DeepONet在求解含参热传导方程中的应用

李 慧¹, 欧 娜^{1*}, 宋晓燕²

¹长沙理工大学数学与统计学院,湖南 长沙 ²湖南工商大学理学院,湖南 长沙

收稿日期: 2025年5月9日; 录用日期: 2025年6月2日; 发布日期: 2025年6月10日

摘要

含参热传导方程在多个科学与工程问题中广泛存在,由于所含参数具有随机性,热传导方程的解也随着 不确定性的传播而具有随机性。为了刻画解的随机性,需要对该方程进行多次模拟仿真实验。另一方面, 在实际应用中,我们需要在给定稀疏观测值的条件下,基于前向模型来识别方程中的未知参数。贝叶斯 方法是识别未知参数的一个有效方法,借助观测数据,可以达到减小该方程中不确定性的目的。然而, 参数和前向模型之间的非线性性导致后验分布没有显式表达式,我们拟利用构造蒙特卡罗马尔科夫链的 方式,来实现后验分布的抽样。马尔科夫链的收敛要求数以百万次前向模型的仿真,构造前向模型的替 代模型是提高抽样效率的有效方式之一。随着未知参数维数的增加,构建替代模型所需要的离线模拟次 数也会呈指数级增加,这给参数识别问题带来了挑战。近年来,基于深度学习的算子学习方法,特别是 DeepONet,为求解此类问题提供了新的思路。本文以DeepONet为核心工具,基于方程的部分解数据, 构建和训练了一个随机热传导系数与观测数据之间的算子映射,并以此作为替代模型用于热传导方程的 参数识别问题当中,提高了参数反演的效率。我们通过数值算例验证了DeepONet在求解含参偏微分方程 时具有较高的精度,在高维未知参数的反演当中具有很好的应用效果。

关键词

DeepONet,热传导方程,深度神经网络,贝叶斯推断

Application of DeepONet in Solving Heat Conduction Equation with Parameters

Hui Li¹, Na Ou^{1*}, Xiaoyan Song²

¹School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha Hunan ²School of Science, Hunan University of Technology and Business, Changsha Hunan

Received: May 9th, 2025; accepted: Jun. 2nd, 2025; published: Jun. 10th, 2025

*通讯作者。

Abstract

The parameterized heat conduction equation is widely present in many scientific and engineering problems. Since the parameters involved are random, the solution of the heat conduction equation is also random as the uncertainty propagates. In order to characterize the randomness of the solution, it is necessary to conduct multiple simulation experiments on the equation. On the other hand, in practical applications, we need to identify the unknown parameters in the equation based on the front-end model under the condition of given sparse observations. The Bayesian method is an effective method to identify unknown parameters. With the help of observed data, the uncertainty in the equation can be reduced. However, the nonlinearity between the parameters and the forward model leads to the lack of explicit expression of the posterior distribution. We intend to use the method of constructing Monte Carlo Markov chains to achieve sampling of the posterior distribution. The convergence of the Markov chain requires millions of simulations of the forward model. Constructing a surrogate model of the forward model is one of the effective ways to improve sampling efficiency. As the dimension of the unknown parameters increases, the number of offline simulations required to construct the surrogate model will also increase exponentially, which brings challenges to the parameter identification problem. In recent years, operator learning methods based on deep learning, especially DeepONet, have provided new ideas for solving such problems. This paper uses DeepONet as the core tool, constructs and trains an operator mapping between a random heat conduction coefficient and observation data based on partial solution data of the equation, and uses this as a substitute model for parameter identification of the heat conduction equation, improving the efficiency of parameter inversion. We verify through numerical examples that DeepONet has high accuracy in solving parameter-containing partial differential equations, and has a good application effect in the inversion of high-dimensional unknown parameters.

Keywords

DeepONet, Heat Conduction Equation, Deep Neural Network, Bayesian Inference

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/

CC O Open Access

1. 引言

长期以来,有限差分法、有限元法、谱方法等传统数值方法在解决这类偏微分方程(PDE)问题方面发 挥了重要作用。然而,这些方法在高维参数空间中,常常面临计算效率低、网格依赖强、对复杂边界适 应性差等问题。DeepONet (Deep Operator Network)作为一种新兴的深度算子学习框架,为求解复杂参数 化偏微分方程提供了新的思路。相比传统方法,DeepONet 可以绕过繁琐的空间离散过程,直接从样本数 据中学习输入函数与解之间的映射关系,具有建模灵活、泛化能力强、计算速度快等优势。正因如此, 将 DeepONet 引入含参热传导方程的求解过程,有望突破传统方法在高维问题中面临的瓶颈,对相关工 程技术问题具有重要的实际意义和研究价值。关于含参热传导方程的研究,国内外学者主要集中在以下 两个方面:一是通过改进传统数值方法,提高在高维参数空间中的求解效率和精度;二是引入机器学习、 尤其是深度学习方法,探索新的解法框架。

