Neural Networks 167 (2023) 80–91

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Prediction of fluid flow in porous media by sparse observations and physics-informed PointNet

Ali Kashefi [a],*, Tapan Mukerji [b]

[a] Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305, United States of America
[b] Department of Energy Science and Engineering, Stanford University, Stanford, CA 94305, United States of America

ARTICLE INFO

Article history:
Received 25 April 2023
Received in revised form 15 July 2023
Accepted 6 August 2023
Available online 9 August 2023

Keywords:
Deep learning
Physics-informed PointNet
Stokes flow
Porous media
Sparse data

ABSTRACT

We predict steady-state Stokes flow of fluids within porous media at pore scales using sparse point observations and a novel class of physics-informed neural networks, called "physics-informed PointNet" (PIPN). Taking the advantages of PIPN into account, three new features become available compared to physics-informed convolutional neural networks for porous medium applications. First, the input of PIPN is exclusively the pore spaces of porous media (rather than both the pore and grain spaces). This feature diminishes required computer memory. Second, PIPN represents the boundary of pore spaces smoothly and realistically (rather than pixel-wise representations). Third, spatial resolution can vary over the physical domain (rather than equally spaced resolutions). This feature enables users to reach an optimal resolution with a minimum computational cost. The performance of our framework is evaluated by the study of the influence of noisy sensor data, pressure observations, and spatial correlation length.

1. Introduction and motivation

Since the late 2018s, deep learning schemes have become popular for the study of various aspects of porous media. Examples are rock image segmentation (Da Wang, Shabaninejad, Armstrong, & Mostaghimi, 2020; Karimpouli & Tahmasebi, 2019b; Niu, Mostaghimi, Shabaninejad, Swietojanski, & Armstrong, 2020; Phan, Ruspini, & Lindseth, 2021); reconstruction and enhancement of rock image resolution (Da Wang, Armstrong, & Mostaghimi, 2019; Liu & Mukerji, 2022; Niu et al., 2021; Wang, Armstrong, & Mostaghimi, 2020); prediction of geometric characteristics such as porosity (Alqahtani, Alzubaidi, Armstrong, Swietojanski, & Mostaghimi, 2020; Bordignon, Figueiredo, Exterkoetter, Rodrigues, & Correia, 2019; Graczyk & Matyka, 2020); prediction of physical properties such as permeability (Al-Zubaidi et al., 2023; Graczyk & Matyka, 2020; Hong & Liu, 2020; Kashefi & Mukerji, 2021; Wu, Yin, & Xiao, 2018), effective diffusivity (Wu et al., 2018), wave propagation velocities (Karimpouli & Tahmasebi, 2019a); and prediction of velocity and pressure fields of fluids within pore spaces (Alhubail, He, AlSinan, Kwak, & Hoteit, 2022; Kamrava, Sahimi, & Tahmasebi, 2021; Lu et al., 2022; Santos et al., 2020; Tartakovsky, Marrero, Perdikaris, Tartakovsky, & Barajas-Solano, 2018; Wang, Chen, Mehana, Lubbers, Bennett, & Kang, 2021; Wang, Chung, Armstrong, & Mostaghimi, 2021). Our focus in the current research is the last one-prediction of fluid velocities and pressure. According to the literature, three main approaches have been taken so far for predicting fluid flow fields in pore spaces of porous media using available deep learning algorithms.

The first approach falls in the category of supervised learning, where the deep learning frameworks used have no information concerning the physics describing fluid flow fields within porous media (Santos et al., 2020; Wang, Chung, et al., 2021). In this approach, plentiful labeled data are generated using numerical solvers (e.g., lattice Boltzmann, finite-element, finite-volume, and fast Fourier transform methods) or collected from lab experiments for training deep learning frameworks to learn an end-to-end mapping from the geometry of a porous medium to its velocity or pressure fields. Convolutional neural networks (CNNs) are commonly used in this approach. The loss functions of these deep learning frameworks are designed based on the mismatch between values predicted by a neural network and training labeled data (i.e., ground truth), mathematically quantified in L^2 norm or other norms.

The second approach also falls in the category of supervised learning; however, the employed deep learning frameworks are "guided" by the problem physics (Kamrava et al., 2021; Wang, Chen, et al., 2021). More precisely, the loss function in this approach is similar to the one specified in the first approach with the difference that it is regularized by extra terms representing the L^2 norm (or other norms) of the residuals of partial differential equations (PDEs), governing the physics of fluid flow in

* Corresponding author.
E-mail addresses: kashefi@stanford.edu (A. Kashefi), mukerji@stanford.edu (T. Mukerji).

https://doi.org/10.1016/j.neunet.2023.08.006
0893-6080/© 2023 Elsevier Ltd. All rights reserved.

porous media. Technically, the PDEs of interest are discretized by a finite difference scheme and then the corresponding stencils are enforced into non-trainable filters in the last layer of CNNs. Compared to the first approach, the second one is faster in convergence, more generalizable, and lastly requires "relatively" smaller labeled training data. One may refer to Kamrava et al. (2021) and Wang, Chen, et al. (2021) for a detailed discussion of this comparison.

Nevertheless with the success of these two approaches, they come with a few shortcomings. First, both require plentiful training data. This is while generating and collecting labeled data are computationally and experimentally expensive. Second, CNNs used in both approaches take the grain and pore spaces as the network inputs and outputs, whereas we exclusively seek for the network predictions only in the pore spaces of porous media where the fluid exists. The integration of the solid grain spaces into CNNs demands extra memory (i.e., RAM). Third, because these two approaches employ CNNs, labeled data used for training have to be in a Cartesian grid format with uniform spacing. This format might lead to unrealistic representation of the geometry of pore spaces, specifically near the pore-grain boundaries. Furthermore, if labeled data are generated using finite element/volume methods on unstructured grids, the generated data have to be interpolated on the uniform Cartesian grids. Such interpolations introduce errors to the training data, and consequently to the network prediction. Additionally, executing the interpolation requires extra effort even before training CNNs. Fourth, CNNs are technically designed for uniform resolutions. Thus, to increase the resolution of an area of interest in pore spaces, users of these two approaches have no choice except to increase the Cartesian grid resolutions everywhere in the domain. This inflexibility enforces high computational costs to the machine learning system of both approaches. Next, we discuss the third approach to resolving and improving the above mentioned limitations.

The third approach falls in the category of weakly supervised learning and is mainly constructed based on the idea of physics-informed neural networks (PINNs). PINNs were first proposed by Raissi, Perdikaris, and Karniadakis (2019) for solving forward and inverse problems and its latter versions with various enhancements and extensions have been introduced such as fPINN (Pang, Lu, & Karniadakis, 2019), nPINN (Pang, D'Elia, Parks, & Karniadakis, 2020), B-PINN (Yang, Meng, & Karniadakis, 2021), hp-VPINN (Kharazmi, Zhang, & Karniadakis, 2021), PPINN (Meng, Li, Zhang, & Karniadakis, 2020), PIPN (Kashefi & Mukerji, 2022), etc. In PINN-based methodology, the loss function is mainly the L^2 norm (or other norms) of the residuals of governing equations of fluid flow fields of porous media, regulated by the L^2 norm of mismatch between neural network predictions and sparse scattered labeled data. Through the lens of porous media, PINNs have been used but only for predicting Darcy's flow on field scale (Alhubail et al., 2022; Tartakovsky et al., 2018). In the current study, for the first time, we use the concept of physics-informed neural networks for the prediction of Stokes flow at pore scales in porous media. Specifically, we use PIPN (Kashefi & Mukerji, 2022) as an advanced version of PINNs. Remarkable advantages of PIPN have been addressed in detail by Kashefi and Mukerji (2022). Using the PIPN technology, all the above mentioned shortcomings of the first two approaches are obviated. Based on the fundamental mathematics of PIPN, only sparse scattered labeled data are required. We exclusively discretize the pore space with a set of scattered point cloud, where the spatial density of point distribution can vary freely over the space. Such freedom allows users to represent the geometry of the pore space and its boundaries smoothly and realistically. Additionally, the PIPN (Kashefi & Mukerji, 2022) framework can conveniently be integrated with unstructured grids, and no data interpolation is needed. In the rest of this article, we illustrate these features practically. In addition, we explore some practical issues in applying PIPN such as (i) the effect of the type of data available at the sparse sensors (with and without pressure data); (ii) the effects of noisy sensor data; and (iii) the impact of sparsity in the sensor locations.

2. Physics-informed PointNet (PIPN) for stokes flows in porous media

2.1. Governing equations and mathematical definition

We first describe the partial differential equations governing the physics of fluid flow fields in porous media at pore scales. The conservation of mass and momentum of an incompressible steady creeping (Stokes) flow of a Newtonian fluid are respectively written as

$$\nabla \cdot \boldsymbol{u} = 0 \text{ in } V, \tag{1}$$

$$\nabla p - \mu \Delta \boldsymbol{u} = \boldsymbol{0} \text{ in } V, \tag{2}$$

where $\boldsymbol{u}$ and p indicate respectively the velocity vector and pressure of the fluid with the dynamic viscosity of μ. The pore space of a porous medium is denoted by V. We further show the x and y components of the velocity vector by u and v, respectively. Note that one may alternatively consider the full Navier–Stokes equations (see e.g., Hassanizadeh & Gray, 1987). The permeability ($\mathcal{K}$) of the porous media in the x direction is computed as Berg (2014), Darcy (1856) and Eshghinejadfard, Daróczy, Janiga, and Thévenin (2016)

$$\mathcal{K} = -\frac{\mu \bar{U}}{\Delta p / L}, \tag{3}$$

where $\bar{U}$ is the average x velocity over the entire space of porous media. The term of $\Delta p/L$ denotes the applied constant pressure gradient over the length L in the x direction of the porous media.

