

DOI: 10.11779/CJGE20211138

基于 PINNs 算法的地下水渗流模型求解及参数反演

张 升, 兰 鹏, 苏晶晶*, 熊海斌

(中南大学土木工程学院, 湖南 长沙 410075)

摘 要: 地下水渗流模型的渗流流速计算(正向求解)和渗流参数反演(反向求解)工程意义重大,但目前能同时解决两类问题的算法较少。针对该问题,引入了物理信息神经网络(PINNs)算法,并加入硬约束进行改进,在正向求解方面,分别建立了渗流方程与达西定律耦合的水头、流速同时求解方法(PINNs-H-I),以及先计算水头再通过自动微分求解流速的计算方法(PINNs-H-II)。对于反向求解,分别采用单(多)物理场神经网络模型的 PINNs 算法反演均质(非均质)渗流参数。通过算例分析表明,相比软约束 PINNs 算法,通过施加硬约束可同时改善正向求解和反向求解的性能,另外在正向渗流速度计算中 PINNs-H-II 方法具有更高的计算精度,同时单(多)物理场神经网络模型 PINNs 算法反演的均质(非均质)渗流参数与实际值符合较好。

关键词: 地下水渗流模拟; 渗流参数反演; 物理信息神经网络; 硬约束

中图分类号: TU445

文献标识码: A

文章编号: 1000-4548(2023)02-0376-08

作者简介: 张 升(1979—),男,教授,主要从事计算土力学方面的科研工作。E-mail: zhang-sheng@csu.edu.cn。

Simulation and parameter identification of groundwater flow model based on PINNs algorithms

ZHANG Sheng, LAN Peng, SU Jingjing, XIONG Haibin

(School of Civil Engineering, Central South University, Changsha 410075, China)

Abstract: The simulation of the Darcy velocity (forward problem) and the identification of seepage parameters (backward problem) in the groundwater flow model are of significance to practical projects, while at present, few algorithms can be used to simultaneously tackle these two problems. The physics-informed neural networks (PINNs) algorithms with the hard constraints are introduced for investigating these two problems at the same time. For the forward problem, two methods are established for deriving the Darcy velocity. One is to address the groundwater head and Darcy velocity concurrently by coupling the seepage flow equation with the Darcy's law (PINNs-H-I), and the other is to calculate the groundwater head first and then solve the Darcy velocity by automatic differentiation (PINNs-H-II). For the backward problem, the PINNs algorithms for the single and multi-physical field neural network models are used to identify the seepage parameters of homogeneous and non-homogeneous seepages, respectively. Furthermore, several examples are presented, and the results show that the hard-constraint PINNs algorithms exhibit better performances for the forward and backward problems compared with the soft-constraint ones. In addition, it is noted that PINNs-H-II possesses higher calculation accuracy, and both the PINNs algorithms for the single and multi-physical field neural network models can accurately identify the seepage parameters in the homogeneous and non-homogeneous seepage.

Key words: groundwater numerical simulation; seepage parameter identification; physics-informed neural network; hard constraint

0 引 言

地下水渗流问题涉及到石油工程、采矿工程、岩土工程、水利工程和环境工程等多个工程应用领域,其中,在给定渗流参数的情况下精确求解达西流场(正向问题)和基于渗流实测数据合理识别渗流参数(反向问题)对于渗流模拟分析具有重要意义^[1-3]。

对于正向问题,鉴于物理试验的局限性,达西渗

流流速的求解主要采用数值模拟,具体方式分为两种^[2]:

①首先对地下水水头进行分析,再基于达西定律求解

基金项目: 国家重点研发计划项目(2017YFE0119500); 湖湘高层次人才聚集工程-创新团队项目(2019RS1008); 湖南省研究生科研创新项目(cx20220109); 中南大学研究生自主探索创新项目(2022zzts0018); 湖南省自然科学基金青年基金项目(2022JJ40566)

收稿日期: 2021-09-26

***通信作者** (E-mail: jingjingsu@csu.edu.cn)

渗流流速^[1-2, 5]; ②建立达西渗流定律和渗流模型的耦合方程, 通过联合求解同时得到渗流流速和水头^[6-7]。两种方式都已被证明能够进行准确的流速计算, 但却难以依据渗流数据对渗流参数进行反演。关于渗流参数的反演, 目前主要基于水头或流速数据, 采用最速下降法、高斯-牛顿法、遗传算法等优化算法^[8]进行求解, 此类方法虽对均质渗流参数具有较好计算表现, 但当地下水水流为非均质流情况时(渗流参数呈现空间变异性), 却难以获得渗流参数的空间分布特点。因此, 发展一种既能计算非均质渗流模型的流速场, 又能反演其渗流参数的新算法具有重要意义。

