

DOI: 10.11779/CJGE202103023

PINNs 算法及其在岩土工程中的应用研究

兰 鹏¹, 李海潮¹, 叶新宇¹, 张 升^{*1}, 盛岱超^{1, 2}

(1. 中南大学土木工程学院, 湖南 长沙 400041; 2. 悉尼科技大学土木与环境工程学院, 悉尼 NSW 2007)

摘要: 物理信息神经网络(PINNs)算法采用自动微分方法将偏微分方程直接嵌入神经网络中, 从而实现对偏微分方程的智能求解, 属于一种新型的无网格算法, 具有收敛速度快和计算精度高等优点。PINNs不仅能够实现对偏微分方程求解, 还能够对偏微分方程未知参数进行反演, 因此对岩土工程复杂问题具有广泛的应用前景。为了验证 PINNs 算法在岩土工程领域的可行性, 对连续排水边界条件下的一维固结理论进行求解和界面参数反演。计算结果表明, PINNs 数值结果与解析解具有高度一致性, 且界面参数反演结果准确, 说明 PINNs 算法能够为岩土工程相关问题提供新的求解思路。

关键词: 物理信息神经网络(PINNs); 自动微分; 无网格算法; 参数反演; 连续排水边界条件

中图分类号: TU43 **文献标识码:** A **文章编号:** 1000-4548(2021)03-0586-07

作者简介: 兰 鹏(1994—), 男, 硕士研究生, 主要从事岩土工程智能化方面的研究工作。E-mail: lanpeng@csu.edu.cn。

PINNs algorithm and its application in geotechnical engineering

LAN Peng¹, LI Hai-chao¹, YE Xin-yu¹, ZHANG Sheng¹, SHENG Dai-chao^{1, 2}

(1. School of Civil Engineering, Changsha Central South University, Changsha 400041, China; 2. School of Civil and Environmental Engineering, Sydney University of Technology, Sydney, NSW 2007, Australia)

Abstract: The physical information neural networks (PINNs) algorithm, a new mesh-free algorithm, uses the automatic differential method to embed the partial differential equation directly into the neural networks so as to realize the intelligent solution of the partial differential equation, which has the advantages of fast convergence speed and high computational accuracy. The PINNs algorithm has a promising application in geotechnical engineering because it can solve the complex partial differential equations (PDEs) and inverse the unknown parameters of the PDEs. In order to verify the feasibility of the PINNs algorithm in geotechnical engineering, the one-dimensional consolidation process with the continuous drainage boundary condition is taken as an example to illustrate the procedures of the PINNs algorithm in terms of both the forward and inverse problems. The results show that the PINNs solution is highly consistent with the analytical one, indicating that the PINNs algorithm can provide an alternative approach for solving the related problems in geotechnical engineering.

Key words: physical information neural networks; automatic differentiation; mesh-free algorithm; parameter inversion; continuous drainage boundary condition

0 引言

岩土工程是一门存在复杂非线性关系的学科, 其中往往涉及对偏微分方程(partial differential equation, PDE)的求解。目前对PDE求解多采用有限差分法(FDM)、有限单元法(FEM)等数值方法。这些数值方法在求解复杂偏微分方程时常常需要将计算进行离散化处理, 模型计算效率和精度与网格的密度和计算步长密切相关。

近些年发展起来的基于深度学习的物理信息神经网络(physical information neural networks, PINNs)算法则基于新的求解思路: 采用自动微分方法将偏微分方程嵌入到神经网络的损失函数中, 从而实现对不同

类型的偏微分方程准确求解, 并且计算精度与计算步长无关。因此, 与传统的数值方法相比, 对于高维度的PDE问题, 其求解过程与结果将不再受到限制。此外, PINNs对于积分微分方程(IDEs), 分数阶微分方程(FDEs)和随机微分方程(SDEs)同样能进行准确求解。同时, PINNs算法也可以对偏微分方程反向问题进行求解, 即利用实测数据来推算出控制方程

基金项目: 国家自然科学基金优秀青年基金项目(51722812); 湖湘高层次人才聚集工程项目(2018RS3016); 湖南省研究生科研创新项目(CX20200220); 中南大学研究生自主探索创新项目(2020zzts613)

收稿日期: 2020-08-06

*通信作者(E-mail: zhang-sheng@csu.edu.cn)

