Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 347 (2019) 568–587
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A reduced basis approach for PDEs on parametrized geometries based on the shifted boundary finite element method and application to a Stokes flow

Efthymios N. Karatzas[a,*], Giovanni Stabile[a], Leo Nouveau[b], Guglielmo Scovazzi[b], Gianluigi Rozza[a]

[a] *SISSA, International School for Advanced Studies, Mathematics Area, mathLab, Trieste, 34136, Italy*
[b] *Civil and Environmental Engineering, Duke University, Durham, NC 27708, United States*

Received 4 August 2018; received in revised form 22 December 2018; accepted 23 December 2018
Available online 8 January 2019

Highlights

- A POD-Galerkin approach to handle problems involving parametrizations of complex geometrical shapes and/or large domain deformations has been developed.
- On the contrary to traditional RB-Galerkin approaches, the proposed methodology do not require the transformation of each geometry to a reference one.
- The tedious handling of cut cells is avoided at the full order level by the SBM formulation, keeping the good properties that other related methods like CutFEM have demonstrated.
- The SBM provides a smooth mapping from the true to the surrogate domain, and for this reason,the stability and performance of the reduced basis are enhanced.
- The efficiency and accuracy of the proposed reduced order model have been numerically verified against the full order model.

Abstract

We propose a model order reduction technique integrating the Shifted Boundary Method (SBM) with a POD-Galerkin strategy. This approach allows to deal with complex parametrized domains in an efficient and straightforward way. The impact of the proposed approach is threefold.

First, problems involving parametrizations of complex geometrical shapes and/or large domain deformations can be efficiently solved at full-order by means of the SBM. This unfitted boundary method permits to avoid remeshing and the tedious handling of cut cells by introducing an approximate surrogate boundary.

Second, the computational effort is reduced by the development of a Reduced Order Model (ROM) technique based on a POD-Galerkin approach.

Third, the SBM provides a smooth mapping from the true to the surrogate domain, and for this reason, the stability and performance of the reduced order basis are enhanced. This feature is the net result of the combination of the proposed ROM approach and the SBM. Similarly, the combination of the SBM with a projection-based ROM gives the great advantage of an

* Corresponding author.
E-mail addresses: efthymios.karatzas@sissa.it (E.N. Karatzas), gstabile@sissa.it (G. Stabile), leo.nouveau@duke.edu (L. Nouveau), guglielmo.scovazzi@duke.edu (G. Scovazzi), grozza@sissa.it (G. Rozza).

https://doi.org/10.1016/j.cma.2018.12.040

easy and fast to implement algorithm considering geometrical parametrization with large deformations. The transformation of each geometry to a reference geometry (morphing) is in fact not required.

These combined advantages will allow the solution of PDE problems more efficiently. We illustrate the performance of this approach on a number of two-dimensional Stokes flow problems.

Keywords: Immersed; Embedded; FEM; Reduced basis; SBM

1. Introduction and motivation

Dealing with the numerical simulation of problems characterized by deforming domains, it is mainly possible to rely on two different approaches: one based on an arbitrary Lagrangian-Eulerian (ALE) formulation [1], where the grid is deformed using a mesh motion algorithm, and one based on an embedded boundary approach, which relies on an undeformed background mesh into which the boundary is embedded or immersed (see [2] and references within). In this work the attention is focused on the second type of approaches and in particular onto ROMs emanating from the recently proposed Shifted Boundary Method presented in [3].

In general, the family of immersed/embedded boundary methods can be very useful in many cases where conformal classic methods are inefficient. Using such methods, it is possible to avoid most of the issues related with mesh deformation and topological changes that normally occur when dealing with geometrical parametrization or fluid–structure interaction problems [4–6]. Starting with the pioneering work of Peskin [7], great effort has been focused on embedded or immersed methods.

However, for both immersed or conforming methods, there are still many cases where the solution of partial differential equations with standard discretization techniques (Finite Element Method, Finite Volume Method, Finite Difference Method) becomes unfeasible. Such situations occur, for example, whenever a large number of different system configurations need to be tested - as in uncertainty quantification, optimization, parametrization studies - or reduced computational times are required as in real-time control problems. A possible way to overcome this limitation is the use of reduced order modeling techniques [8–11]. In particular we rely onto reduced basis methods.

It is beyond the scope of this paper to provide a comprehensive review on immersed/embedded methods and we report here only some of the most significant contributions on the topic. The reader interested in more detail on the different existing approaches, namely Ghost-Cell finite difference methods, Cut-Cell finite volume approach, Immersed Interface, Ghost Fluid, Volume Penalty methods can refer to the review paper [2] and references therein.

In [3,12], a new Embedded Boundary Method (EBM) - called Shifted Boundary Method (SBM) - was introduced and applied to heat transfer problems, advection-diffusion equations, Stokes flow and laminar and turbulent Navier–Stokes equations. The idea of the SBM is to shift the position of the boundaries from their true location to a surrogate one and to appropriately modify the value of boundary conditions on shifted boundaries, with the goal of simplicity, robustness and efficiency. The main characteristic of the SBM is that the surrogate boundary location is chosen so that cut cells are completely avoided while optimal convergence rates and low algebraic condition numbers are preserved. These features make the SBM a candidate for the efficient solution of fluid problems with parametrized geometries, fluid–structure interaction problems, evolutionary time systems, nonlinear problem cases, hyperbolic problems [3,12,13].

The overall objective of this paper is to investigate how the SBM may be employed to effectively and efficiently solve parameterized partial differential problems within the reduced basis (RB) context. To the best of our knowledge, the combination of the reduced basis method with full-order embedded boundary methods has not been investigated with the aim of improving computational performance and allow offline–online computational strategies. We mention here the work of [14] for classical EBMs and model reduction, in which two regions are separated by an evolving interface and the snapshot compression problem is formulated as a binary weighted low-rank approximation. In particular, in the present work, the attention is focused on reduced order methods on parametrized geometries. We remark that the use of an embedded method at full-order level allows us to avoid the need to map all the deformed configurations to reference domains (morphing), as it is traditionally done in systems with parametrized geometry, see e.g. [8,11,15–19] and references therein. Mapping all the possible geometrical configurations to a reference domain, especially in cases with large deformations, may in fact produce highly distorted meshes and therefore lead to ill conditioned problems.

The work is organized as follows: in Section 2 we introduce the formulation and the methods used for the full-order approximation of the equations. In Section 3 the reduced order methodology is discussed in detail, while in Section 4 the proposed ROM technique is tested on a numerical benchmark, dealing with the geometrical parametrized problem of the flow around a circular embedded domain. Finally in Section 5 conclusions and perspectives for future improvements and developments are drawn.

2. The physical problem and the full-order approximation

2.1. Strong formulation of the steady Stokes problem

The Stokes equations describe the flow of a Newtonian, incompressible viscous fluid when the convective forces are negligible with respect to the viscous forces. Consider an open domain $\mathcal{D}$ in $\mathbb{R}^d$, with $d = 2,3$ the number of space dimensions with Lipschitz boundary Γ (split into two sub-boundaries Γ_D, Γ_N) and let a $k-$dimensional parameter space $\mathcal{P}$ with a parameter vector $\mu \in \mathcal{P} \subset \mathbb{R}^k$, the strong form of the stationary Stokes flow system of equations with Dirichlet and Neumann boundary conditions on Γ_D and Γ_N respectively, geometrically parameterized by μ, is given by:

$$
\begin{aligned}
-\nabla \cdot (2\nu \boldsymbol{\epsilon}(\boldsymbol{u}(\mu)) - p(\mu)\boldsymbol{I}) &= \boldsymbol{g}(\mu), && \text{in } \mathcal{D}(\mu),\\
\nabla \cdot \boldsymbol{u}(\mu) &= 0, && \text{in } \mathcal{D}(\mu),\\
\boldsymbol{u}(\mu) &= \boldsymbol{g}_D(\mu), && \text{on } \Gamma_D(\mu),\\
(2\nu \boldsymbol{\epsilon}(\boldsymbol{u}(\mu)) - p(\mu)\boldsymbol{I}) \cdot \boldsymbol{n} &= \boldsymbol{g}_N(\mu), && \text{on } \Gamma_N(\mu).
\end{aligned}
$$

We denote by ν the viscosity, $\boldsymbol{\epsilon}(\boldsymbol{u}) = 1/2(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)$ the velocity strain tensor (i.e., the symmetric gradient of the velocity $\boldsymbol{u}$), p is the pressure, $\boldsymbol{g}$ a body force, $\boldsymbol{g}_D$ the values of the velocity on the Dirichlet boundary and $\boldsymbol{g}_N$ is the normal stress on the Neumman boundary. The first equation represents the conservation of the linear momentum of the fluid, while the second equation is the incompressibility condition and describes the mass conservation.

2.2. Full-order parametrized Shifted Boundary Method formulation

In the following, we recall the full-order Shifted Boundary Method (SBM) for the steady Stokes equations following [3]. The Shifted Boundary Method is an embedded (immersed, non-conformal) finite element method, which relies on a Nitsche-type approach for the weak imposition of Dirichlet boundary conditions [3,12,13]. Weak boundary conditions require less complicated data structures with respect to strongly imposed boundary conditions, and, for this reason, the SBM relies on the Nitsche approach. For the sake of simplicity, in this subsection, we will omit the parameter dependency with respect to μ. We also denote $\upsilon := \upsilon_h$.

We define now an approximate computational domain and an approximate boundary starting from the true boundary and the true computational domain, with the purpose of avoiding cut cells. As seen in Fig. 1, we denote by $\tilde{\Gamma}$ the surrogate boundary composed of the edges/faces of the mesh that are the closest to the true boundary Γ. The closest faces/edges of $\tilde{\Gamma}$ to Γ are detected using the closest-point projection algorithm.

The surrogate boundary $\tilde{\Gamma}$ encloses the surrogate domain $\tilde{\mathcal{D}} \subset \mathcal{D}$, where $\mathcal{D}$ is the original (true) computational domain. Furthermore, $\tilde{\boldsymbol{n}}$ indicates the unit outward-pointing normal to the surrogate boundary $\tilde{\Gamma}$, and it differs from the outward-pointing normal $\boldsymbol{n}$ of Γ. Notice also that the closest-point projection, in spite of the segmented/faceted nature of the surrogate boundary $\tilde{\Gamma}$, is actually a continuous, piecewise-smooth mapping $\boldsymbol{M}$ from points in $\tilde{\Gamma}$ to points in Γ:

$$\boldsymbol{M} : \tilde{\boldsymbol{x}} \in \tilde{\Gamma} \rightarrow \boldsymbol{x} \in \Gamma.$$

The mapping $\boldsymbol{M}$ defines a distance vector function

$$\boldsymbol{d} \equiv d_M(\boldsymbol{x}) = \boldsymbol{x} - \tilde{\boldsymbol{x}} = [M - I](\tilde{\boldsymbol{x}}),$$

where the distance vector $\boldsymbol{d} = \|\boldsymbol{d}\|\boldsymbol{n}$ is aligned along $\boldsymbol{n}$, due to the closest point projection properties, see Fig. 1. Since the distance vector $\boldsymbol{d}$ is aligned with the normal to the true surface Γ and the true surface is smooth between edges and corners, it is legitimate to assume that $\boldsymbol{M}$ is continuous, and piecewise smooth. The unit normal vector $\boldsymbol{n}$ and the unit tangential vectors $\boldsymbol{\tau}_i$ $(1 < i < d-1)$ to the boundary Γ can be easily extended to the boundary $\tilde{\Gamma}$ since

