

物理信息神经网络求解纳维 - 斯托克斯方程组正反问题

李述威

兰州理工大学理学院, 甘肃 兰州

收稿日期: 2025年8月9日; 录用日期: 2025年9月2日; 发布日期: 2025年9月10日

摘 要

基本文提出了一种基于物理信息神经网络的求解器, 用于求解纳维 - 斯托克斯方程组的正反问题。该方法结合深度神经网络与自动微分技术, 直接以空间和时间坐标作为输入, 预测流场的速度和压力变量。此求解器的核心在于构建一个包含物理方程残差(动量方程与连续性方程)、边界条件损失和初始条件损失的多目标损失函数, 并通过引入权重系数平衡各项贡献。网络采用双曲正切函数作为激活函数以保证高阶导数的计算需求, 并利用Adam和L-BFGS-B优化器进行训练。该求解器具有无需网格生成的优势。数值实验证明了求解器的有效性和鲁棒性。

关键词

物理信息神经网络, 纳维 - 斯托克斯方程, 损失函数, 边界条件

Physics-Informed Neural Networks for Solving Forward and Inverse Problems in Navier-Stokes Equations

Shuwei Li

School of Science, Lanzhou University of Technology, Lanzhou Gansu

Received: Aug. 9th, 2025; accepted: Sep. 2nd, 2025; published: Sep. 10th, 2025

Abstract

This paper proposes a Physics-Informed Neural Network (PINN)-based solver for solving forward and inverse problems in the Navier-Stokes equations. The method integrates deep neural networks

with automatic differentiation techniques, directly taking spatial and temporal coordinates as inputs to predict fluid flow velocity and pressure fields. The core methodology involves constructing a multi-objective loss function that incorporates residuals of the governing physics equations (momentum and continuity equations), boundary condition loss, and initial condition loss. Weight coefficients are introduced to balance the contributions of these components. The network utilizes the hyperbolic tangent (tanh) activation function to accommodate the computational requirements for high-order derivatives and is trained using the Adam and L-BFGS-B optimizers. This solver eliminates the need for mesh generation. Numerical experiments validate the effectiveness and robustness of the proposed solver.

Keywords

Physics-Informed Neural Networks, Navier-Stokes Equations, Loss Function, Boundary Conditions

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

由于机器学习方法的快速发展, 基于神经网络的方法正受到越来越多的关注。研究人员已经提出了许多基于神经网络的数值方法, 这成为了一种求解偏微分方程(PDE)的有前景的方法。

许多用于求解偏微分方程的机器学习方法都基于深度神经网络, 例如深度伽辽金方法(DGM) [1]、物理信息神经网络(PINN) [2]、深度 Ritz 方法(DRM) [3]和弱对抗网络(WAN) [4]。这些方法之间的主要区别在于损失函数的构建。例如, PINN 和 DGM 的损失函数是基于偏微分方程强形式残差的 L2 范数。DRM 的损失函数则是基于偏微分方程弱形式对应的能量泛函。WAN 方法通过最小化由偏微分方程弱形式诱导的算子范数来构建损失函数。

过去的五十多年以来, 计算流体动力学(CFD)的方法层出不穷。如果控制方程精确已知或所有尺度都能准确解析, 这些方法就能非常有效。然而, 在许多实际应用中, 要么物理过程尚不完全清楚(例如在反应输运中), 要么求解大时空尺度谱的成本过高, 因此不得不采用亚网格尺度闭包。在过去十年中, 人们已开展多项研究, 尝试将神经网络融入流体力学的求解中, 并采用不同的方法。在[5]中, PINN 方法被用于求解模拟高速气动流的一维和二维欧拉方程。在[6]中, 提出了一种针对非线性守恒律的离散子域上的保守 PINN。在[7]中, 提出了一种用于双曲系统的热力学一致性 PINN。PINN 还在[8]中用于求解超音速流动的逆问题。在[9]中, 引入了一种带有方程权重的 PINN 方法, 该方法引入了一个权重, 使得神经网络能够专注于训练更平滑的部分。

