


RESEARCH ARTICLE

Open Access



# Reduced order modeling of blood perfusion in parametric multipatch liver lobules

Ahsan Ali Siddiqui<sup>1,5\*</sup> , Etienne Jessen<sup>2</sup> , Stein K. F. Stoter<sup>3</sup> , David Néron<sup>4,5</sup>  and Dominik Schillinger<sup>2,5</sup> 

\*Correspondence:  
ahsan.ali.siddiqui@ibnm.uni-hannover.de

<sup>1</sup>Institute of Mechanics and Computational Mechanics, Leibniz University Hannover, Appelstrasse 9, Hannover 30167, Germany

<sup>2</sup>Institute for Mechanics, Technical University of Darmstadt,

Franzika-Braun-Straße 7, Darmstadt 64287, Germany

<sup>3</sup>Department of Mechanical Engineering, Eindhoven University of Technology, Gemini South 2.131 (building 15) De Zaale, Eindhoven 5612 AJ, Netherlands

<sup>4</sup>Laboratoire de Mécanique Paris-Saclay (LMPS), Université Paris-Saclay, CentraleSupélec, ENS Paris-Saclay, CNRS, 4 avenue des sciences, Gif-sur-Yvette 91190, France

<sup>5</sup>International Research Training Group 2657 (IRTG 2657), Leibniz University Hannover, Appelstrasse 11a, Hannover 30167, Germany

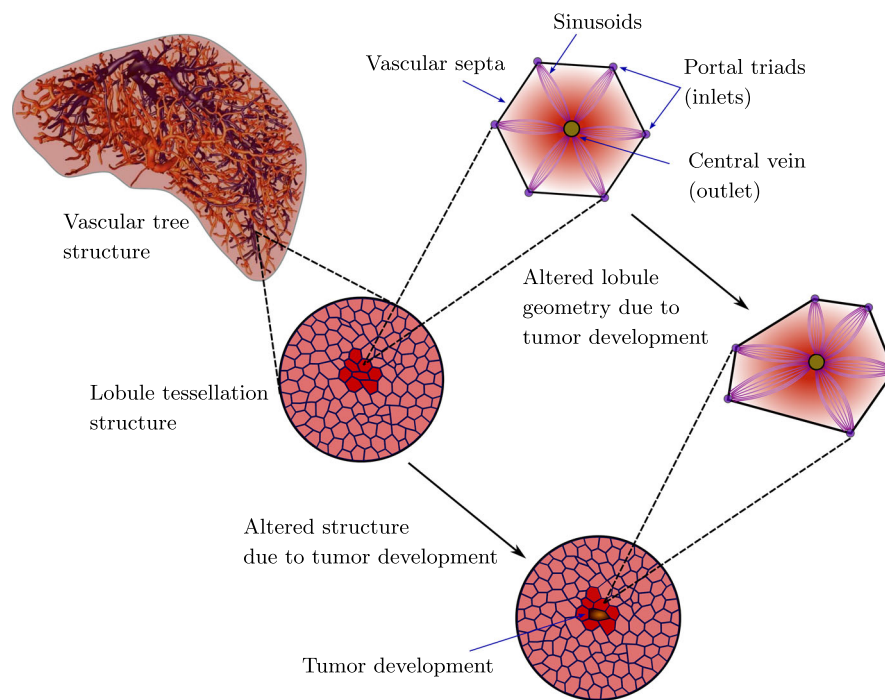
## Abstract

In this paper, we present a computationally efficient reduced order model for obtaining blood perfusion profiles within parametric functional units of the liver called ‘lobules’. We consider Darcy’s equation in two-dimensional hexagonal lobule domains with six flow inlets and one outlet, whose positions are parameterized to represent varying lobule geometries. To avoid the meshing effort for every new lobule domain, we map the parametric domain onto a single reference domain. By making use of the contra-variant Piola mapping, we represent solutions of the parametric domains in the reference domain. We then construct a reduced order model via proper orthogonal decomposition (POD). Additionally, we employ the discrete empirical interpolation method (DEIM) to treat the non-affine parameter dependence that appears due to the geometric mapping. For sampling random shapes and sizes of lobules, we generate Voronoi diagrams (VD) from Delaunay triangulations and use an energy minimization problem to control the packing of the lobule structures. To reduce the dimension of the parameterized problem, we exploit the mesh symmetry of the full lobule domain to split the full domain into six rotationally symmetric subdomains. We then use the same set of reduced order basis (ROB) functions within each subdomain for the construction of the reduced order model. We close our study by a thorough investigation of the accuracy and computational efficiency of the resulting reduced order model.

**Keywords:** Blood perfusion, Liver mechanics, Model order reduction, Proper orthogonal decomposition, Discrete empirical interpolation, Dimension reduction

## Introduction

Darcy’s equation is an elliptic partial differential equation that describes fluid flow through a porous medium. Among many other applications, it is used to model blood perfusion in biological tissues [1–4]. Although there exists robust numerical tools for solving these equations, they are still computationally too expensive when these equations are parameterized or stochastic in nature and have to be solved repeatedly, e.g., for assessment of a system’s reliability [5], system design optimization [6], or parameterized real-time simulations [7]. A promising solution to this limitation is offered by model order reduction



**Fig. 1** Liver structure from the organ scale to the lobule scale showing the vascular tree structure with bifurcations (branching of vessels of bigger diameter into smaller ones), the tessellation structure along with the description of a single lobule, and influence of an abnormality like tumor growth on the surrounding lobules in the tessellation structure

techniques [8–14], one of which is the projection based reduced basis method [12–14]. As described in [12–14], the goal is to build a reduced, computationally inexpensive model upon a high fidelity or computationally expensive model, while maintaining a ‘controlled’ accuracy of the high fidelity model.

In this paper, we consider such a computationally expensive system and demonstrate the use of the reduced basis method to achieve computationally efficient results for a range of model parameters. Our goal is to enable the efficient modeling of blood perfusion in the liver at the mesoscale level of functional units with varying geometries. Most existing work idealizes liver lobules as perfect two-dimensional hexagons [15,16], but there is a need to give up this idealization in order to be able to generate lobule structures that fill more complex domains [17]. The blood circulation in the liver is a complex process involving the entry of blood through two primary vessels: the hepatic artery and the portal vein, and discharge through one vessel: the hepatic vein. Within the liver, these vessels branch out into progressively smaller vessels forming an intricate vascular tree structure [18–21]. At the terminal end of this structure, functional units known as *lobules* of varying shapes and sizes exist in a tessellation structure regulating the blood flow through the liver parenchyma (functional tissue), as depicted in Fig. 1.

This hierarchical structure ensures the homogeneous distribution of blood throughout the liver, facilitating essential functions such as oxygen and nutrient delivery, toxin removal, and metabolic processes. However, chronic liver diseases such as cirrhosis, hepatocellular carcinoma, hepatitis, etc., alter the liver (and the lobule) structure permanently, and as a result, the blood flow through the lobules is affected. To elaborate further, in

case of liver carcinoma, such as hepatocellular carcinoma, the tumor deforms the liver parenchyma as it grows and exerts force on the neighboring area (surrounding lobules) [22]. As a result, the geometry of these lobules (in combination with their tissue permeability) changes, and subsequently, the blood perfusion through the lobules is altered. As depicted in Fig. 1, the lobules affected by the tumor development (highlighted by the dark red color) would allow the blood to perfuse differently in their original configuration (in a healthy state) and altered configuration (during the tumor development). This change in blood perfusion is closely linked to the delivery of oxygen and nutrients to the hepatocytes and other liver cells within lobules, responsible for healthy functioning of the liver. On the other hand, the blood perfused through these lobules, rich in nutrients and oxygen, could also be delivered to the tumor cells, and facilitate its growth.

Therefore, understanding blood perfusion at the lobule level becomes important for early diagnosis and improved clinical treatments. Some of the state-of-the-art techniques used in the clinical setting range from non-invasive to invasive methods. Non-invasive (or minimally invasive) methods, such as perfusion computed tomography (CT), involves acquiring dynamic high resolution scans which, after post-processing, reveal the perfusion maps with estimated blood volume and flow rates [23]. Another example is the indocyanine green (ICG) clearance test, which is often used as the perioperative assessment of perfusion characteristics of the liver. It measures the liver's ability to clear the ICG-dye from the blood stream, reflecting hepatic blood flow and liver functioning [24]. Invasive methods are preferred when the non-invasive ones are inconclusive or when histological studies have to be carried out. These include liver biopsy, laparoscopy, and hepatic venous pressure gradient (HPVG) measurements—to name a few. So we see on the one hand that various clinical techniques are available, but each one has its limitations. Some are cost-intensive, while others are limited either in terms of information acquisition or pose high risk factors (e.g. due to radiation exposure or surgical invasion).

Computation-based methods on the other hand provide a good alternative to understand the perfusion characteristics of liver across different length scales. Image-based modeling techniques have been applied in [20,25] to mimic the blood flow and approximate the tissue permeability in different directions in the three-dimensional lobules, however these studies were limited to sample-specific reconstruction of the lobule geometries. The geometrical features of these lobules could be patient-specific and could even evolve over time for the same patient due to liver abnormalities, such as the one mentioned earlier. To our knowledge, there is currently a gap in research concerning the modeling of blood perfusion in lobules for parameterized geometries.

In this work, we consider the two-dimensional representation of hexagonal lobules and focus on obtaining parametric velocity fields in a single liver lobule, whose geometry is fully parameterized with respect to a set of geometric parameters. We present a reduced order modeling framework for efficiently representing velocity fields in real-time within the parameterized lobule domains that is intended to serve as a proof of concept. The focus on developing a reduced order modeling strategy for a single lobule stems from our broader perspective of representing parametric velocity fields within the parameterized tessellation structures of lobules, where the velocity fields within each lobule could be represented by this reduced order model.

Moreover, our recent developments in an optimization-driven computational framework to generate synthetic vascular trees (that are in good agreement with the real vascular

tree structures of a human liver) [21,26] and modeling of liver tissue in connection with these synthetically generated vascular trees [27] can be seen as further foundations to a future coupled tree-lobule modeling and simulation framework that covers larger parts (or the complete) liver. The circulation of blood through these synthetically generated tree structures coupled with its perfusion through the lobule-tessellation structure would enable the modeling of the blood flow across different length scales. In the clinical context, such a future modeling and simulation framework could thus help in better understanding the correlated effects of the different perfusion mechanisms at different length scales.

For example, the liver resection surgery involves removing parts of the liver tissue affected by the tumor. However, the complex liver structure makes it difficult to predict the blood redistribution within the remaining liver tissue, and therefore, surgeon are always critical of the postoperative risks that could lead to liver failure. Another example is the early diagnosis and monitoring of disease progression (such as tumor growth), where the disease evolves over a range of stages, and typically the symptoms in the early stages are not apparent. Our future extended modeling and simulation framework could help in resolving the blood perfusion to the level of lobules, and therefore, could assist surgeons in more safely determining resection domains of the liver by minimizing the damage caused to the healthy liver tissue. In addition, such a framework could also support better diagnosis and treatments of liver diseases, based on monitoring the real-time perfusion changes at the level of lobules and adjusting the treatment plans accordingly.

## Problem setup

### Governing equations and weak form

Darcy's equation relates fluid velocities linearly to the pressure gradient under viscous effects of the fluid and the medium's permeability. It can be used for modeling blood flow in lobules by considering blood as an incompressible Newtonian fluid [16,28–32]. The set of equations are:

$$-\frac{\kappa}{\mu} \nabla p = \mathbf{u} \quad \text{in } \Omega, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (2)$$

and the set of boundary conditions are:

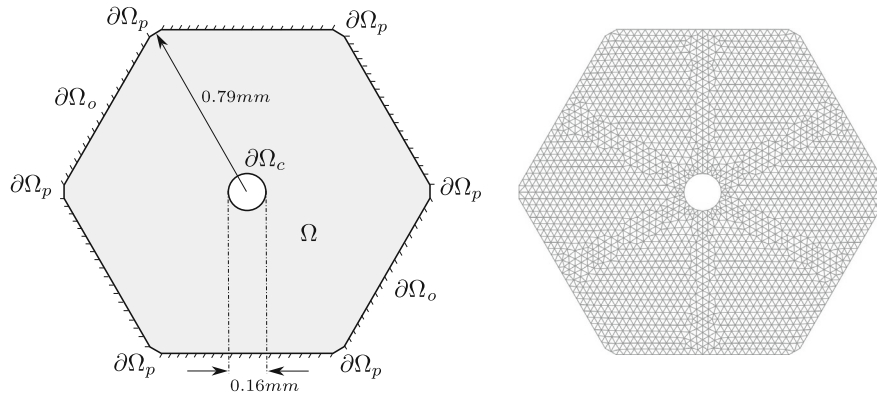
$$p = p_p \quad \text{on } \partial\Omega_p, \quad (3)$$

$$p = p_c \quad \text{on } \partial\Omega_c, \quad (4)$$

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad \text{on } \partial\Omega_o, \quad (5)$$

where  $\mathbf{u}$  is the flow velocity,  $p$  is the pressure,  $\kappa$  is the permeability tensor of the sinusoids and  $\mu$  is the blood viscosity. Eqs. (3) and (4) refer to the blood pressure at the lobule inlets (portal triads) and outlet (central vein), respectively. The last condition in Eq. (5) corresponds to the no-flow condition at the lobule walls or vascular septa. The lobule geometry with the locations of appropriate boundary conditions and the corresponding finite element mesh are illustrated in Fig. 2.

