

Nonintrusive proper generalised decomposition for parametrised incompressible flow problems in OpenFOAM

V. Tsiolakis^{1,2,3}, M. Giacomini², R. Sevilla³, C. Othmer¹ and A. Huerta²

October 31, 2019

Abstract

The computational cost of parametric studies currently represents the major limitation to the application of simulation-based engineering techniques in a daily industrial environment. This work presents the first nonintrusive implementation of the proper generalised decomposition (PGD) in OpenFOAM, for the approximation of parametrised laminar incompressible Navier-Stokes equations. The key feature of this approach is the seamless integration of a reduced order model (ROM) in the framework of an industrially validated computational fluid dynamics software. This is of special importance in an industrial environment because in the online phase of the PGD ROM the description of the flow for a specific set of parameters is obtained simply via interpolation of the generalised solution, without the need of any extra solution step. On the one hand, the spatial problems arising from the PGD separation of the unknowns are treated using the classical solution strategies of OpenFOAM, namely the semi-implicit method for pressure linked equations (SIMPLE) algorithm. On the other hand, the parametric iteration is solved via a collocation approach. The resulting ROM is applied to several benchmark tests of laminar incompressible Navier-Stokes flows, in two and three dimensions, with different parameters affecting the flow features. Eventually, the capability of the proposed strategy to treat industrial problems is verified by applying the methodology to a parametrised flow control in a realistic geometry of interest for the automotive industry.

¹Volkswagen AG, Brieffach 011/1777, D-38436, Wolfsburg, Germany

²Laboratori de Càlcul Numèric (LaCàN), ETS de Ingenieros de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Barcelona, Spain

³Zienkiewicz Centre for Computational Engineering, College of Engineering, Swansea University, Wales, UK

Corresponding author: Matteo Giacomini. *E-mail*: matteo.giacomini@upc.edu

Keywords: Reduced order models, proper generalised decomposition, finite volume, incompressible laminar Navier-Stokes, pressure Poisson equation, parametrised flows, OpenFOAM, nonintrusiveness

1 Introduction and motivations

Computational fluid dynamics (CFD) is a key component in the current industrial design pipeline. Simulations of incompressible flows are performed on a daily basis to solve different problems both in automotive and aeronautical industries. Owing to its robustness, the most widely spread CFD methodology is the finite volume (FV) method [1–7]. Using this technique, numerically evaluated quantities of interest (e.g. drag and lift) have proved to match reasonably well experimental results in many realistic engineering problems.

Nonetheless, design and optimisation cycles in a production environment require multiple queries of the same problem with boundary conditions, physical properties of the fluid and geometry of the domain varying within a range of values of interest. In this context, parameters act as extra-coordinates of a high-dimensional partial differential equation (PDE). The computational cost of such parametric studies currently represents the major limitation to the application of simulation-based engineering techniques in a daily industrial environment. It is well-known that the computational complexity of approximating the PDEs describing the problems under analysis increases exponentially with the number of parameters considered. In recent years, reduced order models [8,9], including reduced basis (RB) [10–16], proper orthogonal decomposition (POD) [17–23] and hierarchical model reduction (HiMod) [24–27], have been proposed to reduce the computational burden of parametric analysis for several physical problems, including incompressible flows. The aforementioned techniques rely on an *a posteriori* reduction based on snapshots computed as solutions of the full-order model for different values of the parameters under analysis.

An alternative approach is represented by PGD [28–32]. This method features an *a priori* reduction [33–35], using a separable approximation of the solution, which depends explicitly on the parameters under analysis. In this context, during an offline phase, a reduced basis is constructed with no *a priori* knowledge of the solution, whereas efficient online evaluations of the generalised solution are performed by simple interpolation in the parametric space. The PGD framework has been first applied to incompressible Navier-Stokes equations in [36] to separate x and y directions in two-dimensional problems and in [37, 38] to separate space and time discretisations of unsteady flows. Moreover, special attention has been devoted to the study of convective phenomena and convection stabilisation using PGD-based separated representations [39, 40]. Applications of PGD to different physical phenomena including wave propagation, incompressible flows, nonlinear thermal problems and elastic metamaterials have been studied in [41–44]. Of course, parametric solutions involving the geometry of the domain, which are beyond the contributions of this paper, are of great interest in an industrial environment for the speed up of shape de-

sign and optimisation procedures. A separated solution of a PDE involving parametrised geometries was first proposed in [45] and further investigated in [42, 46, 47]. Such technique can be extended to the PGD algorithm for OpenFOAM proposed in this work via the definition of two nested computational meshes, one with the FV cells used for the flow computation and one featuring macrocells for the description of the geometry. In the context of geometrically parametrised problems, it is worth mentioning that a seamless integration of PGD and computer aided design (CAD) tools has been recently proposed in [48] to exploit the potential of parametric representation of curves and surfaces via the control points of nonuniform rational B-splines (NURBS).

In the context of flow problems, model reduction techniques based on Galerkin projection have been extensively studied in the literature [49–51]. In this framework, several strategies have been proposed to construct the trial basis, using POD [52], RB [53] or the empirical interpolation method [54]. Concerning incompressible Navier-Stokes equations, in [55, 56] supremiser stabilisations techniques have been investigated to couple the FV method with POD to solve parametrised turbulent flow problems. Alternative projection methods based on minimisation of the residual of the momentum equation only [57] and on a least-squares Petrov-Galerkin approach [58–60] have been proposed. More recently, special attention has also been devoted to FV-based structure-preserving ROMs for conservation laws [61].

Another key aspect for the application of simulation-based techniques to industrial problems is the capability of the proposed methods to provide verified and certified results. This problem has been classically treated by equipping numerical methods with reliable and fully-computable *a posteriori* error estimators using equilibrated fluxes [62–64] and flux-free approaches [65–67] to control the error of the solution as well as of quantities of interest [68–75]. Nonetheless, these approaches require intrusive modifications of existing computational libraries and may not be feasible in the context of commercial software. Hence, although the effort of the academic community in this direction, such solutions have not been successfully and widely integrated in codes utilised by the industry. More recently, to circumvent this issue, great effort has been devoted to nonintrusive implementations in which novel numerical methodologies are externally coupled to existing commercial and open-source software used in industry on a daily basis. Some contributions in this direction have been successfully proposed coupling PGD with Abaqus[®] for mechanical problems [76] and PGD with SAMTECH[®] for shape optimisation problems [77]. For flow problems, the coupling of POD and OpenFOAM has been discussed in [56, 78]. The present contribution is the first nonintrusive integration of the PGD framework in OpenFOAM for the solution of parametrised incompressible Navier-Stokes problems in the laminar regime. The resulting algorithm, henceforth referred to as `pgdFoam`, relies on internal functions and routines of OpenFOAM [79] and exploits the incompressible flow solver `simpleFoam` for the spatial iteration of the alternating direction scheme.

The rest of this paper is organised as follows. Section 2 recalls the incompressible Navier-Stokes equations and their FV approximation. The parametrised equations for

laminar Navier-Stokes flows are introduced in Section 3 as well as their PGD approximation and its nonintrusive implementation in OpenFOAM. Numerical simulations to validate the discussed reduced-order strategy are provided in Section 4, whereas its application to parametrised flow control problems in laminar regime is presented in Section 5. Section 6 summarises the results and two appendices complement the information with some technical details on the formulation and the OpenFOAM spatial solver utilised.