的未知系数或源项。目前, PINNs 算法已开始用于求解经典的偏微分方程。如: Raissi 等^[1]对 Burgers 和 Schrödinger 方程进行了准确求解, 并对 Navier-Stokes 方程进行参数反演。Lu 等^[2]提出了基于残差的自适应细化 (RAR) 方法和构造几何方法, 改善了在复杂计算域 PINNs 算法的训练效果。由于 PINNs 算法对 PDE 求解的精确性和特殊的参数反演方式, 已经开始应用于计算流体力学^[3-5]、超材料设计^[6]和反应扩散系统^[7]等领域。但在同样需要求解 PDE 问题的岩土工程领域, PINNs 算法的探究与应用仍处于初步阶段, 暂未形成趋势和一定的研究规模。

因此, 为了探讨新算法在岩土工程领域中应用的可能性, 本文以连续排水边界条件下的一维固结问题为例, 推导了基于 PINNs 算法的求解过程, 通过对比解析解与 FDM 数值解, 验证了 PINNs 方法在岩土工程领域应用的可行性。在此基础上, 分析了神经网络结构和修正损失函数的权重对 PINNs 计算精度的影响。最后, 利用 PINNs 算法对反演问题特殊的求解方式, 探讨了连续排水条件一维固结方程中 PINNs 在未知界面参数反演中的应用。

1 PINNs 算法

1.1 前馈神经网络

PINNs 算法采用自动微分方法将 PDE 嵌入于神经网络中, 通过约束损失函数, 对方程进行求解。其中, 前馈神经网络 (feedforward neural network, FNN) 算法简单, 网络层次清晰, 对于绝大多数的 PDE 问题都能够进行准确求解。FNN 结构如图 1 所示, 主要由输入层、隐藏层和输出层 3 部分组成, 在每一层中包括层内无连接、层间全连接的神经元。FNN 在进行计算时是由输入层依次计算到输出层, 其中每个节点的计算方法如下:

$$x_j = \sigma(W_{i-1}X_{i-1} + b_{i-1}) \quad , \quad (1)$$

式中, x_j 为第 i 层第 j 个神经元的值, W_{i-1} 为第 $i-1$ 层到第 i 层的第 j 个神经元的权重向量, X_{i-1} 为第 $i-1$ 层所有神经元的值组成的向量, b_{i-1} 为第 $i-1$ 层的偏置, σ 为隐藏层激活函数。

激活函数主要用于增强网络表示能力和学习能力, 目前常见的激活函数包括 sigmoid 函数, tanh 函数, ReLU 函数, Leaky ReLU 函数。在 PINNs 中, 通常采用能够无限可微的 tanh 函数作为激活函数, tanh 函数定义为

$$\sigma(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad . \quad (2)$$

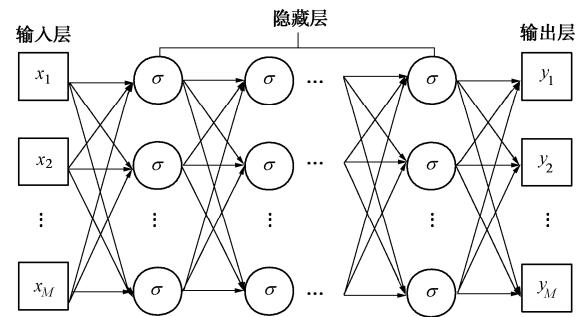


图 1 FNN 结构图

Fig. 1 Structural diagram of FNN

1.2 PINNs 算法求解偏微分方程

根据一般定义, 在计算域 $\Omega \subset \mathbb{R}^d$ 中, 带有约束条件的偏微分方程可以表示为以下形式:

$$f\left(x; \frac{\partial u}{\partial x_1}, \dots, \frac{\partial u}{\partial x_d}, \frac{\partial^2 u}{\partial x_1 \partial x_1}, \dots, \frac{\partial^2 u}{\partial x_1 \partial x_d}; \dots; \lambda\right) = 0 \quad (x \in \Omega) \quad . \quad (3)$$

式中, $x = (x_1, x_2, \dots, x_d)$ 和 $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_d)$ 分别为偏微分方程的变量和参数。 $u(x)$ 为边界条件 $B(u, x) = 0$ 所对应的偏微分方程的解。需要注意的是, PINNs 算法将时间 t 认为是 x 的特殊分量, 即时域是一种特殊的计算域; 偏微分方程的初始条件视作第一类边界条件的特殊形式以简化模型。

在 PINNs 算法求解过程中, 首先需要构建前馈神经网络 $u_{NN}(x; \theta)$ 作为偏微分方程解 $u(x)$ 的近似, 其中, 参数 θ 为权重矩阵和偏置向量的组合。其次 u_{NN} 关于 x 的偏导数采用 TensorFlow^[8] 或 PyTorch^[9] 内置的自动微分方法进行简单求解。