许多研究人员也关注将机器学习方法引入纳维-斯托克斯方程组(NS)的计算。文献[10]使用卷积神经网络(CNN)构建了圆柱尾流的预测模型; 文献[11]使用时序神经网络 (CNN) 建立了数据驱动模型, 用于预测圆柱尾流本征正交分解(POD)模式的系数。文献[12]基于少量速度测量数据, 双向循环神经网络(BIR)被用于预测圆柱尾流的 POD 系数, 获得了比扩展 POD 方法更精确的结果[13] [14]。此外, 深度学习技术也被应用于粒子图像测速(PIV), 用于分析湍流边界层的实验室数据[15]。文献[16] [17]对各种机器学习技术的引入在流体力学领域取得的进展进行了全面的总结。

在本文中, 我们结合自动微分技术, 开发了一种基于 PINN 的 Navier-Stokes 方程组求解器。该方法无需网格且易于实现, 对于多物理场和流体力学反问题依然有效, 尤其是在存在一些(多保真度)数据可以

弥补缺失物理场的情况下。我们使用开发的 PINN 求解器模拟了一系列流动问题。例如稳态科瓦斯奈流问题、非稳态圆柱绕流问题等。我们也为对传统 CFD 方法有挑战的问题进行了一系列测试，例如边界数据大噪声问题、边界不完整甚至缺失问题以及流体性质未知的反问题等。数值实验结果证明，我们开发的求解器不仅可以很好地模拟传统 CFD 方法可以计算的问题，对传统 CFD 方法有挑战的问题也可以保持一定精度。

本文其余部分安排如下：第 2 节介绍模型问题和 PINN 方法。第 3 节提出用于求解 Navier-Stokes 方程组的 PINN 求解器。第 4 节通过一系列数值测试来评估我们方法的准确性。最后，第 5 节总结了主要结论。

2. 准备

2.1. 纳维 - 斯托克斯(Navier-Stokes)方程

在本节我们介绍了非定常不可压缩三维 Navier-Stokes 方程组的速度 - 压力形式。

不可压缩 Navier-Stokes 方程组为：

$$\begin{aligned}\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} &= -\nabla p + \frac{1}{Re} \nabla^2 \mathbf{u} \quad \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0 \quad \text{in } \Omega, \\ \mathbf{u} &= \mathbf{u}_\Gamma \quad \text{on } \Gamma_D, \\ \frac{\partial \mathbf{u}}{\partial n} &= 0 \quad \text{on } \Gamma_N,\end{aligned}\tag{1}$$

其中， t 为无量纲时间， $\mathbf{u}(x, t) = [u, v, w]^T$ 为无量纲速度矢量， p 为无量纲压力， $Re = U_{ref} D_{ref} / \nu$ 是由特征长度 D_{ref} 、参考速度 U_{ref} 和运动粘度 ν 定义的雷诺数。求解方程(1)需要初始条件和边界条件。此处， Γ_D 和 Γ_N 分别表示 Dirichlet 边界和 Neumann 边界。需要注意的是，本文仅考虑由参考解提供的 Dirichlet 边界条件；其他边界条件可以在 PINN 求解器中轻松处理。

2.2. 物理信息神经网络

物理信息神经网络(PINNs)的核心思想是通过最小化偏微分方程(PDE)和边界条件的残差，训练一个深度神经网络 $u_\theta(x)$ 来逼近真实解 $u(x)$ 。给定训练数据 $\mathcal{T} = \mathcal{T}_D \cup \mathcal{T}_B$ ，其中包含两个集合：

域内采样点 $\mathcal{T}_D = \{x_j^d\}_{j=1}^{N_D} \subset \Omega$

边界采样点 $\mathcal{T}_B = \{x_j^b\}_{j=1}^{N_B} \subset \partial\Omega$

这些点分别表示计算域内和边界上的离散点。用于求解问题(1)的 PINN 损失函数具体形式如下：

$$\text{Loss}(\theta, \mathcal{T}) = \frac{1}{N_D} \sum_{j=1}^{N_D} \alpha \left| \mathcal{L}(u_\theta(x_j^d)) - f(x_j^d) \right|^2 + \frac{1}{N_B} \sum_{j=1}^{N_B} \beta \left| \mathcal{B}(u_\theta(x_j^b)) - h(x_j^b) \right|^2,\tag{2}$$