2 Finite volume approximation of laminar incompressible Navier-Stokes equations

In this section, the steady Navier-Stokes equations for the simulation of incompressible viscous laminar flows in d spatial dimensions are recalled. Let $\Omega \subset \mathbb{R}^d$, $\partial\Omega = \Gamma_D \cup \Gamma_N$ be an open bounded domain with disjoint Dirichlet, Γ_D , and Neumann, Γ_N , boundaries. The flow problem under analysis consists of computing the velocity field \mathbf{u} and the pressure p such that

$$\left\{ \begin{array}{ll} \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\nu \nabla \mathbf{u}) + \nabla p = \mathbf{s} & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{u}_D & \text{on } \Gamma_D, \\ \mathbf{n} \cdot (\nu \nabla \mathbf{u} - p \mathbf{I}_d) = \mathbf{t} & \text{on } \Gamma_N, \end{array} \right. \quad (1)$$

where the first equation describes the balance of momentum and the second one the conservation of mass. In Equation (1), \mathbf{s} represents a volumetric source term, $\nu > 0$ is the dynamic viscosity and \mathbf{I}_d is the $d \times d$ identity matrix. On the Dirichlet boundary Γ_D , the value \mathbf{u}_D of the velocity is imposed, whereas on Γ_N the pseudo-traction \mathbf{t} is applied. From the modelling point of view, inlet surfaces and physical walls are described as Dirichlet boundaries with an imposed entering velocity profile and a homogeneous datum, respectively, whereas outlet surfaces feature homogeneous Neumann boundary conditions. For the sake of simplicity and without loss of generality, Γ_N is henceforth assumed to be an outlet boundary, that is a null \mathbf{t} is considered.

2.1 A cell-centred finite volume approximation using OpenFOAM

In this section, the formulation of a FV scheme for the laminar incompressible Navier-Stokes equations is briefly recalled to introduce the notation needed for the high-dimensional parametrised problem of Section 3. The domain Ω is partitioned in N nonoverlapping cells V_i , $i=1, \dots, N$ such that $\Omega := \bigcup_{i=1}^N V_i$ and $V_i \cap V_j = \emptyset$, for $i \neq j$. The FV discretisation is constructed starting from the integral formulation of Equation (1), namely find (\mathbf{u}, p) ,

constant on each cell V_i , $i=1, \dots, N$, such that $\mathbf{u}=\mathbf{u}_D$ on Γ_D and it holds

$$\left\{ \begin{array}{l} \int_{V_i} \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) dV - \int_{V_i} \nabla \cdot (\nu \nabla \mathbf{u}) dV + \int_{V_i} \nabla p dV = \int_{V_i} \mathbf{s} dV, \\ \int_{V_i} \nabla \cdot \mathbf{u} dV = 0. \end{array} \right. \quad (2)$$

OpenFOAM implements a cell-centred FV rationale in which piecewise constant approximations are sought for velocity and pressure in each cell of the computational mesh and the degrees of freedom of the discretised problem are located at the centroid of each finite volume. Employing Gauss's theorem, the integrals in Equation (2) are rewritten in terms of fluxes over the boundaries of the cells and approximated using classical central differencing schemes [4, 5]. Moreover, to handle the nonlinearity in the convection term, OpenFOAM considers a relaxation approach introducing a fictitious time variable. The resulting solution strategy relies on the SIMPLE algorithm [80] which belongs to the family of fractional-step projection methods [81, Sect. 6.7]. A brief description of this method is provided in B.

3 Nonintrusive proper generalised decomposition for parametrised laminar flow problems

Consider now the case in which the user-prescribed data in Equation (1), i.e. the viscosity coefficient, the source term and the boundary conditions, depend on a set of parameters $\boldsymbol{\mu} \in \mathcal{I} \subset \mathbb{R}^M$, with M being the number of parameters. More precisely, the set \mathcal{I} describing the range of admissible parameters can be defined as the Cartesian product of the domains of the M parameters, namely, $\mathcal{I} := \mathcal{I}_1 \times \mathcal{I}_2 \times \dots \times \mathcal{I}_M$ with $\mu_i \in \mathcal{I}_i$ for $i=1, \dots, M$. Within this context, $\boldsymbol{\mu}$ is treated as a set of additional independent variables, or parametric coordinates, instead of problem parameters. For the purpose of discretisation, each interval \mathcal{I}_i is subdivided in N_μ subintervals. The unknown pair (\mathbf{u}, p) is thus sought in a high-dimensional space described by the independent variables $(\mathbf{x}, \boldsymbol{\mu}) \in \Omega \times \mathcal{I}$ and fulfils the following parametrised Navier-Stokes equations on each cell V_i , $i=1, \dots, N$

$$\left\{ \begin{array}{l} \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) dV d\mathcal{I} - \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\nu \nabla \mathbf{u}) dV d\mathcal{I} \\ \quad + \int_{\mathcal{I}} \int_{V_i} \nabla p dV d\mathcal{I} = \int_{\mathcal{I}} \int_{V_i} \mathbf{s} dV d\mathcal{I}, \\ \int_{\mathcal{I}} \int_{V_i} \nabla \cdot \mathbf{u} dV d\mathcal{I} = 0. \end{array} \right. \quad (3)$$

In the following sections, the rationale for the construction of a separated solution of the parametrised laminar Navier-Stokes equations is recalled and the proposed nonintrusive implementation of the alternating direction scheme in OpenFOAM is presented.

3.1 The proper generalised decomposition rationale

PGD constructs an approximation $(\mathbf{u}_{\text{PGD}}^n, p_{\text{PGD}}^n)$ of the solution (\mathbf{u}, p) of Equation (3) in terms of a sum of n separable functions, or modes. Each mode is the product of functions depending solely on one of the arguments $\mathbf{x}, \mu_1, \dots, \mu_M$. For the sake of readability and without loss of generality, only the spatial coordinates \mathbf{x} and the parametric ones $\boldsymbol{\mu}$ are henceforth separated.

Following [42], the so-called *single parameter approximation* is detailed. That is, for each mode, a unique scalar parametric function $\phi(\boldsymbol{\mu})$ is considered for all the variables and the resulting separated form of the unknowns is

$$\begin{cases} \mathbf{u}_{\text{PGD}}^n(\mathbf{x}, \boldsymbol{\mu}) = \mathbf{u}_{\text{PGD}}^{n-1}(\mathbf{x}, \boldsymbol{\mu}) + \sigma_u^n \mathbf{f}_u^n(\mathbf{x}) \phi^n(\boldsymbol{\mu}), \\ p_{\text{PGD}}^n(\mathbf{x}, \boldsymbol{\mu}) = p_{\text{PGD}}^{n-1}(\mathbf{x}, \boldsymbol{\mu}) + \sigma_p^n f_p^n(\mathbf{x}) \phi^n(\boldsymbol{\mu}), \end{cases} \quad (4)$$

where the superindex n denotes the, *a priori* unknown, number of terms in the PGD expansion and the positive scalar coefficients σ_u^n and σ_p^n represent the amplitude of the n -th mode for velocity and pressure, respectively. These coefficients are obtained normalising the modal functions, namely

$$\sigma_u^n := \|\mathbf{f}_u^n\| \quad \text{and} \quad \sigma_p^n := \|f_p^n\|,$$

with $\|\phi^n\| = 1$. Appropriate user-defined norms on the spatial and parametric domains need to be introduced for each function. For all the simulations in Section 4 and 5, the \mathcal{L}_2 norm has been considered for normalisation.

Remark 1. The normalisation coefficients play a critical role in checking the convergence of the PGD algorithm and may be used as quantitative stopping criterion in the PGD enrichment procedure described in Section 3.2.

For a discussion on alternative formulations of the separation in Equation (4), involving both scalar and vector-valued parametric functions, the interested reader is referred to [42]. Henceforth and except in case of ambiguity, the dependence of the modes on \mathbf{x} and $\boldsymbol{\mu}$ is omitted.

