A REDUCED BASIS APPROACH FOR PDES ON PARAMETRIZED
GEOMETRIES BASED ON THE SHIFTED BOUNDARY FINITE ELEMENT
METHOD AND APPLICATION TO FLUID DYNAMICS

EFTHYMIOS N. KARATZAS1,*, GIOVANNI STABILE1, LEO NOUVEAU†, GUGLIELMO SCOVAZZI2, AND GIANLUIGI ROZZA1

Abstract. We propose a model order reduction technique integrating the Shifted Boundary Method
(SBM) with a POD-Galerkin strategy. This approach allows to treat more 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, an unfitted boundary
method that avoids remeshing and the tedious handling of cut cells by introducing an approximate
surrogate boundary.

Second, the computational effort is further 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 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.

1. Introduction and Motivation

When numerically simulating problems characterized by deforming domains, it is mainly possible to
rely on two different approaches: one based on an arbitrary Lagrangian-Eulerian (ALE) formulation
[22], where the grid is deformed using a mesh motion algorithm, or one based on an embedded boundary
approach, which relies on an undeformed background mesh into which the boundary is embedded or
immersed (see [31] 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 [29].

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 [6, 3, 47]. Starting with the
pioneering work of Peskin [33], 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 fast computing times are required - as in
real-time control problems. A possible way to overcome this limitation is the use of reduced order
modeling techniques [21, 34, 13, 7]. In particular we rely onto reduced basis methods.

It is behind the scope of this paper to provide a comprehensive review on immersed/embedded
methods and we report here only some of the most important contributions on the topic. The reader
interested on more details on the different existing approaches, namely Ghost-Cell finite difference

1SISSA, International School for Advanced Studies, Mathematics Area, mathLab Trieste, Italy.
2Civil and Environmental Engineering, Duke University, Durham, NC 27708, United States.
E-mail addresses: efthymios.karatzas@sissa.it, gstabile@sissa.it, leo.nouveau@duke.edu,
guglielmo.scovazzi@duke.edu, grozza@sissa.it.
2010 Mathematics Subject Classification. 78M34, 97N40, 35Q35.
*Corresponding Author.

arXiv:1807.07790v2 [math.NA] 26 Jul 2018
methods, Cut-Cell finite volume approach, Immersed Interface, Ghost Fluid, Volume Penalty methods can refer to the review paper [31] and references therein.

In [29, 30], a new Embedded Boundary Method (EBM) - called Shifted Boundary Method (SBM) - was introduced and applied to heat transfer problems, the advection-diffusion equations, the Stokes flow and the laminar and turbulent Navier-Stokes equations. The idea of the SBM is to shift the position of boundaries from their true to a surrogate location 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 [29, 30, 43].

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 [2] for classical EBM 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. [21, 41, 38, 4, 40, 39, 7] 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 it follows: In § 2 we introduce the formulation and the methods used for the full-order approximation of the equations. In § 3 the reduced order methodology is discussed in details, while in § 4 and in A the proposed ROM technique is tested on two different numerical benchmarks, dealing with the geometrical parametrized problem of the flow around a circular embedded domain, and a Poisson problem on a parametrized geometry, respectively. Finally in § 5 conclusions and perspectives are drawn by highlighting future improvements and developments.

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 in a domain when convective forces are negligible with respect to viscous forces. Consider an open domain $\mathcal{D}$ in $\mathbb{R}^d$, with $d=2,3$ the number of space dimensions with Lipschitz boundary $\Gamma$ (split into two sub-boundaries $\Gamma_D, \Gamma_N$) 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 $\Gamma_D$ and $\Gamma_N$ respectively, geometrically parameterized by $\mu$, is given by:

$$
-\nabla \cdot (2\nu \epsilon(\mathbf{u}(\mu)) - p(\mu)\mathbf{I}) = \mathbf{g}(\mu), \quad \text{in } \mathcal{D}(\mu),
$$

$$
\nabla \cdot \mathbf{u}(\mu) = 0, \quad \text{in } \mathcal{D}(\mu),
$$

$$
\mathbf{u}(\mu) = \mathbf{g}_D(\mu), \quad \text{on } \Gamma_D(\mu),
$$

$$
(2\nu \epsilon(\mathbf{u}(\mu)) - p(\mu)\mathbf{I}) \cdot \mathbf{n} = \mathbf{g}_N(\mu), \quad \text{on } \Gamma_N(\mu).
$$

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

2.2. Full-order parametrized Nitsche shifted boundary variational weak formulation. In the following, we recall the full-order Shifted Boundary Method for the steady Stokes equations following [29]. For the sake of simplicity, in this subsection, we will omit the parameter dependency with respect to $\mu$. We also denote $\nu := \nu_h$.