近年来, 随着人工智能的快速发展, 基于深度学习万能逼近定理^[10]建立的物理信息神经网络(Physics-Informed Neural Networks, PINNs)算法受到了学者的广泛关注, 其核心原理: 利用自动微分技术(Automatic Differentiation, AD)^[11]将偏微分方程嵌入到神经网络的损失函数当中, 再通过优化算法对损失函数进行寻优, 从而实现神经网络对方程解的近似, 具有不受数据量限制、求解格式稳定等特点。此外, 相对于有限元法(FEM)、有限差分法(FDM)等, PINNs 算法可通过在损失函数中加入实测数据的损失项, 对方程的未知物理参数(如渗流方程中的均质或非均质渗流参数)进行反演, 甚至可通过较少数据直接挖掘物理规律并构建出相应的物理方程, 故在岩土工程研究中具有一定的应用前景。目前, PINNs 算法已在一些领域得到了较广泛的应用, 如 Haghighat 等^[12]将 PINNs 算法引入到固体力学中, 基于 Python 语言的 Keras 开发了 SciANN 框架, 对带缺陷的平板拉伸变形行为进行分析, 得到了平板内各点的位移。基于 SciANN 框架, 唐明健等^[13]对矩形薄板的正反问题进行了分析, 准确给出了矩阵薄板各点的挠度, 并利用薄板挠度反演得到了边界的受力情况。Raissi 等^[14]通过将 PINNs 算法应用于流体力学当中, 准确得到了 Burgers 方程、退化 Navier-Stokes 方程的数值解, 并实现了方程中流动参数的识别。Chen 等^[15]基于实测数据, 利用 PINNs 算法在大量备选的偏微分算子中, 寻找合适的算子, 实现了 Burgers 方程的智能建立。PINNs 算法在岩土工程领域也有一些探索, Zhang 等^[16]引入了独热编码向量, 与 PINNs 算法相结合, 对多孔介质的较大变形固结问题进行了分析。Bekele 等^[17]采用 PINNs 研究了多孔介质中的流体流动与固体变形之间的相互作用。兰鹏等^[18]采用 PINNs 算法对连续排水边界的一维固结问题进行了分析, 得到了土体的超孔隙水压力消散规律, 并实现了连续排水边界界面参数的准确反演。然而, 上述的 PINNs 算法的应用存在如下限制, 一方面, PINNs 算法在进行反问题分析时, 学

者采用的是单物理场神经网络模型, 此模型只能对未知参数为定值的情况进行反演, 当反演的参数呈现空间变异性时, 则无法进行识别。因此, 对于非均质地下水渗流问题, 采用单物理场神经网络模型 PINNs 算法无法获得非均质渗流参数分布, 此时需引入多物理场神经网络方法^[19-22]; 另一方面, 上述应用中, PINNs 算法在进行求解时, 对于边界的处理主要是软边界形式。

事实上, PINNs 算法软硬约束边界的设置会对计算结果产生较为显著的影响, 原因在于 PINNs 算法在进行求解时, 需要构造方程、初始和边界条件的损失函数。初始和边界条件的损失函数主要是通过初始和边界上随机选取样本点及其取值获得, 所以在进行优化时, 只能保证在选取的样本点处拟合结果较好, 无法保证在初始和边界上均满足条件, 故而属于一种软约束形式。针对软约束这一问题, Lu 等^[23]和 Sun 等^[24]引入辅助函数对其进行修正, 使 PINNs 计算的结果自动满足边界或者初始条件, 此时边界约束形式被称为硬约束形式, 相比于软约束, 具有更高的计算精度。目前, PINNs 算法对于硬约束的应用较少, 主要集中在如下方面, Lagari 等^[25]总结了不同类型的边界条件的 PINNs 硬约束构造方法, 随后, Lu 等^[26]和陆至彬等^[27]分别将硬约束引入到了超材料设计和平板导热问题分析中, 结果表明, 硬约束可以获得更高的求解精度。但是在地下水渗流领域, PINNs 硬约束应用暂未形成一定的研究规模。

本文将基于 PINNs 算法, 通过施加硬约束边界条件, 解决渗流模型的渗流流速计算和渗流参数反演问题。在正向问题中, 对比了两种 PINNs 方式求解渗流流速(渗流方程和达西定律耦合, 先求解水头再进行自动微分)的计算精度; 在反向问题中, 分别构造了单物理场或多物理场神经网络模型对均质或非均质渗流参数进行反演。最后, 为了验证 PINNs 算法在地下水渗流问题应用的可行性, 对均质和非均质渗流的若干算例进行分析, 表明硬约束 PINNs 算法对水头进行自动微分求解渗流流速计算精度更高, 且能够准确反演不同分布方式的渗流参数。

1 原理和算法

1.1 渗流控制方程

对于计算域为 $\Omega \subset \mathbb{R}^n$ 的稳定渗流问题, 其控制模型为^[4]

$$-\nabla^T(K\nabla H(\mathbf{x})) - W = 0 \quad (\mathbf{x} \in \Omega) \quad (1)$$

式中, ∇ 为梯度算子, $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 为空间坐标, K 为渗流参数张量(对于非均质渗流可表示为 $\mathbf{x}$ 的函

数), H 为地下水水头值, W 为源汇项。

相关的边界条件 (BC) 可以表示为

$$\mathcal{B}(H(\mathbf{x})) = 0 \quad (\mathbf{x} \in \partial\Omega) \quad (2)$$

式中, $\partial\Omega$ 为计算域的边界。基于达西定律, 可得到 $V(\mathbf{x})$ 的表达式:

$$V(\mathbf{x}) = -K \nabla H(\mathbf{x}) \quad (3)$$

1.2 PINNs 算法求解达西流速分布和渗流参数

(1) 神经网络

运用 PINNs 算法进行渗流流速求解时, 首先需构造神经网络 (常见的为前馈神经网络) 对水头或流速做近似估计。前馈神经网络由神经元连接而成, 分为输入层、隐藏层、输出层 3 部分。各神经元输入间通过相应的权重和偏置进行线性连接, 进而传输到非线性激活函数 σ 中得到其输出, 具体计算方式如下所示,

$$g(z) = \sigma(\sum_{j=1}^p w_j z_j + b) \quad (4)$$

式中, $z = [z_1, z_2, \dots, z_p]$ 为神经元的输入, $g(z)$ 为神经元输出, w_j 为输入 z_j 的所占的权重, b 为对该神经元所引入的偏置。一般地, 常见的激活函数 σ 包括 $\text{Tanh} = (e^x - e^{-x}) / (e^x + e^{-x})$ 、 $\text{ReLU} = \max\{0, x\}$ 和 $\text{Leaky ReLU} = \max\{ax, x\}$ (a 为任意常数) 等。本文所求解的控制方程, 需要计算二阶偏导数, 而 ReLU 和 Leaky ReLU 的二阶导数为 0, 因此选择能够无限次可微 Tanh 作为激活函数。

