Chemical Engineering Journal 504 (2025) 158577

Contents lists available at ScienceDirect

Chemical Engineering Journal

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

Robust pore-resolved CFD through porous monoliths reconstructed by micro-computed tomography: From digitization to flow prediction

Olivier Guévremont [a], Lucka Barbeau [a], Vaiana Moreau [b], Federico Galli [c], Nick Virgilio [b], Bruno Blais [a,*]

[a] CHAOS laboratory, Department of Chemical Engineering, Polytechnique Montreal, PO Box 6079, Stn Centre-Ville, Montreal, H3C 3A7, Quebec, Canada
[b] CREPEC, Department of Chemical Engineering, Polytechnique Montreal, PO Box 6079, Stn Centre-Ville, Montreal, H3C 3A7, Quebec, Canada
[c] Department of Chemical and Biotechnological Engineering, Université de Sherbrooke, 2500 Bd de l'Université, Sherbrooke, J1K 2R1, Quebec, Canada

ARTICLE INFO

Keywords:
Pore-resolved CFD
Radial basis functions
Immersed boundary
Porous media
Finite element method
High-performance computing

ABSTRACT

Porous media are ubiquitous in energy storage and conversion, catalysis, biomechanics, hydrogeology, as well as many other fields. These materials possess high surface-to-volume ratios and their complex channels can restrict and guide the flow. However, optimizing design parameters for specific applications remains challenging due to the intricate structure of porous media. Pore-resolved CFD reveals the effects of their structure on flow characteristics, but is limited by the performance of mesh generation algorithms for such complex geometries. To alleviate this issue, we use a sharp immersed boundary method which enables usage of Cartesian, non-conformal grids, within a massively parallel finite element framework. This method preserves the order convergence of the scheme and allows for adaptive mesh refinement (AMR). We introduce a radial basis function-based representation of solids that allows to solve the flow through complex geometries with precision. We verify the method using the method of manufactured solutions. We validate it using pressure drop measurements through porous silicone monoliths digitized by X-ray computed microtomography, for pore Reynolds numbers up to 30. Simulations are conducted using grids of 200M cells distributed over 8k cores, which would require 16 times more cells without AMR. Results reveal that pore network structure is the principal factor describing pressure evolution and that preferential channels are dominant at this scale. In this work, we demonstrate a robust and efficient workflow for pore-resolved simulations of porous monoliths. This work bridges the gap between sub-millimetric flow and macroscopic properties, which will open the door to design and optimize processes through the usage of physics-based digital twins of complex porous media.

1. Flow in porous media based on microtomography

Porous media are attractive materials for a wide variety of application fields: heterogeneous catalysis [1,2], separation (chromatography) [2], biomedical engineering (interstitial flows, drug delivery) [3,4], filtration [5], and hydrogeology [6], to name a few. They usually exhibit high specific surface and channel networks that can form tortuous, merging and forking paths that limit the thickness of boundary layers, depending on properties such as porosity, pore size distributions, tortuosity and isotropy. Continuous flow micro-reactors, in which channels are sub-millimetric, take advantage of the low transport distances resulting from narrow channels, as well as a fine control of the process parameters [7]. Along with the commonly studied residence time distribution and pressure drop, accurately predicting the velocity profile in these sub-millimeter channels is essential for the design of these equipment. The mixing effects and boundary layers must be resolved in order to predict and control mixing, chemical reactions, and other transport phenomena. However, obtaining quality data regarding the velocity field is challenging. Experimental methods such as Magnetic Resonance Velocimetry (MRV) [8] or Particle Image Velocimetry (PIV) [9] are limited: MRV is limited by spatial resolution and PIV by the opacity of the medium.

Given these challenges, Computational Fluid Dynamics (CFD) has been used for decades to solve the pore-resolved flow in porous media [10,11] (PRCFD). At first, most of the work published on the subject considered the flow in idealized and periodic Representative Elementary Volumes (REV) to reduce computational costs [12], but this approach is limited to scale-independent domains, simple geometries and is complex to validate experimentally. Lately however, increases in hardware and software capabilities gave rise to more investigations coupled with realistic media considering all of their heterogeneities.

* Corresponding author.
E-mail address: bruno.blais@polymtl.ca (B. Blais).

https://doi.org/10.1016/j.cej.2024.158577
Received 7 August 2024; Received in revised form 2 December 2024; Accepted 11 December 2024
Available online 18 December 2024
1385-8947/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

Digitization techniques such as focused ion beam scanning electron microscopy (FIB-SEM) [13], X-ray microcomputed tomography (μCT) [14] and nuclear magnetic resonance (NMR) [15,16] have been used to obtain stacks of images representing complex media in a voxelized manner. These voxelized representations have both been used as-is with the Lattice Boltzmann method (LBM) [17,18], or have been used to construct volume meshes of the phases, to be simulated with the Finite Volume method (FVM) [19–24] or the Finite Element method (FEM) [18,25,26].

Starting from voxelized representations of real porous media has the advantage of keeping a realistic morphology (surface, volume, structure), but requires multiple pre-treatment steps when using FVM or FEM including [27]: (1) digitization, (2) segmentation of phases, (3) creation of a surface mesh, (4) decimation and smoothing of the surface mesh, and (5) generation of volume meshes. While recent work involving the digitization of porous media to run CFD simulations have used most of these steps, Kuhlmann et al. [27] highlighted the complexity of the process, the fact that strings of software need to be used to produce a mesh, and how expertise both in digital image processing and CFD is a rare occurrence. This inhibits usage and advancement in the field of PRCFD.

An alternative path to account for the complex morphologies of porous media in CFD is to avoid conformal meshes altogether, and instead use immersed boundary methods (IBM) [28–31]. This method of representing a solid embedded in a fluid was first introduced by Peskin [32] and was based on the addition of forcing terms in the Navier–Stokes equations. IBM have been implemented and used with packings of particles both for Finite Difference method (FDM) [33,34] and FVM [35,36]. Das et al. [28] have used IBM to solve the flow through packings of non spherical particles. To our knowledge, IBM have been used for flows in porous media obtained from digitized media only by Das et al. [30] (FVM), Ilinca and Hétu [37–39] (FEM) and Lesueur et al. [40] (FEM). Das et al. [30] used a sharp IBM to accurately account for the solid represented by a surface mesh when solving the flow. Ilinca and Hétu [37–39] solved the flow using the cut cell method by adding degrees of freedom on the embedded surface, creating a conformal mesh locally during assembly before these degrees of freedom are removed by static condensation. Lesueur et al. [40] moved and fitted the existing nodes of cut cells from a non-conformal grid to the solid surface before assembling the system based on this newly formed conformal mesh.

We believe that the scarcity of works starting from μCT to simulate the flow in porous media is also due to the prohibitive cost of evaluating the distance to highly detailed objects obtained from digitization. When solving the flow using IBM, the distance between each discretization point and the solid surface must be computed repeatedly, which is not viable when the solid is represented as a collection of millions of voxels or polygons, as is the case for porous media. To alleviate this issue, we propose a Radial Basis Function (RBF) network approach in which an approximate signed distance function (SDF) of the solid is encoded, as described by Liu et al. [41]. RBF networks are suitable to represent complex functions in space [42–44] and have been used to approximate distance fields accurately [45]. In addition, they provide complete support of the approximated functions in the vicinity of the nodes and can be used with compact basis functions, rendering RBF-encoded shapes portable and efficient in massively parallel simulations [41,46]. RBF networks representing solids have been used for structural topology optimization [41] and CFD around simple shapes, including: cylinders [47,48], a NACA0012 airfoil [49], and platelets (blood) [50]. However, the application of RBF to represent complex porous media and its coupling to large scale CFD simulations has not been demonstrated.

The purpose of this work is to extend the applicability of IBM to complex static solids (e.g. porous media) in a flexible, parallelized and accurate manner. We introduce a novel method of integrating RBF networks in a CFD framework that is based on distributed grids. We verify the implementation and workflow using a type of manufactured solutions that are applicable to any geometry. Finally, we formulate and validate experimentally a workflow to predict the flow field across real digitized porous monoliths in their entirety, focusing on flows at pore Reynolds number values below 30. Applications of such flows include environmental engineering (groundwater flow models [51,52]), filtration (e.g. membrane filtration [53]), heat exchange [54], microreactors [7], and fuel cells [55]. The result is a physics-based digital twin of a real porous medium that can predict both sub-millimetric flow characteristics from numerical solid representations, and provide macroscopic averaged properties that are experimentally measurable.

2. Methodology

To simulate the flow through real porous monoliths, we synthetize silicone monoliths (Appendix A), digitize them (Section 2.2.1), and construct RBF-network approximations of their geometry (Section 2.2.3). We verify the method and its orders of convergence (Section 3). We measure experimentally the pressure drop at various flow rates (Section 4.1) to compare to values obtained by simulation (Section 4.2), for validation. The workflow (excluding verification and validation) is shown in Fig. 1.

2.1. Finite element formulation and immersed boundary method

We use the FEM to solve the flow of the fluid phase by solving the incompressible Navier–Stokes equations, where $\boldsymbol{u}$ is the velocity, p is the pressure, ρ is the density, $\boldsymbol{\tau}$ is the deviatoric stress tensor, f is a source term and ν is the kinematic viscosity. We include the density in the pressure definition and use $p^* = \frac{p}{\rho}$.

$$\nabla \cdot \boldsymbol{u} = 0 \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p^* + \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f} \tag{2}$$

$$\boldsymbol{\tau} = \nu \left((\nabla \boldsymbol{u}) + (\nabla \boldsymbol{u})^T \right) \tag{3}$$

The solver used in this work is part of the open-source software Lethe [56], based on the deal.II library [57]. It uses distributed adaptive meshes from p4est [58] to enable simulations over a large number of cores (>10k). Lethe uses a streamline upwind Petrov–Galerkin and pressure-stabilizing Petrov–Galerkin (SUPG/PSPG) stabilized finite element formulation to solve the Navier–Stokes equations, which allows to use the same order of elements for velocity and pressure [56]. We use linear cubic elements for both velocity and pressure (Q_1Q_1), as well as quadratic-linear elements (Q_2Q_1) for verification. Transient simulations are realized using a backward difference formula (BDF) of first order [59].

We use more specifically Lethe's Sharp-interface Immersed Boundary Method (SIBM), which was used to simulate the flow around spherical particles [60], the flow–particle coupling of multiple particles [61,62] and the non-Newtonian flow around a sphere [63]. We use the SIBM to account for complex geometries while avoiding conformal mesh generation. SIBM requires a SDF and its gradient to properly apply the constraints to the discretized Navier–Stokes equations. The steps to obtain the SDF of a porous medium are described in the next subsections.

2.2. Digitization

We synthetize porous silicone samples (cylinders of 6 mm height and 8 mm diameter) following the method described in Appendix A, then digitize them using μCT. Even though the same protocol was used for each sample, they each have a different morphology due to the stochasticity of polymer blend extrusion and the subsequent coarsening process. We segment the data to generate a surface mesh representing

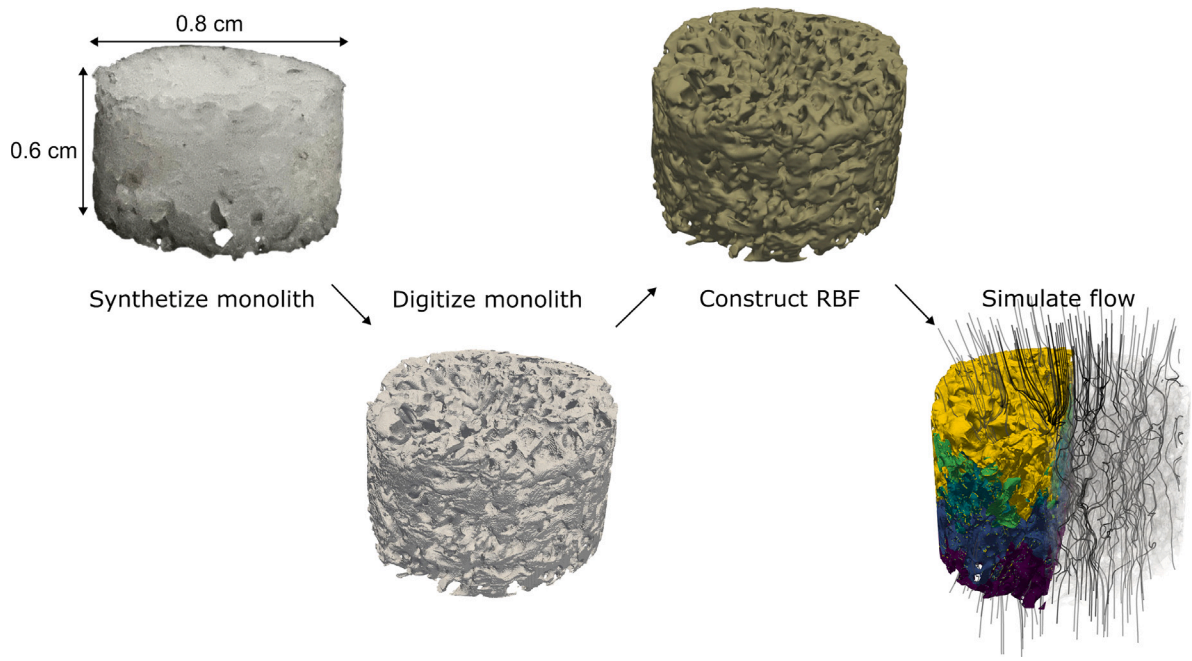

Fig. 1. Process of synthetizing a monolith (Appendix A), digitizing this monolith (Section 2.2.1), constructing a RBF-network representation of its shape (Section 2.2.3) and simulating the flow through its pores (Section 4.2). The flow simulation results show the monolith colored by the normalized pressure field at the surface, with the streamlines following the velocity field.

the solid, then train RBF networks to approximate the SDF around the digitized objects. The RBF-encoded SDF can then be used to represent the surface of the porous media implicitly at the zeroth isovalue. The μCT and RBF-network training steps correspond to the second and third steps in Fig. 1.

