arXiv:2304.12299v1 [physics.flu-dyn] 5 Mar 2023

利用深度神经算子开发用于地下水流建模的经济高效仿真器

Maria Luisa Taccari[a,*], He Wang[b], Somdatta Goswami[c], Jonathan Nuttall[d], Xiaohui Chen[a], Peter K. Jimack[b]

[a]School of Civil Engineering, University of Leeds, Leeds, UK
[b]School of Computing, University of Leeds, Leeds, UK
[c]Division of Applied Mathematics, Brown University, Providence, U.S.A[d]Deltares, Delft, The Netherlands

摘要

当前地下水模型在实施过程中面临着巨大的计算负担。为此，我们提出了一种经济高效的仿真器，可高效、准确地预测含水层开采的影响。我们的方法采用深度神经算子（DeepONet），通过深度神经网络学习在无限维函数空间之间进行映射的算子。我们的目标是推断存在抽水井时承压含水层中的水头分布。我们在四个问题上成功测试了 DeepONet，包括两个正问题、一个反演分析问题和一个非线性系统问题。此外，我们提出了一种基于 DeepONet 架构的新型扩展，使其能够针对训练期间未见过的不同水力传导系数场和抽水井位置生成准确的预测值。仿真器的预测值与目标数据高度吻合，表现出色，表明所提出的模型可作为一种高效、快速的工具，支持各类需要反复进行地下水流问题正向数值模拟或反演模拟的任务。总之，我们的工作为开发经济高效且准确的地下水模型提供了一条可行途径。

关键词：深度神经算子，地下水流动，代理建模，深度学习

1. 引言

现有地下水数值模型的计算效率，在处理大规模或高度非线性系统时会成为问题，尤其是在决策依赖实时模拟推断的情况下。许多地下水问题具有非线性特征，因此必须采用迭代过程求解方程；而重新校准初始模型以纳入新的观测数据时，计算成本可能会迅速增加 [1]。因此，地下水模型沉重的计算负担可能会限制其在决策过程中的应用，例如用于管理荷兰干旱风险的国家水文模型（NWM） [2]。

近年来，深度学习技术在函数逼近中的应用备受关注，因为它们有望开发出高效的低保真模型，用于在广泛的应用领域中近似计算成本高昂的数值方法 [3, 4, 5, 6, 7]。这类近似方法已取得成功的一个领域是

*通讯作者。电子邮箱：cnmlt@leeds.ac.uk

将深度卷积神经网络（卷积神经网络）用作动态多相流问题的代理模型 [8, 9, 10, 11]。基于深度卷积神经网络的代理模型将问题视为图像到图像回归，其中输入函数和输出函数均表示为图像，深度卷积神经网络则学习两者之间的映射。由此得到的模型能够在任意时间点，准确预测具有高度非均质含水层渗透系数场的压力场和饱和度场。然而，将深度卷积神经网络用作代理模型面临若干挑战。它仅适用于输入函数和输出函数定义在格点网格上的问题，且训练数据涵盖计算域内的所有网格值。对于位于训练区域内的任意查询点，都无法评估其解。模型的精度及其架构均取决于网格分辨率，这意味着要保持精度，就必须针对不同的网格分辨率重新训练模型 [12]。此外，对于每种不同的区域几何形状、输入参数集或初始/边界条件（初始/边界条件），都需要单独进行模拟。

为了推广解，我们需要研究更高层次的开采，以学习从输入函数空间到输出函数空间的映射（而不是像函数回归中那样映射到向量空间）。为此，算子通用逼近定理 [13] 表明，深度神经网络有望用于从数据中学习非线性算子。神经算子于2019年以深度算子网络（DeepONet）的形式被提出[14], ，能够学习两个无限维巴拿赫空间之间的映射，为多维复杂动力学的实时预测提供了独特的模拟框架。训练完成后，DeepONet 具有离散化不变性，这意味着同一组网络参数可在底层函数数据的不同参数化之间共享，因此可用于获得任意空间和时间位置处的解（插值）。此外，近期的一项理论研究 [15] 表明，DeepONet能够打破输入空间中的维数灾难。

本文通过多个问题设置证明，DeepONet 能够对显式算子和隐式算子进行快速且准确的推断，因此可作为高效的代理模型，用于近似地下水流区域中的各种关注量。有关该方法的详细信息以及相关工作的概述，读者可参阅第 3 节。具体而言，我们采用 DeepONet 框架设计了一种高效的仿真模型，用于估计开采对非均质承压含水层中水头分布的影响。我们首次证明，DeepONet 能够有效应用于地下水问题，并阐述了这种学习方法在地下水流区域中的一些潜在优势。所提出的方法能够高效地

学习正问题和逆问题的解，后者采用传统方法求解时通常十分困难且耗时。我们成功地利用 DeepONet 对非线性系统进行快速推断，而采用标准数值求解器则需要使用迭代方法。最后，我们对原始 DeepONet 进行了修改，从而能够在井随机位于非均质含水层中的情况下，成功预测空间变化的地下水水头分布。

本文组织如下。第 2 节讨论地下水流动的问题陈述，以及本研究分析的四个实验： i) 从水力传导系数场到地下水场的映射， ii) 从抽水井位置和水力传导系数场到地下水场的映射， iii) 学习从水头到水力传导系数场的逆映射，以及最后的 iv) 带有水头相关边界条件的非线性问题。第 3 节总结 DeepONet 方法，并概述模型设置以及代理模型的训练。第 4 节将神经算子应用于上述四种实验设置，而第 5 节总结研究结果并展望未来研究方向。

2. 问题陈述

本研究聚焦于利用神经算子学习由地下水流动控制方程的解所表示的非线性算子。本节介绍控制地下水流动的偏微分方程（PDE），以及典型数值求解器所面临的计算限制。随后，我们讨论四个实验，这些实验旨在展示所提出方法的适用范围。

2.1. 控制偏微分方程

定义地下水在二维空间中运动的控制偏微分方程结合了达西定律和质量守恒原理 [16], ，其表达式为:

$$S_s \frac{\partial h}{\partial t} - \nabla \cdot (K \nabla h) = q_s, \tag{1}$$

其受到特定边界条件的约束。在方程 1 中， h 是水头 [L]， K 是空间变化的水力传导系数场 [L/T]， q_s 是单位体积地下水源汇体积通量 [1/T]， S_s 是比储水量 [1/L], ，而 t [T] 是时间。附录 A 中提供了描述所有符号的表格。

美国地质调查局（USGS）的有限差分流动模型 MODFLOW [16] 在过去 30 多年中被研究人员、顾问和政府广泛用于高效模拟地下水流动[17]。作为英格兰和威尔士的国家环境监管机构，环境署使用 MODFLOW 分析各种情景对主要含水层水文和地质水文行为的影响 [1]。该模型使环境署能够研究含水层系统如何响应取水率变化、补给率变化以及新补给源或排泄源的引入。这些信息有助于环境署在管理国家水资源和保护环境方面作出明智决策。对于空间网格为 200m、时间分辨率为每月一至三个时间步、模拟时间跨度从数月至数年的地下水模型，其运行所需的计算时间可能高得难以承受，尤其是在满足特定约束、旨在最大化地下水开采量的优化问题中。此外，求解用于推断含水层材料性质的逆问题，需要进行多次模拟，以发现缺失的物理规律或校准所构建逆问题中的自由参数。这种计算负担推动了基于深度神经网络的仿真器的发展，使其能够在运行时显著降低计算成本的同时提供高精度预测值。在本研究中，我们采用了基于深度神经算子的代理模型，该模型可作为对多个输入函数的控制方程进行数值近似的可行替代方案，从而高效预测含水层中开采的影响。

2.2. 算子学习任务

算子记为 $\mathcal{G}$，是一种数学函数，它接受一个或多个函数作为输入，并生成另一个函数作为输出。给定输入函数 $\boldsymbol{u}(\boldsymbol{x}) \in \mathbb{R}^{d_x}$ 和输出函数 $\boldsymbol{v}(\boldsymbol{x}) \in \mathbb{R}^{d_y}$，算子 $\mathcal{G}$ 定义为 $\mathcal{G}: \ \boldsymbol{u}(\boldsymbol{x}) \in \mathbb{R}^{d_x} \mapsto \boldsymbol{v}(\boldsymbol{x}) \in \mathbb{R}^{d_y}$，其中 $\mathbb{R}^{d_x}$ 和 $\mathbb{R}^{d_y}$ 分别表示输入和输出的维度，而 $\boldsymbol{x}$ 表示定义输出空间的时空坐标。偏微分方程（PDE）可视为一个算子：输入空间由确定问题定义所需的函数组成，例如初始条件和边界条件（ICs/BCs）、强迫函数以及系数（这些量可能随空间和时间变化）。输出空间是偏微分方程的解所属的 Sobolev空间。我们的目标是使用神经算子 $\mathcal{G}_{\boldsymbol{\theta}}$近似第 2.1 节中介绍的偏微分方程，其中 $\boldsymbol{\theta}$ 统称神经算子的参数，包括权重 $\mathbf{W}$ 和偏置 $\mathbf{b}$。DeepONet 的数学表述将在第 3.1 小节中介绍。我们通过下一节介绍的四个计算实验，展示使用神经算子近似地下水流的有效性。