其中 $\alpha > 0$ 和 $\beta > 0$ 是不同配点处的惩罚权重，默认通常设为 1。损失函数中的导数可通过自动微分技术 [18] 计算。由于该损失函数关于参数高度非线性且非凸，通常采用基于梯度的优化器(如 Adam 和 LBFGS)进行最小化。

3. 方法论

在本章我们讨论如何利用物理信息神经网络求解纳维 - 斯托克斯方程组。我们通过深度神经网络来

近似 Navier-Stokes 方程组的解, 该网络以空间和时间坐标作为输入, 预测对应的速度和压力场, 即 $(t, x, y, z) \mapsto (u, v, w, p)$ 。

3.1. 网络架构

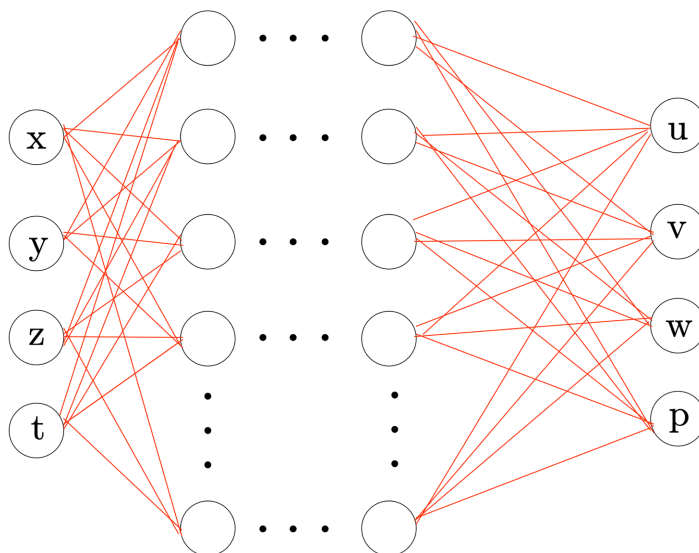


Figure 1. Neural network architecture diagram

图 1. 神经网络结构图

我们使用单个网络多组输出的方式实现对 Navier-Stokes 方程组解的近似。用于求解式(1)的物理信息神经网络结构如图 1 所示, 对应的神经网络表达式为:

$$\begin{cases} y_0 = x, \\ y_k = \sigma_k(W_k \cdot y_{k-1} + b_k), k = 1, 2, \dots, K, \\ (u, v, w, p)^T = W_{K+1} \cdot y_K + b_{K+1}, \end{cases} \quad (3)$$

其中, $W \in \mathbb{R}^{m \times n-1}$ 和 $b \in \mathbb{R}^m$ 分别为权重和偏置, $n_0 = 4$ 是输入的维度。 $x \in \mathbb{R}^{n_0}$ 是输入变量, $\sigma(\cdot)$ 是激活函数, 本文的非线性激活函数 σ 采用双曲正切函数 $\tanh$ 。 $\theta = (W_1, \dots, W_{K+1}, b_1, \dots, b_{K+1})$ 是神经网络中的参数集合, $(u, v, w, p)^T$ 表示输入 x, y, z, t 在参数 θ 下的输出。

在 PINN 求解器中, 偏微分方程(PDE)导数的数值逼近依赖于神经网络输出对其输入的导数。因此, 激活函数的可微性至关重要。然而, 常用激活函数(如 ReLU)无法提供高阶连续导数(特别是二阶连续可导性)。相比之下, 具有连续导数的 Sigmoid 型激活函数(如双曲正切 Tanh 和逻辑 Sigmoid)能够满足这一要求。其中, 双曲正切(Tanh)函数在输入接近 0 时表现出近似的恒等映射特性, 这通常使其具有优于逻辑 Sigmoid 函数的逼近性能[19]。基于以上原因, 本文选用双曲正切函数(Tanh)作为神经网络的激活函数。

式(1)中的残差包括动量方程的误差和无散度约束的误差。为了计算 Navier-Stokes 方程组的残差, 偏微分算子通过自动微分(AD)计算, 可直接在深度学习框架(如 TensorFlow 中的“tf.gradients()”)中实现。在自动微分中, 控制方程的导数由物理信息神经网络的输出对输入的导数近似表示。