2.2.1. X-ray microcomputed tomography

We digitize all samples using μCT analysis with a ZEISS Xradia 520 Versa XRM instrument. We position each sample in the support used for the pressure drop experiments before the scan to ensure that the potential compression of pores is considered within the μCT scan. We set the voxel size of the captured images to be at most 10 μm. We use Dragonfly Pro Version 2020.1, Build 809 (Object Research Systems, Inc, Montreal, Quebec, Canada) to segment the phases (solid and pores). We apply a Gaussian filtering to the image, then identify the solid region visually to define a Region of Interest (ROI). We set the complementary region as the pores. We generate a smoothed surface mesh on the solid ROI after selecting a threshold parameter value, which removes the staircase-like effect associated to the voxelized representation of digitized media. As highlighted in the biomedical literature [64–66], the segmentation procedure can have a significant effect on the results obtained from subsequent analyses, especially for threshold-based segmentation. The threshold parameter is related to the intensity of voxel coloration, and must be set between 0 and 100% in Dragonfly. As the selected value increases, the resulting representation of the solid goes from full to more skeletal. Care must be exerted since the level of porosity can have a significant effect on the pressure drop. Given the lack of information for our specific combination of materials, μCT equipment and settings, we use threshold values of {1, 50, 99}% when generating surface meshes from solid ROIs to establish the sensitivity on the threshold of the solid reconstruction and, consequently, of the pressure drop.

2.2.2. Characterization of silicone monoliths

We use Dragonfly and its integrated workflow for pore network analysis using OpenPNM [67]. OpenPNM generates networks by using PoreSpy's Sub-Network of an Over-segmented Watershed algorithm [68,69]. The pore size and distance between connected pores distributions obtained are shown in Appendix B. The mean pore size (by volume) and standard deviation reported in Table 1 are based on the distribution of inscribed diameter (largest sphere that can fit in a pore), under the assumption of normal pore size distribution. The distance between pairs of pores is computed between their centroids under the same assumption.

Table 1
Porous characteristics of each sample: mean pore size d_p (by volume), pore size standard deviation σ_p (by volume), mean distance between connected pores d_c (by number), distance between connected pores standard deviation σ_c (by number), specific surface S_p and porosity ε.

#	d_p [μm]	σ_p [μm]	d_c [μm]	σ_c [μm]	S_p [cm^{-1}]	ε [%]
1	276	93	417	175	68	53
2	346	80	440	189	50	50
3	366	132	530	199	60	55

We compute the specific surface and porosity using the voxel volume and surface from the solid and pores ROI, and correct the surface by a factor 2/3, as proposed by Yeong and Torquato [70,71], to account for the surface voxelization into a staircase-like fashion of digitized random porous structures.

2.2.3. RBF network training

We use a Radial Basis Function (RBF) network generation tool based on the open-source Bitpit library [72], which takes advantage of its Parallel Balanced Linear Octree (PABLO), Levelset and RBF capabilities. The RBF network ($\tilde{\lambda}$) approximates the SDF (λ) of the shape by evaluating the contribution of each RBF-node based on the evaluation point ($\boldsymbol{x}$), basis functions (φ_i) and weights (γ_i), obtained from training. The expression of the RBF network is:

$$\lambda(\boldsymbol{x}) \approx \tilde{\lambda}(\boldsymbol{x}) = \sum_i \varphi_i(\boldsymbol{x})\gamma_i \tag{4}$$

The training steps are:

1. Load the STereoLithography (STL) file representing the porous medium obtained from μCT (Section 2.2.1);

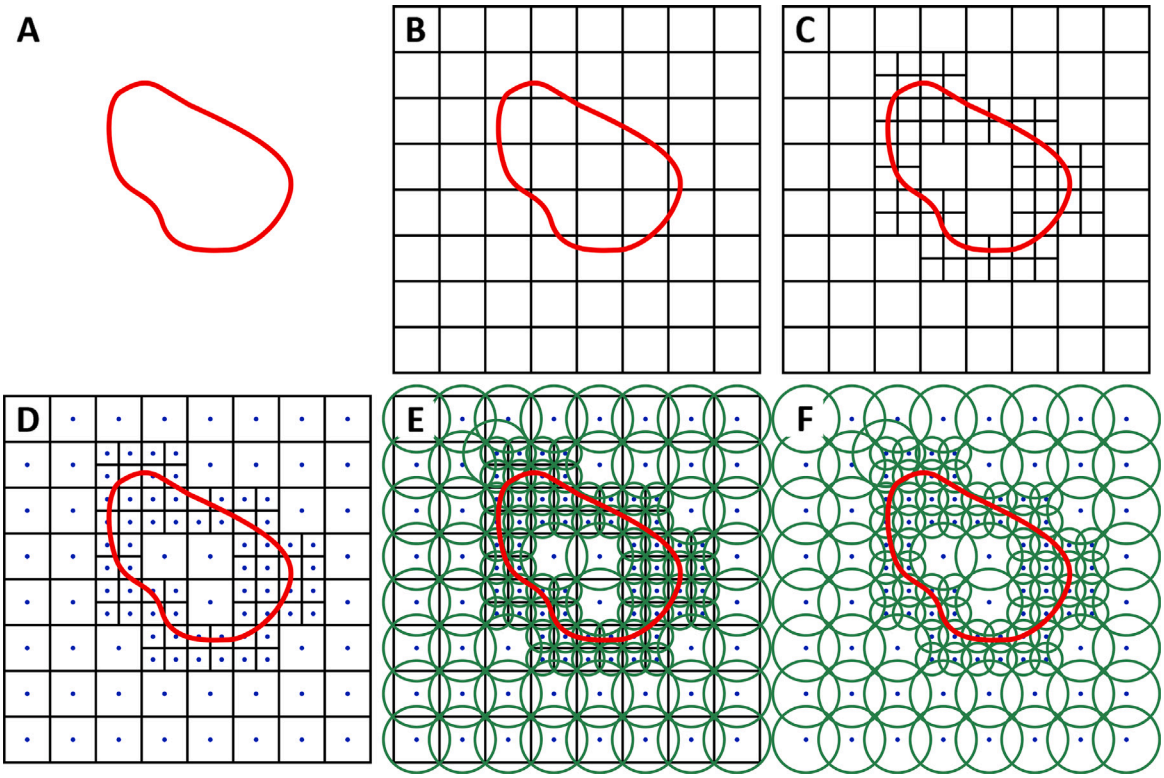

Fig. 2. Steps used to place RBF-nodes around an object and associate compact basis functions to each node. (**A**) Around a given object, (**B**) a uniform grid is formed (**C**) then adaptively refined near the boundary. (**D**) Nodes are placed at the center of each cell, (**E**) then a basis function is associated to each node. (**F**) The grid is not needed anymore.

2. Build a uniform cartesian grid of $(2^m)^3$ cells around it, based on a bounding box p percent larger than the object. We will generate RBF-nodes at each cell centroid; these will also be used as collocation points. This step and subsequent ones are illustrated in Fig. 2.
3. (Optional) Refine adaptively the cells cut by the object. Repeat this step n times to obtain the desired level of detail;
4. Assemble a system of equations structured as $\boldsymbol{A\gamma} = \boldsymbol{b}$. The elements of $\boldsymbol{A}$ and $\boldsymbol{b}$ are respectively $a_{i,j} = \varphi_j(\boldsymbol{x}_i)$ and $b_i = \lambda(\boldsymbol{x}_i)$, computed from the surface mesh. Each row i, corresponding to a cell centroid, is built as shown in Eq. (5), where j indicates the column of the matrix.

$$\lambda(\boldsymbol{x}_i) = \sum_j \varphi_j(\boldsymbol{x}_i)\gamma_j \tag{5}$$

   The matrix $\boldsymbol{A}$ is sparse when using compact support for the basis functions;
5. Solve the system to obtain the weights γ_i using a Least Squares Conjugate Gradient (LSCG) solver, because A is poorly conditioned when adaptive refinement is used. Conjugate Gradient (CG) solvers can be used otherwise.
6. Export the resulting RBF network.

The number of RBF-nodes can be minimized using a higher density of nodes close to the zeroth isovalue of the SDF. This ensures that the main features of the shapes are captured, while tolerating a higher error on the SDF further from the zeroth isovalue.

We choose basis functions which have a compact support to ensure a sparse linear system of equations. We use the C2 Wendland basis function for its distance approximation capabilities [73] and because it is twice continuously differentiable: this property is desirable for the SIBM which requires the gradient of the SDF [60,61].

$$\varphi_{\mathrm{i,Wendland}}(\boldsymbol{x}) = \begin{cases} \left(1 - \frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\kappa_i}\right)^4 \left(4\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\kappa_i} + 1\right), & \text{if } \frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\kappa_i} \le 1 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

The support radius of each RBF-node is $\kappa_i = \alpha h_i$, where α is a user-defined constant and h_i is the cell length. This definition of κ_i provides a way to use compact basis functions while capturing the features of the media. We use $\alpha = 3$ for this work since it is a good compromise between speed, efficiency and accuracy. Training is carried out on the Digital Research Alliance of Canada's cluster Narval, on a single node using 2 sockets with 32 AMD Rome 7532 (2.40 GHz 256M cache L3) processors and 249 GB of Random Access Memory.

2.3. RBF networks for immersed boundary application

2.3.1. Initialization

In Lethe, the initialization ensures that each cell has access to a list of the RBF-nodes whose contribution is possibly non-null over the cell, allowing us to remove superfluous data and speedup the RBF network usage. This step takes advantage of the parallel distributed hierarchical structure of the meshes within Lethe. From the coarsest level to the finest, RBF-nodes that do not contribute to the RBF network in locally owned cells are discarded. Starting from the coarsest level and moving information upwards allows to filter relevant RBF-nodes efficiently for each parallel process.

2.3.2. Evaluation

We only sum over RBF-nodes whose support radius overlaps with the evaluation point. The SDF and its gradient are:

$$\lambda(\boldsymbol{x}) \approx \sum_i \varphi_i(\boldsymbol{x})\gamma_i \tag{7}$$

$$\nabla\lambda(\boldsymbol{x}) \approx \sum_i \nabla\varphi_i(\boldsymbol{x})\gamma_i \tag{8}$$

To reduce the computational cost, a caching mechanism is used to rapidly sum over the RBF network.

3. Verification

The SIBM implementation in Lethe has been extensively verified by Barbeau et al. [60–62] and Daunais et al. [63]. It preserves the order of convergence associated to the underlying FEM scheme. In this section, we introduce and use an approach to manufacture solutions around arbitrary shapes. We focus specifically on the verification of the generalization of the SIBM to superquadrics (with implicit Eq. (9)) and the same geometries, defined by RBF networks. In Eq. (9), $\{a, b, c\}$ are the half-lengths and $\{p, q, r\}$ are the blockiness factors. d represents the levels of the implicit superquadric function, with the zeroth isovalue corresponding to the surface itself where $d(\boldsymbol{x}_{\text{surface}}) = \lambda(\boldsymbol{x}_{\text{surface}}) = 0$. It must be noted that while $d(\boldsymbol{x}_{\text{surface}}) = 0$, d is not a SDF function *per se* as $|\nabla d(\boldsymbol{x})| \neq 1$.

$$d(\mathbf{x}) = \left|\frac{x}{a}\right|^p + \left|\frac{y}{b}\right|^q + \left|\frac{z}{c}\right|^r - 1 \tag{9}$$

3.1. Method of manufactured solutions

We use the Method of Manufactured Solutions (MMS) [74] to verify the proper implementation of the SIBM to analytically defined superquadrics and their RBF network approximations. The SIBM in Lethe uses the SDF to locate the boundary and subsequently impose the proper constraints to the system. We use a similar idea to construct the manufactured solutions for velocity: we multiply base velocity functions by a function that is zero at the surface (e.g. d, λ) to obtain a manufactured solution that has zero velocity at the solid surface. This approach can manufacture solutions for any SDF.

The base solutions are, for u_b, v_b, w_b and p being the three components of velocity and the pressure, respectively:

$$u_b = \sin^2(\pi x)\cos(\pi y)\sin(\pi y)\cos(\pi z)\sin(\pi z) \tag{10}$$

$$v_b = \cos(\pi x)\sin(\pi x)\sin^2(\pi y)\cos(\pi z)\sin(\pi z) \tag{11}$$

$$w_b = -2\cos(\pi x)\sin(\pi x)\cos(\pi y)\sin(\pi y)\sin^2(\pi z) \tag{12}$$

$$p = \sin(\pi x) + \sin(\pi y) + \sin(\pi z) \tag{13}$$

The manufactured solution for a superquadric shape is:

$$u = \left(\left|\frac{x}{a}\right|^p + \left|\frac{y}{b}\right|^q + \left|\frac{z}{c}\right|^r - 1\right)\sin^2(\pi x)\cos(\pi y)\sin(\pi y)\cos(\pi z)\sin(\pi z) \tag{14}$$

$$v = \left(\left|\frac{x}{a}\right|^p + \left|\frac{y}{b}\right|^q + \left|\frac{z}{c}\right|^r - 1\right)\cos(\pi x)\sin(\pi x)\sin^2(\pi y)\cos(\pi z)\sin(\pi z) \tag{15}$$

$$w = \left(\left|\frac{x}{a}\right|^p + \left|\frac{y}{b}\right|^q + \left|\frac{z}{c}\right|^r - 1\right)\left(-2\cos(\pi x)\sin(\pi x)\cos(\pi y)\sin(\pi y) \times \sin^2(\pi z)\right) \tag{16}$$

$$p = \sin(\pi x) + \sin(\pi y) + \sin(\pi z) \tag{17}$$

It must be noted that the velocity field is not solenoidal, thus a source term must be added to the continuity equation.

Three shapes are considered for verification: a sphere, a convex superquadric, and a concave superquadric. Their appearances and associated parameters are shown in Fig. 3 and Table 2.