The weak form is obtained by multiplying Eqs. (1) and (2) by test functions  $\mathbf{v}$  and  $q$ , respectively, integrating over the domain, using integration by parts, and substituting the



**Fig. 2** Schematic drawing of the lobule with the boundaries defined (left) and the corresponding finite element mesh (right)

natural boundary conditions from Eqs. (3) and (4) in the formulation. The final weak form becomes:

Find  $\mathbf{u}, p \in \mathbf{H}(\text{div}, \Omega) \times L^2(\Omega)$  s.t.  $\forall \mathbf{v}, q \in \mathbf{H}(\text{div}, \Omega) \times L^2(\Omega)$  :

$$\begin{aligned} \int_{\Omega} \mu \underline{\kappa}^{-1} \mathbf{u} \cdot \mathbf{v} \, d\Omega - \int_{\Omega} p \nabla \cdot \mathbf{v} \, d\Omega + \int_{\Omega} q \nabla \cdot \mathbf{u} \, d\Omega \\ = - \int_{\partial\Omega_p} p_p \mathbf{n} \cdot \mathbf{v} \, dS - \int_{\partial\Omega_c} p_c \mathbf{n} \cdot \mathbf{v} \, dS. \end{aligned} \quad (6)$$

### Mapping of parametric domains to a reference domain

To obtain perfusion profiles for a range of lobules with varying geometry, one possible but naive approach would involve creating a new geometry and generating a new mesh for every new sample. This approach, however, would be inefficient and therefore is not suitable to a model order reduction framework we want to design in the next section. Instead, we create and mesh a *reference domain* ( $\hat{\Omega}$ ) once and utilize a *mapping function* ( $\mathcal{T}$ ) which maps every coordinate in the reference mesh to a new position in the parameterized *physical domain* ( $\Omega$ ). To this end, we utilize a vector-valued Laplace equation for representing  $\mathcal{T}$ , which can also deal with non-affine transformations, as is the case for our application. The mapping function takes the following strong form:

$$\Delta \mathcal{T}(\hat{X}; \theta) = 0 \quad \text{in } \hat{\Omega}, \quad \forall \theta \in \mathbb{P}, \quad (7)$$

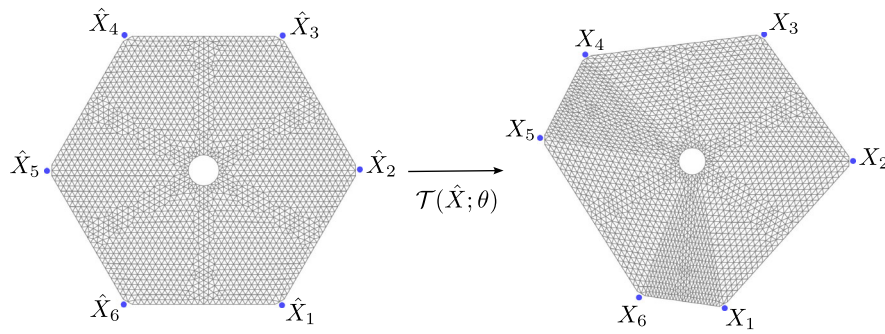
$$\mathcal{T} = \mathcal{T}^D(\theta) \quad \text{on } \partial\hat{\Omega}, \quad (8)$$

and the equivalent weak form becomes:

Find  $\mathcal{T} \in L^2(\hat{\Omega})$  s.t.  $\forall \mathcal{L} \in L^2(\hat{\Omega})$  :

$$\int_{\hat{\Omega}} \nabla \mathcal{T} \nabla \mathcal{L} \, d\hat{\Omega} = 0, \quad (9)$$

where  $\hat{X}$  are the coordinates in  $\hat{\Omega}$ ,  $\mathbb{P}$  represents the parameter space,  $\theta$  is a sample in  $\mathbb{P}$ , and  $\mathcal{T}^D(\theta)$  is a set of parametric control points, represented in blue color in Fig. 3, where every control point can move in  $x$  and  $y$  directions.



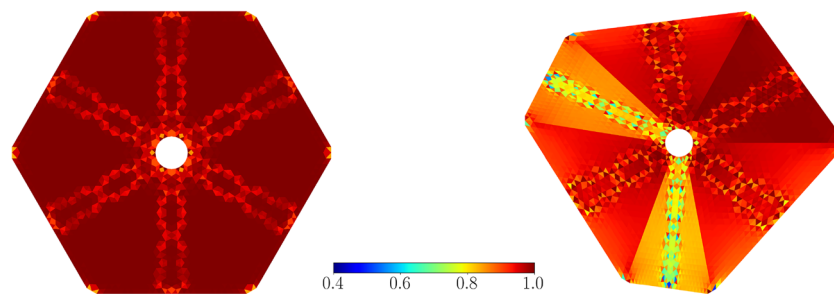
**Fig. 3** An illustration of mapping from  $\hat{\Omega}$  (left) to  $\Omega$  (right) for a random sample  $\theta$

In order to assess the element quality of the mesh obtained from  $\mathcal{T} : \hat{\Omega} \rightarrow \Omega$ , we employ the *radius-ratio* measure [33] given as:

$$\gamma = \frac{2 \times \text{inner radius}}{\text{circum radius}} = \frac{2r}{R}, \tag{10}$$

where  $0 < \gamma \leq 1$ . An element having  $\gamma$  close to 1 indicates a regular and equilateral triangle shaped element, whereas an element with  $\gamma$  close to 0 means a poor or degenerated element. This measure helps in keeping track of element quality in  $\Omega$  for every  $\theta$ .

For the geometric mapping shown in Fig. 3, the  $\gamma$  measure in  $\hat{\Omega}$  and  $\Omega$  is depicted in Fig. 4. For all elements in  $\hat{\Omega}$ , we observe that  $\gamma$  stays close to 1. However, for some elements within  $\Omega$ , this values decreases to 0.4. Moreover, these lower quality elements in  $\Omega$  are associated with regions where the adjacent inlets are positioned closer to each other in comparison with their original positions in  $\hat{\Omega}$ . Based on this, we restrict ourselves to mappings for which  $\gamma_{min} > 0.1$  and regard mappings where  $\gamma_{min} < 0.1$  as the change of topology of the lobule, e.g., from hexagons to pentagons. We further classify the resulting physical domains ( $\Omega$ ) into two categories: (1) mappings that lead to elements with  $\gamma_{min} \geq 0.5$  are considered as *regular* and (2) mappings that lead to elements with  $0.1 \leq \gamma_{min} < 0.5$  are regarded as *distorted*. The classification will help assess the influence of distortion due to geometric mapping in our analysis.



**Fig. 4** Visualization of  $\gamma$  measure in  $\hat{\Omega}$  (left) and  $\Omega$  (right). The elements in  $\hat{\Omega}$  are more regular since the  $\gamma$  measure is close to 1, whereas the elements in  $\hat{\Omega}$  are less regular as the  $\gamma$  measure in some elements gets as low as 0.4

### Contravariant Piola mapping and resulting parameterized model

Having  $\mathcal{T}$  at hand, we use the properties of the *contravariant Piola mapping* of the approximation spaces to map parametric solutions in  $\Omega$  to  $\hat{\Omega}$ . As illustrated in [34, 35], we can represent vectors in  $\mathbf{H}(\text{div}, \Omega)$  belonging to the *physical* coordinates in terms of the *reference* coordinates as:

$$\mathbf{u}(X) = \frac{1}{\det(\underline{\mathbf{F}})} \underline{\mathbf{F}} \hat{\mathbf{u}}(\hat{X}), \text{ where } \underline{\mathbf{F}} = \hat{\nabla} \mathcal{T}. \quad (11)$$

Similarly, operators on these vectors can also be mapped between the coordinate systems. Here we show the mapping for the divergence operator because it appears in our formulation. The divergence operator maps as follows:

$$\nabla \cdot \mathbf{u}(X) = \frac{1}{\det(\underline{\mathbf{F}})} \hat{\nabla} \cdot \hat{\mathbf{u}}(\hat{X}), \quad (12)$$

where  $\underline{\mathbf{F}}$  is the deformation gradient and  $\det(\underline{\mathbf{F}})$  is its determinant.

We take advantage of these mappings and construct the finite element spaces of the solution fields on  $\hat{\Omega}$  only once. We then plug Eqs. (11) and (12) into Eq. (6) to obtain any new parametric solutions. The new parameterized form of Eq. (6) becomes:

Find  $\hat{\mathbf{u}}(\theta), \hat{p}(\theta) \in \mathbf{H}(\text{div}, \hat{\Omega}) \times L^2(\hat{\Omega})$  s.t.  $\forall \hat{\mathbf{v}}, \hat{q} \in \mathbf{H}(\text{div}, \hat{\Omega}) \times L^2(\hat{\Omega})$  :

$$a(\hat{\mathbf{u}}, \hat{\mathbf{v}}; \theta) + b(\hat{p}, \hat{\mathbf{v}}) + b(\hat{q}, \hat{\mathbf{u}}) = L(\hat{\mathbf{v}}), \quad (13)$$

such that:

$$\begin{aligned} a(\hat{\mathbf{u}}, \hat{\mathbf{v}}; \theta) &= \int_{\hat{\Omega}} \underline{\kappa}_{\text{eff}}(\theta) \hat{\mathbf{u}} \cdot \hat{\mathbf{v}} \, d\hat{\Omega}, \\ b(\hat{q}, \hat{\mathbf{u}}) &= \int_{\hat{\Omega}} \hat{q} \hat{\nabla} \cdot \hat{\mathbf{u}} \, d\hat{\Omega}, \\ L(\hat{\mathbf{v}}) &= - \int_{\partial\hat{\Omega}_p} \hat{p}_p \hat{\mathbf{n}} \cdot \hat{\mathbf{v}} \, d\hat{S} - \int_{\partial\hat{\Omega}_c} \hat{p}_c \hat{\mathbf{n}} \cdot \hat{\mathbf{v}} \, d\hat{S}, \end{aligned}$$

where  $\underline{\kappa}_{\text{eff}}(\theta)$  appears as a new parameter dependent second order tensor field in the formulation. It is defined as:

$$\underline{\kappa}_{\text{eff}}(\theta) = \frac{\underline{\mathbf{F}}^T(\theta) \underline{\mu} \underline{\kappa}^{-1} \underline{\mathbf{F}}(\theta)}{\det(\underline{\mathbf{F}}(\theta))}. \quad (14)$$

The final parameterized model incorporates the geometric parameter dependence, captured in  $\underline{\mathbf{F}}(\theta)$ .