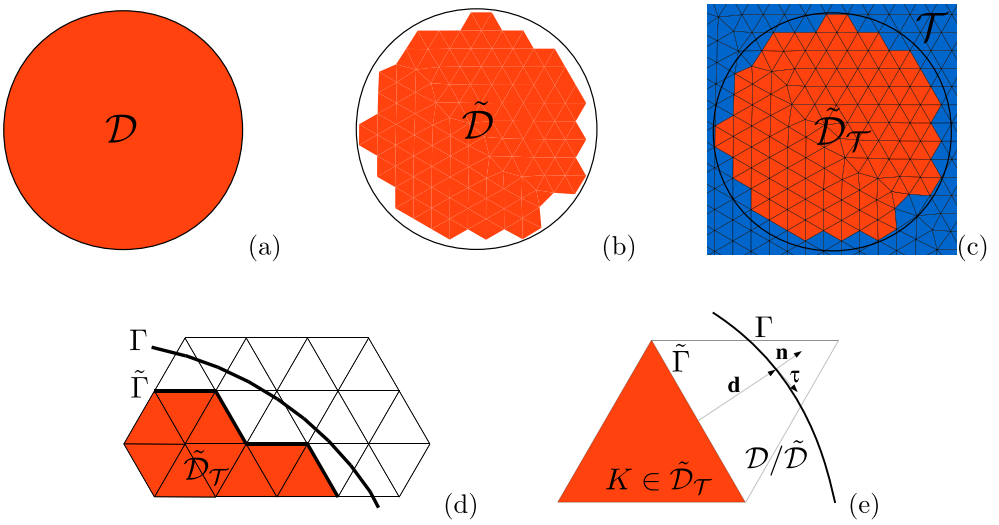

Fig. 1. (a) The geometry of a disk, (b) the SBM surrogate geometry, (c) a zoom on the background mesh together with the surrogate SBM discretized geometry, (d) SBM mesh and surrogate boundary and (e) the normal and distance vector considering one element.

$\bar{\boldsymbol{n}}(\tilde{\boldsymbol{x}}) \equiv \boldsymbol{n}(\boldsymbol{M}(\tilde{\boldsymbol{x}}))$, $\bar{\boldsymbol{\tau}}_i(\boldsymbol{x}) \equiv \boldsymbol{\tau}_i(\boldsymbol{M}(\tilde{\boldsymbol{x}}))$. In the following, whenever we write $\boldsymbol{n}(\tilde{\boldsymbol{x}})$ we actually mean $\bar{\boldsymbol{n}}(\tilde{\boldsymbol{x}})$ at a point $\tilde{\boldsymbol{x}} \in \tilde{\Gamma}$, and similarly for $\boldsymbol{\tau}_i(\tilde{\boldsymbol{x}})$ and $\bar{\boldsymbol{\tau}}_i(\tilde{\boldsymbol{x}})$. We can also introduce the derivative of a function $\boldsymbol{g}$ along the direction $\bar{\boldsymbol{\tau}}_i$ at a point $\tilde{\boldsymbol{x}} \in \tilde{\Gamma}$ as $\nabla_{\bar{\boldsymbol{\tau}}_i} \bar{\boldsymbol{g}} = \nabla \bar{\boldsymbol{g}}(\tilde{\boldsymbol{x}}) \cdot \bar{\boldsymbol{\tau}}_i(\tilde{\boldsymbol{x}})$. The above constructions are the key ingredients in the extension of the boundary conditions on Γ to the boundary $\tilde{\Gamma}$ of the surrogate domain.

Under the important assumption $\boldsymbol{n} \cdot \tilde{\boldsymbol{n}} \geq 0$, which relates to minimal grid resolution, we can now introduce the SBM variational formulation. Consider a discretization of the continuous boundary value problem with a mesh $\tilde{\mathcal{D}}_{\mathcal{T}}$ consisting of simplices K belonging to a tessellation $\mathcal{T}$. Moreover, we introduce the discrete spaces $\boldsymbol{V}_h$ and Q_h, for velocity and pressure and we assume that a stable and convergent base formulation for the Stokes flow exists for these spaces in the case of conformal grids. For the piecewise linear spaces

$$\boldsymbol{V}_h = \left\{ v \in (C^0(\tilde{\mathcal{D}}(\mu)))^2 : \boldsymbol{v}|_K \in (P^1(K))^2, \forall K \in \tilde{\mathcal{D}}_{\mathcal{T}}(\mu) \right\},$$

$$Q_h = \left\{ v \in C^0(\tilde{\mathcal{D}}(\mu)) : v|_K \in P^1(K), \forall K \in \tilde{\mathcal{D}}_{\mathcal{T}}(\mu) \right\},$$

the stabilized formulation of [20] satisfies these assumptions, and will be used in what follows, although alternative choices are possible. Considering the standard notation $(\cdot,\cdot)_{\tilde{\mathcal{D}}}$, $\langle \cdot,\cdot \rangle_{\tilde{\Gamma}_D}$, $\langle \cdot,\cdot \rangle_{\tilde{\Gamma}_N}$ for the $L^2(\tilde{\mathcal{D}})$, $L^2(\tilde{\Gamma}_D)$ and $L^2(\tilde{\Gamma}_N)$ inner products onto the surrogate geometry $\tilde{\mathcal{D}}$, $\tilde{\Gamma}_D$ and $\tilde{\Gamma}_N$, respectively, and that for every element $K \in \mathcal{T}$, we associate a parameter h_K, denoting the diameter of the set K, and the size of the mesh is denoted by $h = \max_{K \in \mathcal{T}} h_K$, the SBM weak form reads:

Find $\boldsymbol{u} \in \boldsymbol{V}_h$ *and* $p \in Q_h$ *such that,* $\forall \boldsymbol{w} \in \boldsymbol{V}_h$ *and* $\forall q \in Q_h$,

$$\begin{cases} a(\boldsymbol{w}, \boldsymbol{u}; \mu) + b(\boldsymbol{w}, p; \mu) = \ell_g(\boldsymbol{w}; \mu), \\ b(\boldsymbol{u}, q; \mu) + \hat{b}(\boldsymbol{u}, q; \mu) + (\text{stab. term.}) = \ell_q(q; \mu), \end{cases} \tag{1}$$

where

$$\begin{aligned} a(\boldsymbol{w}, \boldsymbol{u}; \mu) = {} & (\boldsymbol{\epsilon}(\boldsymbol{w}), 2\nu\boldsymbol{\epsilon}(\boldsymbol{u}))_{\tilde{\mathcal{D}}(\mu)} - \langle \boldsymbol{w} \otimes \tilde{\boldsymbol{n}}, 2\nu\boldsymbol{\epsilon}(\boldsymbol{u}) \rangle_{\tilde{\Gamma}_D(\mu)} - \langle 2\nu\boldsymbol{\epsilon}(\boldsymbol{w}), (\boldsymbol{u} + (\nabla \boldsymbol{u})\boldsymbol{d}) \otimes \tilde{\boldsymbol{n}} \rangle_{\tilde{\Gamma}_D(\mu)} \\ & + \alpha \langle 2\nu / h(\boldsymbol{w} + (\nabla \boldsymbol{w})\boldsymbol{d}), \boldsymbol{u} + (\nabla \boldsymbol{u})\boldsymbol{d} \rangle_{\tilde{\Gamma}_D(\mu)} + \beta \left\langle 2\nu h \nabla_{\bar{\boldsymbol{\tau}}_i} \boldsymbol{w}, \nabla_{\bar{\boldsymbol{\tau}}_i} \boldsymbol{u} \right\rangle_{\tilde{\Gamma}_D(\mu)}, \end{aligned} \tag{2}$$

$$b(\boldsymbol{w}, p; \mu) = -(\nabla \cdot \boldsymbol{w}, p)_{\tilde{\mathcal{D}}(\mu)} + \langle \boldsymbol{w} \cdot \tilde{\boldsymbol{n}}, p \rangle_{\tilde{\Gamma}_D(\mu)}, \tag{3}$$

$$\begin{aligned} \ell_g(\boldsymbol{w}; \mu) = {} & -(\boldsymbol{w}, \boldsymbol{g})_{\tilde{\mathcal{D}}(\mu)} + \left\langle \boldsymbol{w}, \boldsymbol{g}_N \right\rangle_{\tilde{\Gamma}_N(\mu)} - \left\langle 2\nu\boldsymbol{\epsilon}(\boldsymbol{w}), \bar{\boldsymbol{g}}_{\boldsymbol{D}} \otimes \tilde{\boldsymbol{n}} \right\rangle_{\tilde{\Gamma}_D(\mu)} \\ & + \alpha \left\langle 2\nu / h(\boldsymbol{w} + (\nabla \boldsymbol{w})\boldsymbol{d}), \bar{\boldsymbol{g}}_{\boldsymbol{D}} \right\rangle_{\tilde{\Gamma}_D(\mu)}, \end{aligned} \tag{4}$$

$$\hat{b}(\boldsymbol{u}, q; \mu) = \langle q\boldsymbol{d} \otimes \tilde{\boldsymbol{n}}, \nabla \boldsymbol{u} \rangle_{\tilde{\Gamma}_{D(\mu)}}, \tag{5}$$

$$\ell_q(q; \mu) = \langle \bar{\boldsymbol{g}}_{\boldsymbol{D}} \cdot \tilde{\boldsymbol{n}}, q \rangle_{\tilde{\Gamma}_{D(\mu)}}. \tag{6}$$

These equations yield the following algebraic system of equations:

$$\begin{bmatrix} \boldsymbol{A}(\mu) & \boldsymbol{B}^T(\mu) \\ \boldsymbol{B}(\mu) + \hat{\boldsymbol{B}}(\mu) & \boldsymbol{C}(\mu) \end{bmatrix} \begin{bmatrix} \boldsymbol{u}(\mu) \\ p(\mu) \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_g(\mu) \\ \boldsymbol{F}_q(\mu) \end{bmatrix}. \tag{7}$$

In the above system of equations it is important to highlight several factors that will play a crucial role in Section 3. First of all, the discretized differential operators $\boldsymbol{A}$, $\boldsymbol{B}$ and $\hat{\boldsymbol{B}}$ are parameter dependent. Secondly, in the typical saddle point structure of the Stokes problem, the incompressibility equation is partially relaxed adding a stabilization term $\boldsymbol{C}$, derived according to [21]. This stabilization term, at full-order level, permits to circumvent the fulfillment of the "inf-sup" condition and the use of otherwise unstable pair of finite elements, such as $\mathbb{P}_1 - \mathbb{P}_1$, and, as reported in Section 3, it helps to partially preserve the stability of the reduced order model. In principle, the SBM method can also be combined with"inf-sup"-stable elements, such as the Taylor–Hood element: the $\mathbb{P}_1 - \mathbb{P}_1$ was chosen for its simplicity.

The presented formulation is used to solve the full-order problem during an offline stage and to produce the snapshots necessary for the construction of the ROM of Section 3.

3. Reduced order model with a POD-Galerkin method

A Reduced Order Model (ROM) is a simplification of a Full Order Model (FOM) model that preserves its essential behavior and dominant effects with the purpose of reducing solution time or storage capacity. In this section, the projection-based ROM, that in this case relies on a POD-Galerkin approach [8,22], is described and all the relevant issues are highlighted.

The interest is focused on partial differential equations with parametrized geometries that govern fluid dynamics problems and, in particular, on the advantages related with the use of the SBM. The attention is devoted on Reduced Basis (RB) ROM generated starting from high dimensional SBM approximations. Looking at pioneering works dealing with the RB method, starting from FEM full-order approximations, linear elliptic equations have been treated in [19], linear parabolic equations in [23], and non-linear problems in [24,25]. In spite of the great number of works on reduced order modeling for fluid dynamics problems, to the best of the authors' knowledge, only very few research works can be found dealing with embedded boundary methods [14].

