Multiscale Science and Engineering
https://doi.org/10.1007/s42493-024-00106-w

REVIEW

A Review of Physics Informed Neural Networks for Multiscale Analysis and Inverse Problems

Dongjin Kim[1] · **Jaewook Lee**[1]

Received: 13 November 2023 / Revised: 22 January 2024 / Accepted: 24 January 2024

Abstract
This paper presents the fundamentals of Physics Informed Neural Networks (PINNs) and reviews literature on the methodology and application of PINNs. PINNs are universal approximators that integrates physical laws that can be described by partial differential equations (PDEs) and given data, in the learning process. The formulations of PINNs are first presented in an example of linear elasticity problem. Then, the characteristics of PINNs are compared with that of conventional numerical analysis approach. A review of the literature on PINNs for solving not only forward, but also inverse problems is discussed. In addition, special attention is given to the employment of PINNs in multiscale analysis.

Keywords Physics informed neural networks · Forward problem · Inverse problem · Multiscale analysis

Introduction

Physics-Informed Neural Networks (PINNs) aims to integrate physical models and given data in the training process of a neural network. By informing the network of physical principles, it can be expected to achieve more accurate predictions. PINNs leverage the strength of neural networks and the governing principles of physics. This hybrid approach integrates prior physical laws, which can be expressed through differential equations, directly into the architecture of a neural network. More specifically, a physics-informed regularization term is added as additional loss function, which ensures that the predictions made by the network not only fit the given data but also obey the laws of physics that govern the system under investigation. Since its first introduction by Raissi et al. [1], PINNs have been actively studied as a powerful numerical tools for predicting complex responses in various fields such as solid mechanics, thermal fluid, and electrodynamics [2].

Although PINNs present substantial progress, they encounter limitations, especially in managing highly complex and non-linear systems [3], and inefficiently solving forward problems [4]. Their key strength lies in addressing ill-posed inverse problems, especially when data is sparse [5] or noisy [1], enabling the extraction of meaningful insights from challenging datasets [6]. PINNs find diverse applications across fields like fluid dynamics [6], material science [7], and biomechanics [8], showcasing their adaptability. Additional aspect of their versatility may be the incorporation of multiscale expansion, which allows these networks to operate effectively across different scales and domains.

This paper aims to review the literature on PINNs, focusing on the application into inverse problems and multi-scale analyses. In "Fundamentals of Physics Informed Neural Networks (PINNs)" section, the fundamental formulations of PINNs are provided in the example of linear elasticity problem. In addition, the characteristics of PINNs and finite element methods (FEM) are compared to clarify the pros and cons of PINNs. "Forward Analysis Using PINNs" section reviews the studies on PINNs that have been developed as a numerical analysis tool for solving forward problems. The literature is reviewed in terms of PINN methodologies such as loss function formulation, neural network architecture and regularization schemes. Then, the studies for PINNs to solve fluid and solid mechanics problems are summarized. "Inverse Problems with PINNs" section reviews the studies for PINNs to solve inverse problems. Here, the literature on PINNs for inverse problems is

✉ Jaewook Lee
jaewooklee@gist.ac.kr

Dongjin Kim
dongjinkim@gm.gist.ac.kr

1 School of Mechanical Engineering, Gwangju Institute of Science and Technology, 123 Cheomdangwagi-ro, Buk-gu 61005, Gwangju, South Korea

Published online: 13 February 2024

classified in terms of predicted unknown causal factors. In "Multiscale Analysis Using PINNs" section, the literature on multi-scale analysis using PINNs are summarized. Finally, the conclusions are provided in "Conclusion" section.

Fundamentals of Physics Informed Neural Networks (PINNs)

In this section, the fundamental formulations of PINNs are provided in an example of linear elasticity problem. Specifically, the loss functions of PINNs are explained in a solid mechanics example. In addition, the characteristics of PINNs are compared with that of finite element method (FEM), which can show the pros and cons of PINNs.

Theory of PINNs

In PINNs, the approximated solution $\mathbf{u}(\mathbf{x})$ of the governing equation is represented by a deep neural network, with the input being the spatial coordinates $\mathbf{x}$:

$$\mathbf{u}_{NN}(\mathbf{x};\theta) = \mathcal{N}_u(\mathbf{x};\theta) \,, \tag{1}$$

where $\mathcal{N}(\mathbf{x};\theta)$ is neural networks with weights θ. The weights θ are learned by solving an optimization problem that minimizing a loss function $\mathcal{L}$. The loss function $\mathcal{L}$ consists of two terms: a data fitness term $\mathcal{L}_{Data}$ and a physics informed term $\mathcal{L}_{Phys}$:

$$\mathcal{L} = \mathcal{L}_{Data} + \mathcal{L}_{Phys} \,. \tag{2}$$

The data fitness term $\mathcal{L}_{Data}$ aims to match the neural network output with the given data $\mathbf{u}_{Data}$, which can be formulated as:

$$\mathcal{L}_{Data} = \frac{1}{N}\sum_{i=1}^{n}(\mathbf{u}_{NN} - \mathbf{u}_{Data})^2. \tag{3}$$

To explain a physics informed term $\mathcal{L}_{Phys}$ in (2), a solid mechanics will be used as an example in the following. Figure 1 shows the schematic diagram including loss functions of PINNs in linear elasticity problems. In a linear elasticity problem, the approximated solution $\mathbf{u}(\mathbf{x})$ represents the displacement vector. The physics informed loss term $\mathcal{L}_{Phys}$ can be represented as the sum of governing equation loss, $\mathcal{L}_{gov}$, Dirichlet boundary condition loss, $\mathcal{L}_{BC_D}$, and Neumann boundary condition loss $\mathcal{L}_{BC_N}$:

$$\mathcal{L}_{Phys} = \mathcal{L}_{gov} + \mathcal{L}_{BC_D} + \mathcal{L}_{BC_N} \,. \tag{4}$$

The governing equation of a linear elastivity problem is derived from the principle of linear momentum conservation as:

$$\nabla \cdot \boldsymbol{\sigma} + \rho_0 \mathbf{f_b} = \mathbf{0} \quad \text{in } \Omega \,. \tag{5}$$

where $\boldsymbol{\sigma}$ is the stress tensor, ρ_0 is the mass density, $\mathbf{f_b}$ is the body force vector, and Ω is a bounded domain. The constitutive equation that connects the stress tensor $\boldsymbol{\sigma}$ and strain tensor $\boldsymbol{\epsilon}$ can be written as:

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\epsilon} + \lambda \mathrm{tr}(\boldsymbol{\epsilon})\mathbf{I} \,, \tag{6}$$

where μ and λ are the Laḿe constants that are related to Young's modulus E and Poisson's ratio ν. When all displacement gradients are small, the strain tensor $\boldsymbol{\epsilon}$ is defined as

$$\boldsymbol{\epsilon} = \frac{1}{2}\left(\nabla\mathbf{u} + \left(\nabla\mathbf{u}\right)^{\mathrm{T}}\right) . \tag{7}$$