### Pressure and velocity pair for finite element discretization

For the discretization of the  $\mathbf{H}(\text{div}, \Omega)$  and  $L^2(\Omega)$  function spaces corresponding to solution spaces of the velocity and the pressure fields, we opted for a  $BDM_k$  and  $DG_{k-1}$  element pair, where  $k$  is the polynomial degree [36]. We select  $k = 1$  considering mesh resolutions to be fine enough to ensure converged solutions.  $BDM_1$  elements are often used for discretizing  $\mathbf{H}(\text{div}, \Omega)$  function spaces for a variety of fluid mechanics problems, where the degrees of freedom (DoFs) for these elements are the integrals of facet normals with the 1<sup>st</sup> order Lagrange space [37], whereas  $DG_0$  elements are piecewise constant elements

with only one DoF per element. To sum up, we seek the solution of velocity and pressure pair in subspaces  $\mathcal{W}^h \times \mathcal{X}^h$  such that:

$$\mathcal{W}^h \subset \mathbf{H}(\text{div}, \hat{\Omega})(\theta) \quad \forall \theta \in \mathbb{P} \quad (15)$$

$$\mathcal{X}^h \subset L^2(\hat{\Omega}). \quad (16)$$

This pair offers several advantages. Firstly, it guarantees the fulfillment of the *Ladyzhenskaya-Babuška-Brezzi* (LBB) condition, assuring well-posedness of the problem. Secondly, the subspaces  $\mathcal{W}^h \times \mathcal{X}^h$  constructed from the pair  $BDM_p$  and  $DG_{p-1}$  provide divergence-free velocities, as required in Eq. (2), and hence preserves the mass conservation within the system.

For constructing  $\mathcal{W}^h \times \mathcal{X}^h$  in  $\hat{\Omega}$ , obtaining the system of equations in matrix form, and solving these equations, we utilize the FEniCS finite element package [38]. Additionally, to tackle the high dimensional nature of the final problem, as will be discussed in the next section, we use the multiphenics package [39] to transform a full domain problem into equivalent subdomain problems.

## Sampling of geometric variation and parameter space reduction

### Sampling method

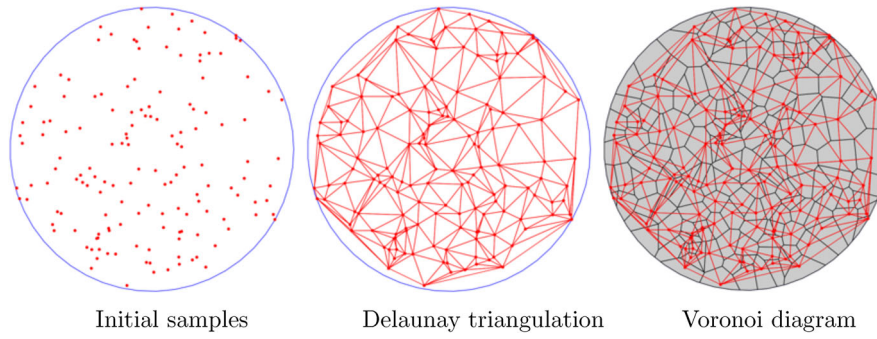
Typically, lobules are modeled as juxtaposed regular hexagonal prisms [15, 28]. While this simplification might be sufficient to analyze the flow characteristics of singular lobules, it fails to correctly capture the spatial patterns observed in vivo. An alternative approach is to generate lobules using Voronoi diagrams (VD). These diagrams partition the plane into distinct cells based on a given set of seed points. All points inside a cell are closer to their corresponding seed than to any other seed. Lau et al. [40] recently showed that the lobule structure inside a porcine liver can be described by Voronoi diagrams with nearly 90% accuracy. Based on these findings, we use Voronoi diagrams to generate lobules in silico. We describe the generation algorithm in two dimensions, but note that it can also be extended to three dimensions.

Let  $\mathcal{S}$  be a set of  $n$  points in 2D Euclidean space  $\mathbb{R}^2$  inside a pre-defined domain  $\Omega$ . A Voronoi diagram is defined as the set of Voronoi cells  $\mathcal{V} = \{\mathcal{V}_i\}_{i=1}^n$ , where each cell is the unique convex set

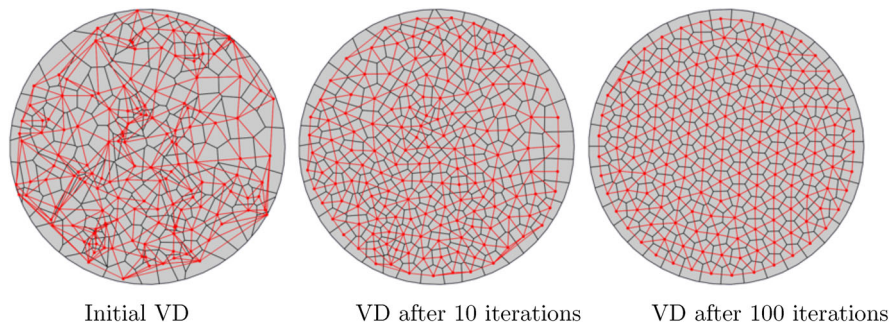
$$\mathcal{V}_i = \{x \in \mathbb{R}^2, \|x - x_i\| \leq \|x - x_j\|, \forall j \neq i\}, \quad (17)$$

where  $x_i$  is the (known) seed position of the  $i$ th Voronoi cell. There exist multiple methods to generate such a Voronoi diagram; either directly from the set of seed points, e.g. the Fortunes algorithm [41], or indirectly from a Delaunay triangulation using the Bowyer-Watson algorithm [42]. Here, we use the Bowyer-Watson algorithm, where the Delaunay triangulation is generated incrementally from the seed points, illustrated for an example in Fig. 5. With the Delaunay triangulation computed, we utilize its duality to Voronoi diagrams to create the latter. Finally, we clip the Voronoi cells against the boundary of the domain, as described in [43], and denote these boundary cells with  $\mathcal{V}_{|\Omega}$ .

Each Voronoi cell now constitutes a single lobule. However, as highlighted in Fig. 5 (right), the resulting lobules can exhibit degenerate features, such as steep angles, and do not resemble in-vivo structures. This stems from the fact that the position of the seeds



**Fig. 5** Duality between Delaunay triangulations (in red) and Voronoi diagrams (grey cells) for a set of points inside a circular domain  $\Omega$



**Fig. 6** Minimization of  $E(\mathcal{S})$  for the generation of a centroidal Voronoi diagram

solely decides the resulting pattern. To alleviate this problem, we extend our approach to allow the seed positions to change. The goal is to evenly space the positions inside our given domain, which can be achieved by considering the following energy problem for our set of seeds points  $\mathcal{S}$ :

$$E(\mathcal{S}) = \sum_{i=1}^n \int_{\mathcal{V}_i \cap \Omega} ||(x - x_i)||^2 dx. \tag{18}$$

The minimum energy is found when the position of each seed point coincides with the centroid of its respective Voronoi cell  $\mathcal{V}_i$ . We minimize Eq. (18) using the L-BFGS-algorithm [44]. The solution to this problem is called the centroidal Voronoi diagram (CVD) [45] and is shown in Fig. 6 for 100 seed points. Based on the domain considered, the seed points converge against a regular tessellation pattern, see Fig. 6 (right). We therefore terminate the algorithm before convergence, as soon as we achieve the qualitative patterns shown in [40], as depicted in Fig. 6 (center). The complete approach enables us to generate lobule patterns of any size from which we can efficiently sample realistic (two-dimensional) lobule shapes.

**From a full domain problem to multiple subdomain problems**

Reduced order modeling techniques have proven to be effective for problems with low dimensional parametric spaces  $\mathbb{P}$  [46–48], where the Kolmogorov  $N$ -width of the solution manifold –  $\mathcal{M} = \{(\hat{\mathbf{u}}(\theta), \hat{p}(\theta)) \mid \theta \in \mathbb{P}\}$  in our case – remains small. However, at this point,

the dimension of our parametrized problem in Eq. (13) is:

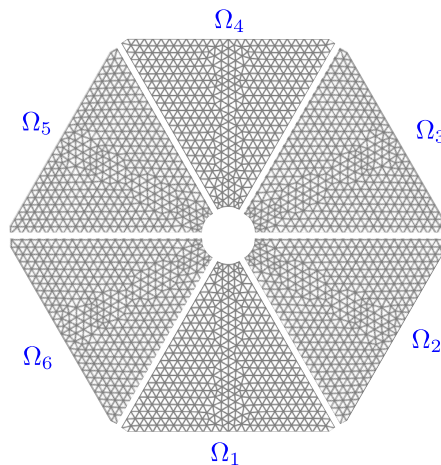
$$\dim(\mathbb{P}) = 2 \text{ (x- and y-coordinate)} \times 6 \text{ (#control points)} = 12, \quad (19)$$

which we would like to further reduce as much as possible to minimize the high computational cost associated with exploring the high dimensional parameter space and for an efficient construction of the reduced order model.

To this end, we split the full domain into 6 subdomains, where the subdomains are rotationally symmetric in the reference domain, as shown in Fig. 7. Each subdomain in the reference configuration is meshed via the same rotationally symmetric mesh, i.e., the reference domain consists of subdomain meshes, where each subdomain mesh is generated by rotation of the adjacent subdomain mesh by  $60^\circ$ . The flow solution in each subdomain is primarily governed by the location of its two inlets, which determine the geometry of the subdomain. In contrast, the influence of the location of the other four inlets that determine the geometry of the other subdomains on this flow solution is significantly smaller. Hence, we neglect this influence and assume that the parameter space can be decoupled into independent parameter directions. Hence, the dimension of our new parameterized problem posed on a subdomain is now reduced to:

$$\dim(\mathbb{P}) = 2 \text{ (x- and y-coordinate)} \times 2 \text{ (#control points)} = 4. \quad (20)$$

We note that we will not compute high-fidelity finite element solutions on a subdomain, as we do not know suitable boundary conditions at the interfaces to the neighboring subdomains. Instead, we will compute high-fidelity solutions on the complete domain, which – due to symmetry – provides us with six subdomain solutions for one sample realization  $\theta$ .



**Fig. 7** Split of full reference domain  $\hat{\Omega}$  into six identical reference subdomains (with identical meshes), where the geometry of each subdomain is now determined by 2 control points only

### Coupling at subdomain interfaces

To couple subdomains together, we use interior penalty type coupling terms [49, 50] at the subdomain interfaces, specified for mixed formulations in [51, 52]. These additional interface coupling terms enforce the continuity of the velocity fields across the subdomains, as required by the original problem. The updated discrete problem takes the following form:

Find  $\hat{\mathbf{u}}_h^i, \hat{p}_h^i \in \mathcal{W}^h \times \mathcal{X}^h$  s.t.  $\forall \hat{\mathbf{v}}_h^i, \hat{q}_h^i \in \mathcal{W}^h \times \mathcal{X}^h$  :

$$a(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i; \theta) + b(\hat{p}_h^i, \hat{\mathbf{v}}_h^i) + b(\hat{q}_h^i, \hat{\mathbf{u}}_h^i) = L(\hat{\mathbf{v}}_h^i), \quad (21)$$

where:

$$\begin{aligned} a(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i; \theta) &= \int_{\hat{\Omega}_i} \kappa_{\text{eff}}^i(\theta) \hat{\mathbf{u}}_h^i \cdot \hat{\mathbf{v}}_h^i d\hat{\Omega}_i + \int_{\partial\hat{\Gamma}_i} \frac{\eta_e}{h_e^i} [\hat{\mathbf{u}}_h^i] [\hat{\mathbf{v}}_h^i] d\hat{\Gamma}_i \\ b(\hat{q}_h^i, \hat{\mathbf{u}}_h^i) &= \int_{\hat{\Omega}_i} \hat{q}_h^i \hat{\nabla} \cdot \hat{\mathbf{u}}_h^i d\hat{\Omega}_i \\ L(\hat{\mathbf{v}}_h^i) &= - \int_{\partial\hat{\Omega}_{pi}} \hat{p}_p \hat{\mathbf{n}}^i \cdot \hat{\mathbf{v}}_h^i d\hat{S}_i - \int_{\partial\hat{\Omega}_{ci}} \hat{p}_c \hat{\mathbf{n}}^i \cdot \hat{\mathbf{v}}_h^i d\hat{S}_i, \end{aligned}$$

where  $i = 1, 2, \dots, 6$  specifies the respective subdomain according to Fig. 7. The classical *jump* operator  $[\cdot]$  is commonly used in discontinuous Galerkin (DG) formulations, see [49] for details. Since the DoFs of the velocity field in  $\mathcal{W}^h$  are the normal integrals, as mentioned in Section “Pressure and velocity pair for finite element discretization”, this operator also acts in the normal direction and penalizes the discontinuity in the velocity field at the subdomain interfaces. The term  $\eta_e$  is a penalty parameter and  $h_e$  is the element size at the subdomain interfaces.