PINNs (Physics-Informed Neural Networks)将物理规律嵌入损失函数中,在无网格的条件下求解问题。 尽管在小规模问题上取得了一定效果,但在处理高维参数时,仍存在训练不稳定、精度不高等问题,相 比之下, DeepONet 作为一种算子学习方法, 不是单纯的拟合函数, 而是直接学习函数到函数的映射。这种方法由 Lu 等人于 2021 年提出后, 在流体力学、电磁场计算和材料设计等多个领域得到了初步验证。 DeepONet 结构中分为"分支网络"和"主干网络", 分别处理输入函数与位置点信息, 从而有效表示算 子映射结构, 适用于解决含参 PDE 类问题。

近年来,深度学习在偏微分方程(PDEs)求解中的表现逐渐获得认可,尤其是在处理复杂物理系统和 含参问题方面,展现出传统数值方法难以比拟的优势。其中,Deep Operator Network (DeepONet)作为一种 能够学习非线性算子的神经网络架构,为解决含参 PDEs,特别是热传导问题,提供了全新的路径。

Karniadakis 等人提出的 DeepONet [1]-[3]能够直接学习从函数空间到函数空间的映射关系,与传统的基于点或网格的方式不同,它跳过了中间离散化步骤,从而在处理高维参数空间、时间演化类问题上展现了更强的泛化能力。Zhou 和 Lyu [4]的研究首次将 DeepONet 应用于扩散类方程的求解,显示出在低采样率和高参数复杂度下仍能保持良好精度,这对含参热传导问题的建模尤为关键。

与其相对的, Physics-informed Neural Networks (PINNs) [5] [6] 虽然在显式引入物理约束方面有一定优势,但在处理含参数的多实例任务上显得力不从心。多个工作[7] [8]指出, PINNs 在面对高维参数空间时难以收敛,而 DeepONet 在训练阶段通过对输入函数的广泛采样,可获得更强的泛化能力,这对解决参数空间大、边界复杂的热传导问题尤为重要。

DeepONet 的核心思想是通过神经网络训练输入与输出之间的关系,在含参热传导方程求解中具有巨大的潜力。例如,Lu et al. (2021) [9]提出了 DeepONet 的基本框架,并通过数值实验验证了该方法在多种物理问题中的适用性。研究表明,DeepONet 能够在高维和复杂边界条件下表现出优异的性能,尤其是在流体力学和热传导等领域的应用中,较传统的数值方法,DeepONet 显著提高了计算效率和精度。

本文旨在探索 DeepONet 在含参热传导方程中的应用效果与适应性,分析其在高维参数空间中的建模 能力和预测精度。全文围绕以下几个方面展开:第2章介绍本研究所需的基本理论知识,包括 DeepONet 工作原理、有限元方法在含参热传导方程中的应用,以及贝叶斯推断等相关概念;第3章围绕参数识别问 题展开实验设计,利用 DeepONet 逼近前向模型,并对系数进行反演,以验证其建模准确性和泛化能力。 总结部分对全文工作进行归纳,并对未来进一步优化模型结构、提升可解释性等方向提出思考与展望。

2. 基本方法

2.1. DeepONet 基本原理

DeepONet 不直接构造一个整体的函数到函数映射,而是一种离散点的方式处理。设G是一个从函数 空间 A 映射到函数空间 B 的非线性算子,即

 $G: \mathcal{A} \to \mathcal{B}$.

以热传导问题为例,输入为

$$\kappa(x) \in \mathcal{A}$$
,

 $\kappa(x)$ 为 $x \in [0,1]$ 上的热导率, 输出为

 $u(x,t) = \mathcal{G}(\kappa)(x,t) \in \mathcal{B},$

u(x,t)为在某些时间下的温度分布。

我们选取输入函数 $\kappa(x)$ 在一些固定位置上的值 $[\kappa(x_1),\kappa(x_2),\dots,\kappa(x_m)]$, 然后学习一个神经网络 N, 使得对于任意一个查询点 x, 有:

$$\mathcal{N}(\kappa(x_1),\cdots,\kappa(x_m);[x,t])\approx u(x,t)$$

这个过程实际上等价于学习算子 g 在一个离散版本下的逼近。

DeepONet 的核心结构由两个子网络组成:分支网络 \mathcal{B} 和主干网络 \mathcal{T} ,其中 \mathcal{B} 以输入函数 $\kappa(x)$ 的离散采样值作为输入,输出一个低维的特征向量,记作 $b \in \mathbb{R}^{p}$ 。 \mathcal{T} 以查询点x或更广义的输入如(x,t)为输入,输出一个相同维度的向量 $t(x) \in \mathbb{R}^{p}$ 。最终,网络的预测值为两者内积:

$$u(x,t) \approx \mathcal{N}(\kappa;x) \coloneqq \langle \mathcal{B}(\kappa), \mathcal{T}(x) \rangle = \sum_{i=1}^{p} b_i \cdot t_i(x)$$

