 以及待识别参数的梯度，从而基于梯度下降算法对参数进行迭代更新。这些参数的梯度求解公式依次为式 (4.2) 至式 (4.5)。

$$\frac{\partial \text{Loss}(W, b, c, k)}{\partial W^l} = \frac{\partial \text{Loss}(W, b, c, k)}{\partial z^l} \frac{\partial z^l}{\partial W^l} = \theta^l \frac{\partial z^l}{\partial W^l} \quad (4.2)$$

$$\frac{\partial \text{Loss}(W, b, c, k)}{\partial b^l} = \frac{\partial \text{Loss}(W, b, c, k)}{\partial z^l} \frac{\partial z^l}{\partial b^l} = \theta^l \frac{\partial z^l}{\partial b^l} \quad (4.3)$$

$$\frac{\partial \text{Loss}(W, b, c, k)}{\partial k} = \frac{\partial \text{Loss}(W, b, c, k)}{\partial z^l} \frac{\partial z^l}{\partial k} = \theta^l \frac{\partial z^l}{\partial k} \quad (4.4)$$

$$\frac{\partial \text{Loss}(W, b, c, k)}{\partial c} = \frac{\partial \text{Loss}(W, b, c, k)}{\partial z^l} \frac{\partial z^l}{\partial c} = \theta^l \frac{\partial z^l}{\partial c} \quad (4.5)$$

由上述推导结果可得，这些参数的梯度求解结果均与 θ^l 密切相关。 θ^l 与 θ^{l+1} 的递推式可以由数学归纳法推导得出，从而解得每个网络层的 θ^l 赋值。将其带入式 (4.2) 至式 (4.5)，即可获得权重矩阵 W 、偏置系数 b 、结构阻尼 c 以及结构刚度 k 的梯度求解结果，进而对神经网络进行迭代更新。

第三章的 CFD 模拟数据被用来构建反问题数值算例的训练集，如表 4.7 所示。训练集是第 0-10 s 的 VIV 信息，时间间隔为 0.1 s，时间步数为 100，每个时间步的空间配点数量为 3000。相应的神经网络训练参数设置如表 4.8 所示。

表 4.7 反问题的训练集

训练集	时间间隔	训练集的时间步数	每个时间步的空间配点数量	训练模型
0-10 s	0.1 s	100	3000	强耦合 PINN 模型

表 4.8 反问题的神经网络训练参数设置

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	10000	1.00E-03	$f(x) = \sin x$
第 2 阶段	300	10000	1.00E-04	
第 3 阶段	200	10000	1.00E-05	
第 4 阶段	300	10000	1.00E-06	

4.5 反问题算例结果分析

计算该数值算例所使用的硬件设备为 HP Zhan 86 Pro G1 MT，处理器为 Intel(R) Core(TM) i7-7700，内存为 8 GB，操作系统为 64 位。图 4.11 为反问题的神经网络损失函数 $Loss$ 的变化曲线，损失函数 $Loss$ 的表达式为式 (3.12)。

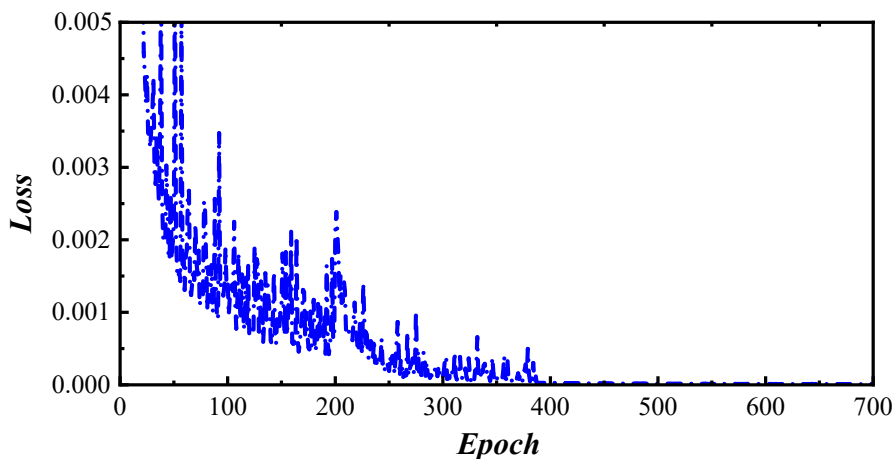


图 4.11 反问题神经网络损失函数 $Loss$ 的变化曲线

由图 4.11 可知，反问题的损失函数约在第 320 次 Epoch 处开始趋于稳定，逐渐收敛至阈值范围内。图 4.12 和图 4.13 分别为结构阻尼 c 和结构刚度 k 在网络训练过程中的迭代曲线。由图可得，结构阻尼 c 的迭代曲线总体呈先上升后下降的趋势，最后稳定于 0.0563，识别出的结构阻尼值 $c=0.0563$ ；结构刚度 k 的迭代曲线总体呈先下降后上升的趋势，最后稳定于 16.7841，识别出的结构刚度值 $k=16.7841$ 。

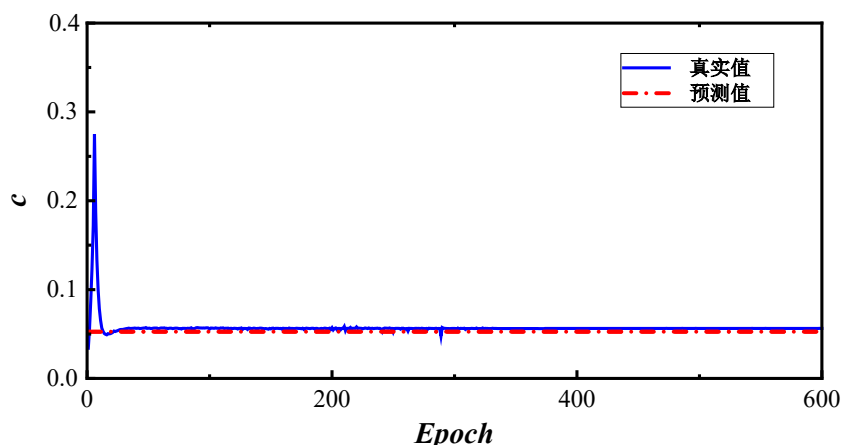


图 4.12 结构阻尼 c 的迭代曲线

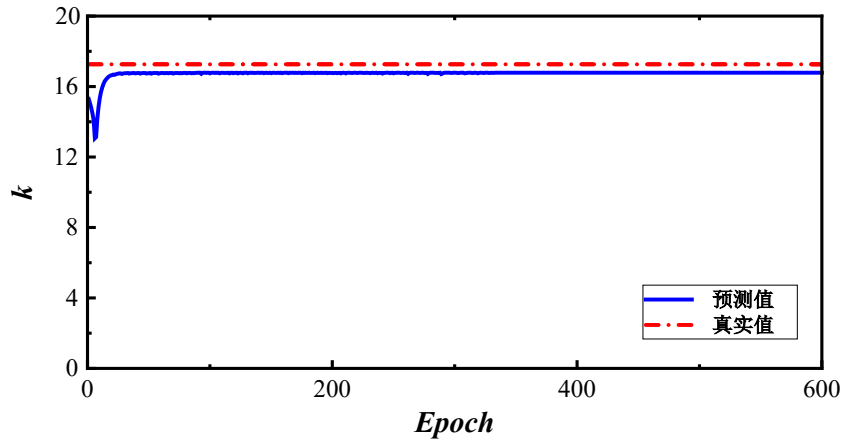

 图 4.13 结构刚度 k 的迭代曲线

表 4.9 结构参数的识别误差

结构阻尼 c			结构刚度 k		
真实值	识别值	识别误差	真实值	识别值	识别误差
0.0526	0.0563	7.03%	17.2603	16.7841	2.76%

表 4.9 为结构阻尼 c 和结构刚度 k 的识别误差。第三章 CFD 数值模拟中设置的结构质量 m 为 2.7325，固有频率 ω 为 2.5133，阻尼比 ζ 为 0.00383，计算可得结构阻尼 c 为 0.0526，结构刚度 k 为 17.2603。结构阻尼 c 和结构刚度 k 的相对识别误差分别为 7.03% 和 2.76%，识别误差均在合理范围内。由此可得，**该数值算例可以准确识别阻尼 c 、刚度 k 等结构参数。**

4.6 本章小结

本章提出分时段递推训练的方法，采用分时段不同训练配置点采样策略，仅利用有限时域长度信息来预测更长时间段的 VIV 信息，有效提高求解 VIV 正反问题的计算效率及精度。通过对正反问题的数值算例进行分析，证明该方法可以在“稀疏”数据条件下实现流场运动信息的时空预测以及结构振动信息的时域预测，同时准确识别阻尼 c 以及刚度 k 等结构参数。本章的主要结论如下所示：

(1) 基于分时段递推训练的方法，可以充分利用预测信息，减少空间配置点的要求，有效提高模型计算效率，实现了仅利用有限时域长度信息来预测更长时间段的 VIV 信息。

(2) 由正问题数值算例结果可得，分时段递推训练的方法可以在“稀疏”数据条件下实现流场运动信息的时空预测以及结构振动信息的时域预测，同时具有良好的计算效率及精度。

对于顺流向速度 u ，单个时间步的预测结果 RMSE 的浮动区间为 (4.1E-05,

2.0E-04); 对于横流向速度 v , 单个时间步的预测结果 RMSE 的浮动区间为(4.0E-05, 2.7E-04); 对于压力 p , 单个时间步的预测结果 RMSE 的浮动区间为 (5.9E-05, 4.6E-04)。

对于固体结构振动位移 r , 单个时间步的最大相对预测误差为 9.73%; 对于流体升力系数幅值 $|C_l|_{max}$, 相对预测误差为 4.5%; 对于流体阻力系数均值 $\overline{C_d}$, 相对预测误差为 3.0%。

(3) 由反问题数值算例结果可得, 该方法可以准确识别阻尼 c 以及刚度 k 等结构参数。

结果阻尼 c 的相对识别误差为 7.03%, 结构刚度 k 的相对识别误差为 2.76%。

第 5 章 基于迁移学习的涡激振动物理信息神经网络模型

5.1 引言

传统的基于 FEM 和纯数据驱动 DNN 的 VIV 分析方法经常面临计算成本和数据集获取的挑战。本章提出了一种基于 TL 增强的 PINN 模型来研究 VIV，也被称为深度迁移学习（deep transfer learning, DTL）代理模型。当与 TL 方法结合使用时，PINN 通过源模型中的公共特征知识提高学习效率并保持目标任务的可实现性，而无需标注大量数据集。分别将 VIV 的 CFD 模拟数据集和实验数据集划分为两部分（源域和目标域），以评估 DTL 代理模型完成小样本目标任务的性能。结果表明，尽管在训练模型过程中获得的数据量逐渐变小，该方法与使用传统 PINN 算法的结果非常吻合。该模型的应用可以突破实际工程中监测设备和方法的局限性，促进 VIV 研究的深入。

5.2 迁移学习概述

5.2.1 迁移学习背景介绍

传统的数据挖掘和机器学习算法均通过统计模型对未来的数据进行预测，这些统计模型往往基于同分布假设对大量标注的样本数据进行训练。但是，标注大量样本数据对于训练成本的控制是很大的挑战。同时样本数据集的差异化分布以及标注过期，也会导致训练成本的增加。因此 TL 方法开始被提出，用于对已标注样本数据的再利用，同时保证统计模型在其余任务中的高精度。

TL 方法是通过寻找多任务之间的潜在共有特征空间，从一个或者多个学习对象中提取知识源任务，并将特征知识应用于目标任务^[117]。与传统监督式机器学习不同，在 TL 方法中源任务和目标任务的角色不再保持对称。TL 方法不再同时学习所有的源任务和目标任务，而是将中心转移到目标任务的完成。面对多任务学习时，传统监督式机器学习和 TL 方法展现了完全不同的学习方法，如图 5.1 所示。传统监督式机器学习试图从零开始学习每一个任务，需要重复标注大量的样本数据。TL 方法通过共享源任务和目标任务之间的特征信息，将源任务

的知识转移到目标任务中，从而以较少的标注样本数据高质量地贴合统计模型。两种技术之间的差异充分证明了 TL 方法降低训练成本的能力。

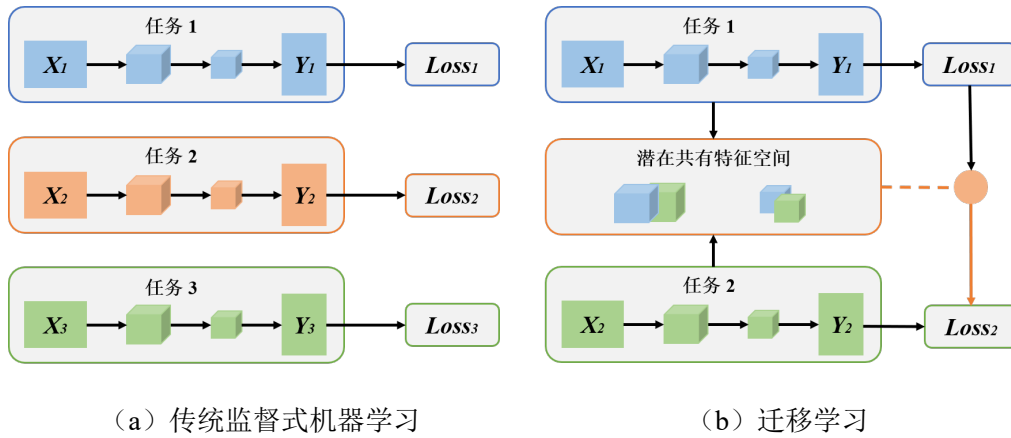


图 5.1 传统监督式机器学习和迁移学习完成多任务学习的差异

5.2.2 迁移学习形式定义及分类

TL 方法中有两个关键的概念：域 (Domain) 和任务 (Task)。域 (Domain) 的表示符号为 D ，包含有两个部分：特征空间 χ 和边际概率分布 $P(X)$ ，其中 $X = \{x_1, x_2, \dots, x_n\} \in \chi$ ，即 $D = \{\chi, P(X)\}$ 。任务的表示符号为 Γ ，包含有两个部分：标签空间 γ 和客观预测函数 $f(\cdot)$ ，即 $\Gamma = \{\gamma, f(\cdot)\}$ 。

本文提出的 TL 方法主要考虑两个域：源域 D_S 和目标域 D_T 。为了进一步表示，源域数据可以被写作： $D_S = \{(x_{S1}, y_{S1}), \dots, (x_{Sn}, y_{Sn})\}$ ，这里 $x_{Si} \in \chi_S$ 代表数据实例， $y_{Si} \in \gamma_S$ 代表相应的类标签。同样的，目标域数据可以被写作： $D_T = \{(x_{T1}, y_{T1}), \dots, (x_{Tm}, y_{Tm})\}$ ，这里 $x_{Ti} \in \chi_T$ 是输入量， $y_{Ti} \in \gamma_T$ 是相应的输出量，通常 $0 \leq m \ll n$ 。基于上述概念，TL 方法被定义为如下：源域 D_S 中的学习任务为 Γ_S ，目标域 D_T 中的学习任务为 Γ_T ，TL 方法的最终目的是利用 D_S 和 Γ_S 的特征知识来帮助提高 D_T 中目标预测函数 $f(\cdot)$ 的学习。并且 $D_S \neq D_T$ 或者 $\Gamma_S \neq \Gamma_T$ 。

TL 方法主要分为两类：同构迁移学习 (Isomorphic transfer learning, ITL) 和异构迁移学习 (Heterogeneous transfer learning, HTL)。其中 ITL 方法分为异构特征空间学习 (Cross-modal learning, CML) 和异构类别空间学习 (Cross-category learning, CCL)。HTL 方法分为数据集偏移学习 (Dataset shift learning, DSL)、领域适配学习 (Domain adaptation learning, DAL) 和多任务学习 (Multi-task learning, MTL)。具体的分类前提条件如图 5.2 所示。

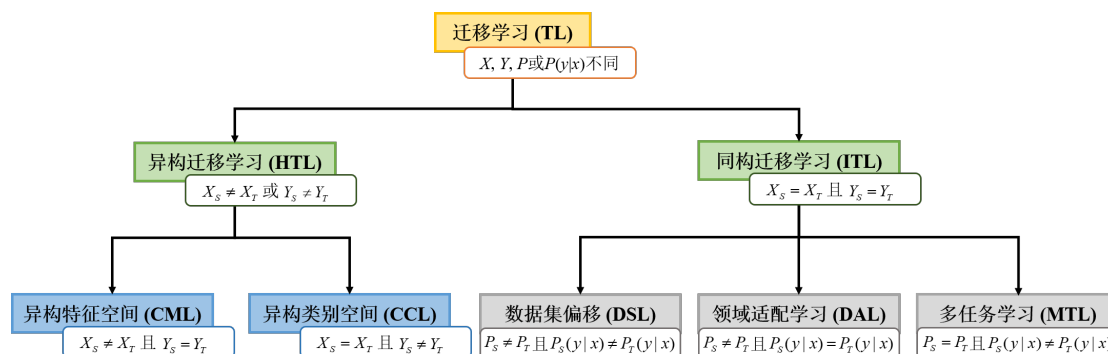


图 5.2 迁移学习的形式分类

5.2.3 常见的迁移学习方法

目前常见的 TL 方法主要包括七类：领域自适应迁移学习^[118] (Domain adaptation transfer learning, DATL)、在线迁移学习^[119] (Online transfer learning, OTL)、终身迁移学习^[120] (Lifelong transfer learning, LTL)、异构迁移学习^[121] (Heterogeneous transfer learning, HTL)、深度迁移学习^[122] (Deep transfer learning, DTL)、强化迁移学习^[123] (Reinforcement transfer learning, RTL) 和对抗迁移学习^[124] (Adversarial transfer learning, ATL)。

(1) 领域自适应迁移学习

传统的领域自适应方法大多依赖于向量子空间的特征映射，容易导致空间结构的多线性问题表达不清楚，无法正确地传递特征知识。DATL 方法通过引入一组对齐矩阵，来将源域和目标域的知识以张量形式对齐到不变张量子空间中，从而避免了特征知识向量化。如图 5.3 所示，与向量子空间相比，张量子空间由多组子空间构成，每组子空间分别代表不同的模态。这种高阶张量建模空间通过研究不同模态之间的相互耦合作用，来捕捉不同域之间的共性与差异，从而更好地共享特征知识。

DATL 方法基于学习的方法直接解决由对齐矩阵和张量子空间建模成的联合优化问题，并通过提出的替代最小化方案从样本数据集中自适应地学习。更具体地说，DATL 方法将来自不同源域的样本数据集特征知识映射到同一张量特征空间中，以此为基础来加强目标域的样本数据集训练。DATL 方法并不要求源域和目标域具有相近的数据集分布式，适用于目标域数据较少或源域和目标域分布不一致的迁移学习模型。

大量的实验表明，DATL 方法可以保持很强的源域鉴别能力，不仅可以抵抗标签噪声的影响，而且在小容量目标域样本中仍然有效。DATL 方法与深度 CNN 模型相结合时，神经网络的领域自适应性能被高度激活，此时表现性能优于所有