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 any cut cells. As
In the following, whenever we write $\tau$ on the boundary conditions on the value problem with a mesh $\tilde{\Gamma}$ introduce the SBM variational formulation. Consider a discretization of the continuous boundary $\Gamma$ boundary alternative choices are possible. Considering the standard notation (23) satisfies these assumptions, and will be used in what follows, although introduce the discrete spaces $V$ For the piecewise linear spaces convergent base formulation for the Stokes flow exists for these spaces in the case of conformal grids.

seen in Figure 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 $\Gamma$. The closest faces/edges of $\tilde{\Gamma}$ to $\Gamma$ 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, $\tilde{n}$ indicates the unit outward-pointing normal to the surrogate boundary $\tilde{\Gamma}$, and it differs from the outward-pointing normal $n$ of $\Gamma$. Notice also that the closest-point projection, in spite of the segmented/faceted nature of the surrogate boundary $\tilde{\Gamma}$, is actually a smooth mapping $M$ from points in $\tilde{\Gamma}$ to points in $\Gamma$:

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

The mapping $M$ defines a distance vector function

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

where the distance vector $d = ||d||n$ is aligned along $n$, due to the closest point projection properties, see Figure 1. The unit normal vector $n$ and the unit tangential vectors $\tau_i$ ($1 < i < d - 1$) to the boundary $\Gamma$ can be easily extended to the boundary $\tilde{\Gamma}$ since $\tilde{n}(\tilde{x}) \equiv n(M(\tilde{x})), \tau_i(x) \equiv \tau_i(M(\tilde{x}))$.

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

Under the important assumption $n \cdot \tilde{n} \geq 0$, 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 $\tilde{\mathcal{D}}$ consisting of simplices $K$ belonging to a tessellation $\mathcal{T}$. Moreover, we introduce the discrete spaces $V_h$ 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

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

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

the stabilized formulation of [23] satisfies these assumptions, and will be used in what follows, although alternative choices are possible. Considering the standard notation $\langle \cdot, \cdot \rangle_{\mathcal{D}}, \langle \cdot, \cdot \rangle_{\tilde{\mathcal{D}}}, \langle \cdot, \cdot \rangle_{\mathcal{D}}$ for the $L^2(\mathcal{D}), L^2(\tilde{\mathcal{D}})$ and $L^2(\tilde{\mathcal{D}})$ inner products onto the surrogate geometry $\tilde{\mathcal{D}}, \tilde{\mathcal{D}}$ and $\tilde{\mathcal{D}}$, 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 \( \mathbf{u} \in \mathbf{V}_h \) and \( p \in \mathbf{Q}_h \) such that, \( \forall \mathbf{w} \in \mathbf{V}_h \) and \( \forall q \in \mathbf{Q}_h \),
\[
\begin{align*}
\{ & a(\mathbf{w}, \mathbf{u}; \mu) + b(\mathbf{w}, p; \mu) = \ell_g(\mathbf{w}; \mu), \\
& b(\mathbf{u}, q; \mu) + \hat{b}(\mathbf{u}, q; \mu) + \text{(stab. term.)} = \ell_q(q; \mu),
\end{align*}
\]
where
\[
\begin{align*}
(2) \quad a(\mathbf{w}, \mathbf{u}; \mu) &= (\epsilon(\mathbf{w}), 2\nu \epsilon(\mathbf{u}))_{\mathbf{D}_1(\mu)} - (\mathbf{w} \cdot \tilde{n}, 2\nu \epsilon(\mathbf{u}))_{\mathbf{F}_D(\mu)} - (2\nu \epsilon(\mathbf{w}), (\mathbf{w} + (\nabla \mathbf{w}) \mathbf{d}) \otimes \tilde{n})_{\mathbf{F}_D(\mu)} \\
&\quad + \alpha (2\nu/h(\mathbf{w} + (\nabla \mathbf{w}) \mathbf{d}), \mathbf{u} + (\nabla \mathbf{u}) \mathbf{d})_{\mathbf{F}_D(\mu)} + \beta (2\nu \nabla \cdot \mathbf{w}, \nabla \cdot \mathbf{u})_{\mathbf{F}_D(\mu)}, \\
(3) \quad b(\mathbf{w}, p; \mu) &= -(\nabla \cdot \mathbf{w}, p)_{\mathbf{D}_1(\mu)} + (\mathbf{w} \cdot \tilde{n}, p)_{\mathbf{F}_D(\mu)}, \\
(4) \quad \ell_g(\mathbf{w}; \mu) &= -(\mathbf{w}, \mathbf{g})_{\mathbf{D}_1(\mu)} + (\mathbf{w}, g_{\nu, \mu})_{\mathbf{F}_D(\mu)} - (2\nu \epsilon(\mathbf{w}), \tilde{\mathbf{g}}_{\mu} \otimes \tilde{n})_{\mathbf{F}_D(\mu)} \\
&\quad + \alpha (2\nu/h(\mathbf{w} + (\nabla \mathbf{w}) \mathbf{d}), \tilde{\mathbf{g}}_{\mu})_{\mathbf{F}_D(\mu)}, \\
(5) \quad \hat{b}(\mathbf{u}, q; \mu) &= \langle \mathbf{q} \mathbf{d} \otimes \tilde{n}, \nabla \mathbf{u} \rangle_{\mathbf{F}_D(\mu)}, \\
(6) \quad \ell_q(q; \mu) &= \langle \mathbf{g}_{\mu} \mathbf{d} \cdot \tilde{n}, q \rangle_{\mathbf{F}_D(\mu)}.
\end{align*}
\]
These equations yield the following algebraic system of equations:
\[
\begin{bmatrix}
A(\mu) & B^T(\mu) \\
B(\mu) + B(\mu) & C(\mu)
\end{bmatrix}
\begin{bmatrix}
\mathbf{u}(\mu) \\
\mathbf{p}(\mu)
\end{bmatrix}
= \begin{bmatrix}
F_g(\mu) \\
F_q(\mu)
\end{bmatrix}.
\]
In the above system of equations it is important to highlight several factors that will play a crucial role in § 3. First of all, the discretized differential operators \( A, B \) and \( \hat{B} \) are parameter dependent. Secondly, in the typical saddle point structure of the Stokes problem, the incompressibility equation is partially relaxed adding a stabilization term \( C \). 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 \( P_1 - P_1 \), and, as reported in § 3, it helps to partially preserve the stability of the reduced order model.