该结构充分利用了算子的线性泛函逼近性质,而训练过程则通过最小化训练样本上的均方误差来进 行优化:

$$\mathcal{L} = \frac{1}{N_1} \sum_{i=1}^{N_1} \sum_{j=1}^{M} \left| \mathcal{N} \left(\kappa_i; x_j \right) - u_i \left(x_j \right) \right|^2$$

其中 κ_i 是第i个输入函数, x_j 是第j个查询点, $u_i(x_i)$ 是真实值。在实际操作当中,会使用图 1 这种结构。 图 1 这种结构分离了 Trunk 与 Branch 两个关键维度,其是学习参数 - 空间双连续性的算子逼近。Trunk 支路从 $\mu(x)$ 接收信息,Branch 从 Loop 节点表征参数循环输入,双路特征在顶层交汇后输出全域解。该 架构通过参数化建模实现了热传导方程在变参数场景下的推演,其模块化设计既保证了网络的可解释性, 也便于不同规模方程的适应性调整。在部分问题中,输入函数并非单一变量,而是多个函数的组合,例 如热源项f(x,t),参数 $\kappa(x)$ 、边界条件g(t)同时参与 PDE 的求解过程。



DeepONet 的逼近能力并不是凭空设想的。其理论基础源于经典的柯西 – 皮卡尔定理和泛函分析中的 Karhunen-Loève 展开理论。Lu 等人在其原始论文中证明,只要算子 g 是连续的,那么存在某种 DeepONet 架构,可以以任意精度逼近该算子。从逼近角度来看,如果函数 $\forall u(x) \in \mathcal{B}$ 可写作:

$$u(x) \approx \sum_{i=1}^{p} c_i \phi_i(x)$$

那么 DeepONet 的输出结构就是在尝试自动学习一组基函数 $\phi_i(x)$,并对每组输入函数 $\kappa(x)$ 给出对应的系数 c_i 。这种机制非常类似于主成分分析(PCA)中的投影思想,只不过它是"非线性函数空间"的对应。DeepONet 之所以能够逼近任意非线性算子,源于其所具有的理论保证。Lu *et al.* (2021)等人在其论文中提出,若*G* 是连续算子,且输入函数 u 属于 Banach 空间 χ ,则存在一组参数 θ ,使得 DeepONet 的输出 $\hat{\mathcal{G}}_{\theta}(u)(y)$ 在 $\chi \times D$ 上一致逼近 $\mathcal{G}(u)(y)$ 。即对于任意 $\varepsilon > 0$,存在神经网络结构及其参数,使得:

$$\sup_{\varepsilon \mathcal{X}, v \in D} \left| \mathcal{G}(u)(y) - \hat{\mathcal{G}}_{\theta}(u)(y) \right| < \varepsilon,$$

这一性质从理论上保证了 DeepONet 对复杂偏微分算子的逼近能力,是其区别于传统数据驱动回归 模型的关键所在。

2.2. 有限元方法产生训练数据

有限元方法(FEM)因其对复杂几何结构和边界条件具有较强适应性,在求解偏微分方程尤其是热传导类问题中得到了广泛应用。对于含参热传导方程而言,FEM 依然是一种可靠且相对成熟的手段,其基本思路是将问题定义域划分为有限数量的子区域,然后在这些单元上构建近似解函数,最终将偏微分方程转化为一组代数方程进行求解。

考虑如下含参热传导方程的通用形式:

$$\frac{\partial u(x,t;\kappa)}{\partial t} = \nabla \cdot \left(\kappa(x)\nabla u(x,t;\kappa)\right) + f(x,t;\kappa), \quad x \in \Omega, t \in [0,T],$$

其中 $u(x,t;\kappa)$ 表示温度场, κ 是控制系统变化的参数, $\kappa(x)$ 为热导率函数, $f(x,t;\kappa)$ 为热源项, Ω 表示 空间区域, T 为终止时间。