The strong form of the governing equation (5) can be directly utilized as the governing equation loss term $\mathcal{L}_{gov}$ in (4):

$$\mathcal{L}_{gov} = \left(\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{\epsilon}(\mathbf{u}_{NN})) - \mathbf{f_b}\right)^2 \tag{8}$$

The boundary conditions are given as:

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma_D \,, \tag{9}$$

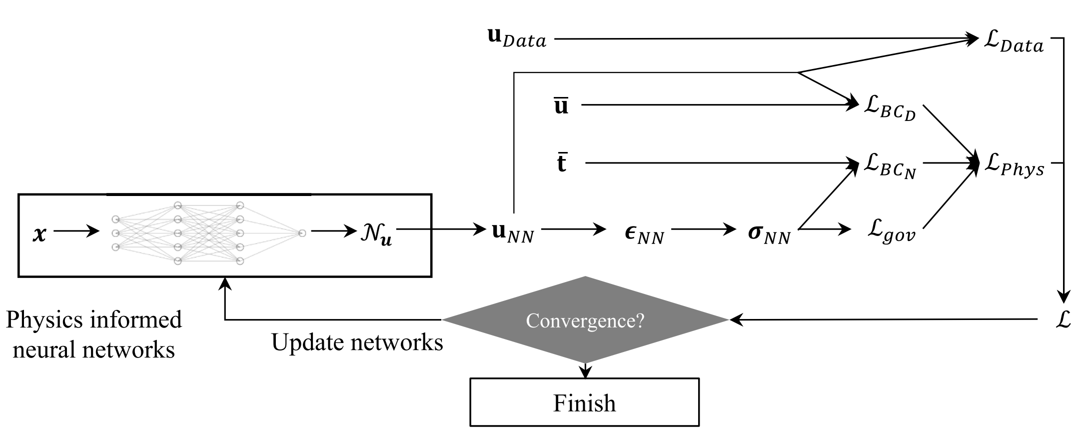

Fig. 1 Schematic diagram including loss functions of PINNs in linear elasticity problems

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{on } \Gamma_N . \tag{10}$$

where $\mathbf{u}$ is the displacement field, $\bar{\mathbf{u}}$ is prescribed displacement on Dirichlet boundary region Γ_D, $\mathbf{n}$ is outward normal vector, and $\bar{\mathbf{t}}$ is prescribed surfaced traction on Neumann boundary region Γ_N. Then, the loss terms for boundary conditions in (4) can be given as:

$$\mathcal{L}_{BC_D} = \left(\mathbf{u}_{NN} - \bar{\mathbf{u}}\right)^2 , \tag{11}$$

$$\mathcal{L}_{BC_N} = \left(\boldsymbol{\sigma}(\boldsymbol{\epsilon}(\mathbf{u}_{NN})) \cdot \mathbf{n} - \bar{\mathbf{t}}\right)^2 . \tag{12}$$

Comparison of PINN and FEM

Table 1 compares the characteristics of PINNs and the conventional numerical analysis method, i.e. Finite Element Method (FEM), which is inspired by Lu et al. [9]. In PINNs, the unknown variables are the weights of the neural networks, and the unknown variables are found using an iterative solver that is less robust than the direct solver. On the other hands, the unknown variables of the FEM are the nodal state variables, and the direct solver can be utilized for linear problems or iterative solver for nonlinear problems. The required information for PINNs is optional. In other words, PINNs can accommodate incomplete problem formulations as long as data is available, which means that PINNs do not necessarily require all detailed informations such as governing equations, boundary conditions and material properties. However, the FEM requires all the information including governing equations, material properties, geometry, and boundary conditions. The flexibility of PINNs gives an advantage over conventional numerical analysis methods like FEM, which allows for the solving of problems where some traditional requirements can be substituted or supplemented with empirical data.

Theoretical Foundations of PINNs in Inverse Problems

In this section, we delve into the theoretical foundations of Physics-Informed Neural Networks (PINNs) as applied to inverse problems, a domain where traditional methods like Finite Element Method (FEM) often encounter limitations. Central to this discussion is the data loss term, $\mathcal{L}_{Data}$, which is pivotal in scenarios where certain analytical components are absent or difficult to quantify. Unlike FEM, which primarily relies on pre-defined physical models, PINNs leverage $\mathcal{L}_{Data}$ to incorporate observational data directly into the solution process. This integration enables PINNs to adaptively reconstruct unobservable parameters or states from available data, a characteristic essential in solving inverse problems. We explore the mathematical formulation of $\mathcal{L}_{Data}$ and its integration into the overall loss function of PINNs, highlighting how this facilitates the handling of complex, data-sparse, or uncertain scenarios typical in inverse problem analysis. This theoretical foundation sets the stage for Chapter 4, where we will present a concrete example demonstrating these principles in a practical application, thereby bridging the gap between theory and real-world implementation.

Forward Analysis Using PINNs

As a numerical analysis tool for solving forward problems, PINNs have been actively studied. The first part of this section reviews methodological efforts in the aspects of of PINN loss functions, neural network architectures, and regularization schemes. In addition, the approach using a domain decomposition, and the approximation of operator by neural networks are reviewed. The second part of this section classifies the literature on solving forward problems using PINNs in terms of physical disciplines.

Table 1 Comparison between PINN and FEM inspired by Lu et al. [9]

	PINNs	FEM
Unknown variables	Weights of the neural networks	Nodal stat variable
Solver	Iterative	Direct (linear) iterative(linear/ nonlinear)
Required information	Optional Governing equations Material property Geometry Boundary condition Data	Mandatory Governing equations Material property Geometry Boundary condition

PINNs Methodologies

Various types of PINN loss functions have been investigated. Raissi et al. [1] introduced the residual expressed in the strong form of the PDE as the loss function. Kharazmi et al. [10] proposed the variational formulation for the loss function to handle governing equation. Samaniego et al. [11] proposed deep energy method (DEM) which use the energy of the system as the loss function.

Special types of neural networks have been studied for the use in PINNs. Rao et al. [12] proposed a deep learning architecture that explicitly encodes the physics. Geneva and Zabaras [13] proposed the auto-regressive dense encoder-decoder convolutional neural network(CNN) for PINNs. Gao et al. [14] proposed PhyGeoNet which is a physics-constrained CNN learning architecture to handle irregular domains. Wandel et al. [15] proposed spline-PINN that combines PINN with Hermite spline kernels based CNN.

