From Computational Fluid Dynamics to Structure Interpretation via Neural Networks: An Application to Flow and Transport in Porous Media

I&EC research
Industrial & Engineering Chemistry Research

pubs.acs.org/IECR

Article

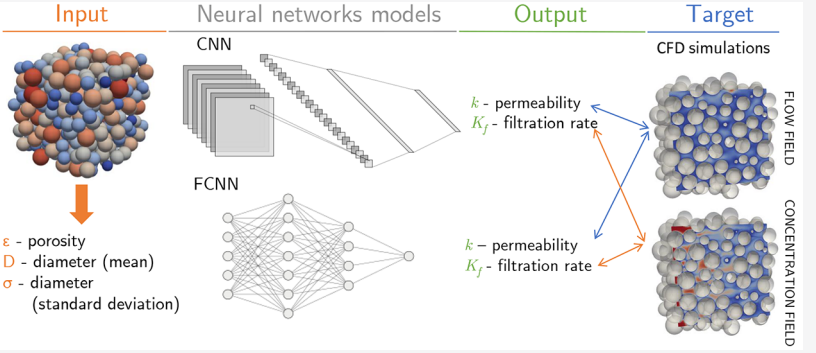
From Computational Fluid Dynamics to Structure Interpretation via Neural Networks: An Application to Flow and Transport in Porous Media

Agnese Marcato, Gianluca Boccardo,* and Daniele Marchisio

Cite This: https://doi.org/10.1021/acs.iecr.1c04760

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: The modeling of flow and transport in porous media is of the utmost importance in many chemical engineering applications, including catalytic reactors, batteries, and CO_2 storage. The aim of this study is to test the use of fully connected (FCNN) and convolutional neural networks (CNN) for the prediction of crucial properties in porous media systems: the permeability and the filtration rate. The data-driven models are trained on a dataset of computational fluid dynamics (CFD) simulations. To this end, the porous media geometries are created *in silico* by a discrete element method, and a rigorous setup of the CFD simulations is presented. The models trained have as input both geometrical and operating conditions features so that they could find application in multiscale modeling, optimization problems, and in-line control. The average error on the prediction of the permeability is lower than 2.5%, and that on the prediction of the filtration rate is lower than 5% in all the neural networks models. These results are achieved with at least a dataset of ∼100 CFD simulations.

INTRODUCTION

The study of flow and transport in porous media is of the utmost importance in chemical engineering.[1] Many fields of application in the traditional chemical industry are impacted by the understanding of these phenomena, such as packed beds catalytic reactors,[2,3] and filtration devices.[4,5] Other examples can be found in large-scale environmental applications, such as groundwater extraction and remediation[6,7] or enhanced oil recovery.[8] Apart from the importance of porous media in these established sectors, it is apparent how the study of transport phenomena in dispersed and random structures will play an increasingly important role in the transition toward a sustainable and carbon neutral economy. The consensus is that true carbon neutrality will not be achieved without the employ of carbon capture and storage solutions on a global scale.[9,10] As it is and was true for oil prospecting and extraction in the pretransition economy, it thus remains clear how a deep understanding of fluid permeation and mass transfer in porous rocks will be essential both to benefit carbon storage sites exploration and for planning safe and long-term CO_2 capture. One other prominent example of the pivotal importance in the energy transition is the growing attention to the study of energy storage systems, specifically batteries (both in their current Li-ion and future beyond-lithium incarnations). Indeed, electrodes, in their microscale reconstruction, are modeled as porous media impregnated by the electrolyte. The long-term safety of battery operation, energy density, and battery cycle life (which constitute the main areas of improvement inasmuch their limits constitute the main barriers to overcome toward a more pervasive grid electrification) are being investigated by means of detailed studies of the transport of electrochemical species inside the electrodes.[11−13]

Special Issue: Machine Learning and Data Science in Chemical Engineering

Received: December 7, 2021
Revised: February 2, 2022
Accepted: February 4, 2022

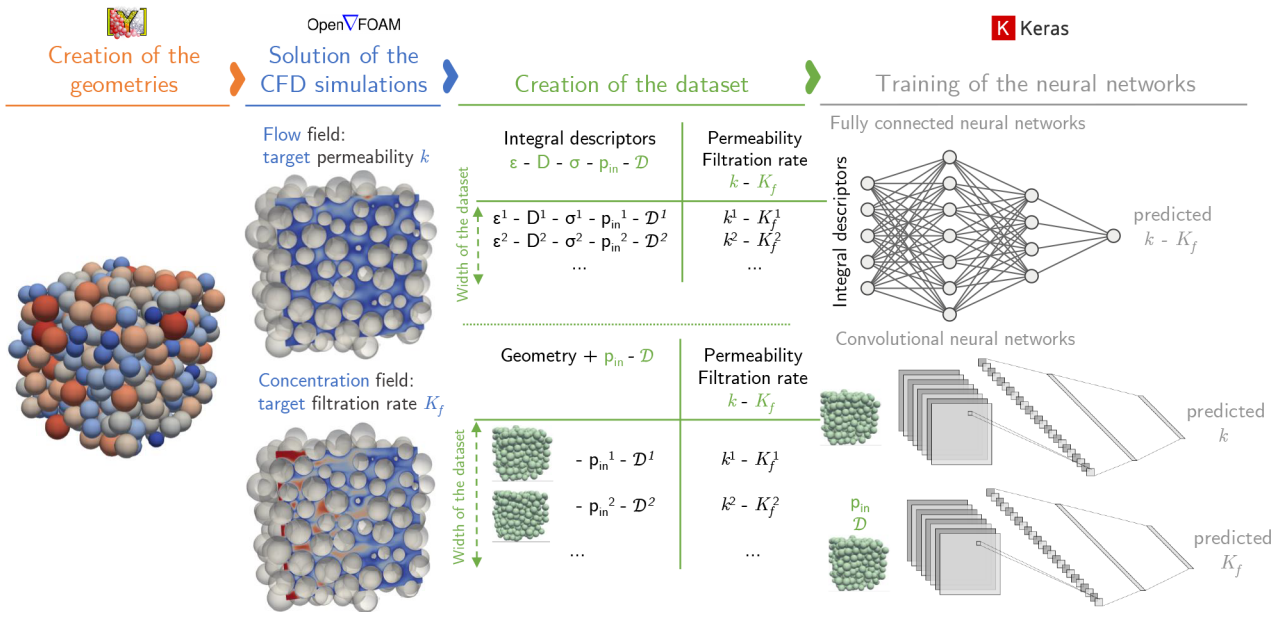


Figure 1. Workflow for the construction of neural networks models for porous media applications: (1) *in silico* creation of the geometries, (2) setup and solution of the CFD simulations, and (3) preparation of the dataset for the FCNN (integral descriptors as input and output) and for the CNN (entire geometry and integral descriptors as input and integral descriptors as output).

Traditional modeling is based on fundamental laws obtained from analytical and experimental methods resulting in expressions of dimensionless numbers and macroscale parameters that characterize the geometry of the porous media. The confidence of these laws is affected by uncertainties due to the nonlinear correlation between the defining features of the geometries and the resulting macroscale parameters, respectively, the input and output features of these models.[14] As a way to improve the confidence of these models and spurred on by the increasing availability of computational resources, in recent years a host of microscale models have been developed that completely describe porous media behavior at the pore scale.[15−18]

Computational fluid dynamics (CFD) can be employed for the microscale solution of the transport equations; in this way, a detailed model can be obtained for the investigated phenomena. CFD models were proposed for filtration,[19] reactors,[20,21] gasification processes,[22] and batteries[23] applications. The main issue related to CFD simulations is their computational cost; in fact, they are time-consuming and cannot easily be used in wide-ranging optimization workflows, multiscale modeling, or in-line control.