3.2. 训练和采样

我们使用式(3)中的神经网络结构, 并定义以下的损失函数(对应式(1))以训练 PINN 求解器参数:

$$\begin{aligned}
L &= L_e + \alpha L_b + \beta L_i, \\
L_e &= \frac{1}{N_e} \sum_{i=1}^4 \sum_{n=1}^{N_e} |e_{vp1}^n|^2, \\
L_b &= \frac{1}{N_b} \sum_{n=1}^{N_b} |\mathbf{u}^n - \mathbf{u}_b^n|^2, \\
L_i &= \frac{1}{N_i} \sum_{n=1}^{N_i} |\mathbf{u}^n - \mathbf{u}_i^n|^2,
\end{aligned} \tag{4}$$

其中: L_e 为方程残差损失, L_b 为边界条件损失, L_i 为初始条件损失, α 和 β 为权重系数, 用于平衡不同损失项的贡献。 L_e 、 L_b 和 L_i 分别表示 Navier-Stokes 方程组残差、边界条件和初始条件对应的损失函数分量; N_b 、 N_i 和 N_e 表示不同损失项的训练数据数量; $\mathbf{u}_b^n = [u_b^n, v_b^n, w_b^n]^T$ 和 $\mathbf{u}_i^n = [u_i^n, v_i^n, w_i^n]^T$ 分别表示边界和第 n 个数据点在初始时刻的给定速度; e_{vp1}^n 表示第 i 个方程在第 n 个数据点处的残差。

通过最小化式(1)对应的损失函数(5), 我们优化式(3)中定义神经网络的权重参数。具体优化过程采用自适应学习率算法 Adam [20] 进行初期预热训练, 随后切换至高精度二阶拟牛顿优化器 L-BFGS-B。该组合策略权衡两者优势: Adam 对初始参数选择不敏感、鲁棒性强但收敛速度较慢; L-BFGS-B 收敛速度快、精度高, 但依赖较好的初始点。神经网络参数利用 Xavier 初始化方案[21]随机初始化, 训练终止条件为损失函数降至预设阈值或达最大迭代次数。

4. 数值实验

在 4.1 节和 4.2 节中, 我们测试了我们方法在传统 CFD 计算比较良好的问题上的有效性, 包括稳态科瓦斯奈流和非稳态二维圆柱绕流问题。之后我们展示了 PINN 求解器的最佳应用场景, 即传统 CFD 方法可能存在局限或计算代价高昂的问题。在 4.3 节中, 我们考虑了边界条件缺失或含有噪声的问题。在 4.4 节中, 我们探讨了流体属性参数未知的反问题, 该类问题通常导致传统方法面临极高的计算成本。

4.1. 稳态科瓦斯奈(Kovaszny)流

我们采用 Kovaszny 流作为首个测试算例来验证 PINN 方法的性能。该二维稳态 Navier-Stokes 流的解析解如下:

$$\begin{aligned}
u(x, y) &= 1 - e^{\zeta(x-0.5)} \cos(2\pi(y+0.5)), \\
v(x, y) &= \frac{\zeta}{2\pi} e^{\zeta(x-0.5)} \sin(2\pi(y+0.5)), \\
p(x, y) &= \frac{1}{2} (1 - e^{2\zeta(x+1)}),
\end{aligned} \tag{5}$$

其中

$$\zeta = \frac{1}{2\nu} - \sqrt{\frac{1}{4\nu^2} + 4\pi^2}, \quad \nu = \frac{1}{Re} = \frac{1}{40}.$$

计算域设定为 $[0, 0.5] \times [0, 1]$ 。每条边界选取 128 个采样点, 在域内随机选取 2048 个点。我们采用 Adam 优化器训练 20000 次, 随后使用 L-BFGS-B 训练至停止(基于增量容差自动终止)。我们采用的 PINN 求解器包含 5、7、10 个隐藏层, 每个隐藏层包含 128 个神经元。计算误差见表 1。这验证了我们的方法可以有效地模拟稳态科瓦斯奈流。

Table 1. Kovasznay flow: Relative L_2 errors of velocity and pressure for varying numbers of hidden layers