To mitigate numerical instability of PINNs, various regularization techniques have been proposed. Wang et al. [16] presented a learning rate annealing algorithm that employs statistics of the gradient while training the model to harmonize the interaction among various elements within combined loss functions. McClenny and Braga-Neto [17] proposed self-adaptive PINNs (SA-PINNs), which utilizes an adaptive training approach with customizable adaptation weights. Chiu et al. [18] introduced CAN-PINNs, a framework that merges the advantages of numerical and automatic differentiation to enhance the robustness and efficiency of PINNs training. Yuan et al. [19] proposed the auxiliary physics informed neural networks (A-PINN), which is a multi-output neural network designed to concurrently depict the main variables and integrals in the governing equations, overcoming the restrictions of integral discretization. Haghighat et al. [20] proposed nonlocal PINN using Peridynamic Differential Operator (PDDO) which can handle localized deformation and sharp gradients in the solution.

In addition, Jagtap et al. [21] proposed a conservative PINN (cPINN) with domain decomposition for nonlinear conservation laws. They extended the cPINN to eXtended PINNs (XPiNN) [22] for the general problems. Dwivedi et al. [23] proposed Distributed PINN (DPINN) architecture to handle multi-domain structure.

The study focusing on approximating the nonlinear operator by neural network [24] have been conducted. In [25, 26], Lu et al. proposed the special neural network architecture, deep operator network (DeepONet), to learn diverse linear or nonlinear explicit and implicit operators. Wang et al. [27] extended the DeepONet using physics informed concept. Figure 2 shows the neural network architecture and loss functions of physics informed DeepONet [24]. Li et al. [28] proposed the Fourier Neural Operator, which is a type of neural operator that represents the integral kernel directly in Fourier space. Li et al. [29] extended the Fourier neural operator using physics-informed concept.

Applications for Various Physics

PINNs have been utilized for solving differential equations in various physical disciplines including fluid and solid mechanics. A variety of approaches have been developed to address fluid mechanics problems. Ranade et al. [30] have presented DiscretizationNet, which uses finite volume discretization techniques to approximate the PDE residual. Oldenburg et al. [31] proposed geometry aware PINN(GAPINN), which is a variational auto encoder to handle an irregular geometries without parameterization. Cheng and Zhang [32] combined PINNs with Resnet blocks (Res-PiNN) to make the neural network

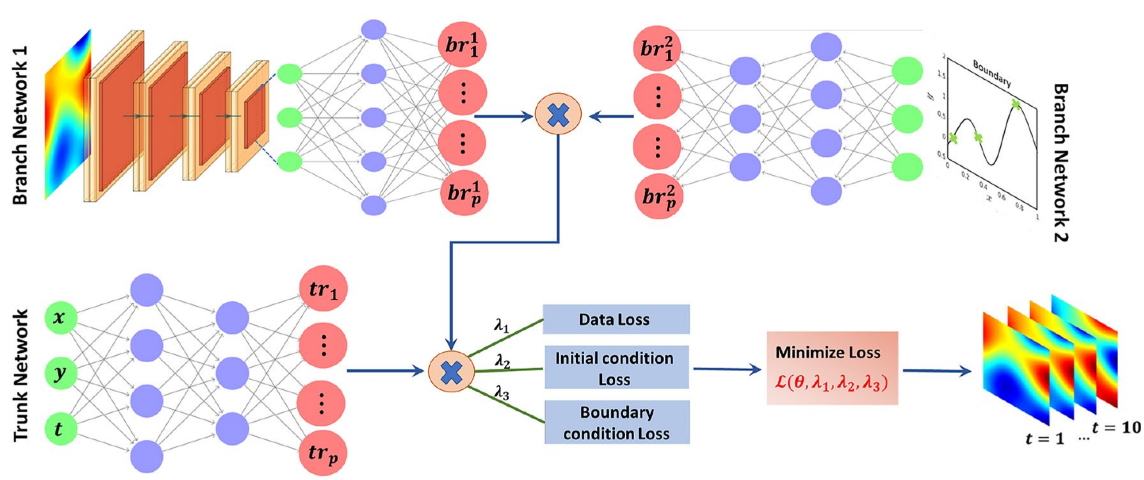

Fig. 2 Schematic representation of physics informed DeepONet [24]

more stable. Mahmoudabadbozchelou et al. [33] proposed Non-newtonian PINNs(nn-PINNs) that can address constitutive models while adhering to mass and momentum conservation laws for the non-Newtonian fluids. Mahmoudabadbozchelou and Jamali [34] also proposed Rheolog-informed neural netowkrs(RhINNs) to treat Thixotropic-Elasto-Visco-Plastic (TEVP) complex fluid. Thakur et al. [35] proposed ViscoelasticNet that can model viscoelastic flow using PINNs. Haghighat et al. [36] applied PINNs to simulate the flow and deformation of multiple phases within porous materials. Almajid and Abu-Al-Saud [37] developed PINNs for the analysis of water-filled porous medium. Wessels et al. [38] proposed Neural Particle Method (NPM) to simulate incompressible fluid flow involving free surfaces. Zhang et al. [39] developed a PINN that establishes a connection between inputs (such as parameterized geometry, initial conditions, and boundary conditions) and outputs (like velocity and pressure) for real-time simulation in design.

The use of PINNs for the analysis of solid mechanics has been an active research topic in computational mechanics. Haghighat et al. [40] introduced the PINNs for the analysis solid mechanics using mixed variables (displacement and stress) with individual networks for linear and non-linear elastoplasticity. Rao et al. [41] demonstrated the effect of mixed variables and proposed a approach to hardly enforce boundary conditions. Yadav et al. [42] proposed DPINN [23] to analyze linear elasticity problem. Henkes et al. [43] applied domain decomposition to handle the non-linear stress and displacement produced by materials with inhomogeneous properties and sharp phase interface. Abueidda et al. [44] proposed Deep Energy Method (DEM) for hyperelastic and viscoelastic material. Fuhg et al. [45] proposed Mixed Deep Energy method (mDEM) that uses mixed variable. This study showed that the mDEM can treat the stress concentration. Abueidda et al. [46] integrated the potential energy functional and the residuals of the governing equations for hyperelasticity. Rezaei et al. [47] also proposed the mixed formulation to handle heterogeneous domain. Bai et al. [48] applied the Least Squares Weighted Residual (LSWR) loss to enhance generalization and alleviates scale discrepancies. Yan et al. [49] utilized the extreme learning machine [50] for PINNs to solve plate and shell problems. Goswami et al. [51] proposed a physics-informed variational formulation of DeepONet (V-DeepONet) to make a surrogate model for brittle fracture. Zhang and Gu [52] applied PINN to analyze digital materials that are a composite as an assembly of material voxels. Song et al. [53] proposed the graph neural networks to analyze a structural system such as a planner frame structure.

Inverse Problems with PINNs

Inverse problem is the process of calculating from a set of actual observations the causal factors that produces them. The examples of inverse problems include image reconstruction, parameter estimation in system identification, deconvolution in signal process. As one of numerical model having strong flexibility, PINNs have been actively studied for solving the inverse problems. Here, the existing works on PINNs for inverse problems were classified according to the unknown causal factors that will be predicted, such as state field, parameter, material field, internal structure, and geometric design. Table 2 shows briefly summarizes previous research articles about PINNs for inverse problems.