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 § 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 [21, 37], 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 [39], linear parabolic equations in [20], and non-linear problems in [48, 19]. 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 [2].

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 more in general of the solution manifold, with a reduced number of basis functions. The method consist of a two-stage procedure: the offline and the online one. During the costly \textit{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 \textit{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 POD-Galerkin ROMs for the incompressible Stokes and Navier-Stokes equations are unstable [11, 4, 18, 41, 40, 46, 44], due to pressure instabilities, while for dynamic instabilities on transient problems see for instance [24, 1, 8, 42, 17]. More details will be given in § 3.3.1.

From a reduced order modeling point of view, the first aim of this work is to investigate how ROMs are applied on an SBM approach, and which are the perspectives. Since our main interest is to generate ROMs for flow problems on parametrized geometries we will consider the case of a Stokes flow problem. However, in A we present also some preliminary results obtained on a Poisson problem, which, originally, served as a simple feasibility test. For sake of brevity, we did not introduce the derivation of the ROM for the Poisson problem and for the interested reader we refer to [27]. 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, [41, 35, 4].

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 [39, 13, 25, 34, 14, 15]. 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 \( \mu^k \in \mathcal{K} = \{ \mu^1, \dots, \mu^N_{\mu} \} \subset \mathcal{P} \) where \( \mathcal{K} \) is a finite dimensional training set of parameters 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 \( N^h_u, N^h_p \) for the velocity and pressure, respectively. The snapshots matrices \( S_u \) and \( S_p \), for velocity and pressure, are then given by \( N_s \) full-order snapshots:

\[
S_u = [u(\mu^1), \ldots, u(\mu^{N_s})] \in \mathbb{R}^{N_u \times N_s}, \quad S_p = [p(\mu^1), \ldots, p(\mu^{N_s})] \in \mathbb{R}^{N_p \times N_s}.
\]

Given a general scalar or vectorial function \( u : \mathcal{D} \to \mathbb{R}^d \), with a certain number of realizations \( u_1, \ldots, u_{N_u} \), the POD problem consists in finding, for each value of the dimension of POD space \( N_{POD} = 1, \ldots, N_s \), the scalar coefficients \( a_1^1, \ldots, a_1^{N_u}, \ldots, a_{N_u}^{N_{POD}} \) and functions \( \varphi_1, \ldots, \varphi_{N_{POD}} \), that minimize the quantity:

\[
E_{N_{POD}} = \sum_{i=1}^{N_s} \sum_{k=1}^{N_{POD}} a_k^i \varphi_k \| u_i - \sum_{k=1}^{N_{POD}} a_k^i \varphi_k \|_{L^2(\mathcal{D})}^2, \quad \forall N_{POD} = 1, \ldots, N_s,
\]

with \( \varphi_i, \varphi_j \)\(_{\mathcal{D}} = \delta_{ij}, \quad \forall i, j = 1, \ldots, N_s \).

In this case the velocity field \( u \) is used as example. It can be shown [28] that the minimization problem of equation (9) is equivalent of solving the following eigenvalue problem:

\[
C^u Q^u = Q^u \Lambda^u, \quad \text{for } C^u_{ij} = (u_i, u_j)_{\mathcal{D}}, \ i, j = 1, \ldots, N_s,
\]

where \( C^u \) is the correlation matrix obtained starting from the snapshots \( S_u \). \( Q^u \) is a square matrix of eigenvectors and \( \Lambda^u \) is a diagonal matrix of eigenvalues.

The basis functions can then be obtained with:

\[
\varphi_i = \frac{1}{N_s \lambda_{ii}^{1/2}} \sum_{j=1}^{N_s} u_j Q^u_{ij},
\]

The same procedure can be repeated also for the pressure field considering the snapshots matrix consisting of the snapshots \( p_1, p_2, \ldots, p_{N_p} \). One can compute the correlation matrix of the pressure field snapshots \( C^p \) and solve a similar eigenvalue problem \( C^p Q^p = Q^p \Lambda^p \). The POD modes \( \chi_i \) for the pressure field can be computed with:

\[
\chi_i = \frac{1}{N_s \lambda_{pp}^{1/2}} \sum_{j=1}^{N_s} p_j Q^p_{ij}.
\]
The POD spaces are constructed for both velocity and pressure using the aforementioned methodology resulting in the spaces:

\[ L_u = [\varphi_1, \ldots, \varphi_{N_u}] \in \mathbb{R}^{N_u \times N_u}, \quad L_p = [\chi_1, \ldots, \chi_{N_p}] \in \mathbb{R}^{N_p \times N_p}. \]

where \( N_u, N_p < N_s \) are chosen according to the eigenvalue decay of \( \lambda^v \) and \( \lambda^p \), [39, 7].