2.3. 计算实验

本研究旨在开发一种地下水流快速仿真器。我们希望证明，所提出的框架可用于高效估计高度非均质承压含水层中由抽水引起的地下水位变化，即相对于抽水前水位的变化（实验 E_1 和E_2）。此外，我们还将该框架用于求解逆问题（实验 E_3）；若采用传统数值求解器，则需要进行大量正向数值模拟。最后，我们求解一个非线性系统（实验 E_4）；该问题通常通过迭代过程获得解，因此耗时较长。本节介绍了针对第 2.1 小节所述地下水流设计的四个计算实验（E_1 - E_4）。图 1 展示了本研究所考虑的不同实验的可视化描述。附录 B 介绍了用于考虑含水层非均质性的数据生成细节。

实验 E# 的描述如下。

• **E1：正问题，** $\mathcal{G}_{\boldsymbol{\theta}}$ ： $K(\boldsymbol{x}) \mapsto h(\boldsymbol{x})$：本实验的目标是学习解算子 $\mathcal{G}_{\boldsymbol{\theta}}$ ，该算子将空间变化的水力传导系数场 $K(\boldsymbol{x})$ 映射为后续某一时间步的水头 $h(\boldsymbol{x})$。换言之，学习目标是在给定恒定抽水速率 q_s 、于 $t = 0$ 开始抽水的一个完全贯穿井条件下，推断含有一口完全贯穿井的非均质承压含水层中的水头分布。给定空间变化的水力传导系数场 $K(\boldsymbol{x})$，并假设区域边界上未规定流量或水头，解是时间点 $t = T$ 的水头分布预测值 $h(\boldsymbol{x})$。

• **E2：多输入函数，** $\mathcal{G}_{\boldsymbol{\theta}}$ ： $[K(\boldsymbol{x}) \; x_P] \mapsto h(\boldsymbol{x})$：本实验的目标是学习一个解算子，以给定空间变化的水力传导系数 $K(\boldsymbol{x})$ 和抽水井位置 x_P 作为输入函数，近似时间 T 的水头。

• **E3：逆问题，** $\mathcal{G}_{\boldsymbol{\theta}}$ ： $[h(\boldsymbol{x}, \; t) \; \mathcal{K}_0(\boldsymbol{x})] \mapsto \mathcal{K}(\boldsymbol{x})$：本实验旨在学习一个逆算子，在给定区域不同时间点的水头后，近似空间变化的水力传导系数场 $K(\boldsymbol{x})$。我们认识到，该问题不存在非平凡解，因此承认，对解场进行更多观测会增加找到模型能够收敛到的唯一解的可能性。为约束逆向建模过程中的解空间，我们将水力传导系数的稀疏观测 $K_0(\boldsymbol{x})$作为额外输入。

• **E4：非线性系统，** $\mathcal{G}_\theta$ **：** $K(\boldsymbol{x}) \mapsto h(\boldsymbol{x})$**：** 本实验的场景是：一口抽水井位于区域中心，另一口水头相关井固定在区域内的不同位置。抽水井具有指定流量边界即，其流量不是水头的函数；而水头相关井的指定流量则根据水头计算得出。学习算子 $\mathcal{G}^{NL}$ 的目标是直接近似非均质含水层的水力传导系数$K(\boldsymbol{x})$与水头分布 $h(\boldsymbol{x})$之间的映射。在传统求解器中，通过迭代循环处理非线性：使用上一次迭代得到的水头反复构建并求解控制方程，直到控制方程的残差处于指定容差范围内。所提出的基于神经算子的解法消除了对

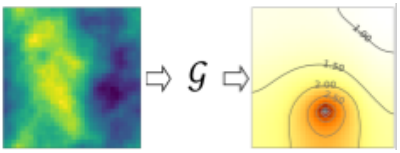
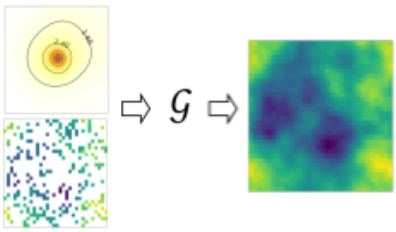
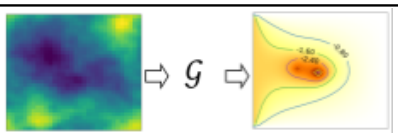
实验	输入函数 $u(x)$	输出函数 $v(x)$	算子 $\mathcal{G}$ 的可视化
E1	$K(\boldsymbol{x})$	$h(\boldsymbol{x})$	
E2	$K(\boldsymbol{x}), x_P$	$h(\boldsymbol{x})$	
E3	$h(\boldsymbol{x}, t)$, $K_0(\boldsymbol{x})$	$K(\boldsymbol{x})$	
E4	$K(\boldsymbol{x})$	$h(\boldsymbol{x})$	

图 1：本文所考虑实验的示意图。图中展示了输入/输出函数以及用于说明算子所学习任务的代表性图形。

3. 解算子近似方法

本节将介绍深度算子网络 DeepONet 的架构，并讨论近期利用神经算子求解偏微分方程的一些研究。考虑两个可分

巴拿赫空间， $\boldsymbol{u} = \boldsymbol{u}(\Omega; \mathbb{R}^{d_x})$ 和 $\boldsymbol{v} = \boldsymbol{v}(\Omega; \mathbb{R}^{d_y})$，其中 Ω 是 $\mathbb{R}^D$ 中的一个有界开集， $\mathbb{R}^{d_x}$ 和 $\mathbb{R}^{d_y}$分别表示输入和输出的维数。由某个时间 T 处随时间变化的偏微分方程（方程 1）的解得到的非线性映射 $\mathcal{G}$，将 $\boldsymbol{u}$ 映射到 $\boldsymbol{v}$。目标是通过参数化映射近似非线性算子 $\mathcal{G}$，即 $\mathcal{G}: \boldsymbol{u} \times \boldsymbol{\Theta} \to \boldsymbol{v}$ 或 $\mathcal{G}_{\boldsymbol{\theta}}: \boldsymbol{u} \to \boldsymbol{v}$，其中 $\mathcal{G}_{\boldsymbol{\theta}}$ 表示参数化映射， $\boldsymbol{\Theta}$ 是一个有限维参数空间。通过在离散化域上生成的数据集 $\{\mathbf{u}^j, \mathbf{v}_j\}_{j=1}^N$上使用反向传播训练神经算子，从而得到最优参数 $\boldsymbol{\theta}^*$ 。

3.1. 深度算子网络

Chen and Chen 提出的算子通用逼近定理 [13] 指出，具有足够宽度的浅层神经网络能够以任意精度逼近任何非线性连续泛函或算子。该定理基于一种特定的神经网络模型，该模型由两个并行的子网络组成，并通过点积将两个网络的输出结合起来。受通用逼近定理的启发，提出了深度算子网络（DeepONet） [14], 用于学习无限维巴拿赫空间之间的映射。DeepONet 架构由两个深度神经网络（深度神经网络）组成：分支网络在固定传感器点 $\{x_1, x_2, \ldots, x_m\}$处对输入函数 $\boldsymbol{u}$进行编码，而主干网络则对与时空坐标 $\zeta = \{x_i, y_i, t_i\}$相关的信息进行编码，在这些坐标处对解算子进行评估，以计算损失函数。该学习过程在一般设置下进行，这意味着对输入函数 $\boldsymbol{u}$进行评估的传感器位置（$x_i{}_{i=1}^m$）不必均匀分布，但在所有输入函数评估中必须保持一致。分支网络将 $[\boldsymbol{u}(x_1)\ \boldsymbol{u}(x_2) \ldots, \boldsymbol{u}(x_m)]^T$作为输入，并输出 $[b_1, b_2, \ldots, b_q]^T \in \mathbb{R}^q$；而主干网络将 ζ作为输入，并生成 $[t_1, t_2, \ldots, t_q]^T \in \mathbb{R}^q$作为输出。两个子网络的输出通过点积进行组合，从而生成所需结果。在最后阶段加入偏置（$b_0 \in \mathbb{R}$）以增强表达能力，得到 $\mathcal{G}(\boldsymbol{u})(\zeta) \approx \sum_{i=k}^q b_k t_k + b_0$。通过最小化均方误差损失函数，可以获得可训练参数 $\boldsymbol{\theta}$ 的优化值。图2展示了 [14]提出的原始 DeepONet 架构。