Unmeasured State Field Regression

Inverse problems for predicting unmeasured state field from the data have been solved using PINNs. Raissi et al. [6] developed a PINNs approach to predict unmeasurable field in fluid mechanics such as shear stresses. Go et al. [5] proposed a PINN-based surrogate model to predict the entire temperature field and the input heat flux for measurement data obtained from a few physical temperature sensors. Arora [54] proposed physics informed super-resolution network (PhySRNet). The PhySRNet can generate the high resolution deformation fields using low resolution deformation fields input when the material is heterogeneous hyperelastic material.

Parameter Estimation

A parameter estimation is one research topic in the literature on inverse problems using PINNs. In [40], Haghighat et al. implemented PINNs to estimate the physical parameters of linear elastostatic and non-linear elastoplasticity problems. For this, multiple networks were utilized, and it was found that the training phase converges more rapidly when transfer learning is used. Figure 3 showed the neural network architecture and loss functions in [40] that was proposed for the identification of model parameters. In [40, 55], material property was estimated using PINNs. Combining the governing physical principles with the provided data enables more accurate and reliable estimations. In [56], Kadeethum et al. estimated the physical properties of porous media. In this work, the effect of different batch sizes was investigated, and it turned out that training with small batch sizes enables better approximations of the physical parameters than using large batches or the full batch. Mao et al. [57] proposed a

Table 2 Summary of PINNs papers for inverse problem

Classification	Reference	Given data	Unknown causal factors
Unmeasured state field regression	Raissi et al. [6]	Concentration data of the passive scalar in domain	Velocity and pressure fields
	Go et al. [5]	Temperature data in domain	Temperature fields and heat flux
	Arora [54]	Low resolution displacement and stress data in domain	High resolution displacement and stress field in domain
Parameter estimation	Haghighat et al. [40]	Displacement and stress data in domain	Lamé parameters (λ, μ) and yield stress
	Hamel et al. [55]	Displacement data on boundary and global force-displacement data	Hyperelastic constitutive model parameters
	Kadeethum et al. [56]	Displacement and pressure data in domain	Parameters of nonlinear Biot's equations
	Mao et al. [57]	Density, velocity and the pressure in domain	adiabatic index
Material field estimation	Tartakovsky et al. [58]	Solution data in domain	Diffusion coefficient distribution in domain
	Deng et al. [59]	Displacement data in domain	Young's modulus and Poisson's ratio distribution in domain
	Kamali et al. [8]	Strain data in domain	Young's modulus and Poisson's ratio distribution in domain
	Shukla et al. [60]	Displacement data in domain	Elements of the stiffness tensor
	Zhang et al. [61]	Displacement data in domain	Shear modulus distribution in domain
	Wei et al. [62]	Displacement data in domain	Young's modulus and Poisson's ratio distribution in domain
Characterizing internal structures	Zhang et al. [7]	Displacement data on boundary	Geometric parameters and material property of inclusion
	Chen et al. [63]	Axial displacement data in domain	Young's modulus and Poisson's ratio distribution in domain
	Mowlavi et al. [64]	Displacement data on boundary	Geometry of inclusion
Geometric design	Fang and Zhan [65]	No data required	Optimum distribution of relative permeability and permittivity
	Zehnder et al. [66]	No data required	Optimum geometry of structure
	Joglekar et al. [67]	No data required	Optimum geometry of structure
	Lu et al. [68]	No data required	Optimum geometry of structure
	Lu et al. [69]	(Network) Low and high fidelity measured data in domain/(Design) No data required	(Network) Real-time high accuracy solution in domain/(Design) Optimal pore locations

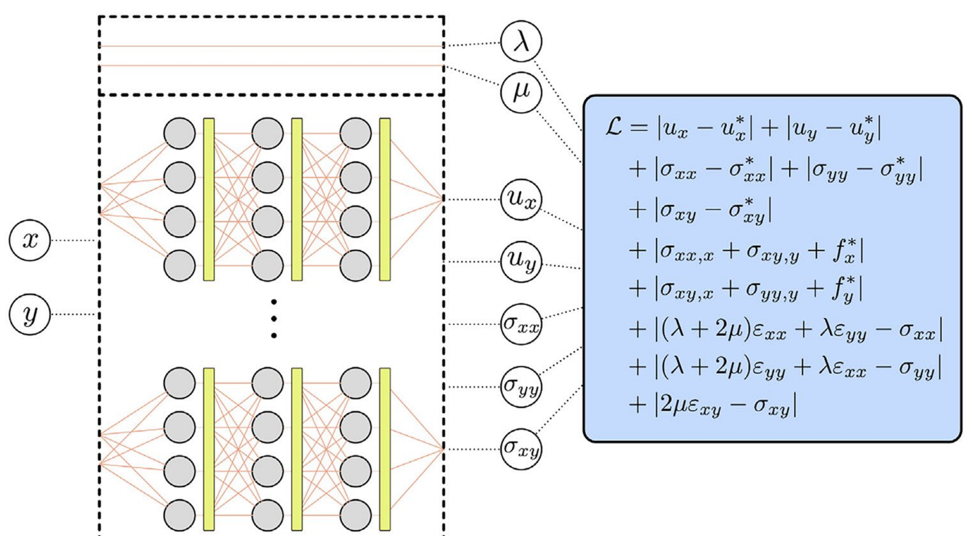

Fig. 3 The network architecture for identification of model parameters[40]

scheme for identifying the adiabatic index of gas by utilizing the data related to the density, pressure and velocity.

Material Field Estimation

The study on PINNs for estimating the distribution of material properties has been conducted. Tartakovsky et al. [58] showed that PINNs can estimate the distribution of an unknown space-dependent diffusion coefficient in a linear diffusion equation and an unknown constitutive relationship in a non-linear diffusion equation when the solution field data are given. Deng et al. [59] predicted the uniformly distributed Young's modulus E or Poisson's ratio ν when noisy strain data are given in whole domain. Moreover, Kamali et al. [8] estimated the complex shape distribution of Young's modulus E and Poisson's ratio ν when strain data are given in whole domain. Shukla et al. [60] find the the distribution of stiffness tensor element of Polycrystalline Nickel using the measured wave field data. Zhang et al. [61] obtained the shear modulus μ distribution of hyperelastic solids using displacement field data. Wei et al. [62] implemented PINNs based on the constitutive equation gap method (CEGM) to determine heterogeneous material property such as Young's modulus and Poisson's ratio when displacement field are given in whole domain.

