



Swansea University
Prifysgol Abertawe



Cronfa - Swansea University Open Access Repository

This is an author produced version of a paper published in:
Computer Methods in Applied Mechanics and Engineering

Cronfa URL for this paper:
<http://cronfa.swan.ac.uk/Record/cronfa51756>

Paper:

Sevilla, R., Zlotnik, S. & Huerta, A. (2020). Solution of geometrically parametrised problems within a CAD environment via model order reduction. *Computer Methods in Applied Mechanics and Engineering*, 358, 112631
<http://dx.doi.org/10.1016/j.cma.2019.112631>

This item is brought to you by Swansea University. Any person downloading material is agreeing to abide by the terms of the repository licence. Copies of full text items may be used or reproduced in any format or medium, without prior permission for personal research or study, educational or non-commercial purposes only. The copyright for any work remains with the original author unless otherwise specified. The full-text must not be sold in any format or medium without the formal permission of the copyright holder.

Permission for multiple reproductions should be obtained from the original author.

Authors are personally responsible for adhering to copyright and publisher restrictions when uploading content to the repository.

<http://www.swansea.ac.uk/library/researchsupport/ris-support/>

Solution of geometrically parametrised problems within a CAD environment via model order reduction

Ruben Sevilla^{a,*}, Sergio Zlotnik^b, Antonio Huerta^b

^a*Zienkiewicz Centre for Computational Engineering, College of Engineering, Swansea University, Bay Campus, SA1 8EN, Wales, United Kingdom.*

^b*Laboratori de Calcul Numeric (LaCàN), ETS de Ingenieros de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya Barcelona, Spain.*

Abstract

The main objective of this work is to describe a general and original approach for computing an off-line solution for a set of parameters describing the geometry of the domain. That is, a solution able to include information for different geometrical parameter values and also allowing to compute readily the sensitivities. Instead of problem dependent approaches, a general framework is presented for standard engineering environments where the geometry is defined by means of NURBS. The parameters controlling the geometry are now the control points characterising the NURBS curves or surfaces. The approach proposed here, valid for 2D and 3D scenarios, allows a seamless integration with CAD preprocessors. The proper generalised decomposition (PGD), which is applied here to compute explicit geometrically parametrised solutions, circumvents the curse of dimensionality. Moreover, optimal convergence rates are shown for PGD approximations of incompressible flows.

Keywords: Geometry parametrisation, Reduced order model, Computer-aided design (CAD), Proper generalised decomposition (PGD).

*Corresponding author

Email addresses: `r.sevilla@swansea.ac.uk` (Ruben Sevilla), `sergio.zlotnik@upc.edu` (Sergio Zlotnik), `antonio.huerta@upc.edu` (Antonio Huerta)

1. Introduction

The current role of computational simulations in modern engineering design is limited by the complexity of the simulations that are required, particularly during the final stages of a design. The main problem is motivated by the number of configurations that need to be tested (e.g. loads, boundary conditions, material parameters and geometric configurations).

One alternative to decrease the computational complexity in this scenario is to introduce a reduced order model [30]. The main idea involves projecting the governing equations describing the full model onto a space with lower dimension that is described using a reduced order basis. Well known methods to produce reduced order basis are Krylov-based methods [18], the reduced basis method [32] and the proper orthogonal decomposition (POD) [5, 23]. More recently, the proper generalised decomposition (PGD) [9, 10] has gained popularity due to its ability to build reduced basis with no prior knowledge of the solution. The PGD starts by considering the solution not only as a function of the standard coordinates (i.e., space and time) but also of any parameter of interest (e.g. boundary conditions, external loads, material parameters). The problem involving a range of all the desired parameters can be solved at the cost of several problems of the same size as the original problem for a particular choice of the parameters. This expensive calculation, usually referred as the *off-line stage*, is performed only once, usually making use of high performance computing resources, to build the reduced order basis that is written as an explicit function of the coordinates (space and time) and the parameters (i.e. a computational vademecum, see [11]). Then, the *on-line stage* consists of just a particularisation of the solution by using, in the simplest case, an interpolation of the already computed results.

The PGD has been successfully applied to numerous multi-dimensional problems involving boundary conditions, material parameters and external loads as extra coordinates, to name a few, see [9, 10] and references therein. The application to problems involving geometrically parametrised domains is generally more challenging and the existing work is usually limited to simple geometries where the extra coordinates involve a scaling: the length of an interval in one-dimensional problems [11], the thickness of extruded geometries [6, 22] or problem specific parameters [19]. More recently, an approach based on an initial subdivision of the computational domain in macro-elements was proposed in [2]. This idea was extended to domains with interfaces in [40] and has been also applied to an engineering design process in [13]. Other reduced order models, such as POD, have been recently considered for general parametric geometries [4] but its application in a PGD framework remains unexplored. The main difficulty of dealing with geometric parameters, compared to material, boundary conditions or other parameters, in this context is to obtain a separable expression of the discrete problem.

In this paper, a new approach to incorporate the geometric parameters as extra coordinates in a PGD framework is proposed. The objective is to produce a general methodology to solve geometrically parametrised problems that can be used for any geometrical model. This is in contrast with existing methodologies [2, 40] that require a piecewise linear approximation of the geometry to ensure that the finite element mesh can be nested in the mesh of macro-elements used to parametrise the geometry. Here, the domain is assumed to be parametrised using the NURBS boundary representation of the domain, as usually done in a computer-aided design (CAD) environment. The control points of the NURBS entities defining the boundary are considered as extra coordinates and a mapping between a reference domain and

the current configuration is proposed by using a solid mechanics analogy. An explicit and separated representation of the mapping is derived and the application of the proposed methodology to Stokes flow problems is presented using examples of increasing difficulty in two and three dimensions. Contrary to the approach in [2, 40], the methodology proposed in this paper considers geometric parameters that are independent of the spatial discretisation, i.e. the control points of the NURBS entities defining the boundary representation of the domain. In addition, the technique presented here enables the solution of the multi-dimensional problem to be computed using curved high-order finite elements [38]. This is in contrast with the technique introduced in [2, 40], which requires an affine mapping between a reference element and the so-called *macro-elements* that are used to parametrise the geometry and it is therefore not applicable to curved high-order elements. In addition, the technique proposed in this work uses geometric parameters that are directly related to the CAD entities defining the boundary of the domain, making it suitable to interactive shape design. The proposed technique is considered an important step towards the full integration of computational geometry and analysis [12, 21, 33, 34], specially in the field of reduced order modelling.

The proposed methodology is related to the shape calculus commonly used in the context of parametric shape optimisation [14]. In fact, the proposed methodology can be used to obtain explicit closed form expressions of the the shape derivatives as they can be computed from the generalised PGD solution at a negligible cost.

It is worth mentioning that a NURBS boundary representation of the domain is considered in this work. Other geometric representations found in CAD systems could also be considered within the proposed framework, provided that appropriate geometric parameters are selected to perform ge-

ometric changes of the CAD model.

The structure of the remainder of the paper is as follows. Section 2 presents the problem statement using the Poisson equation on a geometrically parametrised domain and summarises the application of the PGD. Section 3 describes in detail the proposed technique to build a generalised solution assuming that a mapping between a reference configuration and the current one can be written in separated form. In Section 4, a methodology to integrate this approach in a CAD environment is presented. Considering the geometric parameters as the control points of the NURBS entities describing the boundary of the domain, the methodology to build a mapping that can be explicitly written in separated form is detailed. Finally, Section 5 presents a series of numerical examples of increasing difficulty involving the solution of Stokes flow problems in two and three dimensions.

2. Problem statement and geometrically parametrised solutions

2.1. The Poisson equation on a parametrised domain

The methodology proposed here can be directly extended to second order linear problems such as the Stokes problem that is studied in the examples. However, in order to simplify the presentation, the heat problem (Poisson) is presented in detail for a parametrised domain $\Omega^\mu \subset \mathbb{R}^{n_{sd}}$ (with n_{sd} number of spatial dimensions), whose boundary $\partial\Omega^\mu$ is characterised by a set of geometric parameters $\boldsymbol{\mu} \in \mathcal{I} \subset \mathbb{R}^{n_{pa}}$ (with n_{pa} number of parameters characterising the geometry) and is partitioned into Dirichlet, Γ_D^μ , and Neumann, Γ_N^μ , boundaries such that $\overline{\partial\Omega^\mu} = \overline{\Gamma_D^\mu} \cup \overline{\Gamma_N^\mu}$ and $\Gamma_D^\mu \cap \Gamma_N^\mu = \emptyset$. Note that the set $\mathcal{I} \subset \mathbb{R}^{n_{pa}}$, which characterises the admissible range for parameters $\boldsymbol{\mu}$, can be defined as the Cartesian combination of the range for each parameter, namely, $\mathcal{I} := \mathcal{I}_1 \times \mathcal{I}_2 \times \dots \times \mathcal{I}_{n_{pa}}$ with $\mu_i \in \mathcal{I}_i$ for $i = 1, \dots, n_{pa}$.

For each set of parameter $\boldsymbol{\mu}$, the objective is to determine the parametric solution $u^\mu(\boldsymbol{x})$, with $\boldsymbol{x} \in \Omega^\mu$, of the boundary value problem

$$\begin{cases} -\nabla \cdot (\mathbf{K} \nabla u^\mu) = s & \text{in } \Omega^\mu, \\ u^\mu = u_D & \text{on } \Gamma_D^\mu, \\ \boldsymbol{n}^\mu \cdot (\mathbf{K} \nabla u^\mu) = t & \text{on } \Gamma_N^\mu, \end{cases} \quad (1)$$

where \mathbf{K} is the thermal conductivity (symmetric and positive definite) matrix, s is a source term, u_D is the imposed temperature, t is the imposed heat normal flux and \boldsymbol{n} is the outward unit normal vector. The standard variational form of the previous problem reads: find $u^\mu \in \mathcal{V}^\mu$ for all $v \in \mathcal{H}_{\Gamma_D^\mu}^1$ such that

$$a(\boldsymbol{\mu}; u^\mu, v) = \ell(\boldsymbol{\mu}; v), \quad (2a)$$

where the space of trial functions is $\mathcal{V}^\mu := \{w \in \mathcal{H}^1(\Omega^\mu) : w = u_D \text{ on } \Gamma_D^\mu\}$ and its corresponding test functions space is $\mathcal{H}_{\Gamma_D^\mu}^1 := \{w \in \mathcal{H}^1(\Omega^\mu) : w = 0 \text{ on } \Gamma_D^\mu\}$. The parametric bilinear and linear forms $a(\boldsymbol{\mu}; \cdot, \cdot)$ and $\ell(\boldsymbol{\mu}; \cdot)$ are defined by

$$a(\boldsymbol{\mu}; u, v) := (\nabla v, \mathbf{K} \nabla u)_{\Omega^\mu} \text{ and } \ell(\boldsymbol{\mu}; v) := (v, s)_{\Omega^\mu} + \langle v, t \rangle_{\Gamma_N^\mu}, \quad (2b)$$

where

$$(u, v)_\Omega = \int_\Omega u v \, d\Omega, \quad (\mathbf{u}, \mathbf{v})_\Omega = \int_\Omega \mathbf{u} \cdot \mathbf{v} \, d\Omega, \text{ and } \langle u, v \rangle_\Gamma = \int_\Gamma u v \, d\Gamma,$$

denote, respectively, the \mathcal{L}_2 product of scalar/vector functions in Ω and its traces over Γ .

2.2. The multi-dimensional parametric problem

There are different alternatives to obtain, for any given set of parameters $\boldsymbol{\mu}$, an approximation to the solution $u^\mu(\boldsymbol{x})$ of problem (1). The obvious

option of solving a new problem for every instance of $\boldsymbol{\mu}$ is feasible but too costly. A standard strategy to reduce the cost is to pre-compute off-line some representative samples of the parametric family of solutions (e.g. snapshots for reduced basis methods, principal components for POD). Then, any other instance is computed on-line with a small computational overhead. Here, the PGD is preferred because the off-line phase provides an explicit description of the parametric solution, i.e. a computational vademecum see [11]. Thus, in spite of an off-line phase more involved, the on-line phase is a simple functional evaluation with a negligible computational overhead.

In practice, this can be interpreted as taking $\boldsymbol{\mu}$ as additional independent variables (or parametric coordinates) instead of problem parameters. Hence, the unknown temperature field is not interpreted any more as a parametric solution, denoted as u^μ , but it is seen now as a function in a larger dimensional space and it is written as $u(\boldsymbol{x}, \boldsymbol{\mu})$ with $(\boldsymbol{x}, \boldsymbol{\mu}) \in \Omega^\mu \times \mathcal{I}$. Consequently, formally u lies in a tensor product space, namely, $u \in \mathcal{V}^\mu \otimes \mathcal{L}_2(\mathcal{I}_1) \otimes \mathcal{L}_2(\mathcal{I}_2) \otimes \cdots \otimes \mathcal{L}_2(\mathcal{I}_{n_{\text{pa}}})$. A standard weighted residuals approach, with integrals in $\Omega^\mu \times \mathcal{I}$ and the usual integration by parts only in Ω^μ produces a weak form in this multi-dimensional setup. Namely, find $u \in \mathcal{V}^\mu \otimes \mathcal{L}_2(\mathcal{I}_1) \otimes \mathcal{L}_2(\mathcal{I}_2) \otimes \cdots \otimes \mathcal{L}_2(\mathcal{I}_{n_{\text{pa}}})$ such that

$$A(u, v) = L(v), \quad \forall v \in \mathcal{H}_{\Gamma_D}^1 \otimes \mathcal{L}_2(\mathcal{I}_1) \otimes \mathcal{L}_2(\mathcal{I}_2) \otimes \cdots \otimes \mathcal{L}_2(\mathcal{I}_{n_{\text{pa}}}), \quad (3a)$$

with the following definitions of the bilinear and linear forms

$$\begin{aligned} A(u, v) &:= \int_{\mathcal{I}_1} \int_{\mathcal{I}_2} \cdots \int_{\mathcal{I}_{n_{\text{pa}}}} a(\boldsymbol{\mu}; u, v) d\mu_{n_{\text{pa}}} \cdots d\mu_2 d\mu_1 \quad \text{and} \\ L(v) &:= \int_{\mathcal{I}_1} \int_{\mathcal{I}_2} \cdots \int_{\mathcal{I}_{n_{\text{pa}}}} \ell(\boldsymbol{\mu}; v) d\mu_{n_{\text{pa}}} \cdots d\mu_2 d\mu_1. \end{aligned} \quad (3b)$$

Obviously, the number of dimensions of the solution domain increases with the number of parameters. To circumvent the *curse of dimensionality*, the PGD approach [3, 9–11] is employed here. This approach assumes

a separable structure in the function that approximates u . Note that the tensor product space $\mathcal{V}^\mu \otimes \mathcal{L}_2(\mathcal{I}_1) \otimes \mathcal{L}_2(\mathcal{I}_2) \otimes \cdots \otimes \mathcal{L}_2(\mathcal{I}_{n_{\text{pa}}})$ inherits the multi-dimensional complexity of the problem and, in principle, does not assume separability of the functions.

Moreover, the solution of (3a) requires an affine parameter dependence of the different forms. This is standard in reduced order methods and it is very well discussed in [26, 31]. More precisely, it is required that the different forms are expressed (or at least well approximated) by the sum of products of parameter-dependent functions and parameter-independent operators, for instance

$$a(\boldsymbol{\mu}; u, v) = \sum_{q=1}^Q \left(\prod_{i=1}^{n_{\text{pa}}} \Theta_i^q(\mu_i) \right) a^q(u, v),$$

where Θ_i^q is a function that only depends upon the parameter μ_i . Note that the forms $a^q(u, v)$ do not depend on the parameters (in particular, they are integrated over domains parameter independent). In fact, finding the affine parameter dependence of (2b) is a major concern in subsequent sections and will enable to obtain a separated approximation of the solution, namely

$$u \approx u_{\text{PGD}}^n = \sum_{m=1}^n \psi_m^u(\boldsymbol{\mu}) w_m(\boldsymbol{x}).$$

3. Separated spatial mapping to determine generalised solutions

If the affine parameter dependence must be enforced, it is necessary to integrate in space over domains not depending on the parameters. Note that forms $a(\boldsymbol{\mu}; u, v)$ and $\ell(\boldsymbol{\mu}; v)$, see (2b), are integrated on spacial parametrised domains (domains depending on parameters $\boldsymbol{\mu}$). As suggested in [2, 40], a mapping \mathcal{M}_μ is necessary (not sufficient) in order to have an affine parameter dependence. This mapping transforms a reference domain Ω into the

geometrically parametrised (“deformed”) domain Ω^μ , namely

$$\begin{aligned} \mathcal{M}_\mu &: \Omega \times \mathcal{I} \longrightarrow \Omega^\mu \\ (\mathbf{X}, \boldsymbol{\mu}) &\longmapsto \mathcal{M}_\mu(\mathbf{X}, \boldsymbol{\mu}) = \mathbf{x} = \mathbf{X} + \mathbf{d}(\mathbf{X}, \boldsymbol{\mu}). \end{aligned} \quad (4)$$

As classically in computational mechanics, the reference configuration Ω is associated to a reference coordinate system denoted by \mathbf{X} , whereas the distorted domain Ω^μ will be associated to the spatial description \mathbf{x} . Following this analogy, a *displacement* field \mathbf{d} is used to relate both coordinate systems. It is worth noting that the mapping of Equation 4 resembles the so-called *family of perturbations* employed in a shape analysis context [14] to compute the derivatives of a shape functional.

The mapping \mathcal{M}_μ can be defined in an *ad hoc* manner for each problem [2, 13] or by a more general strategy [2, 40], but always has to induce an affine parameter dependence of forms (2b). The strategy developed here is general and it only requires a separated representation (or at least a good approximation) of the displacement field $\mathbf{d}(\mathbf{X}, \boldsymbol{\mu})$. As it will be shown in Section 4, this work proposes a way to integrate the PGD framework within a CAD environment. Therefore, the displacement field will be constructed based upon the motion of the control points of the NURBS describing the boundary of the computational domain.