Before going into details, we remind the basics of the reduced basis method. The first step consists of the construction of a set of FOM solutions of the parametrized problem under the variation of the input parameters. The final goal of RB methods is to approximate any member of this solution set, and in general of the solution manifold, with a reduced number of basis functions. The method consists of a two-stage procedure: the offline and the online one. During the costly *offline stage*, one assembles the solution set and examines its components in order to construct a reduced basis that approximates any member of the solution set to a prescribed accuracy. This phase involves the solution of a possibly large number of FOM problems and the cost is usually high. During a second stage, namely the *online stage*, after the Galerkin/Petrov–Galerkin projection of the full-order differential operators describing the governing equations onto the reduced basis spaces, it is possible to solve a reduced problem for any new value of the input parameters. This offline–online procedure is effective in many scenarios. We mention here the case when a large number of parameter values are in need of being tested with the consequent repeated evaluation of the system response, and the case when a reduced computational time is required or only limited computational power and memory is available.

We recall here that POD-Galerkin ROMs for the incompressible Stokes and Navier–Stokes equations suffer from stability issues [15,17,18,26–29], due to pressure instabilities, while for dynamic instabilities on transient problems see for instance [30–34]. More details will be given in Section 3.3.1.

From a reduced order modeling point of view, the first aim of this work is to investigate how ROMs are applied with an SBM approach, and which are the perspectives. Since our main interest is to generate ROMs for flow problems on parametrized geometries we consider the case of a Stokes flow problem. However, for the case of a Poisson problem on a parametrized geometry, which, originally, served as a simple feasibility test we refer to [35]. The SBM unfitted/surrogate mesh Nitsche finite element method is used to apply parametrization and reduced order techniques considering Dirichlet or Dirichlet combined with Neumann boundary conditions.

An objective of this work is also to test the efficiency of a geometrically parametrized ROM method without the use of the transformation to reference domains and to highlight the advantages of having a fixed background mesh. Moreover, we will highlight and investigate how common strategies for pressure stabilization can be transferred to the present framework, [15,17,36].

3.1. The Proper Orthogonal Decomposition (POD)

In order to generate the reduced basis spaces, for the projection of the governing equations, one can find in the literature several techniques such as the Proper Orthogonal Decomposition (POD), the Proper Generalized Decomposition (PGD) and the Reduced Basis (RB) with a greedy sampling strategy. For more details about the different strategies the reader may see [9,10,19,37–39]. In this work a POD strategy is exploited and is chosen to apply the POD onto the full snapshots matrices that include parameter dependency. The full-order model is solved for each $\mu^k \in \mathcal{K} = \{\mu^1, \dots, \mu^{N_k}\} \subset \mathcal{P}$ where $\mathcal{K}$ is a finite dimensional training set of samples chosen inside the parameter space $\mathcal{P}$. The number of snapshots is denoted by N_s and the number of degrees of freedom for the discrete full-order solution by N_u^h, N_p^h for the velocity and pressure, respectively. The snapshots matrices $\boldsymbol{\mathcal{S}_u}$ and $\boldsymbol{\mathcal{S}_p}$, for velocity and pressure, are then given by N_s full-order snapshots:

$$\boldsymbol{\mathcal{S}_u} = [\boldsymbol{u}(\mu^1), \dots, \boldsymbol{u}(\mu^{N_s})] \in \mathbb{R}^{N_u^h \times N_s}, \quad \boldsymbol{\mathcal{S}_p} = [p(\mu^1), \dots, p(\mu^{N_s})] \in \mathbb{R}^{N_p^h \times N_s}. \tag{8}$$

Given a general scalar or vectorial function $\boldsymbol{u} : \mathcal{D} \to \mathbb{R}^d$, with a certain number of realizations $\boldsymbol{u}_1, \dots, \boldsymbol{u}_{N_s}$, the POD problem consists in finding, for each value of the dimension of POD space $N_{POD} = 1, \dots, N_s$, the scalar coefficients $a_1^1, \dots, a_1^{N_s}, \dots, a_{N_s}^1, \dots, a_{N_s}^{N_s}$ and functions $\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{N_s}$ that minimize the quantity:

$$E_{N_{POD}} = \sum_{i=1}^{N_s} \|\boldsymbol{u}_i - \sum_{k=1}^{N_{POD}} a_i^k \boldsymbol{\varphi}_k\|^2_{L^2(\mathcal{D})}, \ \forall N_{POD} = 1, \dots, N, \quad \text{with } (\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j)_{\mathcal{D}} = \delta_{ij}, \quad \forall i, j = 1, \dots, N_s. \tag{9}$$

In this case the velocity field $\boldsymbol{u}$ is used as example. It can be shown [40] that the minimization problem of Eq. (9) is equivalent of solving the following eigenvalue problem:

$$\boldsymbol{\mathcal{C}^u}\, \boldsymbol{Q^u} = \boldsymbol{Q^u} \boldsymbol{\lambda^u}, \qquad \text{for } \mathcal{C}^u_{ij} = (\boldsymbol{u}_i, \boldsymbol{u}_j)_{\mathcal{D}}, i, j = 1, \dots, N_s,$$

where $\boldsymbol{\mathcal{C}^u}$ is the correlation matrix obtained starting from the snapshots $\boldsymbol{\mathcal{S}_u}$, $\boldsymbol{Q^u}$ is a square matrix of eigenvectors and $\boldsymbol{\lambda^u}$ is a diagonal matrix of eigenvalues.

The basis functions can then be obtained with:

$$\boldsymbol{\varphi_i} = \frac{1}{N_s \lambda_{ii}^{u\,1/2}} \sum_{j=1}^{N_s} \boldsymbol{u}_j Q^u_{ij}. \tag{10}$$

The same procedure can be repeated also for the pressure field considering the snapshots matrix consisting of the snapshots $p_1, p_2, \dots, p_{N_s}$. One can compute the correlation matrix of the pressure field snapshots $\boldsymbol{C^p}$ and solve a similar eigenvalue problem $\boldsymbol{C^p}\, \boldsymbol{Q^p} = \boldsymbol{Q^p} \boldsymbol{\lambda^p}$. The POD modes χ_i for the pressure field can be computed with:

$$\chi_i = \frac{1}{N_s \lambda_{ii}^{p\,1/2}} \sum_{j=1}^{N_s} p_j Q^p_{ij}. \tag{11}$$

The POD spaces are constructed for both velocity and pressure using the aforementioned methodology resulting in the spaces:

$$\boldsymbol{L}_u = [\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{N_u^r}] \in \mathbb{R}^{N_u^h \times N_u^r}, \quad \boldsymbol{L}_p = [\chi_1, \dots, \chi_{N_p^r}] \in \mathbb{R}^{N_p^h \times N_p^r}. \tag{12}$$

where N_u^r, $N_p^r < N_s$ are chosen according to the eigenvalue decay of $\boldsymbol{\lambda}^u$ and $\boldsymbol{\lambda}^p$, [11,19].

Remark 1. The construction of the reduced order basis is based on the whole background domain. For this reason, the manipulation of the out of interest – outside – the true geometry area, namely "ghost area", needs particular care. In [14] the ghost area solution is set to zero. In the present work we use the solution values as they are computed

applying the shifted boundary method and the smooth map $\boldsymbol{M}$ from the true to the surrogate domain. This allows a smooth extension of the solution to the neighboring ghost elements with values which are decreasing smoothly to zero, see for instance the zoomed image in Fig. 13. This approach guarantees a regular "solution" in the background domain and permits the construction of a reduced basis with better approximation properties. For more details and a full investigation of the possible choices and of the handling of the ghost area we refer to [41].

3.2. The geometrical parametrization: main differences with respect to a reference domain approach

The standard procedure to deal with geometrical parametrization in a reduced order modeling framework is to map all the deformed configurations $\mathcal{D}(\mu)$ to a fixed reference domain $\mathcal{D}$ using a map $\mathcal{M}(\mu) : \mathcal{D} \rightarrow \mathcal{D}(\mu)$ see e.g. [8,11,15–19]. For the case at hand, considering a standard Galerkin formulation, without taking into consideration the stabilization terms, the weak formulation written on a reference domain reads:

$$\begin{cases} \tilde{a}(\boldsymbol{w}, \boldsymbol{u}; \mu) + \tilde{b}(\boldsymbol{w}, p; \mu) = \tilde{\ell}_g(\boldsymbol{w}; \mu) \\ \tilde{b}(\boldsymbol{u}, q; \mu) = 0, \end{cases} \tag{13}$$

where the linear and bilinear forms are now written into a common reference domain:

$$\tilde{a}(\boldsymbol{w}, \boldsymbol{u}; \mu) = \int_{\mathcal{D}} \boldsymbol{\epsilon}(\boldsymbol{w}) 2\nu (J_T(\mu))^{-1} (J_T(\mu))^{-T} |J_T(\mu)| \boldsymbol{\epsilon}(\boldsymbol{u}) \mathrm{d}\boldsymbol{x}, \tag{14}$$

$$\tilde{b}(\boldsymbol{w}, p; \mu) = -\int_{D} \nabla \cdot \boldsymbol{w} (J_T(\mu))^{-1} |J_T(\mu)| p \mathrm{d}\boldsymbol{x}, \tag{15}$$

$$\tilde{\ell}_g(\boldsymbol{w}; \mu) = \int_{\mathcal{D}} |J_T(\mu)| \boldsymbol{g} \cdot \boldsymbol{w} \mathrm{d}\boldsymbol{x}, \tag{16}$$

and J_T is the Jacobian of the map $\mathcal{M}(\mu)$ and $|J_T(\mu)|$ is its determinant. Moreover, in cases where it is possible to obtain a disjoint decomposition of the domain $\{\mathcal{D}_r\}_{r=1}^{R}$ such that for each $r \in \{1, \ldots, R\}$ the map $\mathcal{M}_r(\mu) : \mathcal{D}_r \rightarrow \mathcal{D}_r(\mu)$ is affine, it is possible to rewrite the above integrals as a sum of integrals over each subdomain and to take out from the integral the parametric dependent part. In this way it is possible to have an efficient offline–online splitting. However, especially for complex geometrical deformation, the operation of finding such decomposition is not trivial and may lead to a large number of subdomains. Moreover, in cases with large geometrical deformations, writing the equations on a reference domain may lead to distorted elements and therefore to numerical instabilities.

In the proposed approach, because we are always working onto the same physical domain, there is no need to introduce the change of variables in order to map the integrals to a common reference domain. The linear and bilinear forms of Eq. (1) are "naturally" parametrized through the surrogate boundary $\tilde{\Gamma}(\mu)$ and the surrogate domain $\tilde{\mathcal{D}}(\mu)$.

Our method, similarly to the reference domain approach, does not permit to avoid the non-affinity issue of the discretized differential operators. However, in order to obtain an efficient offline–online splitting, it is possible to rely on hyper reduction techniques such as the empirical interpolation method [42], the GNAT [43] or the Gappy-POD [44].