Characterizing Internal Structures

PINNs have been applied to characterize the unknown internal structures from measurement data. In [7], Zhang et al. presented a general framework based on PINNS for identifying unknown geometric and material parameters. The geometry of the material was parameterized using a differentiable and trainable method that can identify multiple structural features. In [63], Chen et al. integrated a displacement network and an elasticity network to reconstruct the Young's modulus field of a heterogeneous object based on only a measured axial displacement field. It also allows for the removal of the assumption of material incompressibility, enabling the reconstruction of both Young's modulus and Poisson's ratio fields simultaneously. It is demonstrated that using multiple measurements can mitigate the potential error introduced by the "eggshell" effect, in which the presence of stiff material prevents the generation of strain in soft material. In [64], Mowlavi et al. introduced a topology optimization framework based on PINNs that solves geometry detection problems without prior knowledge of the number or types of shapes.

Geometric Design

An inverse design problem aims to find the optimal geometry for a given design goals. A research on applying PINNs for inverse design problems has been actively conducted. Fang and Zhan [65] designed the electromagnetic metamaterials using the PINNs that can recover not only the continuous functions but also piecewise functions. Zehnder et al. [66] proposed a design approach that can handle high-dimensional parameter spaces and highly nonlinear objective landscapes. Here, multilayer perceptrons were used to to parameterize both density and displacement fields. Joglekar et al. [67] also proposed a mesh-free design method by integrating a neural network for density field approximation with a neural network for a displacement field approximation. In [68], PINNs with hard constraints were proposed for an inverse design problem. Here, all the constraints in PINNs were imposed as hard constraints by using the penalty method and the augmented Lagrangian method. It demonstrated the effectiveness of hard constraints for a holography problem in optics and a fluid problem of Stokes flow. Lu et al. [69] applied multi-fidelity DeepONet to the inverse design problem of the phonon Boltzmann transport equation. Here, a multi-fidelity DeepONet includes two independent DeepONets coupled by residual learning and input augmentation, and genetic algorithm and topology optimization were used to solve the optimization problem.

Multiscale Analysis Using PINNs

A lot of efforts have been made to apply PINNs for solving multiscale analysis problems. Multiscale problems with wide range of physical circumstances have been solved by applying PINNs. Wu et al. [70] proposed Asymptotic-Preserving Convolutional Deep Operator Networks (APCONs) to solve the multiscale time-dependent linear transport equations which scale are controlled by the Knudsen number ϵ. Cai et al. [71] proposed DeepM$Mnet to solve multiphysics and multiscale problems which have wide order range of velocity. This study is extended to hypersonics [72]. Lin et al. [73] validated DeepONet's ability to predict dynamics in both deterministic macroscale and stochastic microscale regimes, overcoming challenges like data sparsity and noise.

Approaches using PINNs have been developed for treating high frequency fine scale field. Liu and Cai [74] proposed multiscale DeepONet to handle highly oscillatory continuous functions such as seismic wave responses. Leung et al. [75] proposed neural homogenization-based PINN (NH-PINN) to break down intricate problems into more manageable sub-tasks in a three-step process. This involves solving cell problems, determining homogenized coefficients, and addressing the homogenized equation using PINNs to achieve results. Han and Lee [76] proposed a neural network designed for multiscale homogenization problem, utilizing a training loss that doesn't require derivatives. This network employs Brownian walkers to ascertain the macroscopic description of multiscale PDE solutions, which deal with the high-frequency variable permeability in elliptic equations. Zhang et al. [77] proposed an operator learning algorithm with patches to generate fine-scale solutions for the high-frequency variable permeability of elliptic equations when coarse-scale data is given.

Multiscale approach using DeepOnet was proposed by Yin et al. [78]. In this work, DeepONet is used as a surrogate to approximate fine-scale microscopic solution, while a traditional numerical approach like the finite element method models the larger-scale responses. These two models work concurrently, sharing data at their boundary to achieve a unified, convergent solution. In addition, Ahmed and Stinis [79] proposed the multifidelity operator learning approach, which is employed for closure modeling in multiscale systems. This approach utilizes a blend of Proper Orthogonal Decomposition (POD) and Galerkin projection to establish the low-fidelity model, while a Deep Operator Network (DeepONet) is trained to ascertain correction terms.

In addition, Lin et al. [80] applied DeepONet for rapid, on-the-fly predictions across a wide range of spatial and temporal scales, demonstrating its general applicability in diverse multiscale scenarios. Wang et al. [81] proposed Turbulent-Flow Net(TF-Net), a model designed to understand the multi-scale dynamics of turbulent flows. The network consists of three encoders that learn scale-specific component transformations, and a shared decoder that integrates them to predict the next 2D velocity field. Hall et al. [82] proposed Graph-Informed Neural Networks (GINNs) that merge deep learning with probabilistic graphical models to efficiently represent complex multiscale and multiphysics systems. The GINN was developed to enhance ion diffusion modeling in supercapacitors by integrating a Bayesian network with a homogenized model, optimizing control variables and reducing computational load. Wu et al. [83] proposed physics-informed Deep Homogenization Networks(DHN) to analyze continuum micromechanical behavior of elastic composites. The displacement field within a periodic unit cell is formulated as a two-scale expansion, characterized by averaged and fluctuating components that rely on global and local coordinates, respectively, under various multi-axial loading scenarios. Zhu et al. [84] proposed a convolutional encoder-decoder neural network approach. CNN is used to capture multiscale features of flow. Lou et al. [85] illustrated the capabilities of Physics-informed Neural Networks (PiNN) in addressing inverse multiscale flow challenges. They applied PINNs to reverse modeling in both continuum and rare-field scenarios, as characterized by the Boltzmann–Bhatnagar–Gross–Krook (BGK) collision model.

Conclusion

Recently, PINNs have been actively studied as a promising numerical analysis tools for solving complex physical systems. In this review paper, the fundamental formulations of PINNs is explained in an example of linear elasticity problem. The literature on the application of PINNs for both forward and inverse problems was then reviewed. In forward analysis problems, PINNs have demonstrated remarkable proficiency in predicting complex system behaviors. In solving inverse problems, PINNs have showed its potential for finding hidden parameters and fied distribution from observable data. In addition, the literature have shown the strong ability of PINNs to treat multiscale analysis The application of PINNs in multiscale analysis may lead to a paradigm shift, offering a data-efficient and flexible framework that can handle tha anylsis of multiple scales.

Acknowledgements This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) (NRF-2023R1A2C2003758).

Declarations

Conflict of Interest The authors declare that they have no known competing financial interests that are directly or indirectly related to the work submitted for publication.