的比较器。

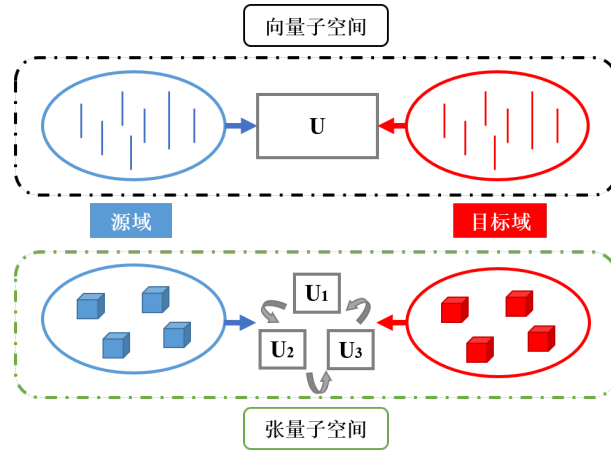


图 5.3 向量空间与张量子空间特征学习对比

(2) 在线迁移学习

OTL 方法将在线学习与迁移学习相结合，旨在完成将知识从源域转移到目标域的在线学习任务。OTL 方法并不假设源域数据和目标域数据符合相同的分类或分布式。OTL 方法中，源域的样本数据集是随着时间推移不断变化的，在迁移学习前并非是确定的。在线学习的动机是通过利用源域数据训练中得到的特征知识，来改进目标域样本数据集的监督式在线学习任务。OTL 方法充分理解了源域数据的分布特征，以便达到更好的迁移效果。

OTL 方法的学习框架包括两类：同构 OTL 方法和异构 OTL 方法。其中同构 OTL 方法适用于源域和目标域共享相同的特征空间的情况，异构 OTL 方法适用于源域和目标域特征空间不同的情况。表 5.1 梳理了 OTL 方法的算法逻辑。

$$y_{2_t} = \text{sign} \left(\omega_{1,t} \prod (h(x_{2_t})) + \omega_{2,t} \prod (f_t(x_{2_t})) - \frac{1}{2} \right) \quad (5.1)$$

$$\omega_{1,t+1} = \frac{\omega_{1,t} * s_t(h)}{\omega_{1,t} * s_t(h) + \omega_{2,t} * s_t(f_t)} \quad (5.2)$$

$$\omega_{2,t+1} = \frac{\omega_{2,t} * s_t(f_t)}{\omega_{1,t} * s_t(h) + \omega_{2,t} * s_t(f_t)} \quad (5.3)$$

$$\tau_t = \min \left\{ C, \frac{\text{loss}_t}{k_2(x_{2_t}, x_{2_t})} \right\} \quad (5.4)$$

$$f_{t+1} = f_t + \tau_t y_{2_t} k_2(x_{2_t}, \cdot) \quad (5.5)$$

表 5.1 在线迁移学习的算法逻辑

<p>输入量</p>	<p>旧的分类器: $h(x) = \sum_{s=1}^S \alpha_s y_{1s} k_1(x_{1s}, x)$;</p> <p>初始权衡 C 以及初始权重: $\omega_{1,1} = \omega_{2,1} = \frac{1}{2}$。</p>
<p>具体算法</p>	<p>初始化 $f_1=0$;</p> <p>$t=1, 2, 3, \dots, T, x_{2t} \in X_2$;</p> <p>利用式 (5.1) 预测 y_{2t}, 同时获取正确的类标签: $y_{2t} \in \{-1, 1\}$;</p> <p>利用式 (5.2) 和式 (5.3) 计算 $\omega_{1,t+1}$ 和 $\omega_{2,t+1}$;</p> <p>计算损失函数: $loss_t = 1 - y_{2t} * f_t(x_{2t})$;</p> <p>如果 $loss_t > 0$, 然后利用式 (5.4) 和式 (5.5) 进行迭代计算。</p>

(3) 终身迁移学习

LTL 方法突破了单一任务领域内进行迁移学习的限制。通过终身学习优化可转移特征知识的共享存储库, 同时建立组投影矩阵, LTL 代理模型能够在不同的任务领域内自主地转移特征知识, 从而真正实现全方位的终身学习。如图 5.4 所示, 为了实现跨领域知识共享, 特征知识储存库 L 被引入, 在所有任务及任务域之间共享。特征知识储存库 L 代表在群特定矩阵集合 $\{B^{(1)}, B^{(2)}, \dots, B^{(Gmax)}\}$ 下的一组潜在因素。组投影矩阵 $\{\psi^{(g^{(1)})}, \psi^{(g^{(2)})}, \dots, \psi^{(g^{(n)})}, \dots, \psi^{(g^{(Gmax)})}\}$ 将共享的潜在因素 L 映射为每组任务相应的基础投影域, 即满足 $B^{(g)} = \Psi^{(g^{(n)})} L$ 。通过这种框架构造, 这些特定于组的基础投影域通过映射 Ψ 同特征知识储存库 L 耦合起来, 促进具有不同特征的任务域转移空间的形成。同时基于群特定矩阵的映射还有助于避免过拟合, 最大限度地在任务组内进行传递。相关实验表明, LTL 方法能够有效地从交叉的任务域中学习, 并在新的任务域中快速地获得高性能表现。

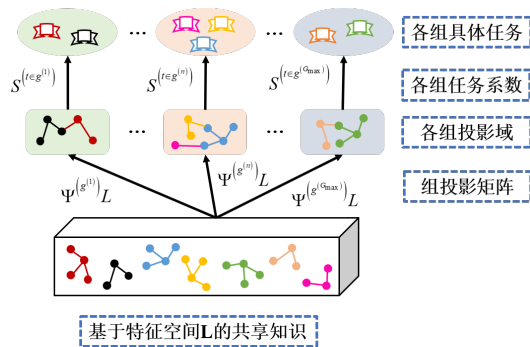


图 5.4 终身迁移学习的知识共享框架

(4) 异构迁移学习

HTL 方法实现了具有不同特征维度的源域和目标域样本数据集之间的迁移学习, 极大地降低了迁移学习样本数据集的构造限制, 适用于目标域数据极度稀

疏或缺乏的情况。HTL 方法基于一种新的无监督异构域自适应方法来弥补源域和目标域样本数据集之间的分布差异,通过少量的并行实例学习来构造稀疏特征变化 (Sparse feature transformation, SFT) 函数。SFT 函数将迁移学习问题转化为稀疏化正则化优化问题。

针对目标域数据稀缺的情况,为了保证有意义的特征变换和最小的跨领域分布发散,源域和目标域的样本数据集被分为非并行数据和并行数据,来构造如下异构特征变化模型,如图 5.5 所示。HTL 方法直接将源域特征转移到目标域特征空间,而不寻求中间的公共特征空间,以此来减少两个任务域之间的转换损失。对于并行样本数据集,通过式 (5.6) 来转化源域数据 X_s^0 ,以最小化平方近似损失来逼近其所对应的目标域数据 X_t^0 ,其中 A 代表并行样本数据集的转化函数。对于非并行样本数据集,二阶矩的匹配策略被用于对齐变换后的特征分布和原目标域的特征分布,通过式 (5.7) 来变换代表源域数据 X_s 分布特性的协方差矩阵 C_s 。其中 B 代表非并行样本数据集的转化函数, C_t 代表目标域数据 X_t 分布特性的协方差矩阵。此稀疏化正则化优化问题的逼近函数 $Loss$ 被表示为式 (5.8),其中 $\|\cdot\|_F$ 代表 Frobenius 矩阵范数, α 是用来平衡两个损失的权衡参数。

$$f(X_s^0) = X_s^0 A \quad (5.6)$$

$$f(C_s) = B^T C_s B \quad (5.7)$$

$$Loss = \min \{ \alpha \|X_s^0 A - X_t^0\|_F^2 + \|B^T C_s B - C_t\|_F^2 \} \quad (5.8)$$

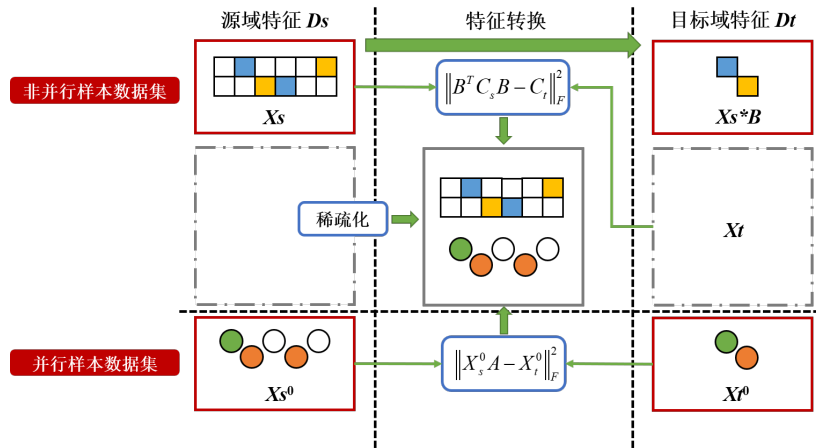


图 5.5 异构特征变化模型

HTL 方法在多个真实的领域自适应样本数据集上进行了实验,结果表明该方法在处理无监督式异构迁移学习问题时具有巨大优势。

(5) 深度迁移学习

深度学习充分利用神经网络的数据拟合能力,由此学习到具有更强泛化能力的特征知识表达。DTL 方法将传统 TL 方法与深度学习相结合,可以将神经网络的强拟合能力迁移应用于不同领域。图 5.6 对比了传统 TL 方法(TCA、GFK 等)与 DTL 方法(BA、DDC、DAN 等)的预报准确性,结果表明 DTL 方法更能显著提高神经网络的数据拟合能力,获得更为精准的预报结果。

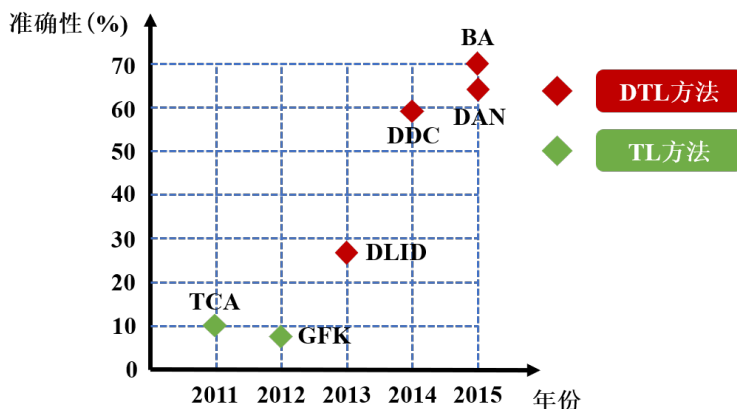


图 5.6 传统迁移学习与深度迁移学习预报准确性对比

DTL 方法的核心手段是 Fine-tuning, 即冻结预训练神经网络模型的部分隐藏层, 利用剩余网络结构对目标域样本数据集进行小范围再训练。Fine-tuning 具有以下优点: 一是不需要针对新任务从头开始训练神经网络, 可以节省大量的训练成本; 二是目标域样本数据集可以看作对神经网络整体训练样本集的扩充, 可以不断增强神经网络模型的鲁棒性和泛化能力。目前 DTL 方法已经被应用于多个领域, 如图像识别、任务分类等。

(6) 强化迁移学习

RTL 方法提出了一种基于自我批评策略梯度方法 (Self-critic policy gradient approach) 的强化学习框架, 以改进传统 TL 方法无法更好地推广到其余不可见样本数据集的问题。RTL 代理学习模型不仅可以记住源域样本数据集的分布, 而且可以学习和适应目标域样本数据集的分布。

RTL 方法的逻辑框架如图 5.7 所示。在每个计算步内, 从源域 D_S 和目标域 D_T 随机抽取小批次的样本数据, 并将其馈送给共享的编码器单元, 解码器会依次解码对应的批次数据。解码完成后, RTL 代理学习模型基于选择程序生成一个语句, 同时通过从输出分布进行取样产生另外一个语句。最后根据式 (5.9) 计算 RTL 方法的损失 $Loss$, 并根据权衡参数 ζ 反向传播误差, 其中 y_t 代表 t 时刻的选择程序, ζ 的取值区间是 $(0, 1)$, 用于权衡从源域和目标域提取样本数据的自我批评损失, θ 代表训练参数, p 代表解码政策。源域源域 D_S 和目标域 D_T 从程序选择和抽样中得到的序列依次定义为 Y_S^1 、 Y_S^2 、 Y_T^1 和 Y_T^2 。 U_S 和 U_T 的表达式依

次为式 (5.10) 和式 (5.11)。图 5.7 中的虚线的粗细显示了 ξ 对代理模型需要依赖 D_S 和 D_T 来反向传播误差的影响程度，其中粗线表示影响更大。RTL 方法很好的解决了目标域中存在不可见样本数据的问题。

$$Loss = (1 - \xi) \log_{p_\theta}^{(y_i^S | U_S)} (r(Y_S^1) - r(Y_S^2)) + \xi \log_{p_\theta}^{(y_i^T | U_T)} (r(Y_T^1) - r(Y_T^2)) \quad (5.9)$$

$$U_S = \{e_S(y_{S,t-1}'), s_t, c_{t-1}, X_S\} \quad (5.10)$$

$$U_T = \{e_S(y_{T,t-1}'), s_t, c_{t-1}, X_T\} \quad (5.11)$$

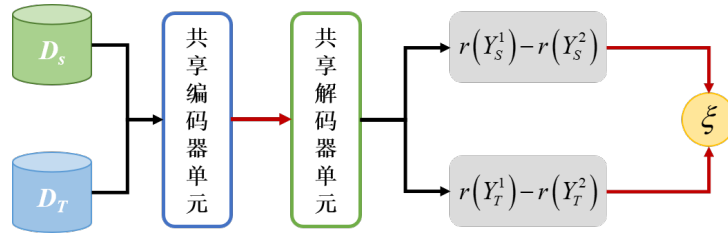


图 5.7 强化迁移学习的逻辑框架

(7) 对抗迁移学习

传统 TL 方法通常是源域向目标域进行的单向迁移，ATL 方法采取任务域双向对抗的迁移，来共享公共特征知识，从而充分利用每个任务域的样本数据集。ATL 方法通过利用特定于任务的决策边界来对齐源域和目标域的分布，因此经过训练的生成器可以在类边界生成模糊特征，这也是双向对抗迁移所具有的特性。类边界是双向对抗的产物，因此通过使用任务特定的分类器作为鉴别器来对齐源域和目标域的特征分布，是 ATL 方法的最终目标，也是为了方便考虑类边界和目标域样本数据集之间的关系。

ATL 方法基于两个分类器的不一致性对目标域样本分布进行预测，两个分类器 F_1 和 F_2 具有不同的特征，如图 5.8 所示。源域之外的目标样本会被两个分类器以不同的方式进行分类，方程 $d(p_1(y|x_i), p_2(y|x_i))$ 被用于度量两个分类器之间的一致，其中 d 是测量两个概率输出之间的散度的函数。

性能相似的两个分类器无法检测目标样本以外的支持来源，因此首先需要训练鉴别器 (F_1 和 F_2) 来最大化目标特征的矛盾。然后训练生成器来欺骗鉴别器，即通过最小化差异来鼓励目标样本在源域数据支持下生成。这种对抗性的学习步骤在 ATL 方法中重复进行，来最大化的获取目标特征。

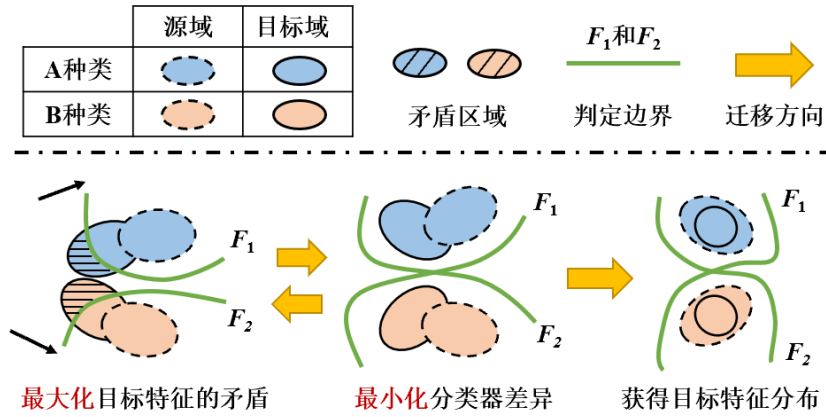


图 5.8 基于两个分类器的对抗学习

5.3 涡激振动深度迁移学习模型

5.3.1 物理驱动深度迁移学习模型

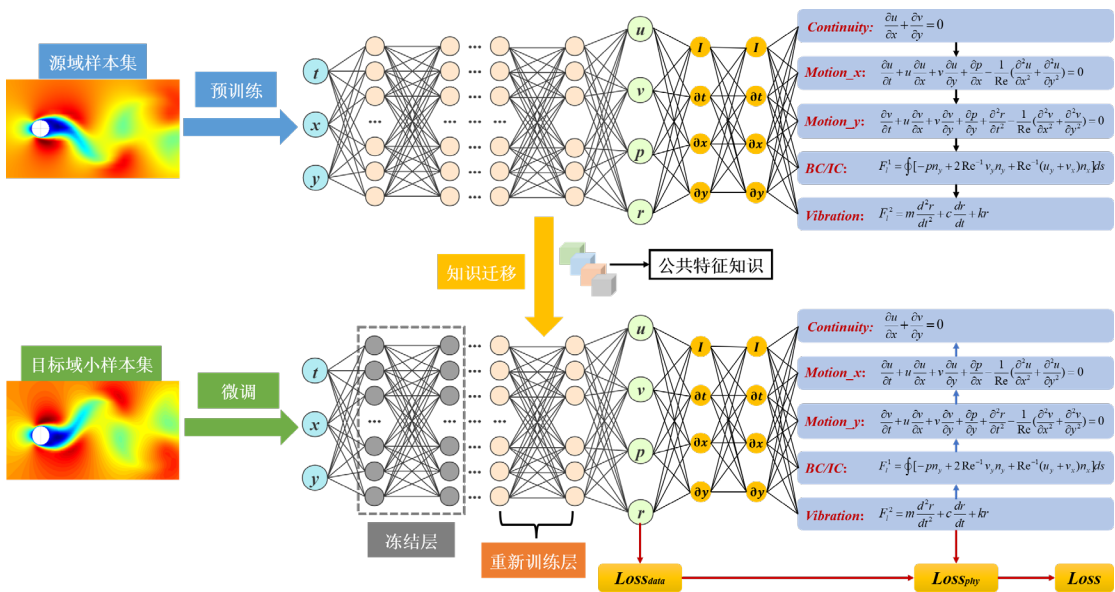


图 5.9 VIV 深度迁移学习模型

图 5.9 是 VIV 深度迁移学习模型，由两部分组成：训练源域样本集的完整 PINN 和训练目标域小样本集的部分冻结 PINN。其中灰色部分表示被冻结的第 1 至 6 层隐藏层。迁移流程如图 5.10 所示，共分为两个阶段：**预训练 (Pre-training)** 以及 **微调 (Fine-tuning)**。