3.2. Superquadrics and uniform RBF

We verify the SIBM in two steps, with their results shown in Figs. 4 and 5. Fig. 4 shows the decrease of the L2-norm of the error of velocity ($\|e_u\|_{\mathcal{L}2}$) and pressure ($\|e_p\|_{\mathcal{L}2}$) in a simulated domain ($[0,1]^3$) when using Q_1Q_1 elements. Fig. 5 shows the same information when using Q_2Q_1 elements. In this section, RBF representations are built without adaptive refinement (see 2.2.3).

$$\|e_u\|_{\mathcal{L}2} = \left(\int_\Omega \|\boldsymbol{u}_{\text{analytical}} - \boldsymbol{u}_{\text{approximated}}\|^2 d\Omega\right)^{\frac{1}{2}} \tag{18}$$

Table 2
Parameters used to generate the shapes for verification.

Shape	Parameter	Value
Sphere	Radius	0.50
Convex superquadric	a	0.50
	b	0.75
	c	0.25
	p	1.50
	q	1.50
	r	1.20
Concave superquadric	a	0.50
	b	0.50
	c	0.50
	p	0.60
	q	0.60
	r	2.00

$$\|e_p\|_{\mathcal{L}2} = \left(\int_\Omega \|p_{\text{analytical}} - p_{\text{approximated}}\|^2 d\Omega\right)^{\frac{1}{2}} \tag{19}$$

First, we verify the implementation of the superquadrics class of shapes. The velocity and pressure L2-errors for the flow around superquadrics are represented by full lines. We calculate the orders of convergence n for the last two grids. We notice that the asymptotic convergence region is reached for all superquadrics when Q_1Q_1 elements are used. In the case of Q_2Q_1 elements, concave superquadrics lead to a slightly inferior convergence rate. This is most likely due to the presence of the singularities and/or sharp features in the geometry.

Second, we verify that RBF-encoded shapes that represent superquadrics (dotted lines) recover the same orders of convergence obtained using analytical superquadrics. By increasing the number of RBF nodes used from 8^3 to 64^3, the results for both velocity and pressure errors collapse gradually on the analytical superquadrics curves. It is quite apparent, when using similar y-axis scales, that the error stops decreasing with cell size for coarse RBF networks regardless of the degree of elements used. The representation error dominates and prevents the total error from decreasing, especially for $\|e_p\|_{\mathcal{L}2}$.

3.3. Adaptive RBF

We verify that using adaptive refinement when producing the background grid during RBF generation greatly decreases the memory and training cost while preserving accuracy. To quantify this effect, adaptively refined RBF are compared to uniformly refined RBF. We compare results obtained using both types of grids with an equal cell size at the solid boundary. The Lethe background grid is composed of 64^3 elements. RBF-generation, based on the convex superquadric (middle shape in Fig. 3), uses a base grid of $(2^{7-n})^3$ cells, where n is the number of adaptive refinements. The grid is subsequently adaptively refined n times before the generation of RBF-nodes. This implies that, for all numbers of adaptive refinements tested, the distance between RBF-nodes close to the solid surface is the same.

Fig. 6 shows that decreasing the density of RBF-nodes far from the surface substantially decreases the solution time and memory requirements of the RBF-encoding, by factors 10 and 100, respectively. After two cycles of adaptive refinement of the RBF-producing background grid, the costs stop decreasing, which suggests that while this strategy is advantageous, its effects reach a plateau value. Specifically, the plateau is reached because almost all initial cells get refined at least once when $n = 3$, since they are so large that they overlap with the shape. The first adaptive refinement cycle is therefore quasi-uniform. Increasing the padding of the bounding box could delay the plateau to higher values of n, but this approach would not reduce the absolute solution time or memory requirements.

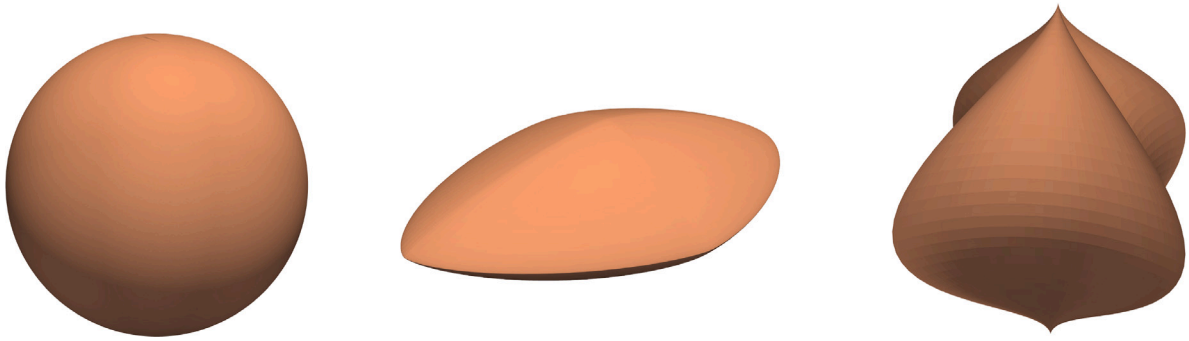

Fig. 3. Shapes used for the verification of the flow around superquadric and RBF-encoded shapes.

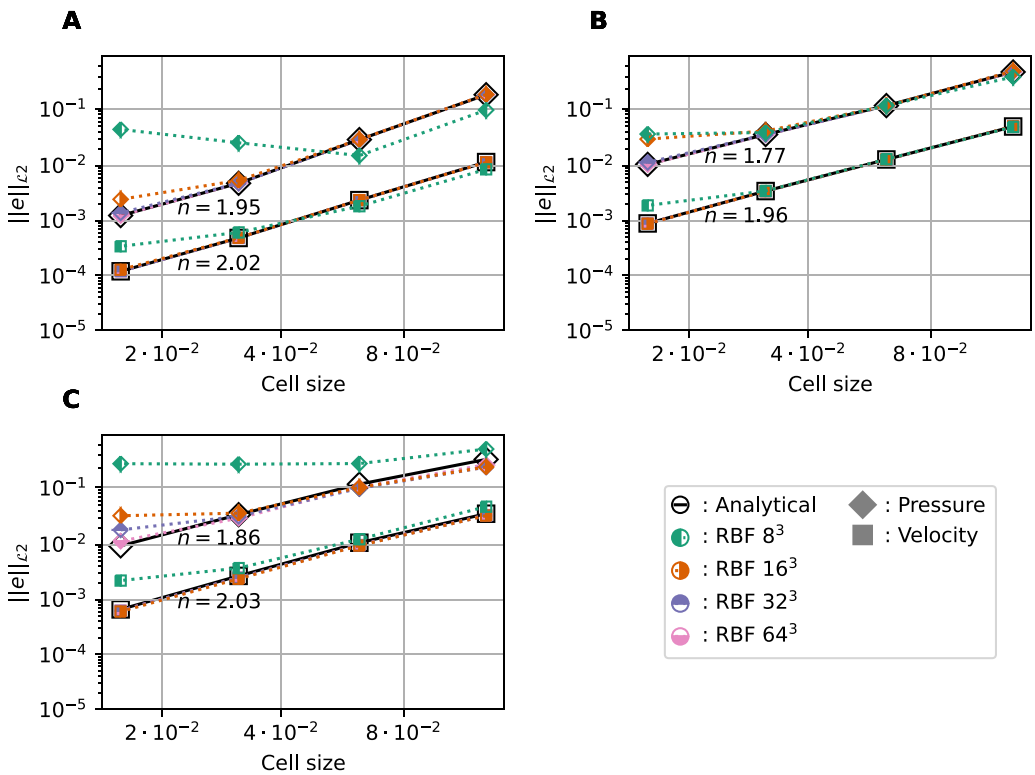

Fig. 4. $\mathcal{L}2$ norm of the error of the MMS for the velocity and pressure fields with Q_1Q_1 elements around **A** a sphere, **B** a convex superquadric (Conv.Sup.) and **C** a concave superquadric (Conc.Sup.), both for analytical (full lines) and for RBF representations (dotted lines). Diamond and square markers signify pressure and velocity errors, respectively. For RBF representations, markers are half-filled in varying orientations depending of the number of RBF-nodes. n is the order of convergence for the last two grids.

4. Validation in porous monoliths

The SIBM implementation in Lethe has been validated by Barbeau et al. [61] for the coupled flow around moving spherical particles: Ten Cate sedimentation case [75] ($Re \leq 31.9$), positively buoyant sphere ($Re = 2500$), pair of drafting, kissing and tumbling (DKT) particles [76] ($Re = 233$), and sedimentation of 64 particles ($Re = 20$). We focus on validating the flow through monoliths at Reynolds number values under 30.

4.1. Experimental pressure drop evaluation setup

We measure the pressure drop at different volumetric flows ($Q = \{0.6, 0.9, 1.2, 1.5, 1.8\}$ mL/s) through three (3) supported porous monolith samples using a controlled syringe pump. This DIY syringe pump (RobotDigg, Shanghai, China) is connected to the support with an assembly of two check valves (0.5 psi break pressure) and a mixing valve that allow the pump to automatically refill itself, allowing for fewer manipulations and easier repetitions. During the filling phase, water flows from the feed reservoir into the syringe, while the check valve between the mixing valve and the monolith (C1) remains closed due to negative relative pressure. When the syringe is emptied, water is directed to the monolith as C1 is then subjected to positive pressure; at the same time, the check valve between the syringe and the feed reservoir (C2) is blocked. The valves are pressure-actuated. A differential pressure sensor (MPX5010DP from NXP, 0–10 kPa range) and a cylinder filled with water are also connected for pressure drop measurements and sensor calibration. The pressure sensor is connected to the tubing network on one side and open to the atmosphere on the other. Sensor calibration is done using a water column when the 3-way plug valve is correctly positioned. The samples are fit in the support using latex tubing having an inner diameter of 8 mm. Fig. 7 shows the elements composing the experimental setup and a photograph of the setup is provided in Appendix C. We use distilled water at a temperature of 24 °C.

We calibrate the pressure sensor in an initial step during which the 3-way plug valve connects a 600 mm water column to the sensor.

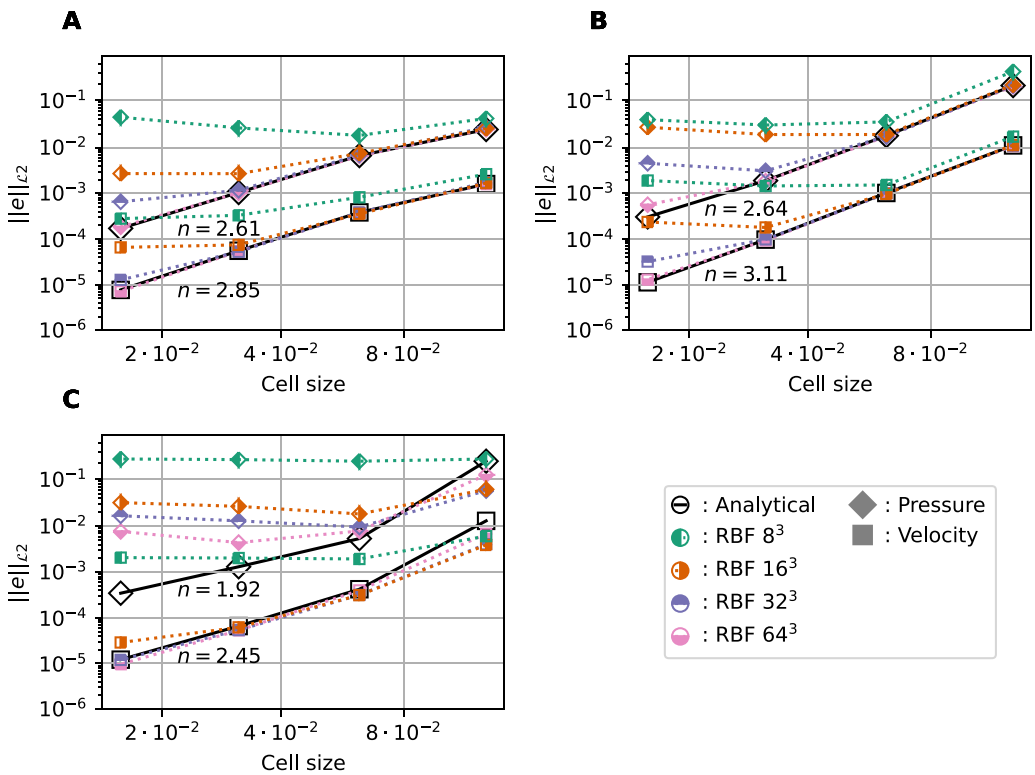

Fig. 5. $\mathcal{L}2$ norm of the error of the MMS for the velocity and pressure fields with Q_2Q_1 elements around **A** a sphere, **B** a convex superquadric (Conv.Sup.) and **C** a concave superquadric (Conc.Sup.), both for analytical (full lines) and for RBF representations (dotted lines). Diamond and square markers signify pressure and velocity errors, respectively. For RBF representations, markers are half-filled in varying orientations depending of the number of RBF-nodes. n is the order of convergence for the last two grids.

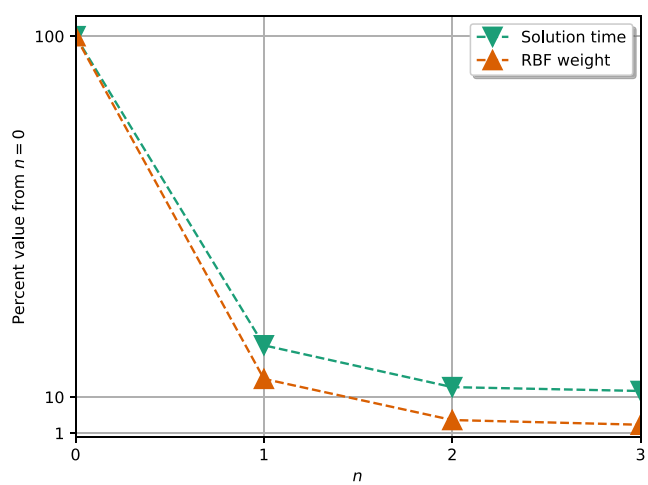

Fig. 6. Relative value for solving time and RBF memory requirement compared to when only uniform refinement is used.