3.3. The projection stage and the generation of the ROM

Once the POD functional spaces are set, the reduced velocity and pressure fields can be approximated with:

$$\boldsymbol{u}^{\boldsymbol{r}} \approx \sum_{i=1}^{N_u^r} a_i(\mu) \boldsymbol{\varphi}_i(\boldsymbol{x}) = \boldsymbol{L}_u \boldsymbol{a}(\mu), \qquad p^r \approx \sum_{i=1}^{N_p^r} b_i(\mu) \chi_i(\boldsymbol{x}) = \boldsymbol{L}_p \boldsymbol{b}(\mu). \tag{17}$$

The reduced solution vectors $\boldsymbol{a} \in \mathbb{R}^{N_u^r \times 1}$ and $\boldsymbol{b} \in \mathbb{R}^{N_p^r \times 1}$ depend only on the parameter values and the basis functions $\boldsymbol{\varphi}_i$ and χ_i depend only on the physical space. The unknown vectors of coefficients $\boldsymbol{a}$ and $\boldsymbol{b}$ can be obtained through a Galerkin projection of the full-order system of equations onto the POD reduced basis spaces and with the resolution of a consequent reduced algebraic system:

$$\begin{bmatrix} \boldsymbol{L}_u & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L}_p \end{bmatrix} \begin{bmatrix} \boldsymbol{A}(\mu) & \boldsymbol{B}^T(\mu) \\ \boldsymbol{B}(\mu) + \hat{\boldsymbol{B}}(\mu) & \boldsymbol{C}(\mu) \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_u^T & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L}_p^T \end{bmatrix} \begin{bmatrix} \boldsymbol{u}(\mu) \\ p(\mu) \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_u^T & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L}_p^T \end{bmatrix} \begin{bmatrix} \boldsymbol{F}_g(\mu) \\ \boldsymbol{F}_q(\mu) \end{bmatrix}, \tag{18}$$

which leads to the following algebraic reduced system:

$$\begin{bmatrix} \boldsymbol{A}^r(\mu) & \boldsymbol{B}^{r\,T}(\mu) \\ \boldsymbol{B}^r(\mu) + \hat{\boldsymbol{B}}^r(\mu) & \boldsymbol{C}^r(\mu) \end{bmatrix} \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}^r_g(\mu) \\ \boldsymbol{F}^r_q(\mu) \end{bmatrix}, \tag{19}$$

where $\boldsymbol{A}^r(\mu) \in N^r_u \times N^r_u$, $\boldsymbol{B}^r(\mu)$, $\hat{\boldsymbol{B}}^r(\mu) \in N^r_p \times N^r_u$, $\boldsymbol{C}^r(\mu) \in N^r_p \times N^r_p$ and $\boldsymbol{F}^r_g(\mu) \in N^r_u \times 1$, $\boldsymbol{F}^r_q(\mu) \in N^r_p \times 1$ are the reduced discretized operators and reduced forcing vectors respectively. The dimension of the reduced operators, as seen also in the numerical examples, is usually much smaller with respect to the dimension of the full-order system of equations and therefore less expensive to solve. We remark here that the full-order discretized differential operators that appear in Eq. (7) are parameter dependent and therefore, also at the reduced order level, we need to assemble the FOM problem in order to compute the reduced differential operator. Possible ways to avoid such potentially expensive operation, relying on an approximate affine approximation of the full-order differential operator, could be to use hyper reduction techniques, [42,43,45]. In this work, since the attention is mainly devoted to the methodological development of a reduced order method in an embedded boundary setting rather than in its efficiency, we do not rely on such hyper reduction techniques and we assemble the full-order differential operators also during the online stage. Considering that the most demanding computational effort is spent during the resolution of the full-order problem rather than in the assembly of the differential operators, as reported in Section 4, it is anyway possible to achieve a good computational speedup. We remark here that, during the online stage, also the stabilization term $\boldsymbol{C}$ is projected onto the reduced basis space, the implications of such a choice are reported in the following section.

3.3.1. Stability issues

The reduced problem, as formulated in Section 3, may present stability issues. It is well known in fact that, using a mixed formulation for the approximation of the incompressible Stokes equations, the approximation spaces need to satisfy the "inf-sup" (Ladyzhenskaya–Brezzi–Babuska) condition, see [46,47]. It is required that there should exist a constant $\beta > 0$, independent to the discretization parameter h, such that:

$$\inf_{0 \neq q \in Q_h} \sup_{0 \neq \boldsymbol{v} \in \mathbf{V}_h} \frac{\langle \nabla \cdot \boldsymbol{v}, q \rangle}{\|\nabla \boldsymbol{v}\| \, \|q\|} \geq \beta > 0. \tag{20}$$

Dealing with classic finite element methods, for what concerns the full-order level, this requirement can be met choosing appropriate finite element spaces such as the standard Taylor–Hood ($\mathbb{P}_2 - \mathbb{P}_1$). In our case, at full-order level, as explained in Section 2, in order to tackle this issue, we rely instead on a stabilized finite element technique. Since also the reduced order problem has a saddle point structure we need to ensure that a *reduced* version of the LBB condition is fulfilled; Regardless the full-order discretization technique, even though the snapshots have been obtained by stable numerical methods, there is no guarantee that the original properties of the full-order system are preserved after the Galerkin projection onto the RB spaces [15,17,27]. To overcome this issue, most of the contributions available in literature do not attempt to recover the pressure field and, at reduced order level, resolve only the momentum equation neglecting the contribution of the gradient of pressure. Although, as highlighted in [48], in many applications the pressure term is needed and cannot be neglected. In the proposed approach the velocity space is enriched in order to satisfy a reduced version of the "inf-sup" condition, see [15,17,49].

Remark 2. It is important to remark that, during the projection stage, we perform also the projection of the stabilization terms. As we will see in the numerical experiments this helps to employ a relatively smaller number of additional supremizer modes.

3.3.2. Supremizer enrichment

As mentioned in Section 3.3.1, the problem, formulated using a mixed formulation, needs either to meet the "inf-sup" condition or to be properly stabilized. To achieve this, it is proposed the fulfillment of a reduced (and also parametric) version of the "inf-sup" condition. Within this approach, the velocity supremizer basis functions $\boldsymbol{L}_{\text{sup}}$ are computed and added to the reduced velocity space, which is transformed into $\tilde{\boldsymbol{L}}_u$, namely:

$$\boldsymbol{L}_{\text{sup}} = [\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_{N^r_{\text{sup}}}] \in \mathbb{R}^{N^h_u \times N^r_{\text{sup}}}, \quad \tilde{\boldsymbol{L}}_u = [\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_{N^r_u}, \boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_{N^r_{\text{sup}}}] \in \mathbb{R}^{N^h_u \times (N^r_u + N^r_{\text{sup}})}. \tag{21}$$

These basis functions are chosen solving a supremizer problem that ensures the fulfillment of a reduced version of the "inf-sup" condition. The supremizer solution $\boldsymbol{s}_i$, corresponding to the parameter value μ^i, given a certain pressure

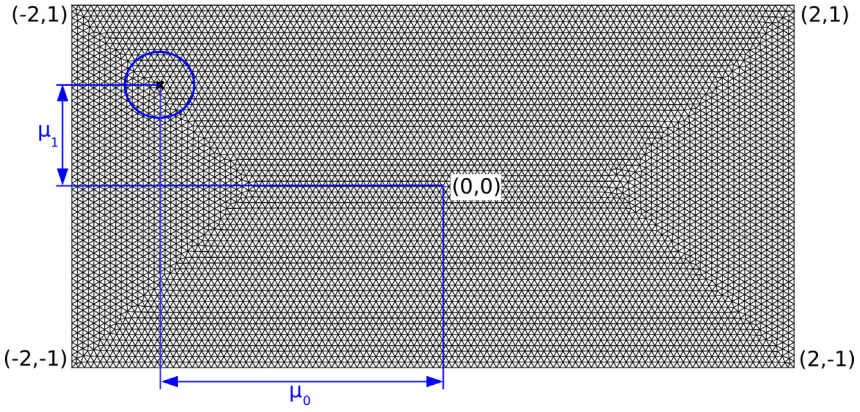

Fig. 2. The background mesh together with a sketch of the embedded domain and the parameters considered in the numerical examples.

basis function χ_i, is the "ingredient" that permits the realization of the "inf-sup" condition. For each pressure basis function the corresponding supremizer element can be found solving the following problem:

$$\Delta \boldsymbol{s}_i = -\nabla \chi_i \text{ in } \mathcal{D}(\mu^i), \quad \boldsymbol{s}_i = \boldsymbol{0} \text{ on } \Gamma(\mu^i). \tag{22}$$

In this case, since we want to rely on the same FOM solver, also the supremizer problem is solved with an SBM approach using the Poisson solver presented in [3]. For more details regarding the derivation of the supremizer stabilization method one may see [15,17,27]. According to the latter references, in order to have a parameter independent velocity enrichment, an *approximated supremizer enrichment* procedure will be followed. This means that the supremizer problem is solved for each pressure snapshot $p(\mu)$ and a snapshots matrix of supremizer solutions is assembled:

$$\boldsymbol{S}_{\text{sup}} = [\boldsymbol{s}(\mu^1), \ldots, \boldsymbol{s}(\mu^{N_s})] \in \mathbb{R}^{N_u^h \times N_s}. \tag{23}$$

A proper orthogonal decomposition procedure is then applied to the resulting snapshots matrix in order to obtain a supremizer POD basis functions $\boldsymbol{\eta}_i$. Furthermore, the supremizer basis functions do not depend on the particular pressure basis functions but are computed during the offline phase, employing the pressure snapshots.

4. Numerical experiments

In the present section we will test the presented methodology on numerical tests investigating a steady Stokes flow around an embedded circular cylinder and a Stokes flow around a more complex geometry morphed using the Free Form Deformation. In the first two numerical example the embedded domain is parametrized through $\mu = (\mu_0, \mu_1)$ according to the following expression:

$$(x - \mu_0)^2 + (y - \mu_1)^2 \leq R^2,$$

where the two parameters μ_0 and μ_1 describe the x and y coordinates of the center of the embedded circular domain, see e.g. Fig. 2. We consider two different test cases, the first one consists of an 1D geometrical parametrization where the first parameter μ_0 is fixed. The second one consists of a 2D geometrical parametrization where both parameters μ_0 and μ_1 are left free. The problem domain is the rectangle $\mathcal{D} = [-2, 2] \times [-1, 1]$, in which a cylinder with constant radius 0.2 is embedded. The viscosity ν is set to 1. A constant velocity in the x direction, $u_{\text{in}} = 1$ is applied at the left side of the domain, and an open boundary condition with $\nabla \boldsymbol{u} \cdot \boldsymbol{n} = p_{\text{out}} = 0$ is applied on the right. A slip (no penetration) boundary condition is applied on the top and bottom edges. On the boundary of the embedded cylinder a no slip boundary condition is applied. The results for the test problems have been obtained with a mesh size of $h = 0.0350$ for the background mesh, see e.g. Fig. 2, using 15022 triangles for the discretization and $\mathbb{P}1/\mathbb{P}1$ finite elements in space with stabilization as described in [20]. All the numerical experiments have been tested with and without supremizer basis enrichment. Some representative snapshots are visualized in Fig. 3.