The introduction of the mapping \mathcal{M}_μ allows to rewrite (2b) as integrals over the reference computational domain Ω and its corresponding boundary $\partial\Omega$, partitioned into the Dirichlet, Γ_D , and Neumann, Γ_N , boundaries, all independent of parameter $\boldsymbol{\mu}$. Namely,

$$\begin{aligned} a(\boldsymbol{\mu}; u, v) &= (\mathbf{J}_\mu^{-1} \nabla_{\mathbf{X}} v, \det(\mathbf{J}_\mu) \mathbf{K} \mathbf{J}_\mu^{-1} \nabla_{\mathbf{X}} u)_\Omega = (\nabla_{\mathbf{X}} v, \mathbf{H}_\mu \nabla_{\mathbf{X}} u)_\Omega \quad \text{and} \\ \ell(\boldsymbol{\mu}; v) &= (v, \det(\mathbf{J}_\mu) s)_\Omega + \langle v, \mathbf{n}^\mu \cdot [\text{adj}(\mathbf{J}_\mu) \mathbf{n}] t \rangle_{\Gamma_N}, \end{aligned} \quad (5)$$

where $[\mathbf{J}_\mu]_{ij} = [\partial x_j / \partial X_i]$ is the Jacobian matrix of the mapping \mathcal{M}_μ and the Nanson’s formula has been used to transform the integral on the Neumann

boundary. It is worth noting that \mathbf{J}_μ and \mathbf{n}^μ are the only elements in the above equations that depend upon the parameters μ . Moreover, in order to compact the notation a new matrix \mathbf{H}_μ is introduced,

$$\mathbf{H}_\mu := \frac{\text{adj}(\mathbf{J}_\mu^T) \mathbf{K} \text{adj}(\mathbf{J}_\mu)}{\det(\mathbf{J}_\mu)}, \quad (6)$$

and the definition of the *adjoint* operator, $\text{adj}(\mathbf{A}) = \det(\mathbf{A}) \mathbf{A}^{-1}$, has been used.

Note that even for *ad hoc* [2] or a piecewise linear [2, 40] mappings, the affine parameter dependence of the different forms in (5) must be now determined.

Remark 1 (Mapping for the Stokes problem). *Despite being more cumbersome, the same rationale can be applied for the Stokes problem without any extra conceptual challenge. The bilinear viscosity form is reproduced here for illustration purposes*

$$a(\mu; \mathbf{u}, \mathbf{v}) = (\nabla \mathbf{v}, \underline{\underline{\mathbf{C}}} : \nabla \mathbf{u})_{\Omega^\mu} = (\nabla_{\mathbf{X}} \mathbf{v}, \hat{\underline{\underline{\mathbf{C}}}}^\mu : \nabla_{\mathbf{X}} \mathbf{u})_\Omega,$$

where $C_{ijkl} = \nu \delta_{ik} \delta_{jl}$, $\hat{C}_{ijkl}^\mu = \nu \det(\mathbf{J}_\mu) \delta_{ik} [\mathbf{J}_\mu^{-T}]_{js} [\mathbf{J}_\mu^{-1}]_{sl}$ and $\nu > 0$ is the kinematic viscosity. Note that $[\mathbf{A}]_{ij}$ denotes the component ij of a matrix \mathbf{A} .

3.1. Separated displacements

The methodology proposed here allows to obtain an affine parameter dependence (namely, a separable expression for \mathbf{H}_μ) quasi-analytically if the mapping can be written (or at least, well approximated) with a separated representation, that is as a sum of separated terms, namely

$$\mathcal{M}_\mu(\mathbf{X}, \mu) = \mathbf{X} + \sum_{m=1}^N \psi_m(\mu) \mathbf{d}_m(\mathbf{X}), \quad (7)$$

where ψ_m and \mathbf{d}_m are parametric and spatial functions respectively and N is the total number of terms required to express the mapping in a separated

form. Section 4 details the strategy to construct the N parametric and spatial functions using the CAD boundary representation of the domain with NURBS.

The separated representation of the mapping, given by equation (7), induces a separated Jacobian matrix

$$\mathbf{J}_\mu(\mathbf{X}, \boldsymbol{\mu}) = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}(\mathbf{X}, \boldsymbol{\mu}) = \mathbf{I}_{\text{nsd}} + \sum_{m=1}^N \psi_m \mathbf{A}_m, \quad (8)$$

where $\mathbf{A}_m := [\partial \mathbf{d}_m / \partial \mathbf{X}]$ for $m = 1, \dots, N$.

3.2. Affine parameter dependence

The affine parameter dependence of $a(\boldsymbol{\mu}; u, v)$ as defined in (5), is in practice determined by obtaining a separated expression of \mathbf{H}_μ . In order to obtain a separated expression for matrix \mathbf{H}_μ , both $\text{adj}(\mathbf{J}_\mu)$ and $\text{det}(\mathbf{J}_\mu)$ are analytically separated. Then, the higher order PGD projection proposed in [25] is used to obtain a compact separation for \mathbf{H}_μ .

The separated representation for the determinant, $\text{det}(\mathbf{J}_\mu)$, can be obtained using *Leibniz formula* from equation (8), whereas for $\text{adj}(\mathbf{J}_\mu)$, the Leverrier's algorithm [17] is employed. This method is a consequence of the Cayley-Hamilton theorem [7, 20] and the Newton's identities [24]. It enables to express the adjoint of a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ in terms of its trace and its powers, namely

$$\text{adj}(\mathbf{A}) = \sum_{s=0}^{n-1} \mathbf{A}^s \sum_{k_1, k_2, \dots, k_{n-1}} \prod_{l=1}^{n-1} \frac{(-1)^{k_l+1}}{l^{k_l} k_l!} \text{tr}(\mathbf{A}^l)^{k_l} \quad (9)$$

where $k_l \in \mathbb{N}_0$ and $s + \sum_{l=1}^{n-1} l k_l = n - 1$. In practice, the two cases of interest are 2D ($n = 2$) and 3D ($n = 3$) problems, which are detailed next.

3.3. Two-dimensional approach

In 2D the Jacobian \mathbf{J}_μ is a 2×2 matrix. From (8), the separated expression for its determinant is

$$\det(\mathbf{J}_\mu) = \left(1 + \sum_{m=1}^N \psi_m(\boldsymbol{\mu}) [\mathbf{A}_m]_{11}\right) \left(1 + \sum_{m=1}^N \psi_m(\boldsymbol{\mu}) [\mathbf{A}_m]_{22}\right) - \left(\sum_{m=1}^N \psi_m(\boldsymbol{\mu}) [\mathbf{A}_m]_{21}\right) \left(\sum_{m=1}^N \psi_m(\boldsymbol{\mu}) [\mathbf{A}_m]_{12}\right), \quad (10)$$

again $[\mathbf{A}]_{ij}$ denotes the component ij of the matrix \mathbf{A} .

To obtain the expression for the adjoint requires to particularise (9), namely

$$\text{adj}(\mathbf{A}) = \text{tr}(\mathbf{A})\mathbf{I}_2 - \mathbf{A},$$

which is a linear mapping because the trace is also linear. Thus, the Jacobian in separated form as presented in (8) can be written as

$$\text{adj}(\mathbf{J}_\mu) = \mathbf{I}_2 + \sum_{m=1}^N \psi_m \text{adj}([\partial \mathbf{d}_m / \partial \mathbf{X}]) = \mathbf{I}_2 + \sum_{m=1}^N \psi_m \text{adj}(\mathbf{A}_m). \quad (11)$$

Consequently, recalling that $\text{adj}(\mathbf{A}^T) = \text{adj}(\mathbf{A})^T$, the matrix \mathbf{H}_μ in (6) can be rewritten as

$$\begin{aligned} \det(\mathbf{J}_\mu) \mathbf{H}_\mu &= \mathbf{K} + \sum_{m=1}^N \psi_m \mathbf{K} \text{adj}(\mathbf{A}_m) + \sum_{m=1}^N \psi_m [\mathbf{K} \text{adj}(\mathbf{A}_m)]^T \\ &\quad + \sum_{m=1}^N \sum_{l=1}^N \psi_m \psi_l \text{adj}(\mathbf{A}_m)^T \mathbf{K} \text{adj}(\mathbf{A}_l). \end{aligned}$$

The separated expression of \mathbf{H}_μ is efficiently obtained with a numerical higher order PGD projection [25].

3.4. Three-dimensional approach

Following the previous rationale, in 3D the Jacobian \mathbf{J}_μ is a 3×3 matrix. The separated expression for the determinant, $\det(\mathbf{J}_\mu)$, is obtained using

Leibniz formula as

$$\det(\mathbf{J}_\mu) = \sum_{\sigma \in \mathcal{S}_3} \text{sgn}(\sigma) \prod_{i=1}^3 \left(1 + \sum_{m=1}^N \psi_m(\boldsymbol{\mu}) [\mathbf{A}_m]_{i\sigma(i)} \right), \quad (12)$$

where \mathcal{S}_3 is the set of the six permutations of the integers $\{1, 2, 3\}$, where the element in position i after the reordering σ is denoted $\sigma(i)$, and $\text{sgn}(\sigma)$ denotes the signature of σ (i.e. $+1$ for even σ and -1 for odd σ). Note that such a separation will induce sums of order N^3 .

The adjoint of the Jacobian is also separated by means of particularising (9) to 3D,

$$\text{adj}(\mathbf{A}) = \frac{1}{2} [(\text{tr}(\mathbf{A}))^2 - \text{tr}(\mathbf{A}^2)] \mathbf{I}_3 - \text{tr}(\mathbf{A}) \mathbf{A} + \mathbf{A}^2,$$

and given the separation of the Jacobian in (8) implies

$$\begin{aligned} \text{adj}(\mathbf{J}_\mu) &= \mathbf{I}_3 + \sum_{m=1}^N \psi_m [\text{tr}(\mathbf{A}_m) \mathbf{I}_3 - \mathbf{A}_m] \\ &+ \sum_{m=1}^N \sum_{l=1}^N \psi_m \psi_l \left[\frac{1}{2} (\text{tr}(\mathbf{A}_m) \text{tr}(\mathbf{A}_l) - \text{tr}(\mathbf{A}_m \mathbf{A}_l)) \mathbf{I}_3 - \text{tr}(\mathbf{A}_m) \mathbf{A}_l + \mathbf{A}_m \mathbf{A}_l \right]. \end{aligned} \quad (13)$$

As done in 2D, once the adjoint is separated, from (6), a separated expression for $\det(\mathbf{J}_\mu) \mathbf{H}_\mu = \text{adj}(\mathbf{J}_\mu^T) \mathbf{K} \text{adj}(\mathbf{J}_\mu)$ can be computed. In this case, it consists of N^4 terms. As previously noted for 2D, the separated expression of \mathbf{H}_μ is obtained with the higher order PGD projection described next.

3.5. The higher order PGD projection

To obtain a separated approximation of \mathbf{H}_μ , namely

$$\mathbf{H}_\mu \approx \mathbf{H}_\mu^{\text{sep}}(\mathbf{x}, \boldsymbol{\mu}) = \sum_{m=1}^{N_H} \psi_m^{\text{H}}(\boldsymbol{\mu}) \mathbf{A}_m^{\text{H}}(\mathbf{X}),$$

equation (6) has to be solved. This is done by means of the higher order PGD projection [25], which in practice obtains the separated approximation with an \mathcal{L}_2 projection, say

$$(\mathbf{v}, \mathbf{H}_\mu^{\text{sep}})_{\Omega \times \mathcal{I}} = (\mathbf{v}, \text{adj}(\mathbf{J}_\mu^T) \mathbf{K} \text{adj}(\mathbf{J}_\mu) / \det(\mathbf{J}_\mu))_{\Omega \times \mathcal{I}} \quad (14)$$

for all \mathbf{v} in a suitable space. In practice, the PGD strategy is implemented. Thus, a greedy approach is used. For computational efficiency, it is critical to use the exact separated expressions for the determinant, see (10) and (12), and the adjoint of the Jacobian, see (11) and (13). In practice, each mode of $\mathbf{H}_\mu^{\text{sep}}$ is obtained by solving

$$\begin{aligned} (\mathbf{v}, \det(\mathbf{J}_\mu) \psi_n^{\mathbf{H}} \mathbf{A}_n^{\mathbf{H}})_{\Omega \times \mathcal{I}} &= (\mathbf{v}, \text{adj}(\mathbf{J}_\mu^T) \mathbf{K} \text{adj}(\mathbf{J}_\mu))_{\Omega \times \mathcal{I}} \\ &\quad - (\mathbf{v}, \det(\mathbf{J}_\mu) \sum_{m=1}^{n-1} \psi_m^{\mathbf{H}} \mathbf{A}_m^{\mathbf{H}})_{\Omega \times \mathcal{I}}, \end{aligned} \quad (15)$$

where now, as usual in PGD, \mathbf{v} lives in the tangent space of the mode. By solving Equation (15) instead of (14), the separated representation of the determinant and adjoint of the Jacobian can be used on the right hand side. This enables evaluating the integrals in the high dimensional space $\Omega \times \mathcal{I}$ as products of integrals in the spatial domain Ω and integrals in the parametric dimensions \mathcal{I}_i , for $i = 1, \dots, \mathbf{n}_{\text{pa}}$.

Note that the higher order PGD projection allows many parameters without any prior knowledge, is computationally efficient, and the precision of this approximation (i.e. the total number of terms N_H) can be controlled by the user. Finally, it is worth emphasising that Equation (15) is solved only once and provides an approximated separated representation of the matrix \mathbf{H}_μ for any value of the geometric parameters.

When the Neumann boundary depends upon the parameters the same PGD-projection rationale is applied to obtain a separated representation of

the normal vector \mathbf{n}^μ appearing in Equation 5. This operation is performed once and provides a separable approximation of \mathbf{n}^μ that can be used for any value of the geometric parameters.

4. Integration within a CAD environment

In this section a procedure to integrate the methodology described in the previous section within a CAD environment, employing non-uniform rational B-spline (NURBS), is proposed. The fundamental concepts on non-uniform rational B-spline (NURBS) curves and surfaces are summarised in [Appendix A](#).

To simplify the presentation, the technique to integrate the PGD approach with a CAD environment is described here for the two dimensional case. The details for the extension to three dimensional domains are given in [Appendix B](#).

The boundary of the parametrised domain, $\partial\Omega^\mu$, is assumed to be described by a set of NURBS curves $\{\mathbf{C}_j^\mu\}_{j=1,\dots,M}$, being M the total number of curves, namely

$$\partial\Omega^\mu = \bigcup_{j=1}^M \mathbf{C}_j^\mu([0, 1]).$$

4.1. Geometric parameters

The geometric parameters $\boldsymbol{\mu} \in \mathcal{I} \subset \mathbb{R}^{\text{npa}}$ are defined as the variations of the original coordinates of the control points of the NURBS curves describing the boundary. More precisely, for each NURBS curve \mathbf{C}_j , with $j = 1, \dots, M$, having $\mathbf{n}_{\text{cp}}^j + 1$ control points, the undisturbed boundary is characterised by the coordinates of the control points: \mathbf{B}_i^j , for $i = 0, \dots, \mathbf{n}_{\text{cp}}^j$. This configuration will be used as the reference one in Ω and will be associated to a reference coordinate system denoted by \mathbf{X} . The distorted domain, Ω^μ , will

be associated to the spatial description \mathbf{x} . The boundary in the spatial domain, $\partial\Omega^\mu$, is defined by the position of the displaced control points, namely $\mathbf{B}_i^j + \delta\mathbf{B}_i^j$. The displacement range for each control point is characterised by

$$\delta\mathbf{B}_i^j \in \mathcal{I}_i^j = [\underline{\delta x}_i^j, \overline{\delta x}_i^j] \times [\underline{\delta y}_i^j, \overline{\delta y}_i^j].$$

In fact, each displacement of a control point i on the j -th NURBS, $\delta\mathbf{B}_i^j$, might depend upon the parameters and can be written as

$$\delta\mathbf{B}_i^j = \mu_1^{i,j} \mathbf{e}_1 + \mu_2^{i,j} \mathbf{e}_2, \quad (16)$$

where \mathbf{e}_i , for $i = 1, 2$, are the unit coordinate vectors. Then $\boldsymbol{\mu} \in \mathcal{I} := \mathcal{I}^1 \times \mathcal{I}^2 \times \dots \times \mathcal{I}^M$, where $\mathcal{I}^j = \mathcal{I}_1^j \times \mathcal{I}_2^j \times \dots \times \mathcal{I}_{n_{cp}^j+1}^j$ is the range of variation of the coordinates of the control points of the curve \mathbf{C}_j .

Consequently, the maximum number of geometric parameters is

$$\mathbf{n}_{pa} = \sum_{j=1}^M (\mathbf{n}_{cp}^j + 1) \mathbf{n}_{sd}, \quad (17)$$

but in practical problems, the number of geometric parameters \mathbf{n}_{pa} is drastically lower than m because not all the control points of all the curves are to be modified during the design stage.

Remark 2. *In a practical setting it is common to introduce some restrictions on the motion of the control points (viz. pure translations, rotations, expansions...), meaning that the motion of a set of control points can be expressed with a significantly low number of parameters.*

4.2. Separated representation of the boundary displacement

The variation of a control point \mathbf{B}_i^j of a NURBS curve \mathbf{C}_j , namely $\delta\mathbf{B}_i^j$, changes the definition of the original curve only in the support of the basis

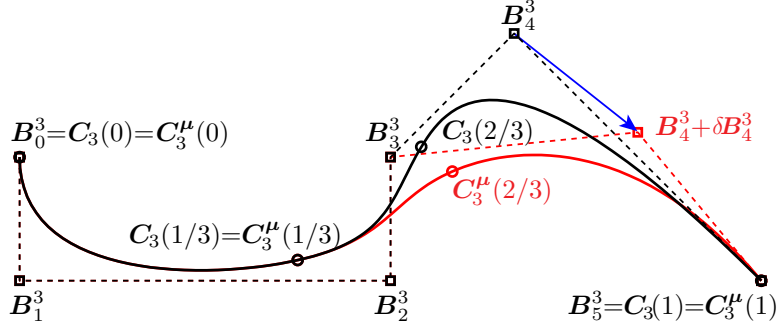


Figure 1: Change in the shape of a NURBS curve induced by the variation of the coordinates of the control point \mathbf{B}_4^3 given by $\delta\mathbf{B}_4^3$.

function R_i^j , given by the subspace of the parametric space $[\lambda_i, \lambda_{i+q^j+1}]$. The modified NURBS curve is parametrised by

$$\mathbf{C}_j^\mu(\lambda) = \sum_{i=0}^{n_{cp}^j} (\mathbf{B}_i^j + \delta\mathbf{B}_i^j) R_i^j(\lambda) \quad \lambda \in [0, 1]. \quad (18)$$

Figure 1 illustrates the effect of modifying the coordinates of one control point of a NURBS. The curve in red is the result of modifying the coordinates of the control point \mathbf{B}_4^3 of the original curve in black, also depicted in Figure A.34. It can be observed that both curves are identical in the interval $[0, 1/3]$ of the parametric space whereas they differ in the interval $[1/3, 1]$, which is the support of the basis function R_4^3 associated to the control point \mathbf{B}_4^3 .