在第一阶段，基于完整的 PINN 模型对源域 D_S 样本集进行预训练，提取代表流场运动信息和结构振动信息的公共特征知识。这些特征知识以不断更新的网

络权重矩阵 $\omega_{i,j}$ 和偏置系数 b_i 的形式保存在预训练模型中。

在第二阶段，冻结 PINN 模型的部分隐藏层，选择较低层次网络的参数进行保留。在神经网络训练过程中，低层次的隐藏层通常提取的是普适特征，可移植性较好；高层次的隐藏层通常提取的是典型特征，呈现很强的独特性。因此，本章的 DTL 模型选择冻结第 1-6 层隐藏层，冻结层的网络权重和偏置不再保持更新，以预训练模型的最终保存结果参与后续训练。在此基础上，利用目标域样本集对预训练模型进行微调更新，实现释放层（第 7-8 层）的重新训练，在目标域中学习新的行为特征，进而在小样本目标任务上具有良好的精度和泛化能力。

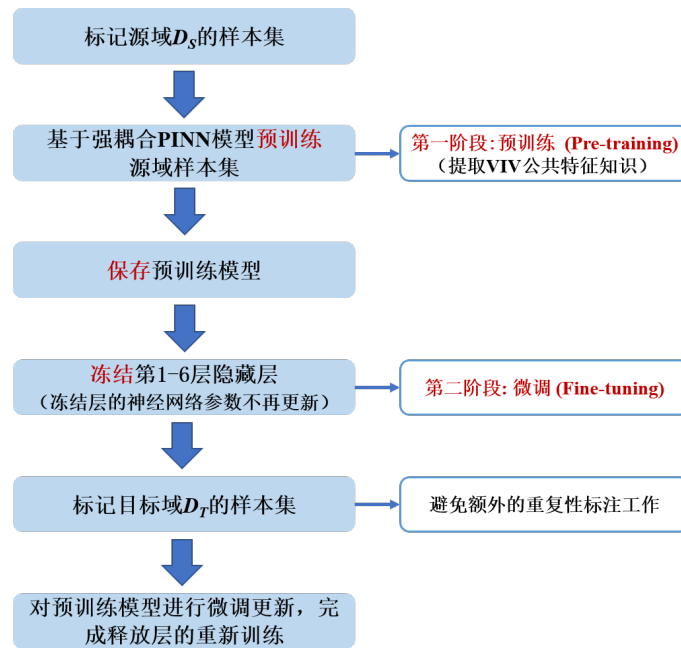


图 5.10 VIV 深度迁移学习模型的迁移流程

5.3.2 算例工况介绍

本章共设置三个数值算例来验证 DTL 代理模型在完成小样本目标任务时的表现性能。每个数值算例的样本集具有不同的数据来源及规模，以此验证该模型的迁移泛化能力。算例 1 和算例 2 的样本集由第三章的 CFD 模型数据构成，算例 3 的样本集由 Raissi 等^[41]人通过粒子图像测速法（Particle image velocimetry, PIV）获得的实验数据构成。

数值算例的样本集按照不同的时间段被划分为源域 D_s 和目标域 D_T 。迁移目标任务是基于小样本集再现流场运动信息和固体振动信息，同时精确度与传统 PINN 算法相当。本章将从训练成本和计算精度两个维度，对比 DTL 模型和传统 PINN 模型的表现性能，来验证 TL 方法对传统 PINN 模型的增强作用。

算例 1 的源域样本集由 0-5 s 的 CFD 模拟数据构成, 间隔时间为 0.05 s, 时间步数为 100; 目标域样本集由 5-10 s 的 CFD 模拟数据构成, 时间步数为 100。

算例 2 的源域样本集由 0-2 s 的 CFD 模拟数据构成, 间隔时间为 0.05 s, 时间步数为 40; 目标域样本集由 2-6 s 的 CFD 模拟数据构成, 时间步数为 80。

算例 3 的源域样本集由 0-7 s 的实验数据构成, 间隔时间为 0.05 s, 时间步数为 140; 目标域样本集由 7-14 s 的实验数据构成, 时间步数为 140。

表 5.2 三个算例的任务域定义

序号	数据来源	源域 D_S	源域 D_S 时间步数	目标域 D_T	目标域 D_T 时间步数
算例 1	CFD 模拟 数据	0-5 s	100	5-10 s	100
算例 2		0-2 s	40	2-6 s	80
算例 3	实验数据	0-7 s	140	7-14 s	140

每个算例中均设置 4 个工况, 包括: 1 个基于传统 PINN 方法训练的对比工况和 3 个基于 DTL 方法训练的验证工况, 具体如表 5.3 所示。其中基于 DTL 方法的 3 个工况将在小样本集上进行训练, 即空间域的配点数量分别下降为对比工况的 1/2、1/4 以及 1/8。

表 5.3 三个算例的工况介绍

算例序号	工况序号	目标域的空间配点数量	训练方法	基础训练模型
算例 1	工况 1.1	12000	PINN 方法	强耦合 PINN 模型
	工况 1.2	6000	DTL 方法	
	工况 1.3	3000		
	工况 1.4	1500		
算例 2	工况 2.1	12000	PINN 方法	
	工况 2.2	6000	DTL 方法	
	工况 2.3	3000		
	工况 2.4	1500		
算例 3	工况 3.1	14400	PINN 方法	
	工况 3.2	7200	DTL 方法	
	工况 3.3	3600		
	工况 3.4	1800		

针对不同数据规模的训练样本集, 神经网络的训练参数也应随之调整, 以确保得到更为准确的训练结果。算例 1 至算例 3 的神经网络训练参数设置分别如表 5.4、表 5.5 和表 5.6 所示。

表 5.4 算例 1 神经网络训练参数设置

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	10000	1.00E-03	$f(x) = \sin x$
第 2 阶段	200	10000	1.00E-04	
第 3 阶段	200	10000	1.00E-05	
第 4 阶段	100	10000	1.00E-06	

表 5.5 算例 2 神经网络训练参数设置

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	300	8000	1.00E-03	$f(x) = \sin x$
第 2 阶段	200	8000	1.00E-04	
第 3 阶段	100	8000	1.00E-05	

表 5.6 算例 3 神经网络训练参数设置

训练阶段	迭代次数 (Epoch)	批次大小 (Batch size)	学习速率 (Learning rate)	激活函数
第 1 阶段	200	10000	1.00E-03	$f(x) = \sin x$
第 2 阶段	300	10000	1.00E-04	
第 3 阶段	200	10000	1.00E-05	

5.4 算例结果分析

(1) 算例 1

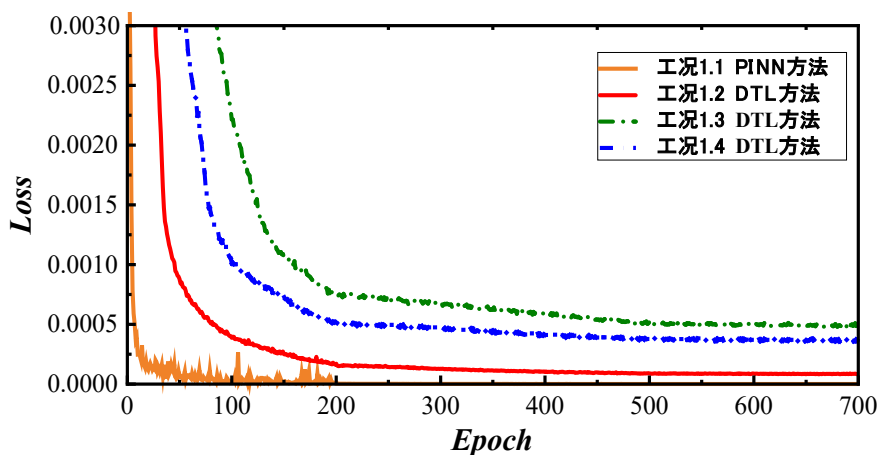
1) 流场运动信息再现情况

计算三个算例的工况时，所使用的硬件设备为 HP Zhan 86 Pro G1 MT，处理器为 Intel(R) Core(TM) i7-7700，内存为 8 GB，操作系统为 64 位。表 5.7 是算例 1 的四个工况训练神经网络模型时所花费的时间成本，工况 1.1 的训练时间为 28.8 h，工况 1.2 的训练时间为 12.6 h，工况 1.3 的训练时间为 5.2 h，工况 1.4 的训练时间为 2.7 h。完成相同的目标域任务时，工况 1.2 至工况 1.4 所花费的训练时间成本约为工况 1.1 的 7/16、1/6 以及 1/11。由此可见，在算例 1 中，通过公共特征知识的转移，DTL 方法有效地降低了完成目标任务所需的时间成本。

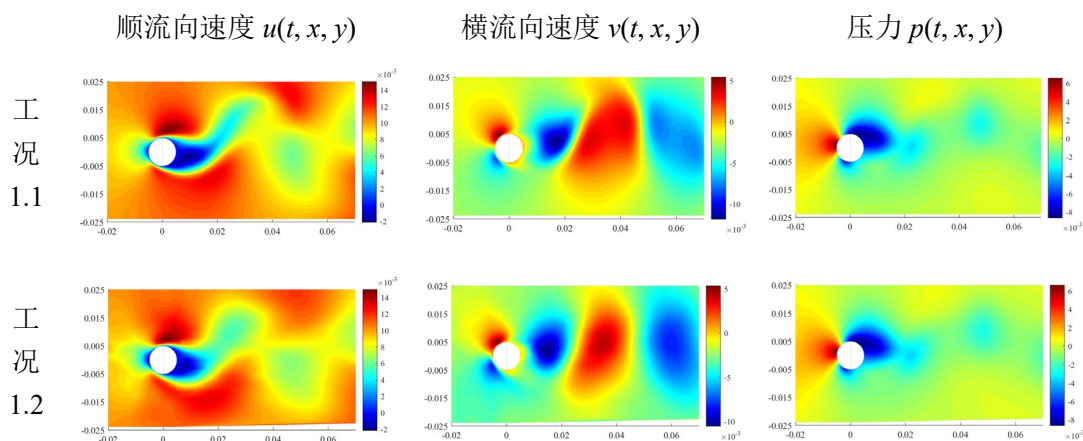
图 5.11 为算例 1 的四个工况在训练迭代过程中损失函数 $Loss$ 的变化曲线，损失函数 $Loss$ 的表达式为式 (3.12)。由图可知，所有工况均在前 200 次 Epoch 内趋于稳定，逐渐收敛至阈值范围内。采用 DTL 方法进行训练的工况 1.2 至工况 1.4 的收敛曲线波动更小，呈现更为稳定的梯度下降趋势。

表 5.7 算例 1 中四个工况的训练时间成本

工况序号	迭代次数 (Epoch)	时间成本 (h)
工况 1.1	700	28.8
工况 1.2	700	12.6
工况 1.3	700	5.2
工况 1.4	700	2.7

图 5.11 算例 1 中四个工况的损失函数 $Loss$ 的变化曲线

算例 1 中四个工况的目标域样本集为第 5-10 s 的 VIV 模拟数据, 共包含 100 个时间步。计算结果表明这四个工况均可以准确再现目标域的流场运动信息, 包括各流场质点的顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 以及压力 $p(t, x, y)$ 。图 5.12 至图 5.14 分别为四个工况第 5 s、第 6 s 以及第 8 s 的流场运动信息再现结果。图 5.15 是流场运动信息再现结果的 RMSE 对比。下文图表里的所有变量均采用国际标准单位。



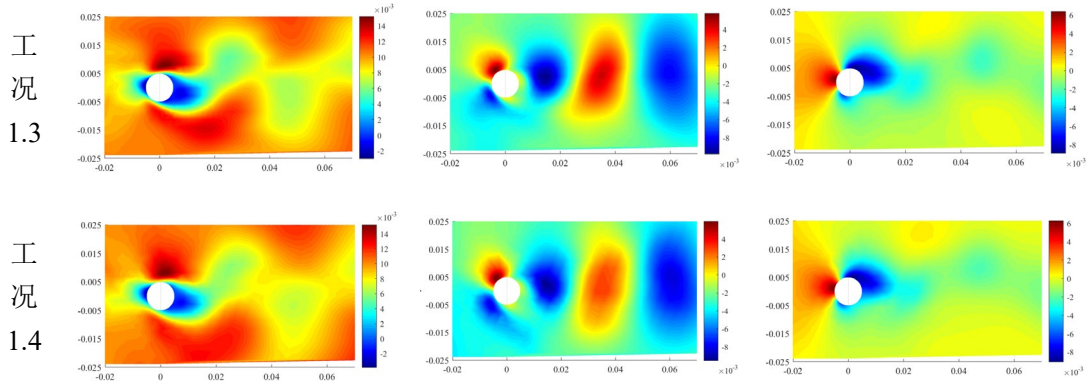


图 5.12 算例 1 中四个工况第 5 s 流场运动信息再现结果

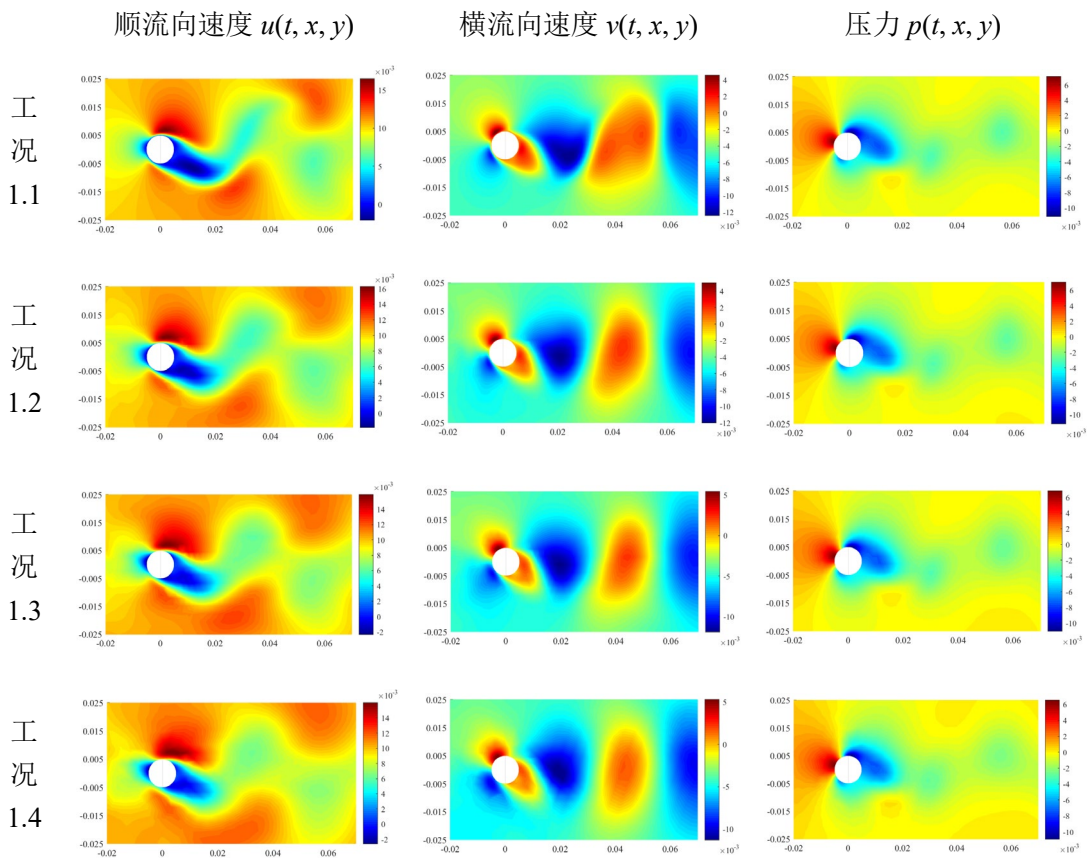
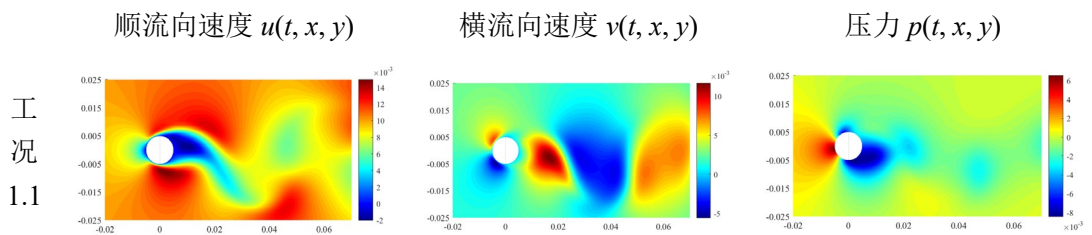


图 5.13 算例 1 中四个工况第 6 s 流场运动信息再现结果



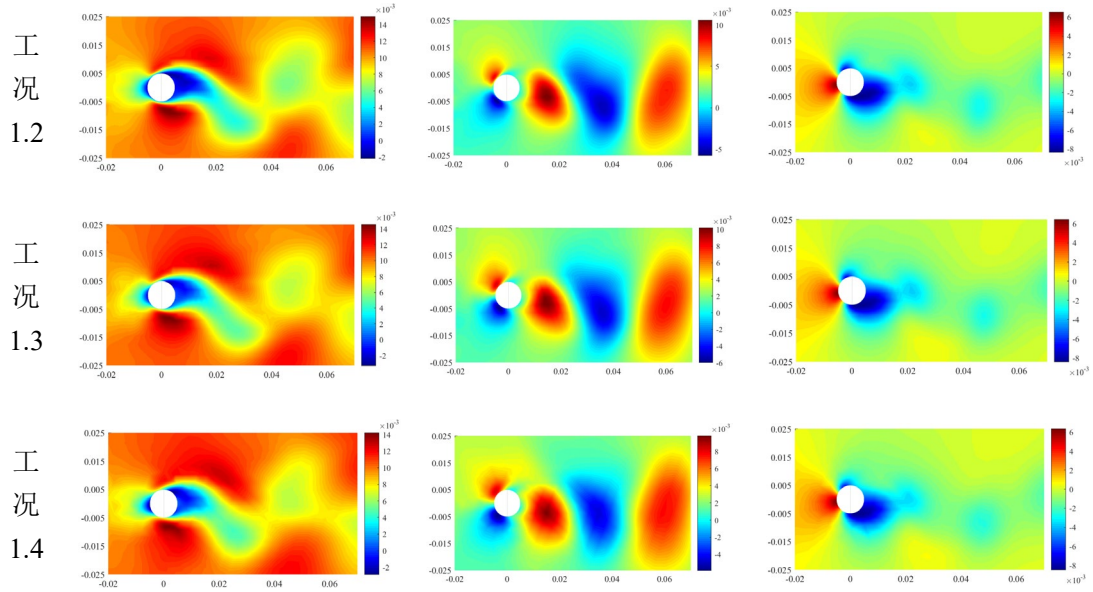
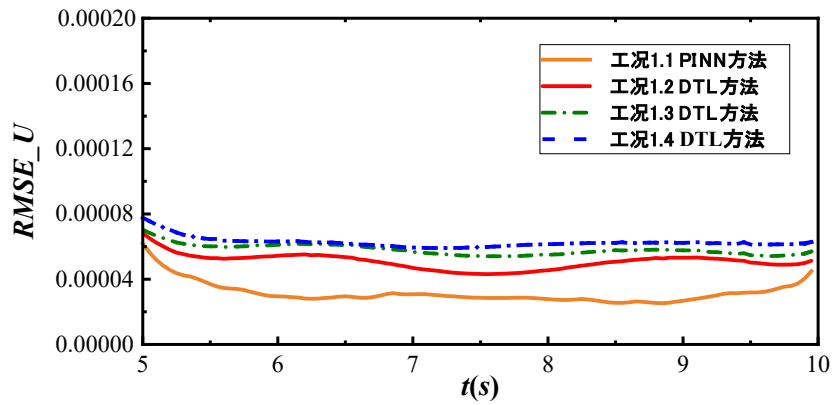
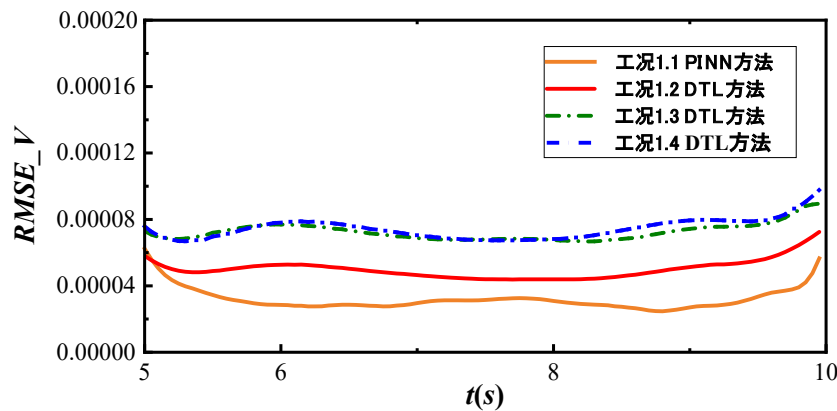