Even if chemical engineering is not traditionally a big-data field, at least by the standards dictated by the current artificial intelligence "explosion", by the use of simulations it is possible to obtain a dataset suitable for the training of a data-driven model.[24] In the past decade, machine learning algorithms spread in scientific research, since they are able to deal with highly nonlinear problems that would be difficultly described by first-principles methods.[25−28] In this way, data can be processed in an optimal way extracting information from them, moreover a trained data driven model can be easily used for optimization and multiscale modeling thanks to its immediate response. Neural networks are one of the most commonly employed machine learning techniques in last years.[29,30]

Datasets obtained from CFD simulations can be used for the optimization of devices[31,32] and for the prediction of fluid dynamics properties and performances.[33−35] In the porous media field, neural networks models can be developed to deal with the randomness of geometries at the pore scale as proposed by Partopour et al.[36] The majority of recent works developed neural networks models and workflows for the prediction of the permeability or geometrical parameters starting from data extracted from Lattice−Boltzmann simulations. Fully connected neural networks (FCNN) were employed for the prediction of the permeability[37,38] and geometrical parameters[39] from integral and hand-selected input features. Convolutional neural networks (CNN) can be used for the prediction of the same objectives. In the latter case, the training of the model is performed by providing the CNN with an image of the entire geometry of the porous medium,[40,41] in this way, the network autonomously extracts the most effective features for the prediction of the output, and it is not necessary to hand-select the integral parameters. In recent years, CNN for the prediction of the flow field were proposed, and the permeability can be extracted from the predicted fields.[42−45] At the moment, little was done to use these modeling techniques to deal with more complex problems, such as reaction and transport in porous media, which is fundamental for chemical engineering applications. Albeit of central importance, the geometry is far from being the only defining factor, and neural networks should be able to deal with both geometrical description and operating conditions information together.

In our work, we test the use of FCNN and CNN for the prediction of the permeability and the filtration rate, which is an integral parameter related to the transport of a chemical species in a porous medium. This work applies (and expands) the workflow that we proposed in our previous article[46] and that is summarized in Figure 1. At first porous media geometries are created *in silico*, then the computational grids are built for the CFD simulations, that are in turn solved so as to prepare a dataset for the training. While for the FCNN case the neural network is provided with integral descriptors of the porous media geometries,[47] in the CNN case the entire geometry

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

together with the operating conditions input parameters are fed to the network model. Both these techniques give good predictions of the permeability and the filtration rate in porous media, with average errors lower than 5%. Beyond this similarity in performance, this proven effectiveness of CNN in building a predictive model for nontrivial transport/reaction phenomena, with no need for arbitrary and potentially unsuccessful parameter hand-picking, will have important consequences in the construction of a general model, as it will be expanded upon in the remainder of this work. In closing, we offer the results of a numerical exploration of the coupling between these deep-learning models and CFD; namely, we conducted a sensitivity study on the size of the dataset used in training the CNN in order to determine the minimum number of CFD simulations to be solved for the sake of obtaining an accurate data driven model, as well as a preliminary tuning of the relevant hyperparameters for the deep-learning model.

METHODS

Governing Equations. The physical system studied in this work is flow and transport in porous media. The characteristic dimension of the grains, i.e., their diameter, is in the order of hundreds of micrometers, so it is possible to model the fluid flow with the continuum approach.[48] The continuity equation (eq 1) and the Navier−Stokes equation (eq 2) are solved under the hypothesis of incompressible flow with constant density and viscosity:

$$\frac{\partial U_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial U_i}{\partial t} + U_j\frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\frac{\partial^2 U_i}{\partial x_j^2} \tag{2}$$

where U_i is the ith component of the velocity, p is the pressure, ρ is the fluid density, and ν is the kinematic viscosity.

A specific aspect of transport in porous media was tackled, namely, the problem of filtration of a colloid via attachment on the surface of the grains. The colloidal particles are injected in water-saturated porous media. Their dimension is assumed to be lower than 1 μm, and their density ranges generally between 5000 and 10 000 kg m^{-3}. The resulting Stokes number is thus lower than unity; as a consequence, the particles are assumed to move with the fluid at the same velocity. For a complete and detailed discussion of the hypothesis readers are referred to Boccardo et al.[19] These hypotheses allow the use of the advection−diffusion equation (eq 3) for the evaluation of the concentration field of the colloid in porous media:

$$\frac{\partial C}{\partial t} + U_j\frac{\partial C}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\mathcal{D}\frac{\partial C}{\partial x_j}\right) \tag{3}$$

where C is the colloid concentration and $\mathcal{D}$ is the diffusion coefficient of the colloid that is considered constant in space, and estimated by the Einstein equation:

$$\mathcal{D} = \frac{k_B T}{3\pi\mu d_p} \tag{4}$$

where k_B is the Boltzmann constant, T is the temperature, μ is the dynamic viscosity of the fluid, and d_p is the dimension of the colloid particles. The solution of eq 3 is also employed to investigate another relevant chemical engineering problem: the transport of a chemical species undergoing a infinitely fast chemical reaction occurring on the surface of the grains, i.e., catalytic particles. In this case, C is the chemical species concentration, and $\mathcal{D}$ represents its diffusion coefficient. A last relevant application, still governed by eq 3, is the lithiation and delithiation phenomena of Li-ion battery electrodes.

A computational fluid dynamics code which implements a finite-volume numerical method is used for the solution of eqs 1−3. The results were exploited for calculating two quantities of interest: the permeability of the porous media and the filtration rate of the colloid on the grains. The permeability was calculated by the Darcy's law:

$$\frac{\Delta P}{L} = \frac{\mu}{k}q \tag{5}$$

where ΔP is the pressure drop across the medium, L is the length of the medium in the direction of the fluid flow, and q is the superficial velocity of the fluid. The filtration rate was calculated by the following relation (eq 6) obtained by volume averaging the advection−diffusion equation:

$$K_f = \frac{F_{tot}^{in} - F_{tot}^{out}}{\langle C\rangle V} \tag{6}$$

where F_{tot} is the total flux (advective and diffusive) through the inlet and the outlet boundaries, $\langle C\rangle$ is the volume average concentration of the colloid, and V is the total volume of the medium. For the complete derivation of the relation, meant to solve some of the scale-dependence issues of the more classical colloid filtration laws,[6] refer to Boccardo et al.[49] For the other applications mentioned above, namely, infinitely fast catalytic chemical reaction and lithiation and delithiation phenomena, similar but alternative quantities can be defined.

Neural Networks. In this section, the theoretical background of neural networks modeling is reviewed with the aim of giving to the reader the basic insights into this field. For a complete discussion of the topic, we refer the reader to the book by Goodfellow et al.[50]

Neural networks are machine learning algorithms suited for the data-driven modeling of nonlinear and multivariable problems. The dataset available for the modeling is partitioned into three sets: the training set, which is used for the training of the neural network; the validation set, which is employed for online evaluation of the network performance; and the test set, which is used to evaluate the generalization capability of the trained network. The training is an optimization technique whose aim is to adjust the parameters of the neural network in order to minimize a loss function. Stochastic gradient descent is the most used training algorithm. The loss function to be minimized compare by a certain metric the predictions of the network with the true values of the dataset.

In this work, neural networks are trained for the prediction of integral quantities characterizing the porous media. Two approaches are tested: the use of FCNN for the prediction of the output starting from a set of input features and the use of CNN for the prediction of the same output from images of the porous media.