Before testing each sample, we execute an empty run with the same sequence of volumetric flows. We subtract the values of the pressure drop when the sample support is empty from the measured pressure drop to isolate the porous monoliths contribution, which are to be compared with simulation results. The data for empty runs are shown in Appendix D. In addition to storing monoliths in distilled water, we run cycles of low and high flow rates prior to measurements to ensure that any residual air is removed and that the flow is monophasic.

Table 3
RBF encoding parameters.

Parameter	Value
m	6
n	2
α	3
p	10
Basis function	C2 Wendland

4.2. Simulation setup

We validate the methodology by comparing the simulated and measured pressure drops of each sample. For each sample, we generate representations at various thresholds $\{1, 50, 99\}$% (low extreme, midpoint, high extreme), based on the parameter explained in Section 2.2.1. We then use the RBF-encoding tool with the parameters shown in Table 3 to construct RBF networks for both porous monoliths and the latex sleeve surrounding samples, to ensure that the domain is properly represented. Fig. 8 shows 3D views of the setup as exported from Dragonfly. The sleeve is tightly fitted around the sample to avoid any bypass of the flow, and since both the sleeve and sample are flexible and deformable they are digitized together.

We set the CFD parameters used in Lethe as shown in Table 4; length unit is centimeter, time unit is second and mass unit is gram. We run the simulations in transient mode until a steady state is reached because the stiffness of the problem, the complex geometry, and the fact that the initial solution is too far from the stationary solution prevent the steady-state solver from converging. The elements are of degree 1 (second order accurate) for both velocity and pressure (Q_1Q_1). The inlet boundary conditions are set for velocity ($\boldsymbol{u} = [u_x, u_y, u_z]$) as

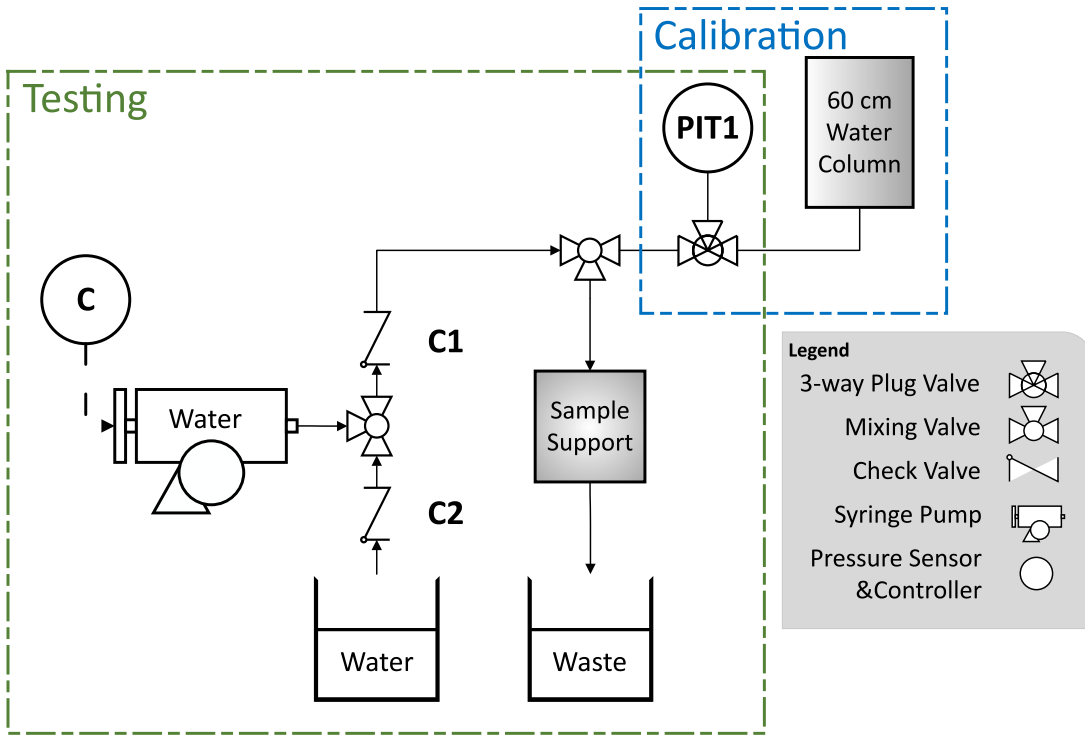

Fig. 7. Schematic of the experimental setup used to automatically impose set volumetric flows to measure pressure drops in porous monolith samples.

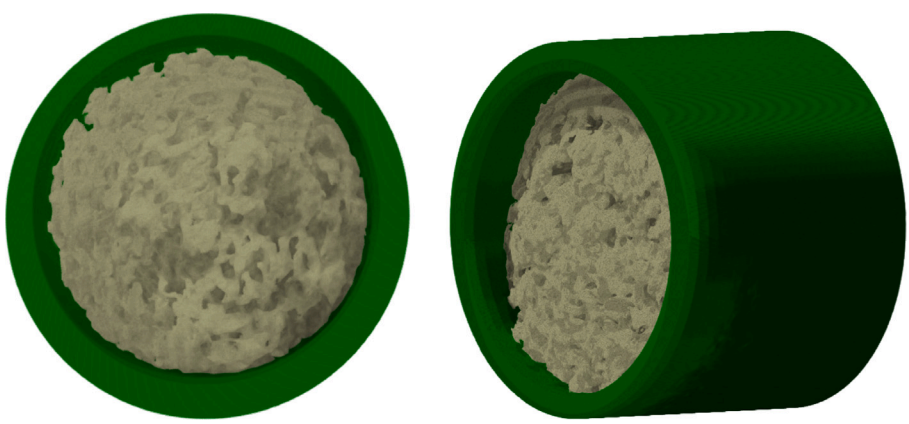

Fig. 8. 3D views (front and side) of a porous monolith and surrounding sleeve, exported from Dragonfly.

Dirichlet, as $u_x = \{0.6, 0.9, 1.2, 1.5, 1.8\}$, $u_y = 0$ and $u_z = 0$. The side walls boundary conditions are set to $\boldsymbol{u} = \mathbf{0}$. The outlet boundary is set to $\nu\nabla\mathbf{u}\cdot\mathbf{n} - p^*\mathcal{I}\cdot\mathbf{n} = \mathbf{0}$, where $\boldsymbol{n}$ is the boundary normal vector, and $\mathcal{I}$ is the identity matrix. The initial solution is $u_x = \{0.6, 0.9, 1.2, 1.5, 1.8\}$, $u_y = 0$, $u_z = 0$ and $p^* = 0$. The background grid is a hyperrectangle defined by the diagonal $[-1.5, -0.5, -0.5] : [1.5, 0.5, 0.5]$ and consists of only cubic cells.

4.3. Grid sensitivity analysis based on pressure drop

We use h-adaptive refinement at fluid–solid interface to generate the grids used for validation. Given the multiple inlet velocities and thresholds tested, we produce a grid convergence study for each extremum combination starting from a cartesian grid of $[3, 1, 1] \times 2^6$ cells in $[x, y, z]$ directions, adaptively refined thrice. We use a three point Richardson extrapolation based on the Q_1Q_1 interpolation's second order of convergence and compute the grid convergence indices (GCI) using a security factor (FS) of 1.25 [74,77]. Grids are systematically refined between each level presented, but remain non-uniform. We

Table 4
CFD parameters for the flow through porous media, where time units are seconds and length units are centimeters.

Parameter	Value
Time integration scheme	BDF1
Time step	$1e^{-4}$
Time end	$2e^{-3}$
Density	1
Kinematic viscosity	$1e^{-2}$
Element type	Q_1Q_1
Initial solution	$u_x = \{0.6, 0.9, 1.2, 1.5, 1.8\}$
Inlet boundary	$u_x = \{0.6, 0.9, 1.2, 1.5, 1.8\}$
Side wall boundary	$\boldsymbol{u} = \mathbf{0}$
Outlet boundary	$\nu\nabla\mathbf{u}\cdot\mathbf{n} - p^*\mathcal{I}\cdot\mathbf{n} = \mathbf{0}$
Domain dimensions	$[3, 1, 1]$

Table 5
Simulated pressure drops and GCI at various thresholds, flows and cell sizes, with a three point Richardson extrapolation (order of convergence = 2).

Inlet velocity [cm/s]	0.6			1.2			1.8		
Threshold [%]	1	50	99	1	50	99	1	50	99
Smallest cell size [μm]	Δp [kPa]								
19.5	0.438	0.340	0.175	1.07	0.795	0.377	1.97	1.42	0.631
13.0	0.381	0.312	0.164	0.876	0.679	0.349	1.53	1.15	0.572
9.77	0.363	0.293	0.161	0.822	0.648	0.341	1.41	1.10	0.555
Richardson extrapolation	0.349	0.273	0.158	0.780	0.624	0.335	1.32	1.06	0.542
Cell size [μm]	GCI [%]								
19.5	–	–	–	–	–	–	–	–	–
13.0	15.0	9.09	6.40	22.1	17.1	7.78	28.4	23.5	10.2
9.77	7.74	10.4	3.30	10.5	7.74	3.86	13.7	8.51	4.89

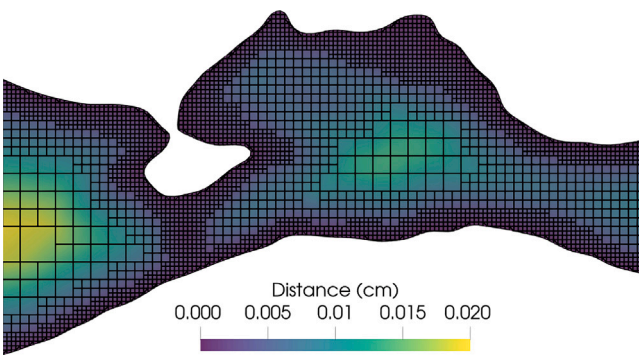

Fig. 9. Sliced view of a pore, with more refined cells near the solid.

consider only sample #1, given its lowest pore size and under the assumption that the higher Reynolds numbers give the most mesh sensitive results.

We conclude from Table 5 that using a cell size of 9.77 μm at the solid surface produces results with a discretization error of at most 14% for pressure drop predictions. The combination of a threshold of 1% and an inlet velocity of 0.6 cm/s results in the highest discretization error, which is of the magnitude of variations induced by the selection of the threshold parameter, and as such is considered acceptable. Fig. 9 shows a sliced view of a pore, with more refined cells near the solid surface, which is a result of distance-based adaptive refinement. For completeness, we included a similar 3D view in Appendix E. We reduce the number of required cells by a factor 16 by using adaptive mesh refinement. The final grids therefore have cell sizes between 9.77 and 78.13 μm; the total number of cells reaches 200M and the simulations are run using 8000 cores spread across 200 compute nodes from the Digital Research Alliance of Canada's cluster Niagara. Each of Niagara's node uses 2 sockets with 20 Intel Skylake cores (2.4 GHz, AVX512), for a total of 40 cores per node. The network connection is a 100 Gb/s EDR Dragonfly+. Each node has access to 202 GB of Random Access Memory.

4.4. Validation with silicone porous monoliths

We compare the simulation results of each porous monolith, with a range of applied flow rates $Q = [0.6, 1.8]$ mL/s and with surface meshes generated using different thresholds $\{1, 50, 99\}$%. We discard simulated pressure drops at 99% threshold because these values are unrealistically low and do not align with experimental observations (see Appendix F). We show the results for each sample in Fig. 10. The experimental error bars correspond to the confidence intervals using a T-Student distribution, with a significance level $\alpha = 0.05$. The agreement is slightly worse at low flow rates for samples #2 and #3, but still acceptable. We attribute this difference to the low signal obtained from the pressure sensor below 0.2 kPa and the low accuracy of the system for small flow rates. The experimental error is of the same magnitude as the error induced by the selection of the threshold parameter, which confirms that the cell size selected in Section 4.3 is appropriate. None of the three main sources of error for validation (discretization, threshold and pressure measurements) stand out or would be worth reducing.

5. Analysis

We use the CFD results of the flow through each sample to gain insight on the phenomena at the microscopic level. The pressure field is first observed slice by slice along the flow axis, both in terms of mean and standard deviation. Then, we highlight the impact of the principal pathways (pore and channels networks) on the progression of pressure fronts and local Reynolds numbers through a sample.

5.1. Pressure drop along the flow axis

The mean pressure $\bar{p}$ is taken as surface average of the fluid phase of a given slice at axial position x, as described by Eq. (20), where θ is the variable (p here). The standard deviation at each axial position is described by Eq. (21). The mean pressure drop along the axis is shown in Fig. 11, as well as the standard deviation of the pressure in Fig. 12. The Reynolds number is defined as the porous Reynolds number in Eq. (22) and the Darcy velocity (u_{Darcy}) is defined in Eq. (23), computed with volumetric flow Q, cross-sectional area A_{section}, and porosity [11].

$$\bar{\theta}(x) = \left(\frac{\iint_{\Omega_{\text{fluid}}} \theta \, \mathrm{d}\Omega}{\iint_{\Omega_{\text{fluid}}} \mathrm{d}\Omega} \right)_x \tag{20}$$

$$\sigma_\theta(x) = \left(\frac{\iint_{\Omega_{\text{fluid}}} \left(\theta - \bar{\theta}(x)\right)^2 \mathrm{d}\Omega}{\iint_{\Omega_{\text{fluid}}} \mathrm{d}\Omega} \right)_x^{1/2} \tag{21}$$

$$Re = \frac{u_{\text{Darcy}} d_{\text{solid}}}{\nu} = \frac{Q d_{\text{solid}}}{A_{\text{section}} \varepsilon \nu} \tag{22}$$

$$u_{\text{Darcy}} = \frac{Q}{A_{\text{section}} \varepsilon} \tag{23}$$

Mean pressure evolutions (Fig. 11) indicate that, at this scale, the topology of the porous networks act as the principal factor to describe the pressure evolution through the monoliths: the threshold parameter and flow rate affect the pressure by a factor, but not the shape of the curves. This makes sense given that all flow rates tested correspond to a laminar regime, with $Re \in [6.8, 27.1]$. The differences between the normalized curves are therefore entirely due to the unique structure of each monolith. In agreement with these observations, the pressure standard deviations (Fig. 12) follow the same patterns for each simulated rate, exhibiting sharp variations at the same locations. The only exception to the observation regarding the threshold is the presence of localized peaks for sample #3 in Fig. 12. Sample #3 with a threshold of 50% exhibits standard deviation peaks that slightly attenuate at higher flow rates. A hypothesis for this phenomenon is that some smaller

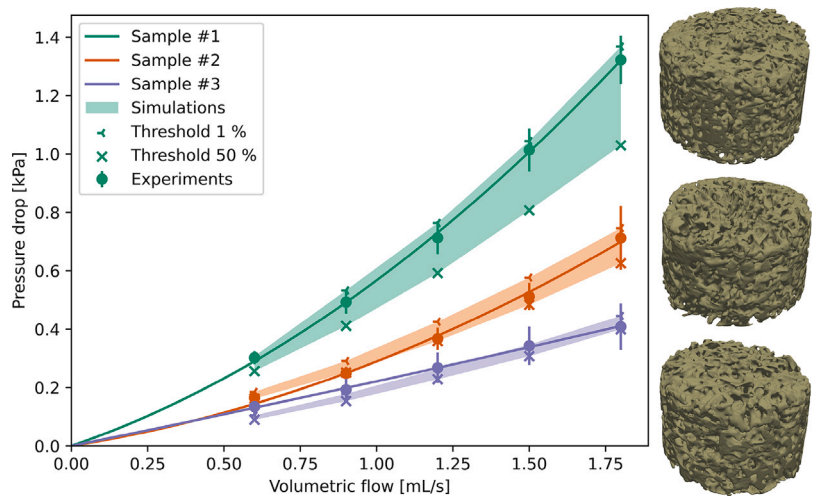

Fig. 10. Experimental and simulated pressure drops of three silicone porous monoliths (average pore size: 300 μm). Darkened zones represent the values covered by thresholds between 1% and 50%. Experimental error bars are the T-Student's confidence interval using $\alpha = 0.05$ (n = 5 number of repetitions). Each porous monolith (right) represents a sample and is aligned to its corresponding curve.