Given a computational mesh for the reference configuration, Ω , the boundary mesh nodes affected by the motion of a control point can be easily identified. The procedure starts by finding the NURBS curve to which each boundary mesh node belongs and its associated parametric coordinate by using a standard NURBS point projection algorithm [28]. For each boundary mesh node \mathbf{X}_k , the index $j \in \{1, \dots, M\}$ and parametric coordinate λ_k such that $\mathbf{C}_j(\lambda_k) = \mathbf{X}_k$ are computed. Then, for a deformed configuration, induced by the variation of a control point \mathbf{B}_i^j of a NURBS curve \mathbf{C}_j , namely

$\delta \mathbf{B}_i^j$, the new position of each boundary node is computed as $\mathbf{x}_k^\mu = \mathbf{C}_j^\mu(\lambda_k)$.

Therefore, the variation of the control points of a boundary curve \mathbf{C}_j induces a displacement of the boundary mesh nodes, namely

$$\mathbf{d}_j^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \mathbf{x}_k^\mu - \mathbf{X}_k \quad , \text{ for all } k \in \mathcal{S},$$

where $\mathcal{S} = \{l \in \{1, 2, \dots, \mathbf{n}_{\text{mn}}\} : \mathbf{X}_l \in \partial\Omega\}$ is the index set of mesh nodes on the boundary of the computational domain and \mathbf{n}_{mn} is the total number of mesh nodes. It is worth noting that \mathbf{d}^∂ denotes a displacement field defined only on the boundary of the domain, different to the displacement field in Equation (7) that is defined in the whole domain.

Using the expression of the original and modified NURBS boundary curves, the displacement of the boundary mesh node $\mathbf{X}_k = \mathbf{C}_j(\lambda_k)$ that belongs to the NURBS curve \mathbf{C}_j can be written in separated form as

$$\mathbf{d}_j^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{i=0}^{\mathbf{n}_{\text{cp}}^j} \delta \mathbf{B}_i^j R_i^j(\lambda_k) = \sum_{i=0}^{\mathbf{n}_{\text{cp}}^j} \sum_{s=1}^{\mathbf{n}_{\text{sd}}} \mu_s^{i,j} \mathbf{e}_s R_i^j(\lambda_k),$$

where the dependence of the displacements of the control points in terms of the parameters described in (16) has been used. Moreover, since the NURBS parameter λ_k is only dependent on the spatial coordinates \mathbf{X}_k , and not on the geometric parameters $\boldsymbol{\mu}$, the previous equation can be written as,

$$\mathbf{d}^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{j=1}^M \mathbf{d}_j^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{j=1}^M \sum_{i=0}^{\mathbf{n}_{\text{cp}}^j} \sum_{s=1}^{\mathbf{n}_{\text{sd}}} \mu_s^{i,j} \mathbf{e}_s R_i^j(\mathbf{C}_j^{-1}(\mathbf{X}_k)),$$

which characterises the displacement of the boundary nodes and has a separated expression of the form

$$\mathbf{d}^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{i=1}^{\mathbf{n}_{\text{pa}}} \phi_i(\mu_i) \boldsymbol{\delta}_i(\mathbf{X}_k). \quad (19)$$

Note that this expression is compatible with the desired structure of the displacements described in Equation (7). In fact, it can be further compacted

by means of the higher order PGD projection [25] to obtain a more compact separation in the form of the displacement field in Equation (7), namely

$$\mathbf{d}^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{i=1}^{\mathbf{n}_{\text{pa}}} \phi_i(\mu_i) \boldsymbol{\delta}_i(\mathbf{X}_k) = \sum_{m=1}^N \psi_m^\partial(\boldsymbol{\mu}) \mathbf{d}_m^\partial(\mathbf{X}_k), \quad (20)$$

where the precise expression of the parametric and spatial functions, ψ_m^∂ and \mathbf{d}_m^∂ respectively, is given by the higher order PGD projection described in Section 3.5.

It is important to note that the total number of terms in the separated representation, N , will be, in the majority of cases, much lower than \mathbf{n}_{pa} , which is defined in (17). In practical applications, a large percentage of mesh boundary nodes will not be affected by a variation of the control points of a particular NURBS curve describing the boundary, so the displacement will be zero for a large number of boundary nodes. More precisely, the set of mesh nodes affected by the variation of a control point \mathbf{B}_i^j of a NURBS curve \mathbf{C}_j can be defined as

$$\mathcal{S}_i^j = \{l \in \mathcal{S} : \mathbf{X}_l = \mathbf{C}_j(\lambda_l) \text{ for } \lambda_l \in [\lambda_i, \lambda_{i+q+1}]\}. \quad (21)$$

4.3. Separated representation of the geometric mapping

The proposed strategy to build a mapping $\mathcal{M}_\boldsymbol{\mu}$ between the reference configuration, Ω , and the current configuration, $\Omega^\boldsymbol{\mu}$, consists on solving a solid mechanics problem. The reference configuration is assumed to be a linear elastic medium and the displacement of mesh boundary nodes, induced by the variation of NURBS control points, is interpreted as a Dirichlet boundary condition. The following problem governing the static deformation of Ω is considered

$$\begin{cases} \nabla_{\mathbf{X}} \cdot \boldsymbol{\sigma} + \mathbf{f} = \mathbf{0} & \text{in } \Omega \\ \mathbf{d}(\mathbf{X}) = \mathbf{d}^\partial(\mathbf{X}, \boldsymbol{\mu}) & \text{on } \partial\Omega, \end{cases} \quad (22)$$

where \mathbf{f} is an external force defined by the user. The stress tensor $\boldsymbol{\sigma}$ is given by

$$\boldsymbol{\sigma} = \frac{E\nu}{(1+\nu)(1-2\nu)} \text{tr}(\boldsymbol{\varepsilon})\mathbf{I} + \frac{E}{1+\nu}\boldsymbol{\varepsilon},$$

where E and ν denote the Young's Modulus and the Poisson's ratio of the elastic medium and the deformation tensor is defined as

$$\boldsymbol{\varepsilon} = \boldsymbol{\nabla}_{\mathbf{X}}^S \mathbf{d} := \frac{1}{2} (\boldsymbol{\nabla}_{\mathbf{X}} \mathbf{d} + (\boldsymbol{\nabla}_{\mathbf{X}} \mathbf{d})^T).$$

with \mathbf{d} being the unknown displacement field.

This strategy has been successfully applied in the context of high-order curved mesh generation, see [27, 29, 39] for further details. The key aspect here is to write the approximated solution, \mathbf{d}_h , in separated form by using the separated representation of the imposed boundary displacement derived in Section 4.2.

The discretisation of the weak formulation associated to the strong form of the solid mechanics problem (22) leads to a system of linear equations that can be written as

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{Bmatrix} \mathbf{d} \\ \mathbf{d}^\partial \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \end{Bmatrix}, \quad (23)$$

where \mathbf{d} and \mathbf{d}^∂ are vectors containing the nodal values of the approximated displacement \mathbf{d}_h and the imposed displacement of boundary nodes respectively.

The matrix \mathbf{A}_{11} in (23) is symmetric and positive definite. Thus, solving for \mathbf{d} induces a linear application applied on \mathbf{d}^∂ , see Appendix C for more details. Since \mathbf{d}^∂ has a separable expression, see (19), the solution of the previous system will induce a separated expression for all nodal values and, consequently, the piecewise interpolation, standard in finite elements,

produces a separated representation at any point of the domain, namely

$$\mathbf{d}_h(\mathbf{X}, \boldsymbol{\mu}) = \sum_{m=1}^N \psi_m(\boldsymbol{\mu}) \mathbf{d}_m(\mathbf{X}), \quad (24)$$

which is exactly the desired structure of the displacements in Equation 7.

5. Numerical examples

This section presents three numerical examples that show the optimal approximation properties of the proposed PGD approach and its potential for two and three dimensional problems involving geometric parameters as extra coordinates. The examples involve the simulation of Stokes flows using a separable expression that employs the same parametric function for both velocity and pressure. This alternative was shown to be superior to other approaches in [15], for instance, to satisfy the so-called Ladyzhenskaya-Babuška-Brezzi (LBB) condition [16]. In all the examples k denotes the degree of approximation used for the velocity field and the parametric functions, whereas a degree of approximation $k - 1$ is used for the pressure field. This ensures satisfaction of the LBB, although the accuracy of the pressure field is expected to be lower than the accuracy of the velocity field.

5.1. Rotating Couette flow

The first example considers the Couette flow around two infinite coaxial circular cylinders centred at the origin and with radius R_{in} and R_{out} respectively, with $R_{\text{in}} < R_{\text{out}}$, as represented in Figure 2. The boundary conditions correspond to known angular velocities, Ω_{in} and Ω_{out} , at R_{in} and R_{out} , respectively. It is worth noting that the pressure must be specified at a point to remove its indeterminacy, as only Dirichlet boundary conditions for the velocity are considered. Here the pressure is imposed at one point of the outer boundary.

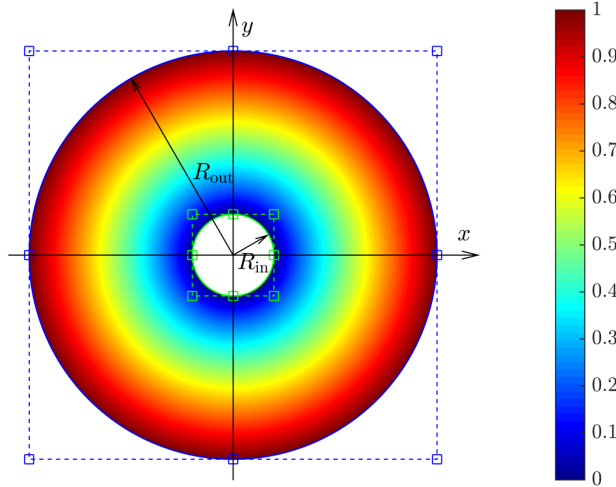


Figure 2: Computational domain for the solution of the rotating Couette flow and magnitude of the velocity of the analytical solution.

The analytical solution for this problem is known [8]. The azimuthal component of the velocity is given by

$$v_\phi = \frac{R_{\text{out}}^2 \Omega_{\text{out}} - R_{\text{in}}^2 \Omega_{\text{in}}}{R_{\text{out}}^2 - R_{\text{in}}^2} r + \frac{(\Omega_{\text{in}} - \Omega_{\text{out}}) R_{\text{out}}^2 R_{\text{in}}^2}{R_{\text{out}}^2 - R_{\text{in}}^2} \frac{1}{r} \quad (25)$$

where $r = \|\mathbf{x}\|_2$. The magnitude of the velocity of the analytical solution is also depicted in Figure 2 for $\Omega_{\text{in}} = 0$ and $\Omega_{\text{out}} = 1$.

The inner radius R_{in} is considered an extra parameter within the proposed PGD framework and the objective is to find, in the off-line stage, the generalised velocity and pressure fields for $R_{\text{in}} \in [1, 2.5]$. It is worth noting that the variation of the inner radius induces the variation of the eight control points (i.e. 16 parameters in two dimensions) of the NURBS curve describing the inner circle, as represented in Figure 2. However, as mentioned in Remark 2, the motion of these control points can be controlled by a single parameter μ representing the variation of the radius of the inner circle.

In this example the reference configuration corresponds to $R_{\text{in}} = 1$ and $R_{\text{out}} = 5$ and $\mu \in \mathcal{I}_1 = [0, 1.5]$. Three unstructured triangular meshes of

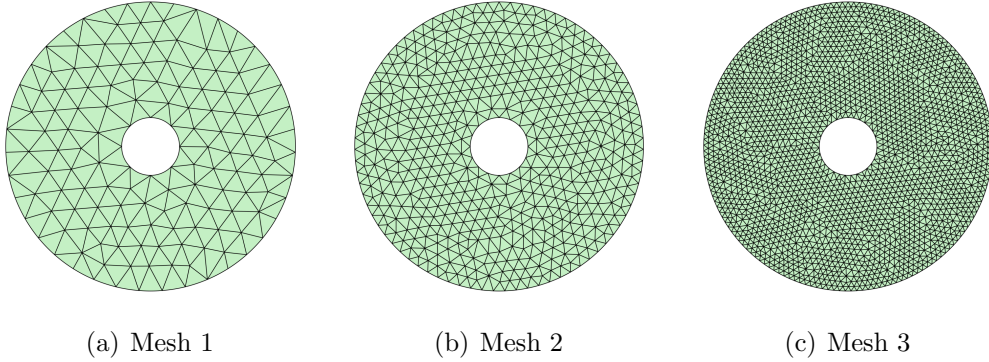


Figure 3: Three unstructured triangular meshes of the reference domain used for the solution of the Couette flow problem.

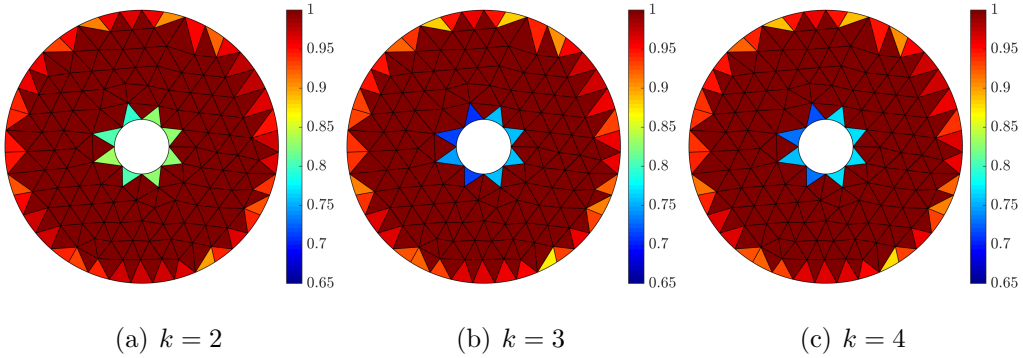


Figure 4: Quality of the coarsest mesh of the reference domain, shown in Figure 3 (a), using different degrees of approximation.

the reference domain, with 251, 1,023 and 4,256 elements respectively, are represented in Figure 3. These meshes are generated using the technique proposed in [35] to guarantee that elements without an edge on a curved boundary can be mapped to a reference triangle using an affine mapping.

The quality of the coarsest mesh of the reference domain, measured using the scaled Jacobian [39], is represented in Figure 4. The minimum quality observed is, as expected near the inner boundary where the elements show the maximum distortion.

Figure 5 shows the deformed configuration for three values of the param-

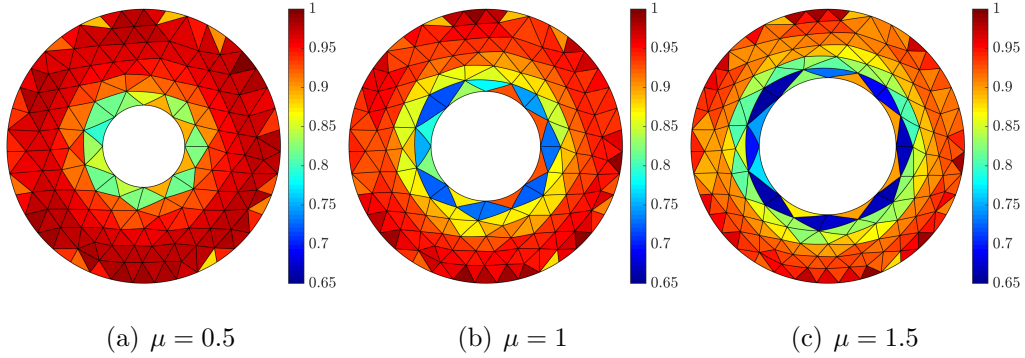


Figure 5: Quality of the coarsest mesh of the deformed domain with $k = 4$ for different values of the parameter μ .

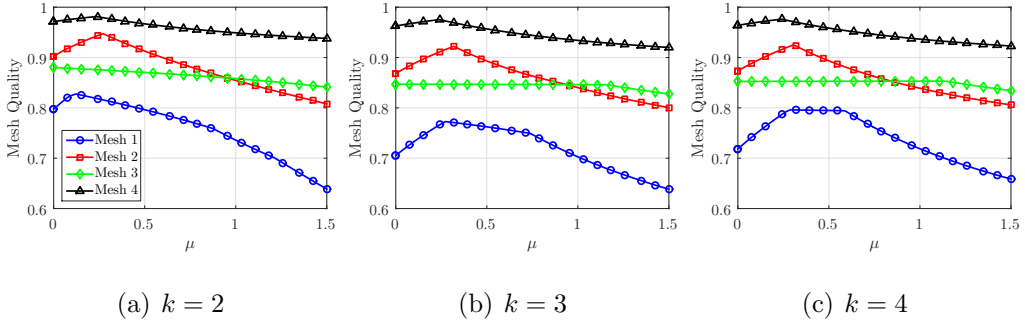


Figure 6: Evolution of the quality of the deformed meshes corresponding to the reference mesh of Figure 3 (a) as a function of the parameter μ using different meshes and degrees of approximation.

eter μ , namely $\mu = 0.5$, $\mu = 1$ and $\mu = 1.5$. These plots also represent the quality of the deformed mesh obtained by using the elastic analogy described in Section 4.3. In all cases, the minimum quality is higher than 0.65.

To further illustrate the robustness of the mesh deformation technique employed within the proposed PGD framework, Figure 6 shows the evolution of the quality of the deformed meshes as a function of the parameter μ using four different meshes and three different degrees of approximation. As expected, the worst case scenario corresponds to the maximum deformation induced by a parameter $\mu = 1.5$, but in all cases the minimum quality is

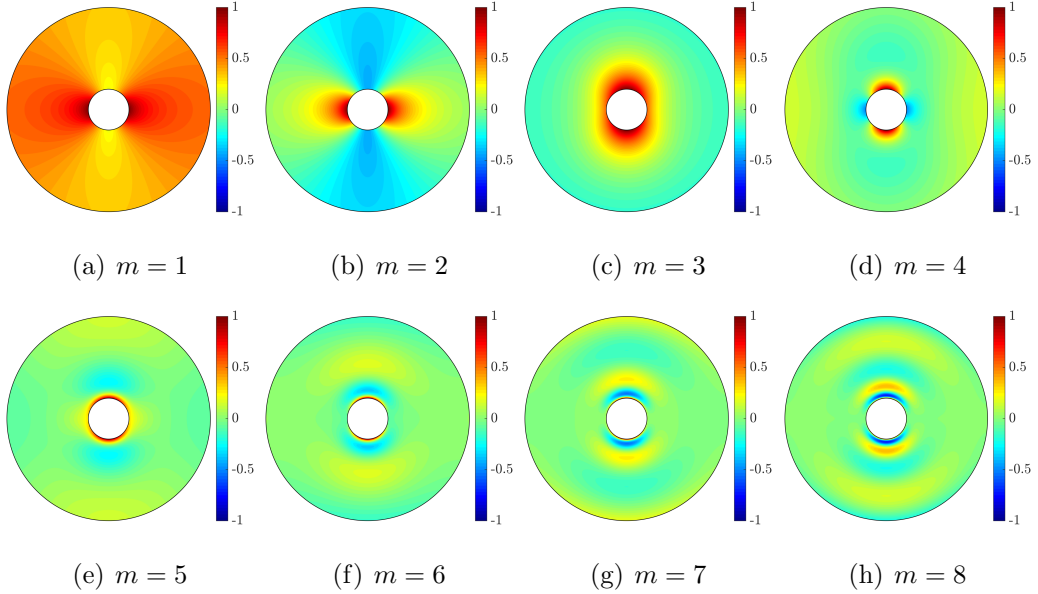


Figure 7: First eight normalised spatial modes of $[\mathbf{H}_\mu]_{11}$ on the mesh of Figure 3 (c) with 800 elements in the parametric dimension and with $k = 4$.

always higher than 0.6. Furthermore, it is worth noting that the influence of the parameter μ on the quality of the deformed meshes is less important for finer meshes.