Mathematically, our goal is to solve an inverse problem of the Stokes flow using PIPN. It can be described as follows: given no slip boundary condition on the wall boundaries and a set of sparse labeled data of the velocity and pressure fields at sensor locations, find the full velocity and pressure fields at inquiry points. Moreover, we compute the permeability of the porous media using the predicted velocity fields.

2.2. Physics-informed PointNet (PIPN)

Historically, Kashefi, Rempe, and Guibas (2021) used PointNet (Qi, Su, Mo, & Guibas, 2017) for the first time for supervised deep learning of incompressible flows on irregular geometries. The successes of applying PointNet (Qi et al., 2017) to the area of computational mechanics motivated Kashefi and Mukerji (2022) for proposing physics-informed PointNet (PIPN), which is a weakly supervised deep learning framework for incompressible flows. The PIPN methodology has been expressed in detail by Kashefi and Mukerji (2022). Here we illustrate the PIPN framework from a general point of view and explain how to specialize it for the porous medium applications. Fig. 1 depicts the general flowchart of PIPN. Accordingly, the space of a porous medium (V) is represented by N points such that each point has the spatial x and y coordinates. In the next stage, we feed the constructed point clouds into PointNet (Qi et al., 2017), while the outputs of PointNet (Qi et al., 2017) are the velocity (u, v) and pressure (p) values at the corresponding input points. Afterwards, we take spatial derivatives of outputs (u, v, p) with respect to the corresponding inputs (x, y) using the automatic differentiation

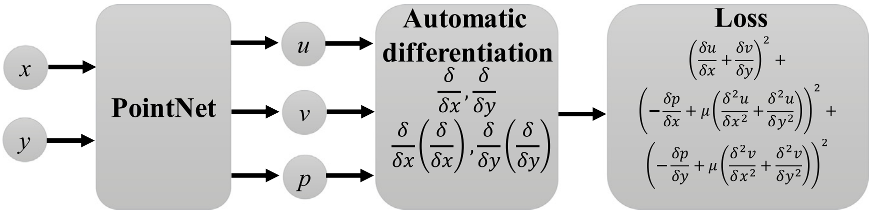
x
y
PointNet
u
v
p
Automatic differentiation
$\frac{\delta}{\delta x}, \frac{\delta}{\delta y}$
$\frac{\delta}{\delta x}\left(\frac{\delta}{\delta x}\right), \frac{\delta}{\delta y}\left(\frac{\delta}{\delta y}\right)$
Loss
$\left(\frac{\delta u}{\delta x}+\frac{\delta v}{\delta y}\right)^2+$
$\left(-\frac{\delta p}{\delta x}+\mu\left(\frac{\delta^2 u}{\delta x^2}+\frac{\delta^2 u}{\delta y^2}\right)\right)^2+$
$\left(-\frac{\delta p}{\delta y}+\mu\left(\frac{\delta^2 v}{\delta x^2}+\frac{\delta^2 v}{\delta y^2}\right)\right)^2$

Fig. 1. Schematic flowchart of Physics-informed PointNet (PIPN) for Stokes flow in porous media. A full description of the loss function is presented in Eqs. (5)–(11).

technology of TensorFlow (Abadi et al., 2016). Finally, we build up the loss function of PIPN as the summation of residuals of governing equations (Eqs. (1)–(2)) as well as the mismatch between predicted outputs and sparse labeled data, all quantified in L^2 norm. Similar to the original version of PIPN proposed by Kashefi and Mukerji (2022), we use the hyperbolic tangent activation function defined as

$$\tanh(\lambda) = \frac{\exp(2\lambda)-1}{\exp(2\lambda)+1}, \tag{4}$$

in all the layers of PIPN. Note that due to the presence of second-order derivatives of velocity fields in Eq. (2), choosing an activation function with a well-defined second-order derivative is essential. The hyperbolic tangent activation function satisfies this criterion. Audiences interested in details of PIPN may refer to Kashefi and Mukerji (2022).

The residuals of conservation of mass ($r^{\text{continuity}}$), conservation of momentum in the $x-$direction (r^{momentum_x}) and in the $y-$direction (r^{momentum_y}), no slip boundary condition of the velocity ($r^{\text{velocity}_{\text{wall}}}$), sparse observations of the velocity field ($r^{\text{velocity}_{\text{obs}}}$) and pressure field ($r^{\text{pressure}_{\text{obs}}}$) are respectively written as

$$r^{\text{continuity}} = \frac{1}{M_1}\sum_{k=1}^{M_1}\left(\frac{\delta \tilde{u}'_k}{\delta \tilde{x}_k}+\frac{\delta \tilde{v}'_k}{\delta \tilde{y}_k}\right)^2, \tag{5}$$

$$r^{\text{momentum}_x} = \frac{1}{M_1}\sum_{k=1}^{M_1}\left(\frac{\delta \tilde{p}'_k}{\delta \tilde{x}_k}-\tilde{\mu}\left(\frac{\delta}{\delta \tilde{x}_k}\left(\frac{\delta \tilde{u}'_k}{\delta \tilde{x}_k}\right)+\frac{\delta}{\delta \tilde{y}_k}\left(\frac{\delta \tilde{u}'_k}{\delta \tilde{y}_k}\right)\right)\right)^2, \tag{6}$$

$$r^{\text{momentum}_y} = \frac{1}{M_1}\sum_{k=1}^{M_1}\left(\frac{\delta \tilde{p}'_k}{\delta \tilde{y}_k}-\tilde{\mu}\left(\frac{\delta}{\delta \tilde{x}_k}\left(\frac{\delta \tilde{v}'_k}{\delta \tilde{x}_k}\right)+\frac{\delta}{\delta \tilde{y}_k}\left(\frac{\delta \tilde{v}'_k}{\delta \tilde{y}_k}\right)\right)\right)^2, \tag{7}$$

$$r^{\text{velocity}_{\text{wall}}} = \frac{1}{M_2}\sum_{k=1}^{M_2}\left(\left(\tilde{u}'_k-0\right)^2+\left(\tilde{v}'_k-0\right)^2\right), \tag{8}$$

$$r^{\text{velocity}_{\text{obs}}} = \frac{1}{M_3}\sum_{k=1}^{M_3}\left(\left(\tilde{u}'_k-\tilde{u}_k\right)^2+\left(\tilde{v}'_k-\tilde{v}_k\right)^2\right), \tag{9}$$

$$r^{\text{pressure}_{\text{obs}}} = \frac{1}{M_3}\sum_{k=1}^{M_3}\left(\tilde{p}'_k-p_k\right)^2, \tag{10}$$

where δ stands for the automatic differentiation operator in the TensorFlow software (Abadi et al., 2016). The number of interior points, points located on wall boundaries, and virtual sensors measuring velocity and pressure values are respectively indicated by M_1, M_2, and M_3. Note that $M_1 + M_2 = N$.

We normalize the output of the velocity and pressure fields because the output of the hyperbolic activation function (see Eq. (4)) only covers the range of $[-1, 1]$. The scaled ground truth velocity and pressure values are shown by ($\tilde{u}$, $\tilde{v}$, $\tilde{p}$), while the predicted velocity and pressure fields by PIPN are denoted by ($\tilde{u}'$, $\tilde{v}'$, $\tilde{p}'$). Additionally, the spatial coordinates of x and y (as the PIPN input) are scaled in the range of $[-1, 1]$ and are shown by $\tilde{x}$ and $\tilde{y}$, respectively. Moreover, $\tilde{\mu}$ is the scaled viscosity. In this sense, the final form of the loss function ($\mathcal{J}$) is determined as

$$\mathcal{J} = \lambda_1 r^{\text{continuity}} + \lambda_2 r^{\text{momentum}_x} + \lambda_3 r^{\text{momentum}_y} + \lambda_4 r^{\text{velocity}_{\text{wall}}} + \lambda_5 r^{\text{velocity}_{\text{obs}}} + \lambda_6 r^{\text{pressure}_{\text{obs}}}, \tag{11}$$

where λ_i ($1 \leq i \leq 6$) are the corresponding weights of each component of the loss function, while they take the inverse of the units of their associated residuals as their own unit. In this way, the loss function ($\mathcal{J}$) is unitless. These weights (λ_i; $1 \leq i \leq 6$) are, indeed, hyperparameters that need to be tuned for reaching the highest possible performance of PIPN. In this study, λ_i are kept constant during training; however, one may use adaptive techniques for online tuning of λ_i as the network is trained (Xiang, Peng, Zheng, Zhao, & Yao, 2021).