(2) 自动微分 (AD)

运用 PINNs 算法对渗流流速进行求解时, 主要利用 AD 技术计算网络输出函数对输入变量的导数。其中, AD 是一种介于符号微分和数值微分的方法, 其计算原理是将复杂解析函数分解成一系列初等运算的组合, 通过符号微分进行求导, 再代入相关数值, 通过计算机存储中间结果, 最后利用链式法则得到期望导数值^[11]。因此, 与数值微分相比, AD 能够替代网格尺度的差分运算, 避免了数值微分在求解过程中方程离散带来的计算误差, 提高了计算精度, 且自动微分的方式主要针对神经网络的输出进行, 这使梯度的计算是精确的。

目前常见的深度学习框架 TensorFlow^[28] 和 PyTorch^[29] 等都内置自动微分模块, 本文主要采用 TensorFlow2.3 进行 PINNs 计算。

(3) PINNs 算法求解达西流速

PINNs 算法进行渗流流速求解时, 有两种不同的求解策略。方法一是通过 PINNs 算法求解达西渗流定律和渗流模型的耦合方程, 其求解原理如图 1 所示, 首先构造输入为 $\mathbf{x}$, 输出为 $\hat{H}(\mathbf{x}; \theta)$ 和 $\hat{V}(\mathbf{x}; \theta)$ 的 FNN 分别对水头 H 和流速 V 做近似估计, 其中, θ 为神经网络中所有权重和偏置参数的集合。其次, 通过 AD 对式 (1), (3) 和 BC 构造如下损失函数 MSE:

$$\text{MSE} = \omega_p \text{MSE}_{\text{PDE}} + \omega_B \text{MSE}_{\text{BC}} \quad (5)$$

其中,

$$\text{MSE}_{\text{PDE}} = \frac{1}{|\tau_p|} \sum_{\mathbf{x} \in \tau_p} \left[\left\| -\nabla^T (K \nabla \hat{H}(\mathbf{x}; \theta)) - W \right\|_2^2 + \left\| \hat{V}(\mathbf{x}; \theta) + K \nabla \hat{H}(\mathbf{x}; \theta) \right\|_2^2 \right] \quad (6)$$

$$\text{MSE}_{\text{BC}} = \frac{1}{|\tau_B|} \sum_{\mathbf{x} \in \tau_B} \left\| \mathcal{B}(\hat{H}(\mathbf{x}; \theta)) \right\|_2^2 \quad (7)$$

式中: $\|\cdot\|_2$ 为欧几里得范数, τ_p , τ_B 分别为计算区域内部和边界区域中用于训练神经网络的残差点集合, 一般可以通过在定义域内随机抽样^[30] 或进行指定位置设置; ω_p , ω_B 分别为修正损失项 MSE_{PDE} , MSE_{BC} 的权重。最后, 通过 Adam^[31]、L-BFGS-B^[32] 等优化算法对损失函数进行寻优即可得到最优的神经网络参数 $\hat{\theta}$:

$$(\hat{\theta}) = \arg \min_{\theta} \text{MSE}(\theta) \quad (8)$$

同理, 方法二则是通过 PINNs 算法构造输入为 $\mathbf{x}$, 输出为 $\hat{H}(\mathbf{x}; \theta)$ 的神经网络计算地下水头分布 $\hat{H}(\mathbf{x}; \hat{\theta})$, 再基于达西定律, 利用 AD 计算求解 $\hat{V}(\mathbf{x}; \theta)$ 。值得一提的是, 由于方法二求解的控制方程不同, 此时 MSE_{PDE} 变为

$$\text{MSE}_{\text{PDE}} = \frac{1}{|\tau_p|} \sum_{\mathbf{x} \in \tau_p} \left\| -\nabla^T (K \nabla \hat{H}(\mathbf{x}; \theta)) - W \right\|_2^2 \quad (9)$$

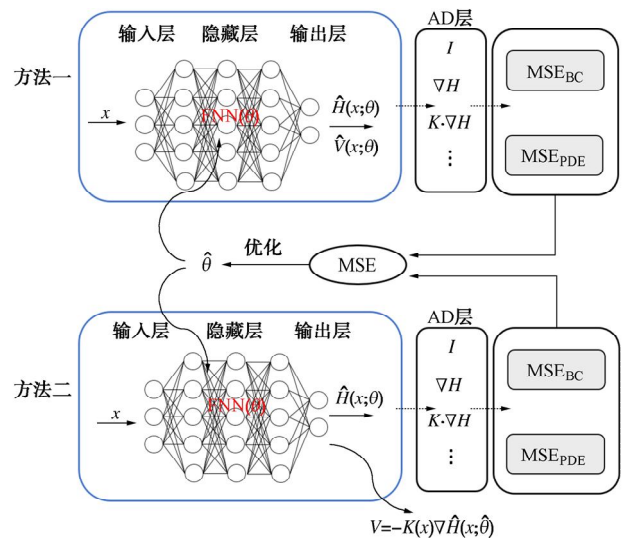


图 1 PINNs 算法求解渗流流速示意图

Fig. 1 Schematic diagram of PINNs algorithms for deriving Darcy velocity

(4) PINNs 算法识别渗流参数

PINNs 算法不仅能够实现达西流速模拟, 还能根据水头的实测或模拟数据对渗流参数进行反演, 其原理如图 2 所示。

在进行均质渗流参数反演时, 基于正向水头求解的损失函数及实测点 $\mathbf{x}_D$ 的 H 值 ($H(\mathbf{x}_D)$), 加入衡量实测数据与预测值间偏差的 MSE_{Data} 损失项及其权重

ω_D :

$$\omega_D \text{MSE}_{\text{Data}} = \omega_D \frac{1}{|\tau_D|} \sum_{x \in \tau_D} \|\hat{H}(x; \theta) - H(x_D)\|_2^2. \quad (10)$$