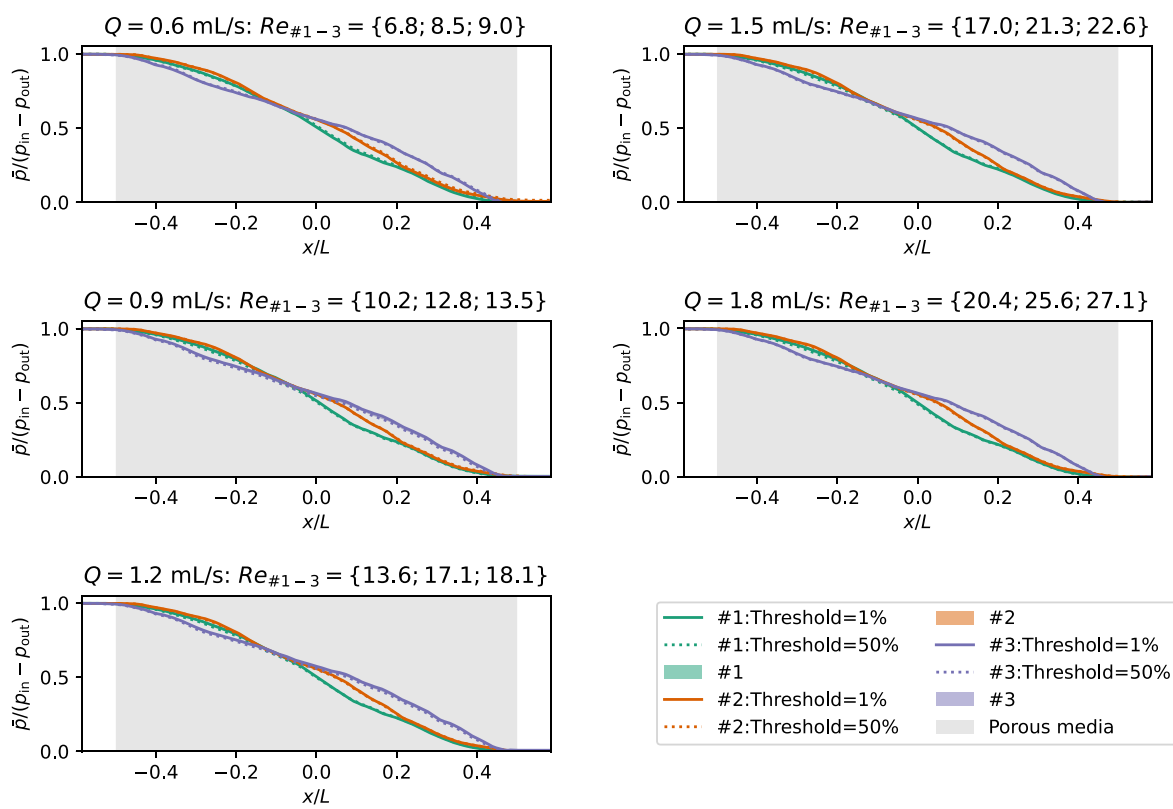

Fig. 11. Mean pressure evolution along the flow axis. Each subplot represents a set volumetric flow rate. Reynolds numbers vary for a given flow rate due to specific monolith properties.

channels are blocked when the threshold is 1%, while these same channels are open for a threshold of 50%. Observing this phenomenon only for sample #3 would be due to its high standard deviation for the pore diameter: 132 μm (+42% higher than others).

In addition, entry and exit regions are apparent and represent the major part of the domain; the slope of $\bar{p}$ varies throughout the monoliths and tends to zero at the edges. This is due to the varying compression level of the pores depending on the axial position. As seen in Fig. 13: the regions closer to edges (entry and exit zones) present higher porosity, going from about 0.4 (center of the domain) to 1 (edges). This effect is possibly due to the synthesis process, but also to the way the monoliths are pushed into the support in a tight manner to avoid bypassing. The monoliths are therefore compressed more near the center of their axial position while the pores at the edges remain more open. The resulting effect is that regions at the edge are more porous, resulting in pressure gradients closer to zero.

5.2. Preferential channels and pressure fronts

The porous silicone monoliths exhibit heterogeneities: different pore scales, structures appearing over multiple slices and a few principal channels. Because of the variability of the channels structure and their

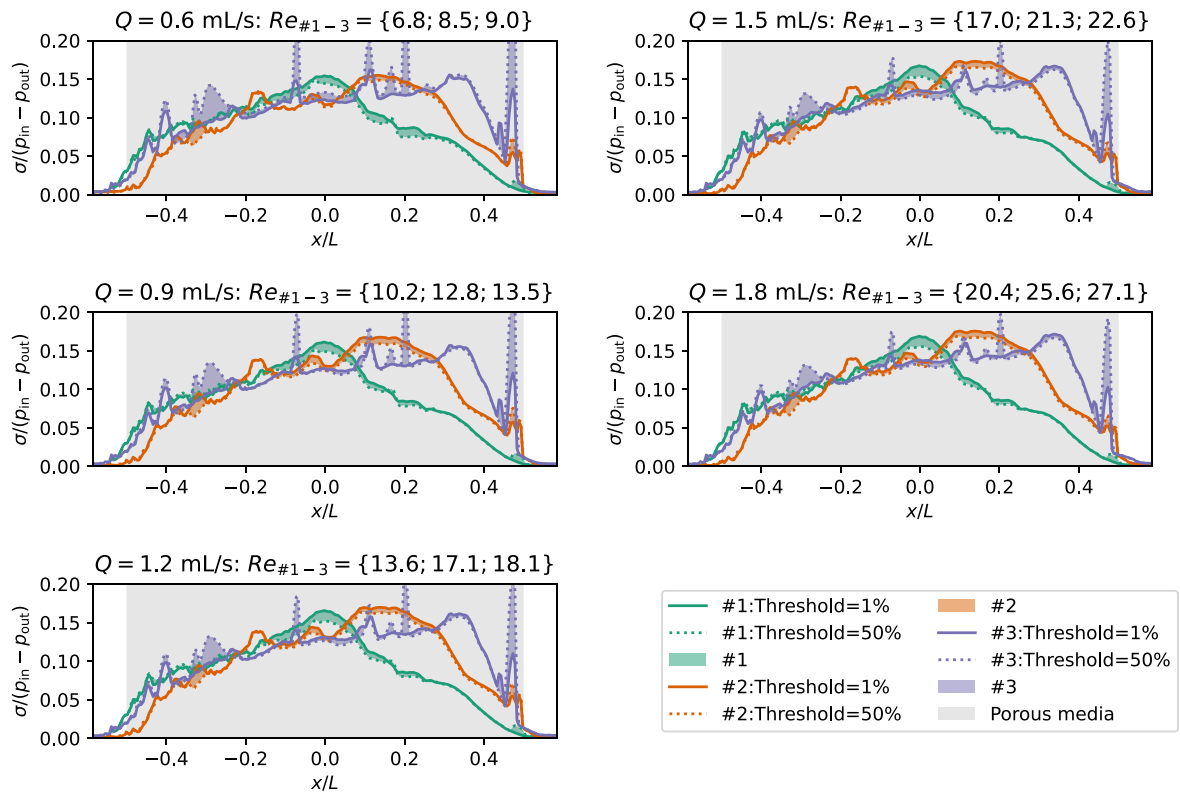

Fig. 12. Standard deviation (σ) of pressure along the flow axis. Each subplot represents a set volumetric flow rate. Reynolds numbers vary for a given flow rate due to specific monolith properties.

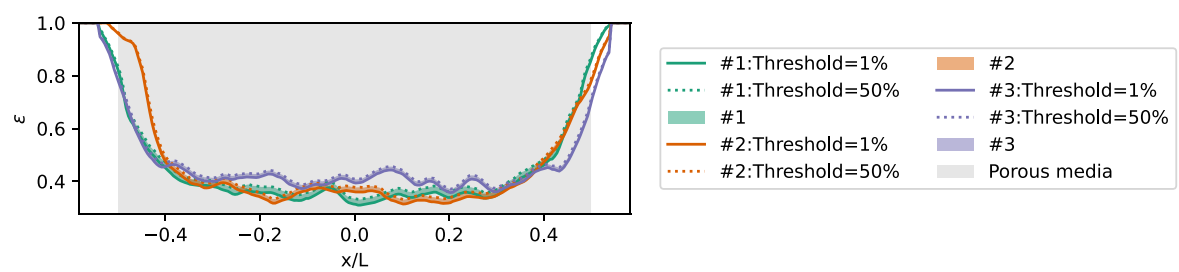

Fig. 13. Local porosity of the porous monoliths along the flow axis, for thresholds of 1 and 50%.

high impact on the pressure field, we can expect the fluid elements to progress along the flow axis at varying rates, which can be seen by the progression of pressure fronts and which is confirmed by the local Reynolds numbers calculations. Fig. 14 shows the locally computed Reynolds number ($Re(\boldsymbol{x}) = \lambda(\boldsymbol{x})\|u\|/\nu$) and pressure field through multiple slices for sample #2, at a flow rate of 1.2 mL/s, with a threshold of 50%. The slice at $x = 0$ shows the magnitude of range of pressures that can be observed in a single slice, where variations can reach half the total pressure drop. More specifically, the pressure field at this position has both yellow and dark blue regions, which indicate a range of pressure of 0.2 kPa.

Fig. 15 highlights that preferential channels go through the monolith with localized high isovalues of Re. For example, a major preferential channel is visible in subfigures A and B at midheight, which coincides with the high Reynolds number peaks at the center of every slice of Fig. 14. A few dead zones are also apparent and are in agreement with Fig. 14. The regions surrounding the main central channel contain thick walls and pores only slightly connected to the pore network. In Fig. 15, these regions are especially apparent in subfigures B and D in the second and third quartiles of the height.

6. Discussion

6.1. Flow regimes

We choose the pore size mean as the characteristic length when computing Reynolds numbers. However, as seen from Table 1, pore size distributions of our samples are wide: large and small pores are present. Table 6 shows pore Reynolds numbers computed using extrema pore sizes (addition or subtraction of two standard deviations). For each sample, Table 6 highlights the sensitivity of the Reynolds number to this decision. Sample #3, for example, shows variations up to a factor 6 for the Reynolds number between its extrema.

The flow regimes are laminar, but they deviate from Darcy flow around the upper bounds of porous Reynolds number studied, while remaining steady flows [78].

6.2. Limitations

For a problem of the same scale (dimension, pore sizes), increasing the flow rate would require finer grids to properly capture thinner boundary layers and, potentially, recirculation zones. Furthermore, it

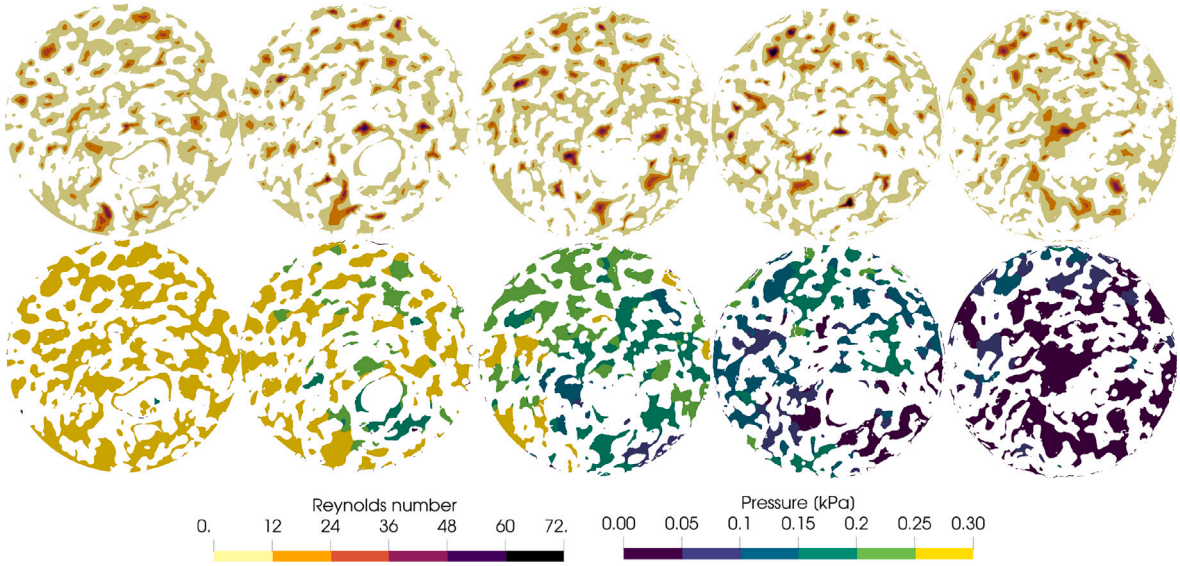

Fig. 14. Local Reynolds number (top row) and pressure field (bottom row) evolving along the flow axis for sample #2, at a flow rate of 1.2 mL/s, with a threshold of 50%. The slices are taken at $x = \{-0.2, -0.1, 0, 0.1, 0.2\}$ cm.