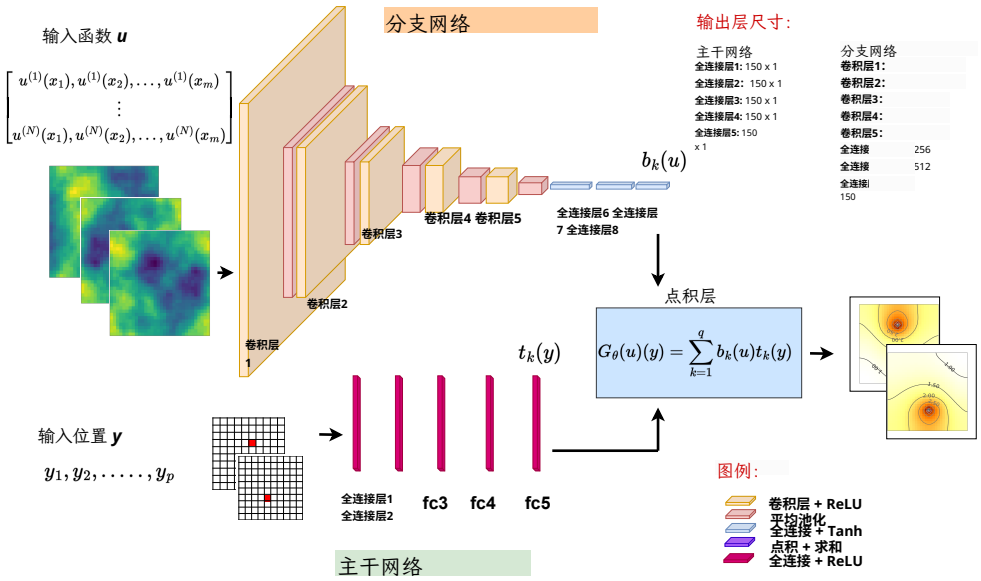


图2：本文所采用的原始 DeepONet 网络架构示意图。本文采用卷积神经网络作为分支网络，并采用全连接前馈神经网络作为主干网络。分支网络和主干网络的输出通过内积进行组合，以近似解算子。

3.2. 相关工作

DeepONet 已在多个应用领域展现出显著成效，例如逼近不规则海浪 [18], 、学习刚性化学动力学 [19], 、气泡动力学 [20], 以及微观结构演化 [21]等。在这些应用中，网络通过使用大型数据集进行训练，以解决正向问题。此外，近期一些研究聚焦于学习从多输入函数到解场的映射[22, 23]。DeepONet 在地下水流问题领域的早期研究，主要通过数据驱动[24] 和物理信息方法 [25, 26, 27]，学习从渗透系数场到简单及复杂几何结构中水头的映射。在这两项研究中，控制地下水流的偏微分方程（达西方程）被用作应用案例，以展示相应研究中提出的框架。在另一项研究中，提出了算子级迁移学习框架 [28] ，并以达西方程为例展示该方法。算子迁移学习的思想是：使用源域中的充足标注数据，在标准回归损失下训练源模型，然后将学习到的变量迁移到第二个目标模型；该目标模型使用来自目标域（不同但相关领域）的极少量标注数据进行训练，其损失函数为混合损失函数，即回归损失与条件嵌入算子差异损失之和。此外， [29], 还提出了另一种算子级迁移学习框架，其中达西方程在一个 L形

源区域并转移到带孔的 L形区域。混合求解器（HINTS）方法的实现可以直接处理目标几何形状的变化，并且不需要重新训练算子。上述讨论的工作均未处理方程 1 中额外的比储水量项，也没有专门致力于在地下水流的多种情景中应用该算子。此外，算子网络首次被设计用于求解**E3** 中的逆问题，以根据水头 $h(\boldsymbol{x})$ 以及 $K(\boldsymbol{x})$ 的稀疏观测，学习水力传导系数场 $K(\boldsymbol{x})$。

在一项独立研究中，Lanthaler 等人对 DeepONet [15] 的逼近误差和泛化误差进行了理论分析。他们将误差分解为编码误差、逼近误差和重构误差，并对总误差的上下界进行了理论推导。他们的分析表明，当不连续性或尖锐特征在空间中的位置不固定时，DeepONet 的精度可能会下降；而对于生成空间位置固定的不连续性或尖锐特征的算子，DeepONet 则能够准确地进行学习。这与我们的观察一致，因此我们提出了一种改进架构来处理此类情景。根据 Hadorn [30], 的研究，如果不增加主干网络中的基函数数量，DeepONet 很难学习每个位置处的尖锐特征。不幸的是，对于高维问题，增加基函数数量是不可行的。为克服 DeepONet 在处理平移不变性方面的这一瓶颈，一种有效的改进方法是消除这种不变性。Shift-DeepONet [30] 和 FlexDeepONet [31] 都添加了预变换子网络，用于对数据进行平移、旋转和缩放。输入函数首先经过这些附加网络，再传递给分支网络；这些网络能够学习对输入函数进行重新缩放和重新居中的方式。变换层将学习到的平移、旋转和缩放参数与评估点的空间坐标相结合：该层的输出作为主干网络的输入，使得主干网络的基函数依赖于输入函数。类似地，DeepONet 的另一种扩展在分支网络和主干网络的输入端分别引入了一个编码器 [32]。嵌入特征通过逐点乘法插入两个子网络的隐藏层中。与传统的 DeepONet 架构相比，这种新颖架构似乎对梯度消失问题更具鲁棒性。

在当前工作中，我们考虑了 [14] 中首次介绍的原始 DeepONet，其优点是架构更为简单。在本文的后续部分，我们提出了 DeepONet 的一种改进版本，以克服原始 DeepONet 在处理源项时存在的局限性：源项并不总是在相同位置定义，并且会导致解场中出现剧烈梯度。下一节将介绍所采用网络架构的详细信息。

3.3. 网络架构与训练

分支网络采用卷积神经网络（CNN），其输入为函数 $\boldsymbol{u}$，这些函数在大小为 32×32 的格点网格上进行评估，并且该网格在本研究开展的所有实验中保持一致。对于实验 **E1** 和**E4**，CNN 具有一个输入通道；然而，对于实验 **E2** 和**E3**，CNN 具有两个输入通道，其中第二个通道分别表示井的位置和目标水力传导系数的稀疏观测。主干网络的输入是 128 个评估点的坐标，这些评估点在区域内随机采样，并且对于每个训练样本都各不相同。所有实验的网络架构详细信息见表1。网络的示意图如图2所示。

表1：所有实验（**E1-E4**）中采用的原始 DeepONet 架构详情。

	层	**卷积核大小**	**宽度**	**激活函数**	**输出**
			分支网络		
1	二维卷积	5×5	16	ReLU	$32 \times 32 \times 16$
2	平均池化	2×2			$16 \times 16 \times 16$
3	二维卷积	5×5	8	ReLU	$16 \times 16 \times 16$
4	平均池化	2×2			$8 \times 8 \times 16$
5	二维卷积	5×5	4	ReLU	$8 \times 8 \times 16$
6	平均池化	2×2			$4 \times 4 \times 16$
7	二维卷积	5×5	4	ReLU	$4 \times 4 \times 16$
8	平均池化	2×2			$2 \times 2 \times 16$
9	二维卷积	5×5	4	ReLU	$2 \times 2 \times 64$
10	平均池化	2×2			重塑为 1×64
11	全连接		1024	Tanh	1×256
12	全连接		1024	Tanh	1×512
13	全连接		1024		1×150
			主干网络		
14	全连接		150	ReLU	150×1
15	全连接		150	ReLU	150×1
16	全连接		150	ReLU	150×1
17	全连接		150	ReLU	150×1
18	全连接		150	ReLU	150×1