采用有限元方法求解该问题的关键步骤包括: 首先, 对空间区域Ω进行剖分, 得到由多个简单几何 单元组成的网格结构, 其次, 选取合适的基函数来构造试函数和检验函数空间, 再将方程的弱形式代入 这些函数空间, 并对参数 κ进行离散处理, 最终形成一个与参数相关的半离散或全离散系统。

为了求解含参热传导方程:我们首先将其转化为变分问题,即构造其弱形式。为此,定义试函数 $u \in V$,测试函数 $v \in V$,其中 $V \subset H^1(\Omega)$ 为合适的函数空间。对原始方程两边同时乘以测试函数v,并在区域 Ω 上积分,得到:

$$\int_{\Omega} \frac{\partial u(x,t;\kappa)}{\partial t} v(x) dx = -\int_{\Omega} \kappa(x) \nabla u(x,t;\kappa) \cdot \nabla v(x) dx + \int_{\Omega} f(x,t;\kappa) v(x) dx.$$

利用分部积分和适当的边界条件,可以得出如下弱形式:

$$\left(\frac{\partial u}{\partial t}, v\right) + a(u, v; \kappa) = (f, v)$$

其中双线性形式 $a(u,v;\kappa) = \int_{\Omega} \kappa(x) \nabla u \cdot \nabla v dx$,而右端项表示源项f与测试函数的内积。

接下来,对空间区域 Ω 进行剖分,构造网格 T_h ,引入有限维子空间 $V_h \subset V$,基函数记作:

 $\{\phi_1(x),\phi_2(x),\dots,\phi_N(x)\}$ 。设近似解为: $u_h(x,t;\kappa) = \sum_{j=1}^N U_j(t;\kappa)\phi_j(x)$,将上述近似代入弱形式,并选择测试

函数为基函数中的任一项 $v = \phi_i(x)$,得到:

$$\sum_{j=1}^{N} \left(\int_{\Omega} \phi_{j}(x) \phi_{i}(x) dx \right) \frac{dU_{j}(t;\kappa)}{dt} + \sum_{j=1}^{N} \left(\int_{\Omega} \kappa(x) \nabla \phi_{j}(x) \cdot \nabla \phi_{i}(x) dx \right) U_{j}(t;\kappa)$$
$$= \int_{\Omega} f(x,t;\kappa) \phi_{i}(x) dx.$$

质量矩阵为

$$M_{ij} = \int_{\Omega} \phi_j(x) \phi_i(x) \mathrm{d}x \,,$$

刚度矩阵为

$$K_{ij}(\kappa) = \int_{\Omega} \kappa(x) \nabla \phi_j(x) \cdot \nabla \phi_i(x) dx,$$

DOI: 10.12677/aam.2025.146298

载荷向量为

$$F_i(t;\kappa) = \int_{\Omega} f(x,t;\kappa) \phi_i(x) dx,$$

解系数向量为

$$U(t;\kappa) = \begin{bmatrix} U_1(t;\kappa) \\ U_2(t;\kappa) \\ \vdots \\ U_N(t;\kappa) \end{bmatrix},$$

最终我们得到如下的半离散线性系统(ODE 形式):

$$M \frac{\mathrm{d}U(t;\kappa)}{\mathrm{d}t} + K(\kappa)U(t;\kappa) = F(t;\kappa).$$

这一表达式表示热传导方程在空间离散后形成的常微分方程组,其解仍依赖于时间t与参数 κ ,其中 M 为对称正定的质量矩阵, $K(\kappa)$ 是随参数 κ 变化的刚度矩阵, $F(t;\kappa)$ 是源项矢量,体现了源项随时间 和参数的变化。该系统的求解可借助时间离散方法,如向后欧拉法、Crank-Nicolson 法或显式 Runge-Kutta 方法,最终获得全离散形式的数值解。

2.3. 贝叶斯推断

贝叶斯推断的核心思想是利用先验知识结合观测数据,更新对参数的认知,从而得到后验分布。在参数识别问题中,后验分布提供了关于未知参数的不确定性信息,使得该方法在噪声数据或不完全观测 情况下仍能进行有效推断。贝叶斯定理的数学表达式如下:

$$P(\kappa|d) = \frac{P(d|\kappa)P(\kappa)}{P(d)},$$

其中 κ 表示待估计的参数, d表示观测数据, $P(\kappa|d)$ 为后验分布, $P(\kappa)$ 为先验分布, $P(\kappa)$ 反映了对参数 κ 的先验知识。似然函数 $P(d|\kappa)$ 反映了参数 κ 对观测数据的拟合程度,通常由测量噪声模型决定,例如高斯噪声下的似然函数可表示为:

$$P(d|\kappa) \propto \exp\left(-\frac{\left\|d-G(\kappa)(X_{I})\right\|}{2\sigma^{2}}\right),$$