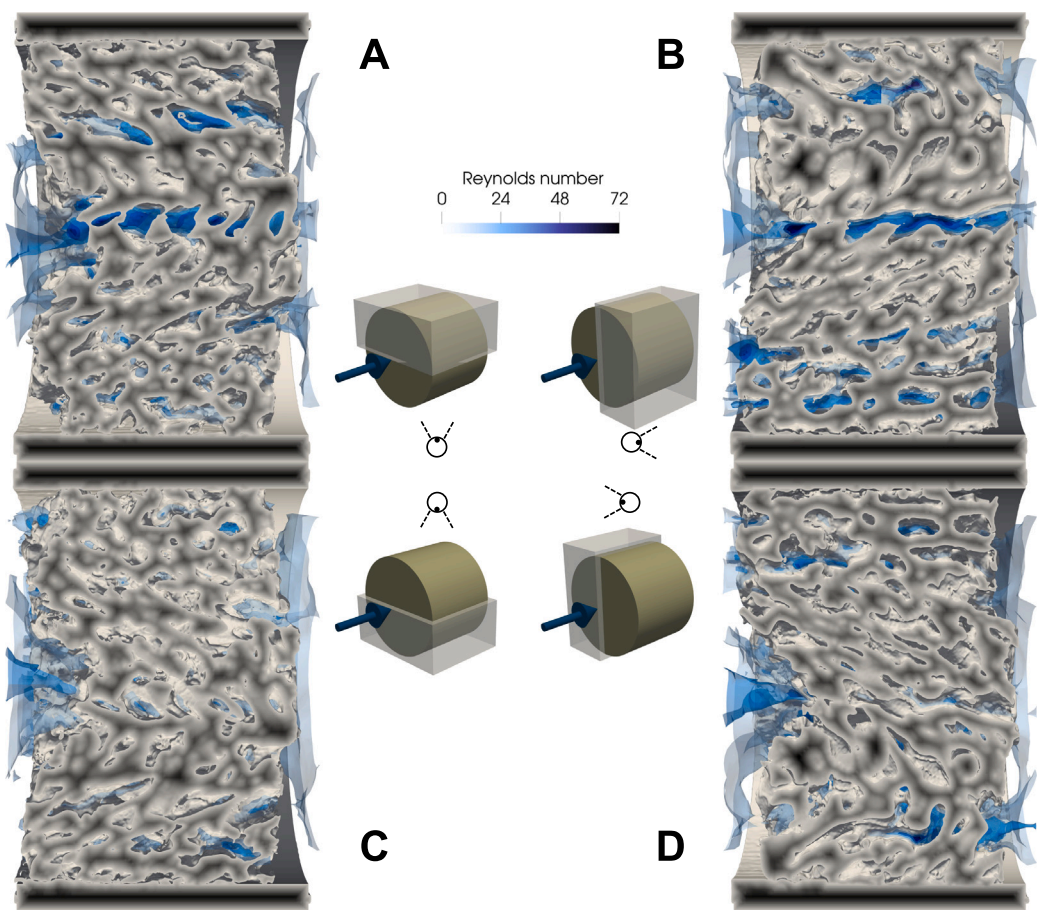

Fig. 15. Reynolds number isovalues through sample #2, at a flow rate of 1.2 mL/s, with a threshold of 50%. Eye symbols represent the points of view of each half. The arrows represent the flow direction.

is unclear if such flow would be steady. A grid composed of cells twice finer than the ones set in Section 4.3 would lead to a problem eight times larger. While challenging to solve, it would be possible with sufficient computing resources. It would theoretically be possible to handle much finer discretizations using a matrix-free approach [79], but this type of method is unavailable within our current implementation of the IBM.

In porous media of higher porosity and pore size (e.g. open-cell foams), it would be possible to handle problems in the transition regime. Limitations then would lie in the overall computing costs due

Fig. A.16. Fabrication of the silicone-based porous monoliths: PS/PLA 50/50 wt% blend is extruded, PS is selectively extracted, then silicone is injected in the PLA mold after which the PLA is also removed.

Table 6
Pore Reynolds numbers for each sample, for various volumetric flows ($Q = \{0.6, 0.9, 1.2, 1.5, 1.8\}$ mL/s) and using the pore sizes at the 2.28th, 50th and 97.72th percentiles. These percentiles correspond to the mean μ to which two standard deviations σ are subtracted or added.

Sample #	1			2			3		
Characteristic length	$\mu - 2\sigma$	μ	$\mu + 2\sigma$	$\mu - 2\sigma$	μ	$\mu + 2\sigma$	$\mu - 2\sigma$	μ	$\mu + 2\sigma$
Flow rate [mL/s]	*Re*								
0.6	2.2	6.8	11.4	4.6	8.5	12.5	2.5	9.0	15.5
0.9	3.3	10.2	17.1	6.9	12.8	18.7	3.8	13.5	23.3
1.2	4.4	13.6	22.8	9.2	17.1	25.0	5.0	18.1	31.1
1.5	5.5	17.0	28.5	11.7	21.3	31.2	6.3	22.6	38.8
1.8	6.7	20.4	34.2	13.8	25.6	37.4	7.5	27.1	46.6

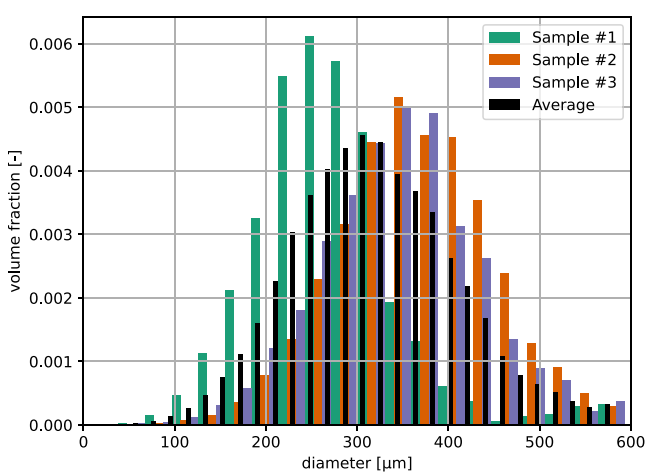

Fig. B.17. Volume-based pore size distribution of each sample.

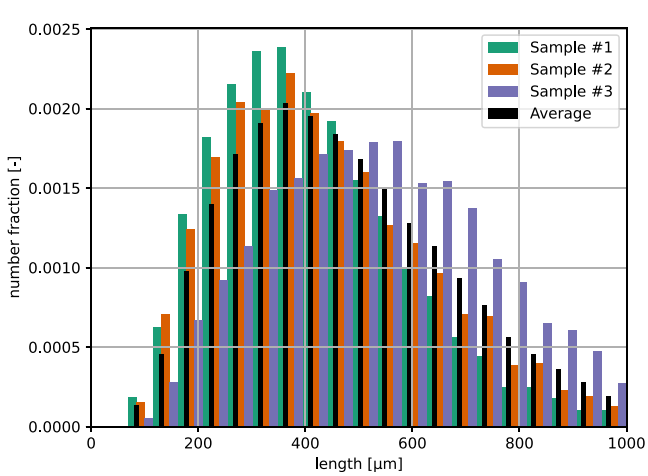

Fig. B.18. Number-based distance between pairs of connected pores, for each sample.

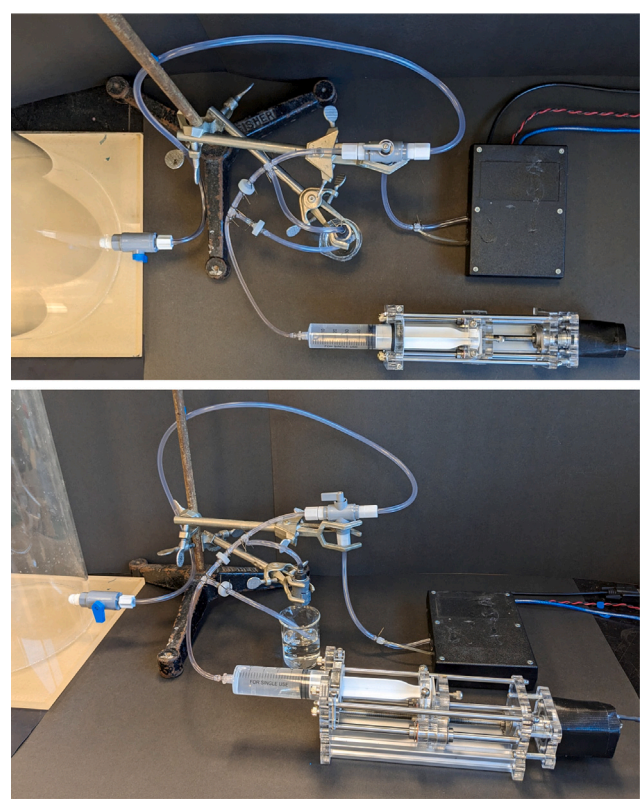

Fig. C.19. Photograph (top and side views) of the experimental setup. We can see the water column (left), the syringe pump (bottom), the controller (right) and the sample support (gray, center) above the beaker.

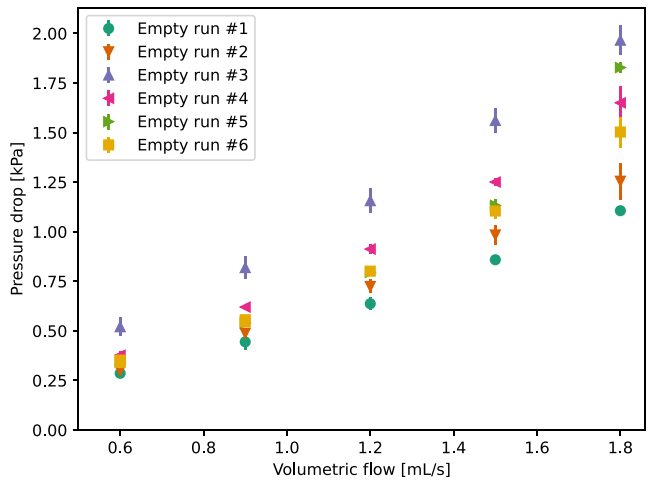

Fig. D.20. Pressure drop data for experiments where no sample is used. These pressure drops are to be subtracted from the experiments with samples to obtain the porous samples contributions. Each point is an average of three measures and confidence intervals are computed using a T-Student distribution, with a significance level $\alpha = 0.05$.

to time integration. Caution would be needed however, to ensure that fine structures of the media are properly captured by the RBF networks.

7. Conclusion

Circumventing transport phenomena limitations is at the core of process scale-up, and understanding and predicting sub-millimetric flow characteristics is a key component to improving the performance of porous media based equipment. In this work, we use pore-resolved CFD to accurately predict the flow through porous monoliths, providing a method for producing physically realistic digital twins of complex porous media.

RBF networks are generated from digitized monoliths and used in large scale simulations that cover complete centimeter-scale media with non-conformal cartesian grids, which alleviates the need for complex mesh generation. The method is verified using the method of manufactured solutions, and validated against pressure drop measurements obtained experimentally.

The proposed method is flexible in both usage and application fields, which makes it a powerful complementary tool for design as it provides understanding of the flow inside complex and possibly opaque media. Media topology can come from digitization or the result of computer-aided design (CAD). Coupling PRCFD to additive manufacturing could enable custom production of highly efficient components for heat exchangers, batteries and micro-porous reactors, paving the way for novel, efficient, and modular technologies.

CRediT authorship contribution statement

Olivier Guévremont: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Lucka Barbeau:** Writing – review & editing, Software, Methodology, Conceptualization. **Vaiana Moreau:** Writing – original draft, Methodology. **Federico Galli:** Writing – review & editing, Supervision, Methodology. **Nick Virgilio:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition. **Bruno Blais:** Writing – review & editing, Supervision, Software, Resources, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Olivier Guevremont reports financial support was provided by Natural Sciences and Engineering Research Council of Canada. Olivier Guevremont reports financial support was provided by Institut de l'Energie Trottier. Bruno Blais reports equipment, drugs, or supplies was provided by Digital Research Alliance of Canada. Federico Galli reports equipment, drugs, or supplies was provided by Digital Research Alliance of Canada. Bruno Blais reports financial support was provided by Natural Sciences and Engineering Research Council of Canada. Olivier Guevremont reports financial support was provided by Quebec Research Fund Nature and Technology. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to acknowledge the support of Lauriane Parès, for help in the laboratory, as well as Teodora Gancheva, for help in synthetizing the porous silicone samples and fitting them in the supports using flexible sleeves. Olivier Guévremont would like to acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC), the Fonds de recherche du Québec – Nature and technologies (FRQNT), Canada and Institut de l'Énergie Trottier (IET), Canada. Bruno Blais acknowledges financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) through the RGPIN-2020-04510 Discovery Grant and the funding from the Multiphysics Multiphase Intensification Automatization Workbench (MMIAOW) Canadian Research Chair Level 2 in computer-assisted design and scale-up of alternative energy vectors for sustainable chemical processes (CRC-2022-00340). The authors would also like to acknowledge the technical support and computing time provided by the Digital Research Alliance of Canada.