表 1. 科瓦兹纳伊流：不同隐藏层数量的速度和压力的相对 L_2 误差

隐藏层数量	$\ e_u\ _{L^2}$	$\ e_v\ _{L^2}$	$\ e_p\ _{L^2}$
5	7.21e-4	6.24e-3	2.63e-3
7	2.82e-4	4.16e-3	2.15e-3
10	2.77e-4	2.16e-3	1.51e-3

4.2. 非稳态二维圆柱绕流

本节我们使用 PINN 求解器模拟雷诺数 $Re=128$ 下圆柱体后方的二维涡脱落。圆柱体置于 $(x,y)=(1,1)$ 处，直径 $D=1$ 。来自[2]的高保真 DNS 数据被用作参考，并为 PINN 求解器的训练提供边界和初值条件数据。我们考虑的求解域定义为 $[2,9]\times[0,4]$ ，时间区间为 $0,[]7$ (超过一个脱落周期)，时间步长 $\Delta t=0.1$ 。对于训练数据，每个边采样 1280 个点，并在域内部采样 256,000 个时空散点计算残差。PINN 求解器包含 5、7、10 个隐藏层，每层有 128 个神经元。训练过程优化器设置同 4.1 节。表 2 展示了计算的误差，表明 PINN 求解器的计算结果与 DNS 结果在定性上相符。这些实验验证了我们方法在非稳态二维圆柱绕流问题上的准确性。

Table 2. Non-steady cylindrical flow: Relative L_2 errors of velocity and pressure for different numbers of hidden layers

表 2. 非稳态圆柱绕流：不同隐藏层数量的速度和压力的相对 L_2 误差

隐藏层数量	$\ e_u\ _{L^2}$	$\ e_v\ _{L^2}$	$\ e_p\ _{L^2}$
5	2.27e-2	3.24e-1	1.12e-1
7	1.91e-2	1.16e-1	9.27e-2
10	3.26e-2	2.51e-1	1.01e-1

4.3. 不适定/扰动边界条件下的科瓦斯奈流动

我们以 Kovasznay 流动为例，并使用与 4.1 节中相同的计算域。对于该流动问题，我们采用求解了三个边界条件缺失的算例(算例 2、3、4)以及一个入口速度含有 10%噪声的算例 5，这些算例对于传统 CFD 求解器是不适定问题。

计算结果总结于表 3 中。残差计算点的数量为 1024。神经网络结构为 5×64 。其中，算例 1 为具有完整边界条件的情况，算例 2 为缺少上下边界条件的情况，算例 3 为缺少出口边界条件的情况，算例 4 模拟了数据同化场景，仅保留入口边界条件($x=0$ 处)，并在计算域中垂线($x=0.25$)上均匀布置 21 个速度测量点 (u,v) ，这些测量点覆盖 $y \in [0,1]$ 区间且间距为 0.05。算例 5 为在入口边界条件上增加了随机噪声的情况。噪声水平设定为 10%。PINN 求解器采用与第 4.1 节相同的网络和优化方法进行训练。算例 1、2、3 的结果显示，PINN 求解器在求解边界条件不完整的纳维 - 斯托克斯方程组时，尽管相比于完整的边界条件的情况下精度有所下降，但是依旧能够对纳维 - 斯托克斯方程组进行有效求解。对于与标准的数据同化问题类似的算例 4，PINN 求解器同样获得了良好的精度。特别注意在算例 5 中，在处理含噪声数据时未应用 L-BFGS-B 优化器。结果表明，PINN 求解器在含噪声边界条件下均能取得相对较好的精度。

此外，我们还求解了算例 4。该算例仅包含入口边界条件及中间垂直线上的观测数据，这与标准的数据同化问题类似，使用 PINN 求解器同样获得了良好精度。此外，我们还验证了 PINN 求解器处理含噪声边界条件的能力。具体而言，在表 3 的算例 5 中，在最重要的边界(入口速度)上添加了随机噪声。噪声水

平设定为 10%，表示噪声幅度与原始数据标准差之比。结果表明，PINN 求解器在含噪声边界条件下能取得相对较好的精度。

Table 3. Kovasznay flow: relative L_2 errors of velocity and pressure under ill-posed/perturbed boundary conditions