Both the full-order and the ROM simulation were run in serial on a personal computer with an Intel® Core™ i7-4770HQ 3.70 GHz CPU. In order to test the accuracy of the proposed method we compared the reduced order

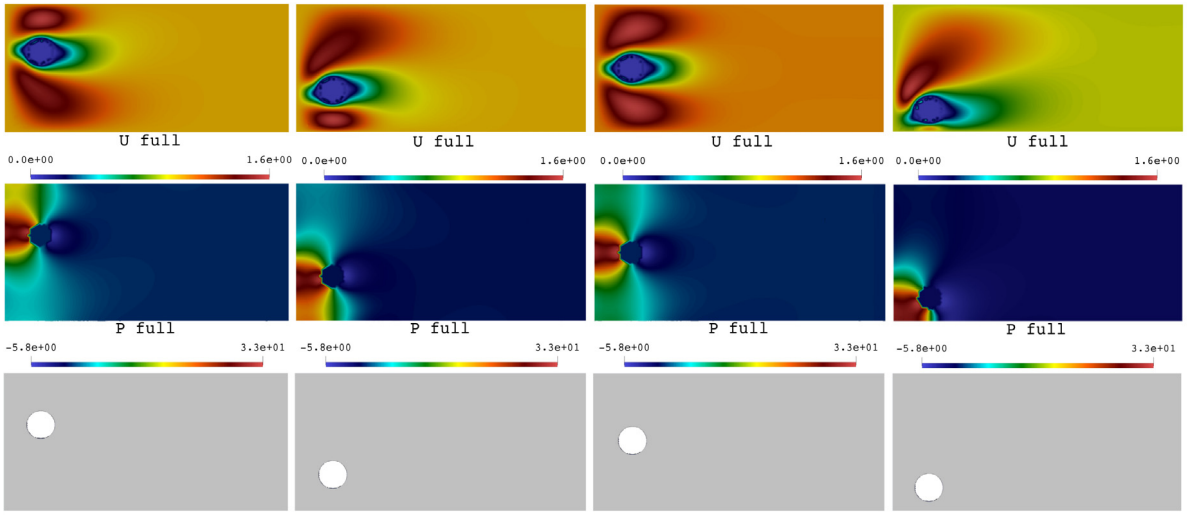

Fig. 3. Shifted Boundary method: the full-order solution snapshots for 4 different values of the input parameter in the case of 1D geometrical parametrization. In the figure we report the velocity $\boldsymbol{u}$ (first row), the pressure p (second row) and the true geometry (third row). Each column corresponds to a different parameter value $\mu_1 = [0.2439, -0.4764, 0.0233, 0.6468]$, while μ_0 is fixed to the value -1.5.

solutions against the full-order ones using 10 different samples of the parameters that were not previously tested during the training stage. In both cases the ROMs are constructed using the methodologies described in Section 3.

For the generation of the POD spaces, we considered 1024 full-order snapshots for the velocity and pressure fields. The snapshots are collected and $N_u^r = j$ velocity modes, $N_p^r = 6j$ pressure modes, and whenever we use supremizers $N_{\text{sup}}^r = 4j$ modes have been selected with $j = 8, 12, 16, 20, 25, 30, 35, 40, 45, 50$. We mention here that we chose to use a larger number of pressure and supremizer modes compared to the number of velocity modes after testing different configurations that we do not report here for sake of brevity. This decision was, in fact, taken after testing other cases e.g. $N_p^r = j, 2j, 3j, 4j, 6j$, $N_{\text{sup}}^r = j, 2j, 3j, 4j$ and its various combinations. We highlight here that a number of supremizer modes which is different with respect to the number of pressure modes is not a classical choice, since in the literature $N_p^r = N_{\text{sup}}^r$ is usually used. As mentioned in Section 3, it has been in fact heuristically verified [17,28] that, using an approximated supremizer approach, the minimum number of supremizer modes is equal to the number of pressure basis functions. In the present case, since at the reduced order level we are projecting also the stabilization term, the loss of stability given by velocity divergence free modes is partially circumvented by a "reduced version" of the stabilization terms.[1]

4.1. 1D geometrical parametrization

In the first test case the first parameter is fixed $\mu_0 = -1.5$ while μ_1 is left free and parametrized. The training set is chosen with an equally spaced distribution $\mu_1 \in [-0.65, 0.65]$. The ROM results are compared against full-order results, see e.g. Table 1 where the comparison is shown directly on the full-order and reduced-order velocity and pressure fields. In Fig. 4 we report the first four modes for the velocity, and the pressure fields.

4.1.1. The role of the supremizer stabilization

In this subsection we investigate the role of the "inf-sup" stabilization technique applied on an SBM stabilized full-order solver which, from the full-order point of view, allows the use of $\mathbb{P}1/\mathbb{P}1$ finite elements. The aim is

[1] We remark that this topic deserves further investigation. In fact, we notice that projection-based ROMs are quite sensitive with respect to the selection of the FOM stabilization technique. In [50], with a similar residual based SUPG stabilization, especially for the case of geometrical parametrization using a reference domain approach, we notice that the supremizer stabilization leads to a better approximation of the pressure field. On the other hand, when a variational multiscale stabilization method is used, we notice that the supremizer stabilization is not needed and the ROM coefficients multiplying the supremizer basis functions turned out to be zero [51].

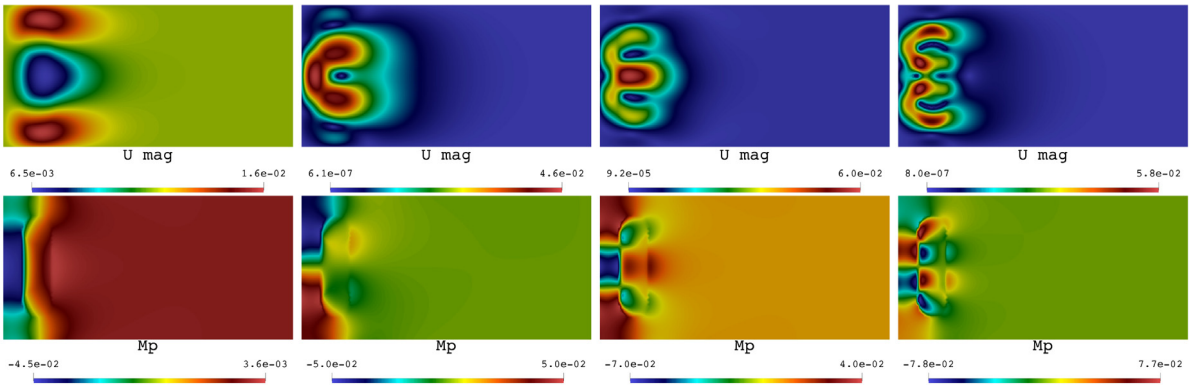

Fig. 4. The first 4 basis components for velocity (first row) and pressure (second row) in the 1D geometrical parametrization case with $\mu_1 \in [-0.65, 0.65]$.

to understand if the full-order stabilization is propagated also at the reduced order level. In general, considering incompressible flow applications, the quantity of interest is in both flow velocity and pressure. However, in many research contributions available in literature, due to the numerical instabilities given by spurious pressure modes, the pressure term is usually neglected [48]. In this work the attention is given to both velocity and pressure and in these first ROM-SBM experiments we want to test if the reduced version of the full-order Shifted Boundary Method solvers, without any additional stabilization technique, produce acceptable results.

The results in Table 1 and Fig. 8 (left plot) show the mean relative error $\|u - u_r\|_{L^2(\mathcal{D})}/\|u\|_{L^2(\mathcal{D})}$ and $\|p - p_r\|_{L^2(\mathcal{D})}/\|p\|_{L^2(\mathcal{D})}$ for a 10 samples testing set and for various different dimensions of the reduced basis spaces. The ROM has been generated starting from 1024 full-order snapshots with j velocity modes, $6j$ pressure modes with and without $4j$ supremizer modes. The comparison shows that even without an additional online stabilization approach, as mentioned in Section 3.1, the pressure relative error reaches acceptable values and has a better behavior respect to cases according to the classic FEM-ROM literature without pressure stabilization methods [17]. For the interested reader we refer for instance to [15]. Also in the case without supremizer stabilization, even with a relatively low number of modes, it is possible to observe good approximation properties for both the velocity and the pressure field.

The supremizer enrichment strategy permits to further reduce the approximation error of the pressure field, without increasing the approximation error of the velocity field. To summarize, in all the numerical experiments, the reduced velocity and pressure solutions reach a reasonable level of accuracy and in case with a supremizer enrichment stabilization it is possible to further decrease the approximation error of the pressure field without deteriorating the accuracy of the reduced velocity solution (see Fig. 5).

In Fig. 6 we report also the visualization of the full-order solutions, the reduced-order solutions and the absolute error for both the velocity and the pressure field in the case with a supremizer stabilization approach. The improved results for pressure are obvious and the plots clearly show at a glance that the FOM and ROM solutions cannot be easily distinguished (see Fig. 7).

4.1.2. Execution times

In this subsection we analyze the execution times of the online stage and we compare them against the full-order computational times. Indicatively the computational time necessary to compute 1024 solves of the full order problem is equal to 1 hour 43 minutes. The offline stage is usually very expensive but in a reduced order modeling framework, fortunately needs to be performed only once. In Table 2 we report the "online" computational time for different dimensions of the reduced order model. This consists of the time necessary for the assembly of the full-order matrices, the generation of the reduced order model and its resolution.

In Fig. 8 (right plot) we report the total execution time for the resolution of the reduced order problem corresponding to 10 solves of the reduced order and full-order solvers respectively. One full-order solve takes 3.716s. As one can see, the ROM leads to a considerable speed-up for all the different analyzed configurations and for both cases with and without supremizer enrichment.

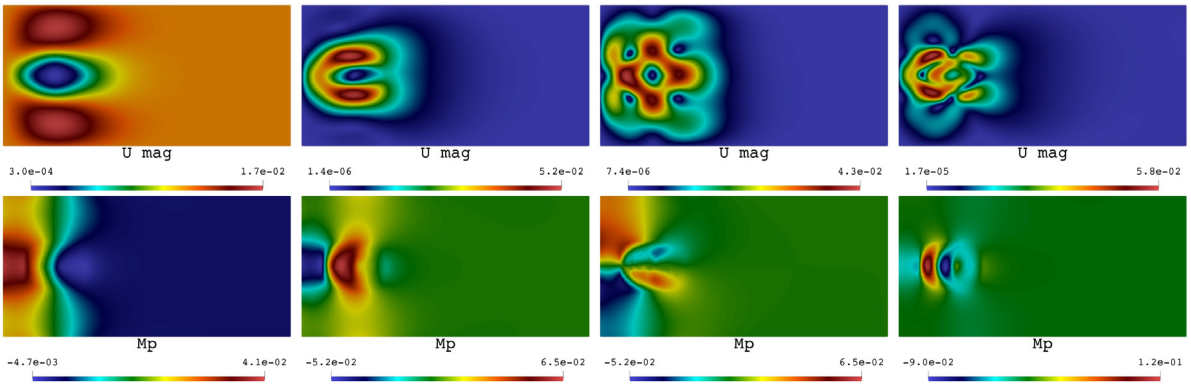

Fig. 5. The first 4 basis components for velocity (first row) and pressure (second row) in the 2D geometrical parametrization case with $\mu \in [-1.5, -1.0] \times [-0.15, 0.15]$.

Table 1
Relative error between the full-order solution and the reduced basis solution for velocity and pressure in the case of the 1D geometrical parametrization. Results are reported for different dimensions of the reduced basis spaces with and without supremizer stabilization.

Supremizers enrichment:	No		Yes	
Number of modes	Relative error u	Relative error p	Relative error u	Relative error p
8	0.0947158	12.309881	0.2406999	22.319781
12	0.0723268	12.133591	0.2078557	5.7159319
16	0.0610052	9.6652163	0.1692787	2.6962056
20	0.0538906	6.1692750	0.1243368	1.2535779
25	0.0434925	3.2331644	0.0770726	0.5568314
30	0.0396132	1.4693532	0.0437348	0.2504069
35	0.0298269	0.7455038	0.0262345	0.1356788
40	0.0177170	0.2918072	0.0121903	0.0611154
45	0.0085905	0.0923509	0.0060355	0.0330206
50	0.0053882	0.0473412	0.0046300	0.0279857

Table 2
Execution time, at the reduced order level, for the case with 1D geometrical parametrization. The computation time includes the assembling of the full-order matrices, their projection and the resolution of reduced problem. Times are for the resolution of 10 different values of the input parameter. The time execution at full-order level is equal to ≈37.16 s.