### POD-DEIM based model order reduction

#### Proper orthogonal decomposition

Proper orthogonal decomposition (POD) is a model order reduction technique and is popular for problems with low dimensional parameter spaces. The idea of POD is to find a set of reduced order basis (ROB) functions that can optimally represent a set of high fidelity solutions, commonly known as the snapshot matrix  $\mathcal{M}_N \subset \mathcal{M}$ , which is the discrete representation of the solution manifold  $\mathcal{M} = \{\mathbf{w}(\theta) \mid \theta \in \mathbb{P}\}$ , where  $\mathbf{w}(\theta)$  – in our case – is the solution tuple of velocity and pressure, i.e.,  $\mathbf{w}(\theta) = (\hat{\mathbf{u}}_h(\theta), \hat{p}_h(\theta))$ . The method consists of two stages, a computationally expensive *offline* stage and a computationally inexpensive *online* stage.

During the computationally expensive *offline* stage,  $\mathcal{M}_N$  is constructed from the solutions of the high-fidelity finite element model such that the parametric dependence of the solutions for a range of parameters  $\theta_i$  for  $i = 1, 2, \dots, N$  in  $\mathbb{P}$  is captured. ROB functions are obtained by performing singular value decomposition (SVD) on  $\mathcal{M}_N$ , which results in  $\mathcal{M}_N = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$ , where  $\mathbf{U}$  corresponds to the left orthogonal matrix and contains sets of desired ROB functions,  $\mathbf{\Sigma}$  is the matrix of the singular values, and  $\mathbf{V}$  is the right orthogonal matrix. The approximation power of the first few ROB functions in  $\mathbf{U}$  (and  $\mathbf{V}$ ) can be illustrated through their corresponding singular values in  $\mathbf{\Sigma}$ . To this end, we turn to the Eckart-Young criterion [53] for the low rank  $n$  approximation of  $\mathcal{M}_N$ . The approximation error (or the energy decay in singular values) as a function of the number

of singular values is given by:

$$\epsilon(n) = \sqrt{\frac{\sum_{i=n}^N \sigma_i^2}{\sum_{i=1}^N \sigma_i^2}}, \quad (22)$$

where  $\sigma_i$  are the singular values,  $n = \dim(\mathbb{V}_{sol})$  represents the number of the first  $n$  dominant singular values in  $\Sigma$  (corresponding to first  $n$  orthonormal basis functions in  $\mathbf{U}$  and  $\mathbf{V}$ ),  $\mathbb{V}_{sol} = \text{span}\{\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_n\}$  is the low-dimensional space of the first  $n$  orthonormal functions (ROB functions) for approximating the solutions, and  $N$  is the total number of singular values in  $\Sigma$ . Chosen a suitable  $n$ , any parameter dependent high fidelity solution  $\mathbf{w}(\theta)$  can be approximated as  $\tilde{\mathbf{w}}(\theta)$  by means of a linear combination as:

$$\mathbf{w}(\theta) \approx \tilde{\mathbf{w}}(n, \theta) = \sum_{i=1}^n \xi_i(\theta) \mathbf{U}_i(\hat{X}) = \begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 & \dots & \mathbf{U}_n \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{bmatrix} \quad (23)$$

By way of Galerkin projection, the high dimensional stiffness matrix  $\mathbf{K}$  and force vector  $\mathbf{f}$  arising from the *bilinear* and the *linear* forms could be projected to the low dimensional space  $\mathbb{V}_{sol}$ , such that  $\underline{\mathbf{K}} \in \mathbb{R}^{n \times n}$  and  $\underline{\mathbf{f}} \in \mathbb{R}^n$ , respectively. The low dimensional system of Eq. (21) takes the following *non-affine* form:

$$\underline{\mathbf{K}}(\theta) \underline{\xi}(\theta) = \underline{\mathbf{f}}, \quad (24)$$

where  $\underline{\xi}(\theta)$  is the vector of coefficients in Eq. (23). The challenge with the *non-affine* representation, however, is that the parameter dependency in  $\theta$  cannot be taken out of  $\underline{\mathbf{K}}$ . In other words, for any new  $\theta_i$ ,  $\underline{\mathbf{K}}(\theta_i)$  has to be first updated and afterwards projected to  $\mathbb{V}_{sol}$  for obtaining  $\underline{\mathbf{K}}(\theta_i)$ . This would render the concept of model order reduction ineffective, and therefore, requires special treatment, as discussed in next section, to have the following *affine* representation:

$$\alpha(\theta) \underline{\mathbf{K}} \underline{\xi}(\theta) = \underline{\mathbf{f}}, \quad (25)$$

where we can finally separate parameter dependencies appearing as  $\alpha(\theta)$  from the reduced stiffness matrix  $\underline{\mathbf{K}}$ .

During the *online* stage, the reduced system in Eq. (25) is solved for  $\underline{\xi}(\theta)$ . This operation is computationally inexpensive, as it merely involves multiplying the new  $\alpha(\theta)$  with the pre-assembled  $\underline{\mathbf{K}}$ , and subsequently solving for the new  $\underline{\xi}(\theta)$ . It thus enables the efficient computation of parametric solutions.

#### Discrete empirical interpolation method

As discussed above, the final form of Eq. (21) does not permit *affine* decomposition. To elaborate, the first term of the *bilinear* form  $a(\hat{\mathbf{u}}_n^i, \hat{\mathbf{v}}_n^i; \theta)$  has the parameter dependency in the  $\underline{\kappa}_{eff}^i(\theta)$ , which cannot be taken out of the *bilinear* form. Therefore, it would need

to be integrated element wise on the underlying high fidelity mesh for any new parameter  $\theta_i$  in  $\mathbb{P}$ .

We use the discrete empirical interpolation method (DEIM) to resolve this issue, which has been successfully used to approximate non-linear and non-affine terms in the context of model order reduction [54–56]. The key idea is to approximate non-affine and non-linear functions by an affine decomposition, which for  $\underline{\kappa}_{eff}^i(\hat{X}, \theta)$  means:

$$\underline{\kappa}_{eff}^i(\hat{X}, \theta) \approx \sum_{j=1}^{n^{\kappa}} c_j^i(\theta) \mathbf{H}_j^i(\hat{X}). \quad (26)$$

Hence, we represent  $\underline{\kappa}_{eff}^i$  for every subdomain  $\hat{\Omega}_i$  as a linear combination of parameter dependent scalar coefficients  $c_j^i(\theta)$  and parameter independent spatial functions  $\mathbf{H}_j^i(\hat{X})$  up to the  $n^{\kappa}$  terms. The new affine representation of  $a(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i; \theta)$  then takes the following form:

$$\begin{aligned} a(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i; \theta) &= \sum_{j=1}^{n^{\kappa}} c_j^i(\theta) a_j(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i) + a_0(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i), \\ a(\hat{\mathbf{u}}_h^i, \hat{\mathbf{v}}_h^i; \theta) &= \sum_{j=1}^{n^{\kappa}} c_j^i(\theta) \int_{\hat{\Omega}_i} \mathbf{H}_j^i(\hat{X}) \hat{\mathbf{u}}_h^i \cdot \hat{\mathbf{v}}_h^i d\hat{\Omega}_i + \int_{\partial\hat{\Gamma}_i} \frac{\eta_e}{h_e^i} [\hat{\mathbf{u}}_h^i] [\hat{\mathbf{v}}_h^i] d\hat{\Gamma}_i. \end{aligned} \quad (27)$$

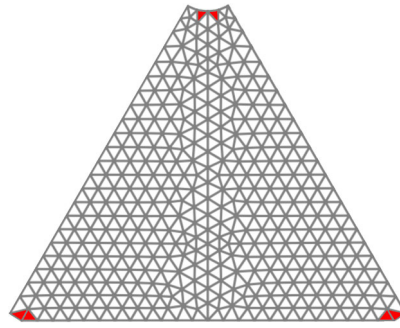
In order to obtain  $\mathbf{H}_j^i(\hat{X})$ , a similar approach to POD is adopted, where in this case  $\mathcal{M}_N$  is constructed from the discrete representation of  $\underline{\kappa}_{eff}^i(\hat{X}, \theta)$ . This is done by projecting  $\underline{\kappa}_{eff}^i(\hat{X}, \theta)$  for every new parameter onto a tensor function space of piecewise constant functions in  $\hat{\Omega}_i$  during the *offline* phase. The ROB functions  $\mathbf{H}_j^i(\hat{X})$  are then obtained by the SVD of  $\mathcal{M}_N$ . Moreover, for every ROB function, a so-called *magic point* corresponding to the spatial location  $\hat{X}^*$  is determined. We follow the procedure in [55] to calculate the parameter dependent coefficients and to determine the location of the *magic points*. The first magic point corresponds to the location  $\hat{X}_1^*$  (of a specific element of the mesh) at which  $\mathbf{H}_1^i$  has the maximum absolute value. The second magic point corresponds to the location  $\hat{X}_2^*$ , where the absolute error between  $\mathbf{H}_2^i$  and its interpolation using  $c_1^i \mathbf{H}_1^i$  is the largest. In general, the  $j$ -th magic point corresponds to the location  $\hat{X}_j^*$ , where  $\mathbf{H}_j^i$  and its interpolation via the previous  $j - 1$  ROB functions (and corresponding coefficients) yields the highest absolute error. The locations of the first few *magic points* in one of the subdomains is visualized in Fig. 8.

During the *online* phase, the parameter dependent coefficients  $c_j^i(\theta)$  are obtained by solving the following system of equations:

$$\mathcal{P}^T \nabla_{\underline{\kappa}}^i(\hat{X}) \mathbf{C}^i(\theta) = \mathcal{P}^T \underline{\kappa}_{eff}^i(\hat{X}, \theta), \quad (28)$$

where  $\nabla_{\underline{\kappa}}^i = \text{span} \{ \mathbf{H}_1^i, \mathbf{H}_2^i, \dots, \mathbf{H}_{n^{\kappa}}^i \} \in \mathbb{R}^{d \times n^{\kappa}}$  is the set of ROB functions,  $\mathbf{C}^i(\theta) = \{c_1^i, c_2^i, \dots, c_{n^{\kappa}}^i\}^T$  is the vector of coefficients,  $\mathcal{P} \in \mathbb{R}^{d \times n^{\kappa}}$  is the projection matrix,  $d$  is the number of DoFs for the discrete representation of  $\underline{\kappa}_{eff}^i(\hat{X}, \theta)$  (projection onto  $\hat{\Omega}_i$ ), and  $n^{\kappa} = \dim(\nabla_{\underline{\kappa}}^i)$  is the number of the first few dominant ROB functions in Eq. (26).

For every column in  $\mathcal{P}$ , all entries are zero except for one, where the value is one. The non-zero entry corresponds to the index of the *magic point*. Thus,  $\mathcal{P}$  is an operator which



**Fig. 8** Location of first few magic points in the  $i$ -th subdomain

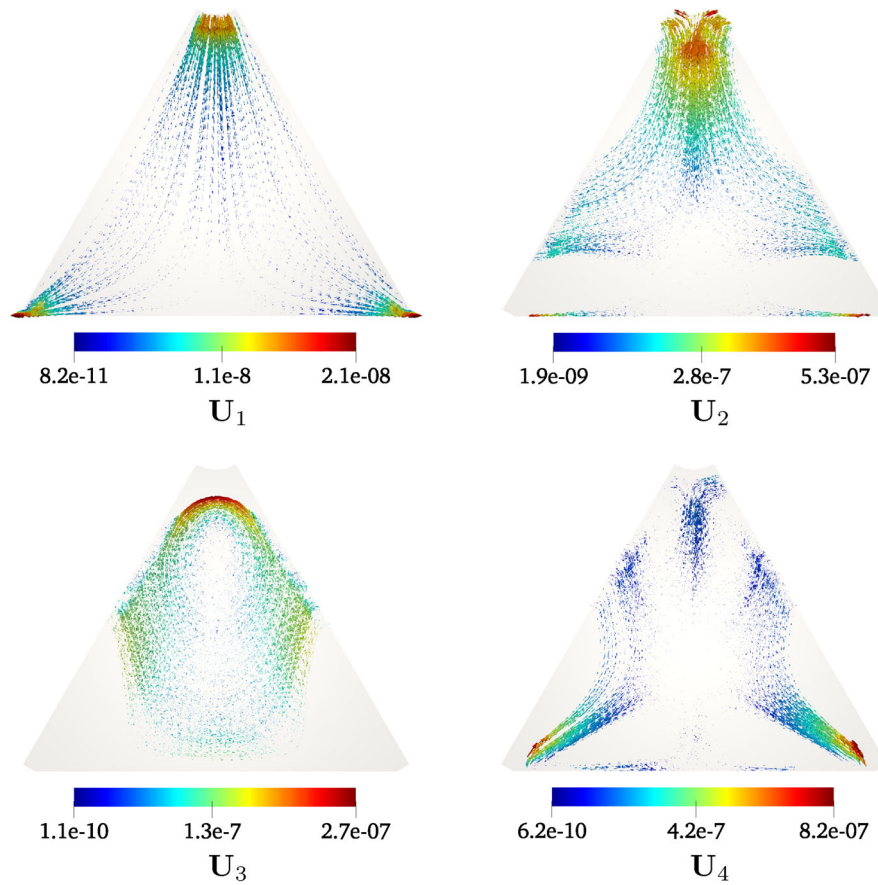
collapses the system of equations to merely solving at the *magic points*. Thus, the left hand side of Eq. (28), i.e.,  $\mathcal{P}^T \nabla_{\underline{\kappa}}^i(\hat{X}) \in \mathbb{R}^{n^{\kappa} \times n^{\kappa}}$  can be multiplied, inverted, and stored during the *offline* phase. During the *online* phase, only  $\underline{\kappa}_{eff}^i(\hat{X}^*, \theta)$  corresponding to the right hand side term  $\mathcal{P}^T \underline{\kappa}_{eff}^i(\hat{X}, \theta) \in \mathbb{R}^{n^{\kappa}}$  has to be computed for obtaining the  $\mathbf{C}^i(\theta)$  for any new  $\theta_i \in \mathbb{P}$ .