表 3. 科瓦兹纳伊流：不适定/扰动边界条件下的速度和压力的相对 L_2 误差

算例	$\ e_u\ _{L^2}$	$\ e_v\ _{L^2}$	$\ e_p\ _{L^2}$
算例 1	6.2e-4	2.44e-3	1.52e-3
算例 2	1.56e-3	9.66e-3	5.03e-2
算例 3	4.96e-3	7.15e-3	2.05e-2
算例 4	6.45e-2	5.25e-2	8.89e-2
算例 5	1.69e-1	1.78e-1	4.52e-1

4.4. 不适定数据下的反演：雷诺数和流场

接下来我们考虑另一个问题：仅给定一些散乱的流速数据，待求解的量是雷诺数及整个流场——若使用传统计算流体动力学(CFD)求解器，该过程可能计算量极大。神经网络规模为 5×128 ，残差点数量为 2048 个，边界点数量为 4×128 个。除 PINN 的权重与偏置项外，雷诺数也被设置为可学习变量，并采用 4.1 节相同的优化方法训练物理信息神经网络。实验结果表明当数据点足够多时，PINN 求解器可以准确反演出未知的雷诺数，结果见表 4。具体而言，雷诺数的相对误差量级为 10^{-2} 。

Table 4. Errors of velocity, pressure, and inversed Reynolds number in flow fields at different Reynolds numbers

表 4. 不同雷诺数下流场的速度和压力误差以及反演雷诺数的误差

Re	$\ e_u\ _{L^2}$	$\ e_v\ _{L^2}$	$\ e_p\ _{L^2}$
1	1.21e-2	2.44e-2	1.52e-2
10	2.56e-2	9.66e-3	2.03e-2
50	9.66e-3	3.15e-2	2.55e-2
100	1.45e-2	1.25e-2	8.89e-3
200	3.69e-2	3.48e-2	1.52e-2

5. 结论

本文基于物理信息神经网络开发了一种纳维-斯托克斯方程组求解器。该求解器将方程组的初值信息和边界条件纳入损失函数，通过优化损失函数使神经网络近似纳维-斯托克斯方程组的解。此外，对于未知雷诺数的问题，我们的求解器也能成功反演出未知的雷诺数。数值实验证明了该方法不仅对传统 CFD 计算表现良好的问题有效，对计算成本过高或可能失效的传统 CFD 难题也依然有效。

参考文献

- [1] Sirignano, J. and Spiliopoulos, K. (2018) DGM: A Deep Learning Algorithm for Solving Partial Differential Equations. *Journal of Computational Physics*, **375**, 1339-1364. <https://doi.org/10.1016/j.jcp.2018.08.029>
- [2] Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2019) Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations. *Journal of Computational Physics*, **378**, 686-707. <https://doi.org/10.1016/j.jcp.2018.10.045>
- [3] E, W. and Yu, B. (2018) The Deep Ritz Method: A Deep Learning-Based Numerical Algorithm for Solving Variational