式中: τ_D 为实测数据组成的集合。同理, 通过优化算法对总损失函数进行优化可得到最终的神经网络参数 $\hat{\theta}$ 和均质渗流参数 $\hat{K}$ 。

PINNs 算法进行非均质渗流参数反演时, 由于 K 随 x 变化, 需要构造多物理场神经网络对渗流参数进行反演, 即构造两个神经网络 $\hat{H}(x; \theta_1)$ 和 $\hat{K}(x; \theta_2)$ 分别对水头 $H(x)$ 和 $K(x)$ 做近似估计, 此时 θ_1 和 θ_2 为各自神经网络的待优化参数, 总的神经网络参数可表示为 $\theta = \{\theta_1, \theta_2\}$ 。相应地, MSE_{PDE} 、 MSE_{BC} 和 MSE_{Data} 分别变为

$$\text{MSE}_{\text{PDE}} = \frac{1}{|\tau_P|} \sum_{x \in \tau_P} \left\| -\nabla^T (\hat{K}(x; \theta_2) \nabla \hat{H}(x; \theta_1)) - W \right\|_2^2, \quad (11)$$

$$\text{MSE}_{\text{BC}} = \frac{1}{|\tau_B|} \sum_{x \in \tau_B} \left\| \mathcal{B}(\hat{H}(x; \theta_1)) \right\|_2^2, \quad (12)$$

$$\text{MSE}_{\text{Data}} = \frac{1}{|\tau_D|} \sum_{x \in \tau_D} \left\| \hat{H}(x; \theta_1) - H(x_D) \right\|_2^2. \quad (13)$$

通过优化算法对 MSE 优化可得到最终的参数 $\hat{\theta}$:

$$(\hat{\theta}) = \arg \min_{\theta_1, \theta_2} \text{MSE}(\theta_1, \theta_2). \quad (14)$$

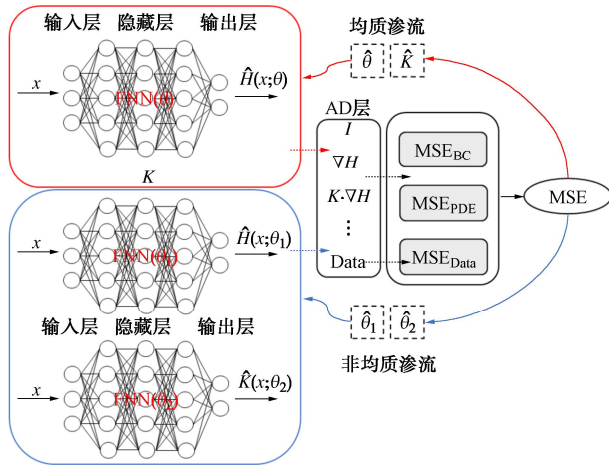


图2 PINNs 算法反演渗流参数示意图

Fig. 2 Schematic diagram of PINNs algorithms for identifying seepage parameters

(5) 硬约束 PINNs 算法

PINNs 算法进行偏微分方程求解或参数反演时, 边界的约束主要是软约束的形式, 导致计算精度会受到一定的影响。

为了解决上述问题, 可以引入合适的函数 $A(x)$ 和 $Z(x)$, 使 PINNs 计算的结果

$$u(x; \theta) = A(x) + Z(x)a^L(x; \theta). \quad (15)$$

能够自动满足边界条件^[25], 式中, $a^L(x; \theta)$ 为神经网络的输出值。以第三类边界条件为例, 假定偏微分方程满足如下边界条件:

$$\hat{B}_j^L f(x) \equiv \left[\lambda_j^L f(x) + \mu_j^L \frac{\partial f(x)}{\partial x_j} \right]_{x_j=a_j}, \quad (16)$$

$$\hat{B}_j^R f(x) \equiv \left[\lambda_j^R f(x) + \mu_j^R \frac{\partial f(x)}{\partial x_j} \right]_{x_j=b_j}. \quad (17)$$

式中: $f(x)$ 为偏微分方程的解, x_j 为 x 第 j 个空间坐标; $\hat{B}_j^L$, $\hat{B}_j^R$ 分别为 $x_j=a_j$ 和 $x_j=b_j$ 的左、右边界条件算子; λ 和 μ 为已知常数。 $A(x)$ 和 $Z(x)$ 对于第三类边界条件, 可以进行如下表示^[25]:

$$A(x) = \left[1 - \prod_{j=1}^n (1 - \mathcal{M}_j) \right] f(x), \quad (18)$$

$$Z(x) = \prod_{j=1}^n (x_j - a_j)^2 (x_j - b_j)^2, \quad (19)$$

式中, $\mathcal{M}_j$ 为作用在 $f(x)$ 的算子, 其定义为

$$\mathcal{M}_j f(x) \equiv \Phi_j^R(x_j) \hat{B}_j^L f(x) - \Phi_j^L(x_j) \hat{B}_j^R f(x), \quad (20)$$

其中,

$$\left. \begin{aligned} \Phi_j^R(x_j) &= \frac{\lambda_j^R (x_j - b_j) - \mu_j^R}{(a_j - b_j) \lambda_j^L \lambda_j^R + \mu_j^L \lambda_j^R - \lambda_j^L \mu_j^R}, \\ \Phi_j^L(x_j) &= \frac{\lambda_j^L (x_j - a_j) - \mu_j^L}{(a_j - b_j) \lambda_j^L \lambda_j^R + \mu_j^L \lambda_j^R - \lambda_j^L \mu_j^R} \end{aligned} \right\} \quad (21)$$