Considering a linearised approach to compute each new mode, Equation (4) can be rewritten as the following *predictor-corrector single parameter approximation*

$$\begin{cases} \mathbf{u}_{\text{PGD}}^n = \tilde{\mathbf{u}}_{\text{PGD}}^n + \sigma_u^n \delta \tilde{\mathbf{u}}_{\text{PGD}}^n = \mathbf{u}_{\text{PGD}}^{n-1} + \sigma_u^n \mathbf{f}_u^n \phi^n + \sigma_u^n \delta \tilde{\mathbf{u}}_{\text{PGD}}^n, \\ p_{\text{PGD}}^n = \tilde{p}_{\text{PGD}}^n + \sigma_p^n \delta \tilde{p}_{\text{PGD}}^n = p_{\text{PGD}}^{n-1} + \sigma_p^n f_p^n \phi^n + \sigma_p^n \delta \tilde{p}_{\text{PGD}}^n, \end{cases} \quad (5)$$

where $\tilde{\mathbf{u}}_{\text{PGD}}^n := \mathbf{u}_{\text{PGD}}^{n-1} + \sigma_u^n \mathbf{f}_u^n \phi^n$ and $\tilde{p}_{\text{PGD}}^n := p_{\text{PGD}}^{n-1} + \sigma_p^n f_p^n \phi^n$ account for the $n-1$ previously computed terms and a prediction of the current mode. More precisely, $(\sigma_u^n \mathbf{f}_u^n \phi^n, \sigma_p^n f_p^n \phi^n)$ play the role of predictors in the computation of the n -th mode, whereas $(\sigma_u^n \delta \tilde{\mathbf{u}}_{\text{PGD}}^n, \sigma_p^n \delta \tilde{p}_{\text{PGD}}^n)$ are the

corresponding correctors featuring the variations Δ in the spatial and parametric functions, namely

$$\begin{cases} \tilde{\delta \mathbf{u}}_{\text{PGD}}^n := \Delta \mathbf{f}_u \phi^n + \mathbf{f}_u^n \Delta \phi + \Delta \mathbf{f}_u \Delta \phi, \\ \tilde{\delta p}_{\text{PGD}}^n := \Delta f_p \phi^n + f_p^n \Delta \phi + \Delta f_p \Delta \phi. \end{cases} \quad (6)$$

Note that the last term in Equation (6) represents a high-order variation which is henceforth neglected. As for the classical *single parameter approximation*, σ_u^n and σ_p^n represent the amplitudes of the n -th velocity and pressure modes. That is, setting $\|\phi^n + \Delta \phi\| = 1$, they are defined as

$$\sigma_u^n := \|\mathbf{f}_u^n + \Delta \mathbf{f}_u\| \quad \text{and} \quad \sigma_p^n := \|f_p^n + \Delta f_p\|.$$

3.2 Predictor-corrector alternating direction scheme

In order to compute $(\mathbf{u}_{\text{PGD}}^n, p_{\text{PGD}}^n)$ in Equation (5), a greedy algorithm is implemented. The first PGD mode $(\mathbf{u}_{\text{PGD}}^0, p_{\text{PGD}}^0)$ is arbitrarily chosen to fulfil the Dirichlet boundary conditions of the problem and the n -th mode is successively computed assuming that term $n-1$ is available [31, 32]. Some variations of this strategy based on Arnoldi-type iterations have been investigated in [82, 83]. In this section, the alternating direction scheme used to compute the PGD modes is described. A key assumption for the application of this method is the separability of the data. For the sake of simplicity and without any loss of generality, the separated form of the viscosity coefficient, see e.g. [46], is reported

$$\nu(\mathbf{x}, \boldsymbol{\mu}) := \psi(\boldsymbol{\mu}) D(\mathbf{x}) = \sum_{i=1}^{n_\nu} \psi_{1,i}(\mu_1) \cdots \psi_{M,i}(\mu_M) D_i(\mathbf{x}), \quad (7)$$

and analogous separations are considered for all the parametric data in the problem under analysis.

By plugging (5) into (3) and gathering the unknown increments $(\sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n, \sigma_p^n \tilde{\delta p}_{\text{PGD}}^n)$ on the left-hand side while leaving on the right-hand side the residuals computed using the previous modes $(\mathbf{u}_{\text{PGD}}^{n-1}, p_{\text{PGD}}^{n-1})$ and the predictions $(\sigma_u^n \mathbf{f}_u^n \phi^n, \sigma_p^n f_p^n \phi^n)$ of the current one, the following equations are obtained

$$\left\{ \begin{array}{l} \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n \otimes \sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} \\ + \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n \otimes \tilde{\mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} + \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\tilde{\mathbf{u}}_{\text{PGD}}^n \otimes \sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} \\ - \int_{\mathcal{I}} \psi \int_{V_i} \nabla \cdot (D \nabla (\sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n)) dV d\mathcal{I} + \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\sigma_p^n \tilde{\delta p}_{\text{PGD}}^n) dV d\mathcal{I} = \mathcal{R}_u, \\ \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\sigma_u^n \tilde{\delta \mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} = \mathcal{R}_p, \end{array} \right. \quad (8)$$

where the residuals are defined as

$$\begin{aligned}
\mathcal{R}_u &:= \mathcal{R}_u(\mathbf{u}_{\text{PGD}}^{n-1}, p_{\text{PGD}}^{n-1}, \sigma_u^n \mathbf{f}_u^n, \sigma_p^n f_p^n, \phi^n) = \mathcal{R}_u(\tilde{\mathbf{u}}_{\text{PGD}}^n, \tilde{p}_{\text{PGD}}^n) \\
&= \int_{\mathcal{I}} \int_{V_i} \mathbf{s} dV d\mathcal{I} - \int_{\mathcal{I}} \int_{V_i} \nabla \cdot (\tilde{\mathbf{u}}_{\text{PGD}}^n \otimes \tilde{\mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} \\
&\quad + \int_{\mathcal{I}} \psi \int_{V_i} \nabla \cdot (D \nabla \tilde{\mathbf{u}}_{\text{PGD}}^n) dV d\mathcal{I} - \int_{\mathcal{I}} \int_{V_i} \nabla \tilde{p}_{\text{PGD}}^n dV d\mathcal{I},
\end{aligned} \tag{9}$$

$$\begin{aligned}
\mathcal{R}_p &:= \mathcal{R}_p(\mathbf{u}_{\text{PGD}}^{n-1}, \sigma_u^n \mathbf{f}_u^n, \phi^n) = \mathcal{R}_p(\tilde{\mathbf{u}}_{\text{PGD}}^n) \\
&= - \int_{\mathcal{I}} \int_{V_i} \nabla \cdot \tilde{\mathbf{u}}_{\text{PGD}}^n dV d\mathcal{I}.
\end{aligned} \tag{10}$$

As classical in ROMs [84, 85], an affine dependence of the forms in (8), (9) and (10) on the parameters is required to construct the PGD approximation. The spatial (respectively, parametric) component of each mode is thus computed by restricting Equation (8) to the tangent manifold associated with the spatial (respectively, parametric) coordinate. Following from Equation (6) and setting a fixed value for the parametric function ϕ^n , the pair $(\sigma_u^n \Delta \mathbf{f}_u, \sigma_p^n \Delta f_p)$ is determined by solving a purely spatial PDE. Recall that the PGD alternating direction scheme handles homogeneous Dirichlet boundary conditions at each iteration of the spatial solver [32], whereas inhomogeneous data are treated by the first arbitrary PGD mode $(\mathbf{u}_{\text{PGD}}^0, p_{\text{PGD}}^0)$ introduced above. In a similar fashion, the increment $\Delta \phi$ is computed as the solution of an algebraic system of equations in the parameter $\boldsymbol{\mu}$ while the spatial functions $(\sigma_u^n \mathbf{f}_u^n, \sigma_p^n f_p^n)$ are considered known.