在原始 DeepONet 架构中，解算子被近似为分支网络和主干网络输出乘积的求和。然而，对于实验 $E2$，我们注意到，让主干网络获知抽水井的位置（输入函数）是实现良好学习效果的关键。因此，我们提出了一种新颖的 DeepONet 架构。如图 3 所示，分支网络池化层的每个输出都与主干网络每一层的输出相结合。来自分支网络的张量经过展平后，连接一个采用 $Sigmoid$ 激活函数的全连接神经网络层。鉴于所得向量（其权重可解释为系数）与主干网络相应隐藏层的维度相同，因此可以通过内积将这两个向量合并。

该结果通过主干网络的后续层传播，并在重塑后与池化层的输出拼接，再传递至分支网络。我们证明，该架构能够准确预测实验 **E2** 中水头的高梯度；对于该实验，原始 DeepONet 给出的预测更加平滑。

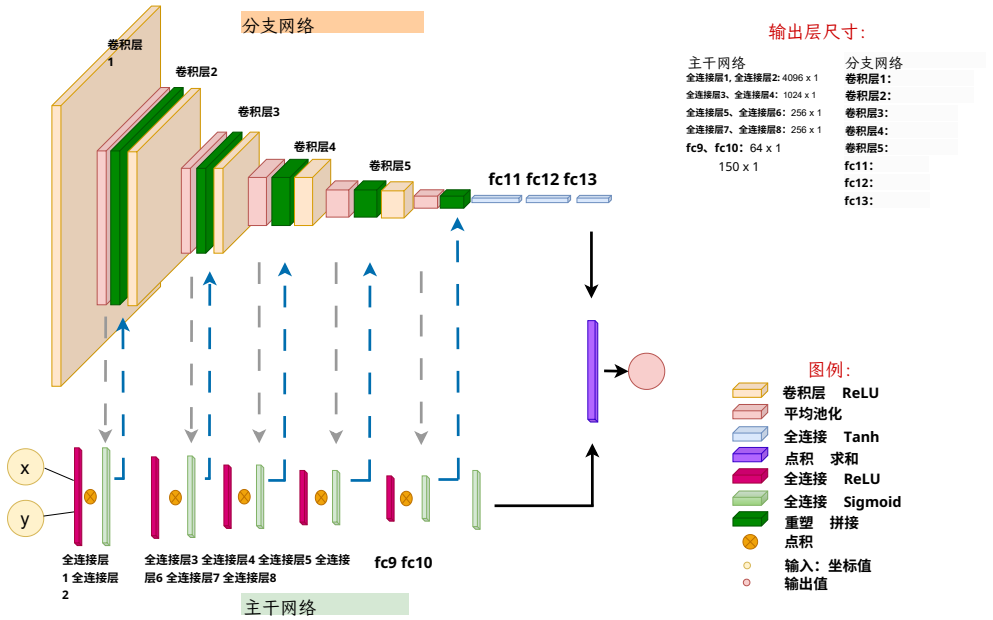


图 3：针对实验 **E2**提出的新型 DeepONet 架构示意图。该架构专门设计用于考虑水泵位置的变化。使用主干网络近似的基函数可根据井的位置进行调整。

该实现使用 JAX 框架，在单个 NVIDIA GeForce RTX 3080 上完成。对于所有测试案例，数据集由 $N_{train} = 1000$ 训练数据和 $N_{test} = 200$ 测试数据组成。网络使用 Adam 优化器 [33] 进行训练，初始学习率为 5×10^{-4}，该学习率每 1000 次迭代按 0.9 的速率指数衰减，批量大小为 100，最多进行 10^5 次迭代。我们每 100 次迭代监控一次损失，如果测试数据的损失值在 2×10^4 次迭代后没有下降，则触发提前停止。

4. 结果

在本节中，我们通过前面讨论的实验表明，DeepONet 可用于准确且高效地逼近一系列地下水流动模拟问题。我们还对两种架构进行了比较研究：原始 DeepONet 和本文提出的新型 DeepONet

架构。所有模型均在区域上定义的少量稀疏点上进行训练，但会在整个区域上针对测试数据进行评估。平均相对误差（均方误差）被用作误差指标，该指标对应于训练期间使用的损失函数，其计算方式是所有模型和所有实验中目标场与预测场之间的平方差。在附录 C 中，我们针对实验 **E3** 和 **E4**，将 DeepONet 的准确性与另外两种广泛使用的深度神经网络架构进行了比较研究。

4.1. $E1$: 固定井位置的正问题

在本实验中，训练数据集和测试数据集中的井的位置被视为相同，算子学习以下映射： $\mathcal{G}_{\boldsymbol{\theta}}^{F1}$: $K(\boldsymbol{x}) \mapsto h(\boldsymbol{x})$。网络训练总共耗时283秒，完成24，600次迭代；误差指标分别在 $MSE_{train} = 1.8 \times 10^{-5}$ 个样本和 $MSE_{test} = 2.5 \times 10^{-4}$ 个样本上计算。图4（顶行）展示了给定非均质水力传导率场时，水头预测值与目标值之间的典型对比。输入和输出均沿每个独立通道进行归一化，即将变量重新缩放到 $[0, 1]$范围内。如预测图所示，DeepONet能够非常准确地预测压力升高以及井周围水头的剧烈增加。我们通过改变神经元数量、层数、卷积核大小、批大小和训练数据等因素进行了多项分析。我们观察到误差率仅发生轻微变化，只有在数据未归一化或学习率较高时才会出现显著恶化。此外，增加查询点数量或在井周围进行更频繁的采样，并未提高预测准确性。

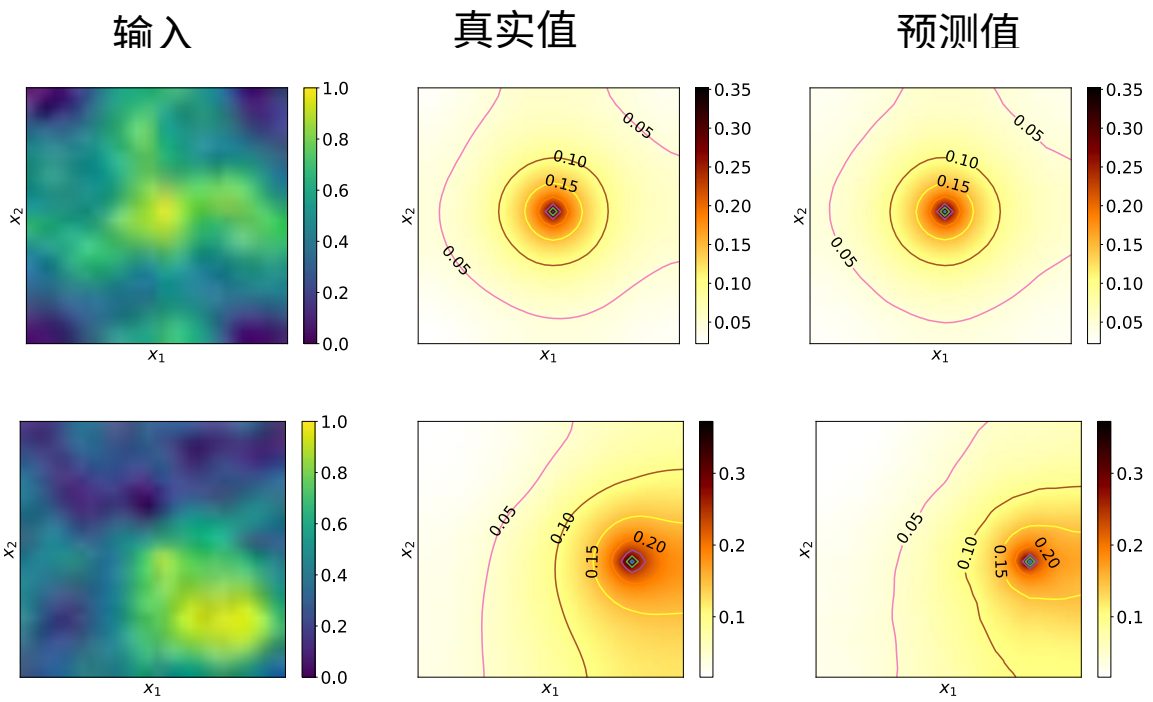


图4：上排：对于具有未见过的非均质水力传导率场的测试样本，使用训练好的原始 DeepONet（实验 $E1$）得到的水头 $h(\boldsymbol{x})$ 预测值。井的位置固定在 $(x,\ y) = (16,\ 16)$，训练数据集和测试数据集中的位置相同。下排：使用原始 DeepONet 得到的水头预测值，其中测试样本包含未见过的非均质水力传导率场以及未见过的抽水井位置（实验 $E2$）。图中所示结果对应于这样的情况：分支网络的输入为水力传导系数场（第一列），主干网络的输入为网络用于求解的坐标以及井的位置坐标。第二列和第三列分别表示 DeepONet 对水头的真实值和预测值。