Neural networks differ in their architecture and the kind of input data they manage, where the simplest ones are the FCNN which take numerical values as input data, called features. The building blocks of a FCNN are the neurons which are arranged in layers: the input layer, with a number of neurons equal to the number of input features; a variable number of hidden layers; and an output layer, which contains a number of neurons equal

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

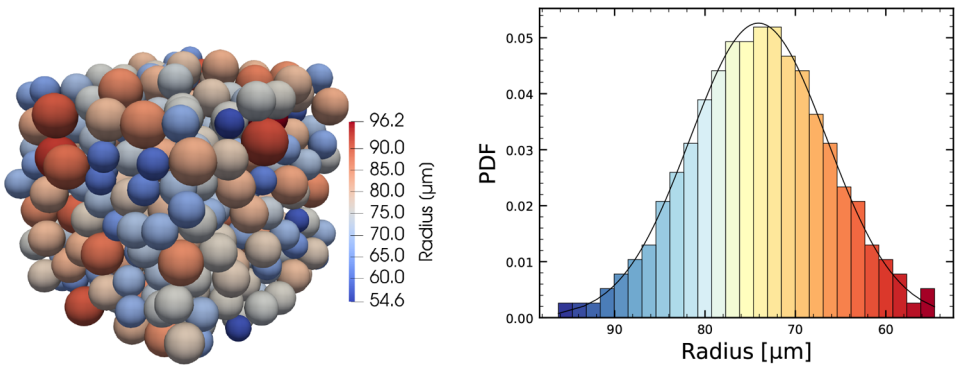


Figure 2. Porous media geometries were created *in silico* by means of YADE with a Gaussian particle size distribution.

to the number of outputs of the network, equal to one in our case study. The mathematical calculations that a neuron performs are quite simple and can be summarized as follows:

$$y_j^{(k)} = f\left(b^{k-1} + \sum_{i=1}^{m_{k-1}} w_{i,j}^{(k)} y_i^{(k-1)}\right) \tag{7}$$

where $y_j^{(k)}$ is the output of the jth neuron of the kth layer, m_{k-1} is the number of neurons in the previous layer, $w_{i,j}^{k}$ is the weight associated with the ith neuron of the previous layer, and b_{k-1} is the bias of the previous layer. The core of the neuron is the activation function f, for which in this work we chose the rectified linear unit ($\mathrm{relu}(z) = \max(0, z)$) since it is one of the less expensive from the computational point of view and does not suffer from the vanishing gradients problem in deep neural networks models.[51]

CNN[52] take gridlike topologies as input features, for example, bidimensional images or three-dimensional voxelizations; the latter is the case in this work. In convolutional layers, neurons are arranged in a number of n-dimensional kernels, the filters, that are applied multiple times to the input space. As a consequence, the number of trainable parameters is decreased, since each neuron of the kernel is applied at every input position, this characteristic is named parameter sharing. Another outcome is the equivariance to translation property; in fact, convolutional layers learn how the input features affect the output irrespective of the effect of the translation of the feature in the domain. Batch normalization[53] could be introduced to optimize the activation function response so as to improve the learning convergence. Other layers that were introduced in the CNN architecture are the pooling layers which help decrease the number of trainable parameters reducing the dimensionality of their input.[54] It is also possible to implement regularization techniques such as dropout in order to improve the generalization capability of the network.[55]

As in many machine learning algorithms, even in neural networks modeling it is necessary to choose a set of optimal hyperparameters for the learning algorithm. For neural networks, beside the choice of hyperparameters for training, it is necessary to chose the optimal architecture, i.e., number of layers and number of neurons per layer. This investigation is computationally expensive but it is fundamental for the creation of an optimal model.

COMPUTATIONAL DETAILS

In this work, data-driven models are trained on a dataset of CFD simulations. The workflow can be summarized as follows:

- Setup of the CFD simulation: Statistically meaningful porous media geometries are created *in silico*, after which appropriate solvers are chosen for the modeling of the physical systems; then a meshing strategy is adopted so as to provide grid independent results.
- Creation of the dataset: The range of variations of the data-driven models inputs are established, and a CFD simulation of each data point, i.e., sample, is solved.
- Training of the neural networks: The type of neural network and its architecture is chosen; then, the hyperparameter tuning is performed in order to select the optimal parameters.

In this section, the computational details of each step of the workflow are detailed.

CFD Model Setup. The porous media geometries for the CFD simulations (Figure 2) were created *in silico* by means of the open-source toolbox Yade which implements a discrete element method (DEM). This tool was selected for the purpose of obtaining periodic packings of spheres with a low computational cost; in fact, the creation of a representation of hundreds of spheres takes few minutes. Since a dataset of hundreds of CFD simulations is required for the training of the neural network model, the fast creation of geometries is an advantage to reduce computational costs. The DEM model integrates the motion equations for each particle of the packing, so the position of a single particle is updated at each time step using the acceleration (a) computed from the total force acting on the particle (F) and its mass (m):

$$ma = \sum_i F_i \tag{8}$$

The forces taken into account are the interparticle normal and shear forces. A periodic boundary condition is set up for the sake of studying the behavior of a material in bulk, and as such avoiding the wall effects on the packing. A complete explanation of the computational code details is provided in the official documentation.[56]

The smallest representative elementary volume (REV) was identified in order to fix the minimum number of grains necessary for a macroscopic characterization independent from the size of the porous medium. Since the grain size distribution of the geometries follows a Gaussian distribution, the REV study was performed considering the highest standard deviation. The representative size was selected on the basis of the porosity, and it is equal to 250 grains. The REV study is detailed in the Supporting Information.

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

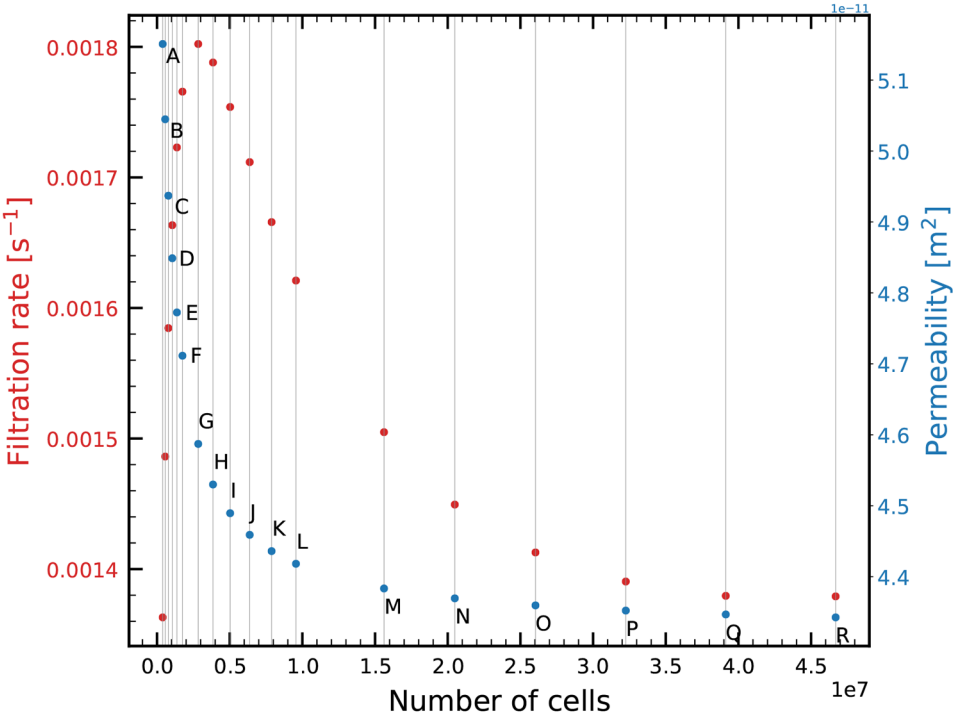


Figure 3. Grid independence study: effect of the number of cells on the permeability and on the filtration rate; refer to Table 1 for the details on the meshing strategies.

The geometries were used for the CFD simulations that were solved by using the open-source toolbox OpenFOAM v7. We refer in this paragraph to the solvers and tools of this software. The system is considered isothermal at 298 K, and the fluid phase has the properties of pure water. Therefore, it is a Newtonian fluid with density equal to 997 kg m^{-3} and kinematic viscosity equal to 0.89×10^{-6} m^2 s^{-1}.

The fluid flows in the porous media in laminar conditions. Thus, it is not necessary to model the turbulence, and the effect of gravity is not considered. The solver simpleFoam is employed for the resolution of eqs 1 and 2. The boundary conditions applied for the numerical solution of these equations are a pressure drop between the inlet and outlet boundaries, a no-slip condition on the grains surface, and symmetry conditions on the other faces of the simulation box.

The transport of the colloid in the fluid is modeled as a scalar transport of the normalized concentration C, due to the hypothesis mentioned in the previous paragraph. The scalarTransportFoam solver is employed for the resolution of eq 3 under steady-state conditions. At the inlet boundary, a normalized concentration equal to one is imposed; instead, at the grains surface, a null concentration is set. The null normalized concentration can be physically interpreted as a clean bed filtration[19] or as an instantaneous reaction on the grains surface.