在计算域内部及边界上选取若干残差点, 并定义为 τ_f 和 τ_b 。这些残差点的初始位置和值的分布满足随机分布, 并将其作为模型的训练数据集。由此可以得到损失函数 $\mathcal{L}(\theta; \tau)$ 的表达式:

$$\mathcal{L}(\theta; \tau) = w_f \mathcal{L}_f(\theta; \tau_f) + w_b \mathcal{L}_b(\theta; \tau_b) \quad , \quad (4)$$

式中, w_f , w_b 分别为修正由主控方程以及边界条件构造的损失函数的权重, 在计算时可以根据损失值反馈结果对权重进行调整, 当所有的损失值在一个数量级时, 可以将权重的值设为 1。 $\mathcal{L}_f(\theta; \tau_f)$ 和 $\mathcal{L}_b(\theta; \tau_b)$ 分别为主控方程及其边界条件所对应的损失函数。

在此基础上, 采用梯度迭代优化器对损失函数进行约束, 得到参数 θ , 此时损失函数 $\mathcal{L}(\theta; \tau)$ 达到最小值。目前, 常见的梯度迭代优化器有 SGD 算法^[10]、Adam 算法^[11]和 L-BFGS 算法^[12]等。其中 Adam 算法当神经网络训练达到设定的迭代次数时会终止, 默认其计算精度会受到迭代次数的影响, 但收敛速度快, 主要用于训练样本比较大的情况; L-BFGS 算法则是

当损失函数的相对变化小于设定值时终止迭代, 对于较少的训练数据能够表现出优越的性能, 但是其收敛速度相较于 Adam 算法具有较大差距。为了实现 PINNs 算法计算效率和计算精度的平衡, 本文采用 Adam 算法和 L-BFGS 算法两种迭代优化器组合方法, 即首先使用 Adam 算法将损失函数最小化, 然后再使用 L-BFGS 算法进行训练, 可有效避免使用一种迭代优化算法带来的精度不足或迭代速度慢等局限性。

1.3 PINNs 算法的参数反演

PINNs 算法在进行参数反演时, 根据实测数据 τ_i , 对偏微分方程中带有的未知参数 λ 进行计算, 需要构造与求解偏微分方程时不同的损失函数 $\mathcal{L}(\theta, \lambda; \tau)$:

$$\mathcal{L}(\theta, \lambda; \tau) = w_f \mathcal{L}_f(\theta, \lambda; \tau_f) + w_b \mathcal{L}_b(\theta, \lambda; \tau_b) + w_i \mathcal{L}_i(\theta, \lambda; \tau_i) \quad (5)$$

式中, $\mathcal{L}_i(\theta, \lambda; \tau_i)$ 为基于实测点所组成的损失函数, w_i 为修正该损失函数的权重。

基于以上原理, 将待求未知参数的 PDE 嵌入 PINNs 算法的损失函数中, 通过约束损失函数进行求解, 具体的求解流程如图 2 所示。

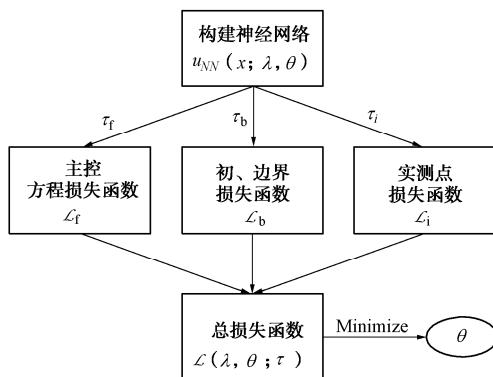


图 2 PINNs 参数反演计算过程

Fig. 2 Calculation process of PINNs parameter inversion

PINNs 算法首先构建神经网络, 针对特定问题, 随机选取残差点, 与实测点一起构建损失函数, 再进行加权形成总的损失函数, 最后通过梯度迭代优化器约束损失函数, 求解未知参数 θ 和 λ 。