A crucial aspect of the proposed PGD strategy is the separation of the matrix \mathbf{H}_μ defined in Equation (6). As discussed previously, \mathbf{H}_μ does not generally admit an exact separable expression and therefore a separable approximation is computed here via the higher-order PGD-projection [25].

Figure 7 shows the first eight normalised spatial modes of the component $[\mathbf{H}_\mu]_{11}$. The results suggest that the first two modes capture the main global features of $[\mathbf{H}_\mu]_{11}$, whereas the rest of modes capture the local variations near the inner circle. The results for the second diagonal component, $[\mathbf{H}_\mu]_{22}$, not displayed for brevity, show the same behaviour but with the expected rotation of 90 degrees due to the symmetry of the domain and the displacement field. Similarly, Figure 8 shows the first eight normalised spatial modes of the

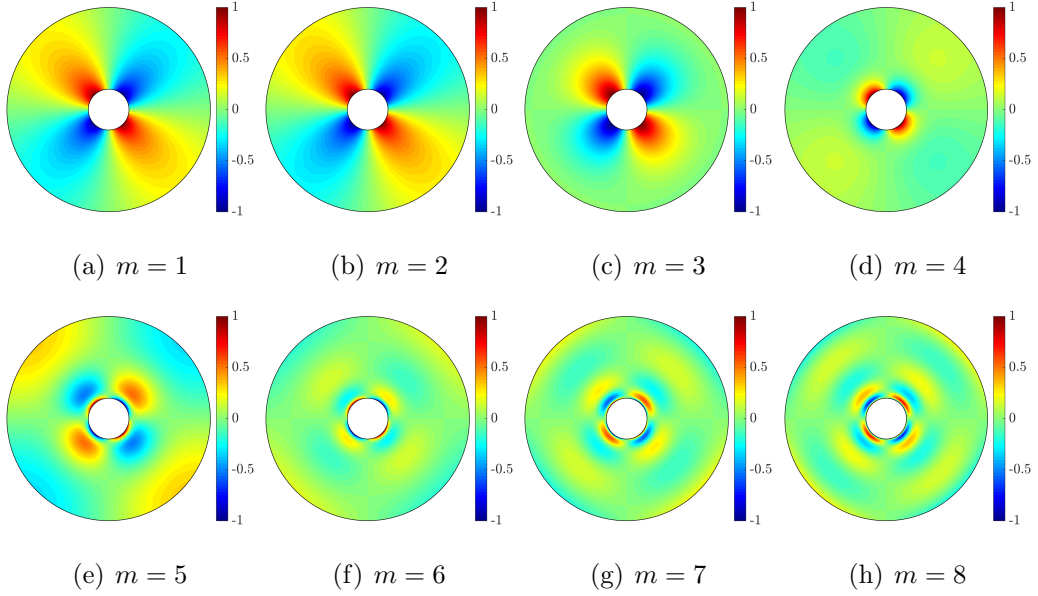


Figure 8: First eight normalised spatial modes of $[\mathbf{H}_\mu]_{12}$ on the mesh of Figure 3 (c) with 800 elements in the parametric dimension and with $k = 4$.

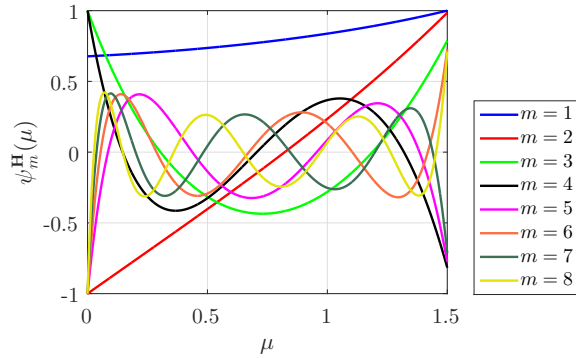


Figure 9: First eight parametric modes of \mathbf{H}_μ on the mesh of Figure 3 (c) with 800 elements in the parametric dimension and with $k = 4$.

component $[\mathbf{H}_\mu]_{12}$. The results show, again how the first modes capture the global behaviour of the component $[\mathbf{H}_\mu]_{12}$, whereas the last modes show relevant spatial variations in the close vicinity of the inner circle.

The first eight normalised parametric modes of \mathbf{H}_μ are represented in Figure 9. It is worth recalling that the same parametric modes are associated

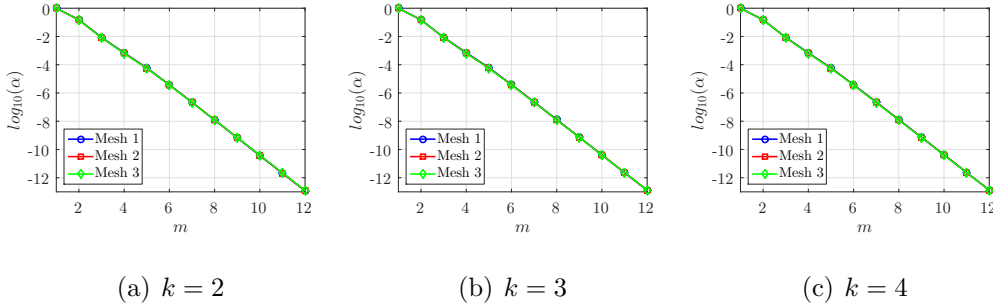


Figure 10: Evolution of the amplitude of the spatial modes α_m of the matrix \mathbf{H}_μ using different meshes and degrees of approximation.

to all the components of the matrix \mathbf{H}_μ .

The amplitude, α_m , corresponding to the mode m of the separation of \mathbf{H}_μ , is computed as the product of the Euclidean norms of the spatial and parametric functions. Figure 10 shows the amplitudes of $[\mathbf{H}_\mu]_{11}$ using three different meshes and three different degrees of approximation. The results show that, for this example, the number of terms required to obtain a separable approximation of the matrix \mathbf{H}_μ using the higher-order PGD-projection is completely independent on the spatial discretisation. In all cases, 12 modes provide a decrease in the amplitude α_m of exactly 13 orders of magnitude and the amplitude is the same in all meshes and for all degrees of approximation.

Using the separation of the matrix \mathbf{H}_μ , the rotating Couette flow problem is solved to obtain the generalised solution of the Stokes problem. The first eight normalised spatial modes of the magnitude of the velocity field are shown in Figure 11. The simulation was performed using the mesh of Figure 3 (c), 800 equally-spaced elements in $\mathcal{I}_1 = [0, 1.5]$ and a degree of approximation $k = 4$. The solution of the coaxial Couette flow problem requires imposing Dirichlet boundary conditions for the velocity at the inner and outer boundaries of the spatial domain. This is implemented within the PGD framework by adding one initial mode that fulfils the non-homogeneous

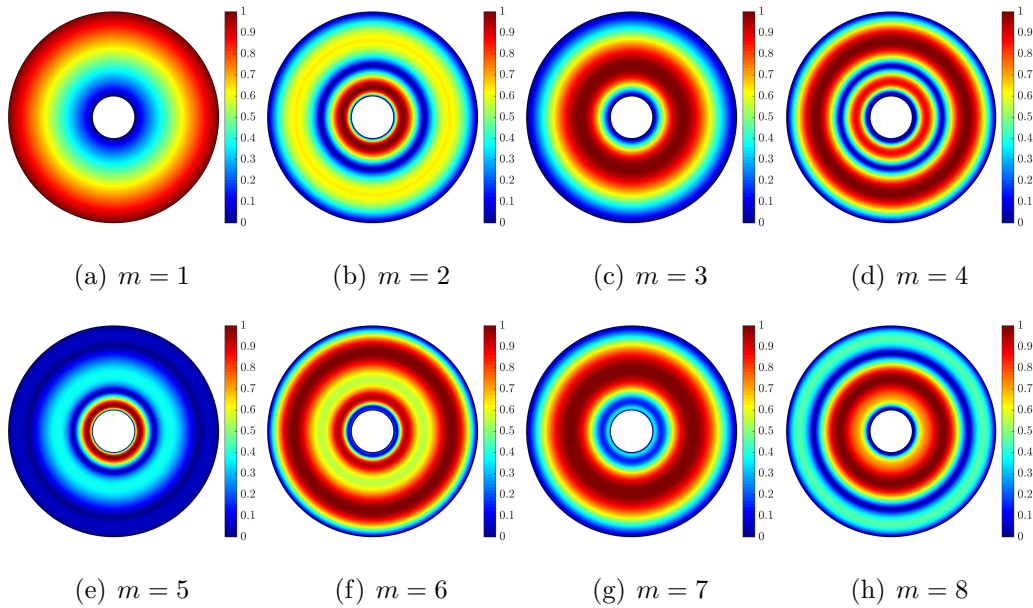


Figure 11: First eight normalised spatial modes of the magnitude of the velocity computed on the mesh of Figure 3 (c), 800 elements in the parametric dimension and a degree of approximation $k = 4$.

Dirichlet boundary conditions. Its spatial part is computed as the solution of the Stokes problem at the reference mesh and its parametric component is taken as a constant function equal to one. The next PGD modes are computed by imposing homogeneous Dirichlet boundary conditions, ensuring that the separated PGD solution satisfies the required Dirichlet boundary conditions.

The first eight normalised parametric modes associated to the spatial modes of Figure 11 are represented in Figure 12. It is worth recalling that the same parametric mode is associated to all the components of the velocity and the pressure fields. The results reveal that the parametric modes of the velocity field show a similar behaviour compared to the parametric modes of the separation of \mathbf{H}_μ , shown in Figure 9. The first two modes are smooth whereas the next modes, that contribute less to the global solution, show a

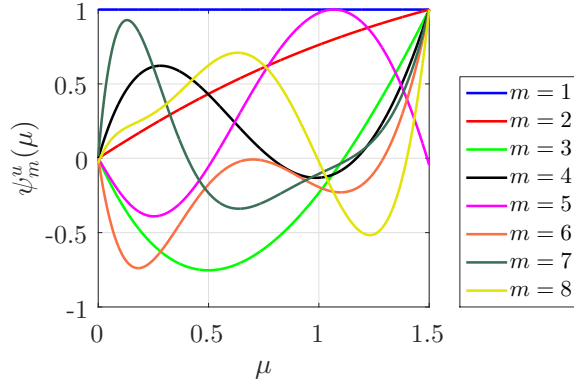


Figure 12: First eight parametric modes of the rotating Couette flow on the mesh of Figure 3 (c) with 800 elements in the parametric dimension and with $k = 4$.

more oscillatory character.

To illustrate the gain in accuracy as the number of modes increases, Figure 13 shows the absolute value of the error in the magnitude of the velocity computed with the proposed PGD approach using N modes for different values of the parameter μ . The computation has been performed on the mesh of Figure 3 (c), with 800 elements in the parametric dimension and a degree of approximation $k = 4$. In all cases, the results illustrate that the computation with four modes is almost three orders of magnitude more accurate than using a single mode.

To further analyse the accuracy of the proposed PGD approach when the number of modes is increased, the relative error in the $\mathcal{L}_2(\Omega \times \mathcal{I}_1)$ norm, defined as

$$\varepsilon_{\text{PGD}} = \left(\frac{\int_{\mathcal{I}_1} \int_{\Omega} (\mathbf{u}_{\text{PGD}} - \mathbf{u}) \cdot (\mathbf{u}_{\text{PGD}} - \mathbf{u}) d\Omega d\mu}{\int_{\mathcal{I}_1} \int_{\Omega} \mathbf{u} \cdot \mathbf{u} d\Omega d\mu} \right)^{1/2}, \quad (26)$$

is studied. Figure 14 depicts the evolution of ε_{PGD} as a function of the number of PGD modes for three different degrees of approximation and four different meshes. The discontinuous lines in Figure 14 show the relative error of the

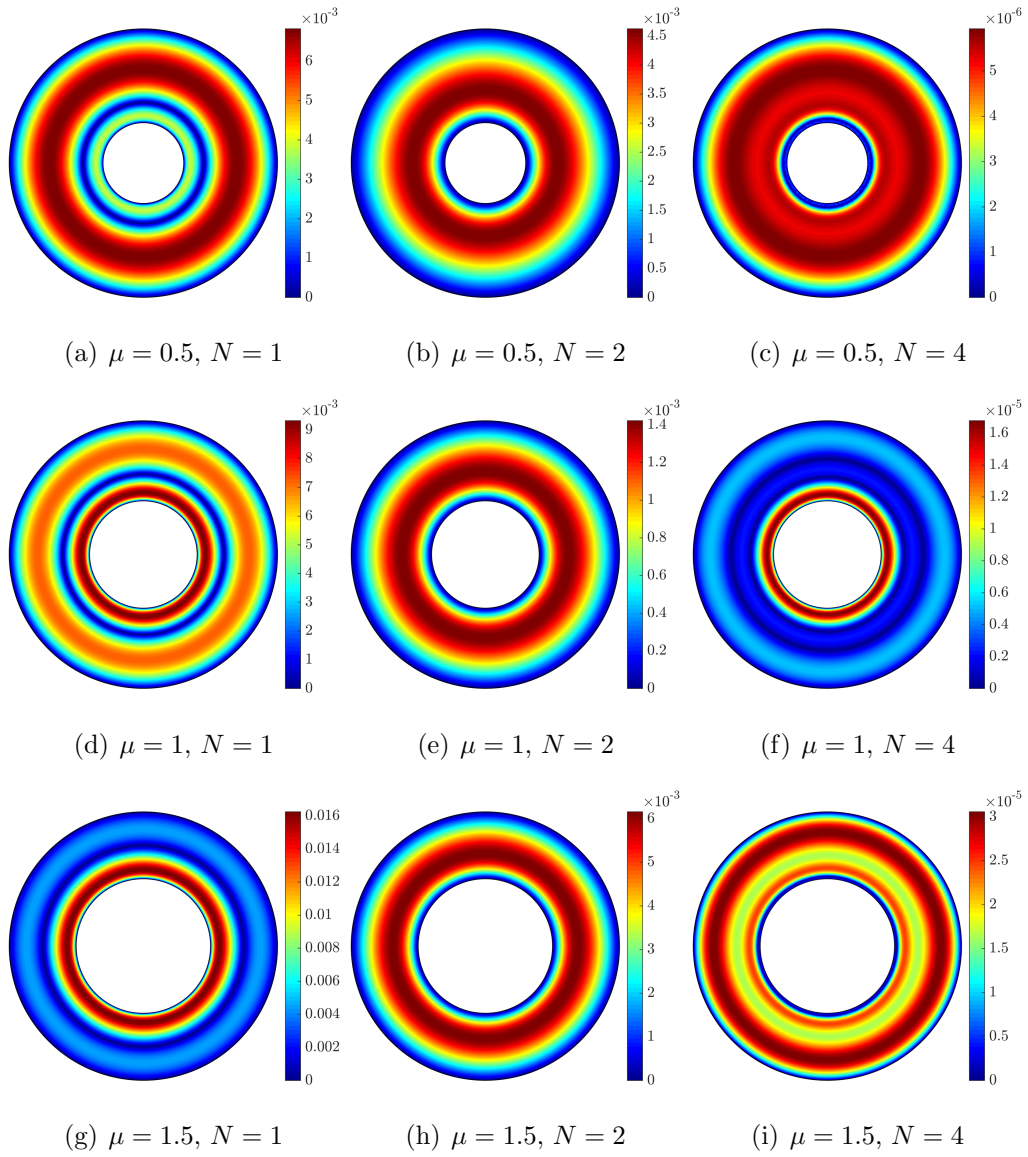


Figure 13: Absolute value of the error in the magnitude of the velocity computed with the proposed PGD approach using N modes for different values of the parameter μ . The computation has been performed on the mesh of Figure 3 (c), with 800 elements in the parametric dimension and a degree of approximation $k = 4$.

standard FEM solution measured in the $\mathcal{L}_2(\Omega \times \mathcal{I}_1)$ norm. The evaluation of the error of the FEM solution in the $\mathcal{L}_2(\Omega \times \mathcal{I}_1)$ norm requires the solution of as many FEM problems as the number of elements in the parametric di-

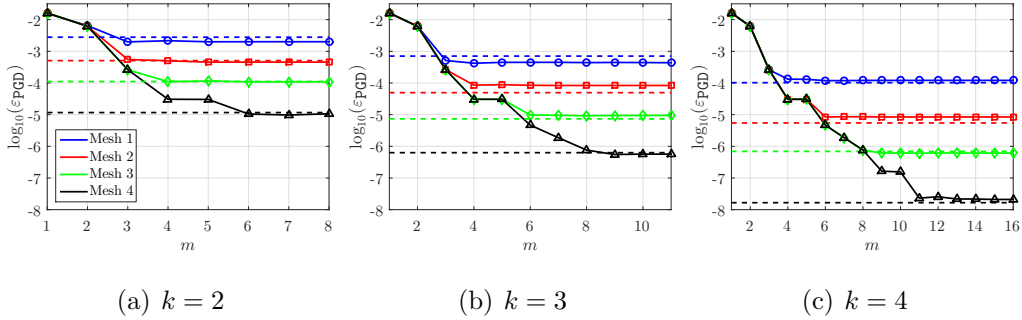


Figure 14: Error of the PGD solution as a function of the number of PGD modes using three different meshes for the solution of the Couette flow problem. The discontinuous line represents the error using standard FEM.

mension used by the PGD approach, multiplied by the number of integration points within each element.

The results clearly illustrate the increased level of accuracy as the number of modes is increased. In addition, the increased accuracy induced by the use of finer meshes or higher degrees of approximation can be observed. It worth noting that the PGD solution achieves its maximum accuracy (i.e. the accuracy of a standard FEM solution) using a low number of modes. For coarse meshes, the maximum accuracy of the PGD solution is achieved with three or four modes. For finer meshes, the PGD approach provides accurate results with only three or four modes and requires between six and 11 modes to reach the same level of error as the standard FEM solution.