Number of modes	With no stabilization execution time (sec)	With stabilization execution time (sec)
8	7.3858961	7.710907
12	7.6042165	8.091225
16	7.9584049	8.290780
20	8.0206915	9.036709
25	8.2229143	9.495323
30	8.9529275	9.972288
35	9.0867916	10.47633
40	9.6555775	11.13931
45	9.8934008	11.49422
50	10.302459	11.92024

Remark 3. It is important to recall that in this case, since the interest is into testing the feasibility and the accuracy of a reduced order model constructed starting from a shifted boundary method full-order solver, we did not employ any hyper reduction technique. This means that, also at the reduced order level, we assembled the full-order discretized

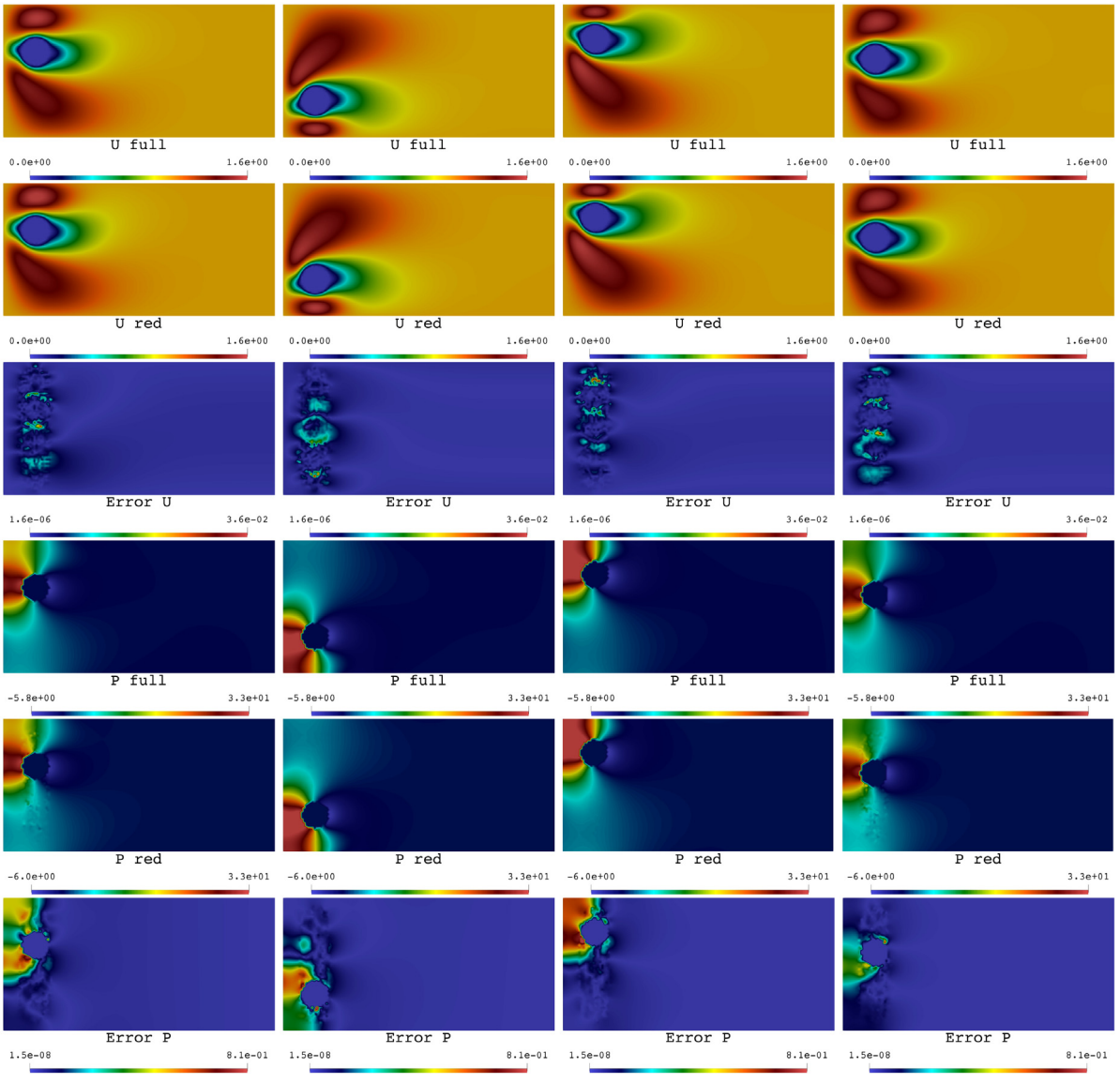

Fig. 6. Results for the 1D geometrical parametrization with $\mu_1 \in [-0.65, 0.65]$ and supremizer basis enrichment. In rows $1-3$ we report the full-order solution, the reduced order solution and the absolute error plots for the velocity field while in rows $4-6$ we report the same quantities for the pressure field. The different columns are for four different values of the input parameter, $\mu_1 = [0.2876, -0.4520, 0.4849, 0.1780]$, while μ_0 is fixed to the value -1.5.

differential operators. However, since most of the computational cost is required by the resolution of the discrete algebraic system rather than into its assembly, we can still achieve reasonable computational speedups.

4.2. 2D geometrical parametrization

The second case considers a 2D geometrical parametrization in the range $\mu = (\mu_0, \mu_1) = [-1.5, -1.0] \times [-0.15, 0.15]$. We perform this test to examine the performances of the methodology on a more complex and demanding scenario. It can be clearly seen in Fig. 9 (right plot) that the supremizer enrichment is pointed out necessary for convergence stability and reliable pressure results. Finally, we notice that, without a supremizer enrichment, in

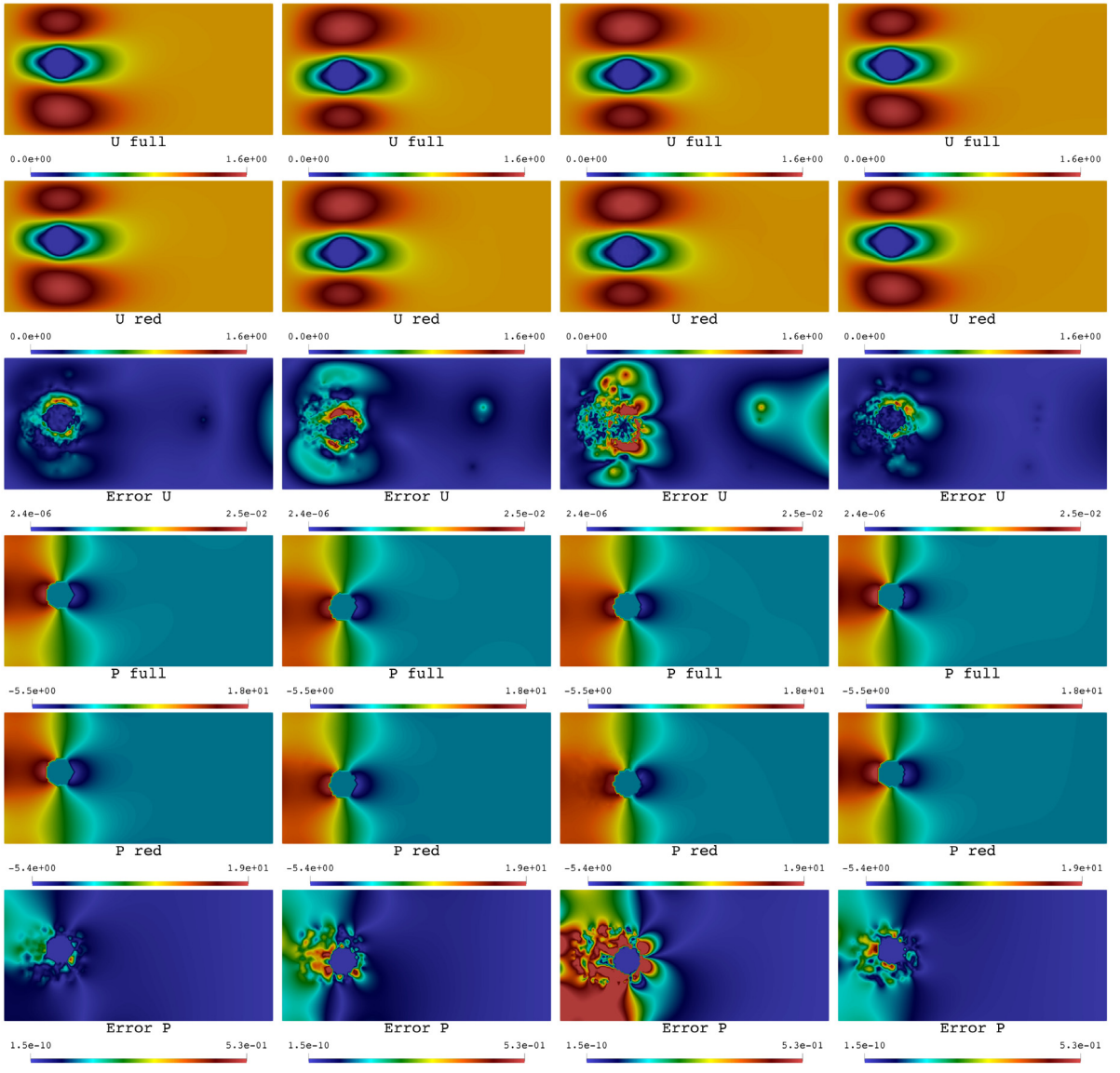

Fig. 7. Results for the 2D geometrical parametrization with $(\mu_0, \mu_1) \in [-1.5, -1.0] \times [-0.15, 0.15]$ and supremizer basis enrichment. In rows $1-3$ we report the full-order solution, the reduced order solution and the absolute error plots for the velocity field while in rows $4-6$ we report the same quantities for the pressure field. The different columns are for four different values of the input parameter $\mu = [(-1.1917, 0.0958), (-1.1095, -0.0958), (-1.0273, -0.0684), (-1.2191, 0.0684)]$.

this 2D geometrical parametrization case the best achieved relative error were equal to 0.0263821 and 0.1854861 for velocity and pressure respectively. The detailed results are omitted for shortness of space. In Table 3, for the supremizer stabilization case, we report the relative error for different dimensions of the reduced basis space and for different dimensions of the initial snapshots matrices (900 and 1024 full order solutions respectively). This test serves to investigate how the number of snapshots used to compute the reduced basis space affects the accuracy of the results and to show that it is very important to use a large number of snapshots especially for reliable pressure results. The table clearly shows that a larger number of initial snapshots increase considerably the computational accuracy of the method. This is given by the obvious fact that an increase of the number of snapshots leads to a better representation of the solution manifold and therefore to a reduced basis space with better approximation properties.