2.3. Computational setting

For the numerical examples in the following, dynamic viscosity of $\mu = 0.001$ Pa·s, a pressure difference of $\Delta p = 0.1$ Pa over a length of $L = 0.0064$ m are set. We generate two-dimensional synthetic binary (pore-grain) media with the length (L) of 0.0064 m in both dimensions and three different spatial correlation lengths (l_c) of 0.0005 m, 0.0009 m, and 0.0017 m using the algorithm of truncated Gaussian simulation (Lantuéjoul, 2001; Xu & Journel, 1993). Practically, we first generate a two-dimensional array (64 by 64) of random numbers from the normal distribution with the mean parameter of 0.0 and the standard deviation of 1.0. Next, we filter the array using a two-dimensional Gaussian smoothing kernel with the standard deviation of 2.0 with a filter size equivalent to a desired spatial correlation length (e.g., 5, 9, or 17). Finally, we make the resulting array binary using a threshold such that the porosity falls in the range of [0.25, 0.40]. For these purposes, we use the MATLAB software. Afterward, we convert the resulting image (i.e., the two-dimensional array) to the standard tessellation language (STL) format such that it becomes readable by the COMSOL software. By decreasing the spatial correlation length, the geometry of a porous medium becomes more complicated and the number of points representing the space of the corresponding porous medium in the point cloud likewise increases, imposing higher computational costs on the PIPN system. In this sense, the PIPN capability is validated for a variety of complexity levels.

Table 1 provides a fraction of sensor points to the total number of point cloud points (M_3/N) as well as the average spatial distance of sensors from each other (d_s). Note that the sensor locations (see Fig. 2) are sparser than the point cloud (see Fig. 3) inputs of PIPN. Additionally, the point cloud (see Fig. 3) has a spatially varying density of points with denser points at boundaries and narrow pore throats.

Generally speaking, since the flow at the wall moves with zero velocity (i.e., no slip condition), $r^{\text{velocity}_{\text{wall}}}$ must have less weights

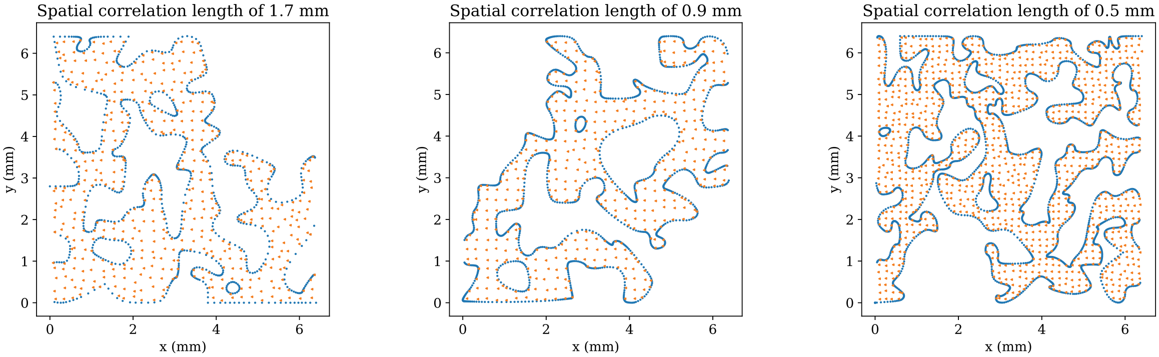

Fig. 2. Sensor locations for porous media with the spatial correlation lengths of 1.7 mm, 0.9 mm, and 0.5 mm.

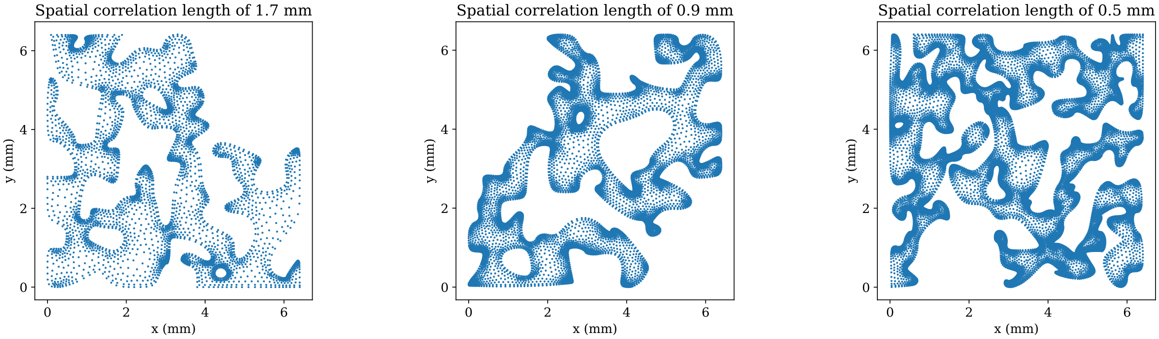

Fig. 3. Point clouds representing porous media with the spatial correlation lengths of 1.7 mm, 0.9 mm, and 0.5 mm.

Table 1
Computational setup for the porous media considered in this study.

Spatial correlation length (l_c)	1.7 mm	0.9 mm	0.5 mm
Number of inquiry points (N)	4231	8727	17661
Number of virtual sensors (M_3)	514	458	1120
Rough percentage of observation (i.e., M_3/N)	12%	5%	6%
Average spatial distance of sensors (d_s)	0.237 mm	0.237 mm	0.16 mm

compared to other residuals; otherwise, PIPN converges to the superficial solution of zero everywhere for all the fields. In this sense, for the porous media under investigation in this study, we set $\lambda_1 = 100$ s, $\lambda_2 = 100$ m^3/N, $\lambda_3 = 100$ m^3/N, $\lambda_5 = 100$ s/m, $\lambda_6 = 100$ m^2/N, and $\lambda_4 = 1$ s/m.

We use the Adam optimizer (Kingma & Ba, 2014) for training the PIPN configuration and set its associated hyperparameters as follows: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\hat{\epsilon} = 10^{-6}$. One may refer to Kingma and Ba (2014) for the definition of these hyperparameters. Furthermore, a constant learning rate of $\alpha = 0.0003$ is chosen for all the porous media considered in this research letter. Training of PIPN is executed until satisfying the condition of $\mathcal{J} \leq 0.0025$. Machine learning computations are executed on a TESLA V100 graphic card with a memory clock rate of 1.38 GHz. To validate predictions of PIPN and generate sparse labeled data at virtual sensor locations, we employ the COMSOL software (see e.g., Azad, Li, Verba, Ideker, & Isgor, 2016; Jafari, Vahab, Broumand, & Khalili, 2023; Pirnia, Duhaime, Ethier, & Dubé, 2019; Shi, Rui, Xu, Wang, & Wang, 2022) to solve Eqs. (1)–(2) using a finite element method (see e.g., Kashefi, 2020; Kashefi & Staples, 2018). Alternatively, sparse observations can be practically obtained by lab experimental techniques (Bultreys et al., 2022; Karlsons et al., 2022; Sabbagh, Kazemi, Soltani, & Nobes, 2020). Additionally, grid vertices of the generated finite element meshes are taken to construct point clouds as the input of PIPN.

Finally, we address two points. First, PIPN was primarily designed for predicting the solutions of desired PDEs on multiple sets of irregular geometries, simultaneously. In this research; however, we employ PIPN for predicting the porous medium flows on a single geometry. Conceptually, one may even use regular PINNs for this purpose. Nevertheless, Kashefi and Mukerji (2022) showed that PIPN was more stable and required fewer inquiry points (N) even for training on a single geometry, compared to a regular PINN. These two features reduce computational costs by default. One may refer to Section 4.1.6 of Kashefi and Mukerji (2022) for a comprehensive comparison between PIPN and regular PINNs. Second, the size of PIPN is scalable and can be conveniently adjusted depending on the size of inquiry points (N). In this study, for example, we use a relatively larger PIPN for porous media with smaller spatial correlation lengths.

3. Results and discussion

3.1. General analysis

A visual comparison between the ground truth and the PIPN prediction for the velocity and pressure fields is made for the porous media with spatial correlation lengths (l_c) of 1.7 mm, 0.9 mm, and 0.5 mm respectively in Figs. 4–6. All in all, a good agreement between the prediction and ground truth at the point cloud locations is observed. In all three cases, the maximum local errors for the velocity fields happen in the narrowest bottlenecks, where the flow accelerates. This observation highlights the importance of conducting precise measurements with a sufficient number of sensors in these locations. Technically; however, this is possible only if the space in the narrowest channels is yet