Appendix A. Synthesis of porous silicone monoliths

We synthetize silicone porous samples by (1) melt-processing of polystyrene (PS) and polylactic acide (PLA) blends, (2) selective extraction of PS, and (3) injection of silicone components in porous PLA molds, (4) crosslinking and (5) selective extraction of PLA. These steps are summarized in Fig. A.16.

A.1. Preparation of porous PLA molds

We prepare PLA porous molds as previously described by Esquirol et al. [80]. First, we dry PS and PLA granules in a vacuum oven at 60 °C during 12 h. We then melt-process a blend of PS and PLA (50/50 wt%) using an AG 34 mm Leistritz co-rotating twin-screw extruder at 190 °C. Following extrusion, we quench the rod-shaped blend in cold water to fix its morphology. We then anneal the blend under quiescent conditions with a hot press at 190 °C for 45 min, to let the morphology coarsen. After annealing, we quench the annealed polymer blend again in cold-water. We trim the annealed PS/PLA rods into cylinders (diameter = 0.8 cm, and thickness = 0.6 cm) by computer numerical control (CNC) machining. Next, we remove the PS phase by selective extraction in a Soxhlet extractor with cyclohexane for 1 week. After drying in a vacuum oven at 60 °C for 2 days, we verify that all PS is extracted by gravimetry. We store the resulting porous PLA molds at room temperature until the next step.

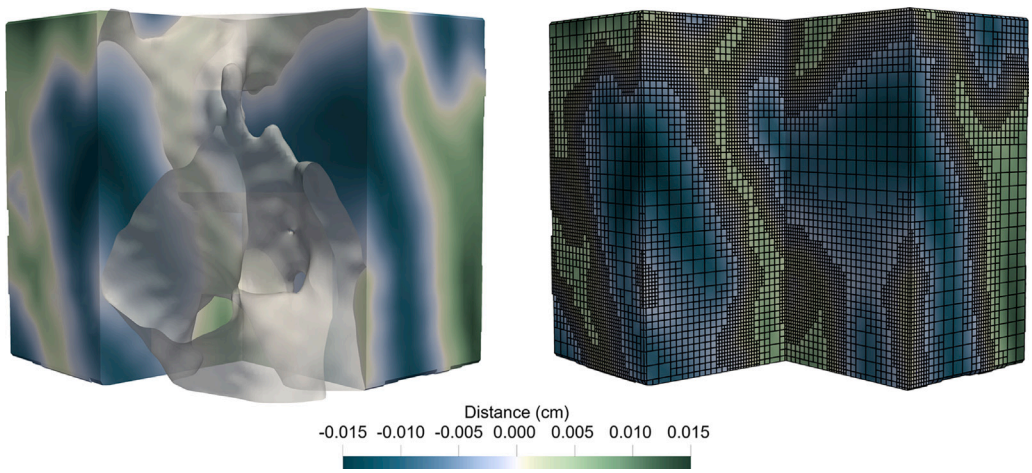

Fig. E.21. Subsection of a pore of sample #2, with visible adaptively refined grid.

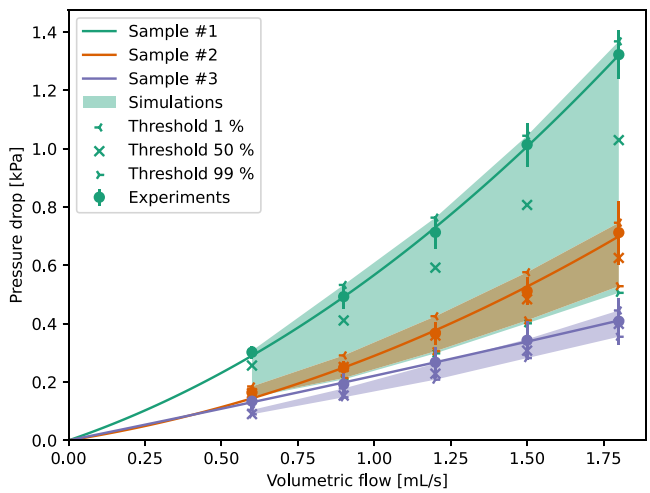

Fig. F.22. Experimental and simulated pressure drops of three silicone porous monoliths (average pore size: 300 μm). Darkened zones represent the values covered by thresholds between 1% and 99%. Experimental error bars are the T-Student's confidence interval using $\alpha = 0.05$ (n = 5 number of repetitions).

A.2. Preparation of porous silicone monoliths

We mix silicone precursor components (Shenzhen E4ulife Technology Co, product #X002NKUNXF, Let's Resin silicone rubber) in a beaker and then transfer to 15 mL falcon tubes. We deposit porous PLA molds individually into the silicone-filled tubes and submerge them to ensure that the porous molds are entirely filled with the solution of silicone precursors. We deposit the tubes in a custom-built injection system applying vacuum/pressure cycles until air stops being released from the molds. We retrieve the silicone-loaded molds and place them in a petri dish at room temperature to allow complete gelation. After 24 h, we remove the excess silicone around the sides of the molds using a cutting tool. We then selectively extract the PLA phase using chloroform (Thermo ScientificTM #42355, changed every two days) under agitation for 10 days. We rinse the resulting silicone monoliths with MilliQ water to remove the residual chloroform. We finally dry the silicone monoliths in a vacuum oven at 60 °C for 2 h, weight the samples to verify PLA extraction, and store in distilled water at room temperature until use.

Appendix B. Porous characteristics distributions

Fig. B.17 shows the pore size distribution of each monoliths (#1–3). The Gaussian shape confirms that describing the distributions in terms of mean and standard deviation is appropriate.

Fig. B.18 shows the distance between pairs of connected pores, for each monolith (#1–3). The distributions are log-normal. The difference between the distributions of samples #2 and #3 in Fig. B.18 might explain the difference of their observed pressure drops.

Appendix C. Experimental setup

Fig. C.19 shows two pictures of the experimental setup used to measure the pressure drop as the fluid passes through the monolith at a given flow rate.

Appendix D. Blank experimental data (without sample)

The data for the blank runs is shown in Fig. D.20: at least three blank runs were done after each manipulation of the experimental setup. Each of the six triplets is presented as a confidence interval using a T-Student distribution, with a significance level $\alpha = 0.05$. Each sequence of pressure measurements used for validation is associated to a blank run triplet which is subtracted from the associated measurements. Six blank runs are presented because experiments were executed in six sessions. The variations are due to the sensor itself and displacement of components of the experimental setup. To minimize the impacts of such variations, we calibrated the pressure sensor before each experiment session and ensured that component movement was minimal by securing the tubing to prevent changes in shape or positioning.

Appendix E. Adaptive refinement of the mesh

Fig. E.21 shows part of a pore of sample #2, with a subsection of its adaptively refined grid.

Appendix F. Validation with silicone porous media including results from all threshold values

Simulated pressure drops through the digitized silicone monoliths are presented in Fig. F.22. The results include CFD simulation based on surface meshes generated using different thresholds $\{1, 50, 99\}$%, as well as previously shown experimental data. Sample #1 is extremely sensitive to the threshold value, likely due to a combined effect of low mean pore size and standard deviation (see Table 1) and steadily low local porosity (see Fig. 13). The artificial occlusion or patency depending on the threshold parameter has stronger effects on this sample because of its smaller and more homogeneous pores.

Data availability

Data will be made available on request.

References

[1] A. Gavriilidis, P. Angeli, E. Cao, K.K. Yeong, Y.S. Wan, Technology and applications of microengineered reactors, Chem. Eng. Res. Des. 80 (2002) 3–30, http://dx.doi.org/10.1205/026387602753393196.

[2] J. Zhang, J. Chen, S. Peng, S. Peng, Z. Zhang, Y. Tong, P.W. Miller, X.-P. Yan, Emerging porous materials in confined spaces: from chromatographic applications to flow chemistry, Chem. Soc. Rev. 48 (2019) 2566–2595.

[3] M.A. Swartz, M.E. Fleury, Interstitial flow and its effects in soft tissues, Annu. Rev. Biomed. Eng. 9 (2007) 229–256.

[4] A.R. Khaled, K. Vafai, The role of porous media in modeling flow and heat transfer in biological tissues, Int. J. Heat Mass Transfer 46 (2003) 4989–5003, http://dx.doi.org/10.1016/S0017-9310(03)00301-6.

[5] I. Yadroitsev, I. Shishkovsky, P. Bertrand, I. Smurov, Manufacturing of fine-structured 3d porous filter elements by selective laser melting, Appl. Surf. Sci. 255 (2009) 5523–5527, http://dx.doi.org/10.1016/J.APSUSC.2008.07.154.

[6] M.A. Dippenaar, Porosity reviewed: quantitative multi-disciplinary understanding, recent advances and applications in vadose zone hydrology, Geotech. Geol. Eng. 32 (2014) 1–19.

[7] Z. Dong, Z. Wen, F. Zhao, S. Kuhn, T. Noël, Scale-up of micro- and milli-reactors: An overview of strategies, design principles and applications, Chem. Eng. Sci.: X 10 (2021) 100097, http://dx.doi.org/10.1016/J.CESX.2021.100097.

[8] A. Ricke, M. Sadeghi, W. Dreher, Magnetic resonance velocimetry for porous media: sources and reduction of measurement errors for improved accuracy, Exp. Fluids 64 (2023) 132.

[9] X. Lu, Y. Zhao, D.J.C. Dennis, Flow measurements in microporous media using micro-particle image velocimetry, Phys. Rev. Fluids 3 (2018) 104202.

[10] A. Golparvar, Y. Zhou, K. Wu, J. Ma, Z. Yu, A comprehensive review of pore scale modeling methodologies for multiphase flow in porous media, Adv. Geo-Energy Res. 2 (2018) 418–440, http://dx.doi.org/10.26804/ager.2018.04.07.

[11] B.D. Wood, X. He, S.V. Apte, Modeling turbulent flows in porous media, Annu. Rev. Fluid Mech. 52 (2020) 171–203.

[12] N. Saxena, A. Hows, R. Hofmann, F.O. Alpak, J. Freeman, S. Hunter, M. Appel, Imaging and computational considerations for image computed permeability: Operating envelope of digital rock physics, Adv. Water Resour. 116 (2018) 127–144, http://dx.doi.org/10.1016/J.ADVWATRES.2018.04.001.

[13] M.E. Curtis, Structural characterization of gas shales on the micro-and nano-scales, in: SPE Canada Unconventional Resources Conference? 2010, pp. SPE–137693.

[14] B.P. Flannery, H.W. Deckman, W.G. Roberge, K.L. D'Amico, Three-dimensional x-ray microtomography, Science 237 (1987) 1439–1444.

[15] L. Chen, A. He, J. Zhao, Q. Kang, Z.Y. Li, J. Carmeliet, N. Shikazono, W.Q. Tao, Pore-scale modeling of complex transport phenomena in porous media, Prog. Energy Combust. Sci. 88 (2022) 100968, http://dx.doi.org/10.1016/J.PECS.2021.100968.

[16] M. Rychlik, M. Morzyński, M. Nowak, W. Stankiewicz, T. Łodygowski, M.B. Ogurkowska, Acquisition and transformation of biomedical objects to cad systems, Strojnicky Cas. 3 (2004) 121–135.

[17] H. Liu, Q. Kang, C.R. Leonardi, S. Schmieschek, A. Narváez, B.D. Jones, J.R. Williams, A.J. Valocchi, J. Harting, Multiphase lattice boltzmann simulations for porous media applications: A review, Comput. Geosci. 20 (2016) 777–805.

[18] P. Sanematsu, Y. Shen, K. Thompson, T. Yu, Y. Wang, D.L. Chang, B. Alramahi, A. Takbiri-Borujeni, M. Tyagi, C. Willson, Image-based stokes flow modeling in bulk proppant packs and propped fractures under high loading stresses, J. Pet. Sci. Eng. 135 (2015) 391–402, http://dx.doi.org/10.1016/J.PETROL.2015.09.017.

[19] P. Ranut, E. Nobile, L. Mancini, High resolution microtomography-based cfd simulation of flow and heat transfer in aluminum metal foams, Appl. Therm. Eng. 69 (2014) 230–240, http://dx.doi.org/10.1016/J.APPLTHERMALENG.2013.11.056.

[20] M. Piller, G. Schena, M. Nolich, S. Favretto, F. Radaelli, E. Rossi, Analysis of hydraulic permeability in porous media: from high resolution x-ray tomography to direct numerical simulation, Transp. Porous Media 80 (2009) 57–78.

[21] M. Aboukhedr, N. Mitroglou, A. Georgoulas, M. Marengo, K. Vogiatzaki, Simulation of droplet spreading on micro-ct reconstructed 3d real porous media using the volume-of-fluid method, in: Ilass Europe. 28th European Conference on Liquid Atomization and Spray Systems, 2017, pp. 410–417.

[22] D. Emmel, J.D. Hofmann, T. Arlt, I. Manke, G.D. Wehinger, D. Schroder, Understanding the impact of compression on the active area of carbon felt electrodes for redox flow batteries, ACS Appl. Energy Mater. 3 (2020) 4384–4393.

[23] C. Sinn, G.R. Pesch, J. Thöming, L. Kiewidt, Coupled conjugate heat transfer and heat production in open-cell ceramic foams investigated using cfd, Int. J. Heat Mass Transfer 139 (2019) 600–612, http://dx.doi.org/10.1016/J.IJHEATMASSTRANSFER.2019.05.042.

[24] Y. Dong, O. Korup, J. Gerdts, B.R. Cuenya, R. Horn, Microtomography-based cfd modeling of a fixed-bed reactor with an open-cell foam monolith and experimental verification by reactor profile measurements, Chem. Eng. J. 353 (2018) 176–188, http://dx.doi.org/10.1016/J.CEJ.2018.07.075.

[25] P. Menegazzi, J. Trapy, A new approach to the modelling of engine cooling systems, Rev. Inst. Fr. Pétrole 52 (1997) 531–539.

[26] S.A. Roberts, H. Mendoza, V.E. Brunini, B.L. Trembacki, D.R. Noble, A.M. Grillet, Insights into lithium-ion battery degradation and safety mechanisms from mesoscale simulations using experimentally reconstructed mesostructures, J. Electrochem. Energy Convers. Storage 13 (2016) 31005.