其中 $G(\kappa)(X_I)$ 是给定参数 κ 下模型在观测位置的预测值, σ^2 用于描述观测数据与前向模型之间的偏差, 替代模型的引入会使得偏差发生变化,这里我们将 σ^2 视为超参数,为其配置逆伽马分布超先验,即

$$p(\sigma^2) \sim IG\left(\frac{\nu}{2}, \frac{\nu\lambda}{2}\right),$$

其中v和 λ 为预先确定好的参数。经推导可得出 σ^2 的条件后验分布为

$$p(\sigma^2 | d, \kappa) \sim IG\left(\frac{n+\nu}{2}, \frac{\left\|d-G(\kappa)(X_1)\right\|^2 + \nu\lambda}{2}\right).$$

在每次迭代中,可根据当前观测误差更新 σ^2 的后验分布参数,并利用 Gibbs 采样对其进行条件抽样,从而实现对噪声方差的不确定性建模。

由于后验分布通常无法解析求解,因此需要采用数值方法进行逼近,如马尔可夫链蒙特卡洛(MCMC)

方法。经典的 Metropolis-Hastings 算法,即通过接受-拒绝机制生成后验分布的样本,这类算法随着未知 参数维数的增加,容易出现退化的现象,即随着参数维数的增加,接受率趋于零,马尔科夫链的混合速 率下降。pCN-MCMC 抽样方法可以有效的克服这个问题,它通过离散随机微分方程来生成预选样本,并 且接受率不会随着参数维数的增加而发生改变,它是一种专门用于函数空间中的贝叶斯采样方法,在维 度升高时仍保持良好的接受率和样本多样性。具体来说,pCN-MCMC 抽样方法通过离散化随机微分方程 产生预选样本,即

$$\boldsymbol{\kappa}^{(*)} = \sqrt{1 - \beta^2} \boldsymbol{\kappa}^{(k)} + \left(1 - \sqrt{1 - \beta^2}\right) \boldsymbol{\mu} + \beta \boldsymbol{\omega}^{(k)}, \quad \boldsymbol{\omega}^{(k)} \sim \mathcal{N}(0, \boldsymbol{\Sigma}),$$

 $\beta \in (0,1)$ 为控制提议步长的超参数。一般地, 需要调试参数 β 使得接受率在 30%~50%。接受概率的 表达式为

$$\alpha\left(\kappa^{(k)},\kappa^{(*)}\right) = \min\left\{1,\exp\left[\Phi\left(\kappa^{(k)}\right) - \Phi\left(\kappa^{(*)}\right)\right]\right\}$$

其中势函数为

$$\Phi(\kappa) = \frac{\left\| d - G(\kappa)(X_I) \right\|}{2\sigma^2}.$$

具体的抽样过程见算法1。

算法 1. pCN-MCMC 算法

Input: 观测位置的集合 X_I , 样本总数 N, 后验样本集合 Q

- 1: 初始化:从先验分布中采样得到初始样本 $\xi^{(0)} \sim \mathcal{N}(\mu, \Sigma)$
- 2: 计算前向模型在 $\kappa^{(0)}$ 处的取值, $U_0 = G(\kappa^{(0)})(X_1)$
- 3: 初始化超参数

$$\left(\sigma^{2}\right)^{(0)} \sim IG\left(\frac{n+\nu}{2}, \frac{\left\|d-U_{0}\right\|^{2}+\nu\lambda}{2}\right)$$

4: for k = 1:N 做循环
(i) 构造提议样本:

$$\kappa^{(*)} = \sqrt{1 - \beta^2} \kappa^{(k)} + \left(1 - \sqrt{1 - \beta^2}\right) \mu + \beta \omega^{(k)}, \quad \omega^{(k)} \sim \mathcal{N}\left(0, \Sigma\right)$$

(ii) 计算前向模型在 $\kappa^{(*)}$ 处的取值, $U_1 = G(\kappa^{(*)})(X_1)$

(iii) 计算接受概率:

$$\alpha\left(\boldsymbol{\xi}^{(k)},\boldsymbol{\xi}^{(*)}\right) = \min\left\{1,\exp\left[\Phi\left(\boldsymbol{\xi}^{(k)}\right) - \Phi\left(\boldsymbol{\xi}^{(*)}\right)\right]\right\}$$

(iv) if
$$v \sim \mathcal{U}(0,1)$$
小于 $\alpha(\xi^{(k)},\xi^{(*)})$ 则

$$\begin{split} \boldsymbol{\kappa}^{(0)} &= \boldsymbol{\kappa}^{(\star)} \\ \boldsymbol{U}_0 &= \boldsymbol{U}_1 \end{split}$$