2 算例分析

为了验证 PINNs 算法在岩土工程中应用, 本文选取连续排水边界条件下的无量纲化一维固结方程为算例进行分析。

Terzaghi 一维固结理论是土力学的经典命题, 针对该命题, 梅国雄等^[13]认为土体边界孔压 u 会随着时间 t 呈指数衰减, 并且假定如下连续排水边界条件:

$$u(0, t) = q \exp(-bt) \quad (6)$$

式中, q 为外加瞬时荷载, b 为非负的界面参数, 用于

表征土体界面的透水性能。

在此基础上, 冯健雪^[14]给出了该固结理论的无量纲表达形式:

$$\frac{\partial \bar{u}}{\partial T_v} = \frac{\partial^2 \bar{u}}{\partial Z^2} \quad (7)$$

$$\bar{u}(Z, 0) = 1 \quad (8)$$

$$\bar{u}(0, T_v) = \exp(-\alpha T_v) \quad (9)$$

$$\bar{u}(1, T_v) = \exp(-\beta T_v) \quad (10)$$

式中, $\bar{u} = u/q$, $Z = z/H$ 和 $T_v = c_v t/H^2$ 分别为无量纲化的超孔隙水压力、土体深度和时间因子。 $\alpha = bH^2/c_v$, $\beta = bH^2/c_v$ 为界面参数, c_v 为土体固结系数, 界面参数用于控制在土体深度为 $Z=0$ 和 $Z=1$ 时, 土体边界孔压 \bar{u} 随着时间因子 T_v 的衰减速率。式 (7) ~ (10) 一起组成了连续排水边界条件下无量纲化控制方程。

其归一化超孔隙水压 \bar{u} 的解析解可以表示为

$$\bar{u}(Z, T_v) = \exp(-\alpha T_v) + Z[\exp(-\beta T_v) - \exp(-\alpha T_v)] + 2 \sum_{n=1}^{\infty} \frac{\sin(n\pi Z)}{n\pi} \left\{ \frac{\alpha}{(n\pi)^2 - \alpha} [\exp(-\alpha T_v) - \exp(-(n\pi)^2 T_v)] - \frac{(-1)^n \beta}{(n\pi)^2 - \beta} [\exp(-\beta T_v) - \exp((n\pi)^2 T_v)] \right\}, \quad (11)$$

式中, n 为有限傅里叶正弦变换的级数项。

2.1 PINNs 算法求解及界面参数反演

PINNs 在进行连续排水边界条件下的无量纲化一维固结理论的求解时, 构造的主控方程损失函数 $\mathcal{L}_f(\theta; \tau_f)$ 的具体表达式:

$$\mathcal{L}_f(\theta; \tau_f) = \frac{1}{|\tau_f|} \sum \left\| f \left(Z, T_v; \frac{\partial \bar{u}_{NN}}{\partial T_v}; \frac{\partial^2 \bar{u}_{NN}}{\partial Z^2} \right) \right\|_2^2. \quad (12)$$

同理, 由边界条件和初始条件, 可以得到损失函数的 $\mathcal{L}_b(\theta; \tau_b)$ 具体表达式:

$$\mathcal{L}_b(\theta; \tau_b) = \sum_{i=1}^3 \frac{1}{|\tau_b|} \sum_{x \in \tau_b} \| B(\bar{u}_{NN}, Z, T_v) \|_2^2, \quad (13)$$

式中, $B_1(\bar{u}_{NN}, Z, T_v) = \bar{u}_{NN}(Z, 0) - 1$ 为控制方程的初始条件, $B_2(\bar{u}_{NN}, Z, T_v) = \bar{u}_{NN}(0, T_v) - \exp(-\alpha T_v)$ 和 $B_3(\bar{u}_{NN}, Z, T_v) = \bar{u}_{NN}(1, T_v) - \exp(-\beta T_v)$ 为控制方程的边界条件。

如图 3 所示, 即首先构建以 Z, T_v 为输入值, \bar{u} 为输出值的前馈神经网络 $FNN(Z, T_v; \theta)$, 并以此得到控制方程解的近似, 其次在控制方程的主控方程、初始条件和边界条件取一定量的残差点 τ_f 和 τ_b , 以此得到损失函数的具体表达式, 最后采用梯度优化算法计算该损失函数最小值, 从而得到参数 θ 。

对于连续排水边界条件下的一维固结理论中界面参数 α 和 β ，由固结土体与其相邻土体的渗透性、压缩模量及相对厚度等决定，在实际工程中，通过测量固结土体与相邻土体的渗透性确定，但是通常渗透系数会随着固结时间而发生变化，因此需要对界面参数进行实时测量。PINNs 能够基于实测数据 τ_i 进行参数反演，从而更为准确地预测建筑物后续的沉降量。其中 $\bar{u}(Z, T_v)$ 在实测数据点上满足 $\Gamma(\bar{u}, Z, T_v) = 0$ 。进行界面参数反演时所对应的损失函数 $\mathcal{L}(\theta, \alpha, \beta; \tau_i)$ 为

$$\begin{aligned} \mathcal{L}(\theta, \alpha, \beta; \tau_i) = & \frac{w_f}{|\tau_f|} \sum_{x \in \tau_f} \left\| f \left(Z, T_v, \frac{\partial \bar{u}_{NN}}{\partial T_v}, \frac{\partial^2 \bar{u}_{NN}}{\partial Z^2} \right) \right\|_2^2 + \\ & w_b \sum_{i=1}^3 \frac{1}{|\tau_b|} \sum_{x \in \tau_b} \| B(\bar{u}_{NN}, \alpha, \beta, Z, T_v) \|_2^2 + \\ & \frac{w_i}{|\tau_i|} \sum_{x \in \tau_i} \| \Gamma(\bar{u}_{NN}, Z, T_v) \|_2^2 \quad . \end{aligned} \quad (14)$$