4.2. $E2$：井位置变化时的正问题

在下一个实验中，我们进一步扩展了网络能力，使其能够学习抽水井位于未见过位置时的解。根据DeepONet的原始形式，分支网络对输入函数进行编码，这些输入函数包括水力传导系数场和源项的位置（井）。我们观察到，当井被移动到多个位置时，原始架构无法捕捉抽水井附近区域中的剧烈梯度。读者可以参考图5的最后一列，其中展示了三个具有代表性的测试样本的原始 DeepONet预测结果的可视化；井的位置（强迫项）被编码为二值图，并作为附加通道与水力传导系数场拼接，输入分支网络现有的卷积神经网络。通过一组

在计算实验中，我们探索了将强迫项编码为分支网络输入的不同可行方式。这些方式包括：将源项的位置表示为笛卡尔坐标系中的坐标集合；将区域中每个点与井的位置之间的距离（大小和角度）作为输入；或者使用以井的位置为中心的高斯函数。在此背景下，我们通过采用单个网络、针对每个输入函数分别使用两个独立的神经网络，以及使用两个相互连接的并行网络，对网络架构进行了修改。所有这些实验要么导致预测结果非常不准确，无法正确确定压力前缘的范围和位置，要么导致源项附近的预测结果更加平滑，正如图5最后一列中此前所强调的那样。附录 D 通过奇异值分解（SVD）分析了原始 DeepONet 在这些测试案例中预测准确率较低的原因。

最后，我们发现，对于这一类问题，要实现良好的学习效果，必须向主干网络提供源项的位置。图 4（底行）展示了一个具有代表性的测试案例中的网络预测结果：主干网络的输入是网络评估解时所用的坐标与抽水井坐标的拼接。分支网络的输入则是整个区域内评估得到的水力传导系数场。通过目视检查可见，对于不同的 $K(\boldsymbol{x})$ 分布以及抽水井位置，预测值都与参考解非常吻合。在训练数据集和测试数据集上计算得到的误差指标分别为2.4×10^{-5} 和 2.7×10^{-4}。然而，在更复杂的情形下，例如区域内有多口井、各井的抽水速率不同，或存在位置并非局限于单个点的特征（河流和排水沟），这种方法的效率会降低。与实际应用类似，建模者通常对地下水系统有预先规划，因此我们决定将源项的位置编码为二值图，并将其与水力传导系数场拼接，作为分支网络的输入。实验表明，当训练数据中的强迫项位置有所变化时，向主干网络提供强迫项的位置是实现良好学习效果的关键。因此，我们采用 DeepONet 的全新架构（图 3），将分支网络和主干网络连接起来，具体架构见 3.3 节。如图 5 所示，对于三个具有代表性的测试样本，连接两个子网络隐藏层的架构明显优于原始 DeepONet。训练耗时 324 秒，训练数据集和测试数据集上的误差指标分别为 6.6×10^{-5} 和 2.6×10^{-4} 。值得注意的是，在正问题的各种情形下，训练误差与测试误差的数量级始终相同，而且视觉结果也令人非常满意。因此可以合理地认为，对于给定的正问题，所得误差已达到 DeepONet 的下限

问题。

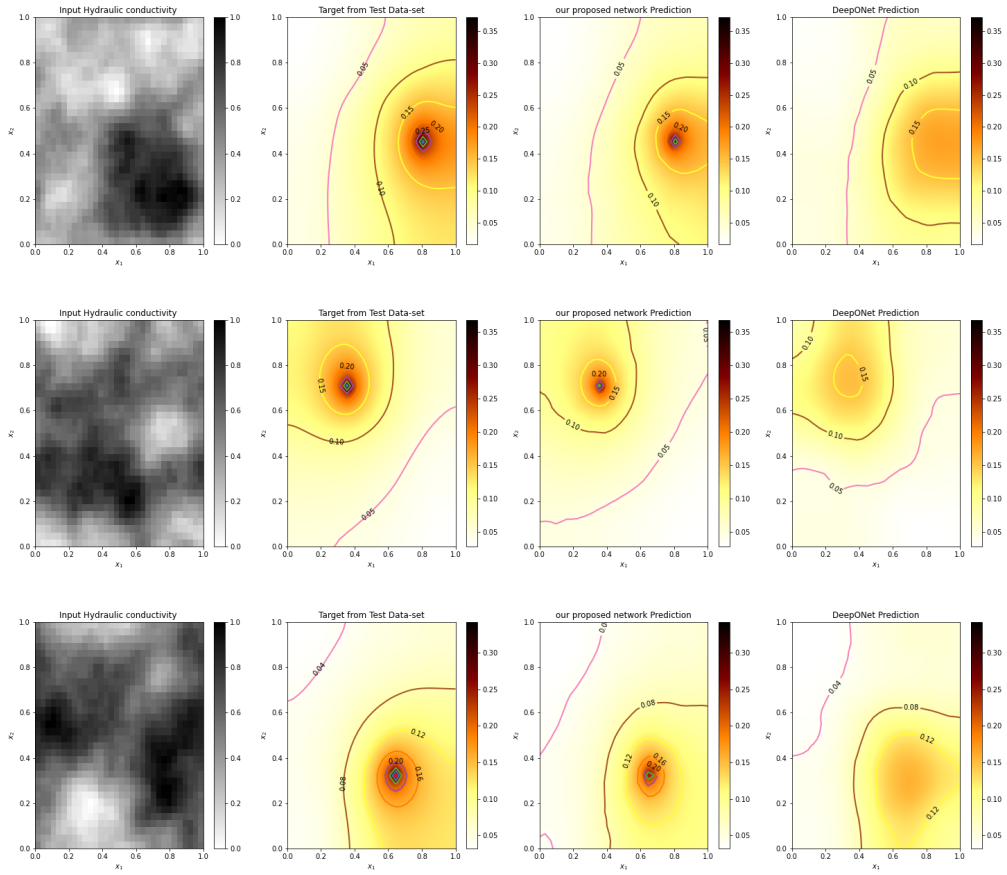


图5：针对未见过的非均质水力传导率场（第一列）和未见过的源项位置（井）所预测的水头。对于三个具有代表性的测试样本，将原始 DeepONet（最后一列）和所提出的网络架构（第三列）的预测值与目标场（第二列）进行了比较。分支网络的输入为水力传导率场以及指示井位置的二值图。主干网络的输入为网络评估解所在位置的坐标。

4.3. $E3$：逆问题

本节探讨算子网络求解逆问题的效率。更具体地说，我们旨在辅助地下属性表征。由于无法直接观测整个地下系统，反演分析的目标是利用渗透系数场的稀疏观测以及一些已知的水头信息，了解非均质含水层的属性（即水力传导系数场）。我们考虑一个测试问题，其中既有整个区域内的水头信息，也有水力传导系数的稀疏观测。在图 6 中，我们展示了一个具有代表性的测试案例及其输入和预测值，预测结果来自

算子网络。当使用水力传导系数场中20%的值（随机采样）作为观测点时，在测试样本上计算得到的误差指标为 1.21×10^{-1}。除了比较目标参考场（第一列）与模拟反演结果（第四列）之外，我们还比较了输入水头（第二列）以及使用传统求解器计算的、与预测传导系数相对应的水头（第三列）。算子网络能够给出可信的预测，在整个测试数据集上都表现出准确且稳定的性能。观测数据为网络提供了信息，与未提供水力传导系数测量值的情况相比，其准确率提高了16%。通过提供不同时间步的水头观测信息，还可以进一步提升性能。当分支网络额外对三个时间间隔相等的时刻的水头进行编码时（而不是仅编码单个时间步），预测性能进一步提升，平均准确率提高了4%。未来，可以进一步发展这一方法，用于指导高效的观测活动，在尽量减少观测需求的同时，提高反演模型的准确性。