图 5.14 算例 1 中四个工况第 8 s 流场运动信息再现结果



(a) 顺流向速度 u



(b) 横流向速度 v

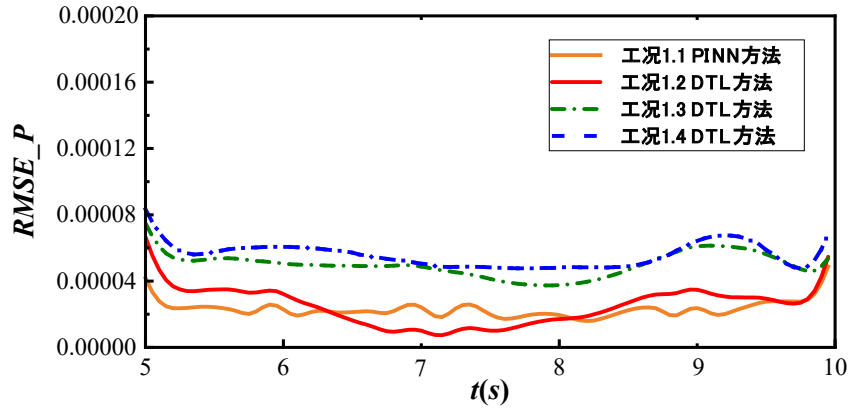

 (c) 压力 p

图 5.15 流场运动信息再现结果的 RMSE 对比

对于顺流向速度 u ，工况 1.1 至工况 1.4 的 RMSE 浮动区间分别是 $(2.5E-05, 6.2E-05)$ 、 $(4.3E-05, 6.8E-05)$ 、 $(5.4E-05, 7.0E-05)$ 以及 $(5.9E-05, 7.8E-05)$ ，幅值分别是 $6.2E-05$ 、 $6.8E-05$ 、 $7.0E-05$ 和 $7.8E-05$ 。

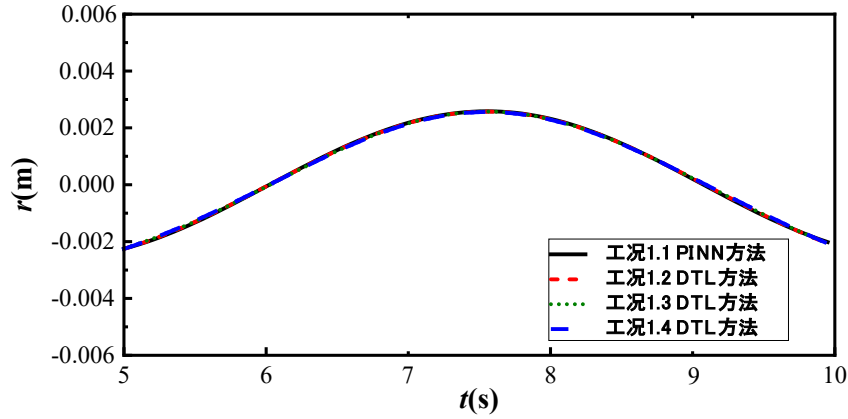
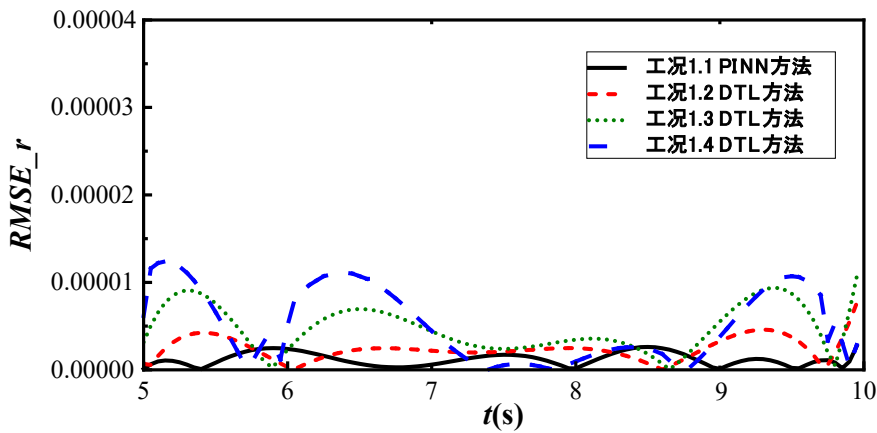
对于横流向速度 v ，工况 1.1 至工况 1.4 的 RMSE 浮动区间分别是 $(2.4E-05, 6.3E-05)$ 、 $(4.4E-05, 7.3E-05)$ 、 $(6.7E-05, 9.0E-05)$ 以及 $(6.7E-05, 9.8E-05)$ ，幅值分别是 $6.3E-05$ 、 $7.3E-05$ 、 $9.0E-05$ 和 $9.8E-05$ 。

对于压力 p ，工况 1.1 至工况 1.4 的浮动区间分别是 $(1.6E-05, 4.9E-05)$ 、 $(8.0E-06, 6.7E-05)$ 、 $(3.7E-05, 7.5E-05)$ 以及 $(4.8E-05, 8.3E-05)$ ，幅值分别是 $4.9E-05$ 、 $6.7E-05$ 、 $7.5E-05$ 和 $8.3E-05$ 。

图 5.15 量化地比较了算例 1 的四个工况对流场运动信息再现结果的 RMSE。综合来看，采用 DTL 方法的三个工况将顺流向速度 u 、横流向速度 v 以及压力 p 再现结果的 RMSE 分别控制在 $8.0E-05$ 、 $1.0E-04$ 以及 $9.0E-05$ 以下。参考实际的流场数据以及工况 1.1 的再现结果，基于 DTL 方法的三个工况对流场运动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

2) 结构振动信息再现情况

结构振动信息主要包括固体结构振动位移 r 、绕流升力 F_l 以及绕流阻力 F_d 。算例 1 目标域样本集的时域范围是 5-10 s，即四个工况需准确再现 100 个时间步的固体结构振动位移 r 。图 5.16 是算例 1 中四个工况对固体结构振动位移 r 的再现结果，图 5.17 是再现结果的 RMSE 对比。

图 5.16 算例 1 中四个工况对固体结构振动位移 r 的再现结果图 5.17 固体结构振动位移 r 再现结果的 RMSE 对比

对于固体结构振动位移 r , 工况 1 至工况 4 的 RMSE 浮动区间分别为 $(8.31\text{E-}08, 2.71\text{E-}06)$ 、 $(2.46\text{E-}08, 7.63\text{E-}06)$ 、 $(5.82\text{E-}08, 1.06\text{E-}05)$ 以及 $(5.88\text{E-}08, 1.24\text{E-}05)$, 幅值分别为 $2.71\text{E-}06$ 、 $7.63\text{E-}06$ 、 $1.06\text{E-}05$ 以及 $1.24\text{E-}05$ 。在单个时间步内, 再现结果的最大相对误差分别为 2.31%、2.87%、3.33% 以及 5.24%。

基于流场运动信息的再现结果和流固耦合面的边界条件, 可以积分计算得到绕流升力 F_l 和绕流阻力 F_d , 积分表达式为式 (3.5) 和式 (3.6)。为了更直观地验证算例 1 中采用 DTL 方法的 3 个工况再现绕流力的准确性, 通过式 (3.22) 和式 (3.23) 将绕流升力 F_l 和绕流阻力 F_d 转化为绕流升力系数 C_l 和绕流阻力系数 C_d , 如图 5.18 所示。最后以绕流升力系数幅值 $|C_{l_{max}}|$ 和绕流阻力系数均值 $\overline{C_d}$ 为指标, 将再现结果与第三章 CFD 模拟结果进行对比, 如表 5.8 所示。

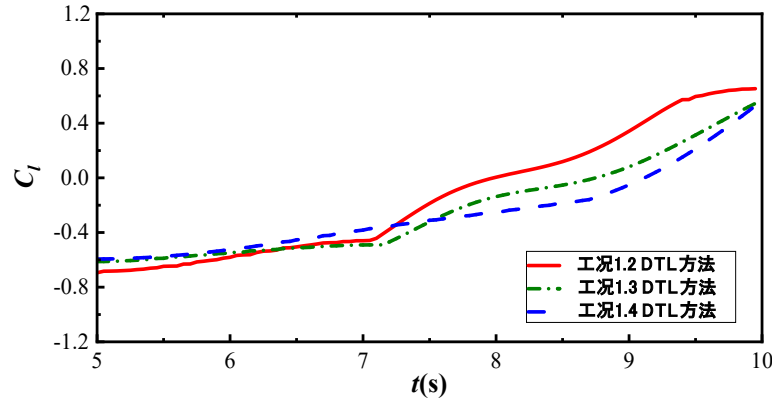
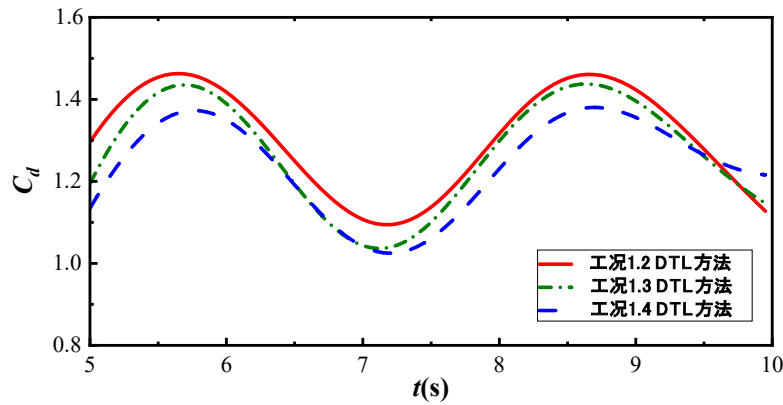

 (a) 流体升力系数 C_l

 (b) 流体阻力系数 C_d

图 5.18 算例 1 中采用 DTL 方法的三个工况对流体力系数的再现结果

表 5.8 流体力系数再现结果正确性验证

序号	再现结果 $ C_l _{max}$	数值模拟 $ C_l _{max}$	误差	再现结果 $\overline{C_d}$	数值模拟 $\overline{C_d}$	误差
工况 1.2	0.69	0.66	0.03	1.31	1.34	0.03
工况 1.3	0.61	0.66	0.05	1.27	1.34	0.07
工况 1.4	0.59	0.66	0.07	1.24	1.34	0.10

对于流体升力系数幅值 $|C_l|_{max}$ ，工况 1.2 至工况 1.4 的再现结果分别为 0.69、0.61 和 0.59，CFD 数值模拟结果为 0.66，再现结果的相对误差分别为 4.5%、7.5% 以及 10.4%。

对于流体阻力系数均值 $\overline{C_d}$ ，工况 1.2 至工况 1.4 的再现结果分别为 1.31、1.27 和 1.24，CFD 数值模拟结果为 1.34，再现结果的相对误差分别为 2.2%、5.2% 以及 7.6%。

综合来看，参考实际的结构振动信息以及 CFD 数值模拟结果，基于 DTL 方法的三个工况对结构振动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

(2) 算例 2

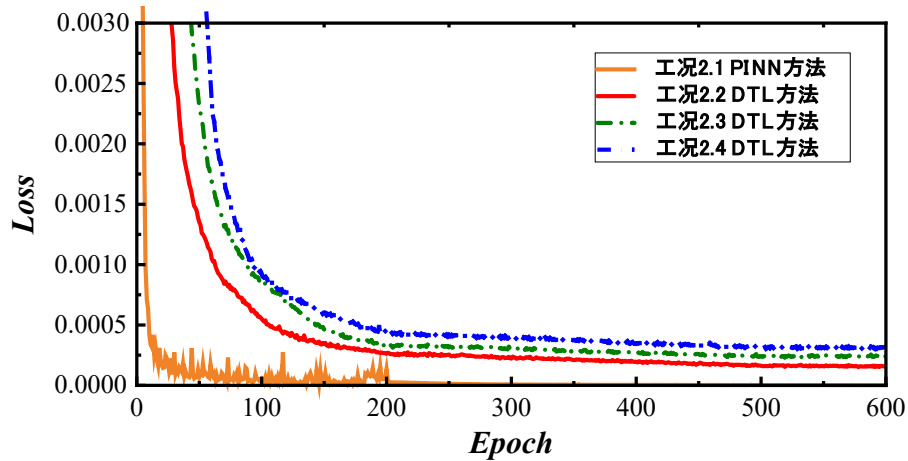
1) 流场运动信息再现情况

表 5.9 是算例 2 的四个工况训练神经网络模型时所花费的时间成本, 工况 2.1 的训练时间为 24.7 h, 工况 2.2 的训练时间为 9.9 h, 工况 2.3 的训练时间为 6.1 h, 工况 2.4 的训练时间为 2.1 h。完成相同的目标域任务时, 工况 2.2 至工况 2.4 所花费的训练时间成本约为工况 2.1 的 2/5、1/4 以及 1/12。由此可见, 在算例 2 中, 通过公共特征知识的转移, DTL 方法有效地降低了完成目标任务所需的时间成本。

表 5.9 算例 2 中四个工况的训练时间成本

工况序号	迭代次数 (Epoch)	时间成本 (h)
工况 2.1	600	24.7
工况 2.2	600	9.9
工况 2.3	600	6.1
工况 2.4	600	2.1

图 5.19 为算例 2 的四个工况在训练迭代过程中损失函数 $Loss$ 的变化曲线, 损失函数 $Loss$ 的表达式为式 (3.12)。由图可知, 所有工况均在前 150 次 Epoch 内趋于稳定, 逐渐收敛至阈值范围内。采用 DTL 方法进行训练的工况 2.2 至工况 2.4 的收敛曲线波动更小, 呈现更为稳定的梯度下降趋势。

图 5.19 算例 2 中四个工况的损失函数 $Loss$ 的变化曲线

算例 2 中四个工况的目标域样本集为第 2-6 s 的 VIV 模拟数据, 共包含 80 个时间步。计算结果表明这四个工况均可以准确再现目标域的流场运动信息, 包括各流场质点的顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 以及压力 $p(t, x, y)$ 。图 5.20 至图 5.22 分别为四个工况第 3 s、第 4 s 以及第 5 s 的流场运动信息再现结

果。图 5.23 是流场运动信息再现结果的 RMSE 对比。下文图表里的所有变量均采用国际标准单位。

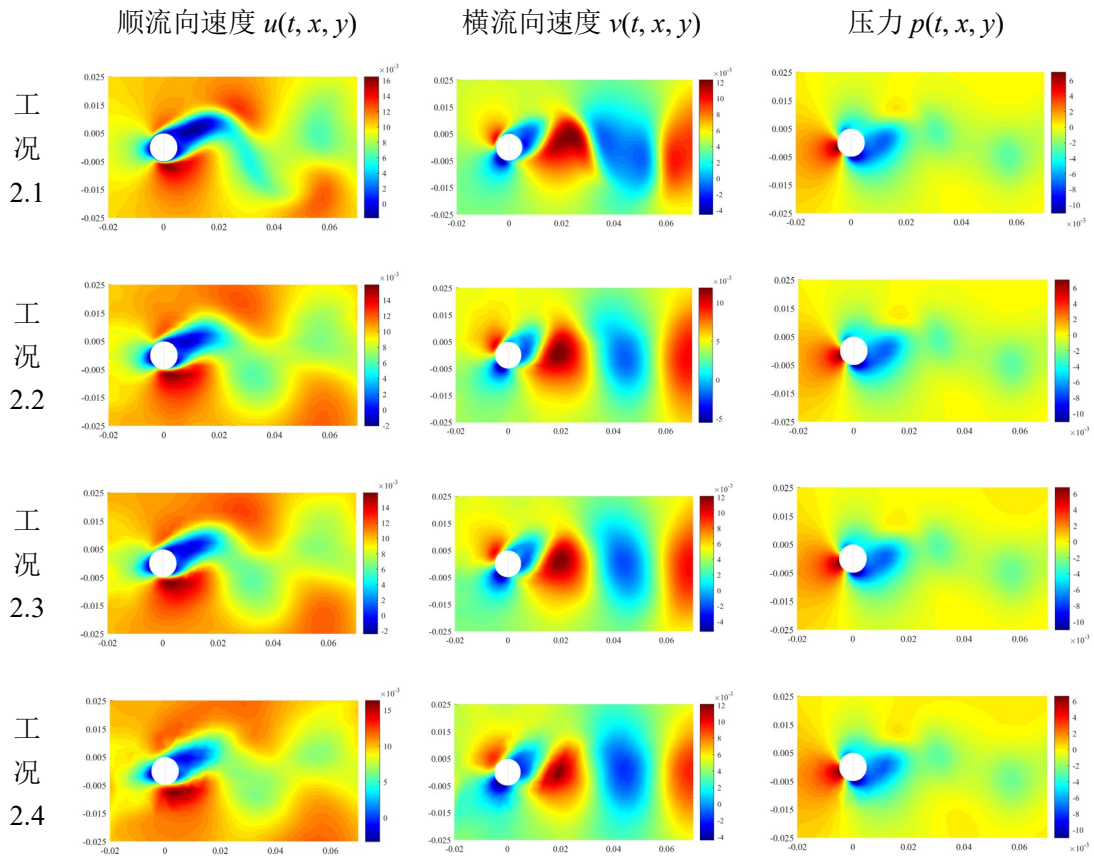
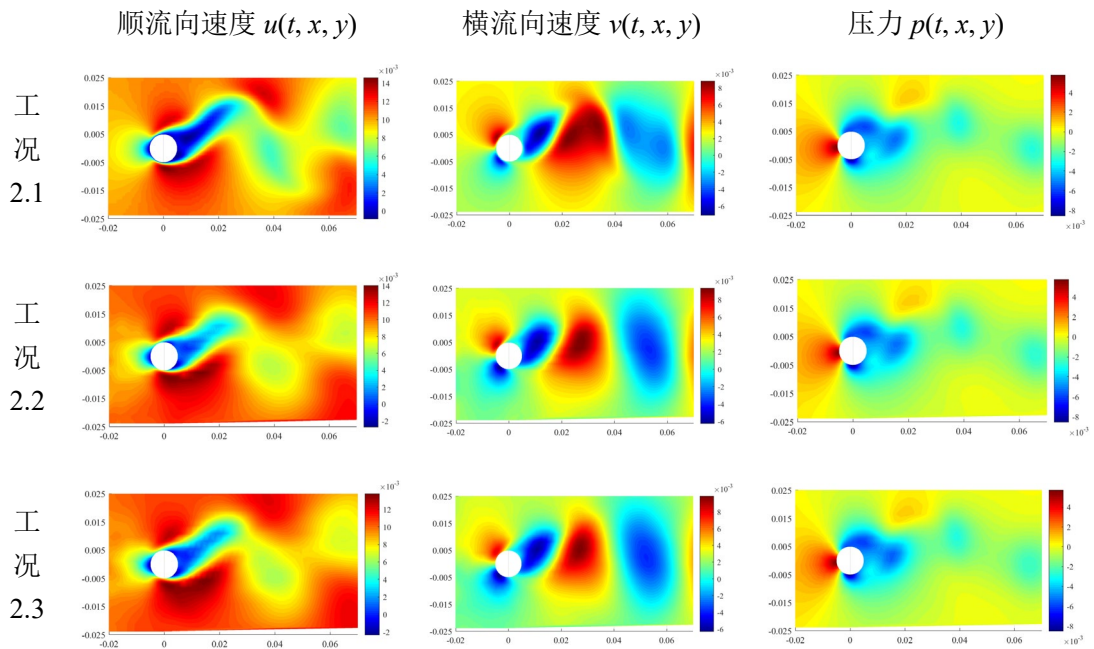