2 数值试验

在本节中, 将 PINNs 算法应用于二维 (空间坐标为 x, y) 均质和非均质无量纲渗流模型 (参数量纲均为 1) 求解中, 通过与解析解和实际解进行对比, 验证硬约束 PINNs 算法对于渗流流速的求解 (以 x 方向的渗流流速 V_x 为例) 和渗流参数反演的准确性。采用如下缩写方式: PINNs-H-I、PINNs-H-II 分别表示方法一、二的硬约束 PINNs 算法, PINNs-S-I、PINNs-S-II 分别表示方法一、二的软约束 PINNs 算法。所有的 PINNs 算法都使用 Python 编写, 修正的损失项权重设置为 1, 优化算法采用学习率为 0.001, 迭代步数为 20000 的 Adam 算法和 L-BFGS-B 算法, 即使用 Adam 算法迭代至指定步数后, 再通过 L-BFGS-B 进行优化, 这样能够提高计算精度和效率^[18], 其他超参数 (PINNs 算法运行之前设置的参数值) 选择如表 1 所示。

表1 算例所用的计算超参数

Table 1 Hyperparameters of PINNs algorithms used in all examples of Section 2

求解方法	τ_P	τ_B	τ_D	NN(H 或 H, V_x)	NN(K)
正向求解	2000	200	—	50×3	—
反向求解	400	200	400	100×6	60×6

注: NN 表示神经网络隐藏层结构。

为了对计算结果的精度进行分析, 采用 L_2 相对误

差 (ε) 作为评价指标:

$$\varepsilon = \frac{\|u(x_i, y_j) - \hat{u}(x_i, y_j)\|_2}{\|u(x_i, y_j)\|_2}, \quad (22)$$

式中, (x_i, y_j) 为计算域内随机分布的点, $u(x_i, y_j)$ 为解析解, $\hat{u}(x_i, y_j)$ 为 PINNs 数值计算结果。

2.1 渗流参数为均质的二维稳定流

地下水二维稳定流的水头和 V_x 的控制方程可由式 (1) 和式 (2) 得出, 令 $K_x=K_y=1$, 求解的计算区域 $\Omega=[0, 1] \times [0, 1]$, 控制方程的解析解 $H=xy(1-x)(1-y)$, 控制方程满足的 Dirichlet 条件和源汇项 W 均由解析解反推得到。

为了进行对比, 本例分析了在不同 y 条件下 ($y=0.3, 0.5, 0.9$), PINNs-H-II、PINNs-S-II、PINNs-H-I、PINNs-S-I 和 FEM 的渗流流速求解精度, 其中 FEM 采用计算域为 1024 个单元的三角形剖分方法求解 H , 再基于达西定律求解 V_x , 如图 3 所示。可以看出, 4 种 PINNs 计算方式都具有较高的计算稳定性, 这表明在不同分布的 τ_p 、 τ_B 下 (随机抽样设置), 利用优化算法进行神经网络参数寻优时, 能够迭代至较优的计算结果; FEM 的计算误差 (主要来自于划分网格时的离散误差, 以及达西渗透流速在节点处不连续导致的误差^[2]) 高于 PINNs-H-II 产生的误差 (由近似误差、泛化误差和优化误差共同决定^[23]), 低于其他 3 种 PINNs 算法所产生的误差, 表明 PINNs-H-II 可替代 FEM 作为精确求解地下水达西渗流流速的计算方法; 同时, 还注意到方法一施加硬约束对计算性能提升不明显, 且计算精度低于 PINNs-H-II, 其原因在于: 方法一的 MSE_{PDE} 是通过水头和流速控制方程同时定义的, 硬约束只对水头施加, 因此无法影响流速计算结果, 而方法二的 MSE_{PDE} 仅由水头控制方程构造, 优化过程硬约束能够对使优化算法更易迭代至全局最优, 并且 H 进行自动微分运算得到 V_x 的过程是数值准确的, 因此优化算法对损失函数进行优化时, 方法一相较于方法二会产生更大的优化误差, 且施加硬约束对计算性能提升不明显。图 4 即为采用 PINNs-H-II 求解 V_x 的计算结果, 其中图 4 (a) 为 V_x 的求解结果, 图 4 (b) 为求解结果与解析解的绝对误差, 可以看出, 采用自动微分技术能够得到连续光滑且数值准确的数值计算结果, 且绝对误差最大为 6.8×10^{-6} , 具有较高的计算精度。

为了验证 PINNs 算法在进行渗流参数反演的准确性, 假定 $K_x=K_y=1$ 是需要反演的量, 并设定残差点 $\tau_p = \tau_D = 400$ (由 x, y 方向上均匀选取 20 个点生成), 实测值则根据解析解得到。在进行计算时, 在使用方法二进行正向求解所构造损失函数的基础上增加一项关于实测值的损失函数, 利用优化算法对损失函数进

行全局寻优, 最终反演结果如图 5 所示。可以看出, 随着优化迭代步数的增加, PINNs 算法施加硬约束后能够更快收敛至准确值, 而最终 20000 步后软约束反演结果为 0.97, 需要更多的迭代步数才能收敛至准确值, 这表明, 通过施加硬约束, PINNs 算法能够提高对均质二维稳定流的渗流参数反演精度。

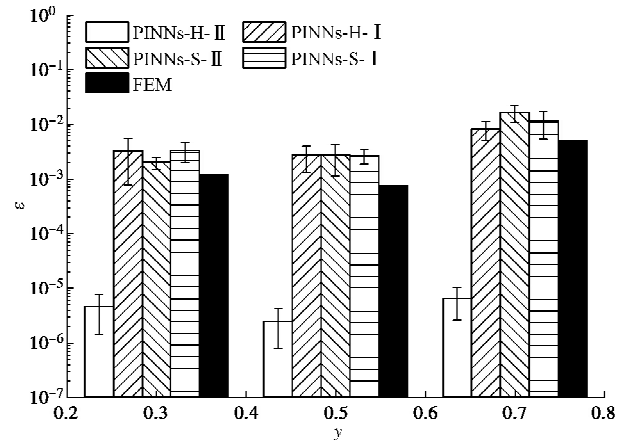


图 3 不同位置 y 下, PINNs-H-II、PINNs-S-II、PINNs-H-I、PINNs-S-I 和 FEM 所计算 V_x 的 L_2 相对误差图