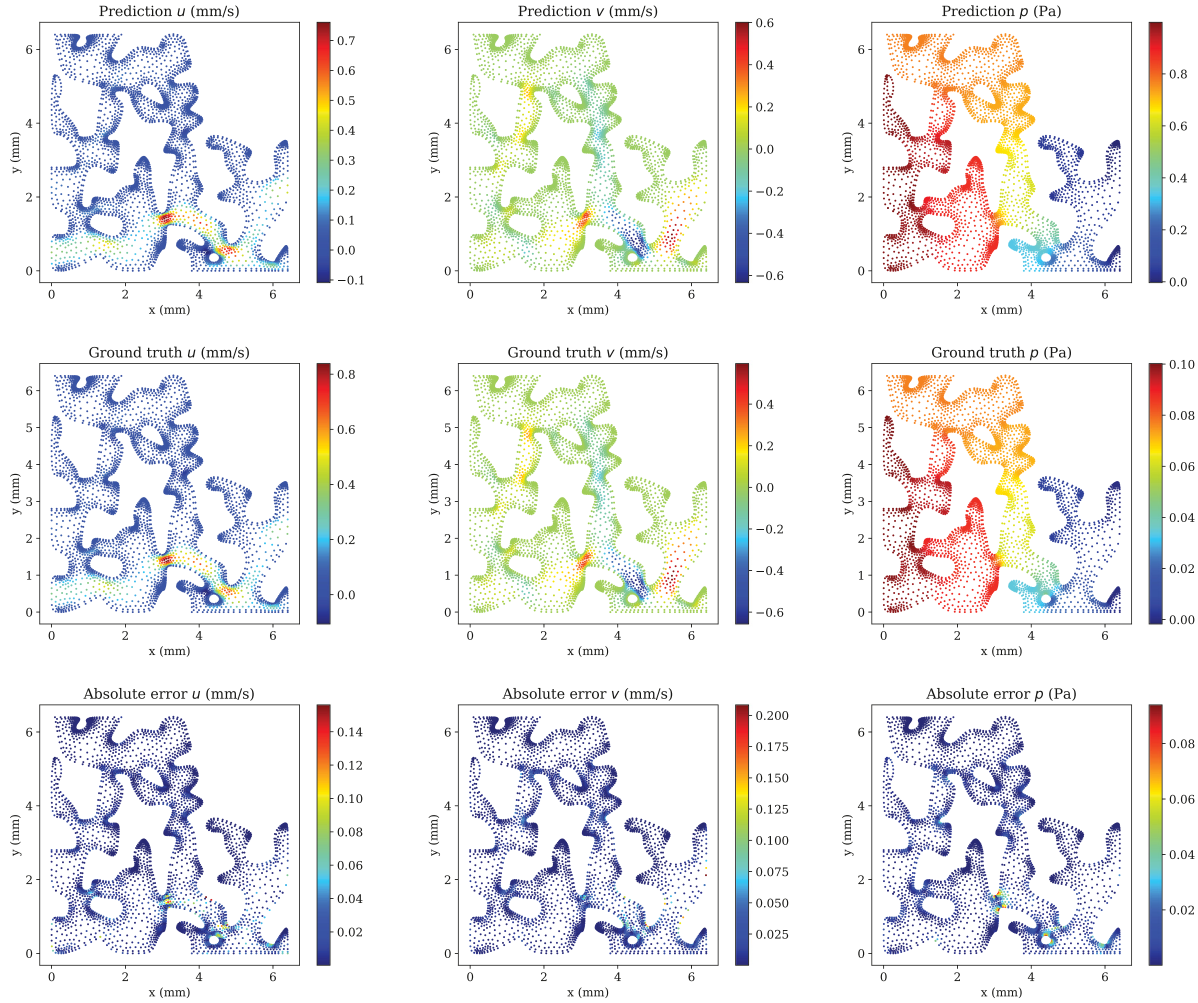

Fig. 4. Comparison between the ground truth by COMSOL and the prediction by PIPN for the velocity and pressure fields of the porous medium with the spatial correlation length of $l_c = 1.7$ mm.

Table 2
Error analysis of the velocity, pressure, and permeability predicted by PIPN for porous media with three different spatial correlation lengths; The L^2 norm is indicated by $\| \cdots \|$ and the absolute norm is shown by $| \cdots |$.

Spatial correlation length (l_c)	1.7 mm	0.9 mm	0.5 mm
$\frac{\|u-\tilde{u}\|}{\|u\|}$	1.51927E−1	1.54239E−1	1.79362E−1
$\frac{\|v-\tilde{v}\|}{\|v\|}$	1.84293E−1	1.47009E−1	1.79883E−1
$\frac{\|p-\tilde{p}\|}{\|p\|}$	1.63618E−2	7.34894E−3	9.96776E−3
$\frac{\|\mathcal{K}-\tilde{\mathcal{K}}\|}{\|\mathcal{K}\|}$	4.21557E−2	8.88220E−3	3.29516E−2
Number of iterations (i.e., epochs)	113141	149819	162145
Wall time per iteration (i.e., epoch)	31.4 s	32.3 s	33.6 s

large enough. Presumably, if this was not the case for a specific situation, we would suggest a potential technique to handle this challenge as follows. After obtaining the velocity field predicted by PIPN, one may use this predicted solution as an initial guess for a numerical classical solver (e.g., COMSOL) and execute the solver. Note that because the prediction by PIPN experiences a low level of errors (as shown in Figs. 4–6) it might only take a few iterations for the solver to converge with an improvement in the accuracy of the velocity fields in the narrowest bottlenecks.

The order of accuracy of the velocity fields for all three porous media is approximately the same due to the fact that PIPN is enforced to satisfy the same criterion (i.e., $\mathcal{J} \leq 0.0025$). Nevertheless, the porous media with shorter spatial correlation lengths (l_c) requires a higher number of iterations due to having a higher number of inquiries and more complicated geometry, as listed in Table 2. Note that the wall time consumed per epoch is approximately equal for all three porous media under investigation. This is because the TensorFlow (Abadi et al., 2016) software constructs the computation graph only once at the beginning of the training and uses it in the rest of the training for calculations. In fact, although the computation graph becomes more complex by decreasing the spatial correlation length, the computation per iteration (i.e., per epoch) only slightly (a few seconds) enhances, as TensorFlow (Abadi et al., 2016) simply updates numbers in the previously constructed graph to calculate the loss value.

To validate more precisely the PIPN performance, we tabulate the relative pointwise error (L^2 norm) of the PIPN prediction in Table 2. Accordingly, the relative error of the velocity fields is

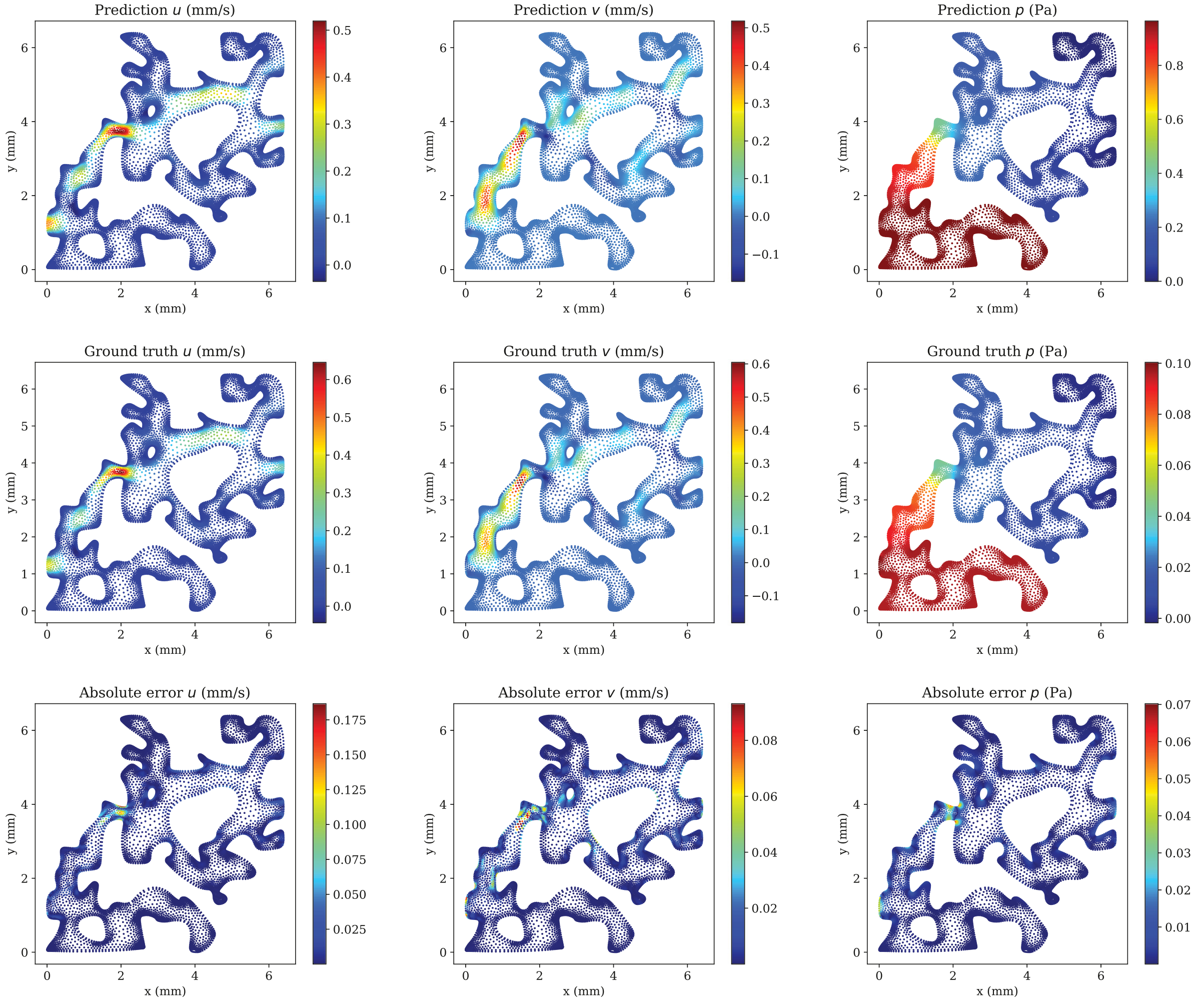

Fig. 5. Comparison between the ground truth by COMSOL and the prediction by PIPN for the velocity and pressure fields of the porous medium with the spatial correlation length of $l_c = 0.9$ mm.

approximately in the range of 15% to 19%. In all the cases, the predicted velocity fields experience a higher level of errors compared to the pressure one due to the fact that the pressure simply linearly decreases in the x-direction, whereas the velocity pattern is more complicated. Additionally, the velocity vector is involved in both the mass and momentum balances, while the pressure only plays a role in the momentum equation (Eq. (2)). According to Table 2, the relative error of the obtained permeability (see Eq. (3)) of the porous media as a result of the predicted velocity field is less than 5% for all three cases. Note that the average spatial distance of sensors from each other (d_s) indicates the level of information that we observe for each porous medium. For the porous media with $l_c = 1.7$ mm and $l_c = 0.9$ mm, $d_s = 0.237$ mm, while for the porous medium with $l_c = 0.5$ mm, $d_s = 0.16$ mm. Hence, we observe more information for the medium with $l_c = 0.5$ mm compared to the two others. As mentioned earlier, these reported d_s are the maximum possible distance for satisfying the convergence criterion in PIPN (i.e., $\mathcal{J} \leq 0.0025$).