图 5.20 算例 2 中四个工况第 3 s 流场运动信息再现结果



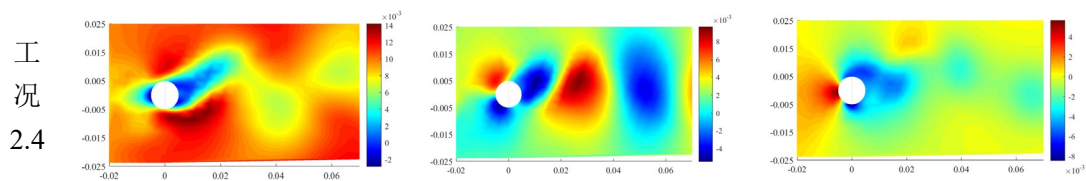


图 5.21 算例 2 中四个工况第 4 s 流场运动信息再现结果

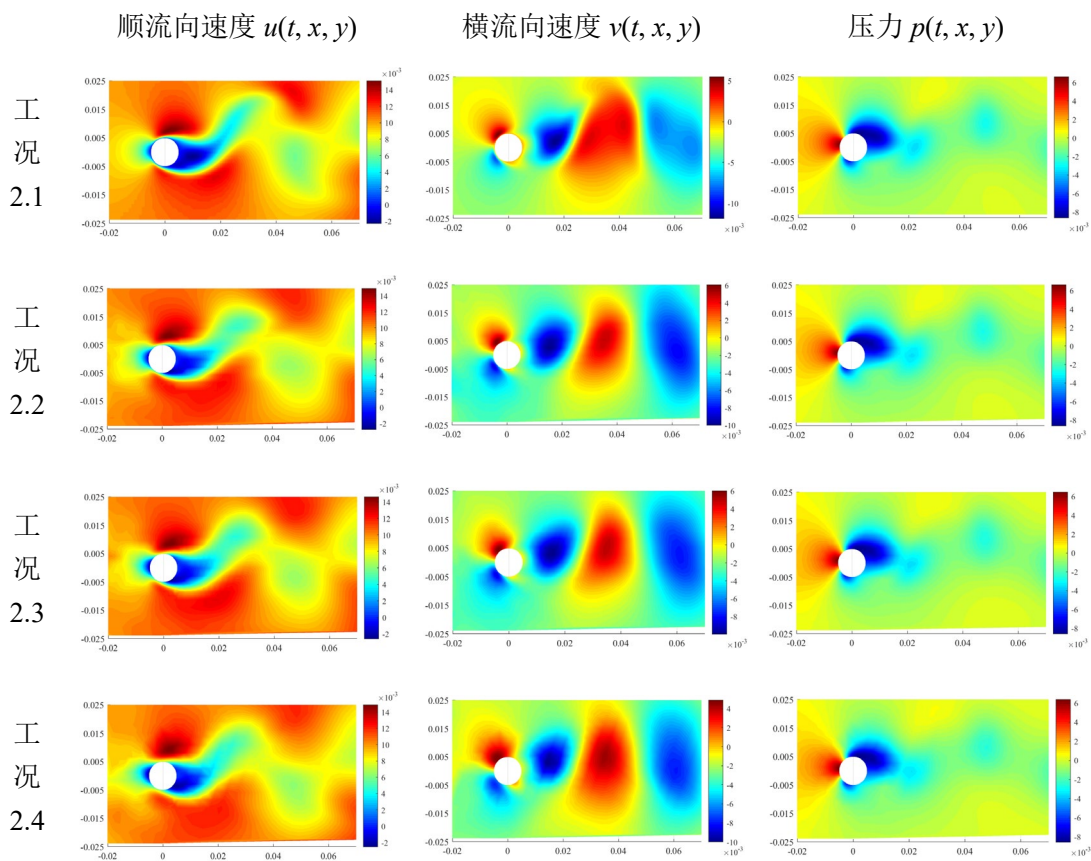
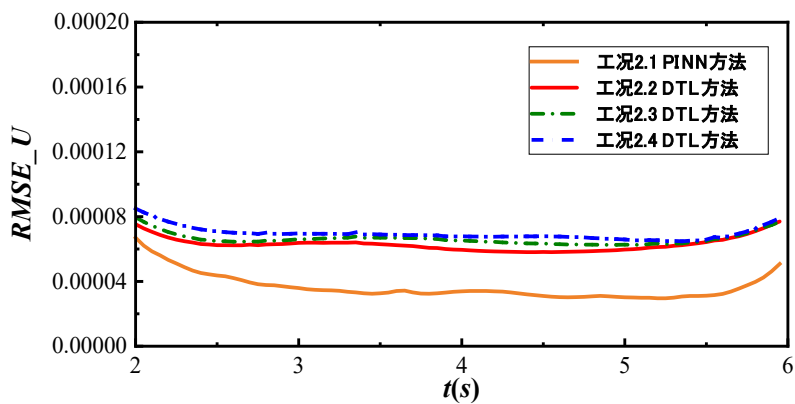


图 5.22 算例 2 中四个工况第 5 s 流场运动信息再现结果



(a) 顺流向速度 u

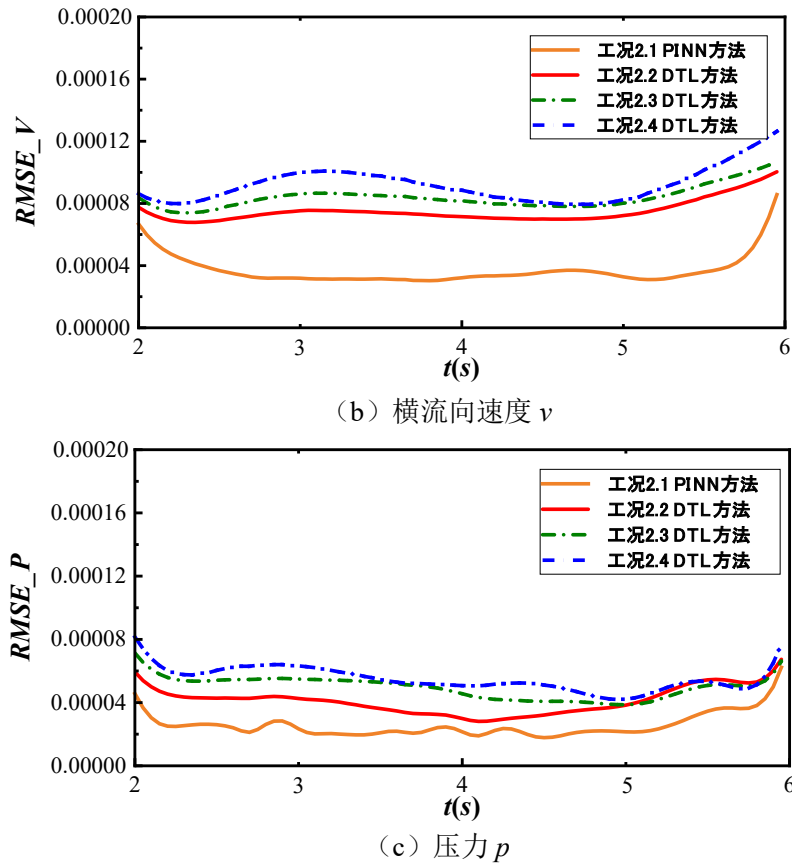


图 5.23 流场运动信息再现结果的 RMSE 对比

对于顺流向速度 u ，工况 2.1 至工况 2.4 的 RMSE 浮动区间分别是 $(3.0\text{E-}05, 6.7\text{E-}05)$ 、 $(5.8\text{E-}05, 7.7\text{E-}05)$ 、 $(6.3\text{E-}05, 8.0\text{E-}05)$ 以及 $(6.5\text{E-}05, 8.5\text{E-}05)$ ，幅值分别是 $6.7\text{E-}05$ 、 $7.7\text{E-}05$ 、 $8.0\text{E-}05$ 和 $8.5\text{E-}05$ 。

对于横流向速度 v ，工况 2.1 至工况 2.4 的 RMSE 浮动区间分别是 $(3.0\text{E-}05, 8.6\text{E-}05)$ 、 $(6.8\text{E-}05, 1.0\text{E-}04)$ 、 $(7.4\text{E-}05, 1.1\text{E-}04)$ 以及 $(7.9\text{E-}05, 1.3\text{E-}04)$ ，幅值分别是 $8.6\text{E-}05$ 、 $1.0\text{E-}04$ 、 $1.1\text{E-}04$ 和 $1.3\text{E-}04$ 。

对于压力 p ，工况 2.1 至工况 2.4 的浮动区间分别是 $(1.8\text{E-}05, 6.2\text{E-}05)$ 、 $(2.8\text{E-}05, 6.7\text{E-}05)$ 、 $(3.9\text{E-}05, 7.1\text{E-}05)$ 以及 $(4.2\text{E-}05, 8.2\text{E-}05)$ ，幅值分别是 $6.2\text{E-}05$ 、 $6.7\text{E-}05$ 、 $7.1\text{E-}05$ 和 $8.2\text{E-}05$ 。

图 5.23 量化地比较了算例 2 的四个工况对流场运动信息再现结果的 RMSE。综合来看，采用 DTL 方法的三个工况将顺流向速度 u 、横流向速度 v 以及压力 p 再现结果的 RMSE 分别控制在 $9.0\text{E-}05$ 、 $1.3\text{E-}04$ 以及 $9.0\text{E-}05$ 以下。参考实际的流场数据以及工况 2.1 的再现结果，基于 DTL 方法的三个工况对流场运动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

2) 结构振动信息再现情况

结构振动信息主要包括固体结构振动位移 r 、绕流升力 F_l 以及绕流阻力 F_d 。

算例 2 目标域样本集的时域范围是 2-6 s，即四个工况需准确再现 80 个时间步的固体结构振动位移 r 。图 5.24 是算例 1 中四个工况对固体结构振动位移 r 的再现结果，图 5.25 是再现结果的 RMSE 对比。

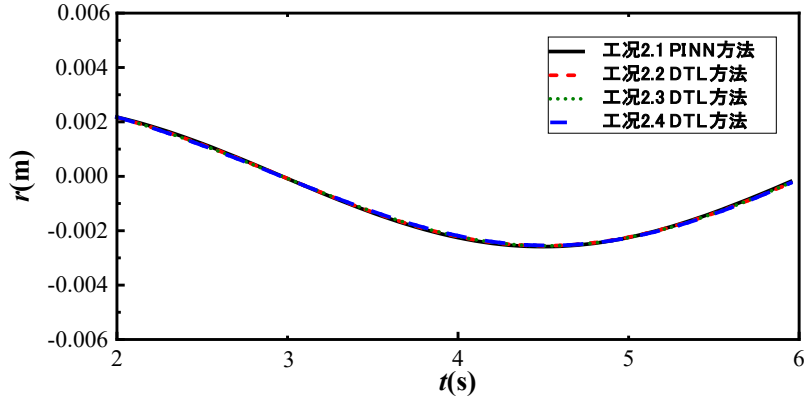


图 5.24 算例 2 中四个工况对固体结构振动位移 r 的再现结果

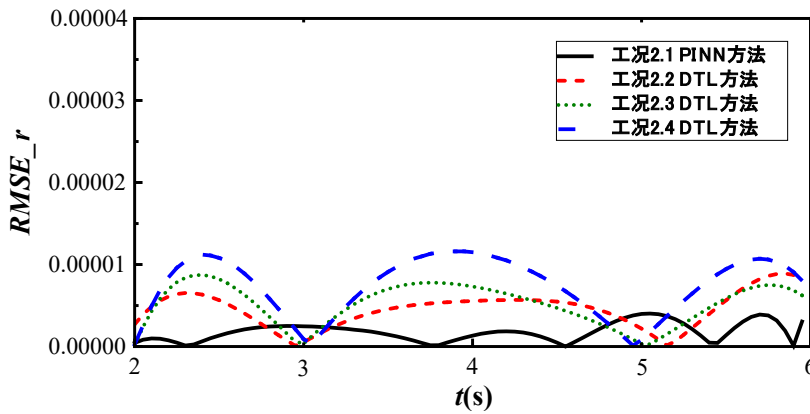


图 5.25 固体结构振动位移 r 再现结果的 RMSE 对比

对于固体结构振动位移 r ，工况 2.1 至工况 2.4 的 RMSE 浮动区间分别为 $(4.38\text{E-}07, 3.04\text{E-}06)$ 、 $(9.69\text{E-}08, 8.89\text{E-}06)$ 、 $(1.09\text{E-}07, 8.73\text{E-}06)$ 以及 $(1.23\text{E-}07, 1.16\text{E-}05)$ ，幅值分别为 $3.04\text{E-}06$ 、 $8.89\text{E-}06$ 、 $8.73\text{E-}06$ 以及 $1.16\text{E-}05$ 。在单个时间步内，再现结果的最大相对误差分别为 8.32%、8.73%、10.23% 以及 11.71%。

基于流场运动信息的再现结果和流固耦合面的边界条件，可以积分计算得到绕流升力 F_l 和绕流阻力 F_d ，积分表达式为式 (3.5) 和式 (3.6)。为了更直观地验证算例 2 中采用 DTL 方法的 3 个工况再现绕流力的准确性，通过式 (3.22) 和式 (3.23) 将绕流升力 F_l 和绕流阻力 F_d 转化为绕流升力系数 C_l 和绕流阻力系数 C_d ，如图 5.26 所示。最后以绕流升力系数幅值 $|C_{l_{max}}|$ 和绕流阻力系数均值 $\overline{C_d}$ 为指标，将再现结果与第三章 CFD 模拟结果进行对比，如表 5.10 所示。

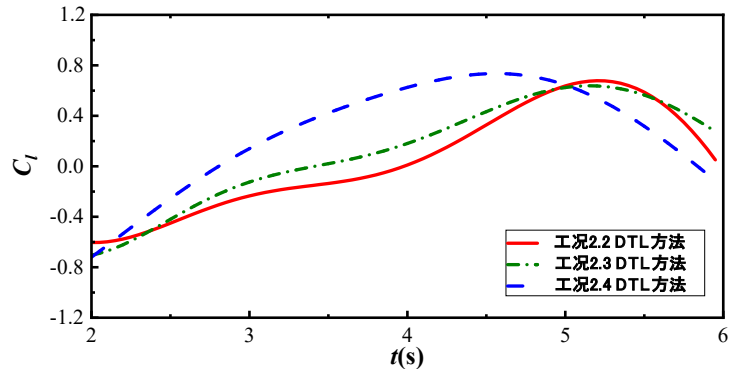
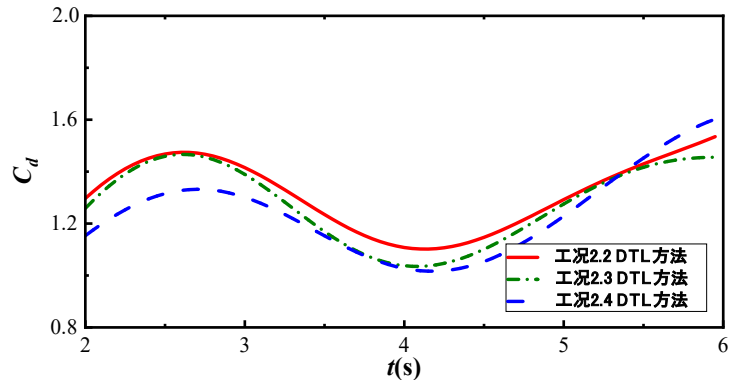

 (a) 流体升力系数 C_l

 (b) 流体阻力系数 C_d

图 5.26 算例 2 采用 DTL 方法的三个工况对流体力系数的再现结果

表 5.10 流体力系数再现结果正确性验证

序号	再现结果 $ C_l _{max}$	数值模拟 $ C_l _{max}$	误差	再现结果 $\overline{C_d}$	数值模拟 $\overline{C_d}$	误差
工况 2.2	0.68	0.66	0.02	1.31	1.34	0.03
工况 2.3	0.70	0.66	0.04	1.28	1.34	0.06
工况 2.4	0.73	0.66	0.07	1.25	1.34	0.09

对于流体升力系数幅值 $|C_l|_{max}$ ，工况 2.2 至工况 2.4 的再现结果分别为 0.68、0.70 和 0.73，CFD 数值模拟结果为 0.66，再现结果的相对误差分别为 3.0%、6.1% 以及 10.6%。

对于流体阻力系数均值 $\overline{C_d}$ ，工况 2.2 至工况 2.4 的再现结果分别为 1.31、1.28 和 1.25，CFD 数值模拟结果为 1.34，再现结果的相对误差分别为 2.2%、4.4% 以及 6.7%。

综合来看，参考实际的结构振动信息以及 CFD 数值模拟结果，基于 DTL 方法的三个工况对结构振动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

(3) 算例 3

1) 流体运动信息再现情况

在 Raissi^[41]实验中, 流场的顺流向 (x 方向) 坐标范围为 $-2D$ 至 $8D$, 横流向 (y 方向) 坐标范围为 $-3D$ 至 $3D$, 如图 5.27 所示, 其中 D 是位于坐标中心的圆柱结构的特征直径。流体是雷诺数 $Re=100$ 的层流, 来流速度 $U=1$ m/s, 实验每隔 0.05 s 测量一次 VIV 数据信息, 总时长为 14 s。

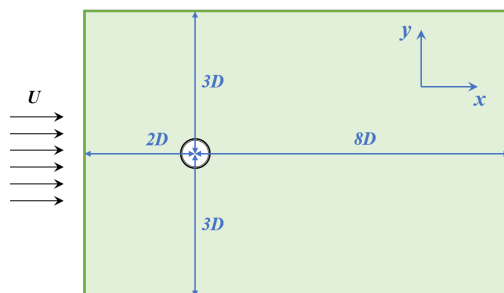


图 5.27 Raissi 实验所设置的流场尺寸

表 5.11 是算例 3 的四个工况训练神经网络模型时所花费的时间成本, 工况 3.1 的训练时间为 48.7 h, 工况 3.2 的训练时间为 22.9 h, 工况 3.3 的训练时间为 10.3 h, 工况 3.4 的训练时间为 5.4 h。完成相同的目标域任务时, 工况 3.2 至工况 3.4 所花费的训练时间成本约为工况 3.1 的 $1/2$ 、 $1/5$ 以及 $1/9$ 。由此可见, 在算例 3 中, 通过公共特征知识的转移, DTL 方法有效地降低了完成目标任务所需的时间成本。