References

1. M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. **378**, 686–707 (2019). https://doi.org/10.1016/j.jcp.2018.10.045
2. G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning. Nat. Rev. Phys. **3**(6), 422–440 (2021). https://doi.org/10.1038/s42254-021-00314-5
3. O. Fuks, H.A. Tchelepi, Limitations of physics informed machine learning for nonlinear two-phase transport in porous media. J. Mach. Learn. Model. Comput. **1**(1), 19–37 (2020). https://doi.org/10.1615/JMachLearnModelComput.2020033905
4. T.G. Grossmann, U.J. Komorowska, J. Latz, C.-B. Schönlieb, Can physics-informed neural networks beat the finite element method? arXiv:2302.04107 (2023)
5. M.-S. Go, J.H. Lim, S. Lee, Physics-informed neural network-based surrogate model for a virtual thermal sensor with real-time simulation. Int. J. Heat Mass Transf. **214**, 124392 (2023). https://doi.org/10.1016/j.ijheatmasstransfer.2023.124392
6. M. Raissi, A. Yazdani, G.E. Karniadakis, Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations. Science **367**(6481), 1026–1030 (2020). https://doi.org/10.1126/science.aaw4741
7. E. Zhang, M. Dao, G.E. Karniadakis, S. Suresh, Analyses of internal structures and defects in materials using physics-informed neural networks. Sci. Adv. **8**(7), 0644 (2022). https://doi.org/10.1126/sciadv.abk0644

8. A. Kamali, M. Sarabian, K. Laksari, Elasticity imaging using physics-informed neural networks: spatial discovery of elastic modulus and Poisson's ratio. Available at SSRN 4203110
9. L. Lu, X. Meng, Z. Mao, G.E. Karniadakis, Deepxde: a deep learning library for solving differential equations. SIAM Rev. **63**(1), 208–228 (2021). https://doi.org/10.1137/19m1274067
10. E. Kharazmi, Z. Zhang, G.E.M. Karniadakis, HP-VPINNS: variational physics-informed neural networks with domain decomposition. Comput. Methods Appl. Mech. Eng. **374**, 113547 (2021). https://doi.org/10.1016/j.cma.2020.113547
11. E. Samaniego, C. Anitescu, S. Goswami, V.M. Nguyen-Thanh, H. Guo, K. Hamdia, X. Zhuang, T. Rabczuk, An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications. Comput. Methods Appl. Mech. Eng. **362**, 112790 (2020). https://doi.org/10.1016/j.cma.2019.112790
12. C. Rao, H. Sun, Y. Liu, Hard encoding of physics for learning spatiotemporal dynamics. arXiv:2105.00557 (2021)
13. N. Geneva, N. Zabaras, Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks. J. Comput. Phys. **403**, 109056 (2020). https://doi.org/10.1016/j.jcp.2019.109056
14. H. Gao, L. Sun, J.-X. Wang, Phygeonet: physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDES on irregular domain. J. Comput. Phys. **428**, 110079 (2021). https://doi.org/10.1016/j.jcp.2020.110079
15. N. Wandel, M. Weinmann, M. Neidlin, R. Klein, Spline-PINN: approaching PDES without data using fast, physics-informed hermite-spline CNNS. Proc. AAAI Conf. Artif. Intell. **36**(8), 8529–8538 (2022). https://doi.org/10.1609/aaai.v36i8.20830
16. S.F. Wang, Y.J. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM J. Sci. Comput. **43**(5), 3055–3081 (2021). https://doi.org/10.1137/20m1318043. (**Xg7ux Times Cited:214 Cited References Count:57**)
17. L. McClenny, U. Braga-Neto, Self-adaptive physics-informed neural networks using a soft attention mechanism. arXiv:2009.04544 (2020)
18. P.-H. Chiu, J.C. Wong, C. Ooi, M.H. Dao, Y.-S. Ong, Can-PINN: a fast physics-informed neural network based on coupled-automatic-numerical differentiation method. Comput. Methods Appl. Mech. Eng. **395**, 114909 (2022). https://doi.org/10.1016/j.cma.2022.114909
19. L. Yuan, Y.-Q. Ni, X.-Y. Deng, S. Hao, A-PINN: auxiliary physics informed neural networks for forward and inverse problems of nonlinear integro-differential equations. J. Comput. Phys. **462**, 111260 (2022). https://doi.org/10.1016/j.jcp.2022.111260
20. E. Haghighat, A.C. Bekar, E. Madenci, R. Juanes, A nonlocal physics-informed deep learning framework using the peridynamic differential operator. Comput. Methods Appl. Mech. Eng. **385**, 114012 (2021). https://doi.org/10.1016/j.cma.2021.114012
21. A.D. Jagtap, E. Kharazmi, G.E. Karniadakis, Conservative physics-informed neural networks on discrete domains for conservation laws: applications to forward and inverse problems. Comput. Methods Appl. Mech. Eng. **365**, 113028 (2020). https://doi.org/10.1016/j.cma.2020.113028
22. A.D. Jagtap, G.E. Karniadakis, Extended physics-informed neural networks (XPINNS): a generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations. Commun. Comput. Phys. **28**, 2002–2041 (2020). https://doi.org/10.4208/cicp.OA-2020-0164
23. V. Dwivedi, N. Parashar, B. Srinivasan, Distributed learning machines for solving forward and inverse problems in partial differential equations. Neurocomputing **420**, 299–316 (2021). https://doi.org/10.1016/j.neucom.2020.09.006
24. S. Goswami, A. Bora, Y. Yu, G.E. Karniadakis, in *Physics-Informed Deep Neural Operator Networks* ed. by T. Rabczuk, K.-J. Bathe (Springer, Cham, 2023), pp. 219–254. https://doi.org/10.1007/978-3-031-36644-4_6
25. L. Lu, P. Jin, G.E. Karniadakis, Deeponet: learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. arXiv:1910.03193 (2019)
26. L. Lu, P. Jin, G. Pang, Z. Zhang, G.E. Karniadakis, Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. Nat. Mach. Intell. **3**(3), 218–229 (2021). https://doi.org/10.1038/s42256-021-00302-5
27. S. Wang, H. Wang, P. Perdikaris, Learning the solution operator of parametric partial differential equations with physics-informed deeponets. Sci. Adv. **7**(40), 8605 (2021). https://doi.org/10.1126/sciadv.abi8605
28. Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations. in *ICLR 2021: The Ninth International Conference on Learning Representations* (2021)
29. Z. Li, H. Zheng, N. Kovachki, D. Jin, H. Chen, B. Liu, K. Azizzadenesheli, A. Anandkumar, Physics-informed neural operator for learning partial differential equations. arXiv:2111.03794 (2021)
30. R. Ranade, C. Hill, J. Pathak, Discretizationnet: a machine-learning based solver for Navier–Stokes equations using finite volume discretization. Comput. Methods Appl. Mech. Eng. **378**, 113722 (2021). https://doi.org/10.1016/j.cma.2021.113722
31. J. Oldenburg, F. Borowski, A. Öner, K.-P. Schmitz, M. Stiehm, Geometry aware physics informed neural network surrogate for solving Navier–Stokes equation (GAPINN). Adv. Model. Simul. Eng. Sci. **9**(1), 8 (2022). https://doi.org/10.1186/s40323-022-00221-z
32. C. Cheng, G.-T. Zhang, Deep learning method based on physics informed neural network with resnet block for solving fluid flow problems. Water **13**(4), 423 (2021)
33. M. Mahmoudabadbozchelou, G.E. Karniadakis, S. Jamali, NN-PINNS: non-Newtonian physics-informed neural networks for complex fluid modeling. Soft Matter **18**(1), 172–185 (2022)
34. M. Mahmoudabadbozchelou, S. Jamali, Rheology-informed neural networks (RHINNS) for forward and inverse metamodelling of complex fluids. Sci. Rep. **11**(1), 12015 (2021). https://doi.org/10.1038/s41598-021-91518-3
35. S. Thakur, M. Raissi, A.M. Ardekani, Viscoelasticnet: a physics informed neural network framework for stress discovery and model selection. arXiv:2209.06972 (2022)
36. E. Haghighat, D. Amini, R. Juanes, Physics-informed neural network simulation of multiphase poroelasticity using stress-split sequential training. Comput. Methods Appl. Mech. Eng. **397**, 115141 (2022). https://doi.org/10.1016/j.cma.2022.115141
37. M.M. Almajid, M.O. Abu-Al-Saud, Prediction of porous media fluid flow using physics informed neural networks. J. Petrol. Sci. Eng. **208**, 109205 (2022). https://doi.org/10.1016/j.petrol.2021.109205
38. H. Wessels, C. Weißenfels, P. Wriggers, The neural particle method—an updated Lagrangian physics informed neural network for computational fluid dynamics. Comput. Methods Appl. Mech. Eng. **368**, 113127 (2020). https://doi.org/10.1016/j.cma.2020.113127
39. T. Zhang, B. Dey, P. Kakkar, A. Dasgupta, A. Chakraborty, Frequency-compensated PINNS for fluid-dynamic design problems. arXiv:2011.01456 (2020)
40. E. Haghighat, M. Raissi, A. Moure, H. Gomez, R. Juanes, A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. Comput. Methods Appl.