3.2. Effect of sparse pressure observations

Next, we investigate the influence of the pressure observation on the prediction accuracy of PIPN. This investigation is important and critical because sparse observations of the pressure field may not be available for various reasons such as technical difficulties or expensive lab experiments. To this end, we modify the loss function by dropping the pressure observation residual $r^{\text{pressure}_{\text{obs}}}$, in the loss function yielding

$$\mathcal{J} = \lambda_1 r^{\text{continuity}} + \lambda_2 r^{\text{momentum}_x} + \lambda_3 r^{\text{momentum}_y} + \lambda_4 r^{\text{velocity}_{\text{wall}}} + \lambda_5 r^{\text{velocity}_{\text{obs}}}. \tag{12}$$

Similar to the previous subsection, we set $\lambda_1 = 100$ s, $\lambda_2 = 100$ m^3/N, $\lambda_3 = 100$ m^3/N, $\lambda_5 = 100$ s/m, and $\lambda_4 = 1$ s/m. We carry out this machine learning experiment for the porous media with the spatial correlation length of $l_c = 1.7$ mm, $l_c = 0.9$ mm, and $l_c = 0.5$ mm; and the outcomes are tabulated in Table 3.

For example, let us discuss the results obtained for the porous medium with the spatial correlation length of $l_c = 1.7$ mm. As a consequence of this modification, the relative error of the predicted velocity field in the x and y directions and the predicted pressure field become 1.43444E−1, 1.71549E−1, and 1.25499, respectively. We observe that although the relative error of the predicted pressure field increases enormously (by 667.024%), the accuracy of the predicted velocity field remains approximately unchanged. A similar observation has been reported by Kashefi and Mukerji (2022) for the natural convection problem. Reasons for this evidence have been articulated in detail both from applied mathematics and machine learning perspectives in Section 4.2.1 of Kashefi and Mukerji (2022). But in a nutshell, it can be explained as follows. PIPN is able to preserve the accuracy of the

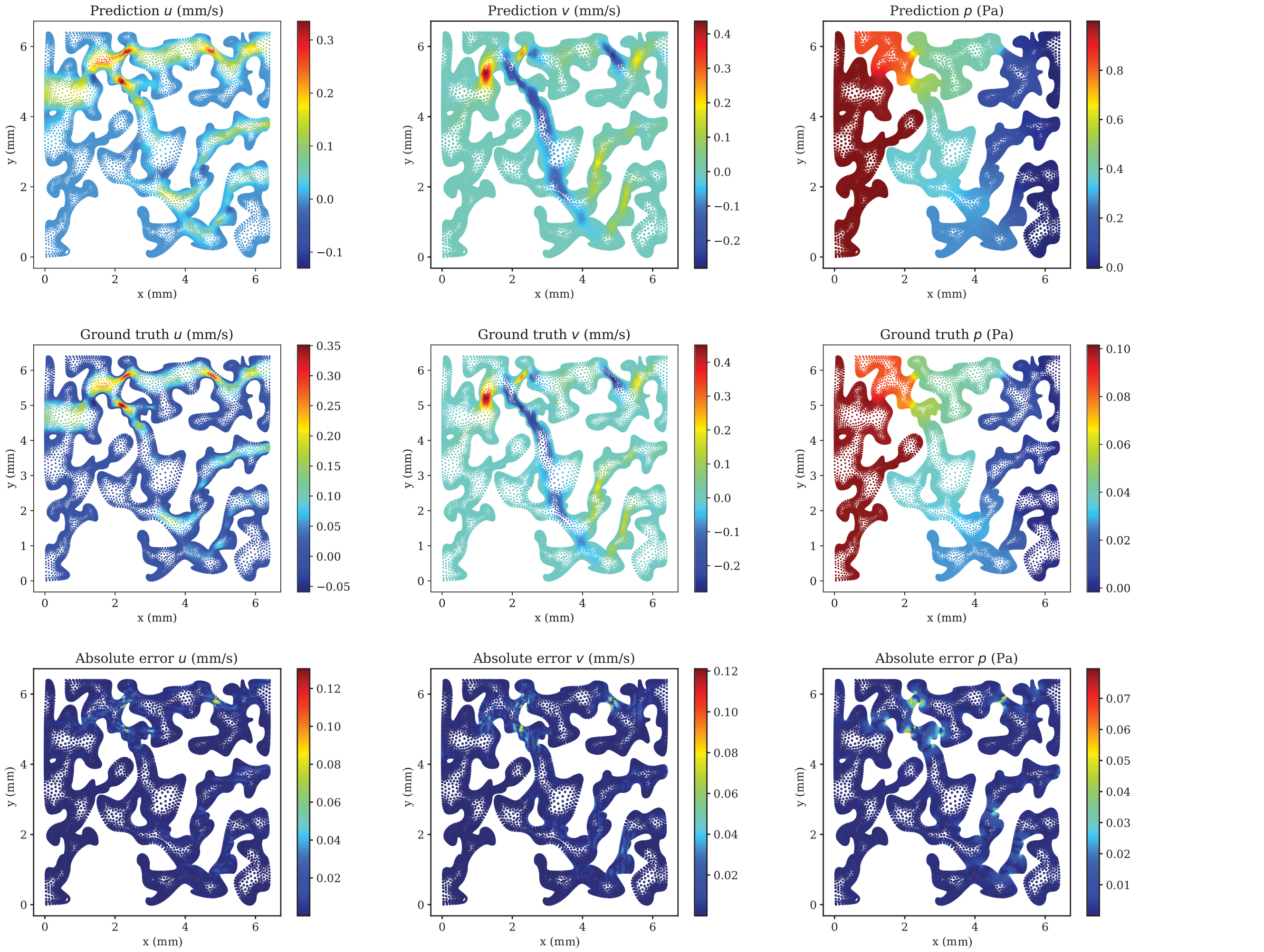

Fig. 6. Comparison between the ground truth by COMSOL and the prediction by PIPN for the velocity and pressure fields of the porous medium with the spatial correlation length of $l_c = 0.5$ mm.

Table 3
Investigation of the effect of the absence of the pressure measurements in the loss function (see Eq. (12)) and the resulting relative errors of the velocity, pressure, and permeability predicted by PIPN for porous media with three different spatial correlation lengths; The L^2 norm is indicated by $\| \cdots \|$.

Spatial correlation length (l_c)	1.7 mm	0.9 mm	0.5 mm
$\frac{\|u-\tilde{u}\|}{\|u\|}$	1.43444E−1	1.99840E−1	1.99135E−1
$\frac{\|v-\tilde{v}\|}{\|v\|}$	1.71549E−1	1.87410E−1	1.84841E−1
$\frac{\|p-\tilde{p}\|}{\|p\|}$	1.25499	7.05858E−1	6.64332E−1
Number of iterations (i.e., epochs)	15519	32601	41426
Wall time per iteration (i.e., epoch)	30.9 s	31.5 s	33.4 s

pressure gradient in the absence of pressure observations in the Stokes equations (Eqs. (1)–(2)). This feature of PIPN relies on the fact that pressure is an implicit variable in the steady Stokes equations (Timmermans, Minev, & Van De Vosse, 1996). Similar information can be observed and realized for the porous media with the spatial correlation length of $l_c = 0.9$ mm and $l_c = 0.5$ mm, as can be inferred in Table 3.

By comparing the information presented in Tables 2 and 3, the number of required iterations (i.e., epochs) for convergence satisfaction decreases for the current test cases. This is because the PIPN loss function becomes less restricted by dropping the pressure observation residual ($r^{\text{pressure}_{\text{obs}}}$) from the loss function. Additionally, since the pressure term is omitted from the PIPN loss function, the computation graph associated with the loss function becomes slightly less complex, and consequently, the wall time expended per iteration (i.e., epoch) slightly lessens.

All in all, the outputs discussed in this subsection show that the PIPN methodology successfully reliably predicts the velocity fields of the porous media even in the absence of the sparse pressure data.