The computational grid, i.e., the mesh, was constructed by means of the OpenFOAM tools blockMesh and snappyHexMesh, and a grid independence study is performed. The operating conditions of the simulation for the grid independence study were selected so that the highest Péclet number is reached; in fact, all the CFD simulations that were solved for the creation of the dataset have a Péclet number lower than 1500. Figure 3 reports the effect of the number of cells on the permeability (on the right axis) and the filtration rate (on the left axis). In Table 1, the meshing strategy and the errors are reported; the errors are calculated as the relative error between the current strategy and the last strategy at which correspond the highest number of cells, so the most accurate result. The meshing strategy adopted for all the CFD simulations solved in this work is the N strategy (Figure 4), which represents an acceptable trade-off between the accuracy and the computational cost of the simulations since the

Table 1. Grid Independence Study: Relative Error between Each Strategy (A−Q) and the R Strategy on the Permeability and the Filtration Rate[a]

meshing strategy	CPD	RL	computational time [h]	error on permeability [%]	error on filtration rate [%]
A	16	0	0.2	18.61	1.17
B	18	0	0.3	16.17	7.77
C	20	0	0.5	13.69	14.90
D	22	0	0.6	11.66	20.61
E	24	0	0.8	9.90	24.94
F	26	0	1.1	8.49	28.03
G	16	1	1.7	5.63	30.68
H	18	1	2.7	4.31	29.65
I	20	1	3.5	3.38	27.18
J	22	1	4.5	2.68	24.12
K	24	1	6.0	2.15	20.79
L	26	1	6.8	1.74	17.54
M	16	2	12.8	0.94	9.12
N	18	2	19.0	0.62	5.10
O	20	2	21.0	0.39	2.44
P	22	2	28.5	0.22	0.83
Q	24	2	40.5	0.09	0.03
R	26	2	38.6	0.00	0.00

[a]CPD: number of cells per mean diameter. RL: Refinement level in snappyHexMesh.

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

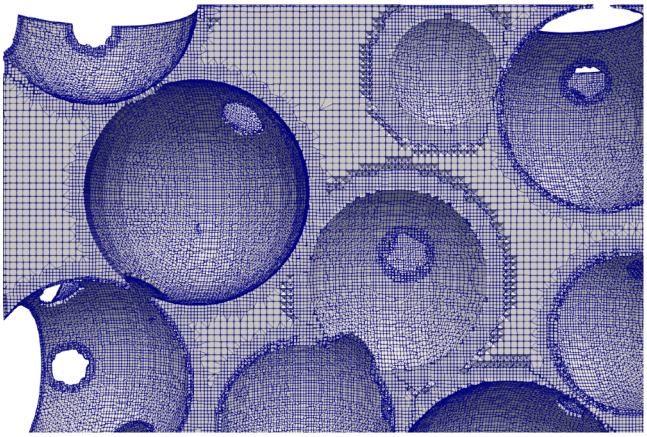


Figure 4. Zoom on the computational grid of a sample representation; the meshing strategy is N of Table 1.

relative error on the permeability is 0.62%, the error on the filtration rate is 5.1% and the CFD simulation requires 19 h on a single core to be solved.

Creation of the Dataset. After the setup of the CFD model, 280 simulations were solved for the creation of the dataset employed for the training of the neural networks. The samples in the dataset differ for some operating conditions and for some geometrical parameters: the inlet pressure of the fluid in the porous media, the colloid dimension (thus the colloid diffusion coefficient), the mean diameter, and the standard deviation of the grains size distribution. The variation of the features ranges as detailed in Table 2.

Table 2. Range of Variation of the Features in the Dataset[a]

feature	range of variation
mean diameter [μm]	100−200
standard deviation	0.0−0.3
inlet pressure [Pa]	8.2×10^{-5}−9.74×10^{-4}
colloid diameter [nm]	30−100

[a]The mean diameter and the normalized standard deviation, i.e., standard deviation normalized by mean diameter, of the particle size distribution are input features for the creation of the geometries. The inlet pressure and the colloid diameter are input features for the CFD simulations.

The CFD simulations were performed in single core on an HPC cluster equipped with 29 nodes with CPU 2x Intel Xeon E5−2680 v3 2.50 GHz 12 cores RAM of 384 GB. The computational time for each simulation is about 19 h (Table 1).

Neural Networks Setup. The dataset is employed for the training of neural networks models. In this work, FCNN and CNN are trained and tested.

Two FCNN were trained, one for the prediction of the permeability and one for the prediction of the filtration rate. The permeability is predicted starting from the geometrical parameters of the porous media: porosity, mean diameter, and standard deviation of the grains diameter distribution. The filtration rate is predicted starting from the same geometrical parameters together with the operating conditions of the filtration simulation: the inlet pressure of the fluid and the diffusion coefficient of the colloid. Different architectures and learning rates were investigated in order to determine an optimal neural network model, in the following sections the outcomes are presented and discussed.

Even for the CNN models, two networks had to be trained. The one for the prediction of the permeability receives exclusively as input the entire geometry of the porous media. The geometry was analytically defined by means of Yade as a list of centers and radii, for the purpose of providing the CNN with a compatible input a voxelization representation is created. Starting from a STL file, the Python's library Trimesh[57] was used for the voxelization of the geometry. The resulting array contains 0 in the solid phase and 1 in the fluid phase, afterward the Euclidean distance transform was applied to the resulting array so as to provide a more descriptive input to the CNN,[42] resulting in every voxel, i.e., component, in the solid phase being marked with 0, and every voxel in the fluid phase with the distance from the closest solid voxel. The CNN for the prediction of the filtration rate takes as input the same geometry together with the inlet pressure of the fluid and the diffusion coefficient, which are concatenated to the flattened output of the CNN portion. The architecture of the CNN is summarized in Figure 5. The hyperparameter tuning in this case was done on the choice of the learning rate, on the scaling of the input data and on the using of the batch normalization layers, in the following section these results are discussed. The effect of the dataset size on the accuracy of the predictions is reported in the Supporting Information.

The numerical inputs and outputs for both types of neural networks were scaled between −1 and 1 in order to optimize the activation function effectiveness. The optimization algorithm that was used for the training in this work is Adam[58] which is considered the best overall choice among the gradient descent optimization algorithms.[59] The loss function minimized by the algorithm is the mean squared error, eq 9, which is defined as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{9}$$

where n is the number of samples used during training, $\hat{y}_i$ is the true output (namely, the CFD simulation results, representing the ground truth in our model), and y_i is the output of the neural network prediction. Given the entire dataset, 70% of it was used as training set, and the remaining part as validation and test set, respectively, 20 and 10%.

The training of the neural networks was performed on graphics processing units nVidia Tesla V100 SXM2, the Python libraries Keras and Tensorflow were used for the setup, the training and the testing. The computational time for the training of the FCNN is about 30 min, instead the training of the CNN takes around 15 h using the entire dataset of 280 samples.

RESULTS AND DISCUSSION

CFD Results. In Figure 6, two contour plots of the velocity and the normalized concentration for a sample simulation are reported. The pressure drop through porous media can be related to the Reynolds number through the Ergun equation:

$$\frac{\Delta P \rho}{G_0^2}\frac{d_g}{L}\frac{\varepsilon^3}{(1-\varepsilon)} = 150\frac{1-\varepsilon}{(d_g G_0)/\mu} + 1.75 \tag{10}$$

where G_0 is the mass flux, d_g is the surface averaged mean diameter, ε is the porosity, and L is the porous medium length in the flux direction. The dimensionless numbers Re^* and ΔP^* can be introduced so that the Ergun equation results in