表 5.11 算例 3 中四个工况的训练时间成本

工况序号	迭代次数 (Epoch)	时间成本 (h)
工况 3.1	700	48.7
工况 3.2	700	22.9
工况 3.3	700	10.3
工况 3.4	700	5.4

图 5.28 为算例 3 的四个工况在训练迭代过程中损失函数 $Loss$ 的变化曲线, 损失函数 $Loss$ 的表达式为式 (3.12)。工况 3.1 约在第 250 次 Epoch 处开始趋于稳定, 工况 3.2 至工况 3.4 分别在第 30 次、第 50 次以及第 180 次 Epoch 处提前开始趋于稳定, 逐渐收敛至阈值范围内。采用 DTL 方法进行训练的工况 3.2 至工况 3.4 的收敛曲线波动更小, 呈现更为稳定的梯度下降趋势。

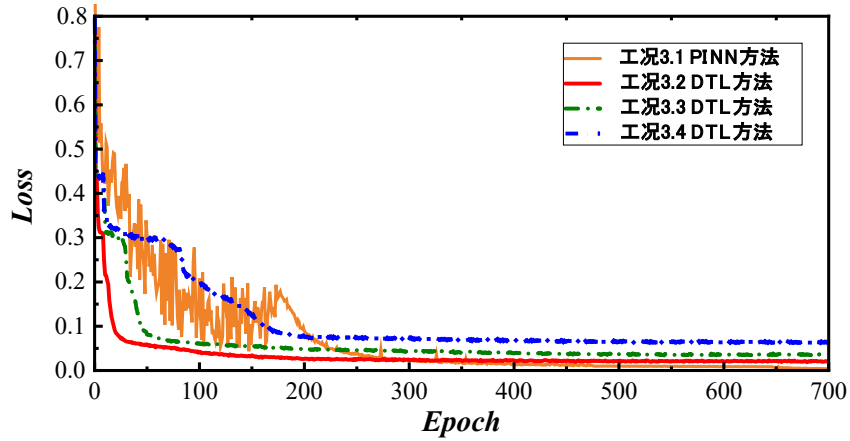


图 5.28 算例 3 中四个工况的损失函数 $Loss$ 的变化曲线

算例 3 中四个工况的目标域样本集为第 7-14 s 的 VIV 模拟数据，共包含 140 个时间步。计算结果表明这四个工况均可以准确再现目标域的流场运动信息，包括各流场质点的顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 以及压力 $p(t, x, y)$ 。图 5.29 至图 5.32 分别为四个工况第 9 s、第 11 s 以及第 13 s 的流场运动信息再现结果。图 5.33 是流场运动信息再现结果的 RMSE 对比。下文图表里的所有变量均采用国际标准单位。

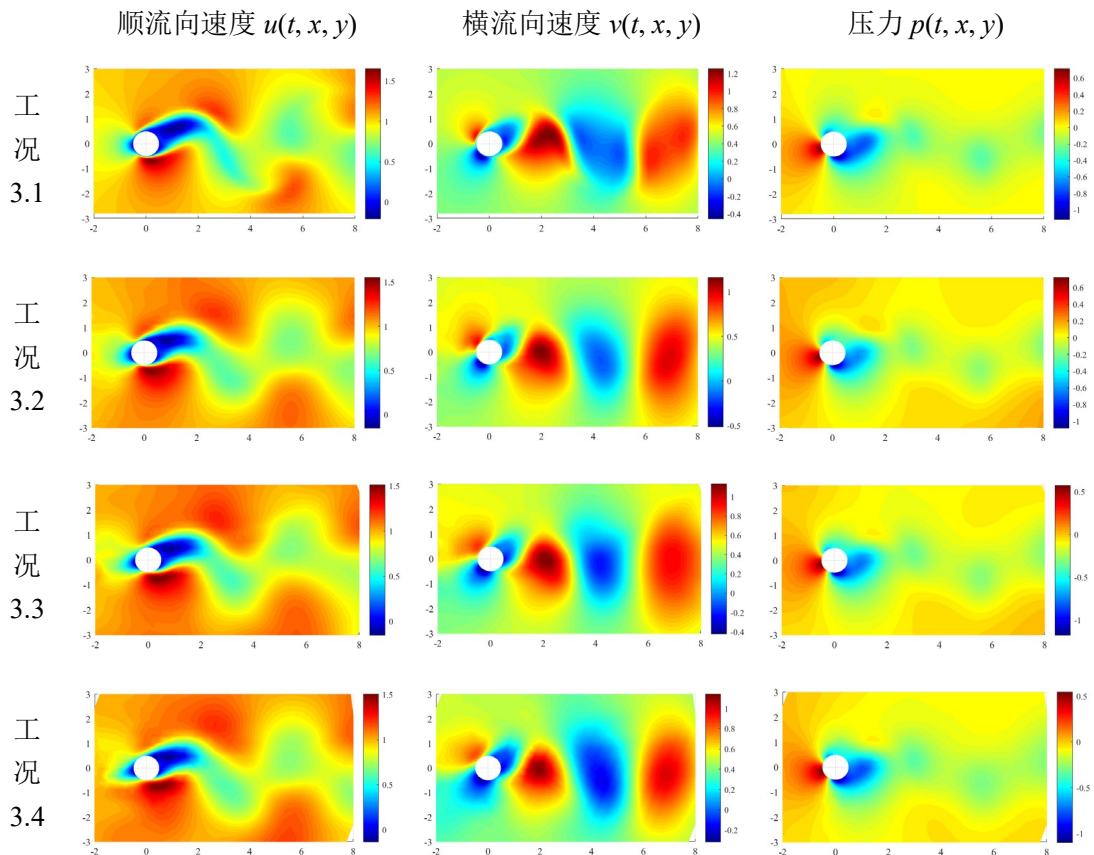


图 5.29 算例 3 中四个工况第 9 s 流场运动信息再现结果

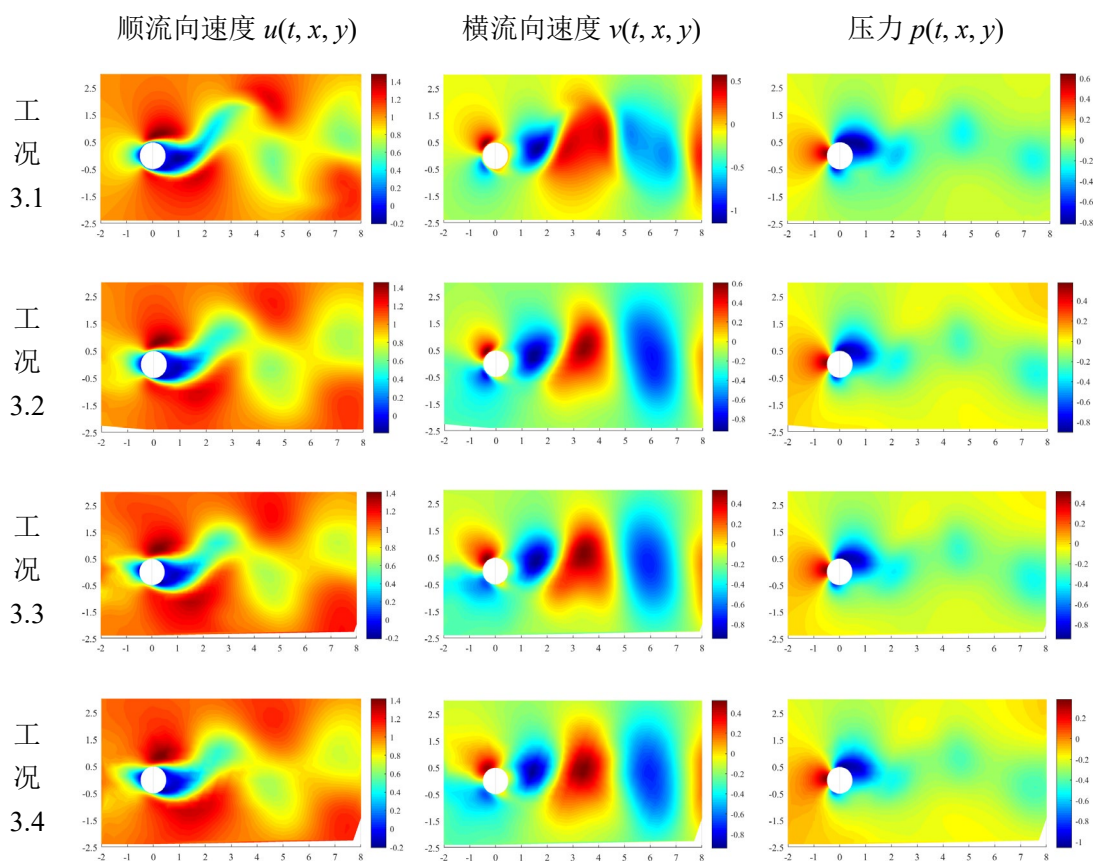
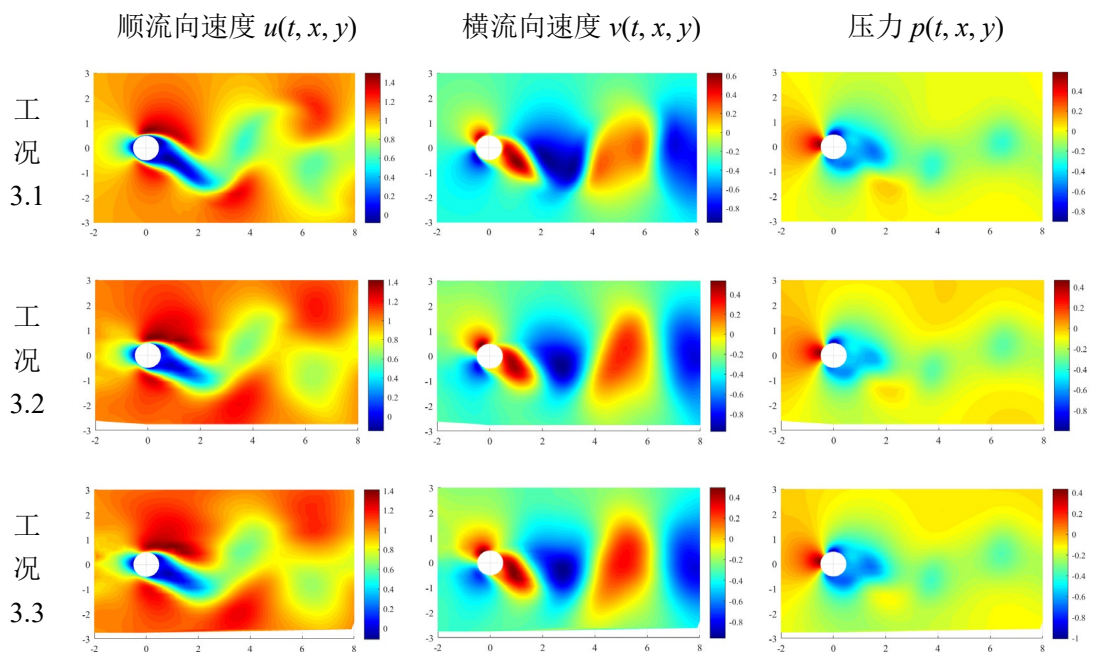


图 5.30 算例 3 中四个工况第 11 s 流场运动信息再现结果



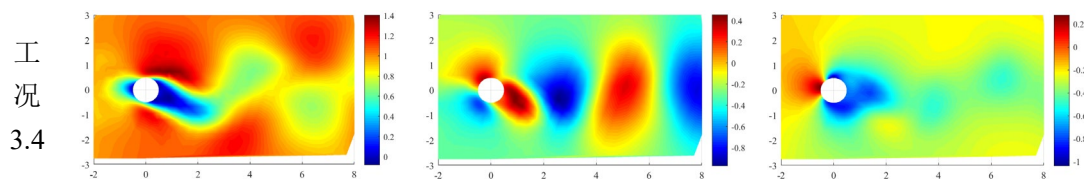
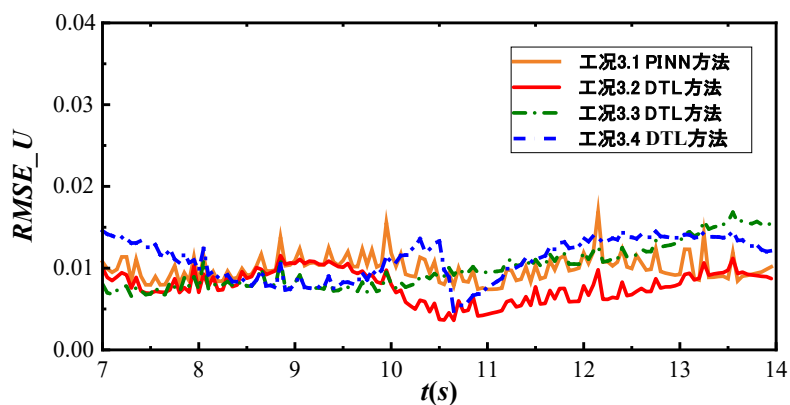
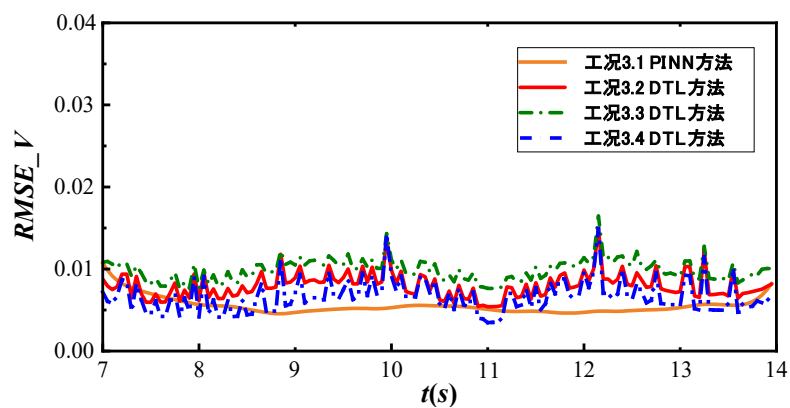


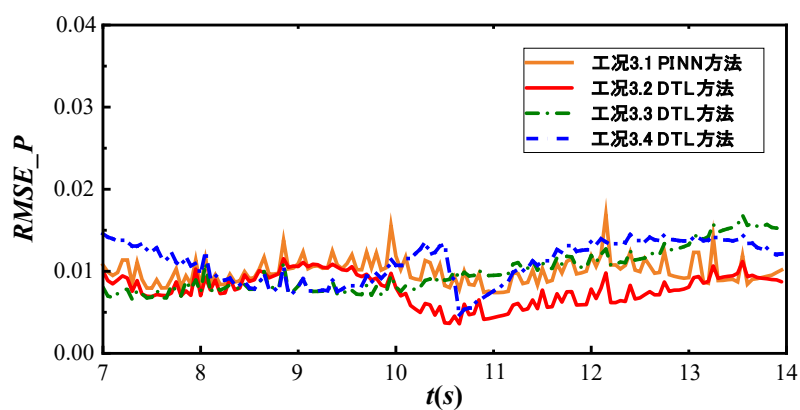
图 5.31 算例 3 中四个工况第 13 s 流场运动信息再现结果



(a) 顺流向速度 u



(b) 顺流向速度 v



(c) 压力 p

图 5.32 流场运动信息再现结果的 RMSE 对比

对于顺流向速度 u ，工况 3.1 至工况 3.4 的 RMSE 浮动区间分别是 (0.0057, 0.0111)、(0.0053, 0.0168)、(0.0095, 0.0181) 以及 (0.0033, 0.0159)，幅值分别是 0.0111、0.0168、0.0181 和 0.0159。

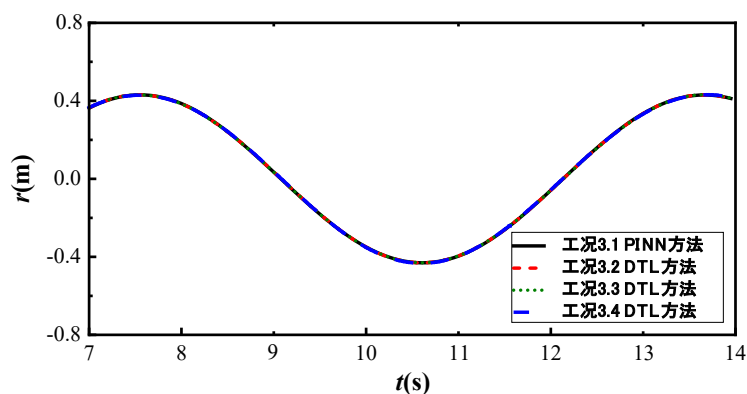
对于横流向速度 v ，工况 3.1 至工况 3.4 的 RMSE 浮动区间分别是 (0.0045, 0.0108)、(0.0054, 0.0149)、(0.0076, 0.0165) 以及 (0.0035, 0.0149)，幅值分别是 0.0108、0.0149、0.0165 和 0.0149。

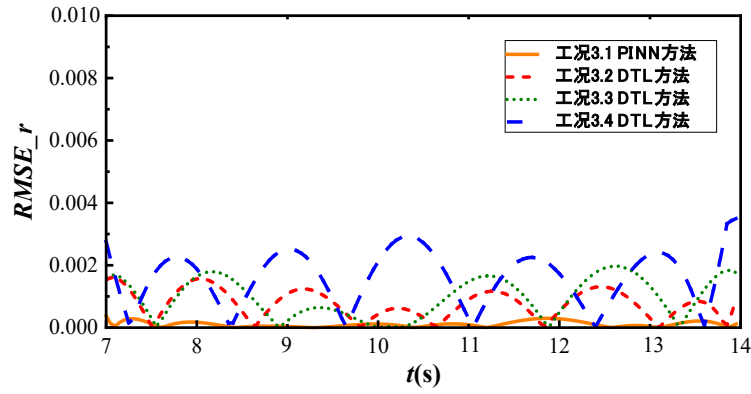
对于压力 p ，工况 3.1 至工况 3.4 的 RMSE 浮动区间分别是(0.0074, 0.0169)、(0.0036, 0.0115)、(0.0066, 0.0168) 以及 (0.0047, 0.0145)，幅值分别是 0.0169、0.0115、0.0168 和 0.0145。

图 5.32 量化地比较了算例 3 的四个工况对流场运动信息再现结果的 RMSE。综合来看，采用 DTL 方法的三个工况将顺流向速度 u 、横流向速度 v 以及压力 p 再现结果的 RMSE 均控制在 0.02 以下。参考实际的流场数据以及工况 3.1 的再现结果，基于 DTL 方法的三个工况对流场运动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