Next, the accuracy of the PGD solution for different values of the parameter μ is studied. Figure 15 shows the difference between the PGD solution and the FEM solution, in logarithmic scale, as a function of the parameter μ and the number of PGD modes using different degrees of approximation. The results show that the accuracy of the PGD solution depends only weakly on the value of the parameter selected. For values of $\mu = 0$, corresponding to the undeformed configuration, the error is lower for a moderate number of

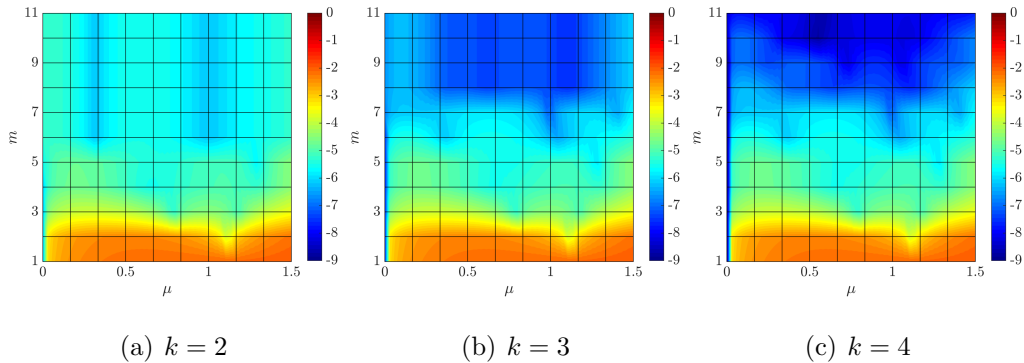


Figure 15: Difference between the PGD solution and the FEM solution, in logarithmic scale, as a function of the parameter μ and the number of PGD modes using different degrees of approximation.

PGD modes. This is expected because for $\mu = 0$ the displacement function \mathbf{d} is null and the Jacobian \mathbf{J}_μ is equal to the identity matrix. But for a large enough number of modes, the difference between the PGD solution and the FEM solution shows very little dependence on the value of μ , illustrating the robustness of the proposed PGD approach.

The last numerical study for the rotation Couette flow considers a mesh convergence analysis for the proposed PGD approach. The PGD solution is computed on the three meshes shown in Figure 3 using three different orders of approximation. Figure 16 shows the evolution of the relative error in the $\mathcal{L}_2(\Omega \times \mathcal{I}_1)$ norm, ε_{PGD} , as a function of the characteristic element size h . For each simulation the minimum number of PGD modes required to achieve the maximum accuracy is considered, as described earlier when presenting the results of Figure 14. The results show that, when enough modes are considered within the PGD framework, the error ε_{PGD} converges with near an optimal rate h^{k+1} .

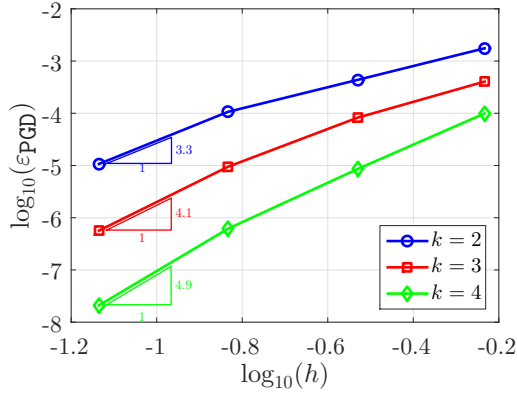


Figure 16: Relative error between the PGD and analytical solutions in the $\mathcal{L}_2(\Omega \times \mathcal{I}_1)$ norm as a function of the characteristic element size h for different degrees of approximation k .

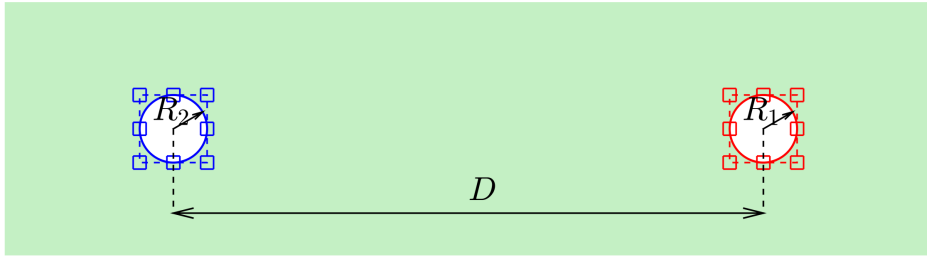


Figure 17: Detail of the computational domain for the solution of the Stokes flow around two circular cylinders.

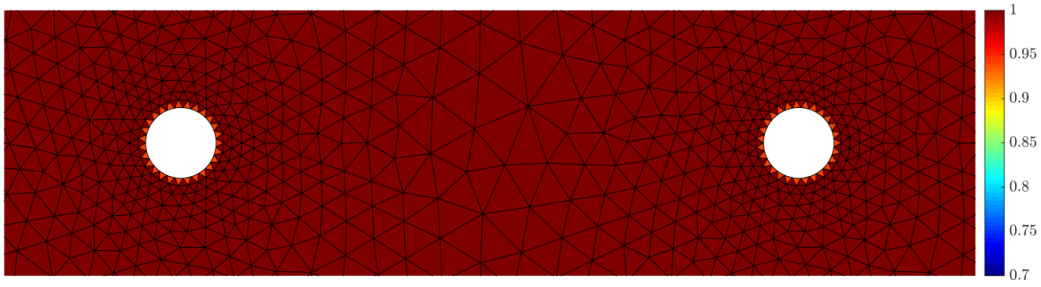
5.2. Stokes flow around two circular cylinders

The second example is inspired by the studies of [1, 36] on the analysis of micro-swimmers. The problem involves the computation of the Stokes flow in a rectangular channel of dimension 40×14 with two circular cylinders of radius R_1 and R_2 , where the distance between their centres is D , as represented in Figure 17. Slip boundary conditions are considered on the surface of the cylinders, a free slip boundary condition on the top and bottom boundaries, a imposed horizontal velocity of unit magnitude on the left boundary and a homogeneous Neumann boundary condition on the right part of the boundary.

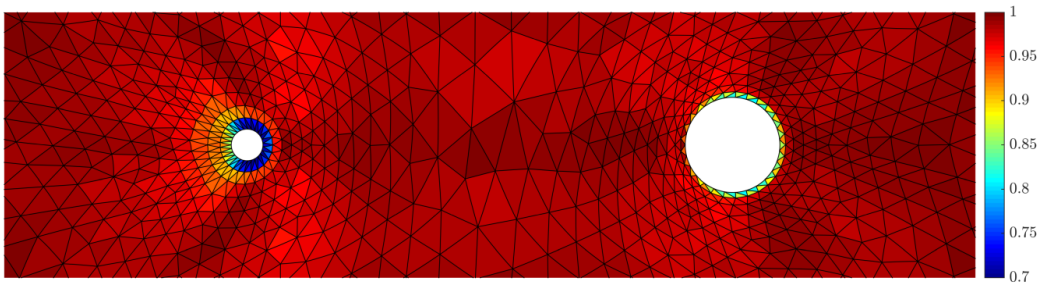
The geometric parameters considered are the radius of the cylinders and the distance between their centres. In addition, as done in the context of microswimmers, it is assumed that the variation of the two radius is linked so that $R_1^2 + R_2^2$ remains constant, that is the area enclosed by the two cylinders is constant. It is worth noting that the variation of these geometric parameters involve the variation of all the control points (i.e. 32 parameters) of the two NURBS describing the circles represented in Figure 17. However, as done in the previous example, it is possible to re-parametrise the motion in terms of only two parameters, $\mu_1 \in \mathcal{I}_1 = [-1, 1]$ which controls the radius of the cylinders and $\mu_2 \in \mathcal{I}_2 = [-1, 1]$ controlling the distance between the cylinders.

The reference configuration corresponds to $R_1 = R_2 = 0.8$ ($\mu_1 = 0$) and $D = 14$ ($\mu_2 = 0$). The minimum and maximum values for the radius of the cylinders are 0.3578 ($\mu_1 = -1$) and 1.0733 ($\mu_1 = 1$) which, given the link between R_1 and R_2 , correspond to the cases where the area of one of the circles is 90% and 10% respectively of the total area occupied by both circles. For the distance, the maximum and minimum values are 15.5 ($\mu_2 = -1$) and 12.5 ($\mu_2 = 1$) respectively.

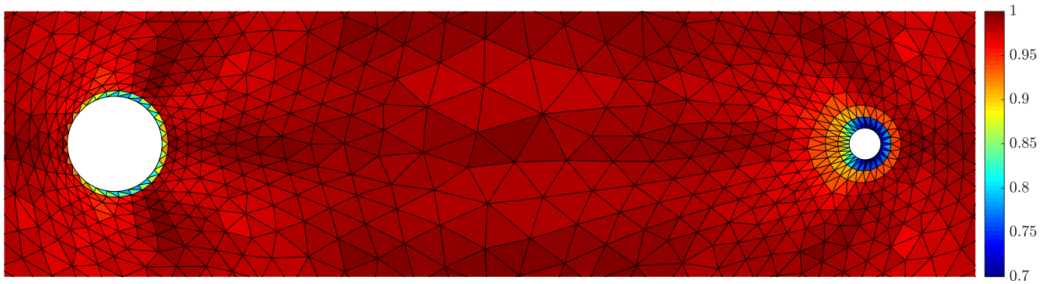
A computational mesh with 2,338 triangular elements is generated for the reference configuration. For a cubic degree of approximation (for geometry and velocity), Figure 18 shows the quality of the mesh for the reference configuration and for two deformed configurations corresponding to the extreme cases with $\mu_1 = \mu_2 = -1$ and $\mu_1 = \mu_2 = 1$. It can be observed that, even for the large deformations corresponding to the extreme cases of Figures 18 (b) and (c), the minimum quality (measured as the scaled Jacobian) is near 0.7, which is similar to the quality observed in the previous example for a simpler problem.



(a) $\mu_1 = \mu_2 = 0$



(b) $\mu_1 = \mu_2 = -1$



(c) $\mu_1 = \mu_2 = 1$

Figure 18: Computational mesh and quality for three values of the geometric parameters μ_1 and μ_2 .

Next, the separation of the matrix \mathbf{H}_μ , defined in Equation (6), is considered. As discussed previously \mathbf{H}_μ does not generally admit an exact separable expression and therefore a separable approximation is computed here via the higher-order PGD-projection described in [25].

Figure 19 shows the first six normalised spatial modes of the component

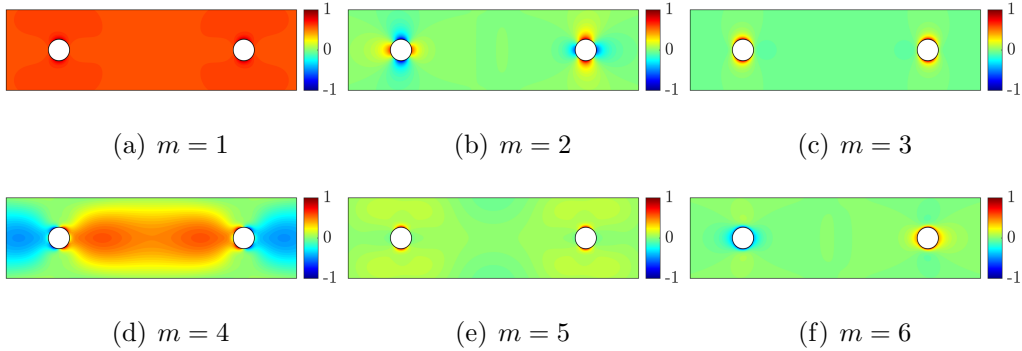


Figure 19: First six normalised spatial modes of $[\mathbf{H}_\mu]_{11}$ on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

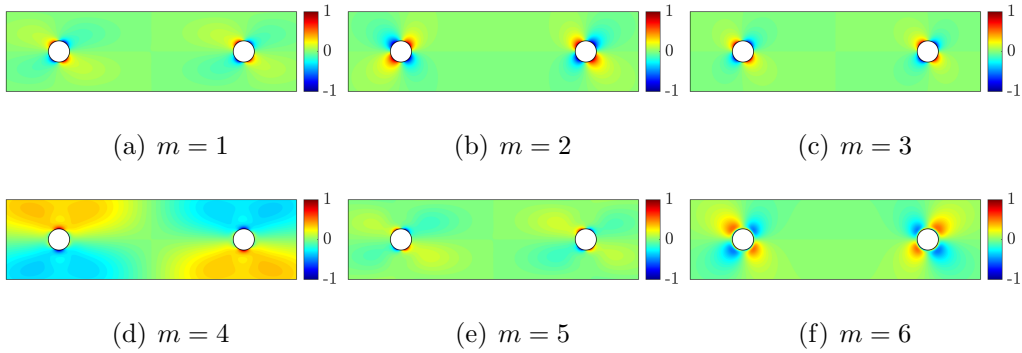


Figure 20: First six normalised spatial modes of $[\mathbf{H}_\mu]_{12}$ on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

$[\mathbf{H}_\mu]_{11}$. Similarly, Figures 20 and 21 show the first six normalised spatial modes of the component $[\mathbf{H}_\mu]_{12}$ and $[\mathbf{H}_\mu]_{22}$ respectively.

It is worth noting that some of the modes in Figure 19 resemble the modes obtained in the previous example and represented in Figure 7, whereas other modes display a completely different spatial variation. This indicates that the similar modes are the ones that carry information about the change of the radius of the cylinders whereas the different ones are related to the variation of the distance between the cylinders.

Contrary to the previous example, it is apparent that more modes are

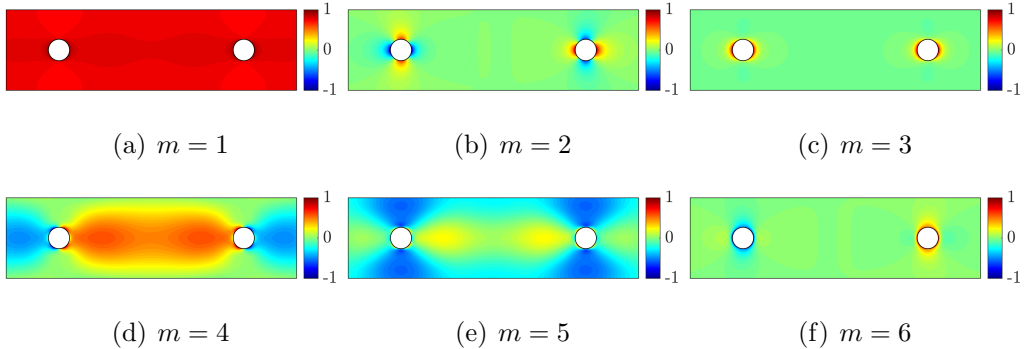


Figure 21: First six normalised spatial modes of $[\mathbf{H}_\mu]_{22}$ on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

necessary to describe the global behaviour of the matrix \mathbf{H}_μ . For instance, in the previous example only two modes were needed to describe the global information of the matrix \mathbf{H}_μ whereas now the fourth mode of all the components of \mathbf{H}_μ and the fifth mode of $[\mathbf{H}_\mu]_{22}$ contain global information. Also, contrary to the previous example, it is interesting to observe the different behaviour of the modes associated to the diagonal terms of the matrix \mathbf{H}_μ . This is due to the more complex motion induced by the geometric parameters, compared to the previous example.

The first eight normalised parametric modes of \mathbf{H}_μ are represented in Figure 22. It is worth recalling that the same parametric modes are associated to all the components of the matrix \mathbf{H}_μ . It is interesting to observe the similar qualitative behaviour of the parametric modes of μ_1 , represented in Figure 22 (a), and the parametric modes obtained in the previous example, represented in Figure 9. This is expected as in both cases these modes are related to the variation of the radius of a circle. The parametric modes associated to the distance between the cylinders, represented in Figure 22 (b), display a less oscillatory character than the modes associated to the variation of the radius of a circle.

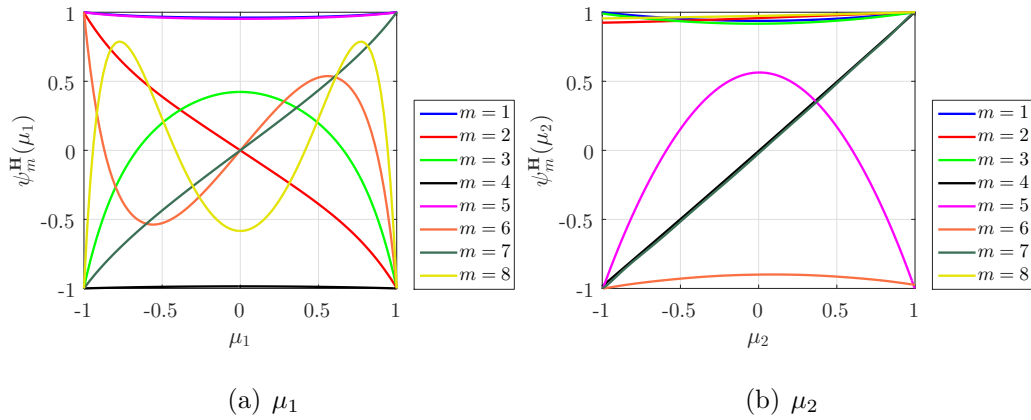


Figure 22: First eight parametric modes of \mathbf{H}_μ on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

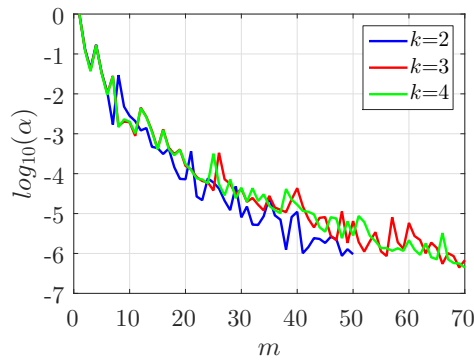


Figure 23: Evolution of the amplitude of the spatial modes α_m of the matrix \mathbf{H}_μ using different degrees of approximation.

To illustrate the increased complexity due to the introduction of two geometric parameters, Figure 23 shows the amplitude, α_m , corresponding to the mode m of the separation of \mathbf{H}_μ , computed as the product of the Euclidean norms of the spatial and parametric functions.