Mech. Eng. **379**, 113741 (2021). https://doi.org/10.1016/j.cma.2021.113741
41. C. Rao, H. Sun, Y. Liu, Physics-informed deep learning for computational elastodynamics without labeled data. J. Eng. Mech. **147**(8), 04021043 (2021). https://doi.org/10.1061/(ASCE)EM.1943-7889.0001947
42. G.K. Yadav, S. Natarajan, B. Srinivasan, Distributed PINN for linear elasticity—a unified approach for smooth, singular, compressible and incompressible media. Int. J. Comput. Methods **19**(08), 2142008 (2022). https://doi.org/10.1142/s0219876221420081
43. A. Henkes, H. Wessels, R. Mahnken, Physics informed neural networks for continuum micromechanics. Comput. Methods Appl. Mech. Eng. **393**, 114790 (2022). https://doi.org/10.1016/j.cma.2022.114790
44. D.W. Abueidda, S. Koric, R.A. Al-Rub, C.M. Parrott, K.A. James, N.A. Sobh, A deep learning energy method for hyperelasticity and viscoelasticity. Eur. J. Mech. A. Solids **95**, 104639 (2022). https://doi.org/10.1016/j.euromechsol.2022.104639
45. J.N. Fuhg, N. Bouklas, The mixed deep energy method for resolving concentration features in finite strain hyperelasticity. J. Comput. Phys. **451**, 110839 (2022). https://doi.org/10.1016/j.jcp.2021.110839
46. D.W. Abueidda, S. Koric, E. Guleryuz, N.A. Sobh, Enhanced physics-informed neural networks for hyperelasticity. Int. J. Numer. Meth. Eng. **124**(7), 1585–1601 (2023). https://doi.org/10.1002/nme.7176
47. S. Rezaei, A. Harandi, A. Moeineddin, B.-X. Xu, S. Reese, A mixed formulation for physics-informed neural networks as a potential solver for engineering problems in heterogeneous domains: comparison with finite element method. Comput. Methods Appl. Mech. Eng. **401**, 115616 (2022). https://doi.org/10.1016/j.cma.2022.115616
48. J. Bai, T. Rabczuk, A. Gupta, L. Alzubaidi, Y. Gu, A physics-informed neural network technique based on a modified loss function for computational 2d and 3d solid mechanics. Comput. Mech. (2022). https://doi.org/10.1007/s00466-022-02252-0
49. C.A. Yan, R. Vescovini, L. Dozio, A framework based on physics-informed neural networks and extreme learning for the analysis of composite structures. Comput. Struct. **265**, 106761 (2022). https://doi.org/10.1016/j.compstruc.2022.106761
50. G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications. Neurocomputing **70**(1), 489–501 (2006). https://doi.org/10.1016/j.neucom.2005.12.126
51. S. Goswami, M. Yin, Y. Yu, G.E. Karniadakis, A physics-informed variational deeponet for predicting crack path in quasi-brittle materials. Comput. Methods Appl. Mech. Eng. **391**, 114587 (2022). https://doi.org/10.1016/j.cma.2022.114587
52. Z. Zhang, G.X. Gu, Physics-informed deep learning for digital materials. Theor. Appl. Mech. Lett. **11**(1), 100220 (2021). https://doi.org/10.1016/j.taml.2021.100220
53. L.H. Song, C. Wang, J.S. Fan, H.M. Lu, Elastic structural analysis based on graph neural network without labeled data. Comput.-Aid. Civil Infrastruct. Eng. **38**(10), 1307–1323 (2023). https://doi.org/10.1111/mice.12944. (**I4gv2 Times Cited:1 Cited References Count:60**)
54. R. Arora, Physrnet: physics informed super-resolution network for application in computational solid mechanics. in *2022 IEEE/ACM International Workshop on Artificial Intelligence and Machine Learning for Scientific Applications (AI4S)*, 13–18. (2022). https://doi.org/10.1109/AI4S56813.2022.00008
55. C.M. Hamel, K.N. Long, S.L. Kramer, Calibrating constitutive models with full-field data via physics informed neural networks. arXiv:2203.16577 (2022)
56. T. Kadeethum, T.M. Jørgensen, H.M. Nick, Physics-informed neural networks for solving inverse problems of nonlinear biot's equations: batch training. in *54th US Rock Mechanics/Geomechanics Symposium*. OnePetro
57. Z. Mao, A.D. Jagtap, G.E. Karniadakis, Physics-informed neural networks for high-speed flows. Comput. Methods Appl. Mech. Eng. **360**, 112789 (2020). https://doi.org/10.1016/j.cma.2019.112789
58. A.M. Tartakovsky, C.O. Marrero, P. Perdikaris, G.D. Tartakovsky, D. Barajas-Solano, Learning parameters and constitutive relationships with physics informed deep neural networks. arXiv:1808.03398 (2018)
59. Y. Deng, C. Chen, Q. Wang, X. Li, Z. Fan, Y. Li, Modeling a typical non-uniform deformation of materials using physics-informed deep learning: applications to forward and inverse problems. Appl. Sci. **13**(7), 4539 (2023)
60. K. Shukla, A.D. Jagtap, J.L. Blackshire, D. Sparkman, G.E. Karniadakis, A physics-informed neural network for quantifying the microstructural properties of polycrystalline nickel using ultrasound data: a promising approach for solving inverse problems. IEEE Signal Process. Mag. **39**(1), 68–77 (2022). https://doi.org/10.1109/MSP.2021.3118904
61. E. Zhang, M. Yin, G.E. Karniadakis, Physics-informed neural networks for nonhomogeneous material identification in elasticity imaging. arXiv:2009.04525 (2020)
62. Y. Wei, Q. Serra, G. Lubineau, E. Florentin, Coupling physics-informed neural networks and constitutive relation error concept to solve a parameter identification problem. Comput. Struct. **283**, 107054 (2023). https://doi.org/10.1016/j.compstruc.2023.107054
63. C.-T. Chen, G.X. Gu, Physics-informed deep-learning for elasticity: forward, inverse, and mixed problems. Adv. Sci. (2023). https://doi.org/10.1002/advs.202300439
64. S. Mowlavi, K. Kamrin, Topology optimization with physics-informed neural networks: application to noninvasive detection of hidden geometries. arXiv:2303.09280 (2023)
65. Z. Fang, J. Zhan, Deep physical informed neural networks for metamaterial design. IEEE Access **8**, 24506–24513 (2020). https://doi.org/10.1109/ACCESS.2019.2963375
66. J. Zehnder, Y. Li, S. Coros, B. Thomaszewski, Ntopo: mesh-free topology optimization using implicit neural representations. Adv. Neural. Inf. Process. Syst. **34**, 10368–10381 (2021)
67. A. Joglekar, H. Chen, L.B. Kara, DMF-TONN: direct mesh-free topology optimization using neural networks. arXiv:2305.04107 (2023)
68. L. Lu, R. Pestourie, W. Yao, Z. Wang, F. Verdugo, S.G. Johnson, Physics-informed neural networks with hard constraints for inverse design. SIAM J. Sci. Comput. **43**(6), 1105–1132 (2021). https://doi.org/10.1137/21m1397908
69. L. Lu, R. Pestourie, S.G. Johnson, G. Romano, Multifidelity deep neural operators for efficient learning of partial differential equations with application to fast inverse design of nanoscale heat transport. Phys. Rev. Res. **4**(2), 023210 (2022). https://doi.org/10.1103/PhysRevResearch.4.023210. (**PRRESEARCH**)
70. K. Wu, X.-B. Yan, S. Jin, Z. Ma, Capturing the diffusive behavior of the multiscale linear transport equations by asymptotic-preserving convolutional deeponets. Comput. Methods Appl. Mech. Eng. **418**, 116531 (2024). https://doi.org/10.1016/j.cma.2023.116531
71. S. Cai, Z. Wang, L. Lu, T.A. Zaki, G.E. Karniadakis, Deepm&mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks. J. Comput. Phys. **436**, 110296 (2021). https://doi.org/10.1016/j.jcp.2021.110296
72. Z. Mao, L. Lu, O. Marxen, T.A. Zaki, G.E. Karniadakis, Deepm&mnet for hypersonics: predicting the coupled flow and finite-rate chemistry behind a normal shock using neural-network approximation of operators. J. Comput. Phys. **447**, 110698 (2021). https://doi.org/10.1016/j.jcp.2021.110698