Note that at each iteration of the alternating direction scheme, $\tilde{\mathbf{u}}_{\text{PGD}}^n$ is known and may be expressed in separated form as $\sum_{m=1}^n \sigma_u^m \mathbf{f}_u^m \phi^m$. Thus, exploiting the separated structure of the unknowns and the affine parametric decomposition of the involved integral forms, the numerical complexity of the high-dimensional PDE is reduced by alternatively solving for the spatial and the parametric unknowns, as detailed in the next subsections.

Remark 2. By restricting Equation (8) to the tangent manifold in the spatial (respectively, parametric) direction, the integral forms are multiplied by ϕ^n (respectively, $(\sigma_u^n \mathbf{f}_u^n, \sigma_p^n f_p^n)$). This is equivalent to the projection of the high-dimensional PDE to the tangent manifold discussed for PGD in the context of finite element approximations [42]. In the framework of FV discretisations, the finite element test functions are set equal to 1 to retrieve the classical integral form of the PDE under analysis. Hence, the restriction of the high-dimensional PDE to the tangent manifold introduces a factor ϕ^n multiplying the integral forms in Equations (8), (9) and (10) for the spatial iteration. Similarly, in the parametric iteration, a factor $\sigma_u^n \mathbf{f}_u^n$ appears in the integral form of the momentum equation and in the residual \mathcal{R}_u , whereas $\sigma_p^n f_p^n$ multiplies the integrand terms in the mass conservation equation and in the residual \mathcal{R}_p .

3.2.1 The spatial iteration

First, the parametric function ϕ^n is fixed and the increments $(\sigma_u^n \Delta \mathbf{f}_u, \sigma_p^n \Delta f_p)$ are determined by solving a spatial PDE. More precisely, restricting Equation (8) to the tangent manifold in the spatial direction, a pair $(\sigma_u^n \Delta \mathbf{f}_u, \sigma_p^n \Delta f_p)$, constant cell-by-cell, is sought such that in each cell V_i , $i=1, \dots, N$ it holds

$$\left\{ \begin{aligned} & \alpha_0 \int_{V_i} \nabla \cdot (\sigma_u^n \Delta \mathbf{f}_u \otimes \sigma_u^n \Delta \mathbf{f}_u) dV \\ & + \int_{V_i} \nabla \cdot \left(\sigma_u^n \Delta \mathbf{f}_u \otimes \sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \right) dV \\ & + \int_{V_i} \nabla \cdot \left(\sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^n \Delta \mathbf{f}_u \right) dV \\ & - \alpha_2 \int_{V_i} \nabla \cdot (D \nabla (\sigma_u^n \Delta \mathbf{f}_u)) dV + \alpha_3 \int_{V_i} \nabla (\sigma_p^n \Delta f_p) dV = R_u^n, \\ & \alpha_3 \int_{V_i} \nabla \cdot (\sigma_u^n \Delta \mathbf{f}_u) dV = R_p^n, \end{aligned} \right. \quad (11)$$

where R_u^n and R_p^n are the spatial residuals associated with the discretisation of the momentum and mass equations, respectively, and each coefficient α_i , $i=0, \dots, 3$ depends solely on the parametric function ϕ^n and on the data of the problem

$$\begin{aligned} \alpha_0 &:= \int_{\mathcal{I}} [\phi^n]^3 d\mathcal{I}, & \alpha_2 &:= \int_{\mathcal{I}} [\phi^n]^2 \psi d\mathcal{I}, \\ \alpha_1^m &:= \int_{\mathcal{I}} [\phi^n]^2 \phi^m d\mathcal{I}, & \alpha_3 &:= \int_{\mathcal{I}} [\phi^n]^2 d\mathcal{I}. \end{aligned} \quad (12)$$

Note that given the separable form of (9)-(10), an efficient implementation of the right-hand side of the spatial iteration may be devised and the corresponding FV discretisation is obtained. A detailed description of the residuals acting as linear functionals on the right-hand side of Equation (11) is provided in A.

The terms in Equation (11) feature a structure similar to the original incompressible Navier-Stokes problem in the spatial domain Ω , see Equation (2). The discretisation is thus performed using the cell-centred FV method implemented in OpenFOAM, see Section 2.1. The main difference is represented by the second and third integrals on the left-hand side of the momentum equation (11). In order to preserve the nonintrusiveness of the discussed PGD approach, Equation (11) is modified by introducing a relaxation in the SIMPLE iterations to treat these two integrals in an explicit way as part of the right-hand side of

the momentum equation, leading to

$$\left\{ \begin{array}{l} \alpha_0 \int_{V_i} \nabla \cdot (\sigma_u^n \Delta \mathbf{f}_u \otimes \sigma_u^n \Delta \mathbf{f}_u) dV \\ -\alpha_2 \int_{V_i} \nabla \cdot (D \nabla (\sigma_u^n \Delta \mathbf{f}_u)) dV + \alpha_3 \int_{V_i} \nabla (\sigma_p^n \Delta f_p) dV = R_u^n \\ -\int_{V_i} \nabla \cdot \left(\sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^{k-1} \Delta \mathbf{f}_u^{k-1} \right) dV \\ -\int_{V_i} \nabla \cdot \left(\sigma_u^{k-1} \Delta \mathbf{f}_u^{k-1} \otimes \sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \right) dV, \\ \alpha_3 \int_{V_i} \nabla \cdot (\sigma_u^n \Delta \mathbf{f}_u) dV = R_p^n, \end{array} \right. \quad (13)$$

where the index $k-1$ is associated with the last computed increment $\sigma_u^{k-1} \Delta \mathbf{f}_u^{k-1}$ in the SIMPLE algorithm, see B. It is straightforward to observe that Equation (13) now features the same structure as the original Navier-Stokes problem (2) for which the SIMPLE algorithm is designed. Hence, the resulting solver for the spatial iteration does not require any modification of the existing OpenFOAM routines and is nonintrusive with respect to `simpleFoam`.

Remark 3. The first integral in Equation (13) solely introduces a higher-order perturbation which, upon convergence of the SIMPLE iterations, is negligible. Hence, an alternative formulation of the momentum equation within the proposed PGD algorithm relies on neglecting such higher-order contribution. The convection term is thus linearised using the last computed approximation $\sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m$ of the unknown velocity field, namely

$$\left\{ \begin{array}{l} \int_{V_i} \nabla \cdot \left(\sigma_u^n \Delta \mathbf{f}_u \otimes \sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \right) dV \\ -\alpha_2 \int_{V_i} \nabla \cdot (D \nabla (\sigma_u^n \Delta \mathbf{f}_u)) dV + \alpha_3 \int_{V_i} \nabla (\sigma_p^n \Delta f_p) dV = R_u^n \\ -\int_{V_i} \nabla \cdot \left(\sum_{m=1}^n \alpha_1^m \sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^{k-1} \Delta \mathbf{f}_u^{k-1} \right) dV, \\ \alpha_3 \int_{V_i} \nabla \cdot (\sigma_u^n \Delta \mathbf{f}_u) dV = R_p^n. \end{array} \right. \quad (14)$$

The left-hand side of Equation (14) mimics the SIMPLE strategy to solve the linearized version of the Navier-Stokes equations because it substitutes the unknown convection field with a relaxation. This approach converges to the same solution of (14) (and this has been verified numerically). Nevertheless it is slightly intrusive in the context of `simpleFoam` as it requires the implementation of the linear convection term. Hence, Equation (13) is preferred for the PGD spatial iteration.

3.2.2 The parametric iteration

In the parametric step, the value of the previously computed spatial functions is fixed $(\mathbf{f}_u^n, f_p^n) \leftarrow (\sigma_u^n \mathbf{f}_u^n + \Delta \mathbf{f}_u, \sigma_p^n f_p^n + \Delta f_p)$ and the parametric increment $\Delta\phi$ acts as unknown. Within the *single parameter approximation* rationale, a unique scalar function depending on $\boldsymbol{\mu}$ is sought.