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

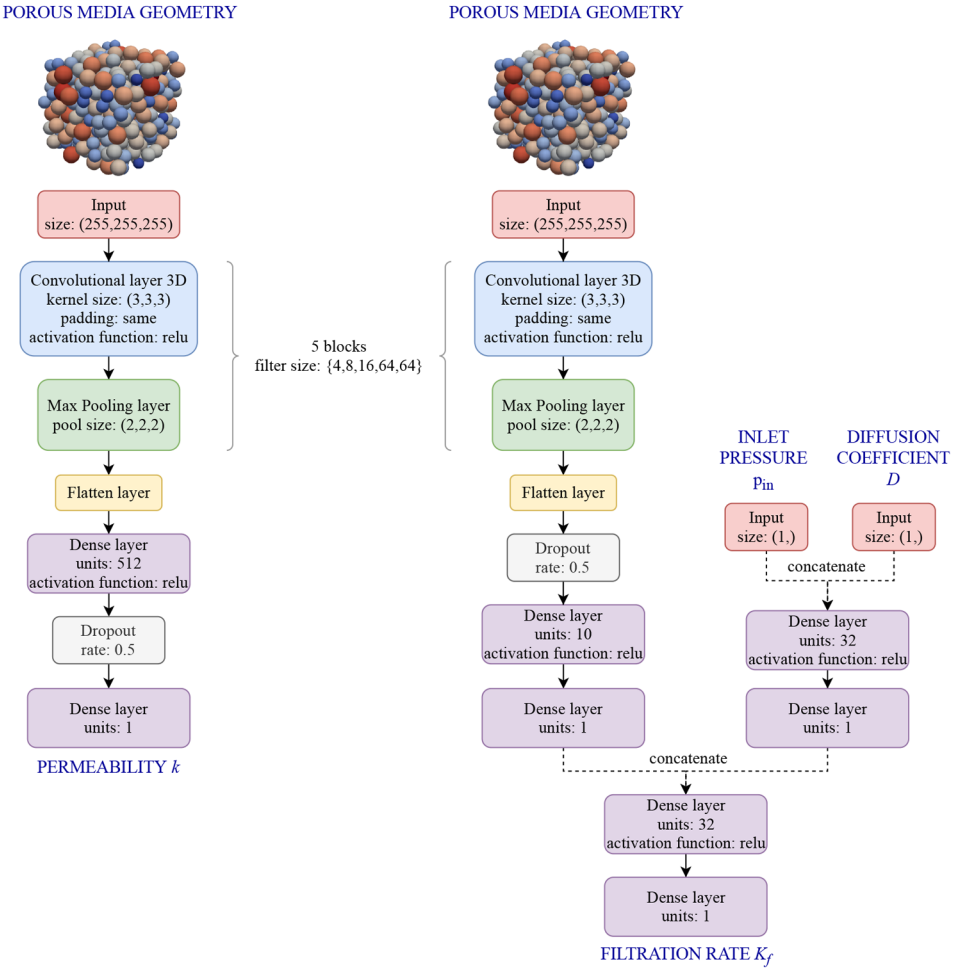


Figure 5. Summary of the convolutional neural networks architectures. The CNN for the prediction of the permeability receives as input the porous media geometry (in this image the input is represented as the packing of spheres for the sake of clarity of presentation), whereas the actual input to the CNN is the Euclidean distance field from the closest solid. The CNN for the prediction of the filtration rate receives both the geometry, the inlet pressure and the diffusion coefficient. The parameters of each layer are reported in the blocks; all unspecified parameters are set as the Keras defaults.

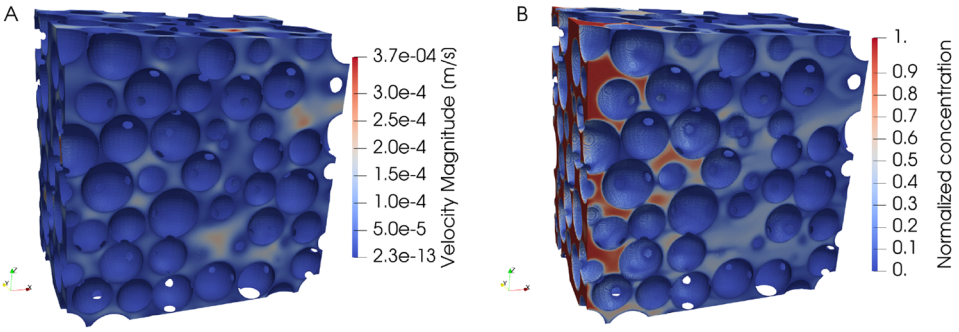


Figure 6. Contour plot of the velocity (A) and of the normalized concentration (B) for a sample of the dataset.

$$\Delta P^* = \frac{150}{Re^*} + 1.75 \tag{11}$$

For the creation of the dataset, 280 simulations were performed; in Figure 7, each point represents a CFD simulation, instead the straight line represents the Ergun equation. Notably, there is a good agreement between the microscale CFD results and the analytical correlation.

Concerning the transport simulations results, the upscaled filtration rate K_f (eq 6) can be related to the Péclet number via an advective Damköhler number (Figure 8). Those are defined as follows:

$$Da = \frac{K_f L}{q} \tag{12}$$

$$Pe = \frac{q d_g}{\mathcal{D}} \tag{13}$$

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

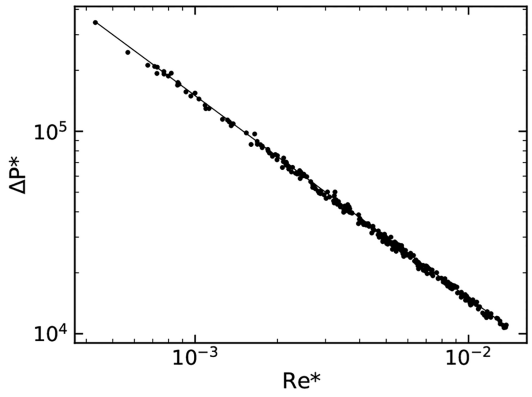


Figure 7. Comparison between the CFD results (points) and the Ergun equation (line, eq 10).

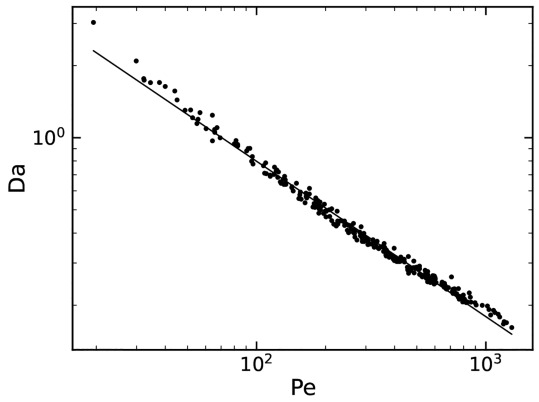


Figure 8. Comparison of the CFD results (points) and the Damköhler–Péclet correlation (line, eq 14).

The filtration rate is usually interpreted via a power-law relationship with the Péclet number,[6,60,61] resulting in the following exponential relationship:[49]

$$Da = APe^{B} \tag{14}$$

where A and B are the coefficients that best fit the results, in our case $A = 1.20$ and $B = -0.65$. The fitting was performed by linear regression on all the data points of the dataset.

It has to be noted that while in this specific case it was possible to find a constitutive equation relating the Péclet (i.e., operating conditions) and Damköhler (i.e., filtration efficiency) numbers with a clear power-law functional form, it has been shown that for arbitrary and realistic geometries this is not the case, as it can be seen in Boccardo et al.,[19] resulting in the need of a model able to interpret these features. This will be explored in the following section.

This effect of complex geometries on fluid flow-reaction structures presents itself more clearly in the diffusion-limited low-Péclet-number range in Figure 8 where a correct estimation of dispersive fluxes is complicated by the complex random geometry by the arising of terms depending on the correlation between solute concentration and flow velocity spatial fluctuations.[17] The reader is referred to the literature for more details.

Neural Networks. In this section, the results of the neural networks modeling are presented. For the case of the FCNN, different kinds of architectures and different learning rates were explored in order to tune the model. In Table 3 the results are summarized. Four different learning rates, in a range commonly used, are proposed: 10^{-3}, 10^{-4}, 10^{-5}, and 10^{-6}; four different architectures were tested with an increasing number of hidden layers, 1−4 hidden layers. The number of components in the square brackets of Table 3 is the number of hidden layers, and the value of each component is the number of neurons per layer. The different learning rates and architectures do not strongly affect the results. The best set of results for the prediction of the permeability can be found with a learning rate equal to 10^{-3} and the architecture type [128, 64, 32]; these parameters result in an average error of 2.46% and a maximum error of 7.62%. The best set of results for the prediction of the filtration rate is a learning rate equal to 10^{-4} and architecture type [128, 64, 32]; these parameters result in an average error of 2.20% and a maximum error of 7.66%. The parity diagrams for these cases are displayed in Figure 9.