- Problems. *Communications in Mathematics and Statistics*, **6**, 1-12. <https://doi.org/10.1007/s40304-018-0127-z>
- [4] Zang, Y., Bao, G., Ye, X. and Zhou, H. (2020) Weak Adversarial Networks for High-Dimensional Partial Differential Equations. *Journal of Computational Physics*, **411**, Article ID: 109409. <https://doi.org/10.1016/j.jcp.2020.109409>
 - [5] Mao, Z., Jagtap, A.D. and Karniadakis, G.E. (2020) Physics-Informed Neural Networks for High-Speed Flows. *Computer Methods in Applied Mechanics and Engineering*, **360**, Article ID: 112789. <https://doi.org/10.1016/j.cma.2019.112789>
 - [6] Jagtap, A.D., Kharazmi, E. and Karniadakis, G.E. (2020) Conservative Physics-Informed Neural Networks on Discrete Domains for Conservation Laws: Applications to Forward and Inverse Problems. *Computer Methods in Applied Mechanics and Engineering*, **365**, Article ID: 113028. <https://doi.org/10.1016/j.cma.2020.113028>
 - [7] Patel, R.G., Manickam, I., Trask, N.A., Wood, M.A., Lee, M., Tomas, I., *et al.* (2022) Thermodynamically Consistent Physics-Informed Neural Networks for Hyperbolic Systems. *Journal of Computational Physics*, **449**, Article ID: 110754. <https://doi.org/10.1016/j.jcp.2021.110754>
 - [8] Jagtap, A.D., Mao, Z., Adams, N. and Karniadakis, G.E. (2022) Physics-Informed Neural Networks for Inverse Problems in Supersonic Flows. *Journal of Computational Physics*, **466**, Article ID: 111402. <https://doi.org/10.1016/j.jcp.2022.111402>
 - [9] Liu, L., Liu, S., Xie, H., Xiong, F., Yu, T., Xiao, M., *et al.* (2023) Discontinuity Computing Using Physics-Informed Neural Networks. *Journal of Scientific Computing*, **98**, Article No. 22. <https://doi.org/10.1007/s10915-023-02412-1>
 - [10] Jin, X., Cheng, P., Chen, W. and Li, H. (2018) Prediction Model of Velocity Field around Circular Cylinder over Various Reynolds Numbers by Fusion Convolutional Neural Networks Based on Pressure on the Cylinder. *Physics of Fluids*, **30**, Article ID: 047105. <https://doi.org/10.1063/1.5024595>
 - [11] Wu, P., Sun, J., Chang, X., Zhang, W., Arcucci, R., Guo, Y., *et al.* (2020) Data-Driven Reduced Order Model with Temporal Convolutional Neural Network. *Computer Methods in Applied Mechanics and Engineering*, **360**, Article ID: 112766. <https://doi.org/10.1016/j.cma.2019.112766>
 - [12] Jin, X., Laima, S., Chen, W. and Li, H. (2020) Time-Resolved Reconstruction of Flow Field around a Circular Cylinder by Recurrent Neural Networks Based on Non-Time-Resolved Particle Image Velocimetry Measurements. *Experiments in Fluids*, **61**, Article No. 114. <https://doi.org/10.1007/s00348-020-2928-6>
 - [13] Hosseini, Z., Martinuzzi, R.J. and Noack, B.R. (2015) Sensor-Based Estimation of the Velocity in the Wake of a Low-Aspect-Ratio Pyramid. *Experiments in Fluids*, **56**, Article No. 13. <https://doi.org/10.1007/s00348-014-1880-8>
 - [14] Discetti, S., Raiola, M. and Ianiro, A. (2018) Estimation of Time-Resolved Turbulent Fields through Correlation of Non-Time-Resolved Field Measurements and Time-Resolved Point Measurements. *Experimental Thermal and Fluid Science*, **93**, 119-130. <https://doi.org/10.1016/j.expthermflusci.2017.12.011>
 - [15] Cai, S., Zhou, S., Xu, C. and Gao, Q. (2019) Dense Motion Estimation of Particle Images via a Convolutional Neural Network. *Experiments in Fluids*, **60**, Article No. 73. <https://doi.org/10.1007/s00348-019-2717-2>
 - [16] Duraisamy, K., Iaccarino, G. and Xiao, H. (2019) Turbulence Modeling in the Age of Data. *Annual Review of Fluid Mechanics*, **51**, 357-377. <https://doi.org/10.1146/annurev-fluid-010518-040547>
 - [17] Brunton, S.L., Noack, B.R. and Koumoutsakos, P. (2020) Machine Learning for Fluid Mechanics. *Annual Review of Fluid Mechanics*, **52**, 477-508. <https://doi.org/10.1146/annurev-fluid-010719-060214>
 - [18] Baydin, A.G., Pearlmutter, B.A., Radul, A.A., *et al.* (2018) Automatic Differentiation in Machine Learning: A Survey. *Journal of Machine Learning Research*, **18**, 1-43.
 - [19] Goodfellow, I., Bengio, Y. and Courville, A. (2016) Deep Learning. MIT Press.
 - [20] Kingma, D.P. and Ba, J. (2014) Adam: A Method for Stochastic Optimization.
 - [21] Glorot, X. and Bengio, Y. (2010) Understanding the Difficulty of Training Deep Feedforward Neural Networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, Sardinia, 13-15 May 2010, 249-256.