**Remark 3.1.** 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 [2] the ghost area solution is set to zero. In the present work we use the solution values as they are computed using the shifted boundary method using the smooth map \( M \) 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 Figure 10. This approach guarantees a regular “solution” in the background domain and permits therefore 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 [26].

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 \( D(\mu) \) to a fixed reference domain \( D \) using a map \( M(\mu) : D \rightarrow D(\mu) \) see e.g. [21, 41, 38, 4, 40, 39, 7]. 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:

\[
\begin{aligned}
&\tilde{a}(\mathbf{w}, \mathbf{u}; \mu) + \tilde{b}(\mathbf{w}, p; \mu) = \tilde{e}(\mathbf{w}; \mu), \\
&\tilde{b}(\mathbf{u}, q; \mu) = 0,
\end{aligned}
\]

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

\[
\begin{aligned}
&\tilde{a}(\mathbf{w}, \mathbf{u}; \mu) = \int_D \epsilon(\mathbf{w})2\nu(J_T(\mu))^{-1}(J_T(\mu))^{-T}|J_T(\mu)|\epsilon(\mathbf{u})d\mathbf{x}, \\
&\tilde{b}(\mathbf{w}, p; \mu) = -\int_D \nabla \cdot \mathbf{w}(J_T(\mu))^{-1}|J_T(\mu)|p d\mathbf{x}, \\
&\tilde{e}(\mathbf{w}; \mu) = \int_D |J_T(\mu)|\mathbf{g} \cdot \mathbf{w} d\mathbf{x},
\end{aligned}
\]

and \( J_T \) is the Jacobian of the map \( M(\mu) \) and \( |J_T(\mu)| \) is its determinant. Moreover, in cases where it is possible to obtain a disjoint decomposition of the domain \( \{D_r\}_{r=1}^R \) such that for each \( r \in \{1, \ldots, R\} \) the map \( M_r(\mu) : D_r \rightarrow D_r(\mu) \) 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 Equation 1 are “naturally” parametrized through the surrogate boundary \( \tilde{F}(\mu) \) and the surrogate domain \( \tilde{D}(\mu) \). Moreover, it is still possible to rely on hyper reduction techniques such as the empirical interpolation method [5], the GNAT [12] or the Gappy-POD [16].

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:

\[
\begin{aligned}
&\mathbf{u}^r \approx \sum_{i=1}^{N_u} a_i(\mu) \varphi_i(x) = L_u a(\mu), \\
&p^r \approx \sum_{i=1}^{N_p} b_i(\mu) \chi_i(x) = L_p b(\mu).
\end{aligned}
\]

The reduced solution vectors \( \mathbf{a} \in \mathbb{R}^{N_u \times 1} \) and \( \mathbf{b} \in \mathbb{R}^{N_p \times 1} \) depend only on the parameter values and the basis functions \( \varphi_i \) and \( \chi_i \) depend only on the physical space. The unknown vectors of coefficients \( \mathbf{a} \) and \( \mathbf{b} \) then 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.

\[
\begin{bmatrix}
L_u & 0 \\
0 & L_p
\end{bmatrix}
\begin{bmatrix}
A(\mu) & B^T(\mu) \\
B(\mu) + \hat{B}(\mu) & C(\mu)
\end{bmatrix}
\begin{bmatrix}
L_u^T & 0 \\
0 & L_p^T
\end{bmatrix}
\begin{bmatrix}
u(\mu) \\
p(\mu)
\end{bmatrix} =
\begin{bmatrix}
L_u^T & 0 \\
0 & L_p^T
\end{bmatrix}
\begin{bmatrix}
F_u(\mu) \\
F_p(\mu)
\end{bmatrix},
\]

which leads to the following algebraic reduced system:

\[
\begin{bmatrix}
A^r(\mu) & B^r(\mu) \\
B^r(\mu) + \hat{B}^r(\mu) & C^r(\mu)
\end{bmatrix}
\begin{bmatrix}
a \\
b
\end{bmatrix} =
\begin{bmatrix}
F^r_u(\mu) \\
F^r_p(\mu)
\end{bmatrix},
\]

where \( A^r(\mu) \in N^r_u \times N^r_u, \ B^r(\mu), \hat{B}^r(\mu) \in N^r_p \times N^r_u, \ C^r(\mu) \in N^r_p \times N^r_p \) and \( F^r_u(\mu) \in N^r_u \times 1, \ F^r_p(\mu) \in N^r_p \times 1 \) are the reduced discretized operators and reduced forcing vector respectively. The dimension of the reduced operator, 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 much inexpensive to solve. We remark here that the full-order discretized differential operators that appear in Equation 7 are parameter dependent and therefore, also at the reduced order level, we need to assemble the problem in order to compute the reduced differential operator. Possible ways to avoid such potentially expensive operation, relying on an affine approximation of the full-order differential operator, could be to use hyper reduction techniques. \cite{49, 5, 12}. 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 § 4, it is anyway possible to achieve a good computational speedup. We remark here that, during the online stage, also the stabilization term \( C \) 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 § 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 \cite{10, 9}. It is required that there should exist a constant \( \beta > 0 \), independent to the discretization parameter \( h \), such that:

\[
\inf_0 \sup_{q \in Q_h} \frac{\langle \nabla \cdot v, q \rangle}{\| \nabla v \|} \geq \beta > 0.
\]

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 \((P_2 - P_1)\). In our case, at full-order level, as explained in § 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 \cite{41, 18, 4}. 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 \cite{32}, 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 \cite{4, 41, 36}.