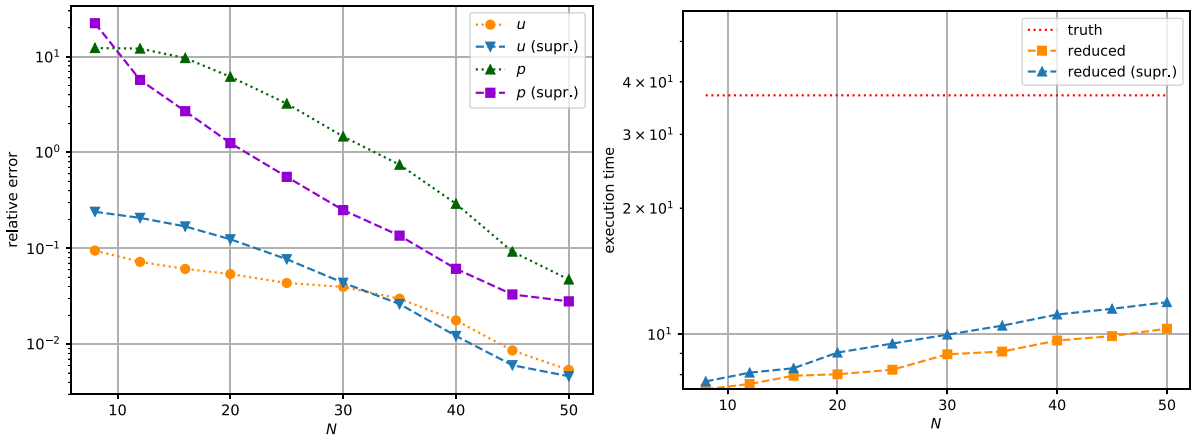

Fig. 8. Visualization of the results for the case with 1D geometrical parametrization. On the left plot are depicted the relative errors for velocity and pressure with and without supremizer stabilization. On the right plot we report the execution times of the reduced order problem for 10 parameters values with and without supremizers enrichment. In both plots, results are reported for various number of modes.

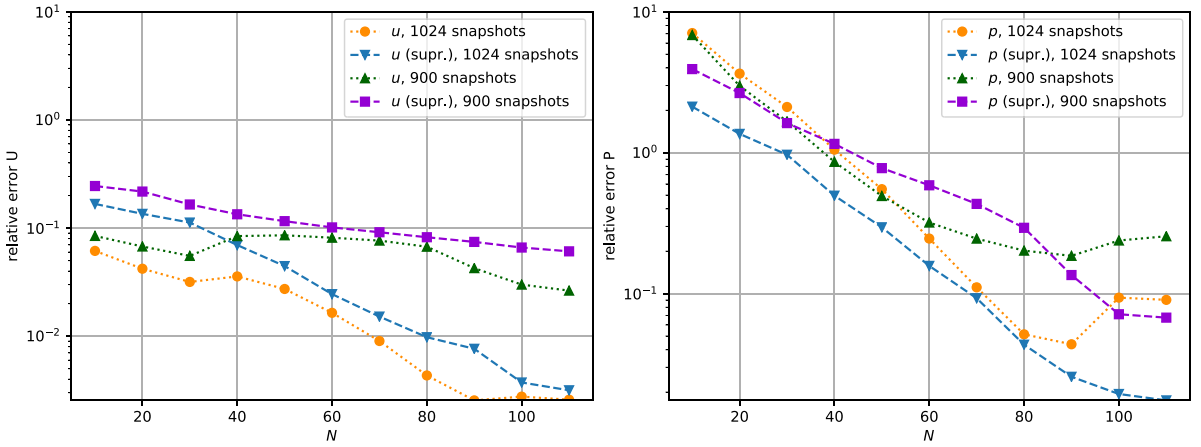

Fig. 9. Visualization of the results for the case with 2D geometrical parametrization. On the left plot are depicted the relative errors for the velocity field with and without supremizer stabilization and for a different number of initial snapshots (900 and 1024). On the right plot the same results are reported for the pressure field. In both plots results are reported for various number of modes.

4.3. Free-Form Deformation geometrical parametrization

In this last numerical example we apply the developed methodology to deal with a more complex geometrical parametrization. The Free-Form Deformation (FFD) tool is used to morph the original undeformed geometry [52]. The Free-Form Deformation is particularly useful in the context of parametrized reduced order model [53] and has been used to deal with complex parametrized domains in both automotive and naval engineering [54,55]. In this numerical example, since we wanted to investigate the applicability of the proposed method to more complex geometrical transformation we start from a simple initial shape and we deform it using free-form deformation.

The undeformed geometry is given by the sketch reported in the left side of Fig. 10. The embedded boundary consists into a circular and a square boundary. The circular domain is deformed using an FFD approach following the sketch reported in the right side of Fig. 10. The free form deformation is based on the definition of a bounding box identified by a set of properly chosen control points. For the particular case the movement of two control points has been parametrized by the parameter vector $\mu = (\mu_1, \mu_2)$ that describes the horizontal displacement of control points located on the left of the bounding box (see Fig. 10). In Fig. 12 we report the convergence analysis for the numerical example. The test has been conducted using an equal number of modes for velocity, pressure and supremizer space

Table 3
Supremizer basis enrichment and the relative error between full-order solution and reduced basis solution for velocity and pressure for two different numbers of snapshots in the case with a 2D geometrical parametrization.

Number of snapshots:	900		1024	
Number of modes	Relative error u	Relative error p	Relative error u	Relative error p
10	0.2448511	3.9240637	0.1672753	2.1243228
20	0.2175821	2.6531343	0.1353706	1.3611011
30	0.1652331	1.6234701	0.1124619	0.9680506
40	0.1340978	1.1560352	0.0696437	0.4958605
50	0.1158443	0.7777786	0.0444991	0.2958338
60	0.1013961	0.5876048	0.0244793	0.1574037
70	0.0914650	0.4335489	0.0151749	0.0928402
80	0.0822658	0.2933336	0.0097848	0.0434299
90	0.0744696	0.1355488	0.0076431	0.0257060
100	0.0660493	0.0714350	0.0037280	0.0194051
110	0.0609040	0.0675720	0.0031577	0.0174815

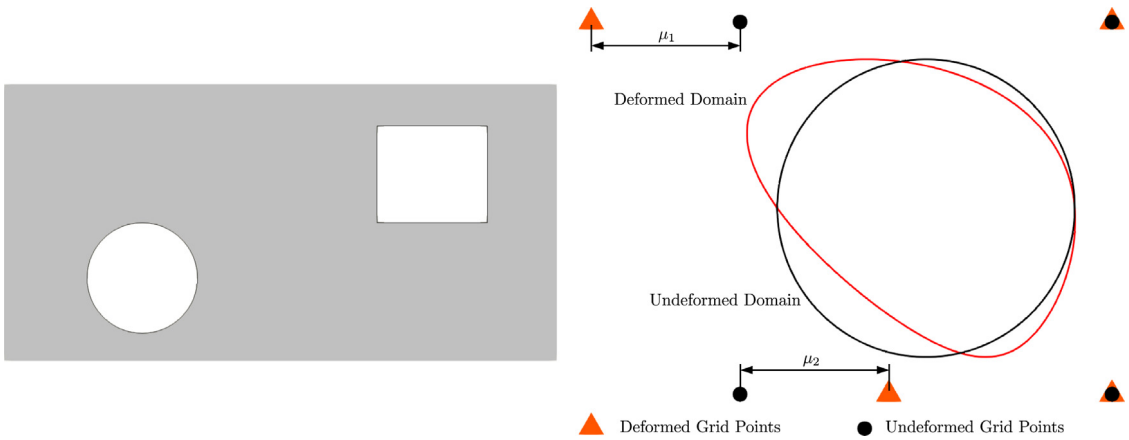

Fig. 10. Sketch of the undeformed embedded domain and parametrization used to morph the circular domain.

Fig. 11. Free-Form deformation geometrical parametrization and visualization of four samples $\mu = [(0.2613, -0.4698), (0.2613, -0.4698), (0.2613, -0.4698), (0.2613, -0.4698)]$.

($N_u^r = N_p^r = N_s^r = 10j$). In Fig. 11 we report the deformed geometry for different samples inside the parameter space. The reduced order model has been trained using 1024 samples chosen randomly inside the parameter space $\mathcal{P}_{\text{train}} \in [-0.6, 0.6]^2$ and tested using 100 samples inside the parameter space $\mathcal{P}_{\text{test}} \in [-0.5, 0.5]^2$.

4.4. Some comments

The numerical experiments clearly show that a projection based reduced order model can be generated on top of a full-order SBM solver. The important aspect is that, especially for complex geometrical parametrization, one can avoid the somehow "expensive" traditional approach of the geometrical transformation to a reference domain and to only rely on a suitable smooth enough background mesh. The experiments have shown that it is possible to obtain accurate results for velocity and pressure even without a supremizer "inf-sup"-stabilization at the reduced level.

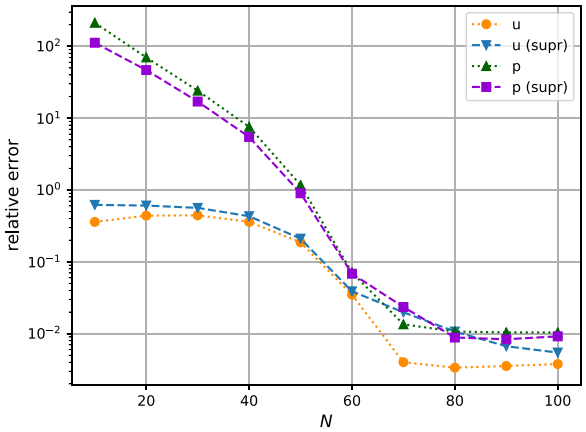

Fig. 12. Results for the numerical example using Free-Form deformation. The relative errors for velocity and pressure with and without supremizer stabilization are depicted for various number of modes.

After comparing the results in Figs. 9 and 8 one can observe that the supremizer stabilization permits to considerably increase the accuracy of the ROM for the pressure field without deteriorating the velocity accuracy. In the case of 1D geometrical parametrization it even permits to increase the accuracy for both the velocity and the pressure field.

5. Conclusions and future developments

In the present work a POD-Galerkin ROM based on SBM full-order simulations was presented. The ROM was developed to be consistent with the full-order model, and both velocity and pressure fields were considered. The reduced order method is applied to approximate the geometrically parametrized Stokes flow around a circular embedded domain. In particular, the focus and originality of the present work stands on the application of a velocity-pressure ROM using a background mesh in the EBM context. A comparison of accuracy was made for simulations with and without pressure stabilization strategies for POD-Galerkin ROMs in combination with the SBM.

The proposed ROM employs a stabilization strategy which is applied in the full order discretization formula. This stabilization proved to be efficient and effective also for the resulting reduced system for both the velocity and pressure fields. However, best accuracy for the pressure fields was achieved by further stabilizing the reduced system using a supremizer stabilization strategy. The attention was in fact devoted to analyze the applicability of the proposed methods to flows. The ROM demonstrated to be capable of reproducing all the main features of the physical phenomenon in an accurate manner leading to a considerable computational reduction.

Future developments will focus on different efficient methodologies for the affine decomposition of the SBM differential operator and in particular to study the applicability of well-known hyper reduction techniques, such as the empirical interpolation method, in the context of an SBM framework. Of future interest are also fluid–structure interaction applications as the ones presented in [56] and Navier–Stokes problems.

Acknowledgments

We acknowledge Dr Andrea Mola and Dr Francesco Ballarin from SISSA at International School for Advanced Studies, Mathematics Area, for fruitful discussions related to software implementation and RB methods, respectively. This work is supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research under Early Career Research Program Grant SC0012169, the U.S. Office of Naval Research under grant N00014-14-1-0311, ExxonMobil Upstream Research Company (Houston, TX), the European Research Council Executive Agency by means of the H2020 ERC Consolidator Grant project AROMA-CFD “Advanced Reduced Order Methods with Applications in Computational Fluid Dynamics” — GA 681447, (PI: Prof. G. Rozza), INdAM-GNCS 2018 and by project FSE – European Social Fund – HEaD “Higher Education and Development” SISSA operazione 1, Regione Autonoma Friuli — Venezia Giulia.