As noted in the previous Section, the correction introduced by the high-order term in the momentum equation is negligible upon convergence. Following the strategy described for the spatial iteration and neglecting this term in the restriction of Equation (8) to the parametric direction of the tangent manifold, $\Delta\phi$ is computed by solving the following algebraic equation

$$\left(\sum_{m=1}^n a_1^m \phi^m - a_2 \psi + a_3 \right) \Delta\phi = r_u^n + r_p^n, \quad (15)$$

where r_u^n and r_p^n are the parametric residuals associated with the discretisation of the momentum and mass equations, respectively, and each coefficient a_i , $i=1, \dots, 3$ depends solely on the spatial functions $(\sigma_u^n \mathbf{f}_u^n, \sigma_p^n f_p^n)$ and on the data of the problem, namely

$$\left\{ \begin{array}{l} a_1^m := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot [\nabla \cdot (\sigma_u^n \mathbf{f}_u^n \otimes \sigma_u^m \mathbf{f}_u^m)] dV \\ \quad + \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot [\nabla \cdot (\sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^n \mathbf{f}_u^n)] dV, \\ a_2 := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot [\nabla \cdot (D \nabla (\sigma_u^n \mathbf{f}_u^n))] dV, \\ a_3 := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot \nabla (\sigma_p^n f_p^n) dV + \int_{V_i} \sigma_p^n f_p^n \nabla \cdot (\sigma_u^n \mathbf{f}_u^n) dV. \end{array} \right. \quad (16)$$

The unknown $\Delta\phi$ is discretised at the nodes of the parametric domain \mathcal{I} and the resulting algebraic equation is solved via a collocation method. Similarly to the spatial iteration, the separable form of (9)-(10) is exploited to perform computationally efficient pointwise evaluations of the residuals at the nodes of \mathcal{I} . The complete derivation of the separated form of the right-hand side is detailed in A.

3.3 A nonintrusive implementation of the proper generalised decomposition in OpenFOAM

A critical aspect to make ROM strategies suitable for application on a daily basis in an industrial environment is their nonintrusiveness with respect to existing solution methodologies validated by companies. Thus, in order to solve parametrised flow problems using OpenFOAM, the proposed PGD algorithm is designed to be nonintrusive with respect to the `simpleFoam` solver as described above. As discussed in Section 3.2, inhomogeneous Dirichlet boundary conditions are treated by means of a spatial mode computed using the

full-order solver, whereas the corresponding parametric mode is set equal to 1 (Algorithm 1 - Step 1). Then, the enrichment process is started and at each iteration of the alternating direction scheme a spatial mode is computed using `simpleFoam` (Algorithm 1 - Steps 7 to 10) and a linear system is solved to determine the corresponding parametric term (Algorithm 1 - Steps 11 to 14). The alternating direction iterations stop when the computed corrections $\Delta \mathbf{f}_\diamond$, $\Delta \phi$ are negligible with respect to the amplitudes σ_\diamond^n , σ_ϕ of the current mode for $\diamond = u, p$ and the residuals ε_\diamond^r are sufficiently small for $\circ = u, p, \phi$ (Algorithm 1 - Steps 6 and 15). The global enrichment strategy ends when the amplitude of the current mode σ_\diamond^n is negligible with respect to the first one σ_\diamond^1 for $\diamond = u, p$ (Algorithm 1 - Step 3).

Remark 4. Alternative criteria may be considered to stop the greedy algorithm, e.g. when the magnitude of the last mode normalised with respect to the sum of the amplitudes of all the computed terms is lower than a user-defined tolerance η_\diamond^* , namely

$$\sigma_\diamond^n < \eta_\diamond^* \sum_{m=1}^n \sigma_\diamond^m, \text{ for } \diamond = u, p.$$

4 Numerical validation

In this section, numerical examples are presented to validate the proposed methodology. First, a test case with known analytical solution is considered to verify the optimal convergence rate of the high-dimensional FV approximation of the velocity and pressure fields, measured in the $\mathcal{L}_2(\Omega \times \mathcal{I})$ norm, for a parametrised viscosity coefficient. In this context, special emphasis is given to the additional error introduced by PGD, highlighting the range of applicability of the discussed reduced-order strategy in terms of expected accuracy of the parametric solution. Moreover, a classical benchmark test for incompressible flow solvers, namely the nonleaky lid-driven cavity, is studied parametrising the imposed velocity of the lid in a range of values of the Reynolds number spanning from 1,000 to 4,000.

4.1 Kovasznay flow with parametrised viscosity

Consider the Kovasznay flow [86] for a parametrised viscosity $\nu(\mu)=\mu$. The analytical solution is

$$\begin{aligned} \mathbf{u}(x, y, \mu) &= \left(1 - e^{\lambda(\mu)x} \cos(2\pi y), \frac{\lambda(\mu)}{2\pi} e^{\lambda(\mu)x} \sin(2\pi y) \right) \\ p(x, y, \mu) &= \frac{1}{2} (1 - e^{2\lambda(\mu)x}) + C \end{aligned} \quad (17)$$

where the constant C is determined by fixing a reference value for the pressure field in one point of the domain, whereas λ is a function of the parametrised viscosity and changes when the Reynolds number is modified, namely,

$$\lambda(\mu) = \frac{1}{2\nu(\mu)} - \sqrt{\frac{1}{(2\nu(\mu))^2} + 4\pi^2}.$$

Algorithm 1 pgdFoam: a nonintrusive PGD implementation in OpenFOAM

Require: Tolerances η_\diamond^* for the greedy algorithm. Tolerances η_\circ for the amplitudes and η_\circ^r for the residuals in the alternating direction iteration. Typical values typ_\circ for the residuals of the spatial and parametric problems. $\diamond = u, p$ and $\circ = u, p, \phi$.

- 1: Compute boundary condition modes: the spatial mode is solution of (2) using `simpleFoam` and the parametric mode is equal to 1.
 - 2: Set $n \leftarrow 1$ and initialise the amplitudes of the spatial modes $\sigma_\diamond^1 \leftarrow 1$.
 - 3: **while** $\sigma_\diamond^n > \eta_\diamond^* \sigma_\diamond^1$ **do**
 - 4: Set $k \leftarrow 0$, the parametric predictor $\phi^n \leftarrow 1$ and the spatial predictors $(\mathbf{f}_u^n, \mathbf{f}_p^n)$ using the last computed modes.
 - 5: Initialise $\varepsilon_\circ \leftarrow 1$, $\varepsilon_\circ^r \leftarrow \text{typ}_\circ$.
 - 6: **while** $\varepsilon_\circ > \eta_\circ$ or $\varepsilon_\circ^r > \eta_\circ^r$ **do**
 - 7: Compute the spatial residuals (22) and coefficients (12).
 - 8: Solve the spatial Navier-Stokes problem (13) using `simpleFoam`.
 - 9: Normalise the spatial predictors: $\sigma_\diamond^n \leftarrow \|\sigma_\diamond^n \mathbf{f}_\diamond^n + \Delta \mathbf{f}_\diamond\|$.
 - 10: Update the spatial predictors: $\mathbf{f}_\diamond^n \leftarrow (\sigma_\diamond^n \mathbf{f}_\diamond^n + \Delta \mathbf{f}_\diamond) / \sigma_\diamond^n$.
 - 11: Compute the parametric residual (24) and coefficients (16).
 - 12: Solve the parametric linear system (15).
 - 13: Normalise the parametric predictor: $\sigma_\phi \leftarrow \|\phi^n + \Delta \phi\|$.
 - 14: Update the parametric predictor: $\phi^n \leftarrow (\phi^n + \Delta \phi) / \sigma_\phi$.
 - 15: Update stopping criterions: $\varepsilon_\circ \leftarrow \|\Delta \mathbf{f}_\diamond\| / \sigma_\diamond^n$, $\varepsilon_\phi \leftarrow \|\Delta \phi\| / \sigma_\phi$, $\varepsilon_\circ^r \leftarrow \|r_\circ\|$.
 - 16: Update the alternating direction iteration counter: $k \leftarrow k + 1$.
 - 17: **end while**
 - 18: Update the mode counter: $n \leftarrow n + 1$.
 - 19: **end while**
-