Fig. 3 Relative errors of L_2 given by PINNs-H-II, PINNs-S-II, PINNs-H-I, PINNs-S-I and FEM methods under different values of y

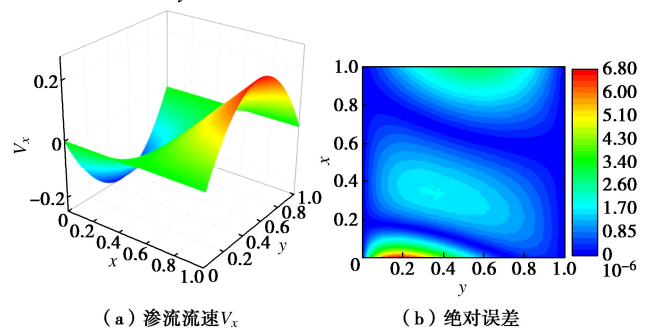


图 4 PINNs-H-II 求解均质渗流流速 V_x 的计算结果图

Fig. 4 Values of Darcy velocity V_x of homogeneous seepage and absolute errors given by PINNs-H-II method

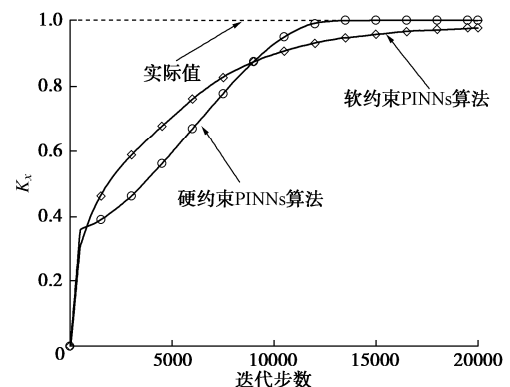


图 5 均质情况下反演渗流参数随迭代步数变化情况

Fig. 5 Variation of homogeneous seepage parameter with iteration step numbers in backward problem

2.2 渗流参数渐变的二维非均质稳定流

本例控制方程可由式(1)和式(2)得出, 令 $K_x=K_y=\frac{20}{2(1-x)(1-y)+1}$, 求解的计算区域 $\Omega=[0, 1] \times [0, 1]$, 控制方程的解析解 $H=xy(1-x)(1-y)$, 控制方程满足的 Dirichlet 条件和源汇项 W 均由解析解反推得到。

图6展示了基于两种硬约束 PINNs 算法的渗流流速的求解结果, 其中图6(a), (b)分别为 PINNs-H-I、PINNs-H-II 与解析解之间的绝对误差, 与2.1节中得到的结果类似, PINNs-H-II 求解结果的绝对误差比 PINNs-H-I 低了约3个数量级, 表明在非均质渗流条件下, PINNs-H-II 的计算精度高于 PINNs-H-I。

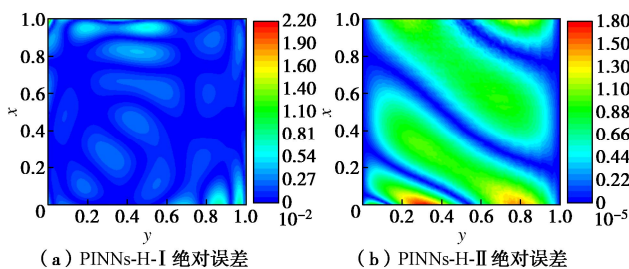


图6 PINNs 算法求解非均质渗流流速 V_x 与解析解的绝对误差云图

Fig. 6 Absolute errors of V_x of non-homogeneous seepage given by PINNs-H-I and PINNs-H-II methods

为了进一步说明软硬约束在边界的作用情况, 对比了在 $y=0$ 下, PINNs-H-II 和 PINNs-S-II 的 x 与 H 的计算关系, 如图7所示。可以看出, 硬约束 PINNs 算法的水头计算结果在边界处恒等于0, 这与实际水头相吻合, 而软约束 PINNs 算法在边界计算结果存在振荡现象, 因此可说明硬约束 PINNs 算法能够对软约束边界条件进行修正, 具有更高的计算精度。

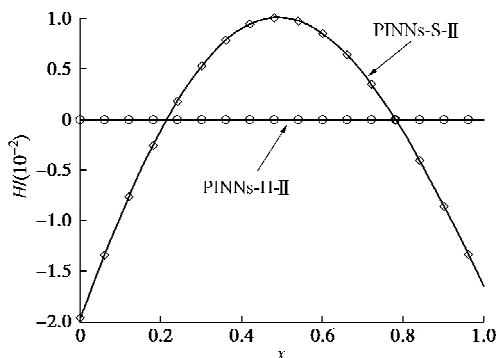


图7 $y=0$ 时, PINNs-H-II 和 PINNs-S-II 所计算的 x 与 H 的关系
Fig. 7 Relationship between x and H via PINNs-H-II and PINNs-S-II methods at $y=0$

对于渗流参数的反演, 在方法二求解水头问题的基础上, 假定 $K_x=K_y=\frac{20}{2(1-x)(1-y)+1}$ 为待反演函数,

构建多物理场神经网络模型, 求解结果如图8所示, 图8(a)为反演的渗流参数, 图8(b)为反演的渗流参数与实际解的绝对误差云图。可以看出, 最大绝对误差在 4.5×10^{-2} 附近, 因此可证明硬约束 PINNs 算法反演具有较高的计算精度, 并且从图8(c)可知, 在 $y=0.5$ 处, 硬约束 PINNs 算法能够准确反演出渗流参数随空间位置 x 的变化关系, 而软约束 PINNs 算法却无法获得准确的求解结果, 说明通过施加硬约束, 能够提高反演的准确性。