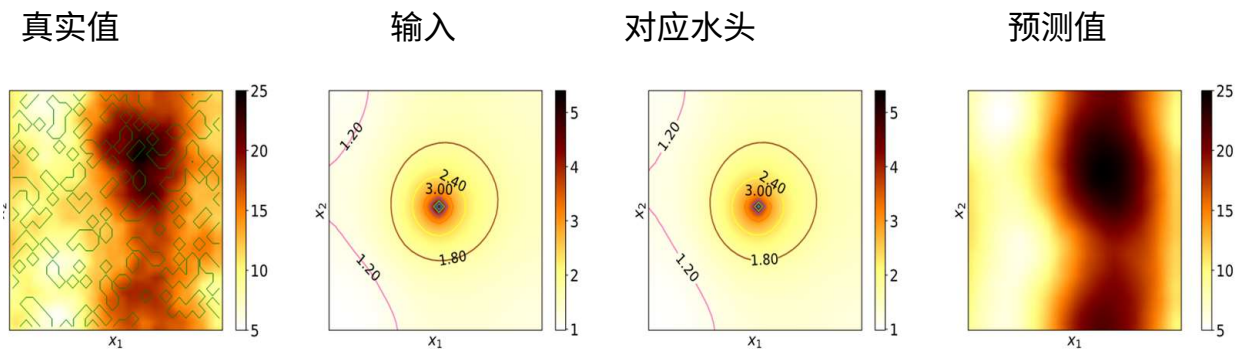


图6：使用原始 DeepONet 进行反演分析的代表性文本样本。第一列显示未知的水力传导系数场及其上方的观测点位置，这些信息与整个区域中的水头（第二列）一起被编码为额外输入。最后一列显示模拟反演结果，第三列则显示使用传统求解器计算的、与预测传导系数相对应的水头。

4.4. $E4$：非线性情况

许多地下水过程呈现非线性行为，因此需要使用迭代求解器来获得水头的解。在本研究中，我们表明，所提出的数据驱动方法能够直接求解该系统，并大幅降低传统迭代求解器所需的计算成本。传统的地下水流动方程有限差分形式可写为：$\boldsymbol{Ah} = \boldsymbol{b}$，其中 $\boldsymbol{h}$ 是时间步结束时的水头值向量，$\boldsymbol{A}$ 是由

水头系数组成的矩阵， $\boldsymbol{b}$是常数项向量 [16]。在非线性系统中， $\boldsymbol{A}$ 矩阵中的各个元素都是水头的函数，因此需要通过非线性外部迭代循环求解该方程组。在我们的示例中，一口抽水井位于区域中心，另一处位置设有一口水头相关井（非线性的完整细节见附录 B）。对于训练过程中未见过的给定水力传导系数场（作为分支网络的输入），神经算子预测的水头在整个区域内与目标值完全吻合（图4.4）。训练过程耗时 430 秒，共进行了 44100 次迭代；在训练数据集和测试数据集上计算的误差指标分别为 3.1×10^{-5} 和 3.9×10^{-5}。

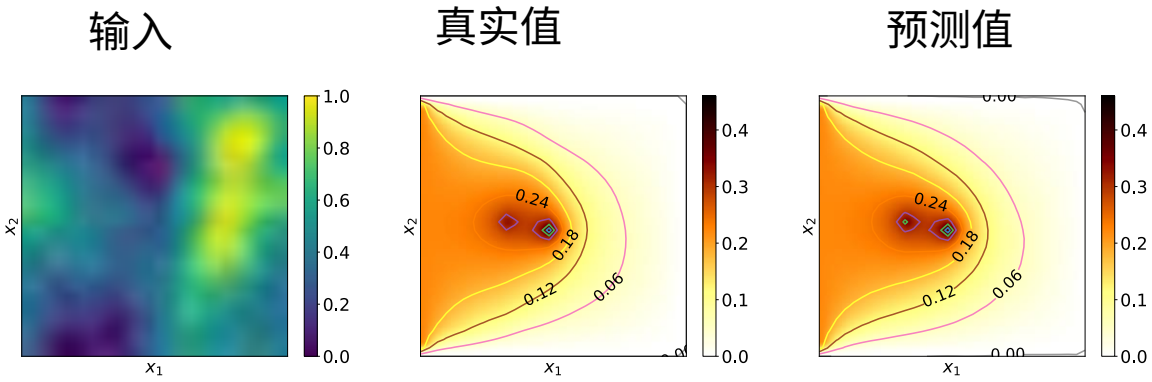


图 7：对于非线性问题中未见过的非均质渗透系数场，DeepONet 预测的水头与真实值的比较，**E4**。

5. 总结与讨论

本文提出了 DeepONet 框架，将其作为代理模型，以高效且准确地计算地下水系统的状态响应。该模型在四项实验中接受训练和测试，结果表明，它能够预测固定抽水井条件下非均质含水层中的水头，推广到未见过的抽水井位置，表征含水层属性（反演分析），并处理非线性系统。所提出的模型能够非常准确地学习空间变化的水力传导系数场与水头场之间的正向和逆向关系。然而，当抽水井位于区域内任意位置时，需要对 DeepONet 的原始形式进行修改。为此，本文提出了一项创新方法，将分支网络的输入连接到主干网络

，使该网络能够准确预测未见过的井位置和水力传导系数场对应的解。通过在多个示例中成功实现神经算子，我们展示了该网络支持一系列任务的能力，这些任务需要对地下水模型进行重复的正向数值模拟。

未来，此类模型将得到扩展，以适用于更复杂、更贴近实际的地下问题。这些问题可能涉及更复杂的预测，涵盖更广泛的开采率、含水层系统几何结构及其属性和边界，以及与其他地表水开采和排放之间的相互作用。

作者贡献

Maria Luisa Taccari：概念构思、方法、软件、验证、正式分析、数据整理、撰写初稿、审阅与编辑、可视化。**He Wang**：方法、正式分析、审阅与编辑、监督。**Somdatta Goswami**：方法、审阅与编辑、监督、可视化。**Jonathan Nuttall**：概念构思、方法、软件。**ChenXiaohui**：监督、经费获取。**Peter K. Jimack**：方法、正式分析、审阅与编辑、监督。

代码和数据可用性

用于生成数据集和训练模型的所有代码将发布于 https://github. com/mlttac/DeepOnet_gwf。

致谢

本研究得到了利兹-约克-赫尔自然环境研究委员会（NERC）博士培训合作项目（DTP）Panorama的资助，资助编号为NE/S007458/1。我们谨此感谢Deltares提供的支持，并对Bennie Minnema为设计非线性测试案例所作出的宝贵贡献表示诚挚谢意。

参考文献

[1] R. Farrell、M. Ververs、P. Davison、P. Howlett、M. Whiteman，将补给模型与地下水流模型嵌入环境署国家地下水建模系统，《伦敦地质学会特别出版物》408（1）（2017）55–69。

[2] M. Mens、B. Minnema、K. Overmars、B. Van den Hurk，开发长期干旱风险管理模型时面临的两难：《荷兰国家水文模型》案例，《Environmental Modelling & Software》143（2021）105100。

[3] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang，物理信息机器学习，Nature Reviews Physics 3 (6) (2021) 422–440。

[4]E. Samaniego, C. Anitescu, S. Goswami, V. M. Nguyen-Thanh, H. Guo, K. Hamdia, X. Zhuang, T. Rabczuk，通过机器学习在计算力学中采用能量方法求解偏微分方程：概念、实现与应用，Computer Methods in Applied Mechanics and Engineering 362 (2020) 112790。

[5] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, S. Hoyer，机器学习加速的计算流体力学，美国国家科学院院刊 118 (21) (2021) e2101784118。

[6] Y. Bar-Sinai, S.Hoyer, J. Hickey, M. P. Brenner，偏微分方程数据驱动离散化的学习，《美国国家科学院院刊》116 (31) (2019年) 15344–15349。

[7]J. Jeon, S. J. Kim，降低计算流体力学模拟计算成本的有限体积法网络，arXiv 预印本 arXiv:2105.03332 (2021)。

[8] M.L.Taccari, J.Nuttall, X.Chen, H.Wang, B.Minnema, P. K.Jimack，作为地下水预测代理模型的注意力 U-Net，《Advances in Water Resources》163 (2022) 104169。

[9] S. Mo, Y. Zhu, N. Zabaras, X. Shi, J. Wu，用于非均质介质中动态多相流不确定性量化的深度卷积编码器-解码器网络，《Water Resources Research》55 (1) (2019年) 703–728。

[10] S. Mo、N.Zabaras、X. Shi、J. Wu，利用深度自回归神经网络解决地下水污染源识别中的高维逆问题，《Water Resources Research》55 (5)（2019年）3856–3881。

[11] Z. Zhong, A. Y. Sun, H. Jeong，利用条件深度卷积生成对抗网络预测非均质地层中的二氧化碳羽流迁移，《Water Resources Research》55 (7) (2019年) 5830–5851。