The parameter μ is sought in the space $\mathcal{I}=[5\times 10^{-3}, 10^{-2}]$, which is discretised with uniform intervals. The corresponding values of the Reynolds number span from 100 to 200. The spatial domain $\Omega=[-1, 1]^2$ is discretised with a family of Cartesian meshes of quadrilateral cells. The characteristic lengths h_x and h_μ of the spatial and parametric discretisations, respectively, are provided in Table 1.

h_x	8.3×10^{-2}	4×10^{-2}	2×10^{-2}	1×10^{-2}	5×10^{-3}	2.5×10^{-3}
h_μ	2×10^{-2}	1×10^{-2}	5×10^{-3}	2.5×10^{-3}	1.25×10^{-3}	6.25×10^{-4}

Table 1: Normalised characteristic lengths of the spatial and parametric discretisations.

A convergence study under uniform mesh refinement is performed for the linearised Navier-Stokes equations using the meshes described in Table 1. In this context, a convective field \mathbf{a} given by the analytical expression of the Kovaszny velocity is introduced in Equation (1) and the convective term $\nabla \cdot (\mathbf{u} \otimes \mathbf{u})$ is replaced by $\nabla \cdot (\mathbf{u} \otimes \mathbf{a})$. As detailed in Section 3.2, an affine separation of the data is required to run PGD. Thus, the convective field \mathbf{a} is separated *a priori* considering the first four terms of the Taylor expansion of $e^{\lambda x}$ in the analytical form of the velocity, see Equation (17). For $\mu=10^{-2}$, the relative $\mathcal{L}_2(\Omega)$ error of the resulting separated velocity field with respect to the exact one is 4.3×10^{-3} and, consequently, a target error of 10^{-2} in the spatial discretisation is considered for the following convergence study. Moreover, the Dirichlet boundary datum \mathbf{u}_D requires five modes to be described in a separated form.

The $\mathcal{L}_2(\Omega \times \mathcal{I})$ error between the PGD approximation $(\mathbf{u}_{\text{PGD}}^n, p_{\text{PGD}}^n)$ computed using fifteen modes and the high-dimensional analytical solution (\mathbf{u}, p) as a function of the characteristic mesh size h_x is displayed in Figure 1. The optimal first-order convergence rate for pressure and second-order one for velocity are obtained.

To run the iterative procedure `pgdFoam` described in Algorithm 1, a stopping criterion $\eta_{(u,p)} \leq 10^{-5}$ is considered, where $\eta_{(u,p)}$ accounts for the relative amplitude of both the velocity and pressure modes, namely

$$\eta_{(u,p)} := \sqrt{\left(\frac{\sigma_u^n}{\sum_{m=1}^n \sigma_u^m}\right)^2 + \left(\frac{\sigma_p^n}{\sum_{m=1}^n \sigma_p^m}\right)^2}. \quad (18)$$

In Figure 2(a), the evolution of the amplitude $\eta_{(u,p)}$, η_u and η_p is displayed for the finest mesh described in Table 1. After ten computed modes, the stopping criterion is fulfilled and the PGD enrichment stops.

As previously mentioned, five terms are required to describe the Dirichlet boundary conditions in a separated form. Henceforth, only the computed modes, starting from the sixth term of the PGD approximation are presented. In Figure 2(b), the first six normalised computed parametric modes are displayed. The corresponding computed spatial modes for pressure and velocity are presented in Figure 3 and 4, respectively.

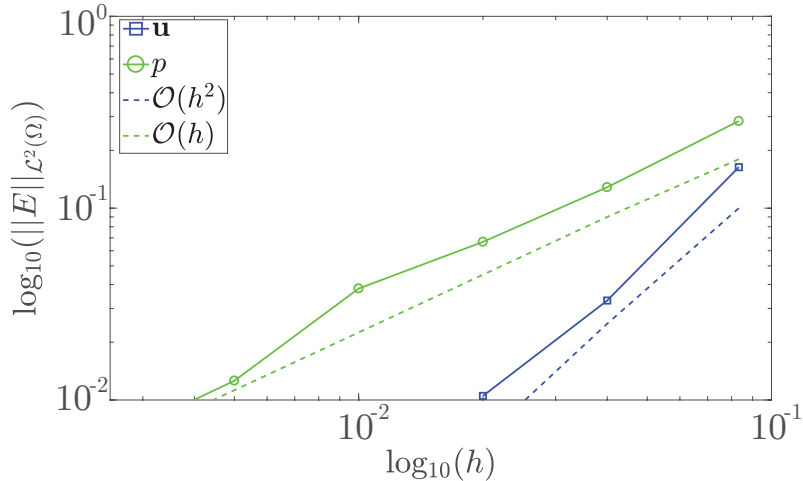
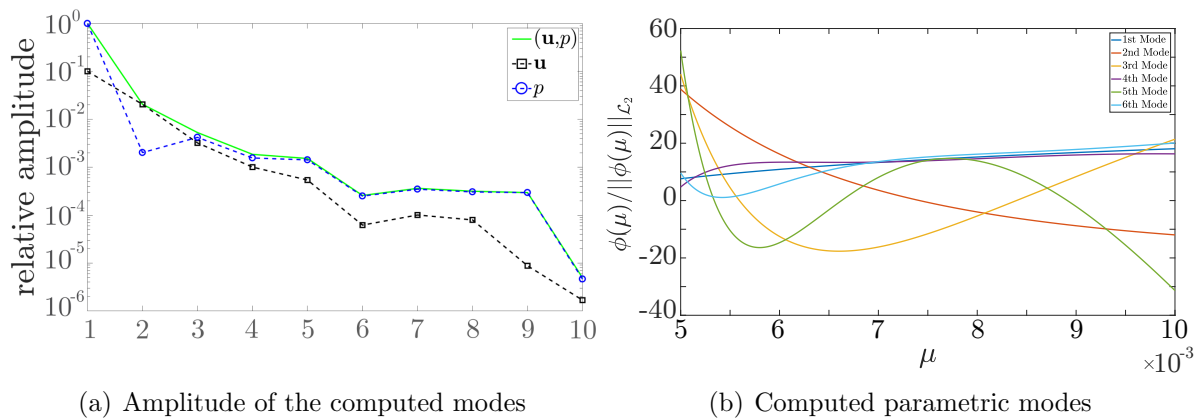


Figure 1: Optimal convergence of the $\mathcal{L}_2(\Omega \times \mathcal{I})$ error of the PGD approximation of the Kovasznay flow with parametrised viscosity with respect to the exact solution as a function of the characteristic mesh size h_x .



(a) Amplitude of the computed modes

(b) Computed parametric modes

Figure 2: PGD approximation of the Kovasznay flow with parametrised viscosity. (a) Relative amplitude of the computed modes \mathbf{f}_u^m (black), f_p^m (blue) and the combined amplitude of (\mathbf{f}_u^m, f_p^m) according to Equation (18). (b) First six normalised computed parametric modes.