通过对损失函数进行约束，就能够求得参数 θ ， α 和 β 。

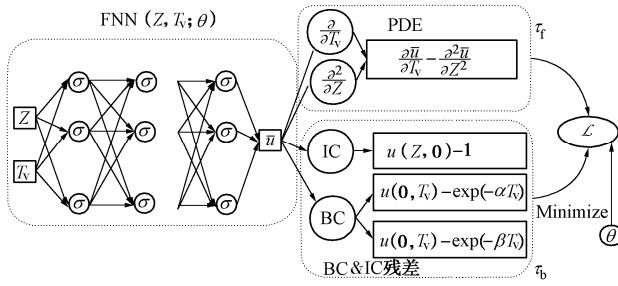


图 3 PINNs 求解连续排水边界计算过程

Fig. 3 Calculation process of continuous drainage boundary solved by PINNs

2.2 求解结果

在数值求解过程中，设置界面参数 $\alpha = 2$ 和 $\beta = 3$ ，神经网络隐藏层的个数为 3，每个隐藏层包含 50 个神经元、1 个 \tanh 激活函数和一个偏置项，Adam 迭代步数为 20000。此外，位于计算域和边界上的残差点的数目分别为 2450 和 240。

基于 PINNs 算法得到的数值解如图 4 所示，在排水条件下，土体内部在初始阶段所形成了超孔隙水压会逐渐消散；其在土体下界面的消散速率明显大于上界面($\alpha < \beta$)。数值解与解析解之间的绝对误差如图 5 所示，其最大值约为 0.005，主要出现在初始时刻且位于土体上下界面；随着超孔隙水压的逐渐消散，绝对误差逐渐趋近于 0。在不同时刻超孔隙水压力 \bar{u} 随土体深度 Z 的变化规律如图 6 所示，在不同的时间因子 ($T_v = 0.1, 0.3, 0.5, 0.7, 0.9$) 条件下，PINNs 数值解都能够与实际解析解值相吻合，且通过与空间步长为 50 和时间步长为 1000 的 FDM 数值方法进行了对