Remark 3.2. 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 supemizer modes.

3.3.2. Supemizer enrichment. As mentioned in § 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 supemizer basis functions \( L_{sup} \) are computed and added to the reduced velocity space, which is transformed into \( \tilde{L}_u \), namely:

\[
L_{sup} = [\eta_1, \ldots, \eta_{N_{sup}}] \in \mathbb{R}^{N_u \times N_{sup}}, \quad \tilde{L}_u = [\varphi_1, \ldots, \varphi_{N_{sup}}, \eta_1, \ldots, \eta_{N_{sup}}] \in \mathbb{R}^{N_u \times (N_u + N_{sup})}.
\]

These basis functions are chosen solving a supemizer problem that ensures the fulfillment of a reduced version of the “inf-sup” condition. The supemizer solution \( s_i \), corresponding to the parameter value
μ, given a certain pressure basis function χ, 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:

\begin{equation}
\Delta s_i = -\nabla \chi_i \text{ in } D(\mu^i), \quad s_i = 0 \text{ on } \Gamma(\mu^i).
\end{equation}

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 [29]. For more details regarding the derivation of the supremizer stabilization method one may see [41, 18, 4]. 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(μ) and a snapshots matrix of supremizer solutions is assembled:

\begin{equation}
S_{sup} = [s(\mu^1), \ldots, s(\mu^N)] \in \mathbb{R}^{N_{h} \times N_{s}}.
\end{equation}

A proper orthogonal decomposition procedure is then applied to the resulting snapshots matrix in order to obtain a supremizer POD basis functions η. 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. The numerical examples include geometrical parametrization of the embedded domain. The embedded domain is in fact parametrized through μ = (μ₀, μ₁) according to the following expression:

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

where the two parameters μ₀ and μ₁ describe the x and y coordinates of the center of the embedded circular domain, see e.g. Figure 2. We consider two different test cases, the first one consists of an 1D geometrical parametrization where the first parameter μ₀ is fixed. The second one consists of a 2D geometrical parametrization where both parameters μ₀ and μ₁ are left free. The problem domain is the rectangle D = [-2, 2] × [-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_{in} = 1 is applied at the left side of the domain, and an open boundary condition with p_{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. Figure 2, using 15022 triangles for the discretization and P1/P1 finite elements in space with stabilization terms as described in [23]. Both experiments have been tested with and without supremizer basis enrichment. Some representative snapshots are visualized in Figure 3, setting the basement for the offline procedure and subsequently for the reduced basis creation. Both the full-order and the ROM simulation was run in serial on a personal computer with an Intel® Core™ i7-4770HQ 3.70GHz CPU. In order to test the accuracy of the proposed method we compared the reduced order solutions against the full-order ones.
Figure 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 \( u \) (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 \( \mu_0 \) is fixed to the value \(-1.5\).

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 §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 \( N_r^v = j \) velocity modes, \( N_r^p = 6j \) pressure modes, and whenever we use supremizers \( N_r^{sup} = 4j \) 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. \( N_r^p = j, 2j, 3j, 4j, 6j \), \( N_r^{sup} = j, 2j, 3j, 4j \) 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 \( N_r^p = N_r^{sup} \) is usually used. As mentioned in §3, it has been in fact heuristically verified \([4, 46]\) 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.

4.1. 1D geometrical parametrization. In the first test case the first parameter is fixed \( \mu_0 = -1.5 \) while \( \mu_1 \) 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 Figure 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 \( P1/P1 \) finite elements. The aim is 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 [32]. 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 Figure 8 (left plot) for the mean relative error \( ||u - u_r||_{L^2(D)}/||u||_{L^2(D)} \) and \( ||p - p_r||_{L^2(D)}/||p||_{L^2(D)} \) for a 10 samples training set and for various number of modes are examined. The ROM has been generated starting from 1024 full-order snapshots with \( j \) velocity modes, \( 6j \) pressure
modes and with and without 4$j$ supremizer modes. The comparison without and with the supremizer stabilization case shows that even without an additional online stabilization approach, as mentioned in § 3.1, the pressure relative error reaches acceptable values and has a better behavior than cases according to the classic FEM-ROM literature without pressure stabilization methods [4]. For the interested reader we refer for instance to [41]. 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. However, we observe also that the pressure field is approximated with a relatively larger error with respect to the velocity field.

However, 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.

In Figure 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.

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.
Figure 6. Results for the 1D geometrical parametrization with $\mu_1 \in [-0.65, 0.65]$ and with 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 $\mu_0$ is fixed to the value $-1.5$.

In Figure 8 (right plot) we report the total execution time for 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 the all different analyzed configurations and for both cases with and without supremizer enrichment.