#### Implementation of the reduced order model of lobules

Finally, we utilize the POD and DEIM algorithms for our lobule problem in Eq. (21). To this end, we generate 1,000 samples according to what we discussed in Section “Sampling method”, which we use to train the reduced order model. We note that due to the symmetry of our model, the corresponding high-fidelity solutions provide snapshots for 6,000 subdomains.

In the offline phase, we construct  $\mathcal{M}_N$  for the solution tuple  $\mathbf{w}(\theta) = (\hat{\mathbf{u}}_h^i(\theta), \hat{p}_h^i(\theta))$  and the effective permeability tensor  $\underline{\kappa}_{eff}^i(\theta)$ . From a practical perspective, we fix  $\hat{\Omega}_1$  in Fig. 7 as the *reference subdomain* and rearrange the solution vectors of  $\hat{\Omega}_i$  for  $i = 2, 3, \dots, 6$  with respect to  $\hat{\Omega}_1$ . The rearrangement is done in such a way that the indices of DoFs corresponding to the locations  $(\hat{X}_j^i)$  for the elements in  $\hat{\Omega}_i$  are aligned with the indices of DoFs corresponding to the same locations  $(\hat{X}_j^1)$  (only rotated by  $(i - 1) \times 60^\circ$ ) in  $\hat{\Omega}_1$ . We emphasize that we generate sets of ROB functions for one subdomain ( $\hat{\Omega}_1$ ), but utilize the same sets of ROB functions for all other subdomains.

Moreover, although the reduced order model developed here is for parameterized hexagonal lobule domains, the construction of the low-dimensional sub-spaces on a triangular-subdomain enables us to represent the parametric velocity fields also in degenerated samples. To elaborate, as mentioned in Section “Mapping of parametric domains to a reference domain”, the mappings for which  $\gamma_{min} < 0.1$  can be considered as the change of topology. In this regard, a pentagonal lobule could be considered as a degenerated hexagonal lobule, with one of the subdomains completely collapsed (and hence removed from the mesh) and the other ones reorganized (to form a closed pentagon). For each of these triangular subdomains in the pentagonal lobule, the same set of reduced order basis functions can be used to obtain the reduced system of equations. Similarly, it is also possible to represent the velocity fields in heptagonal or octagonal lobule domains, since they can be represented as multiples of the triangular subdomain, for which we have constructed the low-dimensional representations.



**Fig. 9** First four ROB functions for velocity field  $\hat{\mathbf{u}}_h$  illustrated as vector fields in  $\hat{\Omega}_1$

For this subdomain, the first four dominant ROB functions for the velocity field and the effective permeability field are depicted in Figs. 9 and 10, respectively.

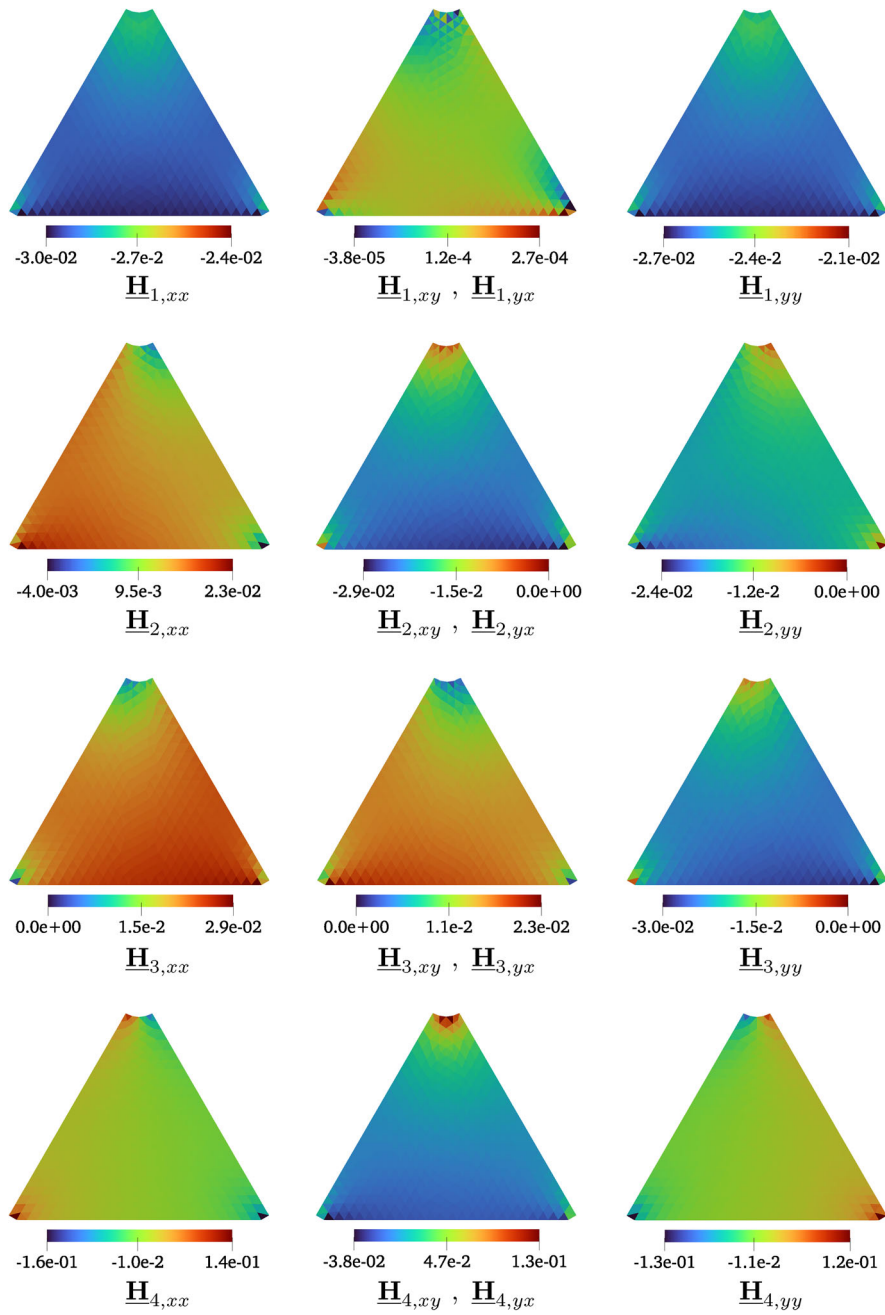
While the solution of the finite element model yields a tuple  $\mathbf{w}(\theta) = (\hat{\mathbf{u}}_h^i(\theta), \hat{p}_h^i(\theta))$ , the trial and test functions  $\hat{p}_h^i$  and  $\hat{q}_h^i$  appear in the *bilinear* forms  $b(\hat{p}_h^i, \hat{\mathbf{v}}_h^i)$  and  $b(\hat{q}_h^i, \hat{\mathbf{u}}_h^i)$  of Eq. (21) along with the divergence operator on the test and trial functions  $\hat{\mathbf{v}}_h^i$  and  $\hat{\mathbf{u}}_h^i$ , respectively. This allows the natural elimination of the pressure field from the formulation due to  $b(\hat{p}_h^i, \hat{\mathbf{v}}_h^i)$  and  $b(\hat{q}_h^i, \hat{\mathbf{u}}_h^i)$  being zero solely because of the divergence conforming nature of  $\hat{\mathbf{u}}^i$  and  $\hat{\mathbf{v}}^i$ , and irrespective of  $\hat{p}_h^i$  and  $\hat{q}_h^i$ . Therefore, we can completely avoid the pressure field for the construction of the reduced order model and focus only on approximating parametric velocity fields. The reduced order system takes the following form:

Find  $\hat{\mathbf{u}}^i \in \mathbb{V}_{sol}$  s.t.  $\forall \hat{\mathbf{v}}^i \in \mathbb{V}_{sol}$ :

$$\sum_{j=1}^{n^k} c_j^i(\theta) a_j(\hat{\mathbf{u}}^i, \hat{\mathbf{v}}^i) + a_0(\hat{\mathbf{u}}^i, \hat{\mathbf{v}}^i) = L(\hat{\mathbf{v}}^i). \quad (29)$$

The corresponding discrete matrix representation follows as:

$$\left( \sum_{j=1}^{n^k} c_j(\theta) \mathbf{K}_j + \mathbf{K}_0 \right) \boldsymbol{\xi}(\theta) = \mathbf{f}. \quad (30)$$



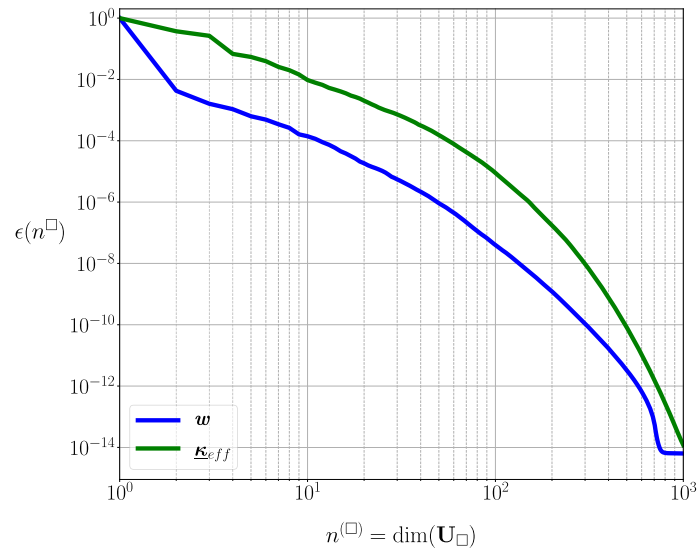
**Fig. 10** First four ROB functions for  $\kappa_{eff}$  with their tensor components  $xx, xy, yx, yy$ , visualized in  $\hat{\Omega}_1$

## Results and discussion

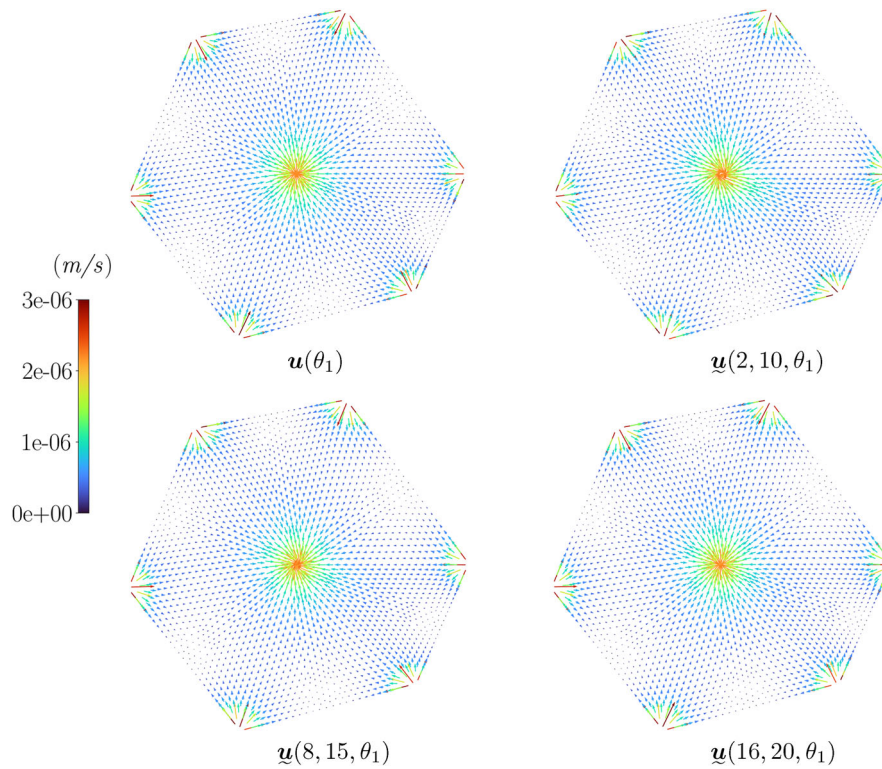
### Qualitative assessment

Based on Eq. (22), we plot the energy decay for 1,000 ROB functions in Fig. 11. We observe a rapid decay in energy for  $w$ , which for  $\kappa_{eff}$  is relatively slower. Nonetheless, with only 10 ROB functions (for both  $w$  and  $\kappa_{eff}$ ), we already find  $\epsilon \leq 10^{-2}$ .