比，结果表明，PINNs 数值解已经能够达到较密的空间网格的 FDM 精度。算例求解表明，PINNs 能够对连续排水条件下的一维固结方程进行准确的求解。

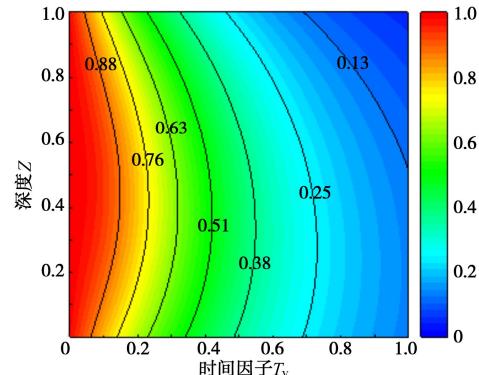


图 4 PINNs 数值解三维云图

Fig. 4 3D cloud images for numerical solution of PINNs

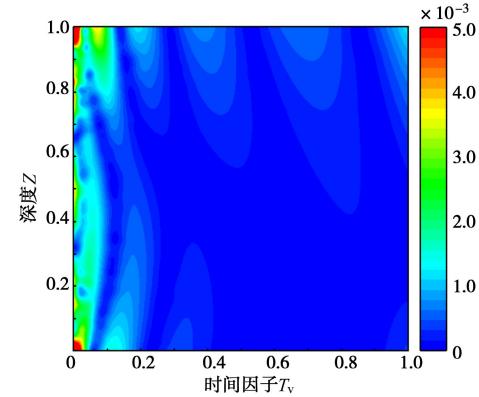


图 5 PINNs 数值解与解析解的绝对误差

Fig. 5 Absolute errors of numerical solution of PINNs and analytical solution

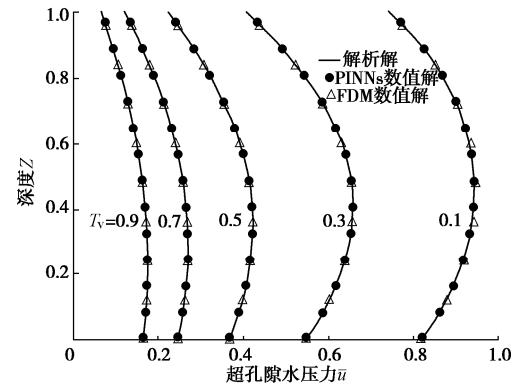


图 6 不同时间因子下的 PINNs 数值解与解析解及 FDM 对比

Fig. 6 Comparison between numerical and analytical solutions of PINNs with different time factors and FDM

在进行界面参数反演时，需要基于实测数据构建损失函数。本文从 Chai 等^[15]设计的试验中收集了 42 组实测数据 τ_i ，并随机选取 2000 个残差点 τ_f 和 200

个残差点 τ_b 构造损失函数。同样的, 设置神经网络的层数为 3, 神经元个数为 50。最终, 界面参数 α 和 β 随计算步数的演化结果如图 7 所示, 求得的界面参数 α 和 β 的反演值分别为 0.999 和 0.741。将反演的 α 和 β 与试验数据进行对比, 结果如图 8 所示, 可以看出, 基于 PINNs 进行界面参数反演的求解结果与试验值的拟合程度较高。其中, α 和 β 的 R^2 为 0.984 和 0.997, 证实了基于 PINNs 进行参数反演的可靠性。

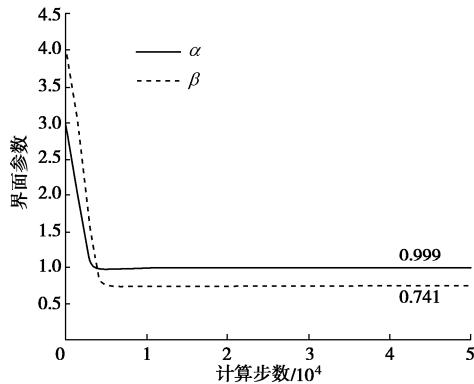


图 7 界面参数反演随计算步数变化情况

Fig. 7 Change of parameter inversion with calculation steps

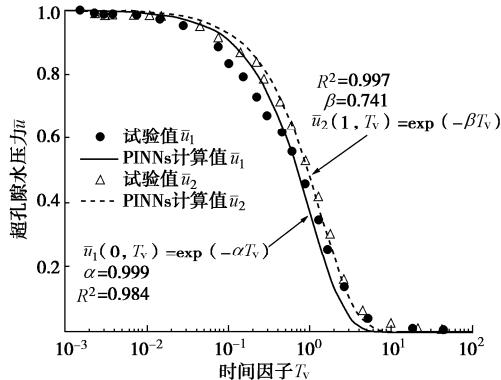


图 8 孔隙水压力的界面参数拟合效果与试验数据结果对比

Fig. 8 Comparison between fitting effect of interface parameters of pore water pressure and experimental data

3 PINNs 计算参数分析

3.1 损失值变化

对于 PINNs 算法本身, 在计算过程中, 模型的损失值会随计算步数的增加, 而逐渐趋于稳定, 当损失值为 0 时, 则可以认为, PINNs 的数值计算结果能够与解析解完全吻合。在以上求解偏微分方程的算例中, 损失值随计算步数的变化规律如图 9 所示, 其中 \mathcal{L}_{PDE} 为主控方程的损失值, \mathcal{L}_{IC} , \mathcal{L}_{BC1} , \mathcal{L}_{BC2} 为初始条件和边界条件对应的损失值, \mathcal{L} 为总的损失值, 可以根据式 (5) 计算得到, 可以看出, PINNs 算法具有收敛