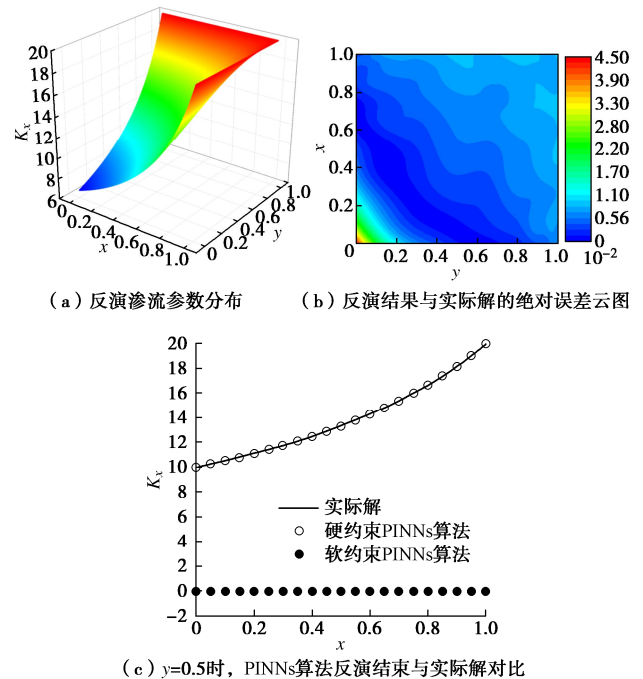


图8 非均质情况下渗流参数反演结果

Fig. 8 Identified seepage parameters and corresponding absolute errors via soft- and hard-constraint PINNs in non-homogeneous seepage

3 结 论

传统的数值算法主要是通过划分网格的方式进行地下水模型的渗流流速计算, 较难实现非均质模型的渗流参数反演, 本文将 PINNs 算法应用于地下渗流模型的正向和反向求解当中, 通过施加硬约束提高了算法的精度, 主要得到以下3点结论。

(1) 基于硬约束 PINNs 算法构造了 PINNs-H-I 和 PINNs-H-II 两种渗流流速计算方法, 分析表明, 两种计算方法都具有较高的求解稳定性, 并通过对比可知, PINNs-H-I 在进行损失函数构造时, MSE_{PDE} 需要增加求解 V 的损失项, 导致损失函数更为复杂, 因此优化算法更难收敛至最优点, 而 PINNs-H-II 仅由渗流方程确定, 进行寻优时具有更高的计算精度。

(2) 对 PINNs 算法施加硬约束可自动满足偏微

分方程的边界要求,并能够提高对渗流模型求解和渗流参数反演的精度。

(3) 针对渗流参数的分布特点,采用硬约束 PINNs 算法分布构造了单物理场或多物理场的神经网络模型,通过最小化神经网络的损失函数,对均质或非均质渗流参数进行了反演。在算例中,通过与实际的渗流参数进行对比,验证了反演参数值的准确性。

参考文献:

- [1] 谢一凡, 吴吉春, 王 益, 等. 一种模拟节点达西流速的多尺度有限元-有限元模型[J]. 岩土工程学报, 2022, 44(1): 107-114, 202. (XIE Yifan, WU Jichun, WANG Yi, et al. Multiscale finite element-finite element model for simulating nodal Darcy velocity[J]. Chinese Journal of Geotechnical Engineering, 2022, 44(1): 107-114, 202. (in Chinese))
- [2] 谢一凡, 吴吉春, 薛禹群, 等. 一种模拟节点达西渗透流速的三次样条多尺度有限单元法[J]. 岩土工程学报, 2015, 37(9): 1727-1732. (XIE Yifan, WU Jichun, XUE Yuqun, et al. Cubic-spline multiscale finite element method for simulation of nodal Darcy velocities in aquifers[J]. Chinese Journal of Geotechnical Engineering, 2015, 37(9): 1727-1732. (in Chinese))
- [3] 李守巨, 上官子昌, 刘迎曦, 等. 地下水渗流模型参数识别的模拟退火算法[J]. 岩石力学与工程学报, 2005(S1): 5031-5036. (LI Shouju, SHANGGUAN Zichang, LIU Yingxi, et al. Parameter identification procedure for groundwater flow model with simulated annealing[J]. Chinese Journal of Rock Mechanics and Engineering, 2005(S1): 5031-5036. (in Chinese))
- [4] 周凌峰, 王 媛, 冯 迪. 求解非均质渗流场的改进数值流形方法[J]. 岩土工程学报, 2021, 43(7): 1288-1296, 1377. (ZHOU Lingfeng, WANG Yuan, FENG Di. An improved numerical manifold method for solving heterogeneous seepage problem[J]. Chinese Journal of Geotechnical Engineering, 2021, 43(7): 1288-1296, 1377. (in Chinese))
- [5] YEH G T. On the computation of Darcian velocity and mass balance in the finite element modeling of groundwater flow[J]. Water Resources Research, 1981, 17(5): 1529-1534.
- [6] D'ANGELO C, SCOTTI A. A mixed finite element method for Darcy flow in fractured porous media with non-matching grids[J]. ESAIM: Mathematical Modelling and Numerical Analysis, 2012, 46(2): 465-489.
- [7] 赵文凤, 谢一凡, 吴吉春. 一种模拟节点达西渗透流速的双重网格多尺度有限单元法[J]. 岩土工程学报, 2020, 42(8): 1474-1481. (ZHAO Wenfeng, XIE Yifan, WU Jichun. A dual-mesh multiscale finite element method for simulating nodal Darcy velocities in aquifers[J]. Chinese Journal of Geotechnical Engineering, 2020, 42(8): 1474-1481. (in Chinese))
- [8] WILLIS R, YEH W W G. Groundwater systems planning and management[M]. Englewood Cliffs, NJ: Prentice-Hall, 1987
- [9] 姚磊华. 遗传算法和高斯牛顿法联合反演地下水渗流模型参数[J]. 岩土工程学报, 2005, 27(8): 885-890. (YAO Leihua. Parameters identification of groundwater flow model with genetic algorithm and Gauss-Newton Method[J]. Chinese Journal of Geotechnical Engineering, 2005, 27(8): 885-890. (in Chinese))
- [10] PINKUS A. Approximation theory of the MLP model in neural networks[J]. Acta Numerica, 1999, 8: 143-195.
- [11] GUNES B A, PEARLMUTTER BARAK A, ANDREYEVICH R A, et al. Automatic differentiation in machine learning: a survey[J]. Journal of Machine Learning Research, 2018, 18(1): 5595-5637.
- [12] HAGHIGHAT E, RAISSI M, MOURE A, et al. A deep learning framework for solution and discovery in solid mechanics[EB/OL]. 2020: arXiv: 2003.02751. <https://arxiv.org/abs/2003.02751>
- [13] 唐明健, 唐和生. 基于物理信息的深度学习求解矩形薄板力学正反问题[J]. 计算力学学报, 2022, 39(1): 120-128. (TANG Mingjian, TANG Hesheng. A physics-informed deep learning method for solving forward and inverse mechanics problems of thin rectangular plates[J]. Chinese Journal of Computational Mechanics, 2022, 39(1): 120-128. (in Chinese))
- [14] RAISSI M, PERDIKARIS P, KARNIADAKIS G E. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations[J]. Journal of Computational Physics, 2019, 378: 686-707.
- [15] CHEN Z, LIU Y, SUN H. Physics-informed learning of governing equations from scarce data[J]. Nature Communications, 2021, 12: 6136.
- [16] ZHANG Q, CHEN Y L, YANG Z Y, et al. Multi-constitutive neural network for large deformation poromechanics problem[EB/OL]. 2020: arXiv: 2010.15549. <https://arxiv.org/abs/2010.15549>
- [17] BEKELE Y W. Physics-informed deep learning for flow and deformation in poroelastic media[EB/OL]. 2020: arXiv:

- 2010.15426. <https://arxiv.org/abs/2010.15426>
- [18] 兰 鹏, 李海潮, 叶新宇, 等. PINNs 算法及其在岩土工程中的应用研究[J]. 岩土工程学报, 2021, **43**(3): 586-592. (LAN Peng, LI Haichao, YE Xinyu, et al. PINNs algorithm and its application in geotechnical engineering[J]. Chinese Journal of Geotechnical Engineering, 2021, **43**(3): 586-592. (in Chinese))
- [19] HE Q Z, BARAJAS-SOLANO D, TARTAKOVSKY G, et al. Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport[J]. Advances in Water Resources, 2020, **141**: 103610.
- [20] TARTAKOVSKY A M, MARRERO C O, PERDIKARIS P, et al. Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems[J]. Water Resources Research, 2020, **56**(5): e2019WR026731.
- [21] BANDAI T, GHEZZEHEI T A. Physics-informed neural networks with monotonicity constraints for Richardson-richards equation: estimation of constitutive relationships and soil water flux density from volumetric water content measurements[J]. Water Resources Research, 2021, **57**(2): e2020WR027642.
- [22] BANDAI T, GHEZZEHEI T A. Forward and inverse modeling of water flow in unsaturated soils with discontinuous hydraulic conductivities using physics-informed neural networks with domain decomposition[J]. Hydrology and Earth System Sciences, 2022, **26**(16): 4469-4495.
- [23] LU L, MENG X H, MAO Z P, et al. DeepXDE: a deep learning library for solving differential equations[J]. SIAM Review, 2021, **63**(1): 208-228.
- [24] SUN L N, GAO H, PAN S W, et al. Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data[J]. Computer Methods in Applied Mechanics and Engineering, 2020, **361**: 112732.
- [25] LAGARI P L, TSOUKALAS L H, SAFARKHANI S, et al. Systematic construction of neural forms for solving partial differential equations inside rectangular domains, subject to initial, boundary and interface conditions[J]. International Journal on Artificial Intelligence Tools, 2020, **29**(5): 2050009.
- [26] LU L, PESTOURIE R, YAO W J, et al. Physics-informed neural networks with hard constraints for inverse design[J]. SIAM Journal on Scientific Computing, 2021, **43**(6): B1105-B1132.
- [27] 陆至彬, 瞿景辉, 刘桦, 等. 基于物理信息神经网络的传热过程物理场代理模型的构建[J]. 化工学报, 2021, **72**(3): 1496-1503. (LU Zhibin, QU Jinghui, LIU Hua, et al. Surrogate modeling for physical fields of heat transfer processes based on physics-informed neural network[J]. CIESC Journal, 2021, **72**(3): 1496-1503. (in Chinese))
- [28] ABADI M, BARHAM P, CHEN J M, et al. TensorFlow: a system for large-scale machine learning[C]//Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation. November 2 - 4, 2016, Savannah, GA, USA. New York: ACM, 2016: 265-283.
- [29] ADAM P, GROSS S, CHINTALA S, et al. Automatic differentiation in PyTorch[C]// 31st Conference on Neural Information Processing Systems (NIPS 2017), 2017, Long Beach.
- [30] MCKAY M D, BECKMAN R J, CONOVER W J. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code[J]. Technometrics, 2000, **42**(1): 55-61.
- [31] KINGMA D P, BA J. Adam: A method for stochastic optimization[C]// 3rd International Conference on Learning Representations, 2015, San Diego.
- [32] ZHU C, BYRD R H, LU P, et al. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization[J]. ACM Transactions on Mathematical Software, 1997, **23**(4): 550-560.