Remark 4.1. 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 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
Figure 7. Results for the 2D geometrical parametrization with $(\mu_0, \mu_1) \in [-1.5, -1.0] \times [-0.15, 0.15]$ and with 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 $\mu = [(-1.1917, 0.0958), (-1.1095, -0.0958), (-1.0273, -0.0684), (-1.2191, 0.0684)]$.

in Figure 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 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 case with a supremizer stabilization technique, 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 many 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 fact that an increase of the number of snapshots leads to a better representation of the solution manifold and therefore to a reduced basis with better approximation properties. As it is natural, we see differences and small instability in convergence for the experiments without supremizer enrichment (green and orange lines in Figure 9), but in the cases with the pressure stabilization the
| 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 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 and with and without supremizer stabilization.

| Number of modes | with no stabilization | with stabilization |
|-----------------|-----------------------|--------------------|
|                 | execution time (sec)  | execution time (sec) |
| 8               | 7.3858961             | 7.710907           |
| 12              | 7.6042165             | 8.091225           |
| 16              | 7.9584049             | 8.290780           |
| 20              | 8.0260915             | 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           |

**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 $\approx 37.16$ sec.

**Figure 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.
Figure 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.

| Number of snapshots: | Number of modes | relative error $u$ | relative error $p$ | relative error $u$ | relative error $p$ |
|----------------------|-----------------|-------------------|-------------------|-------------------|-------------------|
| 900                  | 10              | 0.2448511         | 3.9240637         | 0.1672753         | 2.1243228         |
| 100                  | 20              | 0.2175821         | 2.6531343         | 0.1353706         | 1.3611011         |
| 110                  | 30              | 0.1652331         | 1.6234701         | 0.1124619         | 0.9680506         |
| 120                  | 40              | 0.1340978         | 1.1560352         | 0.0696437         | 0.4958605         |
| 130                  | 50              | 0.1158443         | 0.7777786         | 0.0444991         | 0.2958338         |
| 140                  | 60              | 0.1013961         | 0.5876048         | 0.0244793         | 0.1574037         |
| 150                  | 70              | 0.0914650         | 0.4335489         | 0.0151749         | 0.0928402         |
| 160                  | 80              | 0.0822658         | 0.2933336         | 0.0097848         | 0.0434299         |
| 170                  | 90              | 0.0744696         | 0.1355488         | 0.0076431         | 0.0257060         |
| 180                  | 100             | 0.0660493         | 0.0714350         | 0.0037280         | 0.0194051         |
| 190                  | 110             | 0.0609040         | 0.0675720         | 0.0031577         | 0.0174815         |

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.

differences in convergence are smaller (light blue and purple lines in Figure 9), although not negligible, better for the more snapshots case and with faster and stabilized convergence.

4.3. 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.

After comparing the results in Figure 9 and Figure 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 (additional preliminary tests on a Poisson problem are
provided in A). 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 [45] 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, and the PhD candidate Nabil Atallah from Department of Civil and Environmental Engineering at Duke University for the help and the information provided in the SBM Poisson case. 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.

APPENDIX A. REDUCED ORDER RESULTS FOR THE POISSON PROBLEM AND SBM

In this appendix we report a preliminary test on a parametrized Poisson problem. This preliminary test has been useful before moving to the more complex Stokes problem. The SBM weak discrete formulation for a Poisson problem, with non-homogeneous Dirichlet boundary conditions, reads:

\[
\text{Find } u \in V_h^0 = \left\{ v \in C^0(\bar{\Omega}) : v|_K \in P^1(K), \forall K \in \bar{D}_T(\mu) \right\}, \text{ such that, } \forall w \in V_h^0, \quad a_0(u, w) = l_0(w),
\]

with

\[
a_0(u, w) = (\nabla w, \nabla u)_D - (w + \nabla w \cdot d, \nabla u \cdot \tilde{n})_{F_D} - (\nabla w \cdot \tilde{n}, u + \nabla u \cdot d)_{F_D} + (\alpha/h(w + \nabla w \cdot d), u + \nabla u \cdot d)_{F_D},
\]

\[
l_0(w) = (w, f)_D - (\nabla w \cdot \tilde{n}, \tilde{u}_D)_{F_D} - (\nabla w \cdot d, (\nabla \tilde{u}_D \cdot \tau_i \cdot \tilde{n})_{F_D} + (\alpha/h(w + \nabla w \cdot d), \tilde{u}_D)_{F_D},
\]

which actually is the FOM-SBM discretization.

The embedded domain consists in a circular cylinder and its position inside the domain is parametrized with a geometrical parameter \( \mu_1 \) that is describing the position of the cylinder center with respect to the y-center of the domain. The horizontal coordinate of the center of the cylinder is not parametrized and is located in the center of the domain. The ROM has been trained with 100 samples for \( \mu_1 \in [-0.5, 0.5] \) chosen randomly inside the parameter space. To test the accuracy of the ROM we compared its results on 10 additional samples that were not used to create the ROM and selected randomly within the same range.

In Figure 12, we plot the full-order solution, the reduced solution and the error for the geometrical parametrized Poisson problem: the full and the reduced solution are basically indistinguishable. To verify the behavior of the ROM and its sensitivity with respect to the number of modes, we compare in Figure 11 the average of the \( L^2 \)-norm relative error for the 10 different samples used to test the ROM with the \( L^2 \)-projection of the full-order results on the POD basis functions.
Figure 10. A zoom onto the cylinder of the Poisson numerical example in order to show the smoothing procedure employed by the SBM method inside the ghost area.