快且稳定性高等优点。同时可知, 在本文的案例中, 开始迭代时, 各个损失值在区间 $(10^{-1}, 10^0)$ 之间, 因此将权重设置为 1。

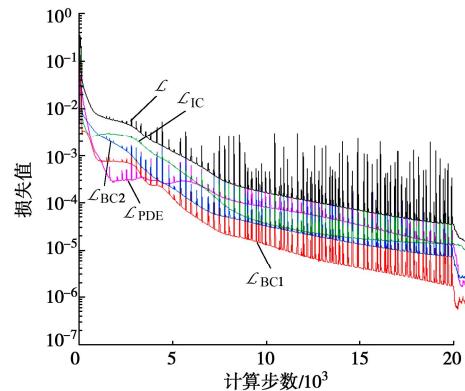


图 9 损失值随计算步数关系

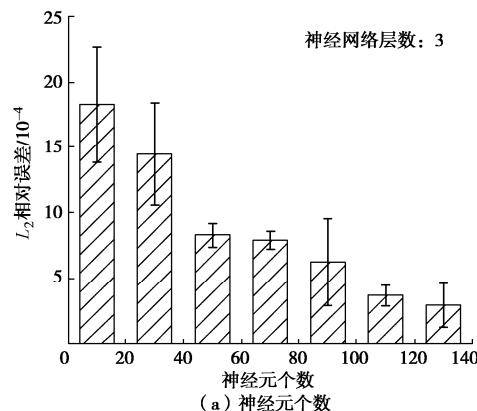
Fig. 9 Relationship between loss value and number of calculation steps

3.2 超参数分析

超参数是指在算法运行之前设置的参数值, 如神经网络层数和神经网络中每层神经元的个数等。通常, 超参数会对求解结果产生重要影响。因此, 利用 PINNs 算法进行求解时, 超参数的合理设置是准确求解的关键。一般情况下, 神经网络结构越复杂, 计算精度越高, 但是计算效率会降低。为了分析神经网络结构对最终结果的影响, 以连续排水边界条件下一维固结理论计算为例, 选择 L_2 相对误差作为评价指标, 其定义为

$$L_2 = \frac{\|y_i - f(x_i)\|_2}{\|y_i\|_2}, \quad (15)$$

式中, $\|\cdot\|_2$ 为欧几里得范数, y_i 为实际值, $f(x_i)$ 为预测值。最终结果如图 10 (a), (b) 所示。计算结果表明, 当神经网络层数或神经元个数较小时误差最大, 但随着增加神经元个数或神经网络层数逐渐增大, 计算误差在减小, 最终会趋于稳定。通过合理的选定神经网络隐含层及其包含的神经元个数, 可以同时满足计算效率和精度的要求。



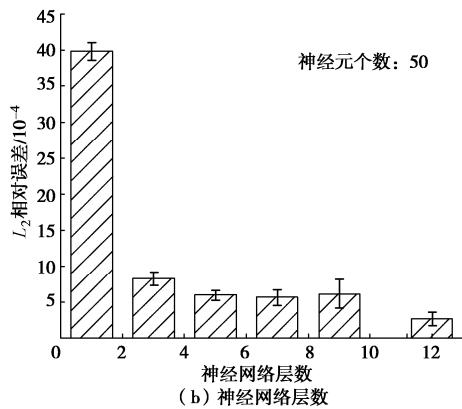


图 10 超参数对 PINNs 数值解精度的影响

Fig. 10 Influence of hyperparameters on precision of numerical solution of PINNs

权重可以对构造的损失函数进行修正, 因此对计算精度会有一定的影响。在算例中, 不同权重的 L_2 相对误差如表 1 所示, 可以看出, 权重改变时, 最终的计算精度将会发生改变, 当权重的数量级相差较大时, PINNs 算法的计算精度会减小, 因此进行超参数调整时应该注意权重的选取。

表 1 不同权重下的 L_2 相对误差Table 1 Relative error of L_2 under different weighting factors

w_f	w_b			
	0.1	1	100	1000
0.1	0.0029	0.0024	0.0018	0.0035
1	0.0027	0.0008	0.0007	0.0019
100	0.0092	0.0029	0.0019	0.0005
1000	0.0402	0.0102	0.0020	0.0018

4 讨 论

PINNs 算法能够为岩土工程中的偏微分方程求解提供一种新的求解思路, 有许多优点, 但是存在的问题同样值得重视。首先在计算效率方面, PINNs 算法相对于 FDM 和 FEM 低。其次对于时域较长的情况, 可能会出现不收敛, 虽然可以通过多台计算机配合计算, 但是会增加计算成本。再次, PINNs 算法在边界条件上施加的是软约束, 对于绝大多数的复杂域和任何类型的边界条件都能够适用, 而对于简单边界条件可以施加硬约束, 具体可以引入一个平滑距离函数 $\hat{d}(x)$, 通过将 $\hat{d}(x)$ 与原始的神经网络相乘, 获得改进后的输出, 该输出能够自动满足边界条件。但是对于一些较为复杂的边界条件, PINNs 算法对于硬约束的施加存在一定的困难, 这将导致 PINNs 算法在计算时与实际的边界存在一定的误差。最后超参数的选择往往通过经验获得, 在目前的研究中许多学者提出了一种元学习功能进行超参数的调整^[16], 但是还尚未形成成熟的研究体系。