We now compare the parametric velocity fields  $\mathbf{u}(\theta)$  obtained from the high fidelity model with the parametric velocity fields  $\underline{\mathbf{u}}(n^{sol}, n^\kappa, \theta)$  obtained from our reduced order model for one parameter combination  $\theta_1$ , randomly chosen out of 1,000 parameter com-



**Fig. 11** Energy decay of first 1000 ROB functions for solution tuple  $\mathbf{w}$  and effective permeability  $\kappa_{eff}$



**Fig. 12** Velocity profiles for a random parameter combination  $\theta_1$  obtained from the high fidelity finite element solution  $\mathbf{u}(\theta_1)$  and our reduced order model  $\mathbf{u}(n^{sol}, n^{\kappa}, \theta_1)$  for different combinations of  $n^{sol}$  and  $n^{\kappa}$

binations for training the reduced order model. We choose the tuple of ROB functions  $(n^{sol}, n^{\kappa}) = (2, 10)$ , since this tuple corresponds to an energy error of  $\epsilon \leq 10^{-2}$  in Fig. 11. We then increase the number of ROB functions in the tuple to observe the improvement in the velocity profiles.

For  $\theta_1$ , the velocity profiles  $\mathbf{u}(\theta_1)$  and  $\mathbf{u}(n^{sol}, n^k, \theta_1)$  are depicted in Fig. 12. We observe that for  $\mathbf{u}(2, 10, \theta_1)$ , there is already a very good agreement with the high-fidelity solution. However, a tuple  $(n^{sol}, n^k)$  can yield good reduced order approximation results for one parameter, but might prove to be insufficient and yield poor approximation for another. Therefore, we will establish a more rigorous relationship between the quality of the reduced order velocity fields and a sufficiently large tuple of ROB functions  $(n^{sol}, n^k)$  for all  $\theta_i$  in  $\mathbb{P}$  in the following.

### Quantitative assessment

We therefore turn to a more rigorous quantitative measure in terms of the norm of the velocity fields. This measure relates the high fidelity and reduced order velocity fields within the full lobule domains and takes into account both the approximation error (from the reduced order model) and the error at the subdomain interfaces (due to the domain split). We relate the tuple  $(n^{sol}, n^k)$  with the relative  $L^2$  errors for all the parameter points in  $\mathbb{P}$ , and define a multi-dimensional relative  $L^2$  error measure as a function of  $n^{sol}$ ,  $n^k$ , and  $\theta$  as follows:

$$e(n^{sol}, n^k, \theta) = \left( \frac{\| \hat{\mathbf{u}}(\theta_i) - \hat{\mathbf{u}}(n^{sol}, n^k, \theta_i) \|_{L^2(\Omega)}}{\| \hat{\mathbf{u}}(\theta_i) \|_{L^2(\Omega)}} \right) \quad \forall \theta_i \in \mathbb{P}, \quad (31)$$

$$e_{max} = \max(e). \quad (32)$$

Based on this measure, we search for the minimum  $n^{sol}$  and  $n^k$  such that  $e_{max}$  stays under a certain threshold. To guarantee efficiency of our reduced order model, we impose an upper limit of 30 for both  $n^{sol}$  and  $n^k$  per subdomain. We further utilize the classification of samples into samples with *regular* mapping and samples with *distorted* mapping as defined in Section “Mapping of parametric domains to a reference domain” to investigate the influence of the mapping on the approximation quality of our reduced order model.

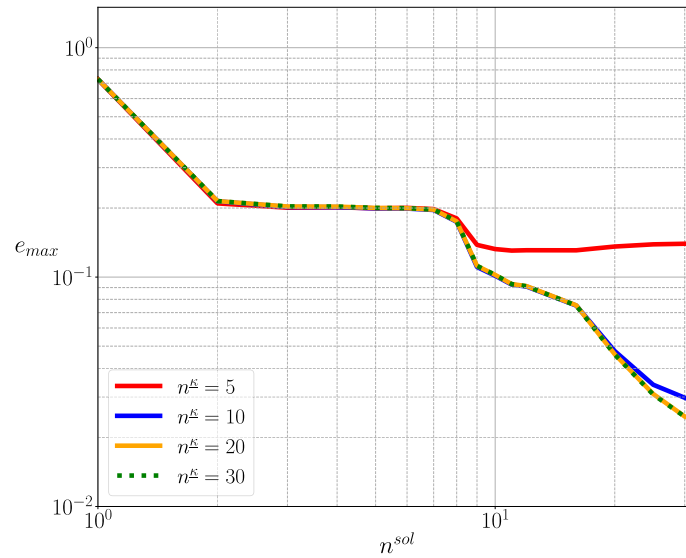
### Samples with regular mapping

Out of 1,000 samples used for training during the offline phase, 314 belonged to this category. For these samples, the maximum error  $e_{max}$  from Eq. (32) is plotted in Fig. 13.

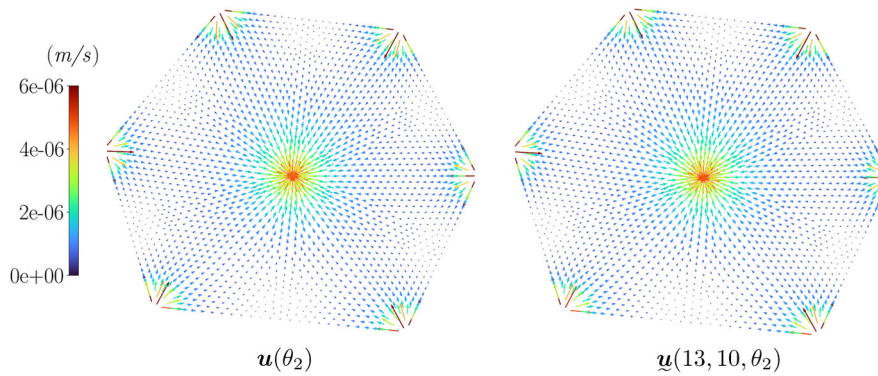
Based on this error plot, we accept an  $e_{max}$  of  $\sim 10\%$  and accordingly select the tuple  $(n^{sol}, n^k) = (13, 10)$  for our reduced order model.

To validate the performance of our reduced order model using this tuple, we generate new samples with the algorithm of Section “Sampling method” and apply new pressures  $p_p$  and  $p_c$  at the portal triads and central vein, respectively. This is done to make sure that the parameter combinations we use for the validation of our reduced order model for this category of samples were not in  $\mathbb{P}$  for the training purpose during the offline stage. The high fidelity solution  $\mathbf{u}(\theta_2)$  and the reduced order solution  $\mathbf{u}(13, 10, \theta_2)$  for a random parameter combination  $\theta_2$  are depicted in Fig. 14.

We observe that the velocity profile obtained from our reduced order model looks identical to that obtained from the high fidelity finite element solution. In terms of the quantitative norm, i.e., the  $e$  measure, the relative  $L^2$  error is found to be 3.6%.



**Fig. 13** Convergence with respect to  $e_{max}$  for samples with regular mappings ( $\gamma_{min} \geq 0.5$ )



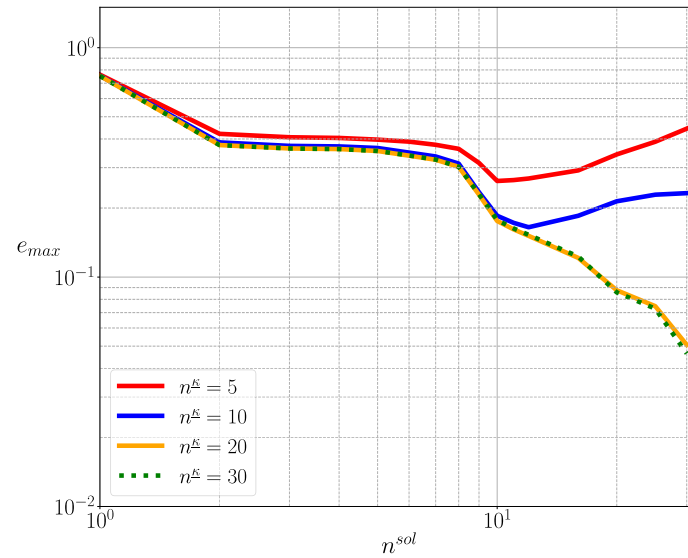
**Fig. 14** Visual comparison of high fidelity velocity field  $\mathbf{u}(\theta_2)$  and reduced order velocity field  $\mathbf{u}(13, 10, \theta_2)$  for a new parameter  $\theta_2$

### **Samples with distorted mapping**

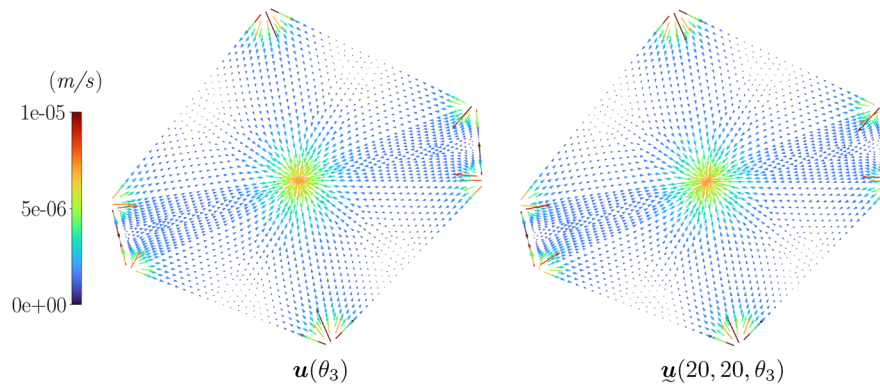
From our initial 1,000 samples in the offline stage computations, a total of 686 samples belong to this category. The convergence of  $e_{max}$  for this category can be seen in Fig. 15 and shows similar trends, but with some exceptions. In comparison to Fig. 13, we see that there is an overall offset of  $\sim 10\%$  between the curves for  $n^{sol} = 1, 2, \dots, 8$ . This observation suggests a higher requirement on  $n^{sol}$  to sufficiently enrich  $\mathbb{V}_{sol}$  for attaining the same level of accuracy. Moreover, we find  $n^k = 10$  for this category to be insufficient for accurately representing  $\mathbb{V}_k$ . Instead, we determine  $n^k = 20$  as an adequate representation of  $\mathbb{V}_k$  since the convergence pattern is identical for  $n^k = 20$  and 30.

Based on  $e_{max}$  at  $\sim 10\%$ , we select the tuple  $(n^{sol}, n^k) = (20, 20)$  as an appropriate compromise between computational efficiency and accuracy for our reduced order model for this category of samples. Once again, we test the reduced order model for new parameter combinations.