3.3. Effect of noisy data

Sensor measurements are usually polluted by noises. To mimic this scenario, for instance, we add 5% random Gaussian noise to the velocity and pressure observations at all sensor locations for the porous media with the spatial correlation length of $l_c =$

Table 4
Error analysis of the velocity, pressure, and permeability predicted by PIPN for porous media with three different spatial correlation lengths when the observed data is polluted with 5% Gaussian noise; The L^2 norm is indicated by $\|\cdots\|$.

Spatial correlation length (l_c)	1.7 mm	0.9 mm	0.5 mm
$\frac{\|u-\tilde{u}\|}{\|u\|}$	1.76658E−1	2.04158E−1	2.03771E−1
$\frac{\|v-\tilde{v}\|}{\|v\|}$	1.97164E−1	2.16941E−1	1.88708E−1
$\frac{\|p-\tilde{p}\|}{\|p\|}$	2.04703E−2	2.58180E−2	2.30487E−2
Number of iterations to reach a plateau	212401	240231	230171

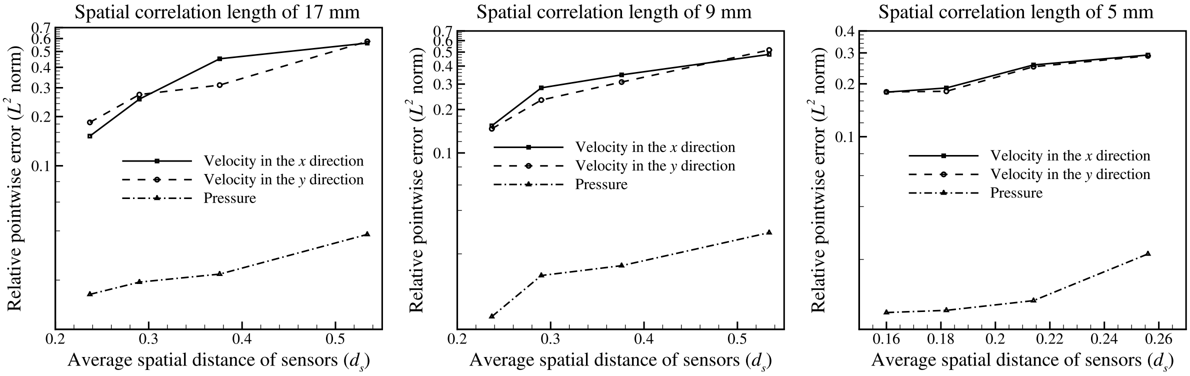

Fig. 7. Relative pointwise error (L^2 norm) as a function of the average spatial distance of sensors (d_s) for the porous media with the spatial correlation lengths (l_c) of 17 mm, 9 mm, and 5 mm.

1.7 mm, $l_c = 0.9$ mm, and $l_c = 0.5$ mm. The results and error analysis are tabulated in Table 4.

As can be realized from the information of Table 4, the error of the predicted velocity and pressure field increases as a result of adding noise to the sparse data. For example, considering the porous medium with the spatial correlation length of $l_c = 1.7$ mm, the relative pointwise error (L^2 norm) of the predicted u component of the velocity, v component of the velocity, and pressure fields become respectively 2.04158E−1, 2.16941E−1, and 2.58180E−2, indicating 16.278%, 6.983%, and 25.110% increase compared to the results obtained with noise-free data (see Tables 2 and 4).

Due to the presence of noise, the PIPN loss function is unable to satisfy the convergence criterion (i.e., $\mathcal{J} \leq 0.0025$) and this is why the PIPN predictions come with a higher level of errors compared to the deep learning experiment with the noise-free data (see Tables 2 and 4). In practice, we continue the training procedure until the loss value reaches a plateau. The corresponding number of iterations for each porous medium is listed in Table 4. By comparing Tables 2 and 4, it is realized that the number of iterations for reaching even a plateau (in the case of the noisy data) is greater than the number of required iterations for satisfying the convergence criterion (in case of the clean data), demonstrating that the noisy data demands more computational costs for the PIPN deep learning solver. Having said that, the relative pointwise error (L^2 norm) is less than approximately 22% for the velocity and pressure variables for all the porous media mentioned in Table 4. By and large, it is concluded that the PIPN methodology is robust even in the presence of noisy sensor data observed in the porous media.

3.4. Effect of the average spatial distance of sensors

In this section, the effect of the average spatial distance of sensors (d_s) on the PIPN prediction accuracy is investigated. To this end, we plot the relative pointwise error (L^2 norm) as a function of d_s for the porous media with the spatial correlation length of $l_c = 17$ mm, $l_c = 9$ mm, and $l_c = 5$ mm in Fig. 7. As can be observed in Fig. 7, by increasing d_s (i.e., decreasing the number of sensors), the relative pointwise error (L^2 norm) increases. For instance, by increasing d_s from 0.237 mm to 0.534 mm in the porous medium with the spatial correlation length (l_c) of 17 mm, the relative pointwise error (L^2 norm) of the velocity in the x direction, the velocity in the y direction, and the pressure field approximately increases by 270%, 212%, and 132%. In fact, the results shown plotted in Fig. 7 reflects the importance of the role of sparse observations in the accuracy of the PIPN outputs.

Now, the question is by increasing d_s how the prediction error spatially grows in the porous medium domain? To answer this question, we show the absolute pointwise errors of the velocity and pressure fields predicted by the PIPN methodology for different values of d_s, for example, for the porous media with the spatial correlation length (l_c) of 17 mm and 9 mm, respectively, in Figs. 8 and 9. Furthermore, the sensor locations associated with each d_s are exhibited in Figs. 8 and 9. In all the cases, maximum local errors happen in areas where the fluid flows, specifically with higher accelerations. By increasing d_s (i.e., decreasing sensor numbers), both the absolute value of the pointwise errors as well as the spatial areas polluted with significant errors in the porous media increase. After performing this machine learning experiment, at the first glance, it seems that one should set more sensors in the area where fluid flows and accelerates. However, the issue with this strategy is that these specific areas are unknown to us before executing the PIPN solver. Consequently, it is reasonable to spread the sensors at equal distances from each other throughout the entire space of a porous medium of interest.

4. Conclusions and future directions

In this research letter, we applied PIPN (Kashefi & Mukerji, 2022) as an innovative physics-informed deep learning strategy for the prediction of the velocity and pressure fields of two-dimensional steady incompressible flows in porous media, while only a sparse scattered set of labeled data were observed. Using

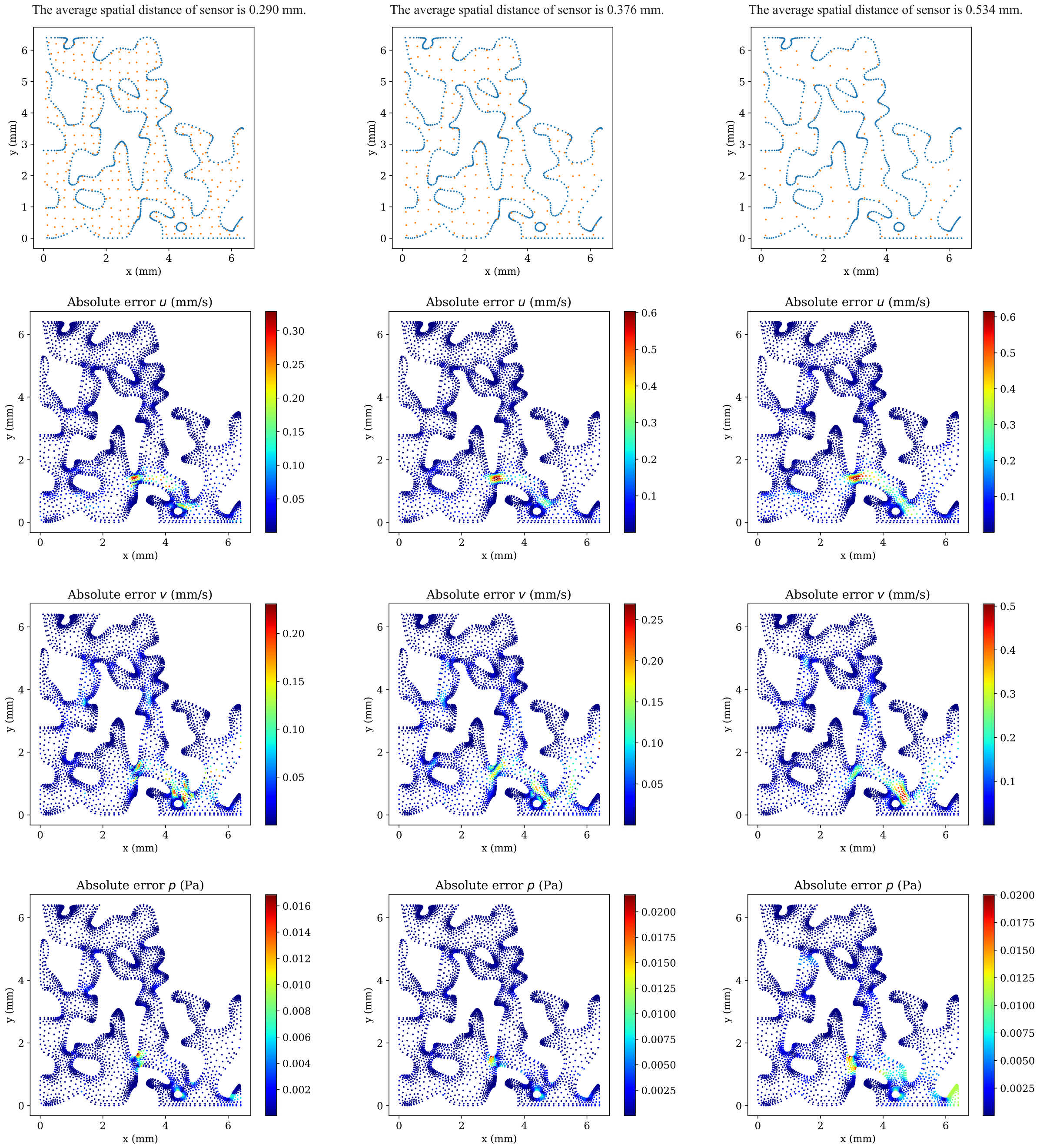

Fig. 8. Absolute pointwise error of the velocity and pressure fields predicted by PIPN for the porous medium with the spatial correlation length (l_c) of 17 mm when the average spatial distance of sensors (d_s) is 0.290 mm, 0.376 mm, and 0.534 mm. The solution by COMSOL is considered as the reference ground truth to compute the absolute pointwise error.