The amplitude, α_m , corresponding to the mode m of the separation of \mathbf{H}_μ , is computed as the product of the Euclidean norms of the spatial and parametric functions. Figure 10 shows the amplitudes of $[\mathbf{H}_\mu]_{11}$ using four different meshes and three different degrees of approximation. The results

show that, for this example, the number of terms required to obtain a separable approximation of the matrix \mathbf{H}_μ using the higher-order PGD-projection is significantly higher than in the previous example. To provide a decrease in the amplitude α_m of six orders of magnitude the number of modes vary from 50 and 63 for an order of approximation ranging from $k = 2$ to $k = 4$, showing again that the number of modes is not highly dependent upon the spatial discretisation.

Using the separation of the matrix \mathbf{H}_μ , the Stokes flow around the two cylinders is computed with the proposed approach to obtain the generalised solution. The first six normalised spatial modes of the magnitude of the velocity and the pressure fields are shown in Figure 24. The simulation was performed using the mesh of Figure 18 (a), 25 equally-spaced elements in $\mathcal{I}_1 = [-1, 1]$, 45 equally-spaced elements in $\mathcal{I}_2 = [-1, 1]$ and a degree of approximation $k = 4$.

The first eight normalised parametric modes associated to the spatial modes of Figure 24 are represented in Figure 25. It is worth recalling that the same parametric mode is associated to all the components of the velocity and the pressure fields.

Contrary to the previous example, it can be observed that the spatial modes for the velocity and the pressure do not resemble the spatial modes of the matrix \mathbf{H}_μ , illustrating the increased complexity of the current problem. In addition, it is worth noting that the parametric modes associated to the first parameter μ_1 shows a more oscillatory character than μ_2 when the separation of the matrix \mathbf{H}_μ is studied (as shown in Figure 22), whereas the the second parameter μ_2 shows a more oscillatory character than μ_1 when the separation of the velocity and pressure is considered (as shown in Figure 25). This indicates that the flow around the two cylinders changes slowly when

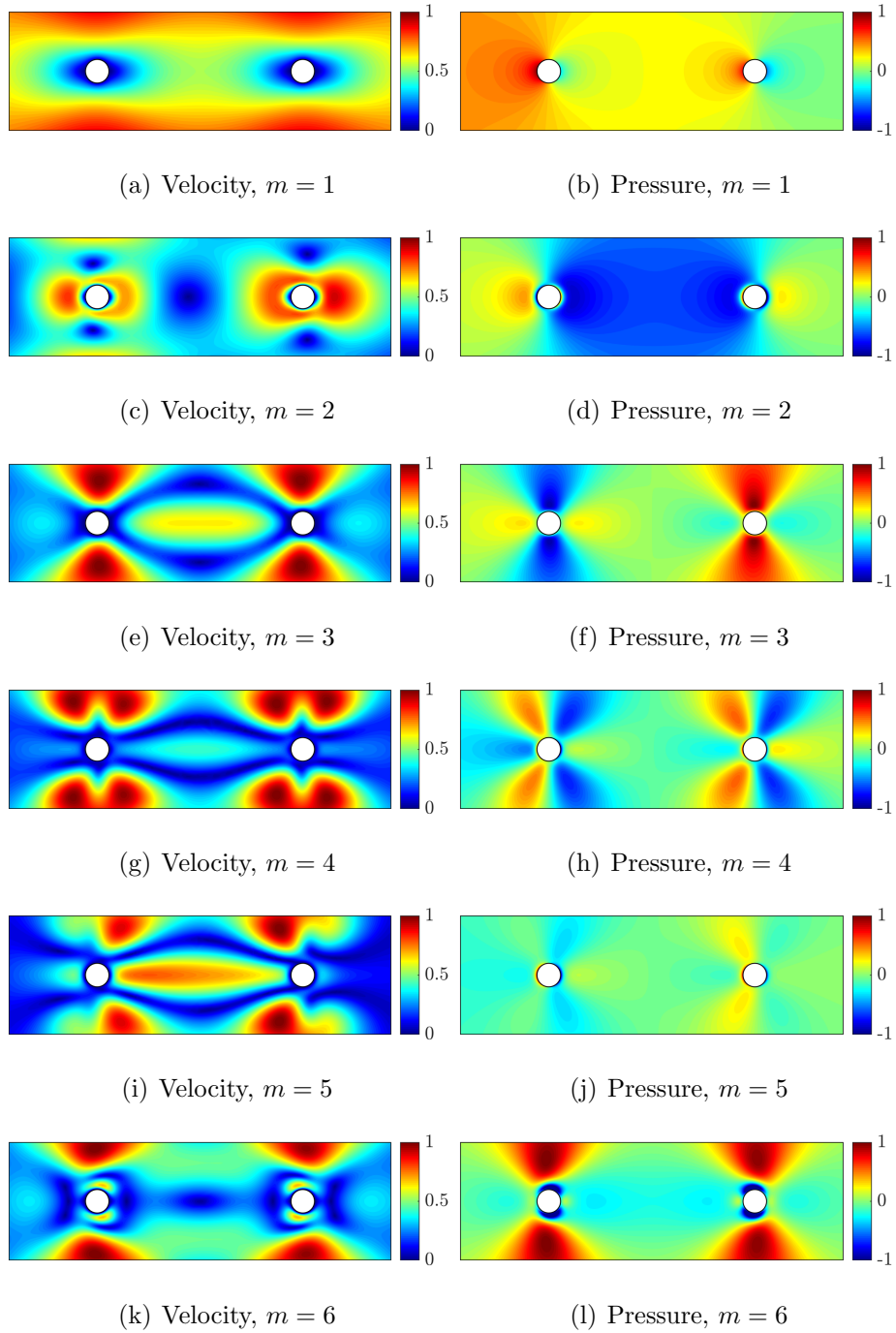


Figure 24: First six normalised spatial modes of the magnitude of the velocity (left) and pressure (right) computed on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

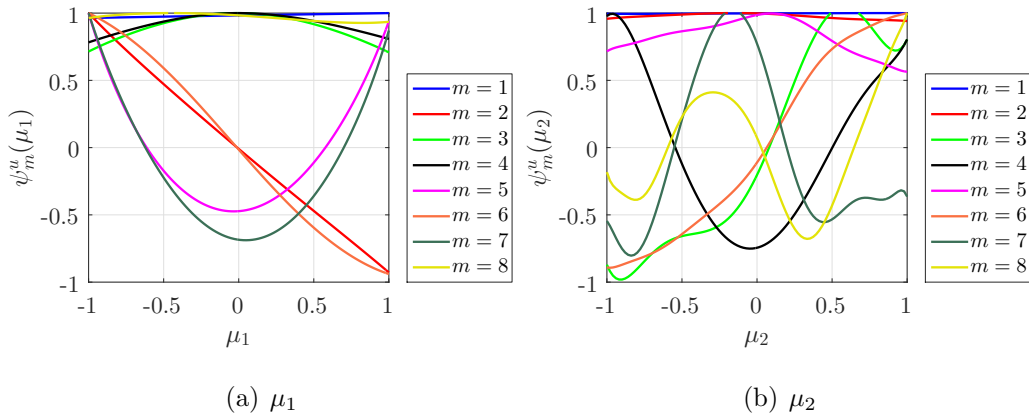


Figure 25: First eight parametric modes of the Stokes flow around two circular cylinders on the mesh of Figure 18 (a) with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

the radius of the cylinders is varied whereas the flow changes more rapidly when the distance between the cylinders is varied.

Next, the solutions obtained with the proposed PGD framework are represented for different values of the geometric parameters. Figure 26 shows the magnitude of the velocity and the pressure fields for the three configurations shown in Figure 18.

To illustrate the accuracy of the proposed approach, Figure 27 shows the difference between the standard FE solution and the PGD solution for the magnitude of the velocity and the pressure corresponding to the three configurations shown in Figure 18. For all the configurations, the results obtained with the proposed PGD framework are in excellent agreement with the results obtained using a standard FE solver. For the first configuration displayed in Figure 27, corresponding to $\mu_1 = \mu_2 = 0$, the difference between the magnitude of the velocity field using FEM and the PGD in the $\mathcal{L}_2(\Omega^\mu)$ norm is 0.0143 and the difference between the pressure field using FEM and the PGD in the $\mathcal{L}_2(\Omega^\mu)$ norm is 0.0277. For the second configuration of Figure 27, corresponding to $\mu_1 = \mu_2 = -1$, the difference between the magnitude of the

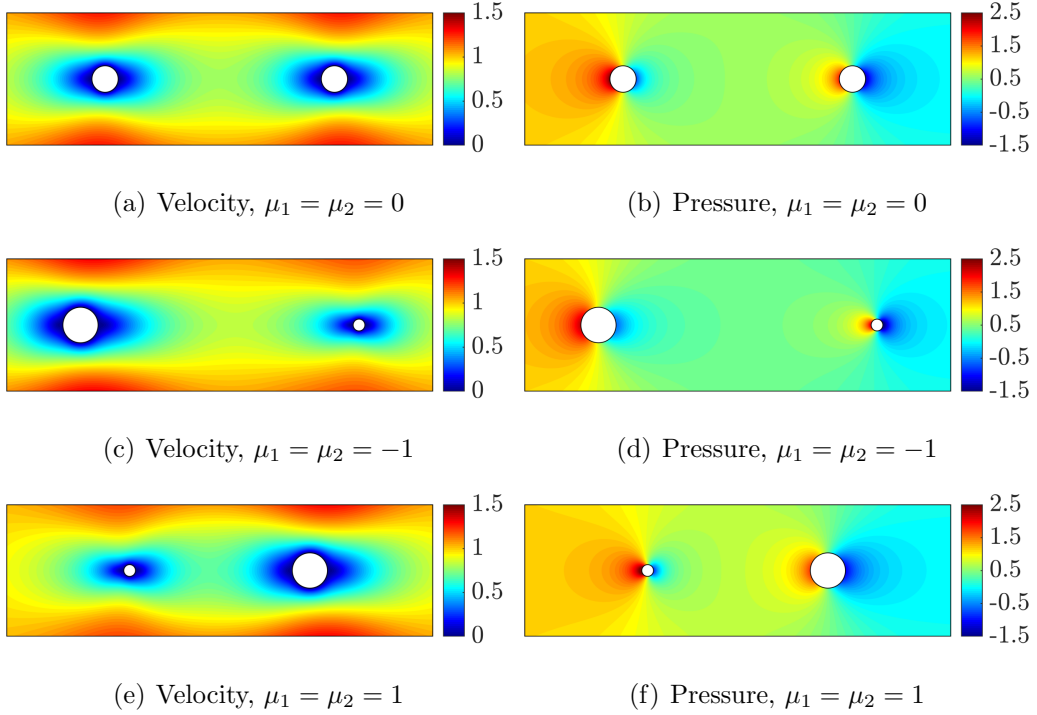


Figure 26: Magnitude of the velocity (left) and pressure (right) computed on the three configurations shown in Figure 18 with 25 elements in \mathcal{I}_1 , 45 elements in \mathcal{I}_2 and with $k = 4$.

velocity and pressure fields using FEM and the PGD in the $\mathcal{L}_2(\Omega^\mu)$ norm are 0.0267 and 0.0592 respectively. Finally, for the last configuration of Figure 27, corresponding to $\mu_1 = \mu_2 = 1$, the difference between the magnitude of the velocity and pressure fields using FEM and the PGD in the $\mathcal{L}_2(\Omega^\mu)$ norm are 0.0218 and 0.0787 respectively. It is worth emphasising that the accurate results obtained, with differences between PGD and standard FE simulations lower than 8% for the whole range of geometric configurations, have been obtained with very coarse meshes in the parametric spaces, only 25 elements in \mathcal{I}_1 and 45 elements in \mathcal{I}_2 .

It is also worth recalling that the higher accuracy obtained in the velocity field, compared to the pressure, is due to the use of a higher order

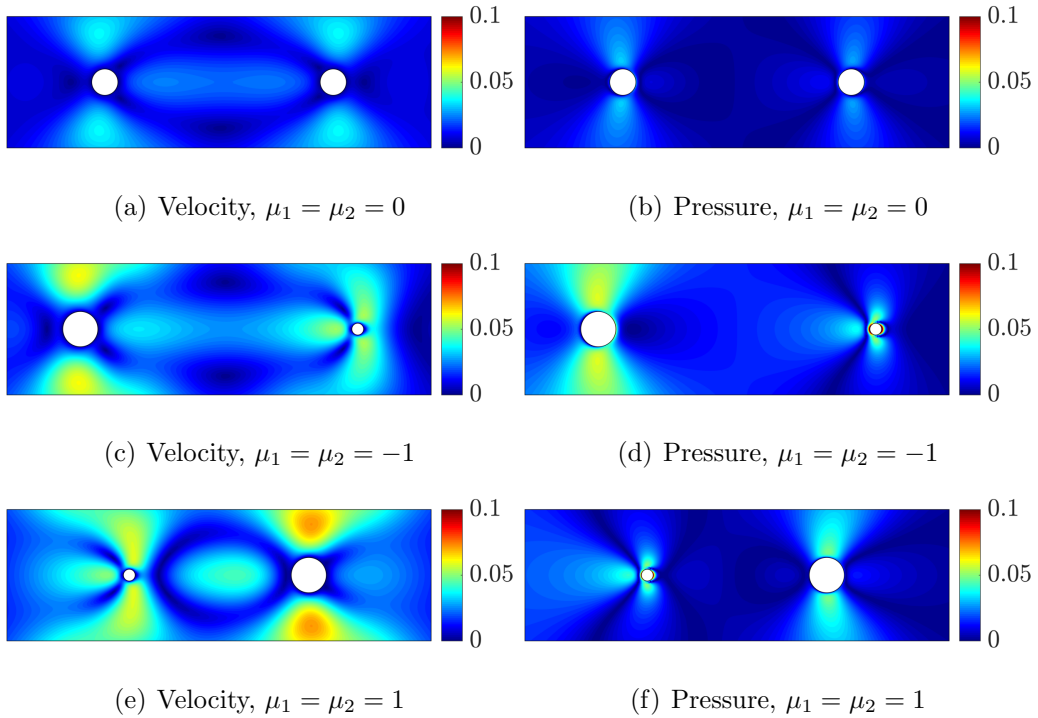


Figure 27: Difference between the standard FE solution and the PGD solution for the magnitude of the velocity (left) and pressure (right) computed on the three configurations shown in Figure 26.

approximation for the velocity to ensure satisfaction of the LBB condition.

5.3. Stokes flow around an arbitrarily shaped vesicle

The last example is inspired by the study of vesicles suspended in a viscous flow presented in [37]. The characterisation of such flows is of interest in many biomechanical applications and the simulations often require the computation to be performed for a large variety of geometric configurations. This example is used to demonstrate the applicability of the proposed technique in three dimensions by using three geometric parameters that lead to substantial variations in the geometric model.

The problem considered here involves the simulation of the Stokes flow around an arbitrarily shaped vesicle in a channel of dimension $10 \times 10 \times 20$.

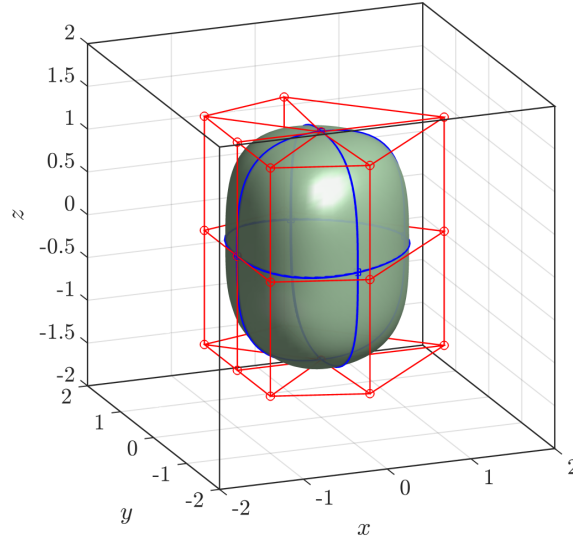


Figure 28: Geometric model of a vesicle, showing the control net and the breaklines.

Slip boundary conditions are considered on the surface of the vesicle, an imposed vertical velocity of unit magnitude in the inflow part of the boundary ($z = -10$), a homogeneous Neumann boundary condition on the outflow part of the boundary ($z = 10$) and free slip boundary conditions on the remaining parts of the boundary.

The generic vesicle considered is modelled using a single degenerate cubic NURBS with 20 control points and four patches, as illustrated in Figure 28. The control points are arranged in three groups, according to their z coordinate being positive, zero or negative. The geometric parametrisation considers the motion of all the three groups independently by using three parameters, μ_1 , μ_2 and μ_3 respectively. Each parameter is used to vary the radial position of the control points whilst maintaining its z coordinate. For the control points with positive z coordinate the radial coordinate is given by $r + 3\mu_1/4$ with $\mu_1 \in [-1, 1]$. Similarly for the control points with zero or positive z coordinate the radial coordinate is $r + 3\mu_2/4$ and $r + 3\mu_3/4$

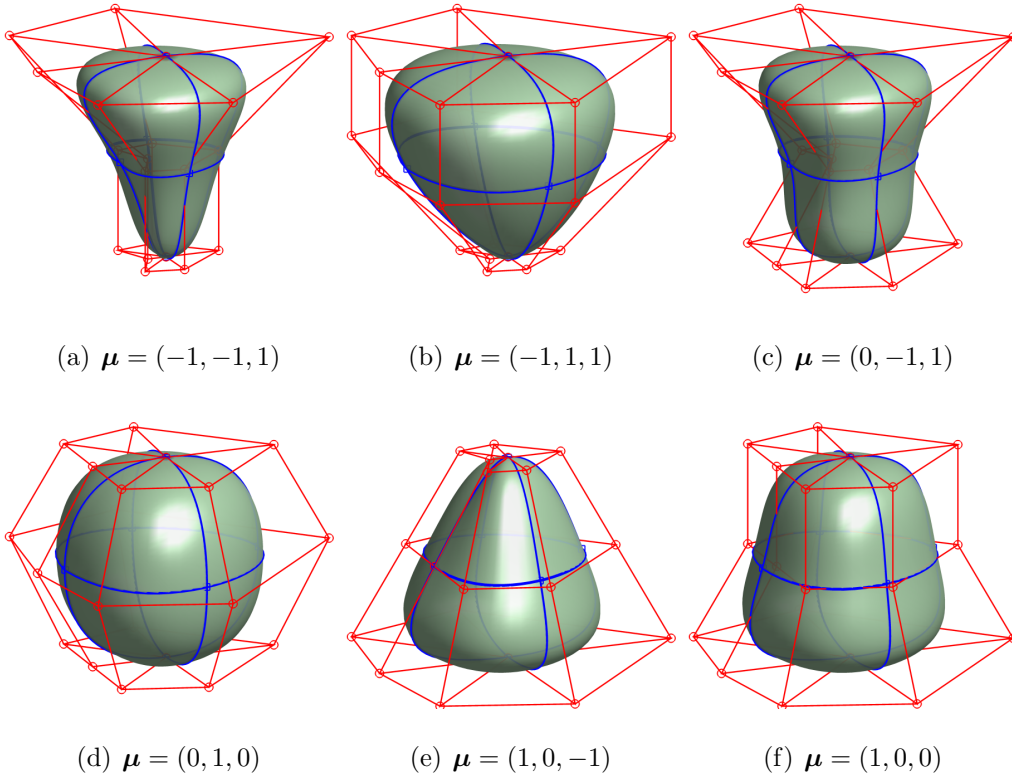


Figure 29: Geometric model of six different vesicle obtained by deforming the reference configuration of Figure 28.

respectively, with $\mu_2 \in [-1, 1]$ and $\mu_3 \in [-1, 1]$.