结束 if 判断语句

(v) 更新超参数

$$\left(\sigma^{2}\right)^{(k)} \sim IG\left(\frac{n+\nu}{2}, \frac{\left\|d-U_{0}\right\|^{2}+\nu\lambda}{2}\right)$$

(vi) 将样本存储于Q。

结束 for 循环

3. 数值算例

本研究中,数值算例主要基于固定的参数分布、边界条件和初始条件进行验证。未来工作将扩展测试范围,涵盖多种参数统计特性、边界条件类型(如 Neumann 和 Robin 条件)以及多样化的初始条件设置,以全面评估 DeepONet 在不同物理场景下的性能和鲁棒性。

在本文中,考虑如下形式的一维热传导偏微分方程(无反应项):

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x} \left(k(x) \frac{\partial u(x,t)}{\partial x} \right), \quad x \in (0,1), t \in (0,T]$$

其中u(x,t)表示位置x处在时间t时刻的温度分布,k(x)为位置相关的热导率函数,它是我们需要识别的 未知参数函数。由于热导率函数通常难以直接测量,因此我们采用贝叶斯推断的方式,通过有限的温度 观测数据推断k(x)的后验分布。该问题的边界条件取为第一类齐次边界条件(Dirichlet 条件):

$$u(0,t) = u(1,t) = 0, \quad \forall t \in [0,T]$$

初始条件 $u(x,0) = u_0(x)$,选取为多峰高斯函数组合形式,具体如下:

$$u_0(x) = \frac{7}{\sqrt{2\pi l_0}} e^{-\frac{(x-0.3)^2}{2l_0^2}} - \frac{2}{\sqrt{2\pi l_1}} e^{-\frac{(x-0.6)^2}{2l_1^2}} + \frac{4}{\sqrt{2\pi l_2}} e^{-\frac{(x-0.8)^2}{2l_2^2}}$$

其中 l_0, l_1, l_2 分别为控制峰宽度的参数。热传导系数 $\kappa(x)$ 的先验为一个均值 $\mu = 3$ 的高斯过程,即 $\kappa(x) \sim gp(\mu, C)$

其中核函数为

$$C(x, x') = 0.5 \exp\left(-\frac{(x-x')^2}{2l^2}\right),$$

长度尺度系数1=0.2。

3.1. 单层训练数据

首先,我们测试了 DeepONet 在输出变量为热传导方程在时间 $t_0 = 0.0015$ 时的解的拟合效果,图 2 展示了随机热传导系数的 4 个实现和相对应的有限元数值解在 $t_0 = 0.0015$ 的空间分布。





图 2. 随机热传导系数的 4 个实现和相对应的解在 t₀ = 0.0015 的空间分布

训练数据组数为*N*₁=1000,测试数据组数为*N*=500,*x*方向上随机取点数为*M*=80。分支网络与 主干网络均采用 6 层全连接网络,每层包含 20 个神经元,激活函数使用 tanh,输出维度均为 25,对应 DeepONet 中的潜在特征维度 *p*=25。两个网络分别包含 4225 和 2245 个参数。将网络训练参数设置为 epoch=5000,学习率为*lr*=0.001,得到结果如图 3 所示。可以看出,训练 MSE 和测试 MSE 都随着迭 代的进行迅速下降,训练损失的收敛过程,从初始较大误差迅速下降至稳定低值,表明模型拟合能力逐 步增强。测试集上的平均误差可看作泛化误差,该曲线亦显示出模型在训练过程中对未见数据的预测能 力逐渐提升。从图中可以观察到,网络在初期损失较大,但随着训练的深入,损失迅速下降并趋于平稳, 验证了 DeepONet 能有效学习输入函数与输出解之间的映射关系。



Figure 3. Training and testing loss function diagram for a single layer of time output variables 图 3. 单层时间输出变量时,训练和测试损失函数图

将测试集中的函数值与空间位置带入训练好的 DeepONet 模型中,可得到相对应的输出变量,将其 与有限元方法(FEM)计算所得的参照解进行对比,我们测试了 DeepONet 在 κ 的两个随机实现上的表现, 如图 4(a)所示,实线为 FEM 产生的参考解,虚线为 DeepONet 的逼近解。对同一个 κ 的样本,参考解与 逼近解的曲线在空间域 $x \in [0,1]$ 上高度吻合。图 4(b)为 DeepONet 逼近解和 FEM 参考解在 x 在 [0,1] 各点 上的平均误差,其中平均误差的定义为:

$$\operatorname{eror}(x_{j}) = \sum_{i=1}^{N} \frac{\left\| G(\kappa^{(i)})(x_{j}) - u(x_{j}, t_{0}; \kappa^{(i)}) \right\|^{2}}{\left\| u(x_{j}, t_{0}; \kappa^{(i)}) \right\|^{2}}.$$