[12] Y. Zhu, N. Zabaras，贝叶斯深度卷积编码器–解码器网络用于代理建模和不确定性量化，《计算物理学杂志》366 (2018) 415–447。[13] T. Chen, H. Chen，利用具有任意激活函数的神经网络逼近非线性算子及其在动力系统中的应用，《IEEE 神经网络汇刊》6 4 (1995) 911–7。[14] L. Lu, P. Jin, G. Pang, Z. Zhang, G. E. Karniadakis，基于算子的通用逼近定理，通过 DeepONet 学习非线性算子，《Nature Machine Intelligence》3 (3) (2021年3月。doi：https://doi.org/10.1038/s42256-021-00302-5)。[15] S. Lanthaler, S. Mishra, G. E. Karniadakis，DeepONet 的误差估计：无限维深度学习框架，《数学及其应用汇刊》6 (1) (2022) tnac001。[16] C. P. Kumar，使用 MODFLOW 对地下水流进行数值建模，《印度科学杂志》2 (2013) 86–92。

[17] J. D. Hughes, M. J.Russcher, C. D. Langevin, E. D. Morway, R. R. McDonald，《用于模拟控制和软件互操作性的 MODFLOW 应用程序编程接口》，Environmental Modelling & Software 148 (2022) 105257。

[18] Q. Cao, S. Goswami, G. E. Karniadakis, S. Chakraborty，《深度神经算子可预测不规则波作用下浮式海上结构物的实时响应》，arXiv 预印本 arXiv:2302.06667 (2023)。

[19] S. Goswami, A. D. Jagtap, H. Babaee, B. T. Susi, G. E. Karniadakis，《使用扩展深度神经算子学习刚性化学动力学》，arXiv 预印本 arXiv:2302.12645 (2023)。

[20] C. Lin, Z. Li, L. Lu, S. Cai, M. Maxey, G. E. Karniadakis，《用于预测多尺度气泡生长动力学的算子学习》，The Journal of Chemical Physics 154 (10) (2021) 104118。

[21] V. Oommen、K. Shukla、S. Goswami、R. Dingreville、G. E. Karniadakis，使用神经算子和自编码器架构学习两相微观结构演化，arXiv预印本 arXiv:2204.07230（2022）。[22] P. Jin、S. Meng、L. Lu，MIONet：通过张量积学习多输入算子，《工业与应用数学学会科学计算杂志》44（6）（2022）A3490–A3514。

[23] S. Goswami、D. S. Li、B. V. Rego、M. Latorre、J. D. Humphrey、G. E. Karniadakis，促成主动脉瘤的异质机械生物学损伤的神经算子学习，《英国皇家学会界面杂志》19（193）（2022）20220410。

[24] L. Lu、X. Meng、S. Cai、Z. Mao、S. Goswami、Z. Zhang、G. E. Karniadakis，基于FAIR数据对两个神经算子进行全面且公平的比较（含实用扩展），Computer Methods in Applied Mechanics and Engineering 393（2022）114778。

[25] S. Goswami、M. Yin、Y. Yu、G. Karniadakis，利用物理信息变分DeepONet预测脆性材料中的裂纹路径，arXiv预印本 arXiv:2108.06905（2021）。

[26] S. Goswami、M. Yin、Y. Yu、G. E. Karniadakis，利用物理信息变分DeepONet预测准脆性材料中的裂纹路径，Computer Methods in Applied Mechanics and Engineering 391（2022）114587。

[27] S. Wang、H. Wang、P. Perdikaris，利用物理信息DeepONet学习参数偏微分方程的解算子，Science Advances 7（40）（2021）eabi8605。

[28] S. Goswami、K. Kontolati、M. D. Shields、G. E. Karniadakis，《条件偏移下用于偏微分方程的深度算子迁移学习》，Nature Machine Intelligence（2022）1–10。

[29] A. Kahana、E. Zhang、S. Goswami、G. Karniadakis、R. Ranade、J. Pathak，《微分方程混合迭代数值求解器的几何可迁移性》，Computational Mechanics（2023）1–14。

[30] P. S. Hadorn，《Shift-DeepONet：扩展深度算子网络以处理不连续输出函数》，硕士论文，苏黎世联邦理工学院，苏黎世（2022年3月16日）。doi:10.3929/ethz-b-000539793。[31] S. Venturi，T. Casey，《用于增强DeepONet灵活性和可解释性的SVD视角》，arXiv预印本 arXiv:2204.12670（2022）。[32] S. Wang，M. A. Bhouri，P. Perdikaris，《通过自监督算子学习实现快速偏微分方程约束优化》，arXiv预印本 arXiv:2110.13297（2021）。[33] D. P. Kingma，J. Ba，《Adam：一种随机优化方法》（2014）。doi:10.48550/ARXIV.1412. 6980。URL https://arxiv.org/abs/1412.6980

[34]地下水流动有限差分建模课程，https://github.com/Olsthoorn/FDModelBuildingCourseGroundwaterFlow/，访问日期：2022年6月22日（2017年）。

[35] J. van Engelen、J. Delsman、H. Bootsma，使用 iMOD Python 软件包开展可复现的大规模地下水建模项目，技术报告，Copernicus Meetings（2022）。

[36] D. Eigen、C. Puhrsch、R. Fergus，使用多尺度深度网络从单幅图像预测深度图，Advances in Neural Information Processing Systems 27（2014）。

[37] Z. Li、N. Kovachki、K. Azizzadenesheli、B. Liu、K. Bhattacharya、A. Stuart、A. Anandkumar，适用于参数偏微分方程的傅里叶神经算子，arXiv 预印本 arXiv:2010.08895（2020）。

[38] T. Peters，《数据驱动的科学与工程：机器学习、动力系统与控制》，Contemporary Physics 60 (4) (2019年) 320–320。arXiv:https://doi.org/10.1080/00107514.2019. 1665103, doi:10.1080/00107514.2019.1665103。URL https://doi.org/10.1080/00107514.2019.1665103

附录 A。符号表

符号表见表 A.2。

符号	含义
h	水头
K	水力传导系数
q_s	单位体积地下水源汇体积通量
S_s	比储水量
t	Time
x_P	抽水井位置
$\mathcal{G}$	非线性算子
$\mathcal{G}_{\boldsymbol{\theta}}$	网络算子逼近器
$\boldsymbol{\theta}$	可学习参数
u	输入函数
u	输入函数
$\boldsymbol{y_j}$	评估点
m	传感器点数量
P	评估点数量
N	输入函数数量
N_T	训练样本对数量
q	两个子网络最后隐藏层的可调权重数量
$\boldsymbol{b}$	由分支网络最后一层定义的输出向量
$\boldsymbol{t}$	由主干网络最后一层定义的输出向量

表 A.2：符号表

附录 B. 生成数据集的数值模拟设置

用于构建数据驱动均方误差（MSE）损失的带标签高保真数据集，是通过采用有限差分格式求解控制方程进行模拟生成的。本节介绍了相关参数和设置。这些数据集均在 Python 生态系统中生成：前三个测试案例使用根据 [34] 改编的脚本，非线性案例则使用 iMOD Python 软件包 [35]。

正问题 该问题包含一个代表承压含水层的单层模型。该层厚 50 m，模型范围 R 为 8×10^3 m。网格定义为 32×32。每个网格单元的边长为 250 m，这对应于实际应用中典型的空间网格 [1]。该含水层的比弹性储水量 S_s 为 2×10^{-4} 1/m，具有高度非均质性，其水平渗透系数为$k\in[5,25]$ m/d。我们使用功率谱为 $P(l)=l^q$ 、 $q=-4$的高斯随机场生成渗透系数分布。随后将得到的连续浮点值重新缩放至范围$k\in[k_{min},k_{max}]$。 K、 k_{min} 和 k_{max}的最小值与最大值在每个样本中各不相同：它们是随机选取的整数，使得 $k_{min}\in[5,10]$ m/d 和 $k_{max}\in[20,25]$ m/d。一口位于（x， y）=（16，16）的抽水井从砂质含水层中抽水。开采量设定为恒定的 $Q=5000$ m^3/d，对应于典型的开采量 [1]。由此生成目标水头，即模拟时间为$T=200$ 天时对应时间步的解。

多输入函数 在第一个测试案例中，源项固定在同一位置，算子学习从水力传导系数场 K 到地下水场 h的映射。在第二个测试案例中，抽水井随机位于区域内，算子学习从井位置 x_P 和 K 场到输出函数的映射。所有其他设置均与上文所述相同。

逆问题 该问题的学习目标是：根据由水开采产生的水头分布，推断整个区域内的水力传导系数。逆问题所采用的数值设置与正问题相同，唯一的区别是抽水井现在位于域中心。正问题的输入和目标数据分别成为逆问题的目标和输入。除根据时间 T的模拟计算得到的水头外，我们还提供了目标水力传导系数的稀疏观测，作为额外输入。