The CNN architectures were set up as described in the previous section. In order to find the optimal architecture and training strategy, a close number of solutions between the many available were explored, in particular the learning rate, as for the FCNN, the scaling of the input data, and the batch normalization layers presence. For the permeability neural network the learning rates tested were 10^{-4}, 10^{-5}, and 10^{-6}, while for the filtration rate neural network they were 10^{-3}, 10^{-4}, and 10^{-5}. The scaling proposed in this work is a min−max strategy (min equal to 0 and max equal to the maximum Euclidean distance in the dataset). In Table 4 the results are summarized. Notably, the

Table 3. Hyperparameter Tuning of the FCNN[a]

	permeability			
	10^{-3}	10^{-4}	10^{-5}	10^{-6}
[32]	2.45% (8.18%)	2.55% (8.03%)	2.52% (8.52%)	2.51% (9.0%)
[64, 32]	2.49% (8.31%)	2.46% (8.53%)	2.51% (8.88%)	2.41% (9.09%)
[128, 64, 32]	2.46% (7.62%)	2.46% (8.49%)	2.42% (8.30%)	2.43% (8.67%)
[256, 128, 64, 32]	2.47% (8.44%)	2.49% (8.73%)	2.51% (8.04%)	2.49% (8.51%)
	filtration rate			
	10^{-3}	10^{-4}	10^{-5}	10^{-6}
[32]	2.23% (10.57%)	2.27% (9.34%)	2.26% (8.24%)	2.81% (9.21%)
[64, 32]	2.05% (10.56%)	2.21% (9.74%)	2.54% (8.11%)	2.67% (11.74%)
[128, 64, 32]	2.27% (11.7%)	2.20% (7.66%)	2.41% (9.64%)	2.69% (12.0%)
[256, 128, 64, 32]	2.17% (10.12%)	2.34% (9.13%)	2.28% (9.09%)	2.44% (8.99%)

[a]The average error and the maximum error on the test set (in parentheses) are reported for the different architectures and learning rates. Underlined entries are the chosen best architecture/hyperparameter coupling.

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

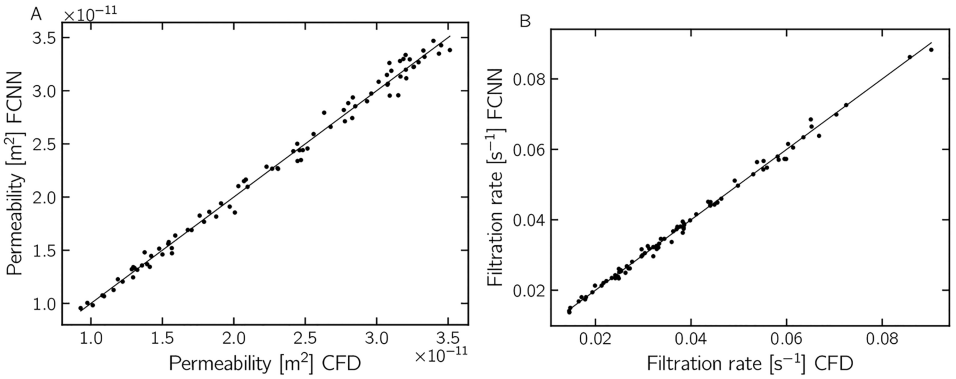


Figure 9. Parity diagrams for the predictions on the test set of the permeability (A) and the filtration rate (B) by the use of FCNN.

Table 4. Hyperparameter Tuning of the CNN[a]

permeability				
normalization	scaling	10^{-4}	10^{-5}	10^{-6}
yes	*no*	5.57% (22.50%)	15.00% (48.40%)	27.90% (95.07%)
no	*yes*	3.14% (15.93%)	2.33% (12.24%)	2.39% (12.24%)
yes	*yes*	7.27% (21.57%)	17.44% (55.34%)	24.65% (58.34%)
no	*no*	39.74% (100.44%)	3.03% (13.66%)	2.52% (12.53%)
filtration rate				
normalization	scaling	10^{-3}	10^{-4}	10^{-5}
yes	*no*	5.90% (15.58%)	6.62% (19.00%)	8.55% (27.90%)
no	*yes*	5.02% (14.99%)	4.90% (12.75%)	6.44% (23.33%)
yes	*yes*	6.68% (24.91%)	7.85% (29.09%)	8.69% (32.49%)
no	*no*	25.01% (65.00%)	25.83% (66.67%)	6.40% (18.20%)

[a]The average error and the maximum error on the test set (in brackets) are reported for the different learning rates, for the presence of batch normalization layers and for the scaling of the input data. Underlined, the chosen best architecture/hyperparameter coupling.

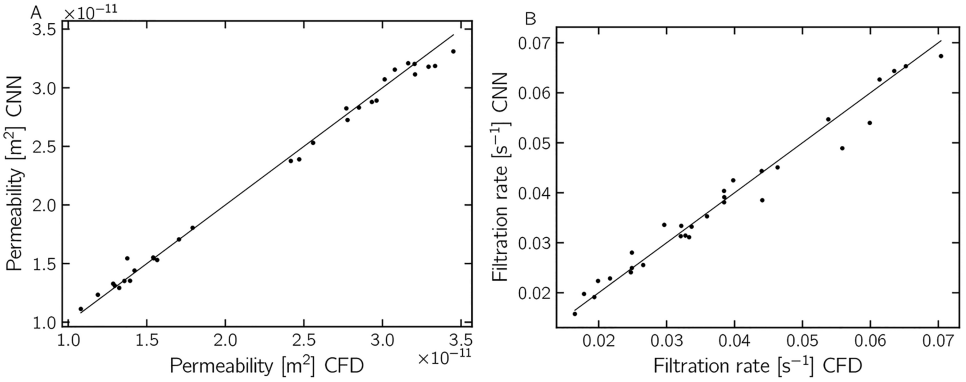


Figure 10. Parity diagrams for the predictions on the test set of the permeability (A) and the filtration rate (B) by the use of CNN.

presence of the normalization layers has a powerful effect on the results. For both the prediction of the permeability and the filtration rate, the best architecture is found if the batch normalization layers are absent and if the scaling is applied. The best learning rate is equal to 10^{-5} for the prediction of the permeability, which leads to an average error of 2.33% and a maximum error of 12.24%. The best learning rate is equal to 10^{-4} for the prediction of the filtration rate, leading to an average error of 4.90% and a maximum error of 12.75%. The parity diagrams for these cases are displayed in Figure 10.

The results in Tables 3 and 4 show that both the FCNN and the CNN accurately predict the permeability for the case study approached in this work. This means that the input features fed to the FCNN precisely describe the geometry, and the convolutional layers of the CNN grasp the most effective features autonomously from the image fed to the network. FCNN better predicts the filtration rate compared to the CNN, even though the latter still has good accuracy in absolute terms. The difference in the prediction of the filtration rate can be addressed to the architecture of the CNN; in fact, this particular architecture may need further hyperparameter tuning. However, from these results it is possible to deduce that the present methodology is promising for other applications and porous media geometries. In particular, the use of CNN will be more and more useful for geometries whose featurization is more complex.

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

CONCLUSIONS

Prior work has documented the efficacy of training neural networks for the prediction of the permeability in various types of porous media systems.[37−41] However, the majority of these works has focused on models that deal with geometrical descriptors. Since the applications of porous media systems in chemical engineering are related to catalytic chemical reactions and filtration, we developed a methodology for the prediction of the filtration rate in porous media. Both fully connected and convolutional neural networks were tested in our workflow.[46] The main difference between these two approaches is the input to the model: In the first case, hand-selected features are the input to the FCNN, in the second case, the entire image is fed to the networks, which are able to autonomously detect the features in the image for the prediction of its target output.