2) 固体振动信息再现情况

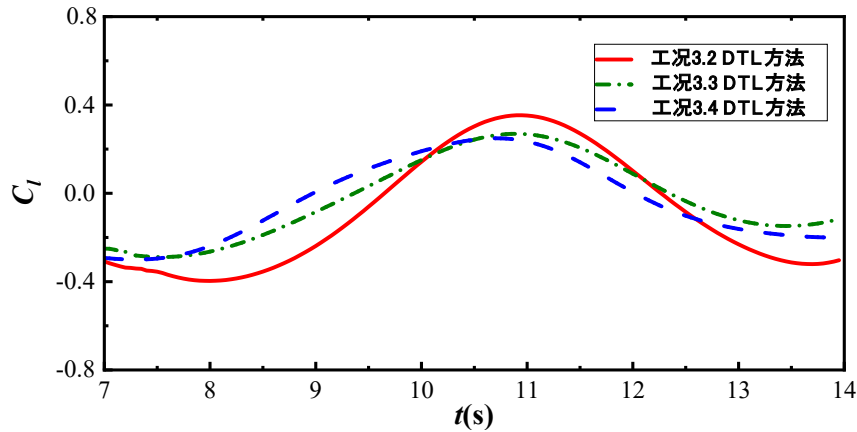
结构振动信息主要包括固体结构振动位移 r 、绕流升力 F_l 以及绕流阻力 F_d 。算例 3 目标域样本集的时域范围是 7-14 s，即四个工况需准确再现 140 个时间步的固体结构振动位移 r 。图 5.33 是算例 3 中四个工况对固体结构振动位移 r 的再现结果，图 5.34 是再现结果的 RMSE 对比。

图 5.33 算例 3 中四个工况对固体结构振动位移 r 的再现结果


 图 5.34 固体结构振动位移 r 再现结果的 RMSE 对比

对于固体结构振动位移 r ，工况 3.1 至工况 3.4 的 RMSE 浮动区间分别为 $(1.07\text{E-}06, 3.79\text{E-}04)$ 、 $(3.77\text{E-}06, 1.61\text{E-}03)$ 、 $(9.00\text{E-}07, 1.97\text{E-}03)$ 以及 $(1.61\text{E-}05, 3.52\text{E-}03)$ ，幅值分别为 $3.79\text{E-}04$ 、 $1.61\text{E-}03$ 、 $1.97\text{E-}03$ 以及 $3.52\text{E-}03$ 。在单个时间步内，再现结果的最大相对误差分别为 1.91%、3.88%、4.78% 以及 8.51%。

基于流场运动信息的再现结果和流固耦合面的边界条件，可以积分计算得到绕流升力 F_l 和绕流阻力 F_d ，积分表达式为式 (3.5) 和式 (3.6)。为了更直观地验证算例 3 中采用 DTL 方法的 3 个工况再现绕流力的准确性，通过式 (3.22) 和式 (3.23) 将绕流升力 F_l 和绕流阻力 F_d 转化为绕流升力系数 C_l 和绕流阻力系数 C_d ，如图 5.35 所示。最后以绕流升力系数幅值 $|C_l|_{max}$ 和绕流阻力系数均值 $\overline{C_d}$ 为指标，将再现结果与参考文献^[125] (雷诺数 $Re=100$) 进行对比，如表 5.12 所示。


 (a) 流体升力系数 C_l

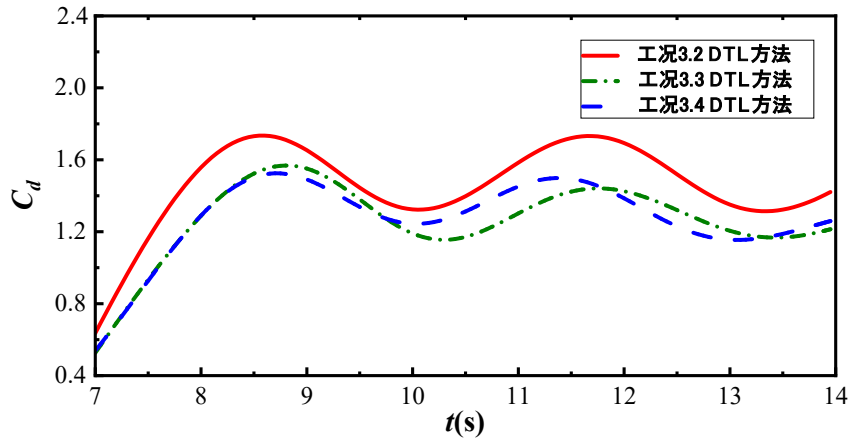
(b) 流体阻力系数 C_d

图 5.35 算例 3 中采用 DTL 方法的三个工况对流体力系数的再现结果

表 5.12 流体力系数再现结果正确性验证

序号	再现结果 $ C_l _{max}$	参考文献 $ C_l _{max}$	误差	再现结果 $\overline{C_d}$	参考文献 $\overline{C_d}$	误差
工况 3.2	0.398	0.355	0.043	1.463	1.372	0.091
工况 3.3	0.289	0.355	0.066	1.274	1.372	0.098
工况 3.4	0.301	0.355	0.054	1.279	1.372	0.093

对于流体升力系数幅值 $|C_l|_{max}$ ，工况 3.2 至工况 3.4 的再现结果分别为 0.398、0.289 以及 0.301，参考文献^[125]的数值模拟结果为 0.355，再现结果的相对误差分别为 12.1%、18.6% 以及 15.2%。

对于流体阻力系数均值 $\overline{C_d}$ ，工况 3.2 至工况 3.4 的再现结果分别为 1.463、1.274 以及 1.279，参考文献^[125]的数值模拟结果为 1.372，再现结果的相对误差分别为 6.7%、7.1% 以及 6.8%。

综合来看，参考实际的结构振动信息以及该文献的模拟结果，基于 DTL 方法的三个工况对结构振动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本数据。

5.5 本章小结

在实际应用中，用于训练的 VIV 数据集往往具有“小样本、高维度、稀疏”等特征，导致传统 PINN 算法的适用范围有限。因此，本章提出了基于 TL 增强的 PINN 模型来研究 VIV。结合迁移策略，利用源模型中的 VIV 公共特征知识来加强目标任务的学习，从而提高学习的效率，且无需大量标注样本集。本章共设置三个具有不同数据来源及规模的数值算例，来评估 DTL 代理模型在完成小

样本目标任务时的运行精度及迁移泛化能力。以训练成本和计算精度为指标，对比了 DTL 模型与传统 PINN 模型的表现性能，以验证 TL 方法对传统 PINN 模型的增强作用。本章的主要结论如下所示：

(1)DTL 代理模型对流场运动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本集。

对于顺流向速度 u ，算例 1 至 3 将单个时间步的再现结果 RMSE 分别控制在 $8.0E-05$ 、 $9.0E-05$ 以及 0.02 以下；

对于横流向速度 v ，算例 1 至 3 将单个时间步的再现结果 RMSE 分别控制在 $1.0E-04$ 、 $1.3E-04$ 以及 0.02 以下；

对于压力 p ，算例 1 至 3 将单个时间步的再现结果 RMSE 分别控制在 $9.0E-05$ 、 $9.0E-05$ 以及 0.02 以下。

(2)DTL 代理模型对结构振动信息的再现精度与传统 PINN 算法基本吻合，且无需标注大量样本集。

对于固体结构振动位移 r ，算例 1 至 3 单个时间步的最大再现相对误差分别为 5.24% 、 11.71% 以及 8.51% ；

对于绕流升力系数幅值 $|C_l|_{max}$ ，算例 1 至 3 的最大再现相对误差分别为 10.4% 、 10.6% 以及 15.2% ；

对于流体阻力系数均值 $\overline{C_d}$ ，算例 1 至 3 的最大再现相对误差分别为 7.6% 、 6.7% 以及 6.8% 。

(3) 与传统 PINN 模型相比，DTL 代理模型凭借更少的训练成本完成相同的目标任务。

算例 1 中，工况 1.2 至工况 1.4 (DTL 方法) 的数据成本分别降低为工况 1.1 (PINN 方法) 的 $1/2$ 、 $1/4$ 和 $1/8$ ，计算时间成本分别降低为 $7/16$ 、 $1/6$ 以及 $1/11$ ；

算例 2 中，工况 2.2 至工况 2.4 (DTL 方法) 的数据成本分别降低为工况 2.1 (PINN 方法) 的 $1/2$ 、 $1/4$ 和 $1/8$ ，计算时间成本分别降低为 $2/5$ 、 $1/4$ 以及 $1/12$ ；

算例 3 中，工况 3.2 至工况 3.4 (DTL 方法) 的数据成本分别降低为工况 3.1 (PINN 方法) 的 $1/2$ 、 $1/4$ 和 $1/8$ ，计算时间成本分别降低为 $1/2$ 、 $1/5$ 以及 $1/9$ 。

(4) 三个数值算例的样本集具有不同数据来源及数据规模，在 3 个算例中，DTL 代理模型完成小样本目标任务时的运行精度均与 PINN 算法基本吻合，证明该模型具有良好的迁移泛化能力。

第 6 章 结论与展望

6.1 主要工作及结论

本文基于物理驱动深度学习对二维圆柱结构 VIV 展开系统研究，主要包括以下内容：梳理 VIV、PINN 以及 TL 的基础理论，为建立强耦合 PINN 模型和 CFD 数值模拟提供理论支持；针对传统 FEM 和数据驱动 DNN 计算精度及效率低下的问题，通过加强物理信息对神经网络的约束，建立基于物理驱动深度学习的强耦合 VIV 模型，同时基于 PINN 比较分析了强弱耦合 VIV 模型的表现性能；为了提高在“稀疏”数据条件下 PINN 求解 VIV 正反问题的计算精度及效率，提出分时段递推训练的方法，采用分时段不同训练配置点采样策略，通过有效利用前一时段的网络训练特征以及预测信息，后一时段逐渐减少训练配置点数，实现仅利用有限时域长度信息来预测更长时间段的 VIV 信息；为了进一步提高在“稀疏”数据条件下 PINN 方法的计算精度及效率，提出 TL 增强的 PINN 模型，结合迁移策略，源模型中的 VIV 公共特征知识可以加强目标任务的学习，从而提高学习效率，且无需大量标注样本集。本文的主要成果及结论如下所示：

(1) 以时间 t 、流场空间坐标 (x, y) 为输入变量构建数据驱动 DNN，并通过全连接层输出流场质点顺流向速度 $u(t, x, y)$ 、横流向速度 $v(t, x, y)$ 、压力 $p(t, x, y)$ 以及固体结构振动位移 $r(t)$ 等变量。在此基础上，将描述粘性不可压缩流体动量守恒的二维 N-S 方程以及 VIV 流固耦合条件方程同时嵌入 DNN 结构体系中，加强物理信息对神经网络的约束，建立强耦合涡激振动 PINN 模型。同时基于计算流体理论和动网格技术对层流状态 ($Re=200$) 的 VIV 进行数值模拟，将 CFD 模拟数据进行标注构建训练样本集。

(2) 与弱耦合 PINN 模型相比，强耦合 PINN 模型在先验知识中添加了流固耦合条件方程，强化了物理信息对神经网络的约束，凭借较少的训练成本获得更高的运行精度。

对于流场运动信息，强耦合 PINN 模型将顺流向速度 u 、横流向速度 v 以及压力 p 单时间步内再现结果的 RMSE 至少降低 45.2%、48.8% 以及 44.3%；对于结构振动信息，强耦合 PINN 模型将固体结构振动位移 r 单个时间步的最大相对再现误差降低 21.7%，流体升力系数幅值 $|C_l|_{max}$ 和流体阻力系数均值 $\overline{C_d}$ 的相对再现误差分别降低至 3.0% 和 4.5%；数据成本和时间成本分别下降至弱耦合 PINN 模型的 1/2 和 3/5。

(3) 在正问题中, 通过优化训练方法及空间配置点采样策略, 充分利用预测信息, 减少空间配置点的要求, 有效提高模型计算效率, 可以在“稀疏”数据条件下实现流场运动信息的时空预测以及结构振动信息的时域预测。

对于流场运动信息, 顺流向速度 u 、横流向速度 v 以及压力 p 单个时间步的预测结果 RMSE 分别控制在 $2.5E-04$ 、 $3.0E-04$ 以及 $5.0E-04$ 以下; 对于结构振动信息, 固体结构振动位移 r 、流体升力系数幅值 $|C_l|_{max}$ 和流体阻力系数均值 $\overline{C_d}$ 单个时间步的相对预测误差分别控制在 9.73%、4.5%和 3.0%以下。

在反问题中, 该方法同样可以准确识别结构阻尼 c 和结构刚度 k 。结构阻尼 c 的相对识别误差为 7.03%, 结构刚度 k 的相对识别误差为 2.76%。

(4) DTL 代理模型对流场运动信息以及结构振动信息的再现精度与传统 PINN 算法基本吻合, 且无需标注大量样本集。在三个数值算例中, DTL 代理模型完成小样本目标任务的运行精度均与 PINN 算法基本吻合, 证明该模型具有良好的迁移泛化能力。

对于流场运动信息, 基于 CFD 模拟数据的算例 1 和 2 将顺流向速度 u 、横流向速度 v 以及压力 p 单个时间步的再现结果 RMSE 分别控制在 $9.0E-05$ 、 $1.3E-04$ 以及 $9.0E-05$ 以下, 基于实验数据的算例 3 将其均控制在 0.02 以下; 对于结构振动信息, 三个数值算例的固体结构振动位移 r 、流体升力系数幅值 $|C_l|_{max}$ 和流体阻力系数均值 $\overline{C_d}$ 单个时间步的最大相对再现误差分别为 11.71%、15.2%以及 7.6%。

6.2 课题展望

本文目前已经获得了一些初步成果, 但是受限于研究时间与作者能力, 仍旧存在部分遗留问题需要进一步探讨。

(1) 本文在建立 VIV 物理模型时, 忽略了结构在顺流向方向上的振动响应, 未将其作为网络输出变量嵌入 PINN 模型中。为了更准确地反映 VIV 响应机理, 需要进一步优化强耦合 PINN 模型。

(2) 本文仅研究了平面内二维圆柱结构 VIV 问题, 未拓展至三维立体物理模型。三维柔性立管的 VIV 还应考虑来流情况随深度的变化、立管的扭转转动等情况, 值得进一步探究。

(3) 本文仅研究了层流模型的 VIV 问题, 未扩展至高雷诺数的湍流模型。构建基于湍流模型的涡激振动 PINN 模型时, 需重点考虑流体的黏性系数, 以便于进一步优化 PINN 模型。

(4) 本文仅研究了圆柱结构的 VIV 问题, 未拓展至其他结构截面形状, 后

续研究可以深入研究截面形状对 VIV 的影响差异。

(4) 受限于研究时间和作者能力, 在数据驱动 DNN 添加的物理信息十分有限, 未来可以探索更多的网络约束条件, 以实现无数据驱动的网络训练。

参考文献

- [1] 及春宁, 李非凡, 陈威霖等. 圆柱涡激振动研究进展与展望. 海洋技术学报, 2015, 34(01): 106-118
- [2] 刘坤. 减少或抑制海洋立管涡激振动专利技术综述. 化学工程与装备, 2015, (05): 112-116
- [3] 廖海黎, 李明水, 马存明等. 桥梁风工程 2020 年度研究进展. 土木与环境工程学报, 2021, 43(S1): 23-30
- [4] 陶涛, 龙凯, 白欣鉴等. 风电机组高柔塔二阶涡激振动特性研究. 太阳能学报, 2022, 43(02): 498-503
- [5] 文广龙. 低雷诺数下柱体结构涡激振动抑制研究: [硕士学位论文]. 湘潭: 湘潭大学, 2021
- [6] 闫超群. 串联双圆柱管涡激振动实验研究: [硕士学位论文]. 大连: 大连理工大学, 2021
- [7] 徐万海, 覃雯琪, 栾英森等. 不同措施对倾斜柔性圆柱结构涡激振动抑制研究. 第二十九届全国水动力学研讨会论文集(下册). 北京: 海洋出版社, 2018. 349-355
- [8] Ramesh Nepali. 低雷诺数下串列布置双方柱双自由度涡激振动研究: [硕士学位论文]. 上海: 上海交通大学, 2020
- [9] 葛耀君, 赵林, 许坤. 大跨桥梁主梁涡激振动研究进展与思考. 中国公路学报, 2019, 32(10): 1-18
- [10] 付小波. 风力发电机塔筒吊装过程中的涡激振动现象及其抑制研究. 四川水力发电, 2020, 39(S2): 77-80
- [11] 桑建设. 桥梁结构涡激振动实例及减振措施比较研究. 价值工程, 2013, 32(24): 100-102
- [12] 宋特. 变截面钢箱梁涡激振动相关特性研究: [硕士学位论文]. 西安: 长安大学, 2020
- [13] 王琦. TMDI 对大跨钢箱连续梁桥涡激振动的控制研究: [硕士学位论文]. 北京: 北京交通大学, 2021
- [14] 姚熊亮, 陈起富. 串列双圆柱在均匀流场中涡激振动综述. 哈尔滨船舶工程学院学报, 1994, (04): 1-16
- [15] 王远岑. 海洋立管多模态涡激振动分岔研究: [博士学位论文]. 天津: 天津大学, 2019
- [16] 孙云卿. 立管涡激振动分岔及多模态相互作用研究: [硕士学位论文]. 天津: 天津大学, 2019
- [17] 唐国强, 滕斌, 吕林等. 深海柔性立管涡激振动经验模型建立及应用. 中国海洋平台, 2010, 25(03): 12-16
- [18] 秦伟, 康庄, 孙丽萍等. 并列双圆柱涡激振动的经验性模型研究. 海洋工程, 2013, 31(02): 11-18
- [19] 娄敏. 海洋输流立管涡激振动试验研究及数值模拟: [博士学位论文]. 青岛: 中国海洋大学, 2017
- [20] Hartlen R T, Currie I G. Lift-oscillator model of vortex-induced vibration. Journal of the Engineering Mechanics Division, 1970, 96(5): 577-591
- [21] Iwan W D, Blevins R D. A model for vortex induced oscillation of structures. Journal of Applied Mechanics, 1974, 41(3): 581-586

- [22] Krenk S, Nielsen S R K. Energy balanced double oscillator model for vortex-induced vibrations. *Journal of Engineering Mechanics*, 1999, 125(3): 263-271
- [23] Xiaomin LI, Haiyan Guo, Fanshun Meng. Nonlinear coupled in-line and cross-flow vortex-induced vibration analysis of top tensioned riser. *China Ocean Engineering*, 2010, 24(4): 749-758
- [24] 陈东阳. 海洋柔性结构流固耦合动力学研究: [博士学位论文]. 南京: 南京理工大学, 2018
- [25] Lucor D, Imas L, Karniadakis G E. Vortex dislocations and force distribution of long flexible cylinders subjected to sheared flows. *Journal of Fluids and Structures*, 2001, 15(3-4): 641-650
- [26] Skaugset K B, Larsen C M. Direct numerical simulation and experimental investigation on suppression of vortex induced vibrations of circular cylinders by radial water jets. *Flow, turbulence and combustion*, 2003, 71(1): 35-59
- [27] Guilmineau E, Queutey P. Numerical simulation of vortex-induced vibration of a circular cylinder with low mass-damping in a turbulent flow. *Journal of fluids and structures*, 2004, 19(4): 449-466
- [28] 郭宇. 深海立管涡激振动的大涡模拟与涡振疲劳累积损伤分析: [硕士学位论文]. 哈尔滨: 哈尔滨工业大学, 2012
- [29] Muralidharan K, Muddada S, Patnaik B S V. Numerical simulation of vortex induced vibrations and its control by suction and blowing. *Applied Mathematical Modelling*, 2013, 37(1-2): 284-307
- [30] Bargi K, Tamimi V, Zeinoddini M. VIV of Tapered Cylinders: 3D LES Numerical Simulation. *International Journal of Maritime Technology*, 2015, 3: 17-31
- [31] 杨志斌. 基于 RANS 的圆柱涡激振动数值模拟: [硕士学位论文]. 天津: 天津大学, 2016
- [32] 李骏, 李威. 基于 SST $k-\omega$ 湍流模型的二维圆柱涡激振动数值仿真计算. *舰船科学技术*, 2015, 37(02): 30-34
- [33] 曾攀. 圆柱涡激振动特性的数值模拟研究: [硕士学位论文]. 天津: 天津大学, 2018
- [34] 赵燕, 林伟群, 杜晓庆等. 悬索桥双吊索尾流致涡激振动的大涡模拟. *振动与冲击*, 2019, 38(16): 129-136
- [35] 翟立宾. 基于机器学习方法的柔性圆柱涡激振动预报: [硕士学位论文]. 天津: 天津大学, 2019
- [36] 高喜峰, 翟立宾, 徐万海. 应用 BP 神经网络的柔性圆柱涡激振动预报. *哈尔滨工程大学学报*, 2020, 41(08): 1150-1155
- [37] 孟杨. 圆柱绕流及其涡激振动的物理模型与机器学习相结合方法研究: [硕士学位论文]. 大连: 大连海事大学, 2020
- [38] 廖洋洋. 基于神经网络的柔性立管涡激振动 (VIV) 疲劳损伤研究: [硕士学位论文]. 天津: 天津大学, 2021
- [39] Misaka T. Estimation of Vortex-Induced Vibration Based on Observed Wakes Using Computational Fluid Dynamics-Trained Deep Neural Network. *Journal of Fluids Engineering*, 2021, 143(10): 104-109
- [40] Foster J A, De Troyer T, Runacres M C, et al. Modelling vortex-induced loads using Recurrent Neural Networks. *IFAC-PapersOnLine*, 2021, 54(20): 32-37
- [41] Raissi M, Wang Z, Triantafyllou M S, et al. Deep learning of vortex-induced vibrations.