For the parameter  $\theta_3$ , the two solutions  $\mathbf{u}(\theta_3)$  and  $\mathbf{u}(20, 20, \theta_3)$  are plotted in Fig. 16. We observe that the velocity profile obtained from the reduced order model again is practically



**Fig. 15** Convergence with respect to  $e_{max}$  for samples with distorted mappings ( $0.5 > \gamma_{min} \geq 0.1$ )

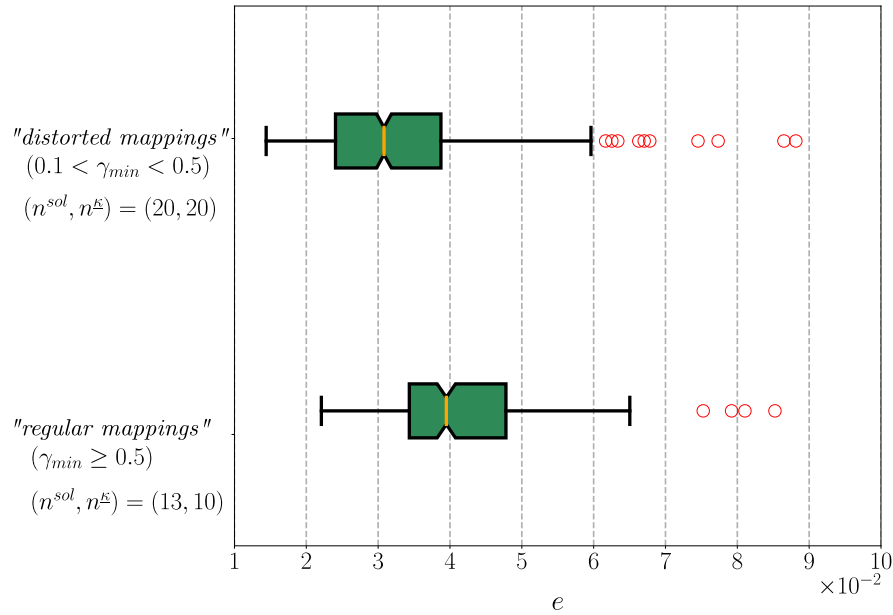


**Fig. 16** Visual comparison of high fidelity velocity field  $\mathbf{u}(\theta_3)$  and reduced order velocity field  $\mathbf{u}(20, 20, \theta_3)$  for a new parameter  $\theta_3$

indistinguishable to that obtained from the high fidelity finite element solution. The  $e$  measure between the two solutions is 5.4%.

The  $e$  measure for all the new parameter combinations we tested for when we validated the reduced order model against the high fidelity solutions can be seen in Fig. 17. For samples with *regular* mappings, the  $e$  median stands at  $\sim 4\%$ , while the first quartile ( $Q_1$ ) and the third quartile ( $Q_3$ ) are at  $\sim 3.4\%$  and  $\sim 4.8\%$ , respectively. For samples with *distorted* mappings, the  $e$  median is observed at  $\sim 3.1\%$ , while  $Q_1$  and  $Q_3$  are at  $\sim 2.4\%$  and  $\sim 3.9\%$ , respectively. Overall, the  $e$  measure stays under 10% for the worst case for either categories, as represented by the red markers in Fig. 17.

These observations confirm that our selection criterion ( $e \leq 10\%$ ) for the appropriate tuple of ROB functions ( $n^{sol}, n^k$ ) is indeed met in practical computations. It also indicates that the training with 1,000 parameter points is sufficient for exploring  $\mathbb{P}$  during the offline stage, and that our reduced order model is sufficiently trained and capable of approximating the high fidelity solutions for a variety of geometry parameters.



**Fig. 17** Validation of reduced order model for 1,000 new parameter points

In terms of the speed up, the cost of computing 1,000 high fidelity solutions during the offline stage took  $\sim 200$  minutes. Computing the same number of reduced order solutions required only  $\sim 15$  minutes, including the computation of the postprocessing of the reduced order velocity fields  $\hat{\mathbf{u}}(\theta)$  and computing the error  $e$  (with the high fidelity finite element solutions  $\hat{\mathbf{u}}(\theta)$  already known). Hence, we gain a speed up of at least 13.

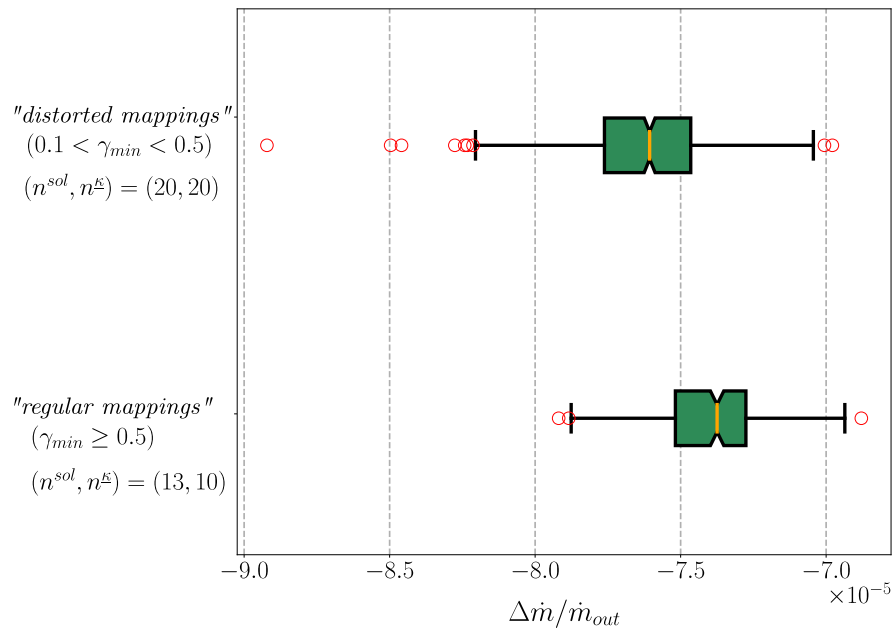
#### Divergence free reduced order velocity fields

Since we obtained the ROB functions on the subdomains and constructed the low-dimensional space  $\mathbb{V}_{sol}$  from the solutions in  $\mathcal{W}^h$ , the solutions  $\hat{\mathbf{u}}^i \in \mathbb{V}_{sol}$  are divergence free for  $\hat{\Omega}_i$  for  $i = 1, 2, \dots, 6$ . This statement, however, does not guarantee the full reduced order velocity field, i.e.,  $\hat{\mathbf{u}}$  in  $\hat{\Omega}$  to be mass conservative because of the weak coupling terms applied in glueing neighboring subdomains together. Nonetheless, the coupling terms at the subdomain interfaces of the finite element model enforced the velocities to be continuous and were projected onto the low-dimensional spaces during the construction of the reduced order model. Therefore, we expect  $\hat{\mathbf{u}}$  in  $\hat{\Omega}$  to be nearly divergence free and practically mass conservative.

To verify our hypothesis, we calculate the relative mass flow rate through  $\hat{\Omega}$  as the difference between rates of mass inflow and outflow normalized by the rate of mass outflow. The expression has the following form:

$$\Delta \dot{m} / \dot{m}^{out} = \frac{\dot{m}^{in} - \dot{m}^{out}}{\dot{m}^{out}}, \quad (33)$$

$$\Delta \dot{m} / \dot{m}^{out} = \frac{\rho \left( \hat{\mathbf{u}}^{in} \cdot \hat{\mathbf{A}}^{in} - \hat{\mathbf{u}}^{out} \cdot \hat{\mathbf{A}}^{out} \right)}{\rho \left( \hat{\mathbf{u}}^{out} \cdot \hat{\mathbf{A}}^{out} \right)}. \quad (34)$$



**Fig. 18** Relative mass flow rate for 1000 new parameter points

In the discretized setting, it takes the following form:

$$\Delta \dot{m} / \dot{m}_{out} = \frac{\left( \sum_{i=1}^6 \int_{\partial \Omega_p} \hat{\mathbf{u}} \cdot \hat{\mathbf{n}} dS_i - \int_{\partial \Omega_c} \hat{\mathbf{u}} \cdot \hat{\mathbf{n}} dS \right)}{\left( \int_{\partial \Omega_c} \hat{\mathbf{u}} \cdot \hat{\mathbf{n}} dS \right)}, \quad (35)$$

where  $\rho$  is the density of blood and  $\hat{\mathbf{n}}$  corresponds to the normal vectors at the central vein (outlet) and 6 portal triads (inlet) boundaries.

The corresponding  $\Delta \dot{m} / \dot{m}_{out}$  measure for the reduced order solutions, whose quality was summarized in Fig. 17, is now plotted in Fig. 18. We find  $|\Delta \dot{m} / \dot{m}_{out}|$  to be very small and under 0.01% for all reduced order solutions, pointing to practically divergence free reduced order parametric velocity fields  $\hat{\mathbf{u}}(\theta)$  in  $\hat{\Omega}$ , and thereby ensuring mass conservation within  $\hat{\Omega}$ . Moreover, although we selected different  $n^{sol}$  for validating the reduced order model, i.e.,  $n^{sol} = 13$  for samples with undistorted mappings and  $n^{sol} = 20$  for samples with distorted mappings, we do not see a significant difference in  $\Delta \dot{m} / \dot{m}_{out}$  for different  $n^{sol}$ . This observation suggests that the enrichment of  $\mathbb{V}_{sol}$  contributes to a better approximation quality of our reduced order model, i.e., smaller errors  $e$ , but has little or no influence on the mass conservation within the system.

## Conclusion and outlook

In this work, we addressed the computationally efficient reduced order computation of blood perfusion in a two-dimensional parametrized lobule model of variable geometry. For modeling the blood perfusion within a liver lobule, we utilized Darcy's equation by considering blood an incompressible fluid and the lobule tissue a porous medium. To account for variable lobule geometries, we utilized a mapping function to map the lobule geometry from a symmetric hexagonal reference lobule to a parametric domain. Leveraging the

properties of the contravariant Piola mapping, we plugged this geometric mapping function into our formulation for representing parametric solutions in the reference domain, and thereby saved the cost of constructing and meshing new geometries.

For the real-time computation of the parametric velocity fields, we used an approximation via a reduced order model based on proper orthogonal decomposition (POD). Additionally, we employed the discrete empirical interpolation method (DEIM) for treating the non-affine parameter dependence introduced through the mapping function. To help tackle the resulting high dimensional parametric problem, we split the full domain into six rotationally symmetric subdomains. In particular, our split into six subdomains allowed us to construct the aforementioned low-dimensional spaces by considering one subdomain only. We then used the same set of reduced basis functions for every subdomain. On the one hand, this idea reduced the dimensionality of the parametrized problem from 12 for the complete domain to only 4, and on the other hand, also implies the utilization of the same low-dimensional spaces to represent the parametric velocity fields in lobules of different topology. To couple the subdomains, we used an interior penalty type formulation at the subdomain interfaces, and therefore, enforced the velocity fields to be continuous within the full domain.

For sampling random shapes and sizes of lobules, we generated a Voronoi diagram from a Delaunay triangulation, where we used an energy minimization formulation to control the packing and shapes of the lobule structures. We randomly picked 1,000 samples for training the reduced order model and obtaining the aforementioned low-dimensional spaces. We assessed the approximation quality of the reduced order model in the entire parameter space via a multi-dimensional error based on the  $L^2$  norm of the velocity and determined the number of reduced basis functions required in the Galerkin projection and the DEIM to guarantee that the reduced order results remain below a 10% error bound. We validated our reduced order model for 1000 new parameter combinations, which were not a part of the samples for the initial training. We demonstrated that the model performed very well, and in particular did not violate our selected error limit. This error limit could be further reduced by enriching the low-dimensional spaces with additional reduced basis functions. For more complicated data structures (e.g., due to a different physical model), more advanced reduction techniques may be considered to achieve a better balance between accuracy and computational effort. However, this level of accuracy is sufficient to indicate the successful applicability of the presented reduced order modeling strategy for the underlying physical model. We also showed that with our first implementation that is far from optimized, the reduced order model – even including the additional effort of the DEIM – is more than one order of magnitude faster than the corresponding high-fidelity finite element computations. We also assessed the mass conservation property of our formulation. Within each subdomain it is exactly satisfied via the choice of our finite element approximation spaces and within the complete domain, the absolute relative mass flow rates of all samples were found to be below 0.01%, indicating that mass conservation is practically satisfied by the reduced order parametric velocity fields.

In the future, we plan to extend our approach to three-dimensional lobules and tessellation structure of lobules, where we expect a decisive advantage of the reduced order over the high fidelity finite element computations in terms of computational efficiency.