Figure 11. Poisson problem: The figure reports, on the left, (a) the mean relative error for the testing L2 projection of the full-order solution onto the POD basis functions, and (b) the mean relative error for the Galerkin projection ROM for various values of the number of modes. On the right, the eigenvalue decay of POD procedure.

Figure 12. Poisson problem: From left to right we report the full-order results, the reduced order results and the absolute value of the error, respectively. The results are for one random value of the parameter in the range of variation.

REFERENCES

[1] I. Akhtar, A. Nayfeh, and C. Ribbens, On the stability and extension of reduced-order Galerkin models in incompressible flow, Theoretical and Computational Fluid Dynamics 23 (2009), no. 3, 213–237.

[2] M. Balajewicz and C. Farhat, Reduction of nonlinear embedded boundary models for problems with evolving interfaces, Journal of Computational Physics 274 (2014), 489 – 504.

[3] F. Ballarin, A. D’Amario, S. Perotto, and G. Rozza, A POD-Selective Inverse Distance Weighting method for fast parametrized shape morphing, arXiv preprint arXiv:1710.09243 (2017).
[4] F. Ballarin, A. Manzoni, A. Quarteroni, and G. Rozza, \textit{Supremizer stabilization of POD-Galerkin approximation of parametrized steady incompressible Navier-Stokes equations}, International Journal for Numerical Methods in Engineering \textbf{102} (2014), no. 5, 1136–1161.

[5] M. Barrault, Y. Maday, N. Nguyen, and A. Patera, \textit{An \textquoteleft empirical interpolation\textquoteleft method: application to efficient reduced-basis discretization of partial differential equations}, Comptes Rendus Mathematique \textbf{339} (2004), no. 9, 667–672.

[6] Y. Bazilevs, K. Takizawa, and T. Tezduyar, \textit{Computational Fluid-Structure Interaction: Methods and Applications}, John Wiley \& Sons, (2013).

[7] P. Benner, M. Ohlberger, A. Patera, G. Rozza, and K. Urban, \textit{Model Reduction of Parametrized Systems}, MS&A series, no. Vol. 17, Springer, (2017).

[8] M. Bergmann, C.-H. Bruneau, and A. Iollo, \textit{Enablers for robust POD models}, Journal of Computational Physics \textbf{228} (2009), no. 2, 516–538.

[9] D. Boffi, F. Brezzi, and M. Fortin, \textit{Mixed Finite Element Methods and Applications}, 1 ed., Springer-Verlag Berlin Heidelberg, (2013).

[10] F. Brezzi and K. J. Bathe, \textit{A discourse on the stability conditions for mixed finite element formulations}, Computer Methods in Applied Mechanics and Engineering \textbf{82} (1990), no. 1-3, 27–57.

[11] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa, \textit{A numerical investigation of velocity-pressure reduced order models for incompressible flows}, Journal of Computational Physics \textbf{259} (2014), 598 – 616.

[12] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem, \textit{The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows}, Journal of Computational Physics \textbf{242} (2013), 623 – 647.

[13] F. Chinesta, A. Huerta, G. Rozza, and K. Willcox, \textit{Model Order Reduction}, Encyclopedia of Computational Mechanics Second Edition (2017), 1–36.

[14] F. Chinesta, P. Ladeveze, and E. Cueto, \textit{A Short Review on Model Order Reduction Based on Proper Generalized Decomposition}, Archives of Computational Methods in Engineering \textbf{18} (2011), no. 4, 395.

[15] A. Dumon, C. Allery, and A. Ammar, \textit{Proper general decomposition (PGD) for the resolution of Navier–Stokes equations}, Journal of Computational Physics \textbf{230} (2011), no. 4, 1387–1407.

[16] R. Everson and L. Sirovich, \textit{Karhunen-loève procedure for gappy data}, Journal of the Optical Society of America A \textbf{12} (1995), no. 8, 1657–1664.

[17] I. Kalashnikova and M. F. Barone, \textit{On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment}, International Journal for Numerical Methods in Engineering \textbf{83} (2010), no. 10, 1345–1375.

[18] A. Gerner and K. Veroy, \textit{Certified Reduced Basis Methods for Parametrized Saddle Point Problems}, SIAM Journal on Scientific Computing \textbf{34} (2012), no. 5, A2812–A2836.

[19] M. Grepl, Y. Maday, N. Nguyen, and A. Patera, \textit{Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations}, ESAIM: M2AN \textbf{41} (2007), no. 3, 575–605.

[20] M. Grepl and A. Patera, \textit{A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations}, ESAIM: M2AN \textbf{39} (2005), no. 1, 157–181.

[21] J. Hesthaven, G. Rozza, and B. Stamm, \textit{Certified Reduced Basis Methods for Parametrized Partial Differential Equations}, Springer Briefs in Mathematics, (2016).

[22] C. Hirt, A. Amsden, and J. Cook, \textit{An Arbitrary Lagrangian-Eulerian Computing Method for all Flow Speeds}, J. Comp. Phys. \textbf{14} (1974), no. 3, 227–253.

[23] T. J. Hughes and L. P. Franca, \textit{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}, Computer Methods in Applied Mechanics and Engineering \textbf{65} (1987), no. 1, 85–96.

[24] A. Iollo, S. Lanteri, and J.-A. Desideri, \textit{Stability Properties of POD-Galerkin Approximations for the Compressible Navier–Stokes Equations, Theoretical and Computational Fluid Dynamics} \textbf{13} (2000), no. 6, 377–396.