由图 4 可知测试集中所有样本点处预测解与 FEM 解的相对误差均值不超过 0.04。我们可以看到在边 界点处和 *x* = 0.6 处的相对误差比较大,这是因为被逼近的解在这些位置变化较为剧烈,神经网络在捕捉 高频信息上的局限性,使得 DeepOnet 产生的逼近解具有较大的误差。

3.2. 多个时间层输出变量

在小节 3.1 中,我们只考虑了 x 作为主干网络的输入变量,进一步地,我们将时间 t 也作为主干网络的输入变量,即主干网络的输入变量维数为 2。训练数据组数为 N_1 =1500,测试数据组数为 N = 500, x 方向上随机取点数为 $M_x = 800$, t 方向上随机取点数为 $M_t = 800$,即 trunk net 的每一组输入变量维度为 $N_2 = M_x * M_t = 6400$ 。分支网络与主干网络的网络架构跟小节 3.2 中的相同。将网络训练参数设置为

epoch = 8000, 学习率为 *lr* = 0.001, 得到结果如图 5 所示, 和前文类似, 训练和测试的 MSE 值都随着迭代的进行迅速下降。



Figure 4. (a) Comparison of the approximate solutions of DeepONet on any two realizations of κ and their corresponding FEM reference solutions; (b) Mean relative error between DeepONet's approximate solution on the test set and the FEM reference solution

图 4. (a) DeepONet 在 κ 任意两个实现上的近似解与相对应的 FEM 参考解对比图; (b) DeepONet 在测试集上的近似 解与 FEM 参考解相对误差的均值



Figure 5. The training and testing loss functions as functions of the number of Epochs when *x* and *t* are used as input variables for the trunk net 图 5. $x \ \pi t$ — 起作为 trunk net 输入变量时,训练和测试损失函数随 Epochs 变化图

图 6 展示了 x 和 t 一起作为主干网络输入变量时, DeepONet 近似解与 FEM 参考解在测试集上的均值, 两种解的均值大体上呈现出相同的形状。另外, 我们按照如下公式计算了每个时间层上的平均相对误差:

$$\operatorname{error}(t_{k}) = \frac{1}{N} \sum_{i=1}^{N} \frac{\sum_{j=1}^{N_{x}} \left\| G(\kappa^{(i)})(x_{j}, t_{k}) - \hat{u}(x_{j}, t_{k}; \kappa^{(i)}) \right\|^{2}}{\sum_{j=1}^{N_{x}} \left\| \hat{u}(x_{j}, t_{k}; \kappa^{(i)}) \right\|^{2}}.$$

DOI: 10.12677/aam.2025.146298



Figure 6. (a) Mean of DeepONet approximate solution; (b) Mean of FEM reference solution 图 6. (a) DeepONet 近似解的均值; (b) FEM 参考解的均值

图 7 展示了 DeepONet 预测热传导问题的误差随时间演变的特点。可以看到误差曲线呈现"U 型" 趋势,这是因为在 *t* = 0 附近温度场突变时模型对高梯度场的捕捉不足,随着热扩散效应主导,在 *t* = 0.02~0.06 区间误差稳定低于 1.5%,表明模型在准稳态阶段与有限元解高度吻合,而在 *t* > 0.06,误差回 升至接近初始水平,这源于温度趋近于零时数值舍入误差被相对放大。



Figure 7. Average relative error between DeepONet approximate solution and reference solution at each time layer 图 7. DeepONet 近似解与参考解在每个时间层上的平均相对误差

3.3. 热传导系数的反演

接下来,我们将尝试将 DeepONet 建立的替代模型用于反问题的求解。反问题的目标是通过有限的 观测数据 u(x,t) 反演出 PDE 的未知参数 $\kappa(x)$ 。在这一小节中,我们取时刻 0.02,0.04,0.06,0.08 分布在区 间 [0.1,0.9] 的温度为测量数据。测量数据由 $\kappa(x)$ 一个先验实现生成,测量误差水平设为 $\sigma = 0.1$ 。为了达 到更好的逼近效果,我们先用 FEM 方法生成 u(x,t) 在观测位置和时刻的值,再用 DeepONet 构造算子映 射,并将其应用于参数反演,预测输出变量的后验信息。