We also plan to use the advantages of our approach in terms of fast computation without meshing effort for variable lobule geometry for stochastic analysis of blood perfusion.

#### Acknowledgements

Same as funding.

#### Author contributions

Conception and design of the work: SKFS, DS Implementation and creation of new computer code: AAS (reduced order model), EJ (sampling algorithms). Data acquisition and analysis: AAS. Interpretation of the data: all authors. Initial draft and writing of the article: AAS, EJ. Substantial revision of the article: SKFS, DN, DS. All authors read and approved the final manuscript.

#### Funding

Open Access funding enabled and organized by Projekt DEAL. A.A. Siddiqui, D. Néron, and D. Schillinger gratefully acknowledge the financial support from the German Research Foundation (DFG) within the International Research Training Group 2657 (IRTG 2657) entitled “Computational Mechanics Techniques in High Dimensions” (grant number 433082294). E. Jessen and D. Schillinger gratefully acknowledge the financial support of the ERC Starting grant project ImageToSim that has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement no. 759001).

#### Availability of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author upon reasonable request.

#### Declarations

##### Competing interests

The authors declare that they have no competing interests.

Received: 27 January 2024 Accepted: 5 September 2024

Published online: 11 December 2024

#### References

1. Khaled A-RA, Vafai K. The role of porous media in modeling flow and heat transfer in biological tissues. *Int J Heat Mass Transf.* 2003;46(26):4989–5003.
2. Soltani M, Chen P. Numerical modeling of interstitial fluid flow coupled with blood flow through a remodeled solid tumor microvascular network. *PLoS ONE.* 2013;8(6):e67025.
3. Mosharaf-Dehkordi M. A fully coupled porous media and channels flow approach for simulation of blood and bile flow through the liver lobules. *Comput Methods Biomech Biomed Eng.* 2019;22(9):901–15.
4. Stoter SKF, Müller P, Cicalese L, Tuveri M, Schillinger D, Hughes TJR. A diffuse interface method for the Navier-Stokes/Darcy equations: perfusion profile for a patient-specific human liver based on MRI scans. *Comput Methods Appl Mech Eng.* 2017;321:70–102.
5. Melito GM, Müller TS, Badeli V, Ellermann K, Brenn G, Reinbacher-Köstinger A. Sensitivity analysis study on the effect of the fluid mechanics assumptions for the computation of electrical conductivity of flowing human blood. *Reliabil Eng Syst Saf.* 2021;213: 107663.
6. Mohammadi B, Pironneau O. Shape optimization in fluid mechanics. *Annu Rev Fluid Mech.* 2004;36(1):255–79.
7. Cotin S, Delingette H, Ayache N. Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Trans Visual Comput Graphics.* 1999;5(1):62–73.
8. Néron D, Ladevèze P. Proper generalized decomposition for multiscale and multiphysics problems. *Arch Comput Methods Eng.* 2010;17(4):351–72.
9. Courard A, Néron D, Ladevèze P, Ballere L. Integration of PGD-virtual charts into an engineering design process. *Comput Mech.* 2016;57(4):637–51.
10. Heyberger C, Boucard P-A, Néron D. Multiparametric analysis within the proper generalized decomposition framework. *Comput Mech.* 2012;49(3):277–89.
11. Quarteroni A, Rozza G. *Reduced order methods for modeling and computational reduction.* Cham: Springer; 2014.
12. Benner P, Ohlberger M, Cohen A, Willcox K. *Model reduction and approximation.* Comput Sci Eng Soc Industr Appl Math. 2017.
13. Quarteroni A, Manzoni A, Negri F. *Reduced basis methods for partial differential equations.* UNITEXT, vol. 92. Cham. 2016.
14. Hesthaven JS, Rozza G, Stamm B. *Certified reduced basis methods for parametrized partial differential equations.* Cham: Springer; 2016.
15. Ricken T, Werner D, Holzhütter H, König M, Dahmen U, Dirsch O. Modeling function-perfusion behavior in liver lobules including tissue, blood, glucose, lactate and glycogen by use of a coupled two-scale pde-ode approach. *Biomech Model Mechanobiol.* 2015;14:515–36.
16. Ricken T, Lambers L. On computational approaches of liver lobule function and perfusion simulation. *GAMM-Mitteilungen.* 2019;42(4):201900016.
17. Rohan E, Camprová Turjanicová J, Liška V. Geometrical model of lobular structure and its importance for the liver perfusion analysis. *PLoS ONE.* 2021;16(12):0260068.
18. Greenway CV, Stark RD. Hepatic vascular bed. *Physiol Rev.* 1971;51(1):23–65.

19. Majno P, Mentha G, Toso C, Morel P, Peitgen HO, Fasel JHD. Anatomy of the liver: an outline with three levels of complexity—a further step towards tailored territorial liver resections. *J Hepatol*. 2014;60(3):654–62.
20. Debbaut C, Segers P, Cornillie P, Casteleyn C, Dierick M, Laleman W, Monbaliu D. Analyzing the human liver vascular architecture by combining vascular corrosion casting and micro-ct scanning: a feasibility study. *J Anat*. 2014;224(4):509–17.
21. Jessen E, Steinbach MC, Debbaut C, Schillinger D. Rigorous mathematical optimization of synthetic hepatic vascular trees. *J R Soc Interface*. 2022;19(191):20220087.
22. Wang Y, Brodin E, Nishii K, Frieboes HB, Mumenthaler SM, Sparks JL, Macklin P. Impact of tumor-parenchyma biomechanics on liver metastatic progression: a multi-model approach. *Sci Rep*. 2021;11(1):1710.
23. Oğul H, Kantarci M, Genç B, Pirimoğlu B, Çullu N, Kızrak Y, Yılmaz Ö, Karabulut N. Perfusion CT imaging of the liver: review of clinical applications. *Diagn Interv Radiol*. 2014;20(5):379–89.
24. Levesque E, Martin E, Dudau D, Lim C, Dhonneur G, Azoulay D. Current use and perspective of indocyanine green clearance in liver diseases. *Anaesthesia Crit Care Pain Med*. 2016;35(1):49–57.
25. Ahmadi-Badejani R, Mosharaf-Dehkordi M, Ahmadikia H. An image-based geometric model for numerical simulation of blood perfusion within the liver lobules. *Comput Methods Biomech Biomed Eng*. 2020;23:987–1004.
26. Jessen E, Steinbach MC, Debbaut C, Schillinger D. Branching exponents of synthetic vascular trees under different optimality principles. *IEEE Trans Biomed Eng*. 2024;71(4):1345–54.
27. Ebrahem A, Jessen E, Eikelder MFP, Gangwar T, Mika M, Schillinger D. Connecting continuum poroelasticity with discrete synthetic vascular trees for modelling liver tissue. *Proc Roy Soc A Math Phys Eng Sci*. 2024;480(2285):20230421.
28. Bonfiglio A, Leungchavaphongse K, Repetto R, Siggers JH. Mathematical modeling of the circulation in the liver lobule. *J Biomech Eng*. 2010;132(11):111011.
29. Ricken T, Dahmen U, Dirsch O. A biphasic model for sinusoidal liver perfusion remodeling after outflow obstruction. *Biomech Model Mechanobiol*. 2010;9:435–50.
30. Debbaut C, Vierendeels J, Casteleyn C, Cornillie P, Van Loo D, Simoens P, Van Hoorebeke L, Monbaliu D, Segers P. Perfusion characteristics of the human hepatic microcirculation based on three-dimensional reconstructions and computational fluid dynamic analysis. *J Biomech Eng*. 2012;134:011003.
31. Debbaut C, Vierendeels J, Siggers JH, Repetto R, Monbaliu D, Segers P. A 3D porous media liver lobule model: the importance of vascular septa and anisotropic permeability for homogeneous perfusion. *Comput Methods Biomech Biomed Engin*. 2014;17(12):1295–310.
32. Rohan E, Lukeš V, Jonášová A. Modeling of the contrast-enhanced perfusion test in liver based on the multi-compartment flow in porous media. *J Math Biol*. 2018;77(2):421–54.
33. Pébay P, Baker T. Analysis of triangle quality measures. *Math Comput*. 2003;72(244):1817–39.
34. Brezzi F, Fortin M. Function spaces and finite element approximations. In: Brezzi F, Fortin M, editors. *Mixed and hybrid finite element methods*. Springer series in computational mathematics. New York, NY: Springer; 1991. p. 89–132.
35. Rognes ME, Kirby RC, Logg A. Efficient assembly of  $H(\text{div})$  and  $H(\text{curl})$  conforming finite elements. *SIAM J Sci Comput*. 2010;31(6):4130–51.
36. Brezzi F, Hughes T, Marini L, Masud A. Mixed discontinuous Galerkin methods for Darcy flow. *J Sci Comput*. 2005;22:119–45.
37. Brezzi F, Douglas J, Marini LD. Two families of mixed finite elements for second order elliptic problems. *Numer Math*. 1985;47(2):217–35.
38. Alnaes M, Blechta J, Hake J, Johansson A, Kehlet B, Logg A, Richardson C, Ring J, Rognes ME, Wells GN. The FEniCS Project Version 1.5. *Arch Nume Softw*. 2015;3(100).
39. Multiphenics—easy prototyping of multiphysics problems in FEniCS. <https://mathlab.sissa.it/multiphenics>.
40. Lau C, Kalantari B, Batts K, Ferrell L, Nyberg S, Graham R, Moreira RK. The Voronoi theory of the normal liver lobular architecture and its applicability in hepatic zonation. *Sci Rep*. 2021;11(1):9343.
41. Fortune S. A sweepline algorithm for voronoi diagrams. In: *Proceedings of the Second Annual Symposium on Computational Geometry*, pp. 313–322; 1986.
42. Bowyer A. Computing dirichlet tessellations. *Comput J*. 1981;24(2):162–6.
43. Yan D-M, Wang W, Lévy B, Liu Y. Efficient computation of clipped voronoi diagram for mesh generation. *Comput Aided Des*. 2013;45(4):843–52.
44. Liu DC, Nocedal J. On the limited memory bfgs method for large scale optimization. *Math Program*. 1989;45(1–3):503–28.
45. Du Q, Gunzburger M. Grid generation and optimization based on centroidal voronoi tessellations. *Appl Math Comput*. 2002;133(2–3):591–607.
46. Lucia DJ, Beran PS, Silva WA. Reduced-order modeling: new approaches for computational physics. *Prog Aerosp Sci*. 2004;40(1):51–117.
47. Zahr MJ, Farhat C. Progressive construction of a parametric reduced-order model for PDE-constrained optimization. *Int J Numer Meth Eng*. 2015;102(5):1111–35.
48. Haasdonk B, Ohlberger M. Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Math Comput Model Dyn Syst*. 2011;17(2):145–61.
49. Arnold DN, Brezzi F, Cockburn B, Marini LD. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J Numer Anal*. 2002;39(5):1749–79.
50. Schillinger D, Harari I, Hsu M-C, Kamensky D, Stoter SK, Yu Y, Zhao Y. The non-symmetric nitsche method for the parameter-free imposition of weak boundary and coupling conditions in immersed finite elements. *Comput Methods Appl Mech Eng*. 2016;309:625–52.
51. Hong Q, Wang F, Wu S, Xu J. A unified study of continuous and discontinuous Galerkin methods. *Sci China Math*. 2019;62(1):1–32.
52. Wang F, Wu S, Xu J. A mixed discontinuous Galerkin method for linear elasticity with strongly imposed symmetry. *J Sci Comput*. 2020;83(1):2.
53. Eckart C, Young G. The approximation of one matrix by another of lower rank. *Psychometrika*. 1936;1(3):211–8.

54. Chaturantabut S, Sorensen DC. Discrete Empirical Interpolation for nonlinear model reduction. In: Proceedings of the 48th IEEE Conference on Decision and Control (CDC) Held Jointly with 2009 28th Chinese Control Conference, pp. 4316–4321; 2009.
55. Chaturantabut S, Sorensen DC. Nonlinear model reduction via discrete empirical interpolation. *SIAM J Sci Comput.* 2010;32(5):2737–64.
56. Stoter SKF, Jessen E, Niedens V, Schillinger D. A DEIM driven reduced basis method for the diffuse Stokes/Darcy model coupled at parametric phase-field interfaces. *Comput Geosci.* 2022;26(6):1465–502.

### **Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.