73. C. Lin, Z. Li, L. Lu, S. Cai, M. Maxey, G.E. Karniadakis, Operator learning for predicting multiscale bubble growth dynamics. J. Chem. Phys. (2021). https://doi.org/10.1063/5.0041203
74. L. Liu, W. Cai, Multiscale deeponet for nonlinear operators in oscillatory function spaces for building seismic wave responses. arXiv:2111.04860 (2021)
75. W.T. Leung, G. Lin, Z. Zhang, Nh-pinn: neural homogenization-based physics-informed neural network for multiscale problems. J. Comput. Phys. **470**, 111539 (2022). https://doi.org/10.1016/j.jcp.2022.111539
76. J. Han, Y. Lee, A neural network approach for homogenization of multiscale problems. Multiscale Model. Simul. **21**(2), 716–734 (2023). https://doi.org/10.1137/22m1500903
77. Z. Zhang, C. Moya, W.T. Leung, G. Lin, H. Schaeffer, Bayesian deep operator learning for homogenized to fine-scale maps for multiscale PDE. arXiv:2308.14188 (2023)
78. M. Yin, E. Zhang, Y. Yu, G.E. Karniadakis, Interfacing finite elements with deep neural operators for fast multiscale modeling of mechanics problems. Comput. Methods Appl. Mech. Eng. **402**, 115027 (2022). https://doi.org/10.1016/j.cma.2022.115027
79. S.E. Ahmed, P. Stinis, A multifidelity deep operator network approach to closure for multiscale systems. Comput. Methods Appl. Mech. Eng. **414**, 116161 (2023). https://doi.org/10.1016/j.cma.2023.116161
80. C. Lin, M. Maxey, Z. Li, G.E. Karniadakis, A seamless multiscale operator neural network for inferring bubble dynamics. J. Fluid Mech. **929**, 18 (2021). https://doi.org/10.1017/jfm.2021.866
81. R. Wang, K. Kashinath, M. Mustafa, A. Albert, R. Yu, Towards physics-informed deep learning for turbulent flow prediction. Assoc. Comput. Mach. (2020). https://doi.org/10.1145/3394486.3403198
82. E.J. Hall, S. Taverniers, M.A. Katsoulakis, D.M. Tartakovsky, Ginns: graph-informed neural networks for multiscale physics. J. Comput. Phys. **433**, 110192 (2021). https://doi.org/10.1016/j.jcp.2021.110192
83. J. Wu, J. Jiang, Q. Chen, G. Chatzigeorgiou, F. Meraghni, Deep homogenization networks for elastic heterogeneous materials with two- and three-dimensional periodicity. Int. J. Solids Struct. **284**, 112521 (2023). https://doi.org/10.1016/j.ijsolstr.2023.112521
84. Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, P. Perdikaris, Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. J. Comput. Phys. **394**, 56–81 (2019). https://doi.org/10.1016/j.jcp.2019.05.024
85. Q. Lou, X. Meng, G.E. Karniadakis, Physics-informed neural networks for solving forward and inverse flow problems via the Boltzmann-Bgk formulation. J. Comput. Phys. **447**, 110676 (2021). https://doi.org/10.1016/j.jcp.2021.110676