PIPN, first, we lessened the required computational memories by not taking the grain spaces of porous media into the machine learning framework. Second, the point cloud allowed us to represent the geometry of pore spaces of porous media more realistically. Third, we had the freedom to vary the spatial resolution of pores spaces to optimize the computational costs. Specifically, the effect of noisy sensor data, pressure observations, and spatial correlation lengths was investigated through visual results and quantitative error analysis.

One of our outlook projects is the extension of PIPN to three-dimensional and multiphase flows in porous media. More specifically, there are currently serious challenges for three-dimensional modeling (e.g., see Saxena et al., 2017). By switching to a three-dimensional space, the number of points (N) in point clouds of porous media significantly increases and it leads to more complicated and lengthier computation graphs in PIPN, requiring larger GPU memories and longer wall clock time for running the PIPN platform. Additionally, it is conjectured that the accuracy of predictions by PIPN may be more sensitive to the number of sensors and the spatial distribution of sensors for the inverse problem in a three-dimensional space. One approach to overcome these barriers is to moderate the associated computational costs via parallel computing and domain decomposition techniques (Shukla, Jagtap, & Karniadakis, 2021).

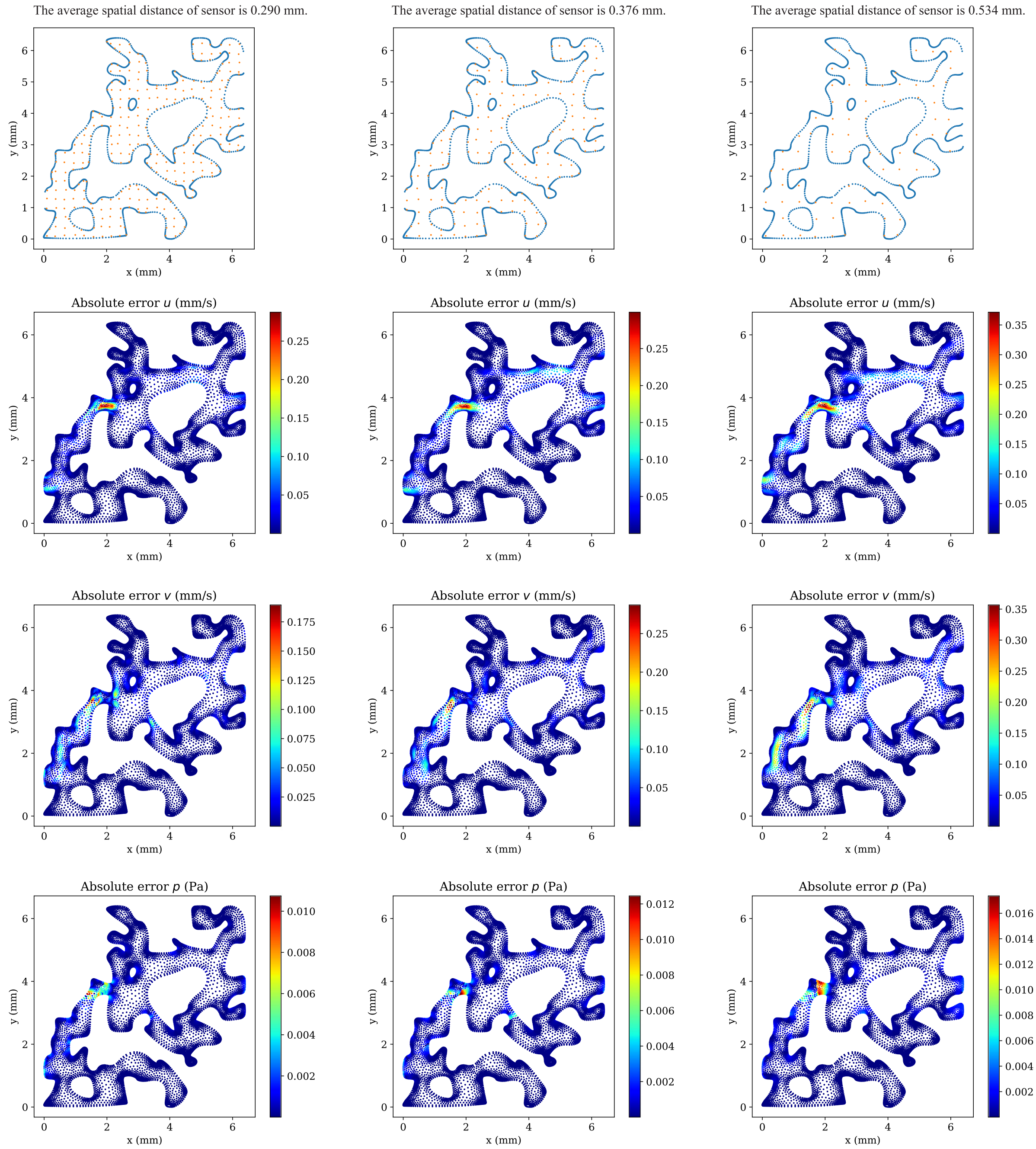

Fig. 9. Absolute pointwise error of the velocity and pressure fields predicted by PIPN for the porous medium with the spatial correlation length (l_c) of 9 mm when the average spatial distance of sensors (d_s) is 0.290 mm, 0.376 mm, and 0.534 mm. The solution by COMSOL is considered as the reference ground truth to compute the absolute pointwise error.

CRediT authorship contribution statement

Ali Kashefi: Conceptualization, Methodology, Software, Formal analysis, Visualization, Writing – original draft, Writing – review & editing. **Tapan Mukerji:** Conceptualization, Methodology, Formal analysis, Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to the code in the manuscript.

Acknowledgments

Funding from Shell-Stanford collaborative project on Digital Rock Physics 2.0 is acknowledged for supporting this research project. Moreover, we are thankful to the Stanford Research Computing Center for computational resources. Additionally, the authors would like to thank the reviewers for their beneficial comments and suggestions.

Computer code availability

The software and data are available on the following GitHub repository:

https://github.com/Ali-Stanford/PhysicsInformedPointNetPorousMedia.

- Name of code: PIPN_for_Porous_Media.py
- Developer and contact address: Ali Kashefi (kashefi@stanford.edu)
- Year first available: 2022
- Hardware required: Graphics Processing Unit (GPU) with at least 48 GB of RAM
- Software required: TensorFlow; Python; Matplotlib; Numpy
- Program language: Python
- Program size: 600 lines

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

Al-Zubaidi, F., Mostaghimi, P., Niu, Y., Armstrong, R. T., Mohammadi, G., McClure, J. E., et al. (2023). Effective permeability of an immiscible fluid in porous media determined from its geometric state. *Physical Review Fluids*, *8*(6), Article 064004.

Alhubail, A., He, X., AlSinan, M., Kwak, H., & Hoteit, H. (2022). Extended physics-informed neural networks for solving fluid flow problems in highly heterogeneous media. In *International petroleum technology conference*. OnePetro.

Alqahtani, N., Alzubaidi, F., Armstrong, R. T., Swietojanski, P., & Mostaghimi, P. (2020). Machine learning for predicting properties of porous media from 2D X-ray images. *Journal of Petroleum Science and Engineering*, *184*, Article 106514.

Azad, V. J., Li, C., Verba, C., Ideker, J. H., & Isgor, O. B. (2016). A COMSOL–GEMS interface for modeling coupled reactive-transport geochemical processes. *Computers & Geosciences*, *92*, 79–89.

Berg, C. F. (2014). Permeability description by characteristic length, tortuosity, constriction and porosity. *Transport in Porous Media*, *103*(3), 381–400.

Bordignon, F., Figueiredo, L., Exterkoetter, R., Rodrigues, B. B., & Correia, M. (2019). Deep learning for grain size and porosity distributions estimation on micro-CT images. In *Proceedings of the 16th international congress of the Brazilian geophysical society & expogef* (pp. 1–6).