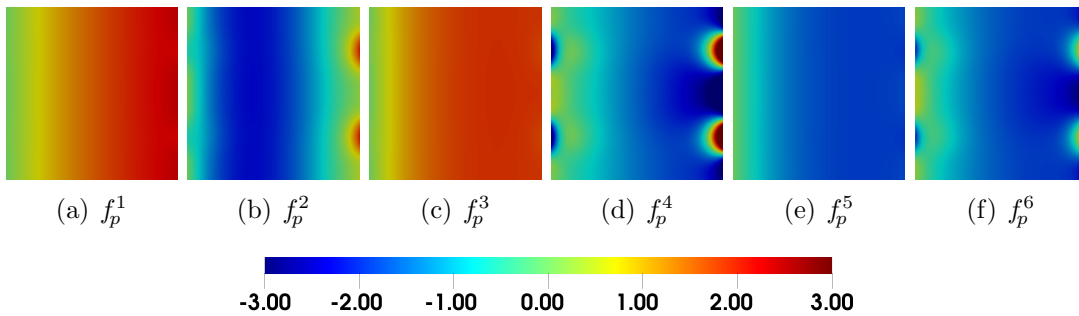


Figure 3: PGD approximation of the Kovaszny flow with parametrised viscosity. First six computed spatial modes f_p^m , $m = 1, \dots, 6$ for pressure.

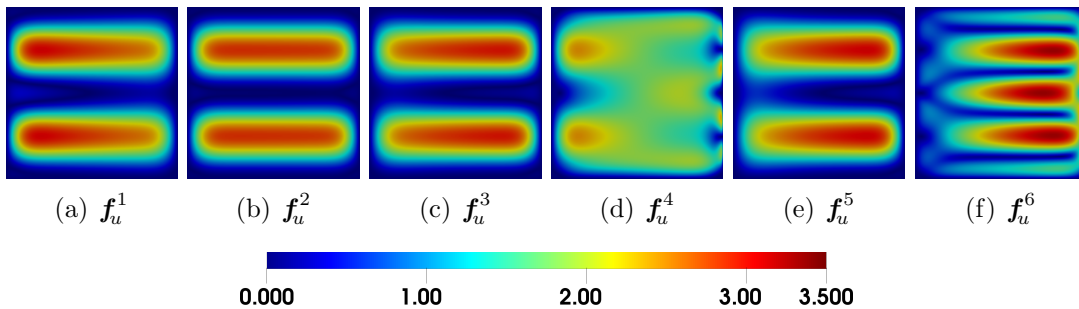


Figure 4: PGD approximation of the Kovaszny flow with parametrised viscosity. First six computed spatial modes f_u^m , $m = 1, \dots, 6$ for velocity.

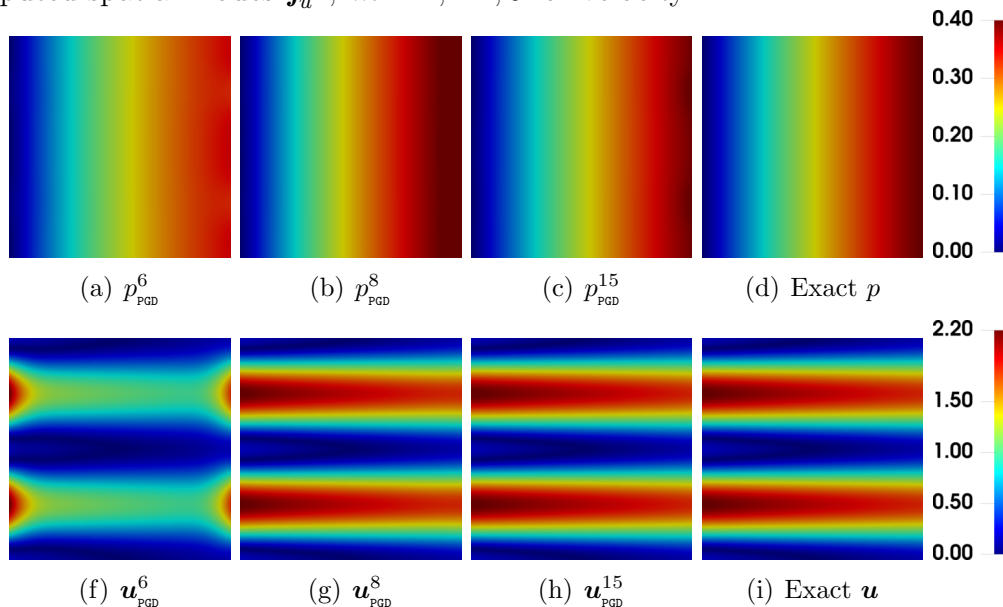


Figure 5: Comparison of the PGD approximation to the analytical solution of the Kovaszny flow for $\text{Re} = 200$, that is $\mu = 10^{-2}$. Pressure (top) and velocity (bottom) using $n=6$, 8 and 15 terms.

The PGD approximation $(\mathbf{u}_{\text{PGD}}^n, p_{\text{PGD}}^n)$ using $n=6, 8, 15$, that is with one, three and ten computed modes, respectively, is compared to the analytical solution for the case of $\text{Re}=200$, in Figure 5.

4.2 Two-dimensional cavity with parametrised lid velocity

In this section, the classical benchmark problem of the nonleaky lid-driven cavity is studied [87]. The unitary square $\Omega=[0, 1]^2$ is considered as spatial domain and homogeneous Dirichlet boundary conditions are imposed on the lateral and bottom walls. On the top wall, a velocity $\mathbf{u}_{\text{lid}}(\mathbf{x}, \mu)=400\mu\mathbf{u}_{\text{lid}}(\mathbf{x})$ is enforced, where the parameter $\mu \in [0.25, 1]$ acts as a scaling factor of the maximum velocity of the lid, whereas $\mathbf{u}_{\text{lid}}(\mathbf{x})$ is a velocity profile featuring two ramps on the top-left and top-right corners of the domain to account for the change between null and maximum velocity. As classical in the literature treating the lid-driven cavity example, for $x \in [0, 0.06]$ and $x \in [0.94, 1]$, the horizontal component of the lid velocity changes linearly from 0 to 400μ and vice versa. The dynamic viscosity is set to $\nu=0.1\text{ m}^2/\text{s}$ and the values considered for the Reynolds number span from 1,000 to 4,000.

The nonlinear term of the Navier-Stokes equations is now treated as described in Section 3.2. The mode handling the boundary conditions is obtained as a full-order solution of the Navier-Stokes equations using the `simpleFoam` algorithm for a lid velocity computed using $\mu=1$, that is for a maximum horizontal velocity of 400 m/s. The corresponding parametric boundary condition mode is set to be linearly evolving from $\mu=0.25$ to $\mu=1$, that is $\phi(\mu)=\mu$.

Following the rationale described in the previous section, two different stopping criterions are considered for the PGD enrichment strategy, namely $\eta_{(u,p)} \leq 10^{-3}$ and $\eta_{(u,p)} \leq 10^{-4}$. Figure 6(a) displays the relative amplitude of the computed modes. Note that the first stopping point is achieved after seven computed modes, whereas fourteen terms are required to fulfil the lower tolerance. The corresponding computed parametric modes are presented on Figure 6(b). It is worth noting that all the computed parametric modes are close or equal to 0 for $\mu=1$. This is due to the fact that the boundary conditions of the problem are imposed by means of a full-order solution computed for the maximum value of μ in the parametric space. Hence, the case of $\mu=1$ is accurately described by the PGD approximation using solely the mode obtained via `simpleFoam`.

Now, the online evaluations of the PGD approximation of the velocity and pressure fields for different values of the parameter μ are compared to the full-order solutions computed using `simpleFoam`. The corresponding relative $\mathcal{L}_2(\Omega)$ errors are presented in Figure 7 as a function of the number of modes utilised in the PGD approximation. The boundary condition mode and the first seven computed modes, for which $\eta_{(u,p)} \leq 10^{-3}$, provide a good approximation of both velocity and pressure and limited corrections are introduced by the following modes until the stopping criterion of 10^{-4} is fulfilled. For the case $\mu=1$, a small error of the order of 10^{-4} appears starting from the fifth computed mode, i.e.