While the two models exhibit similar (and satisfactory) accuracy, it is notable that a comparable number of CFD simulations between the two cases were sufficient to reach such accuracy, notwithstanding the clearly higher complexity of the convolutional model with respect to the classical fully connected network. Even more so, in light of the much higher flexibility of use of the model based on CNN, which (at the cost of higher computational cost for its training) will prove to be much more useful by means of opening to this kind of analysis even systems featuring complicated geometries, as are frequently found in chemical engineering both in purification/filtration apparatuses and in packed bed catalytic reactors. Beside its usefulness in dealing with complicated *structures*, these convolutional models have already proven in this work to be able to treat complicated *phenomena*, by providing a way to robustly construct accurate models even in cases for which it may not be immediate (or feasible) to choose integral input features with which to build a predictive analytical model, a case in point being the problem of colloidal filtration as explored in this work. The long-term application of this methodology will be the use of neural networks for the prediction of other microscale properties that lack analytical models that correlate them to microscale properties.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.iecr.1c04760.

Representative volume study and the effect of the dataset size on the accuracy of the predictions (PDF)

AUTHOR INFORMATION

Corresponding Author

Gianluca Boccardo − *DISAT - Dipartimento Scienza Applicata e Tecnologia, Politecnico di Torino, 10129 Torino, Italy;* orcid.org/0000-0003-1264-8237; Email: gianluca.boccardo@polito.it

Authors

Agnese Marcato − *DISAT - Dipartimento Scienza Applicata e Tecnologia, Politecnico di Torino, 10129 Torino, Italy*

Daniele Marchisio − *DISAT - Dipartimento Scienza Applicata e Tecnologia, Politecnico di Torino, 10129 Torino, Italy*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.iecr.1c04760

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors gratefully acknowledge Lorenzo Vasquez Giuliano for his support in the creation of the geometries. Computational resources were provided by hpc@polito, which is a project of Academic Computing within the Department of Control and Computer Engineering at the Politecnico di Torino (http://www.hpc.polito.it) We acknowledge the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support.

REFERENCES

(1) Molnar, I. L.; Pensini, E.; Asad, M. A.; Mitchell, C. A.; Nitsche, L. C.; Pyrak-Nolte, L. J.; Miño, G. L.; Krol, M. M. Colloid transport in porous media: a review of classical mechanisms and emerging topics. *Transport Porous Med.* **2019**, *130*, 129−156.

(2) Boccardo, G.; Augier, F.; Haroun, Y.; Ferré, D.; Marchisio, D. L. Validation of a novel open-source work-flow for the simulation of packed-bed reactors. *Chem. Eng. J.* **2015**, *279*, 809−820.

(3) Pangarkar, K.; Schildhauer, T. J.; van Ommen, J. R.; Nijenhuis, J.; Kapteijn, F.; Moulijn, J. A. Structured packings for multiphase catalytic reactors. *Ind. Eng. Chem. Res.* **2008**, *47*, 3720−3751.

(4) Iliev, O.; Kirsch, R.; Lakdawala, Z.; Rief, S.; Steiner, K. Modeling and simulation of filtration processes. In *Currents in Industrial Mathematics*; Neunzert, H., Prätzel-Wolters, D., Eds.; Springer: Berlin, 2015; pp 163−228.

(5) Bensaid, S.; Marchisio, D.; Fino, D.; Saracco, G.; Specchia, V. Modelling of diesel particulate filtration in wall-flow traps. *Chem. Eng. J.* **2009**, *154*, 211−218.

(6) Yao, K.-M.; Habibian, M. T.; O'Melia, C. R. Water and waste water filtration. Concepts and applications. *Environ. Sci. Technol.* **1971**, *5*, 1105−1112.

(7) Crevacore, E.; Boccardo, G.; Marchisio, D.; Sethi, R. Microscale colloidal transport simulations for groundwater remediation. *Chem. Engineer. Trans.* **2016**, *47*, 271−276.

(8) Farajzadeh, R.; Andrianov, A.; Krastev, R.; Hirasaki, G.; Rossen, W. R. Foam−oil interaction in porous media: Implications for foam assisted enhanced oil recovery. *Adv. Colloid Interfac.* **2012**, *183*, 1−13.

(9) Gabrielli, P.; Gazzani, M.; Mazzotti, M. The role of carbon capture and utilization, carbon capture and storage, and biomass to enable a net-zero-CO2 emissions chemical industry. *Ind. Eng. Chem. Res.* **2020**, *59*, 7033−7045.

(10) Baciocchi, R.; Storti, G.; Mazzotti, M. Process design and energy requirements for the capture of carbon dioxide from air. *Chem. Eng. Process.* **2006**, *45*, 1047−1058.

(11) Franco, A. A. Multiscale modelling and numerical simulation of rechargeable lithium ion batteries: concepts, methods and challenges. *Rsc Adv.* **2013**, *3*, 13027−13058.

(12) Fang, R.; Schmidt, C. P.; Wall, W. A. A coupled finite element approach to spatially resolved lithium plating and stripping in three-dimensional anode microstructures of lithium-ion cells. 2021. Preprint submitted to *J. Comput. Phys.* https://mediatum.ub.tum.de/1616408 (accessed 2021-12-30).

(13) Cunha, R. P.; Lombardo, T.; Primo, E. N.; Franco, A. A. Artificial intelligence investigation of NMC cathode manufacturing parameters interdependencies. *Batter. Supercaps* **2020**, *3*, 60−67.

(14) Icardi, M.; Boccardo, G.; Marchisio, D. L.; Tosco, T.; Sethi, R. Pore-scale simulation of fluid flow and solute dispersion in three-dimensional porous media. *Phys. Rev. E* **2014**, *90*, 013032.

(15) Boccardo, G.; Crevacore, E.; Passalacqua, A.; Icardi, M. Computational analysis of transport in three-dimensional heterogeneous materials. *Comput. Vis. Sci.* **2020**, *23*, 4.

(16) Pettersson, K.; Maggiolo, D.; Sasic, S.; Johansson, P.; Sasic-Kalagasidis, A. On the impact of porous media microstructure on