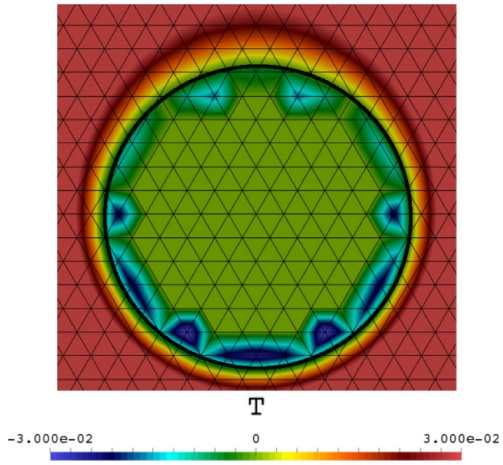

Fig. 13. A zoom onto the cylinder in order to show the smoothing procedure employed by the SBM method inside the ghost area.

References

[1] C. Hirt, A. Amsden, J. Cook, An arbitrary Lagrangian-Eulerian computing method for all flow speeds, J. Comput. Phys. 14 (3) (1974) 227–253.
[2] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (1) (2005) 239–261.
[3] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. Part I: Poisson and Stokes problems, J. Comput. Phys. 372 (2018) 972–995.
[4] Y. Bazilevs, K. Takizawa, T. Tezduyar, Computational Fluid-Structure Interaction: Methods and Applications, John Wiley & Sons, 2013.
[5] F. Ballarin, A. D'Amario, S. Perotto, G. Rozza, A POD-selective inverse distance weighting method for fast parametrized shape morphing, 2017, arXiv preprint arXiv:1710.09243.
[6] G. Stabile, G. Rozza, Efficient Geometrical parametrization for finite-volume based reduced order methods, submitted for publication, 2018.
[7] C.S. Peskin, Flow patterns around heart valves: A numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.
[8] J. Hesthaven, G. Rozza, B. Stamm, Certified Reduced Basis Methods for Parametrized Partial Differential Equations, in: Springer Briefs in Mathematics, Springer International Publishing, 2016.
[9] A. Quarteroni, A. Manzoni, F. Negri, Reduced Basis Methods for Partial Differential Equations, Springer International Publishing, 2016.
[10] F. Chinesta, A. Huerta, G. Rozza, K. Willcox, Model Order Reduction, Encyclopedia of Computational Mechanics, second ed., American Cancer Society, 2017, pp. 1–36.
[11] P. Benner, M. Ohlberger, A. Patera, G. Rozza, K. Urban, Model Reduction of Parametrized Systems, in: MS&A Series, vol. 17, Springer, 2017.
[12] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. Part II: Linear advection–diffusion and incompressible Navier–Stokes equations, J. Comput. Phys. 372 (2018) 996–1026.
[13] T. Song, A. Main, G. Scovazzi, M. Ricchiuto, The shifted boundary method for hyperbolic systems: Embedded domain computations of linear waves and shallow water flows, J. Comput. Phys. 369 (2018) 45–79.
[14] M. Balajewicz, C. Farhat, Reduction of nonlinear embedded boundary models for problems with evolving interfaces, J. Comput. Phys. 274 (2014) 489–504.
[15] G. Rozza, K. Veroy, On the stability of the reduced basis method for Stokes equations in parametrized domains, Comput. Methods Appl. Mech. Engrg. 196 (7) (2007) 1244–1260.
[16] G. Rozza, Reduced basis methods for Stokes equations in domains with non-affine parameter dependence, Comput. Vis. Sci. 12 (1) (2009) 23–35.
[17] F. Ballarin, A. Manzoni, A. Quarteroni, G. Rozza, Supremizer stabilization of POD-Galerkin approximation of parametrized steady incompressible Navier-Stokes equations, Internat. J. Numer. Methods Engrg. 102 (5) (2014) 1136–1161.
[18] G. Rozza, D.B.P. Huynh, A. Manzoni, Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants, Numer. Math. 125 (1) (2013) 115–152.
[19] G. Rozza, D. Huynh, A. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: Application to transport and continuum mechanics, Arch. Comput. Methods Eng. 15 (3) (2008) 229–275.
[20] T.J. Hughes, L.P. Franca, A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: Symmetric formulations that converge for all velocity/pressure spaces, Comput. Methods Appl. Mech. Engrg. 65 (1) (1987) 85–96.
[21] T. Hughes, L. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Comput. Methods Appl. Mech. Engrg. 59 (1) (1986) 85–99.

[22] G. Rozza, Reduced basis methods for elliptic equations in subdomains with a-posteriori error bounds and adaptivity, Appl. Numer. Math. 55 (4) (2005) 403–424.
[23] M. Grepl, A. Patera, A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations, ESAIM Math. Model. Numer. Anal 39 (1) (2005) 157–181.
[24] K. Veroy, C. Prud'homme, A. Patera, Reduced-basis approximation of the viscous Burgers equation: rigorous a posteriori error bounds, C. R. Math. 337 (9) (2003) 619–624.
[25] M. Grepl, Y. Maday, N. Nguyen, A. Patera, Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations, ESAIM Math. Model. Numer. Anal 41 (3) (2007) 575–605.
[26] A. Caiazzo, T. Iliescu, V. John, S. Schyschlowa, A numerical investigation of velocity-pressure reduced order models for incompressible flows, J. Comput. Phys. 259 (2014) 598–616.
[27] A. Gerner, K. Veroy, Certified reduced basis methods for parametrized saddle point problems, SIAM J. Sci. Comput. 34 (5) (2012) A2812–A2836.
[28] G. Stabile, G. Rozza, Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier-Stokes equations, Comput. Fluids 173 (2018) 273–284.
[29] G. Stabile, S. Hijazi, A. Mola, S. Lorenzi, G. Rozza, POD-Galerkin reduced order methods for CFD using finite volume discretisation: vortex shedding around a circular cylinder, Commun. Appl. Ind. Math. 8 (1) (2017) 210–236.
[30] A. Iollo, S. Lanteri, J.-A. Désidéri, Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations, Theor. Comput. Fluid Dyn. 13 (6) (2000) 377–396.
[31] I. Akhtar, A. Nayfeh, C. Ribbens, On the stability and extension of reduced-order Galerkin models in incompressible flows, Theor. Comput. Fluid Dyn. 23 (3) (2009) 213–237.
[32] M. Bergmann, C.-H. Bruneau, A. Iollo, Enablers for robust POD models, J. Comput. Phys. 228 (2) (2009) 516–538.
[33] S. Sirisup, G. Karniadakis, Stability and accuracy of periodic flow solutions obtained by a POD-penalty method, Physica D 202 (3–4) (2005) 218–237.
[34] L. Fick, Y. Maday, A.T. Patera, T. Taddei, A stabilized POD model for turbulent flows over a range of Reynolds numbers: Optimal parameter sampling and constrained projection, J. Comput. Phys. 371 (2018) 214–243.
[35] E.N. Karatzas, G. Stabile, N. Atallah, G. Scovazzi, G. Rozza, A reduced order approach for the embedded shifted boundary FEM and a heat exchange system on parametrized geometries, International Union of Theoretical and Applied Mechanics, Symposium on Model order reduction of coupled systems, in press (2018), arXiv preprint arXiv:1807.07753.
[36] A. Quarteroni, G. Rozza, Numerical solution of parametrized Navier–Stokes equations by reduced basis methods, Numer. Methods Partial Differential Equations 23 (4) (2007) 923–948.
[37] I. Kalashnikova, M.F. Barone, On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment, Internat. J. Numer. Methods Engrg. 83 (10) (2010) 1345–1375.
[38] F. Chinesta, P. Ladeveze, E. Cueto, A short review on model order reduction based on proper generalized decomposition, Arch. Comput. Methods Eng. 18 (4) (2011) 395.
[39] A. Dumon, C. Allery, A. Ammar, Proper general decomposition (PGD) for the resolution of Navier–Stokes equations, J. Comput. Phys. 230 (4) (2011) 1387–1407.
[40] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics, SIAM J. Numer. Anal. 40 (2) (2002) 492–515.
[41] E.N. Karatzas, F. Ballarin, G. Rozza, Projection-based reduced order models for a cut finite element method in parametrized domains, submitted for publication, 2018.
[42] M. Barrault, Y. Maday, N. Nguyen, A. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations, C. R. Math. 339 (9) (2004) 667–672.
[43] K. Carlberg, C. Farhat, J. Cortial, D. Amsallem, The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows, J. Comput. Phys. 242 (2013) 623–647.
[44] R. Everson, L. Sirovich, Karhunen-Loève procedure for gappy data, J. Opt. Soc. Amer. A 12 (8) (1995) 1657–1664.
[45] D. Xiao, F. Fang, A. Buchan, C. Pain, I. Navon, J. Du, G. Hu, Non linear model reduction for the Navier–Stokes equations using residual DEIM method, J. Comput. Phys. 263 (2014) 1–18.
[46] F. Brezzi, K.J. Bathe, A discourse on the stability conditions for mixed finite element formulations, Comput. Methods Appl. Mech. Engrg. 82 (1–3) (1990) 27–57.
[47] D. Boffi, F. Brezzi, M. Fortin, Mixed Finite Element Methods and Applications, first ed., Springer-Verlag, Berlin, Heidelberg, 2013.
[48] B.R. Noack, P. Papas, P.A. Monkewitz, The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows, J. Fluid Mech. 523 (2005) 339–365.
[49] A. Quarteroni, G. Rozza, Numerical solution of parametrized Navier–Stokes equations by reduced basis methods, Numer. Methods Partial Differential Equations 23 (4) (2007) 923–948.
[50] S. Ali, F. Ballarin, G. Rozza, Stabilized reduced basis methods for parametrized steady Stokes and Navier-Stokes equations, submitted for publication, 2018.
[51] G. Stabile, F. Ballarin, G. Zuccarino, G. Rozza, A reduced order variational multiscale approach for turbulent flows, (2018) arXiv preprint arXiv:1809.11101.
[52] T. Sederberg, S. Parry, Free-form deformation of solid geometric models, in: Proceedings of SIGGRAPH - Special Interest Group on GRAPHics and Interactive Techniques, SIGGRAPH, 1986, pp. 151–159.
[53] T. Lassila, G. Rozza, Parametric free-form shape design with PDE models and reduced basis method, Comput. Methods Appl. Mech. Engrg. 199 (23–24) (2010) 1583–1592.

[54] F. Salmoiraghi, A. Scardigli, H. Telib, G. Rozza, Free-form deformation, mesh morphing and reduced-order methods: enablers for efficient aerodynamic shape optimisation, Int. J. Comput. Fluid Dyn. 0 (0) (2018) 1–15, http://dx.doi.org/10.1080/10618562.2018.1514115.

[55] M. Tezzele, F. Salmoiraghi, A. Mola, G. Rozza, Dimension reduction in heterogeneous parametric spaces with application to naval engineering shape design problems, Adv. Model. Simul. Eng. Sci. 5 (1) (2018) 25, http://dx.doi.org/10.1186/s40323-018-0118-3.

[56] G. Stabile, H.G. Matthies, C. Borri, A novel reduced order model for vortex induced vibrations of long flexible cylinders, Ocean Eng. 156 (2018) 191–207.