非线性系统 该问题由一个表示承压含水层的单层模型组成。该层厚度为 60 m，地表高程为 0 m，模型范围 R 为 3.2×10^4m。区域离散为 32×32，每个网格单元边长为 1000 m。含水层的比储水量为 $S_s=1.6\times10^{-4}$ [1/m]，其水力传导系数沿区域变化，并且对于每个数据样本均有所不同。非均质含水层的 K场为具有功率谱 $P(l)=l^q$ 的高斯随机场，其中 $q=-4$，其数值在 $k\in[k_{min},k_{max}]$ 范围内重新缩放，其中 $k_{min}=5\times10^-4m/d$ 且

$k_{max} = 5 \times 10^{-}3m/d$。图 B.8 展示了模型的设置。初始水头处处对应于地表高程。我们施加狄利克雷边界条件：沿边界 x=0 的定水头单元等于 -2 m，区域其他三条边的定水头单元等于地表高程。一口抽水井位于域中心，即(x，y) =(16，16)。该抽水井具有指定流量边界：流量随时间保持恒定，等于 0。5 m^2/d。一口水头相关井位于 (x，y) =(10，16)，其流量根据水头计算。该井的流量与水头之间的关系见图 B.8——右图。

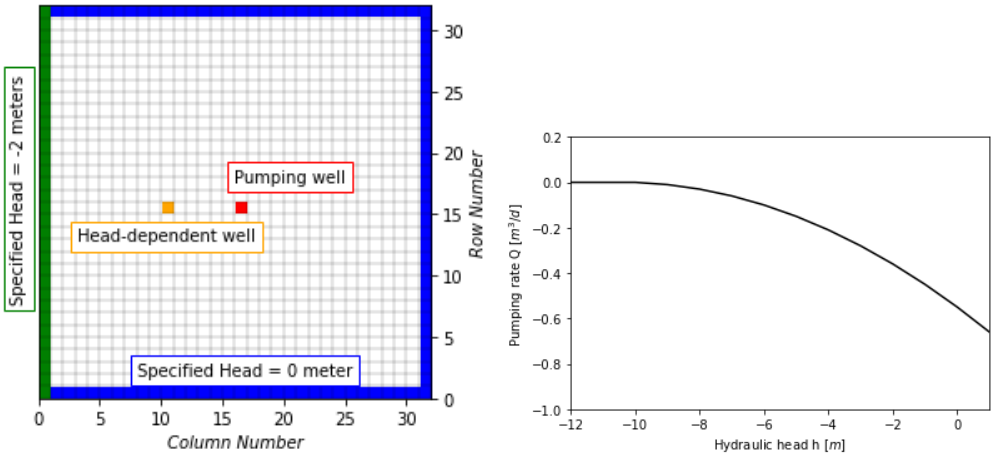


图 B.8：逆问题测试案例的设置。左：边界条件和源项；右：水头相关井的流量与水头之间的函数关系。

指定此用户自定义函数的一种简单方法，是叠加 MODFLOW 中已经实现的边界条件。排水边界是一种边界条件，其以一个称为导水系数的速率从含水层中排出水量，该速率与含水层中的水头和排水高程之差成正比。如果含水层中的水头低于排水高程，则排水边界会停用，流量为零。为了构建水头相关井的流量与水头之间的函数，我们决定在同一个网格单元中放置 11 个排水边界，使其上下重叠排列。最上方的排水边界位于高程 0 m 处，各排水边界在垂直方向上的间距为 1 m，且所有排水边界的导水系数均相同，为 $0.010m^2/d$。这 11 个独立分段函数的叠加结果，对应于网格单元中通量与水头之间的单一函数。可以将其视为一个单独的水头相关井，该井按照以下关系排水

i如图 B.8 右图所示，并通过插值得到近似函数：

$$Q(u) = \begin{cases} 0 & \text{if } u \leq -11 \\ -5.0 \cdot 10^{-7}x^5 + 2.0 \cdot 10^{-6}x^4 + 2.4 \cdot 10^{-4}x^3 + 2.9 \cdot 10^{-3}x^2 - 0.11x - 0.66 & \text{if } u > -11 \end{cases} \tag{B.1}$$

使用 iMOD Python 对单个 200 天的应力周期进行计算。计算输出为水头，如图 1（第四行）所示。

附录 C。比较神经网络模型

在本节中，我们将原始 DeepONet 的准确性与两种常用的神经网络模型进行比较。第一个模型是一种图像到图像回归方法，采用编码器-解码器（ED）架构[36] ，由空间下采样过程和随后的空间上采样过程构成。第二个模型是傅里叶神经算子（FNO） [37], ，利用快速傅里叶变换直接在傅里叶空间中对积分核进行参数化。

编码器网络由多个卷积层组成，后接批归一化和 ReLU 激活函数。维度最低的层包含 128 个特征图。解码器网络与编码器网络类似，但使用转置卷积层代替卷积层。每个卷积层使用大小为 4×4 、步幅为 2×2 的卷积核，并采用零填充。对于 $2D$ Darcy 流问题，我们采用 [37] 中所介绍的 FNO 实现。

不同网络架构的相对 L^2 测试误差汇总于表 C.3 中。在所有训练好的模型中，可以看出 DeepONet 架构具有最佳的预测准确率。此外，与另外两种必须使用整个区域内的点进行训练的神经算子方法相比，DeepONet 的优势在于可以基于少量稀疏观测进行训练。然而，为了进行公平比较，我们在考虑全场观测数据的情况下训练了 DeepONet。

相对 L^2 误差 / 模型	逆问题的相对 L^2 误差	非线性系统的相对 L^2 误差
DeepONet	$2.23E-01 \pm 4.30E-02$	$2.68E-02 \pm 7.53E-03$
ED	$3.70E-01 \pm 7.17E-02$	$3.16E-02 \pm 1.22E-02$
FNO	$3.16E-01 \pm 6.53E-02$	$3.67E-02 \pm 8.26E-03$

表 C.3：逆问题和非线性系统的基准测试：DeepONet、编码器-解码器和傅里叶神经算子的相对 L^2 预测值误差的均值和标准差。

附录 D. 训练数据的 SVD 分析

在源项可能出现在任意位置的情况下，原始 DeepONet 在逼近算子时表现出较低的准确率，这可归因于该模型处理平移、旋转和拉伸等对称变换时存在局限。Venturi [31] 将 DeepONet 描述为一种基于线性投影的方法，并指出当对象发生平移、旋转或缩放时，该方法存在与奇异值分解（SVD）相同的低效率问题 [38]。Venturi 通过展示主干网络如何提取投影的模态，建议对训练数据使用 SVD。这种方法有助于在探索阶段设计主干网络及其输出数量。

对输出解执行 SVD 可以得到捕获数据集中的方差或能量所需的模态数量。标记数据集中的每个输出解，其维度为 32×32，都会被重塑为列向量；随后，所有生成的向量都以列的形式水平堆叠在矩阵 $\mathbf{X}$ 中。图 D.9 展示了对前两个测试案例的数据集矩阵 $\mathbf{X}$ 进行 SVD 分析的结果。当抽水井固定在域中心时，第一模态中包含的能量等于 85%。当抽水井可以放置在任意位置时，其数值显著下降：第一模态仅捕获原始数据集 24% 的能量，并且 SVD 产生的奇异值衰减缓慢。结果表明，SVD 未能考虑数据的平移特性，因此失效。我们得出结论，这也是原始 DeepONet 无法准确预测未见过的井位置处的解的原因。

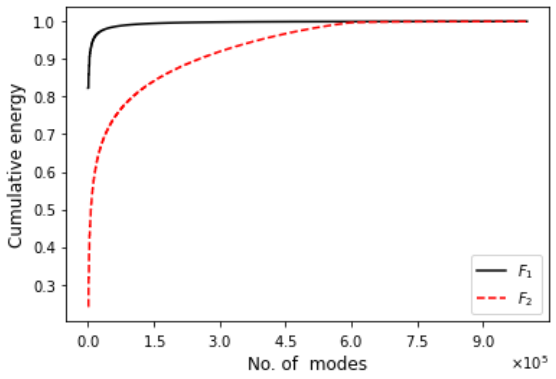


图 D.9：对实验 **E1** 的 F_1 和实验 **E2**的 F_2 数据集输出函数进行 SVD 分析：1）在整个训练数据集和测试数据集中，抽水井均位于区域的同一位置；2）抽水井可以位于任意位置。