rainfall infiltration of thin homogeneous green roof growth substrates. *J. Hydrol.* **2020**, *582*, 124286.
(17) Maggiolo, D.; Picano, F.; Zanini, F.; Carmignato, S.; Guarnieri, M.; Sasic, S.; Ström, H. Solute transport and reaction in porous electrodes at high Schmidt numbers. *J. Fluid Mech.* **2020**, *896*, A13.
(18) Municchi, F.; Di Pasquale, N.; Dentz, M.; Icardi, M. Heterogeneous Multi-Rate mass transfer models in OPENFOAM. *Comput. Phys. Commun.* **2021**, *261*, 107763.
(19) Boccardo, G.; Marchisio, D. L.; Sethi, R. Microscale simulation of particle deposition in porous media. *J. Colloid Interface Sci.* **2014**, *417*, 227−237.
(20) Dixon, A. G.; Nijemeisland, M. CFD as a design tool for fixed-bed reactors. *Ind. Eng. Chem. Res.* **2001**, *40*, 5246−5254.
(21) Boccardo, G.; Sethi, R.; Marchisio, D. L. Fine and ultrafine particle deposition in packed-bed catalytic reactors. *Chem. Eng. Sci.* **2019**, *198*, 290−304.
(22) Fan, C.; Jin, H.; Chen, Y.; Ge, Z.; Zhao, Q. A numerical study on gasification of a single-pore char particle in supercritical water. *J. Therm. Anal. Calorim.* **2020**, *141*, 1591.
(23) Goldin, G. M.; Colclasure, A. M.; Wiedemann, A. H.; Kee, R. J. Three-dimensional particle-resolved models of Li-ion batteries to assist the evaluation of empirical parameters in one-dimensional models. *Electrochimi. Acta* **2012**, *64*, 118−129.
(24) Venkatasubramanian, V. The promise of artificial intelligence in chemical engineering: Is it here, finally. *AIChE J.* **2019**, *65*, 466−478.
(25) Noé, F.; De Fabritiis, G.; Clementi, C. Machine learning for protein folding and dynamics. *Curr. Opin. Struc. Biol.* **2020**, *60*, 77−84.
(26) Reichstein, M.; Camps-Valls, G.; Stevens, B.; Jung, M.; Denzler, J.; Carvalhais, N.; et al. Deep learning and process understanding for data-driven Earth system science. *Nature* **2019**, *566*, 195−204.
(27) Di Pasquale, N.; Davie, S. J.; Popelier, P. L. Optimization algorithms in optimal predictions of atomistic properties by kriging. *J. Chem. Theory Comput.* **2016**, *12*, 1499−1513.
(28) Davie, S. J.; Di Pasquale, N.; Popelier, P. L. Incorporation of local structure into kriging models for the prediction of atomistic properties in the water decamer. *J. Comput. Chem.* **2016**, *37*, 2409−2422.
(29) Di Pasquale, N.; Elliott, J. D.; Hadjidoukas, P.; Carbone, P. Dynamically Polarizable Force Fields for Surface Simulations via Multi-output Classification Neural Networks. *J. Chem. Theory Comput.* **2021**, *17*, 4477−4485.
(30) Oviedo, F.; Ren, Z.; Sun, S.; Settens, C.; Liu, Z.; Hartono, N. T. P.; Ramasamy, S.; DeCost, B. L.; Tian, S. I.; Romano, G.; et al. Fast and interpretable classification of small X-ray diffraction datasets using data augmentation and deep neural networks. *npj Comput. Mater.* **2019**, *5*, 60.
(31) Smith, J. D.; Neto, A. A.; Cremaschi, S.; Crunkleton, D. W. CFD-based optimization of a flooded bed algae bioreactor. *Ind. Eng. Chem. Res.* **2013**, *52*, 7181−7188.
(32) Sun, X.; Kim, S.; Yang, S. D.; Kim, H. S.; Yoon, J. Y. Multi-objective optimization of a Stairmand cyclone separator using response surface methodology and computational fluid dynamics. *Powder Technol.* **2017**, *320*, 51−65.
(33) Saeedan, M.; Solaimany Nazar, A. R.; Abbasi, Y.; Karimi, R. CFD Investigation and neutral network modeling of heat transfer and pressure drop of nanofluids in double pipe helically baffled heat exchanger with a 3-D fined tube. *Appl. Therm. Eng.* **2016**, *100*, 721−729.
(34) Babanezhad, M.; Pishnamazi, M.; Marjani, A.; Shirazian, S. Bubbly flow prediction with randomized neural cells artificial learning and fuzzy systems based on k−ε turbulence and Eulerian model data set. *Sci. Rep.* **2020**, *10*, 13837.
(35) Babanezhad, M.; Rezakazemi, M.; Hajilary, N.; Shirazian, S. Liquid-phase chemical reactors: Development of 3D hybrid model based on CFD-adaptive network-based fuzzy inference system. *Can. J. Chem. Eng.* **2019**, *97*, 1676−1684.
(36) Partopour, B.; Dixon, A. G. 110th anniversary: commentary: CFD as a modeling tool for fixed bed reactors. *Ind. Eng. Chem. Res.* **2019**, *58*, 5733−5736.
(37) Rabbani, A.; Babaei, M. Hybrid pore-network and lattice-Boltzmann permeability modelling accelerated by machine learning. *Adv. Water Resour.* **2019**, *126*, 116−128.
(38) Liu, S.; Zolfaghari, A.; Sattarin, S.; Dahaghi, A. K.; Negahban, S. Application of neural networks in multiphase flow through porous media: Predicting capillary pressure and relative permeability curves. *J. Petrol. Sci. Eng.* **2019**, *180*, 445−455.
(39) Alqahtani, N.; Alzubaidi, F.; Armstrong, R. T.; Swietojanski, P.; Mostaghimi, P. Machine learning for predicting properties of porous media from 2d X-ray images. *J. Petrol. Sci. Eng.* **2020**, *184*, 106514.
(40) Wu, H.; Fang, W.-Z.; Kang, Q.; Tao, W.-Q.; Qiao, R. Predicting effective diffusivity of porous media from images by deep learning. *Sci. Rep.* **2019**, *9*, 20387.
(41) Wu, J.; Yin, X.; Xiao, H. Seeing permeability from images: fast prediction with convolutional neural networks. *Sci. Bull.* **2018**, *63*, 1215−1222.
(42) Santos, J. E.; Xu, D.; Jo, H.; Landry, C. J.; Prodanović, M.; Pyrcz, M. J. PoreFlow-Net: A 3D convolutional neural network to predict fluid flow through porous media. *Adv. Water Resour.* **2020**, *138*, 103539.
(43) Santos, J. E.; Yin, Y.; Jo, H.; Pan, W.; Kang, Q.; Viswanathan, H. S.; Prodanović, M.; Pyrcz, M. J.; Lubbers, N. Computationally Efficient Multiscale Neural Networks Applied to Fluid Flow in Complex 3D Porous Media. *Transp. Porous Media* **2021**, *140*, 241.
(44) Hennigh, O. Lat-net: compressing lattice Boltzmann flow simulations using deep neural networks. *arXiv (Machine Learning)*, May 25, 2017. 1705.09036, ver. 1. https://arxiv.org/pdf/1705.09036 (accessed Dec. 30, 2021).
(45) Wang, Y. D.; Chung, T.; Armstrong, R. T.; Mostaghimi, P. ML-LBM: predicting and accelerating steady state flow simulation in porous media with convolutional neural networks. *Transp. Porous Media* **2021**, *138*, 49−75.
(46) Marcato, A.; Boccardo, G.; Marchisio, D. A computational workflow to study particle transport and filtration in porous media: Coupling CFD and deep learning. *Chem. Eng. J.* **2021**, *417*, 128936.
(47) Marcato, A.; Boccardo, G.; Marchisio, D. L. *A Computational Workflow to Study Particle Transport in Porous Media: Coupling CFD and Deep Learning*; Elsevier: Amsterdam, 2020; Computer Aided Chemical Engineering Vol. *48*; pp 1759−1764.
(48) Andersson, B.; Andersson, R.; Håkansson, L.; Mortensen, M.; Sudiyo, R.; Van Wachem, B. *Computational Fluid Dynamics for Engineers*; Cambridge University Press, 2011.
(49) Boccardo, G.; Crevacore, E.; Sethi, R.; Icardi, M. A robust upscaling of the effective particle deposition rate in porous media. *J. Contam. Hydrol.* **2018**, *212*, 3−13.
(50) Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press, 2016; http://www.deeplearningbook.org (accessed 2021/12/30).
(51) Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Unc. Fuzz. Knowl. Based Syst.* **1998**, *6*, 107−116.
(52) LeCun, Y. Generalization and network design strategies. In *Connectionsism In Perspective*; Elsevier, 1989; Vol. *19*, pp 143−155.
(53) Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proc. Int. Conf. Mach. Learn.* **2015**, *37*, 448−456.
(54) Boureau, Y.-L.; Ponce, J.; LeCun, Y. A theoretical analysis of feature pooling in visual recognition. *Proc. Int. Conf. Mach. Learn.* **2010**, 111−118.
(55) Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929−1958.
(56) Smilauer, V. *Yade Documentation*, 3rd ed.; The Yade Project, 2021. http://yade-dem.org/doc/ (accessed 2021/12/30).
(57) Dawson-Haggerty, M. *trimesh* version 3.2.0. https://trimsh.org/ (accessed 2021/12/30).
(58) Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv (Machine Learning)*, Dec. 22, 2014, 1412.6980, ver. 1. https://arxiv.org/pdf/1412.6980 (accessed Dec. 30, 2021).

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX

(59) Ruder, S. An overview of gradient descent optimization algorithms. *arXiv (Machine Learning)*, Sept. 15, 2016, 101609.04747, ver. 1. https://arxiv.org/pdf/1609.04747 (accessed Dec. 30, 2021).
(60) Tufenkji, N.; Elimelech, M. Correlation equation for predicting single-collector efficiency in physicochemical filtration in saturated porous media. *Environ. Sci. Technol.* **2004**, *38*, 529−536.
(61) Levich, V. G. *Physicochemical Hydrodynamics*; Prentice-Hall Inc.: Hoboken, NJ, 1962.

https://doi.org/10.1021/acs.iecr.1c04760
Ind. Eng. Chem. Res. XXXX, XXX, XXX−XXX