The reference configuration, corresponding to $\boldsymbol{\mu} = (\mu_1, \mu_2, \mu_3) = (0, 0, 0)$, is shown in Figure 28.

The generalised PGD solution is computed using a tetrahedral mesh with 6,712 elements, 10 equally-spaced elements in each parametric interval $\mathcal{I}_1 = \mathcal{I}_2 = \mathcal{I}_3 = [-1, 1]$ and a degree of approximation $k = 4$.

To illustrate the variation in the geometry induced by the selected geometric parameters, Figure 29 shows six different geometric configurations using different values for the geometric parameters. It can be observed that the variations of the control points considered lead to substantial changes in the geometric model. These variations induce a deformation of the mesh

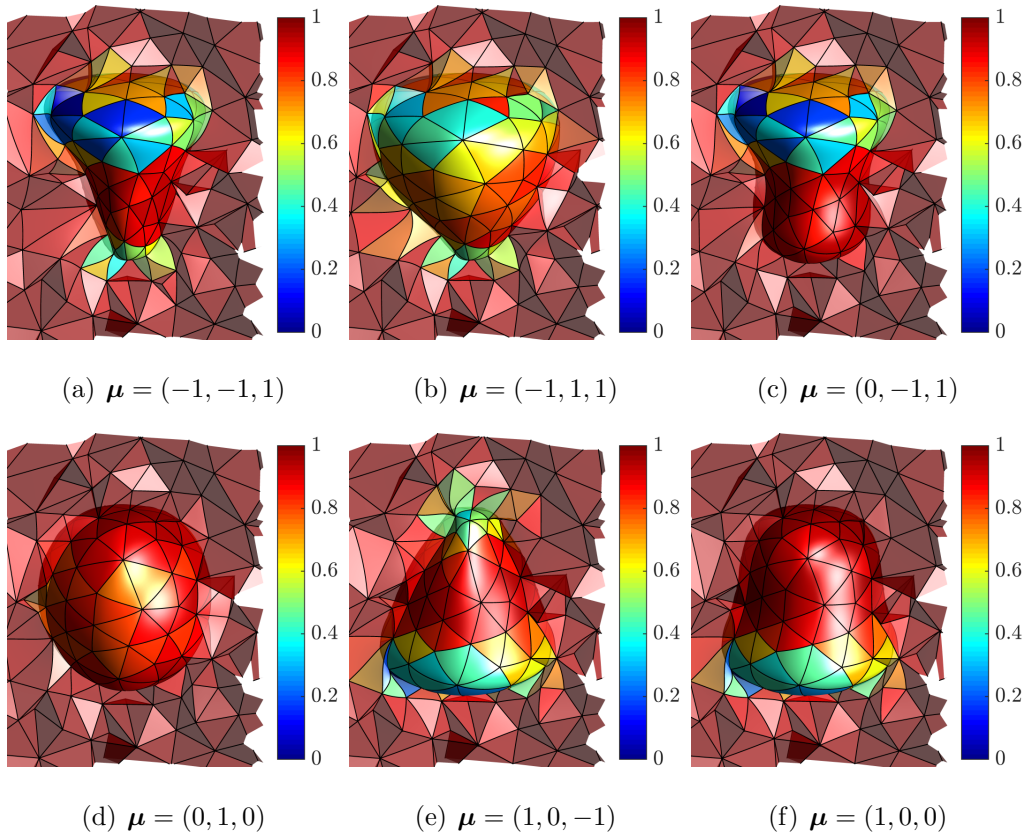


Figure 30: Detail of the six deformed high-order meshes corresponding to the configurations shown in Figure 29.

generated for the reference configuration, computed in a separated form by employing the strategy described in Section 4.3. Figure 30 shows six deformed meshes near the vesicle that correspond to the six variations of the geometric parameters depicted in Figure 29. The colours represent the quality of the elements, measured as the scaled Jacobian.

Following the strategy described in this work, the matrix $\mathbf{H}_{\boldsymbol{\mu}}$ is separated first using the higher-order PGD-projection. To provide a decrease in the amplitude α_m of five orders of magnitude, 50 modes are required in this example. It is worth noting that this is almost the same number of modes required for the previous two dimensional example to provide the same degree

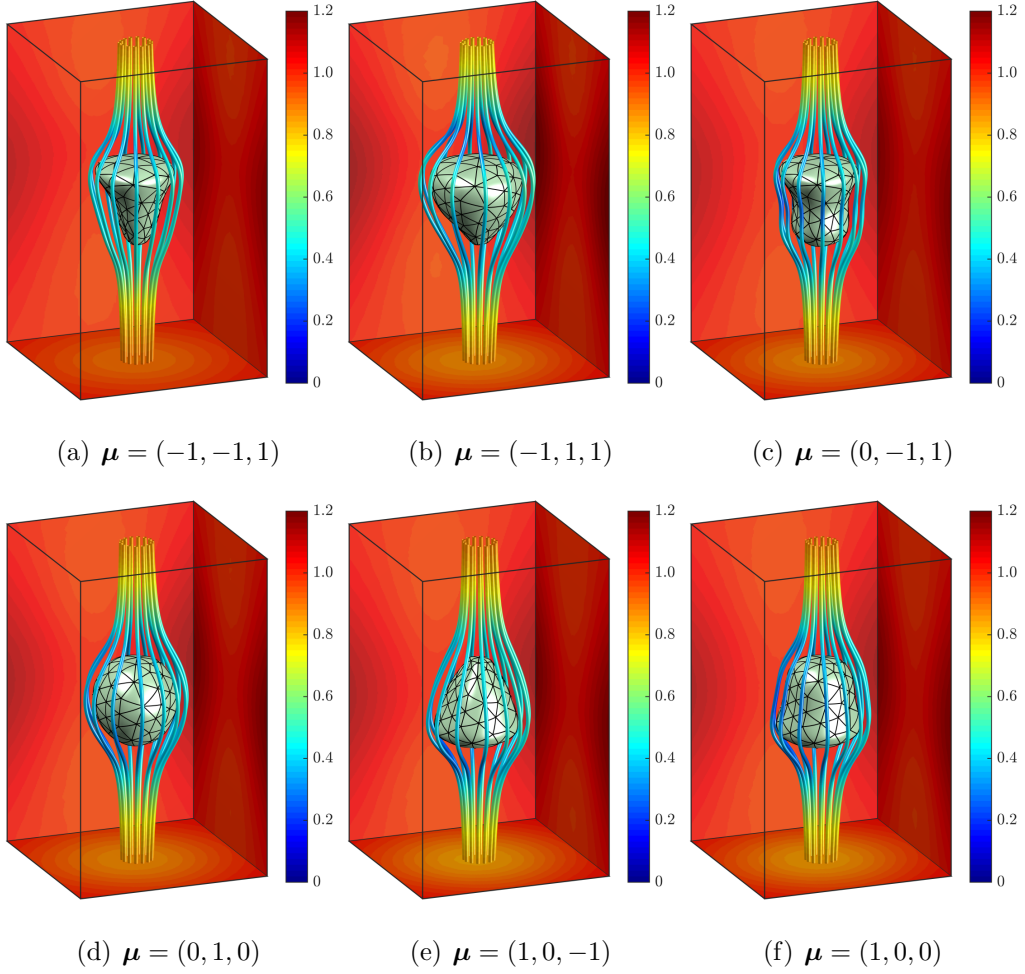


Figure 31: Magnitude of the velocity and isolines for the six configurations shown in Figure 29.

in the amplitude of the modes of the matrix \mathbf{H}_μ .

Using the separation of the matrix \mathbf{H}_μ , the Stokes flow around the vesicle is computed with the proposed approach to obtain the generalised solution. A total of 160 modes are computed, being the amplitude of the last mode three orders of magnitude lower than the amplitude of the first mode.

Figures 31 and 32 show the velocity and pressure fields for the six geometric configurations shown in Figure 29. These configurations are obtained

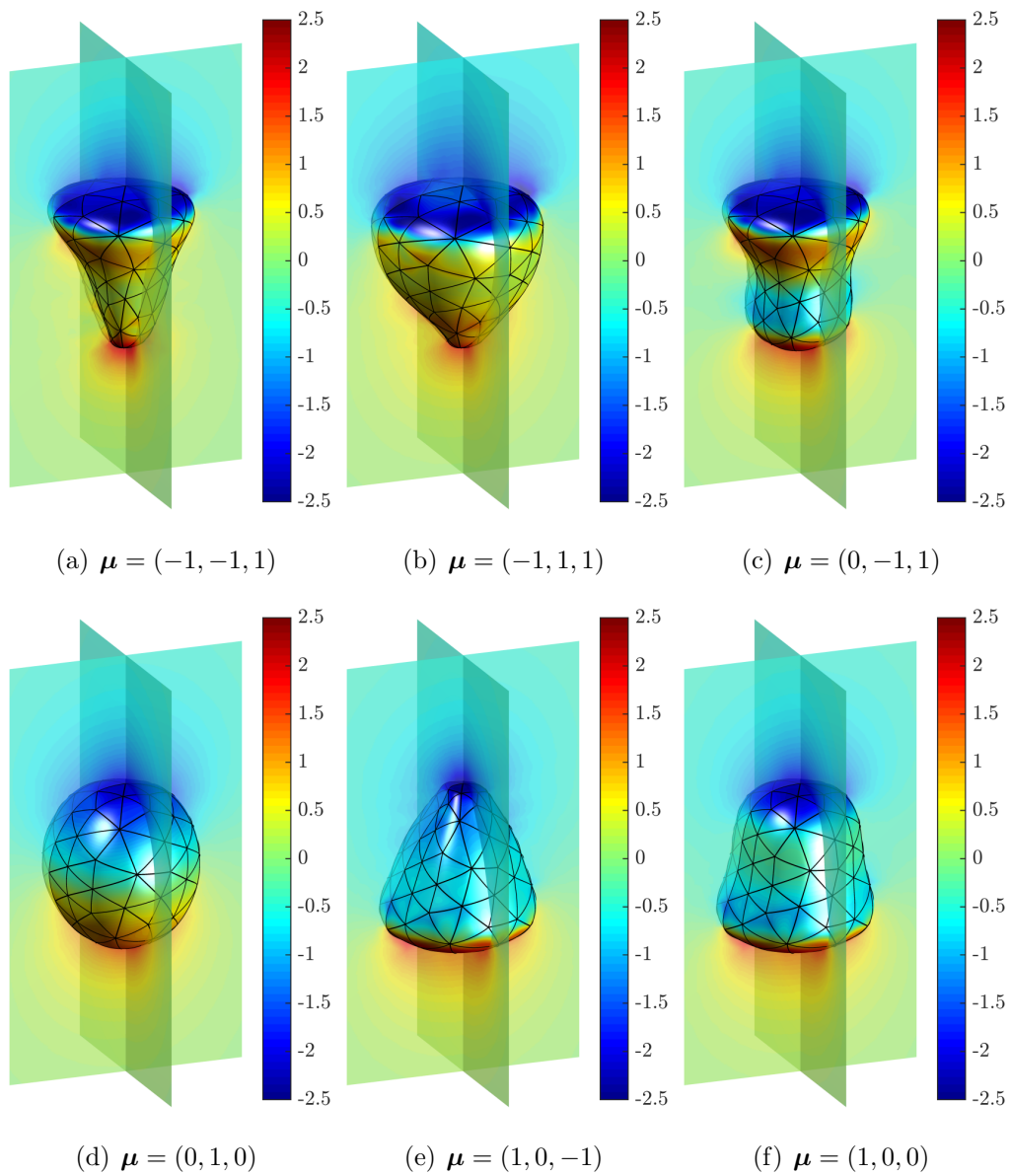


Figure 32: Pressure field for the six configurations shown in Figure 29.

in the on-line phase as particularisations of the gearlised, six dimensional, PGD solution. Both the velocity and pressure fields illustrate the ability of the proposed PGD approach to capture significant changes in the flow field induced by geometric variations of the CAD model.

To quantify the accuracy of the proposed technique in a more complex problem in three dimensions and with three geometric parameters, the particularised solutions for the velocity and pressure are compared to standard finite element computations performed on the deformed configurations for the six cases displayed in Figures 31 and 32. For the velocity field, the difference between the PGD and the finite element solution, measured in the $\mathcal{L}_2(\Omega)$ norm, is 0.7%, 0.8%, 0.8%, 0.4%, 0.6% and 0.7% for the six cases shown in Figure 31 respectively. Similarly, for the pressure field, the difference between the PGD and the finite element solution, measured in the $\mathcal{L}_2(\Omega)$ norm, is 7%, 7%, 7%, 3%, 5% and 6% for the six cases shown in Figure 32 respectively. It is worth emphasising that the higher accuracy observed in the velocity is due to the use of a higher interpolation degree for the velocity field, compared to the pressure field, in order to satisfy the LBB conditions.

6. Concluding remarks

A computational framework for the computation of off-line solutions for a set of parameters describing the geometry of a domain has been presented. The proposed approach considers as parameters of the generalised PGD solution the control points of the CAD boundary representation of the computational domain. A mapping between a reference configuration and the current configuration is proposed by interpreting the geometric changes as a displacement field derived from a linear elastic problem. The key aspect of the proposed approach is that the displacement field is explicitly written in a separable form. This approach enables the incorporation of the PGD rationale in a high-order finite element context.

The potential of the proposed approach is shown for a variety of problems involving the solution of the Stokes equation in geometrically parametrised

domains, both in two and three dimensions. The problems, of increasing difficulty show the optimal approximation properties of the method and its ability to accurately capture the flow features for significant changes of the geometric model. For the most complex problem considered in this work, the generalised solution computed with three geometric parameters in three dimensions show good agreement when compared to a standard finite element computation, with errors in the velocity field lower than 1% and errors in the pressure field below 8%.

Next, the necessary concepts about NURBS curves are briefly recalled and the proposed strategy to build a geometric mapping \mathcal{M}_μ that can be written in the separated form (7) is presented in detail.

Appendix A. Key concepts on NURBS

Appendix A.1. NURBS curves

A q th-degree NURBS curve is a piecewise rational function defined in parametric form as

$$\mathbf{C}(\lambda) = \sum_{i=0}^{\mathbf{n}_{\text{cp}}} \mathbf{B}_i R_i(\lambda) \quad \lambda \in [0, 1] \quad (\text{A.1})$$

where $\{\mathbf{B}_i\}$ are the coordinates of the $\mathbf{n}_{\text{cp}} + 1$ *control points* (forming the *control polygon*) and $\{R_i(\lambda)\}$ are rational basis functions defined as

$$R_i(\lambda) = \nu_i C_i^q(\lambda) / \left(\sum_{i=0}^{\mathbf{n}_{\text{cp}}} \nu_i C_i^q(\lambda) \right).$$

In the above expression $\{\nu_i\}$ are the control weights associated to the control points and $\{C_i^q(\lambda)\}$ are the normalized B-spline basis functions of

degree q , which are defined recursively by

$$C_i^0(\lambda) = \begin{cases} 1 & \text{if } \lambda \in [\lambda_i, \lambda_{i+1}) \\ 0 & \text{elsewhere} \end{cases}$$

$$C_i^k(\lambda) = \frac{\lambda - \lambda_i}{\lambda_{i+k} - \lambda_i} C_i^{k-1}(\lambda) + \frac{\lambda_{i+k+1} - \lambda}{\lambda_{i+k+1} - \lambda_{i+1}} C_{i+1}^{k-1}(\lambda)$$

for $k = 1, \dots, q$ and where λ_i (for $i = 0, \dots, \mathbf{n}_k$) are the *knots* or *breakpoints*, which are assumed ordered $0 \leq \lambda_i \leq \lambda_{i+1} \leq 1$. They form the so-called *knot vector*,

$$\Lambda = \{\underbrace{0, \dots, 0}_{q+1}, \lambda_{q+1}, \dots, \lambda_{\mathbf{n}_k - q - 1}, \underbrace{1, \dots, 1}_{q+1}\},$$

which uniquely describes the B-spline basis functions. The number of control points, $\mathbf{n}_{cp} + 1$, and knots, $\mathbf{n}_k + 1$, are related to the degree of the parametrisation, q , by the relation $\mathbf{n}_k = \mathbf{n}_{cp} + q + 1$, see [28] for more details.