[25] I. Kalashnikova and M. F. Barone, \textit{On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment}, International Journal for Numerical Methods in Engineering \textbf{83} (2010), no. 10, 1345–1375.

[26] E. Karatzaas, F. Ballarin, and G. Rozza, \textit{Projection-based reduced order models for a cut finite element method in parametrized domains}, Submitted (2018).

[27] E. Karatzaas, G. Stabile, N. Atallah, G. Rozza, and G. Scovazzi, \textit{A Reduced Order Approach for the Embedded Shifted Boundary FEM and a Heat Exchange System on Parametrized Geometries}, Submitted (2018).

[28] K. Kunisch and S. Volkwein, \textit{Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics}, SIAM Journal on Numerical Analysis \textbf{40} (2002), no. 2, 492–515.

[29] A. Main and G. Scovazzi, \textit{The shifted boundary method for embedded domain computations. Part I: Poisson and Stokes problems}, Journal of Computational Physics, in press (2017), doi: 10.1016/j.jcp.2017.10.026.

[30] A. Main and G. Scovazzi, \textit{The shifted boundary method for embedded domain computations. Part II: Linear advection-diffusion and incompressible Navier-Stokes equations}, Journal of Computational Physics, in press (2018), doi: 10.1016/j.jcp.2018.01.023 (2018).

[31] R. Mittal and G. Iaccarino, \textit{Immersed boundary methods}, Annual Review of Fluid Mechanics \textbf{37} (2005), no. 1, 239–261.

[32] B. R. Noack, P. Papas, and P. A. Monkewitz, \textit{The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows}, Journal of Fluid Mechanics \textbf{523} (2005), 339–365.

[33] C. S. Peskin, \textit{Flow patterns around heart valves: A numerical method}, Journal of Computational Physics \textbf{10} (1972), no. 2, 262 – 271.

[34] A. Quarteroni, A. Manzoni, and F. Negri, \textit{Reduced Basis Methods for Partial Differential Equations}, Springer International Publishing, (2016).

[35] M. Barrault, Y. Maday, N. Nguyen, and A. Patera, \textit{An \textquoteleft empirical interpolation\textquoteleft method: application to efficient reduced-basis discretization of partial differential equations}, Comptes Rendus Mathematique \textbf{339} (2004), no. 9, 667–672.
[35] A. Quarteroni and G. Rozza, *Numerical solution of parametrized Navier–Stokes equations by reduced basis methods*, Numerical Methods for Partial Differential Equations 23 (2007), no. 4, 923–948.

[36] ———, *Numerical solution of parametrized Navier–Stokes equations by reduced basis methods*, Numerical Methods for Partial Differential Equations 23 (2007), no. 4, 923–948.

[37] G. Rozza, *Reduced Basis Methods for Elliptic Equations in subdomains with A-Posteriori Error Bounds and Adaptivity*, App. Num. Math. 55 (2005), no. 4, 403–424.

[38] G. Rozza, *Reduced basis methods for Stokes equations in domains with non-affine parameter dependence*, Computing and Visualization in Science 12 (2009), no. 1, 23–35.

[39] G. Rozza, D. Huynh, and A. Patera, *Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: Application to transport and continuum mechanics*, Archives of Computational Methods in Engineering 15 (2008), no. 3, 229–275.

[40] G. Rozza, D. B. P. Huynh, and A. Manzoni, *Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants*, Numerische Mathematik 125 (2013), no. 1, 115–152.

[41] G. Rozza and K. Veroy, *On the stability of the reduced basis method for Stokes equations in parametrized domains*, Computer Methods in Applied Mechanics and Engineering 196 (2007), no. 7, 1244–1260.

[42] S. Sirisup and G. Karniadakis, *Stability and accuracy of periodic flow solutions obtained by a POD-penalty method*, Physica D: Nonlinear Phenomena 202 (2005), no. 3-4, 218 – 237.

[43] T. Song, A. Main, G. Scovazzi, and M. Ricchiuto, *The shifted boundary method for hyperbolic systems: Embedded domain computations of linear waves and shallow water flows*, Journal of Computational Physics 369 (2018), 45–79.

[44] G. Stabile, S. Hijazi, A. Mola, S. Lorenzi, and G. Rozza, *POD-Galerkin reduced order methods for cfd using finite volume discretisation: vortex shedding around a circular cylinder*, Communications in Applied and Industrial Mathematics 8 (2017), no. 1, 210 – 236.

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

[46] G. Stabile and G. Rozza, *Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier-Stokes equations*, Computers & Fluids, in press (2018), doi:10.1016/j.compfluid.2018.01.035 (2018).

[47] ———, *Geometrical parametrization for finite-volume based reduced order methods*, Submitted (2018).

[48] K. Veroy, C. Prud’homme, and A. Patera, *Reduced-basis approximation of the viscous Burgers equation: rigorous a posteriori error bounds*, Comptes Rendus Mathematique 337 (2003), no. 9, 619–624.

[49] D. Xiao, F. F., A. Buchan, C. Pain, I. Navon, J. Du, and G. Hu, *Non linear model reduction for the Navier–Stokes equations using residual DEIM method*, Journal of Computational Physics 263 (2014), 1–18.