- Journal of Fluid Mechanics, 2019, 861: 119-137
- [42] Wang J X, Wu J, Ling J, et al. A comprehensive physics-informed machine learning framework for predictive turbulence modeling. arXiv preprint arXiv: 1701.07102, 2017
- [43] Cheng C, Xu P F, Li Y Z, et al. Deep learning based on PINN for solving 2 D0F vortex induced vibration of cylinder with high Reynolds number. arXiv preprint arXiv: 2106.01545, 2021
- [44] Kharazmi E, Fan D, Wang Z, et al. Inferring vortex induced vibrations of flexible cylinders using physics-informed neural networks. Journal of Fluids and Structures, 2021, 107: 103367
- [45] Li S, Laima S, Li H. Physics-guided deep learning framework for predictive modeling of bridge vortex-induced vibrations from field monitoring. Physics of Fluids, 2021, 33(3): 037113
- [46] Bai X D, Zhang W. Machine learning for vortex induced vibration in turbulent flow. Computers & Fluids, 2022, 235: 105266
- [47] Cai S, Mao Z, Wang Z, et al. Physics-informed neural networks (PINNs) for fluid mechanics: A review. Acta Mechanica Sinica, 2022, 1-12
- [48] 李野, 陈松灿. 基于物理信息的神经网络: 最新进展与展望. 计算机科学, 2022, 49(04): 254-262
- [49] Raissi M, Perdikaris P, Karniadakis G E. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. arXiv preprint arXiv: 1711.10561, 2017
- [50] Fang Z, Zhan J. Deep physical informed neural networks for metamaterial design. IEEE Access, 2019, 8: 24506-24513
- [51] Mao Z, Jagtap A D, Karniadakis G E. Physics-informed neural networks for high-speed flows. Computer Methods in Applied Mechanics and Engineering, 2020, 360: 112789
- [52] Raissi M, Yazdani A, Karniadakis G E. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. Science, 2020, 367(6481): 1026-1030
- [53] Jin X, Cai S, Li H, et al. NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations. Journal of Computational Physics, 2021, 426: 109951
- [54] Chen Y, Lu L, Karniadakis G E, et al. Physics-informed neural networks for inverse problems in nano-optics and metamaterials. Optics express, 2020, 28(8): 11618-11633
- [55] Guo H, Zhuang X, Liang D, et al. Stochastic groundwater flow analysis in heterogeneous aquifer with modified neural architecture search (NAS) based physics-informed neural networks using transfer learning. arXiv preprint arXiv: 2010.12344, 2020
- [56] Wang C, Bentinevna E, Zhou W, et al. Physics-informed neural network super resolution for advection-diffusion models. arXiv preprint arXiv: 2011.02519, 2020
- [57] Misyris G S, Venzke A, Chatzivasileiadis S. Physics-informed neural networks for power systems. 2020 IEEE Power & Energy Society General Meeting (PESGM), 2020: 1-5
- [58] Shukla K, Di Leoni P C, Blackshire J, et al. Physics-informed neural network for ultrasound nondestructive quantification of surface breaking cracks. Journal of Nondestructive Evaluation, 2020, 39(3): 1-20
- [59] Sahli Costabal F, Yang Y, Perdikaris P, et al. Physics-informed neural networks for cardiac activation mapping. Frontiers in Physics, 2020, 8: 42
- [60] Stiasny J, Misyris G S, Chatzivasileiadis S. Physics-informed neural networks for non-linear

- system identification for power system dynamics. 2021 IEEE Madrid PowerTech, 2021: 1-6
- [61] 曾靖文. 基于物理信息神经网络的 Burgers 方程应用研究: [硕士学位论文]. 新疆: 新疆大学, 2021
- [62] Haghghat E, Raissi M, Moure A, et al. A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 2021, 379: 113741
- [63] 王伟杰. 基于物理信息神经网络的连续结晶过程建模与优化: [硕士学位论文]. 大连: 大连理工大学, 2021
- [64] Zhang E, Dao M, Karniadakis G E, et al. Analyses of internal structures and defects in materials using physics-informed neural networks. *Science advances*, 2022, 8(7): eabk0644
- [65] Gokhale G, Claessens B, Develder C. Physics informed neural networks for control oriented thermal modeling of buildings. *Applied Energy*, 2022, 314: 118852
- [66] Iman M, Rasheed K, Arabnia H R. A Review of Deep Transfer Learning and Recent Advancements. *arXiv preprint arXiv: 2201.09679*, 2022
- [67] Li W, Huang R, Li J, et al. A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges. *Mechanical Systems and Signal Processing*, 2022, 167: 108487
- [68] Neyshabur B, Sedghi H, Zhang C. What is being transferred in transfer learning. *Advances in neural information processing systems*, 2020, 33: 512-523
- [69] Zhuang F, Qi Z, Duan K, et al. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2020, 109(1): 43-76
- [70] Zafar T, Wang Z. Time-dependent reliability prediction using transfer learning. *Structural and Multidisciplinary Optimization*, 2020, 62(1): 147-158
- [71] Kaur T, Gandhi T K. Deep convolutional neural networks with transfer learning for automated brain image classification. *Machine Vision and Applications*, 2020, 31(3): 1-16
- [72] Chen J, Chen J, Zhang D, et al. Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture*, 2020, 173: 105393
- [73] Wu Z, Jiang H, Zhao K, et al. An adaptive deep transfer learning method for bearing fault diagnosis. *Measurement*, 2020, 151: 107227
- [74] Shu Y, Li B. Surface Defect Detection and Recognition Method for Multi-Scale Commutator Based on Deep Transfer Learning. *Arabian Journal for Science and Engineering*, 2021: 1-12
- [75] Ye R, Dai Q. Implementing transfer learning across different datasets for time series forecasting. *Pattern Recognition*, 2021, 109: 107617
- [76] Gupta V, Choudhary K, Tavazza F, et al. Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. *Nature communications*, 2021, 12(1): 1-10
- [77] Lin Y, Nie Z, Ma H. Dynamics-based cross-domain structural damage detection through deep transfer learning. *Computer-Aided Civil and Infrastructure Engineering*, 2022, 37(1): 24-54
- [78] 宋虹, 黄维平, 付雪鹏. 基于模型试验和数值模拟的柔性串列圆柱体涡激振动研究. *振动与冲击*, 2020, 39(06): 64-73
- [79] 吴晓笛. 基于浸入边界-格子 Boltzmann 通量法的椭圆柱涡激振动数值研究: [博士学位论文]. 哈尔滨: 哈尔滨工业大学, 2019
- [80] 张杰. 深海立管参激-涡激联合振动与疲劳特性研究: [博士学位论文]. 天津: 天津大

- 学, 2014
- [81] 孔祥鑫. 多因素影响下柱体绕流及其涡激振动特性研究: [硕士学位论文]. 青岛: 青岛科技大学, 2018
- [82] 马焯璇. 柔性圆柱涡激振动的水动力学系数识别: [硕士学位论文]. 天津: 天津大学, 2017
- [83] 王亚非. 双自由度圆柱体的涡激振动研究: [硕士学位论文]. 哈尔滨: 哈尔滨工程大学, 2016
- [84] Bouwmans T, Javed S, Sultana M, et al. Deep neural network concepts for background subtraction: A systematic review and comparative evaluation. *Neural Networks*, 2019, 117: 8-66
- [85] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural computation*, 1997, 9(8): 1735-1780
- [86] Cortes C, Vapnik V. Support-vector networks. *Machine learning*, 1995, 20(3): 273-297
- [87] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. *Neural computation*, 2006, 18(7): 1527-1554
- [88] Liu W, Wang Z, Liu X, et al. A survey of deep neural network architectures and their applications. *Neurocomputing*, 2017, 234: 11-26
- [89] Braham M, Van Droogenbroeck M. Deep background subtraction with scene-specific convolutional neural networks. 2016 international conference on systems, signals and image processing (IWSSIP), 2016: 1-4
- [90] Gast J, Roth S. Lightweight probabilistic deep networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018: 3369-3378
- [91] Deng Y, Ren Z, Kong Y, et al. A hierarchical fused fuzzy deep neural network for data classification. *IEEE Transactions on Fuzzy Systems*, 2016, 25(4): 1006-1012
- [92] Creswell A, White T, Dumoulin V, et al. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 2018, 35(1): 53-65
- [93] 刘超, 张晓晖, 胡清平. 超低照度下微光图像增强神经网络损失函数设计分析. *国防科技大学学报*, 2018, 40(04): 67-73
- [94] 李一波, 郭培宜, 张森悦. 深度卷积神经网络中激活函数的研究. *计算机技术与发展*, 2021, 31(09): 61-66
- [95] 袁群勇. 深度神经网络的训练优化方法研究: [博士学位论文]. 广州: 华南理工大学, 2020
- [96] von Rueden L, Mayer S, Beckh K, et al. Informed Machine Learning-A Taxonomy and Survey of Integrating Knowledge into Learning Systems. *arXiv preprint arXiv: 1903.12394*, 2019
- [97] Pang G, D'Elia M, Parks M, et al. nPINNs: nonlocal Physics-Informed Neural Networks for a parametrized nonlocal universal Laplacian operator. *Journal of Computational Physics*, 2020, 422: 109760
- [98] Wang S, Zhang H, Jiang X. Fractional Physics-informed Neural Networks for Time-fractional Phase Field Models. 2021
- [99] Meng X, Li Z, Zhang D, et al. PPINN: Parareal physics-informed neural network for time-dependent PDEs. *Computer Methods in Applied Mechanics and Engineering*, 2020, 370: 113250
- [100] Jagtap A D, Kharazmi E, Karniadakis G E. Conservative physics-informed neural

- networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 2020, 365: 113028
- [101] Dwivedi V, Parashar N, Srinivasan B. Distributed physics informed neural network for data-efficient solution to partial differential equations. *arXiv preprint arXiv: 1907.08967*, 2019
- [102] Yang L, Meng X, Karniadakis G E. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 2021, 425: 109913
- [103] Kharazmi E, Zhang Z, Karniadakis G E. Variational physics-informed neural networks for solving partial differential equations. *arXiv preprint arXiv: 1912.00873*, 2019
- [104] Lu L, Meng X, Mao Z, et al. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 2021, 63(1): 208-228
- [105] Jo H, Son H, Hwang H J, et al. Deep neural network approach to forward-inverse problems. *arXiv preprint arXiv: 1907.12925*, 2019
- [106] 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. *Journal of Computational physics*, 2019, 378: 686-707
- [107] Mishra S, Molinaro R. Estimates on the generalization error of Physics Informed Neural Networks (PINNs) for approximating a class of inverse problems for PDEs. *arXiv preprint arXiv: 2007.01138*, 2020
- [108] Chen Y, Lu L, Karniadakis G E, et al. Physics-informed neural networks for inverse problems in nano-optics and metamaterials. *Optics express*, 2020, 28(8): 11618-11633
- [109] Jagtap A D, Kawaguchi K, Karniadakis G E. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *Journal of Computational Physics*, 2020, 404: 109136
- [110] Jagtap A D, Kawaguchi K, Em Karniadakis G. Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks. *Proceedings of the Royal Society A*, 2020, 476(2239), 20200334
- [111] Wang S, Teng Y, Perdikaris P. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 2021, 43(5): A3055-A3081
- [112] Wang S, Yu X, Perdikaris P. When and why PINNs fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 2022, 449: 110768
- [113] Braza M, Chassaing P, Minh H H. Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *Journal of fluid mechanics*, 1986, 165: 79-130
- [114] Higuchi H, Zhang J F, Muzas B, et al. Experimental study on unsteady flows and multiple near-wake patterns behind slotted disks. *14th Aerodynamic Decelerator Systems Technology Conference*, 1997: 1480
- [115] 魏志理, 孙德军, 尹协远. 圆柱尾迹流场中横向振荡翼型绕流的数值模拟. *水动力学研究与进展(A 辑)*, 2006, (03): 299-308
- [116] 常书平, 王永生, 庞之洋. 用基于 SST 模型的 DES 方法数值模拟圆柱绕流. *舰船科学技术*, 2009, 31(02): 30-33

- [117] Pan S J, Yang Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 2009, 22(10): 1345-1359
- [118] Lu H, Zhang L, Cao Z, et al. When unsupervised domain adaptation meets tensor representations. *Proceedings of the IEEE international conference on computer vision*, 2017: 599-608
- [119] Zhao P, Hoi S C H, Wang J, et al. Online transfer learning. *Artificial intelligence*, 2014, 216: 76-102
- [120] Ammar H B, Eaton E, Luna J M, et al. Autonomous cross-domain knowledge transfer in lifelong policy gradient reinforcement learning. *Twenty-fourth international joint conference on artificial intelligence*, 2015
- [121] Shen C, Guo Y. Unsupervised heterogeneous domain adaptation with sparse feature transformation. *Asian conference on machine learning*, 2018: 375-390
- [122] Bousmalis K, Trigeorgis G, Silberman N, et al. Domain separation networks. *Advances in neural information processing systems*, 2016, 29
- [123] Keneshloo Y, Ramakrishnan N, Reddy C K. Deep transfer reinforcement learning for text summarization. *Society for Industrial and Applied Mathematics*, 2019: 675-683
- [124] Saito K, Watanabe K, Ushiku Y, et al. Maximum classifier discrepancy for unsupervised domain adaptation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018: 3723-3732
- [125] 何鸿涛. 圆柱绕流及其控制的数值模拟研究: [硕士学位论文]. 北京: 北京交通大学, 2009

致谢

疫情之下的毕业季是艰难且充满不确定性的，是缓慢的、煎熬的、晦涩且孤独的。收到教育部盲审合格的那一瞬间，便赋予了今天灿烂至极的美好。是论文的完稿之日，是开始鼓起勇气向同济的七年时光说再见的日子，也是那颗数月来悬着的心终于落地的日子，坚定且铿锵有力。

感谢我的导师唐和生研究员，在课题进展以及论文成稿过程中对我的悉心指导。受教于唐老师的三年时光中，看到了他对于学术研究的专注与严谨，也感受到了他对于学生的默默关心与认真负责。每次在深夜里收到老师对于论文的批改意见，我都深受感动，是他为我们的顺利毕业保驾护航。希望老师今后的生活一切顺遂，身体永远健康。

感谢和泉教研室的薛松涛教授，薛老师早已声名远扬但却为人低调和蔼，虽与薛老师只有过数面之缘，仍然从心底里对薛老师感到无限敬佩。同时感谢和泉教研室的谢丽宇副教授，谢老师是和泉大家庭的“大总管”，在我们的学习以及日常生活中总是无私地提供帮助。

感谢我的父母，永远做我最温暖的港湾，在背后坚定地支持我所做的每一个决定。终于我结束了我的学生生涯，希望往后的日子里能慢慢回报你们对我所有的爱与包容，你们辛苦了。

感谢廖洋洋师兄，在学术研究上给予我最坚强的支持，在封校期间多次舒缓我因毕业压力带来的负面情绪。师兄为人真诚、洒脱，希望他未来的科研之路一切顺利。

感谢郭雪媛师姐、吴雅琴师姐以及班鑫磊师姐，命运像是偷偷安排好一样，三个大姐姐依次陪伴我度过研一、研二和研三的生活。因为你们，我的研究生生活更加自由与自我。希望你们未来的生活美满幸福，永远被爱。

感谢我的同门黎思维、郭泰昆、龚玲、郑志泉、曹跃辉与姜康，我们相互鼓励与加油，向毕业的目标发起最后的冲刺。感谢师姐仝运佳、谢雅娟、薛智奇、曾璟琳、冉凌霄，师兄李宪之、张力、赵涛涛、王云蹇、管帅、陈子暘、康建飞、濮昱、陈豪、易卓然，师弟王泽宇、贺畅、何展朋、夏子涵、车兴儒、龙盼、赵锦桐等，师妹丁宁等。你们是我在教研室生活中最有意义的记忆碎片。

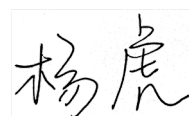
感谢我的好友任远、蔡筱君和张家骏，很少跟你们说谢谢，但在此刻很想对你们说声谢谢，谢谢你们让我人生的大多数时刻都觉得这个世界很美好。祝我们爬不同的山时，还能回到同一条路上，不常常见面，却时时惦记。希望你们能爱你所爱，行你所行，听从你心，无问西东。祝你们幸福一生。

感谢 BLACKPINK，你们是我生命中很重要的光，是我努力想去靠近的人，希望你们拥有更加光明且灿烂的未来。

感谢疫情下所有坚守岗位的医护人员、老师以及志愿者们，你们辛苦了，我们终将战胜疫情，向你们致敬。

最后感谢我自己，不忘初心地走过在上海、在同济的七年时光。也想对自己说声：干得不错，辛苦了。

一期一会，这是结束，亦是新的开始，最后感谢每一次遇见。

Handwritten signature in black ink, reading '杨晨' (Yang Chen).

2022年5月12日凌晨于彰武八号楼1412

个人简历、在读期间发表的学术成果

个人简历:

杨虎，男，1997年6月生。

2019年6月毕业于同济大学土木工程专业，获工学学士学位。

2019年9月入同济大学土木工程学院结构防灾减灾工程系攻读硕士学位。

已发表论文:

- [1] 唐和生, 杨虎. 涡激振动的深度迁移学习. 第二十届全国结构风工程会议暨第六届全国风工程研究生论坛论文集. 广州: 华南理工大学出版社, 2022