调节参数 β 使接收概率在 35% 左右,我们得到 40,000 个后验样本。基于保留的后 20,000 个样本,我 们估计了后验样本的均值和 3 倍标准差区间,如图 8 所示。可以看出,虽然后验均值与真实的系数之间 有一定的差别,后验置信区间可以将大部分真实系数场的取值囊括其中。

在参数反演过程中,观测数据的噪声对后验推断带来显著挑战。本文通过引入噪声方差的超先验分布,并采用 pCN-MCMC 采样方法,有效地处理了噪声不确定性,保证了反演的稳定性。此外,先验分布的选择对反演结果影响较大,本研究采用高斯过程先验,未来将考虑结合物理知识设计更合理的先验分布,以提升反演的准确性和可靠性。

与传统有限元方法相比, DeepONet 在训练阶段需要一定的计算资源, 但其推断阶段的计算速度显著 提高, 能够实现数百至数千倍的加速, 极大提升了参数反演中前向模型的计算效率。从精度角度看, DeepONet 在大部分测试样本中与有限元解保持较好一致, 尤其在准稳态阶段误差较低, 但在边界及高梯 度区域存在一定误差, 提示未来需进一步优化网络结构以提升高频信息捕捉能力。综上, DeepONet 作为 高效的替代模型, 在实际工程参数识别中展现出较高的应用价值。



Figure 8. Comparison of the posterior sample mean, 3 times standard deviation interval and reference coefficient based on the DeepONet substitution model

4. 总结

本论文围绕算子学习中的前沿方法 DeepONet,在求解含参数热传导方程问题中的应用进行了实验验证。我们利用 DeepONet 构建了热传导系数与热传导方程的解之间的映射,并将其作为前向模型的替代模型,用于高维未知函数的贝叶斯推断当中,节省了后验分布的抽样时间,并能得到一个比较好的效果。 DeepONet 模型能够有效捕捉输入函数与空间解之间的复杂非线性映射关系,在测试样本中展现出较高的预测精度与良好的收敛速度。本文通过理论与实验相结合的方式,初步验证了 DeepONet 在求解含参热传导类偏微分方程问题中的潜力,为其在更复杂物理系统中的应用奠定了基础。

致 谢

欧娜感谢国家自然科学基金委 11901060,湖南省自然科学基金 2021JJ40557 以及湖南省教育厅优秀 青年项目 22B0333 的支持;宋晓燕感谢国家自然科学基金委 12301551,湖南省自然科学基金 2022JJ40125 以及湖南省教育厅优秀青年项目 22B0635 的支持。

图 8. 基于 DeepONet 替代模型的后验样本均值, 3 倍标准差区间和参考系数的对比图

参考文献

- Cotter, S.L., Roberts, G.O., Stuart, A.M. and White, D. (2013) MCMC Methods for Functions: Modifying Old Algorithms to Make Them Faster. *Statistical Science*, 28, 424-446. <u>https://doi.org/10.1214/13-sts421</u>
- [2] Lyu, Z., Xie, Y. and Karniadakis, G.E. (2021) Deeponet: Learning Nonlinear Operators for Identifying Differential Equations from Data. *Proceedings of the National Academy of Sciences of the United States of America*, **118**, e2020382118.
- [3] Zhao, T., Wang, Z. and Karniadakis, G.E. (2021) Physics-Informed DeepONets for Nonlinear Operators with Noisy Data. *Journal of Computational Physics*, 426, Article ID: 109913.
- [4] Zhou, L. and Lyu, Z. (2021) Solving Nonlinear Reaction-Diffusion Equations with Deeponet. *Mathematics of Computation*, **90**, 121-139.
- [5] 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. <u>https://doi.org/10.1016/j.jcp.2018.10.045</u>
- [6] Tarantola, A. (2005) Inverse Problem Theory and Methods for Model Parameter Estimation. Society for Industrial & Applied Mathematics. <u>https://doi.org/10.1137/1.9780898717921</u>
- [7] Yin, M., Zhang, E., Yu, Y. and Karniadakis, G.E. (2022) Interfacing Finite Elements with Deep Neural Operators for Fast Multiscale Modeling of Mechanics Problems. *Computer Methods in Applied Mechanics and Engineering*, 402, Article ID: 115027. <u>https://doi.org/10.1016/j.cma.2022.115027</u>
- [8] Wang, X. and Sun, W. (2022) Solving Nonlinear Reaction-Diffusion Systems via Neural Networks. *Physics Letters A*, 426, Article ID: 127897
- [9] Lu, L., Jin, P. and Karniadakis, G.E. (2019) DeepXDE: A Deep Learning Library for Solving Differential Equations. *SIAM Journal on Scientific Computing*, **41**, A2423-A2446.