Figure A.33 shows an example of a two dimensional domain Ω^μ where the boundary is described by five NURBS curves. The curve \mathbf{C}_3 in Figure A.33 (b) is represented in Figure A.34 with the corresponding control polygon, formed by six control points and the breakpoints. The knot vector for this curve is given by

$$\Lambda = \{0, 0, 0, 1/3, 2/3, 1, 1, 1\}.$$

Appendix A.2. NURBS surfaces

A NURBS surface of degree q in λ and degree r in κ , is a piecewise rational function defined in parametric form as

$$\mathbf{S}(\lambda, \kappa) = \sum_{i=0}^{\mathbf{n}_{cp}} \sum_{l=0}^{\mathbf{m}_{cp}} \mathbf{B}_{il} R_{i,l}(\lambda, \kappa) \quad 0 \leq \lambda, \kappa \leq 1,$$

where $\{\mathbf{B}_{il}\}$ are the coordinates of the $(\mathbf{n}_{cp} + 1)(\mathbf{m}_{cp} + 1)$ *control points* (defining the *control net*) and $\{R_{i,l}\}$ are rational basis functions defined as

$$R_{i,l}(\lambda, \kappa) = \nu_{il} S_{i,l}^{q,r}(\lambda, \kappa) / \left(\sum_{i=0}^{\mathbf{n}_{cp}} \sum_{l=0}^{\mathbf{m}_{cp}} \nu_{il} S_{i,l}^{q,r}(\lambda, \kappa) \right).$$

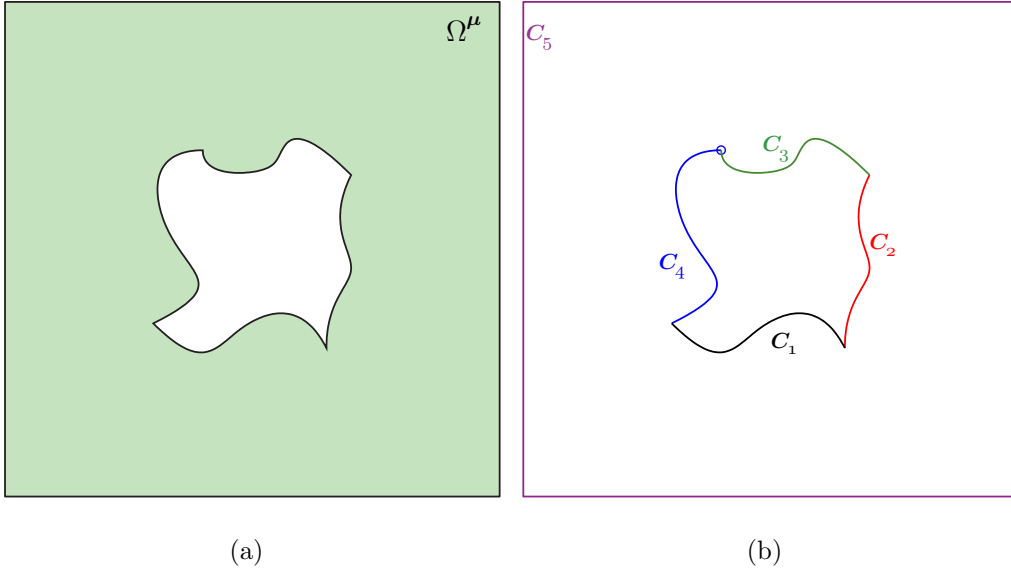


Figure A.33: (a) Domain Ω^μ and (b) NURBS curves describing the boundary of Ω^μ , where each colour represents a different curve.

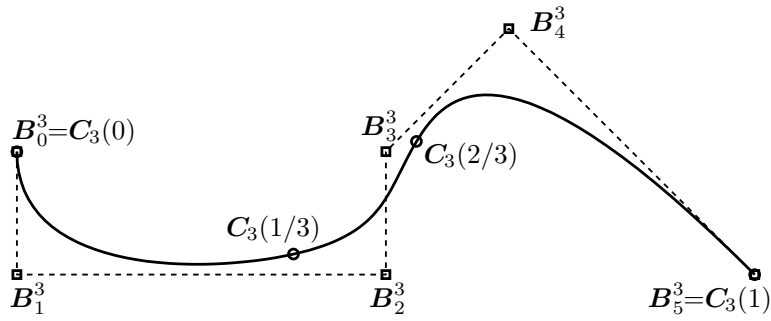


Figure A.34: NURBS curve (solid line), control points (denoted by \square), control polygon (dashed line) and breakpoints (denoted by \circ).

In the above expression $\{\nu_{il}\}$ are the control weights associated to the control points and $\{S_{i,l}^{q,r}(\lambda, \kappa)\}$ are the 2D B-spline basis functions of degree q in λ and r in κ . Each 2D B-Spline basis function is defined as a tensor product of 1D basis functions, that is

$$S_{i,l}^{q,r}(\lambda, \kappa) := C_i^q(\lambda)C_l^r(\kappa).$$

Appendix B. Integration within a CAD environment in 3D

In three dimensions, the boundary of the parameterized domain, $\partial\Omega^\mu$, is assumed to be described by using a set NURBS surfaces $\{\mathbf{S}_j^\mu\}_{j=1,\dots,M}$, being M the total number of surfaces, namely

$$\partial\Omega^\mu = \bigcup_{j=1}^M \mathbf{S}_j^\mu([0, 1]^2).$$

Next, the necessary concepts about NURBS surfaces are briefly recalled and the minimum changes that are required to extend the technique presented in Section 4 to three dimensional domains are detailed.

Appendix B.1. Geometric parameters

The geometric parameters $\boldsymbol{\mu} \in \mathcal{I} \subset \mathbb{R}^{\text{npa}}$ are defined as the variations of the original coordinates of the control points of the NURBS surfaces describing the boundary. More precisely, for each NURBS curve \mathbf{S}_j , with $j = 1, \dots, M$, having $(\mathbf{n}_{\text{cp}}^j + 1)(\mathbf{m}_{\text{cp}}^j + 1)$ control points, the undisturbed boundary is characterised by the coordinates of the control points: \mathbf{B}_{il}^j , for $i = 0, \dots, \mathbf{n}_{\text{cp}}^j$ and $l = 0, \dots, \mathbf{m}_{\text{cp}}^j$. The boundary of the distorted domain, $\partial\Omega^\mu$, is defined by the position of the displaced control points, namely $\mathbf{B}_{il}^j + \delta\mathbf{B}_{il}^j$. The displacement range for each control point is characterised by

$$\delta\mathbf{B}_{il}^{j} \in \mathcal{I}_{il}^j = [\underline{\delta x}_{il}^j, \overline{\delta x}_{il}^j] \times [\underline{\delta y}_{il}^j, \overline{\delta y}_{il}^j] \times [\underline{\delta z}_{il}^j, \overline{\delta z}_{il}^j].$$

In fact, each displacement of a control point on the j -th NURBS, $\delta\mathbf{B}_{il}^j$, might depend upon the parameters and can be written as

$$\delta\mathbf{B}_{il}^j = \mu_1^{il,j} \mathbf{e}_1 + \mu_2^{il,j} \mathbf{e}_2 + \mu_3^{il,j} \mathbf{e}_3, \quad (\text{B.1})$$

where \mathbf{e}_i , for $i = 1, 2, 3$, are the unit coordinate vectors. Then $\boldsymbol{\mu} \in \mathcal{I} := \mathcal{I}^1 \times \mathcal{I}^2 \times \dots \times \mathcal{I}^M$, where $\mathcal{I}^j = \mathcal{I}_1^j \times \mathcal{I}_2^j \times \dots \times \mathcal{I}_{(\mathbf{n}_{\text{cp}}^j+1)(\mathbf{m}_{\text{cp}}^j+1)}^j$ is the range of variation of the coordinates of the control points of the curve \mathbf{S}_j .

Appendix B.2. Separated representation of the boundary displacement

The variation of a control point \mathbf{B}_{il}^j of a NURBS surface \mathbf{S}_j , namely $\delta\mathbf{B}_{il}^j$, changes the definition of the original curve only in the support of the basis function R_{il}^j , given by the subspace of the parametric space $[\lambda_i, \lambda_{i+q^j+1}] \times [\kappa_l, \kappa_{l+r^j+1}]$. The modified NURBS surface is parametrised by

$$\mathbf{S}_j^\mu(\lambda, \kappa) = \sum_{i=0}^{n_{cp}^j} \sum_{l=0}^{m_{cp}^j} (\mathbf{B}_{il}^j + \delta\mathbf{B}_{il}^j) R_{il}^j(\lambda, \kappa) \quad 0 \leq \lambda, \kappa \leq 1.$$

As in the two dimensional case, the displacement of the boundary mesh node $\mathbf{X}_k = \mathbf{S}_j(\lambda_k, \kappa_k)$ that belongs to the NURBS curve \mathbf{S}_j can be written in separated form as

$$\mathbf{d}_j^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{i=0}^{n_{cp}^j} \sum_{l=0}^{m_{cp}^j} \sum_{s=1}^{n_{sd}} \mu_s^{i,j} \mathbf{e}_s R_{il}^j(\lambda_k, \kappa_k),$$

where the dependence of the displacements of the control points in terms of the parameters described in Equation (B.1) has been used. Moreover, since the NURBS parameterd (λ_k, κ_k) is only dependent on the spatial coordinates \mathbf{X}_k , and not on the geometric parameters $\boldsymbol{\mu}$, the previous equation can be written as,

$$\mathbf{d}^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{j=1}^M \mathbf{d}_j^\partial(\mathbf{X}_k, \boldsymbol{\mu}) = \sum_{j=1}^M \sum_{i=0}^{n_{cp}^j} \sum_{l=0}^{m_{cp}^j} \sum_{s=1}^{n_{sd}} \mu_s^{i,j} \mathbf{e}_s R_{il}^j(\mathbf{S}_j^{-1}(\mathbf{X}_k)),$$

which characterises the displacement of the boundary nodes and has the desired separated form, as in the two dimensional case given by Equation (19).

Appendix B.3. Separated representation of the geometric mapping

The strategy to obtain a separated representation of the geometric mapping is not dependent on the dimensionality of the problem. Therefore, the strategy described in Section 4.3 is also valid in three dimensions. The implementation details are given in Appendix C.

Appendix C. Implementation details of the separated representation of the geometric mapping

To obtain a separated representation of the displacement function in the whole domain, the solid mechanics problem (22) is considered. Its discretisation leads to the system of equations (23).

Assuming that the mesh nodes are ordered so that the boundary nodes are first, the vector \mathbf{D} is given by

$$\mathbf{D} = \left(D_1^{X_1}, \dots, D_1^{X_{n_{\text{sd}}}}, D_2^{X_1}, \dots, D_2^{X_{n_{\text{sd}}}}, \dots, D_{|\mathcal{S}|}^{X_1}, \dots, D_{|\mathcal{S}|}^{X_{n_{\text{sd}}}} \right)^T,$$

where $D_k^{X_l}$ is the imposed displacement of node \mathbf{X}_k in the X_l direction.

As usual in a FE context, the modified system of linear equations to be solved, after accounting for the Dirichlet boundary conditions, is

$$\mathbf{A}_{11}\mathbf{d} = -\mathbf{A}_{12}\mathbf{D}.$$

The solution of this system of linear equations can be written as

$$\mathbf{d} = -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{D} = \sum_{k=1}^{|\mathcal{S}|} \mathbf{B}^k \mathbf{D}_k \quad (\text{C.1})$$

where $\mathbf{D}_k = \left(D_k^{X_1}, \dots, D_k^{X_{n_{\text{sd}}}} \right)^T$ and \mathbf{B}^k denotes the block of the matrix $\mathbf{B} := -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$ containing the columns from $n_{\text{sd}}(k-1) + 1$ to $n_{\text{sd}}k$, with dimension $n_{\text{sd}}|\mathcal{S}| \times n_{\text{sd}}$.

Using the separated representation of the imposed displacement (19) for each boundary node, the following separated representation of the nodal values of the displacement is obtained

$$\mathbf{d} = \sum_{k=1}^{|\mathcal{S}|} \mathbf{B}^k \sum_{j=1}^M \sum_{i=0}^{n_{\text{cp}}^j} \Phi_i^j(\boldsymbol{\mu}) \Theta_i^j(\mathbf{X}_k),$$

leading to the separated representation of the approximation of the displacement function given by Equation (24).

It is worth noting that the matrix \mathbf{B} in Equation (C.1) only depends on the spatial discretisation of the original configuration, Ω , and the selected material parameters (E and ν) and it is independent on the geometric parameters $\boldsymbol{\mu}$. Therefore, it is possible to pre-compute and store the matrix \mathbf{B} so that the separated representation of the displacement of Equation (C.1) can be computed with a negligible cost for different imposed boundary displacements (i.e. for different configurations Ω^μ induced by different variations of the geometric parameters).

It is important to recall that the dimension of the matrix \mathbf{B} is $\mathbf{n}_{\text{sd}}(\mathbf{n}_{\text{mn}} - |\mathcal{S}|) \times \mathbf{n}_{\text{sd}}|\mathcal{S}|$, which, in practical applications, is much lower than the size of a standard FE matrix, namely $\mathbf{n}_{\text{sd}}\mathbf{n}_{\text{mn}} \times \mathbf{n}_{\text{sd}}\mathbf{n}_{\text{mn}}$.

Acknowledgements

This work is partially supported by the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska–Curie actions (Grant numbers: 675919 and 764636) and the Spanish Ministry of Economy and Competitiveness (Grant number: DPI2017-85139-C2-2-R). The second and third authors are also grateful for the financial support provided by the Generalitat de Catalunya (Grant number: 2017-SGR-1278).

References

- [1] F. Alouges, A. DeSimone, and A. Lefebvre. Optimal strokes for axisymmetric microswimmers. *The European Physical Journal E: Soft Matter and Biological Physics*, 28(3):279–284, 2009.
- [2] A. Ammar, A. Huerta, F. Chinesta, E. Cueto, and A. Leygue. Parametric solutions involving geometry: a step towards efficient shape op-

- timization. *Computer Methods in Applied Mechanics and Engineering*, 268:178–193, 2014.
- [3] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics*, 139:153–176, 2006.
- [4] F. Ballarin, A. D’Amario, S. Perotto, and G. Rozza. A POD-selective inverse distance weighting method for fast parametrized shape morphing. *International Journal for Numerical Methods in Engineering*, 117(8):860–884, 2019.
- [5] G. Berkooz, P. Holmes, and J. L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993.
- [6] B. Bognet, F. Bordeu, F. Chinesta, A. Leygue, and A. Poitou. Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. *Computer Methods in Applied Mechanics and Engineering*, 201:1–12, 2012.
- [7] A. Cayley. A memoir on the theory of matrices. *Philosophical Transactions of the Royal Society of London*, 148:17–37, 1858.
- [8] P. R. Childs. *Rotating flow*. Elsevier, 2010.
- [9] F. Chinesta, E. Cueto, and A. Huerta. PGD for solving multidimensional and parametric models. In *Separated representations and PGD-based model reduction*, volume 554 of *CISM Courses and Lectures*, pages 27–89. Springer, Vienna, 2014.

- [10] F. Chinesta, R. Keunings, and A. Leygue. *The proper generalized decomposition for advanced numerical simulations. A primer*. Springer Briefs in Applied Sciences and Technology. Springer, Cham, 2014.
- [11] F. Chinesta, A. Leygue, F. Bordeu, J. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. PGD-based computational vademecum for efficient design, optimization and control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013.
- [12] F. Cirak, M. Ortiz, and P. Schröder. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering*, 47(12):2039–2072, 2000.
- [13] A. Courard, D. Néron, P. Ladevèze, and L. Ballere. Integration of PGD-virtual charts into an engineering design process. *Computational Mechanics*, 57(4):637–651, 2016.
- [14] M. C. Delfour and J.-P. Zolésio. *Shapes and geometries: metrics, analysis, differential calculus, and optimization*, volume 22. SIAM, 2011.
- [15] P. Díez, S. Zlotnik, and A. Huerta. Generalized parametric solutions in Stokes flow. *Computer Methods in Applied Mechanics and Engineering*, 326:223–240, 2017.
- [16] J. Donea and A. Huerta. *Finite Element Methods for Flow Problems*. John Wiley & Sons, 2003.
- [17] D. K. Faddeev and V. N. Faddeeva. *Computational Methods of Linear Algebra*. W. H. Freeman and Company, 1963.
- [18] R. W. Freund. Model reduction methods based on Krylov subspaces. *Acta Numerica*, 12:267–319, 2003.

- [19] T. Heuzé, A. Leygue, and G. Racineux. Parametric modeling of an electromagnetic compression device with the proper generalized decomposition. *International Journal of Material Forming*, 9(1):101–113, 2016.
- [20] A. S. Householder. *The theory of matrices in numerical analysis*. Dover Publications, Inc., New York, 1975. Reprint of 1964 edition.
- [21] G. Legrain. A NURBS enhanced extended finite element approach for unfitted CAD analysis. *Computational Mechanics*, 52(4):913–929, 2013.
- [22] A. Leygue and E. Verron. A first step towards the use of proper general decomposition method for structural optimization. *Archives of Computational Methods in Engineering*, 17(4):465–472, 2010.
- [23] T. Lieu, C. Farhat, and M. Lesoinne. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer Methods in Applied Mechanics and Engineering*, 195(41):5730–5742, 2006.
- [24] D. Mead. Newton’s identities. *American Mathematical Monthly*, pages 749–751, 1992.
- [25] D. Modesto, S. Zlotnik, and A. Huerta. Proper Generalized Decomposition for parameterized helmholtz problems in heterogeneous and unbounded domains: application to harbor agitation. *Computer Methods in Applied Mechanics and Engineering*, 295:127–149, 2015.
- [26] A. T. Patera and G. Rozza. *Reduced Basis Approximation and A-Posteriori Error Estimation for Parametrized Partial Differential Equations*. MIT Pappalardo Graduate Monographs in Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA, 2007.

- [27] P.-O. Persson and J. Peraire. Curved mesh generation and mesh refinement using lagrangian solid mechanics. In *Proceedings of the 47th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA, 2009.
- [28] L. Piegl and W. Tiller. *The NURBS book*. Springer-Verlag, Berlin Heidelberg, 1995.
- [29] R. Poya, R. Sevilla, and A. J. Gil. A unified approach for a posteriori high-order curved mesh generation using solid mechanics. *Computational Mechanics*, 58(3):457–490, 2016.
- [30] A. Quarteroni and G. Rozza. *Reduced Order Methods for Modeling and Computational Reduction*, volume 9. Springer, 2014.
- [31] G. Rozza. Fundamentals of reduced basis method for problems governed by parametrized PDEs and applications. In *Separated representations and PGD-based model reduction*, volume 554 of *CISM Courses and Lectures*, pages 153–227. Springer, Vienna, 2014.
- [32] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [33] R. Sevilla, S. Fernández-Méndez, and A. Huerta. NURBS-enhanced finite element method (NEFEM). *Archives of Computational Methods in Engineering*, 18(4):441, 2011.
- [34] R. Sevilla and A. Huerta. HDG-NEFEM with degree adaptivity for Stokes flows. *Journal of Scientific Computing*, 77(3):1953–1980, 2018.

- [35] R. Sevilla, L. Rees, and O. Hassan. The generation of triangular meshes for NURBS-enhanced FEM. *International Journal for Numerical Methods in Engineering*, 108(8):941–968, 2016.
- [36] H. A. Stone and A. D. Samuel. Propulsion of microorganisms by surface distortions. *Physical Review Letters*, 77(19):4102, 1996.
- [37] S. K. Veerapaneni, A. Rahimian, G. Biros, and D. Zorin. A fast algorithm for simulating vesicle flows in three dimensions. *Journal of Computational Physics*, 230(14):5610–5634, 2011.
- [38] P. E. Vincent and A. Jameson. Facilitating the adoption of unstructured high-order methods amongst a wider community of fluid dynamicists. *Mathematical Modelling of Natural Phenomena*, 6(3):97–140, 2011.
- [39] Z. Q. Xie, R. Sevilla, O. Hassan, and K. Morgan. The generation of arbitrary order curved meshes for 3D finite element analysis. *Computational Mechanics*, 51(3):361–374, 2013.
- [40] S. Zlotnik, P. Díez, D. Modesto, and A. Huerta. Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications. *International Journal for Numerical Methods in Engineering*, 103(10):737–758, 2015.