PINNs 算法在偏微分方程求解以及参数的反演有广阔的应用前景, 但是目前在岩土领域中对于 PINNs 的研究还尚未形成一定的研究规模。本文通过一个简单的算例证明了 PINNs 算法在岩土工程中应用的可能性, 但是对于更为复杂的 PDE 方程, PINNs 算法的求解还存在一定问题, 这将是笔者接下来所要研究的重点。

5 结 论

本文将 PINNs 算法应用于复杂非线性关系的岩土工程中, 并选取了一个算例进行分析, 主要结论如下:

(1) PINNs 算法作为一种新型的无网格智能算法, 具有计算精度高且收敛速度快等优点, 其核心思想是采用自动微分方法将偏微分方程嵌入于神经网络的损失函数中。

(2) PINNs 算法在连续排水边界条件下一维固结理论的求解结果与解析解和 FDM 数值解具有高度的一致性。此外, 通过参数敏感性分析表明, 随着神经网络的层数以及神经元个数的增加, 计算精度会逐渐提高, 最终趋于稳定。

(3) 同时, PINNs 能够对偏微分方程的未知参数进行反演。在算例中, 通过对界面参数进行反演并与试验数据进行对比, 结果表明, 反演参数值与试验值具有高度吻合性, 可以为软土地基固结沉降预测提供新的思路, 在岩土工程领域中具有非常广阔的应用前景。

参 考 文 献:

- [1] RAISSI M, PERDIKARIS P, KARNIADAKIS G E, et al. 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.
- [2] LU L, MENG X, MAO Z, et al. DeepXDE: A deep learning library for solving differential equations[J]. ArXiv Preprint ArXiv:1907.04502, 2019.
- [3] 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.
- [4] MAO Z, JAGTAP A D, KARNIADAKIS G E. Physics-informed neural networks for high-speed flows[J]. Computer Methods in Applied Mechanics and Engineering,

- 2020, **360**: 112789.
- [5] RAISSI M, KARNIADAKIS G E. Hidden physics models: machine learning of nonlinear partial, differential equations[J]. *Journal of Computational Physics*, 2018, **357**: 125 – 141.
- [6] FANG Z, ZHAN J. Deep physical informed neural networks for metamaterial design[J]. *IEEE Access*, 2019, **8**: 24506 – 24513.
- [7] PANG G, LU L, KARNIADAKIS G E, et al. FPINNs: Fractional physics-informed neural networks[J]. *SIAM Journal on Scientific Computing*, 2019, **41**(4): A2603 – A2626.
- [8] ABADI M, BARHAM P, CHEN J, et al. TensorFlow: a system for large-scale machine learning[C]// 12th USENIX Symposium on Operating Systems Design and Implementation, 2016, Savannah: 265 – 283.
- [9] 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.
- [10] ROBBINS H, MONRO S. A stochastic approximation method[J]. *Annals of Mathematical Stats*, 1951, **22**(3): 400 – 407.
- [11] KINGMA D P, BA J. Adam: a method for stochastic optimization[C]// 3rd International Conference on Learning Representations, 2015, San Diego.
- [12] BYRD R H, LU P, NOCEDAL J, et al. A limited memory algorithm for bound constrained optimization[J]. *SIAM Journal on Scientific Computing*, 1995, **16**(5): 1190 – 1208.
- [13] 梅国雄, 夏君, 梅岭. 基于不对称连续排水边界的太沙基一维固结方程及其解答[J]. *岩土工程学报*, 2011, **33**(1): 28 – 31. (MEI Guo-xiong, XIA Jun, MEI Ling. Terzaghi's one-dimensional consolidation equation and its solution based on asymmetric continuous drainage boundary[J]. *Chinese Journal of Geotechnical Engineering*, 2011, **33**(1): 28 – 31. (in Chinese))
- [14] 冯健雪. 连续排水边界条件下成层地基一维固结理论研究[D]. 南宁: 广西大学, 2019. (FENG Jian-xue. Studies on One-dimensional Consolidation Theory for Layered Soils under Continuous Drainage Boundary Conditions[D]. Nanning: Guangxi University, 2019. (in Chinese))
- [15] CHAI J, MATSUNAGA K, SAKAI A, et al. Comparison of vacuum consolidation with surcharge load induced consolidation of a two-layer system[J]. *Géotechnique*, 2009, **59**(7): 637 – 641.
- [16] FINN C, ABBEEL P, LEVINE S, et al. Model-agnostic meta-learning for fast adaptation of deep networks[C]// 34th International Conference on Machine Learning, 2017, Sydney: 1126 – 1135.