Bultreys, T., Van Offenwert, S., Goethals, W., Boone, M. N., Aelterman, J., & Cnudde, V. (2022). X-ray tomographic micro-particle velocimetry in porous media. *Physics of Fluids*, *34*(4), Article 042008.

Da Wang, Y., Armstrong, R. T., & Mostaghimi, P. (2019). Enhancing resolution of digital rock images with super resolution convolutional neural networks. *Journal of Petroleum Science and Engineering*, *182*, Article 106261.

Da Wang, Y., Shabaninejad, M., Armstrong, R. T., & Mostaghimi, P. (2020). Physical accuracy of deep neural networks for 2D and 3D multi-mineral segmentation of rock micro-CT images. arxiv 2020. arXiv preprint arXiv:2002.05322.

Darcy, H. (1856). *Les fontaines publiques de la ville de Dijon: Exposition et application des principes à suivre et des formules à employer dans les questions de distribution d'eau: vol. 1*, Victor dalmont.

Eshghinejadfard, A., Daróczy, L., Janiga, G., & Thévenin, D. (2016). Calculation of the permeability in porous media using the lattice Boltzmann method. *International Journal of Heat and Fluid Flow*, *62*, 93–103.

Graczyk, K. M., & Matyka, M. (2020). Predicting porosity, permeability, and tortuosity of porous media from images by deep learning. *Scientific Reports*, *10*(1), 1–11.

Hassanizadeh, S. M., & Gray, W. G. (1987). High velocity flow in porous media. *Transport in Porous Media*, *2*, 521–531.

Hong, J., & Liu, J. (2020). Rapid estimation of permeability from digital rock using 3D convolutional neural network. *Computational Geosciences*, *24*(4), 1523–1539.

Jafari, A., Vahab, M., Broumand, P., & Khalili, N. (2023). An extended finite element method implementation in COMSOL multiphysics: Thermo-hydro-mechanical modeling of fluid flow in discontinuous porous media. *Computers and Geotechnics*, *159*, Article 105458.

Kamrava, S., Sahimi, M., & Tahmasebi, P. (2021). Simulating fluid flow in complex porous materials by integrating the governing equations with deep-layered machines. *NPJ Computational Materials*, *7*(1), 1–9.

Karimpouli, S., & Tahmasebi, P. (2019a). Image-based velocity estimation of rock using convolutional neural networks. *Neural Networks*, *111*, 89–97.

Karimpouli, S., & Tahmasebi, P. (2019b). Segmentation of digital rock images using deep convolutional autoencoder networks. *Computers & Geosciences*, *126*, 142–150.

Karlsons, K., de Kort, D., Alpak, F., Dietderich, J., Freeman, J., Appel, M., et al. (2022). Integrating pore-scale flow MRI and X-ray μCT for validation of numerical flow simulations in porous sedimentary rocks. *Transport in Porous Media*, *143*(2), 373–396.

Kashefi, A. (2020). A coarse-grid incremental pressure projection method for accelerating low Reynolds number incompressible flow simulations. *Iran Journal of Computer Science*, *3*(1), 13–23.

Kashefi, A., & Mukerji, T. (2021). Point-cloud deep learning of porous media for permeability prediction. *Physics of Fluids*, *33*(9), Article 097109.

Kashefi, A., & Mukerji, T. (2022). Physics-informed PointNet: A deep learning solver for steady-state incompressible flows and thermal fields on multiple sets of irregular geometries. *Journal of Computational Physics*, *468*, Article 111510.

Kashefi, A., Rempe, D., & Guibas, L. J. (2021). A point-cloud deep learning framework for prediction of fluid flow fields on irregular geometries. *Physics of Fluids*, *33*(2), Article 027104.

Kashefi, A., & Staples, A. E. (2018). A finite-element coarse-grid projection method for incompressible flow simulations. *Advances in Computational Mathematics*, *44*(4), 1063–1090.

Kharazmi, E., Zhang, Z., & Karniadakis, G. E. (2021). hp-VPINNs: Variational physics-informed neural networks with domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, *374*, Article 113547.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Lantuéjoul, C. (2001). *Geostatistical simulation: Models and algorithms*. (1139), Springer Science & Business Media.

Liu, M., & Mukerji, T. (2022). Multiscale fusion of digital rock images based on deep generative adversarial networks. *Geophysical Research Letters*, *49*(9), Article e2022GL098342.

Lu, L., Meng, X., Cai, S., Mao, Z., Goswami, S., Zhang, Z., et al. (2022). A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data. *Computer Methods in Applied Mechanics and Engineering*, *393*, Article 114778.

Meng, X., Li, Z., Zhang, D., & Karniadakis, G. E. (2020). PPINN: Parareal physics-informed neural network for time-dependent PDEs. *Computer Methods in Applied Mechanics and Engineering*, *370*, Article 113250.

Niu, Y., Da Wang, Y., Mostaghimi, P., McClure, J. E., Yin, J., & Armstrong, R. T. (2021). Geometrical-based generative adversarial network to enhance digital rock image quality. *Physical Review A*, *15*(6), Article 064033.

Niu, Y., Mostaghimi, P., Shabaninejad, M., Swietojanski, P., & Armstrong, R. T. (2020). Digital rock segmentation for petrophysical analysis with reduced user bias using convolutional neural networks. *Water Resources Research*, *56*(2), Article e2019WR026597.

Pang, G., D'Elia, M., Parks, M., & Karniadakis, G. E. (2020). nPINNs: Nonlocal physics-informed neural networks for a parametrized nonlocal universal Laplacian operator. Algorithms and applications. *Journal of Computational Physics*, *422*, Article 109760.

Pang, G., Lu, L., & Karniadakis, G. E. (2019). fPINNs: Fractional physics-informed neural networks. *SIAM Journal on Scientific Computing*, *41*(4), A2603–A2626.

Phan, J., Ruspini, L. C., & Lindseth, F. (2021). Automatic segmentation tool for 3D digital rocks by deep learning. *Scientific Reports*, *11*(1), 1–15.

Pirnia, P., Duhaime, F., Ethier, Y., & Dubé, J. (2019). ICY: An interface between COMSOL multiphysics and discrete element code YADE for the modelling of porous media. *Computers & Geosciences*, *123*, 38–46.

Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 652–660).

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, *378*, 686–707.

Sabbagh, R., Kazemi, M. A., Soltani, H., & Nobes, D. S. (2020). Micro-and macro-scale measurement of flow velocity in porous media: A shadow imaging approach for 2D and 3D. *Optics*, *1*(1), 6.

Santos, J. E., Xu, D., Jo, H., Landry, C. J., Prodanović, M., & Pyrcz, M. J. (2020). PoreFlow-Net: A 3D convolutional neural network to predict fluid flow through porous media. *Advances in Water Resources*, *138*, Article 103539.

Saxena, N., Hofmann, R., Alpak, F. O., Berg, S., Dietderich, J., Agarwal, U., et al. (2017). References and benchmarks for pore-scale flow simulated using micro-CT images of porous media and digital rocks. *Advances in Water Resources*, *109*, 211–235.

Shi, Y., Rui, S., Xu, S., Wang, N., & Wang, Y. (2022). COMSOL modeling of heat transfer in SVE process. *Environments*, *9*(5), 58.

Shukla, K., Jagtap, A. D., & Karniadakis, G. E. (2021). Parallel physics-informed neural networks via domain decomposition. *Journal of Computational Physics*, *447*, Article 110683.

Tartakovsky, A. M., Marrero, C. O., Perdikaris, P., Tartakovsky, G. D., & Barajas-Solano, D. (2018). Learning parameters and constitutive relationships with physics informed deep neural networks. arXiv preprint arXiv:1808.03398.

Timmermans, L. J. P., Minev, P. D., & Van De Vosse, F. N. (1996). An approximate projection scheme for incompressible flow using spectral elements. *International Journal for Numerical Methods in Fluids*, *22*(7), 673–688.

Wang, Y. D., Armstrong, R. T., & Mostaghimi, P. (2020). Boosting resolution and recovering texture of 2D and 3D micro-CT images with deep learning. *Water Resources Research*, *56*(1), Article e2019WR026052.

Wang, K., Chen, Y., Mehana, M., Lubbers, N., Bennett, K. C., Kang, Q., et al. (2021). A physics-informed and hierarchically regularized data-driven model for predicting fluid flow through porous media. *Journal of Computational Physics*, *443*, Article 110526.

Wang, Y. D., Chung, T., Armstrong, R. T., & Mostaghimi, P. (2021). ML-LBM: Predicting and accelerating steady state flow simulation in porous media with convolutional neural networks. *Transport in Porous Media*, *138*(1), 49–75.

Wu, J., Yin, X., & Xiao, H. (2018). Seeing permeability from images: Fast prediction with convolutional neural networks. *Science Bulletin*, *63*(18), 1215–1222.

Xiang, Z., Peng, W., Zheng, X., Zhao, X., & Yao, W. (2021). Self-adaptive loss balanced physics-informed neural networks for the incompressible Navier-Stokes equations. arXiv preprint arXiv:2104.06217.

Xu, W., & Journel, A. G. (1993). GTSIM: Gaussian truncated simulations of reservoir units in a West Texas carbonate field. *Paper SPE*, *27412*.

Yang, L., Meng, X., & Karniadakis, G. E. (2021). B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, *425*, Article 109913.