[27] K. Kuhlmann, C. Sinn, J.M.U. Siebert, G. Wehinger, J. Thöming, G.R. Pesch, From μct data to cfd: an open-source workflow for engineering applications, Eng. Appl. Comput. Fluid Mech. 16 (2022) 1706–1723, http://dx.doi.org/10.1080/19942060.2022.2109758.

[28] S. Das, N.G. Deen, J.A. Kuipers, Multiscale modeling of fixed-bed reactors with porous (open-cell foam) non-spherical particles: Hydrodynamics, Chem. Eng. J. 334 (2018) 741–759, http://dx.doi.org/10.1016/J.CEJ.2017.10.047.

[29] S. Das, A. Panda, N.G. Deen, J.A. Kuipers, A sharp-interface immersed boundary method to simulate convective and conjugate heat transfer through highly complex periodic porous structures, Chem. Eng. Sci. 191 (2018) 1–18, http://dx.doi.org/10.1016/J.CES.2018.04.061.

[30] S. Das, N.G. Deen, J.A. Kuipers, Direct numerical simulation for flow and heat transfer through random open-cell solid foams: Development of an ibm based cfd model, Catal. Today 273 (2016) 140–150, http://dx.doi.org/10.1016/J.CATTOD.2016.03.048, sharp immersed boundary
From microCT
Temperature 2nd order Dirichlet and Neumann IB
Closure for pressure drop and heat transfer coefficients are used
Validation.

[31] V. Chandra, S. Das, E.A.J.F. Peters, J.A.M. Kuipers, Direct numerical simulation of hydrodynamic dispersion in open-cell solid foams, Chem. Eng. J. 358 (2019) 1305–1323, http://dx.doi.org/10.1016/j.cej.2018.10.017, URL https://www.sciencedirect.com/science/article/pii/S1385894718319600.

[32] C.S. Peskin, Flow patterns around heart valves: A numerical method, J. Comput. Phys. 10 (1972) 252–271, http://dx.doi.org/10.1016/0021-9991(72)90065-4.

[33] J. Lu, S. Das, E.A. Peters, J.A. Kuipers, Direct numerical simulation of fluid flow and mass transfer in dense fluid-particle systems with surface reactions, Chem. Eng. Sci. 176 (2018) 1–18, http://dx.doi.org/10.1016/J.CES.2017.10.018.

[34] J. Lu, E. Peters, J.A.M. Kuipers, Direct numerical simulation of fluid flow and dependently coupled heat and mass transfer in fluid-particle systems, Chem. Eng. Sci. 204 (2019) 203–219, http://dx.doi.org/10.1016/j.ces.2019.02.043, URL https://www.sciencedirect.com/science/article/pii/S0009250919302118.

[35] V. Chandra, E. Peters, J. Kuipers, Direct numerical simulation of a non-isothermal non-adiabatic packed bed reactor, Chem. Eng. J. 385 (2020) 123641, http://dx.doi.org/10.1016/j.cej.2019.123641, URL https://linkinghub.elsevier.com/retrieve/pii/S1385894719330566, 340 particles
Continuum based
No empirical closures
Validation.

[36] V. Chandra, D. Vogels, E.A. Peters, J.A. Kuipers, A multi-scale model for the fischer-tropsch synthesis in a wall-cooled packed bed reactor, Chem. Eng. J. 410 (2021) 128245, http://dx.doi.org/10.1016/J.CEJ.2020.128245.

[37] F. Ilinca, J.F. Hétu, A finite element immersed boundary method for fluid flow around moving objects, Comput. & Fluids 39 (2010) 1656–1671, http://dx.doi.org/10.1016/J.COMPFLUID.2010.06.002.

[38] F. Ilinca, J.-F. Hétu, A finite element immersed boundary method for fluid flow around rigid objects, Internat. J. Numer. Methods Fluids 65 (2011) 856–875, http://dx.doi.org/10.1002/fld.2222, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.2222.

[39] F. Ilinca, J.-F. Hétu, Solution of flow around complex-shaped surfaces by an immersed boundary-body conformal enrichment method, Internat. J. Numer. Methods Fluids 69 (2012) 824–841, http://dx.doi.org/10.1002/fld.2615, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.2615.

[40] M. Lesueur, H. Rattez, O. Colomés, μCt scans permeability computation with an unfitted boundary method to improve coarsening accuracy, Comput. Geosci. 166 (2022) 105118, http://dx.doi.org/10.1016/J.CAGEO.2022.105118.

[41] H. Liu, Y. Tian, H. Zong, Q. Ma, M.Y. Wang, L. Zhang, Fully parallel level set method for large-scale structural topology optimization, Comput. Struct. 221 (2019) 13–27, http://dx.doi.org/10.1016/J.COMPSTRUC.2019.05.010.

[42] T. Poggio, F. Girosi, Networks for approximation and learning, Proc. IEEE 78 (1990) 1481–1497, http://dx.doi.org/10.1109/5.58326.

[43] J. Park, I.W. Sandberg, Universal approximation using radial-basis-function networks, Neural Comput. 3 (1991) 246–257, http://dx.doi.org/10.1162/neco.1991.3.2.246.

[44] Y. Liao, S.C. Fang, H.L. Nuttle, Relaxed conditions for radial-basis function networks to be universal approximators, Neural Netw. 16 (2003) 1019–1028, http://dx.doi.org/10.1016/S0893-6080(02)00227-7.

[45] J.C. Carr, R.K. Beatson, J.B. Cherrie, T.J. Mitchell, W.R. Fright, B.C. McCallum, T.R. Evans, Reconstruction and representation of 3d objects with radial basis functions, in: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, 2001, pp. 67–76.

[46] M.E. Biancolini, Fast Radial Basis Functions for Engineering Applications, Springer, 2017.

[47] F. Toja-Silva, J. Favier, A. Pinelli, Radial basis function (rbf)-based interpolation and spreading for the immersed boundary method, Comput. & Fluids 105 (2014) 66–75, http://dx.doi.org/10.1016/J.COMPFLUID.2014.09.026.
[48] N. Thai-Quang, N. Mai-Duy, C.-D. Tran, T. Tran-Cong, A direct forcing immersed boundary method employed with compact integrated rbf approximations for heat transfer and fluid flow problems, CMES Comput. Model. Eng. Sci. 96 (2013) 49–90.
[49] J. jian Xin, T. qiu Li, F. long Shi, A radial basis function for reconstructing complex immersed boundaries in ghost cell method, J. Hydrodyn. 30 (2018) 890–897.
[50] V. Shankar, G.B. Wright, R.M. Kirby, A.L. Fogelson, Augmenting the immersed boundary method with radial basis functions (rbfs) for the modeling of platelets in hemodynamic flows, Internat. J. Numer. Methods Fluids 79 (2015) 536–557.
[51] G. Medici, L. Smeraglia, A. Torabi, C. Botter, Review of modeling approaches to groundwater flow in deformed carbonate aquifers, Groundwater 59 (2021) 334–351.
[52] L. Wang, Y. Li, G. Zhao, N. Chen, Y. Xu, Experimental investigation of flow characteristics in porous media at low reynolds numbers (re-> 0) under different constant hydraulic heads, Water 11 (2019) 2317.
[53] Y. Wang, M. Brannock, S. Cox, G. Leslie, Cfd simulations of membrane filtration zone in a submerged hollow fibre membrane bioreactor using a porous media approach, J. Membr. Sci. 363 (2010) 57–66, http://dx.doi.org/10.1016/J.MEMSCI.2010.07.008.
[54] A.A. Mehrizi, M. Farhadi, K. Sedighi, M.A. Delavar, Effect of fin position and porosity on heat transfer improvement in a plate porous media heat exchanger, J. Taiwan Inst. Chem. Eng. 44 (2013) 420–431, http://dx.doi.org/10.1016/J.JTICE.2012.12.018.
[55] F. Barreras, A. Lozano, L. Valiño, C. Marín, A. Pascau, Flow distribution in a bipolar plate of a proton exchange membrane fuel cell: experiments and numerical simulation studies, J. Power Sources 144 (2005) 54–66, http://dx.doi.org/10.1016/J.JPOWSOUR.2004.11.066.
[56] B. Blais, L. Barbeau, V. Bibeau, S. Gauvin, T.E. Geitani, S. Golshan, R. Kamble, G. Mirakhori, J. Chaouki, Lethe: An open-source parallel high-order adaptative cfd solver for incompressible flows, SoftwareX 12 (2020) 100579, http://dx.doi.org/10.1016/j.softx.2020.100579, URL https://www.sciencedirect.com/science/article/pii/S2352711020302922.
[57] W. Bangerth, R. Hartmann, G. Kanschat, Deal.ii—a general-purpose object-oriented finite element library, ACM Trans. Math. Software 33 (2007) 24–es, http://dx.doi.org/10.1145/1268776.1268779.
[58] C. Burstedde, L.C. Wilcox, O. Ghattas, p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees, SIAM J. Sci. Comput. 33 (2011) 1103–1133, http://dx.doi.org/10.1137/100791634.
[59] A. Hay, S. Etienne, D. Pelletier, A. Garon, Hp-adaptive time integration based on the bdf for viscous flows, J. Comput. Phys. 291 (2015) 151–176, http://dx.doi.org/10.1016/J.JCP.2015.03.022.
[60] L. Barbeau, S. Étienne, C. Béguin, B. Blais, Development of a high-order continuous galerkin sharp-interface immersed boundary method and its application to incompressible flow problems, Comput. & Fluids 239 (2022) 105415, http://dx.doi.org/10.1016/J.COMPFLUID.2022.105415.
[61] L. Barbeau, S. Golshan, J. Deng, S. Étienne, C. Béguin, B. Blais, High-order moving immersed boundary and its application to a resolved cfd-dem model, Comput. & Fluids 268 (2024) 106094, http://dx.doi.org/10.1016/J.COMPFLUID.2023.106094.
[62] L. Barbeau, S. Étienne, C. Béguin, B. Blais, Solid–fluid force modeling: Insights from comparing a reduced order model for a pair of particles with resolved cfd-dem, Int. J. Multiph. Flow 178 (2024) 104882, http://dx.doi.org/10.1016/J.IJMULTIPHASEFLOW.2024.104882.
[63] C.A. Daunais, L. Barbeau, B. Blais, An extensive study of shear thinning flow around a spherical particle for power-law and carreau fluids, J. Non-Newton. Fluid Mech. 311 (2023) 104951, http://dx.doi.org/10.1016/J.JNNFM.2022.104951.
[64] M.K. Stock, H.M. Garvin, L.K. Corron, C.N. Hulse, L.E. Cirillo, A.R. Klales, K.L. Colman, K.E. Stull, The importance of processing procedures and threshold values in ct scan segmentation of skeletal elements: An example using the immature os coxa, Forensic Sci. Int. 309 (2020) 110232, http://dx.doi.org/10.1016/J.FORSCIINT.2020.110232.
[65] T. Badriyah, N. Sakinah, I. Syarif, D.R. Syarif, Segmentation stroke objects based on ct scan image using thresholding method, in: 2019 First International Conference on Smart Technology & Urban Development, STUD, 2019, pp. 1–6, http://dx.doi.org/10.1109/STUD49732.2019.9018825.
[66] L. Friedli, D. Kloukos, G. Kanavakis, D. Halazonetis, N. Gkantidis, The effect of threshold level on bone segmentation of cranial base structures from ct and cbct images, Sci. Rep. 10 (2020) 7361.
[67] J. Gostick, M. Aghighi, J. Hinebaugh, T. Tranter, M.A. Hoeh, H. Day, B. Spellacy, M.H. Sharqawy, A. Bazylak, A. Burns, W. Lehnert, A. Putz, Openpnm: A pore network modeling package, Comput. Sci. Eng. 18 (2016) 60–74, http://dx.doi.org/10.1109/MCSE.2016.49.
[68] J.T. Gostick, Z.A. Khan, T.G. Tranter, M.D.R. Kok, M. Agnaou, M. Sadeghi, R. Jervis, Porespy: A python toolkit for quantitative analysis of porous media images, J. Open Source Softw. 4 (2019) 1296.
[69] J.T. Gostick, Versatile and efficient pore network extraction method using marker-based watershed segmentation, Phys. Rev. E 96 (2017) 23307.
[70] C.L.Y. Yeong, S. Torquato, Reconstructing random media, Phys. Rev. E 57 (1998) 495–506, http://dx.doi.org/10.1103/PhysRevE.57.495, URL https://link.aps.org/doi/10.1103/PhysRevE.57.495.
[71] C.L.Y. Yeong, S. Torquato, Reconstructing random media. ii. three-dimensional media from two-dimensional cuts, Phys. Rev. E 58 (1998) 224–233, http://dx.doi.org/10.1103/PhysRevE.58.224, URL https://link.aps.org/doi/10.1103/PhysRevE.58.224.
[72] O.E. Srl, Bitpit open source library for scientific high performance computing, 2024, URL https://optimad.github.io/bitpit/.
[73] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Adv. Comput. Math. 4 (1995) 389–396.
[74] W.L. Oberkampf, C.J. Roy, Verification and Validation in Scientific Computing, Cambridge University Press, 2010.
[75] A.T. Cate, C.H. Nieuwstad, J.J. Derksen, H.E.A. den Akker, Particle imaging velocimetry experiments and lattice-boltzmann simulations on a single sphere settling under gravity, Phys. Fluids 14 (2002) 4012–4025.
[76] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, J. Periaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363–426.
[77] P.J. Roache, Verification of codes and calculations, AIAA J. 36 (1998) 696–702.
[78] A. Dybbs, R.V. Edwards, A new look at porous media fluid mechanics—darcy to turbulent, in: Fundamentals of Transport Phenomena in Porous Media, 1984, pp. 199–256.
[79] L.P. Saavedra, P. Munch, B. Blais, A matrix-free stabilized solver for the incompressible navier-stokes equations, Available at SSRN 4981567.
[80] A.-L. Esquirol, P. Sarazin, N. Virgilio, Tunable porous hydrogels from cocontinuous polymer blends, Macromolecules 47 (2014) 3068–3075, http://dx.doi.org/10.1021/ma402603b.