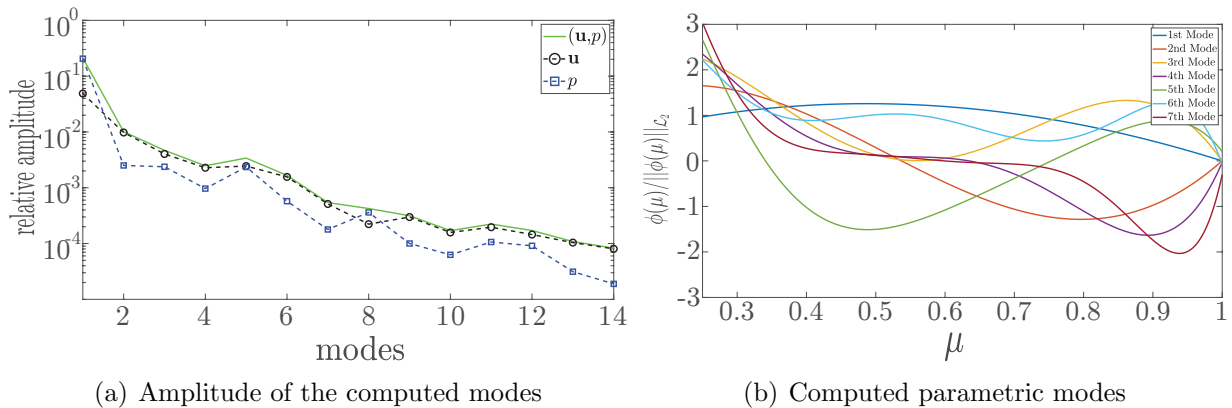


Figure 6: PGD of the cavity flow with parametrised lid velocity. (a) Relative amplitude of the computed modes \mathbf{f}_u^m (black), f_p^m (blue) and the combined amplitude of (\mathbf{f}_u^m, f_p^m) according to Equation (18). (b) First seven normalised computed parametric modes.

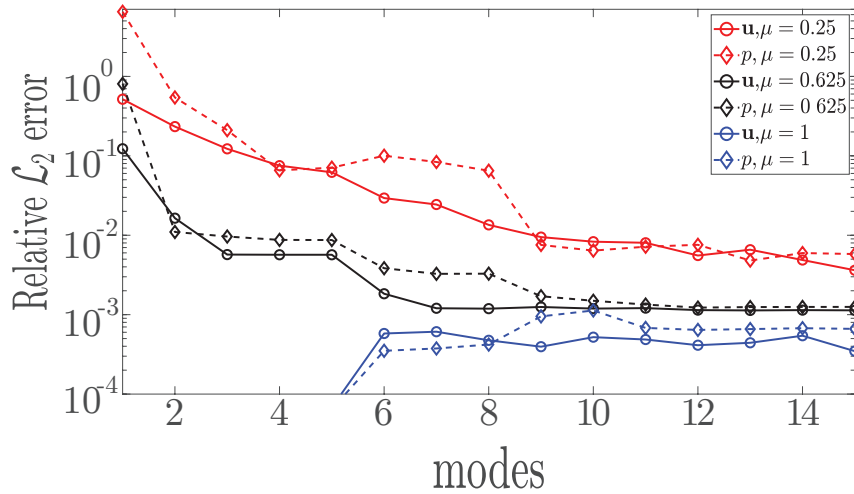


Figure 7: Relative $\mathcal{L}_2(\Omega)$ errors of the PGD approximation of the cavity flow with parametrised lid velocity with respect to the full-order solution as a function of the global number of modes (i.e. boundary conditions and computed) utilised in the PGD expansion.

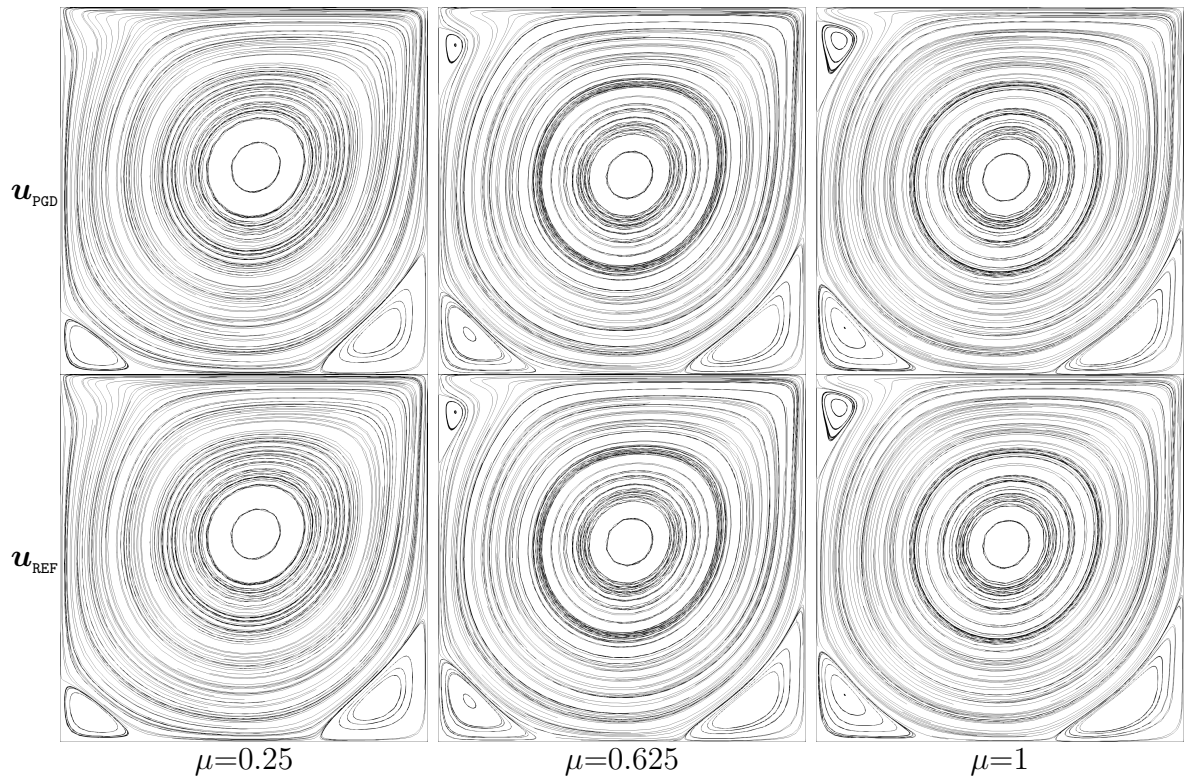


Figure 8: Comparison of the PGD approximation (top) and the full-order solution (bottom) of the parametrised lid-driven cavity problem for $\mu=0.25$, $\mu=0.625$ and $\mu=1$, corresponding to a maximum velocity of the lid of 100, 250 and 400 m/s, respectively.

$n=6$. This is due to the fact that the boundary condition mode already captures all the features of the flow, being a full-order solution of the Navier-Stokes equations as previously mentioned.

A qualitative comparison of the reduced-order and full-order solutions of the parametrised lid-driven cavity problem is displayed in Figure 8. The PGD approximations for $\mu=0.25$, $\mu=0.625$ and $\mu=1$ are presented as long as their corresponding simulations obtained using `simpleFoam`. The cases under analysis are associated with a maximum horizontal velocity of the lid of 100, 250 and 400 m/s, respectively. It is worth noting that `pgdFoam` is able to capture the topological changes of the flow with great accuracy, managing to identify location and size of the vortices, as well as their appearance and disappearance according to the values of the Reynolds number considered in the analysis.

5 Application to parametrised laminar flow control problems

Dynamically controlling the features of a flow is a challenging problem with several high-impact applications including, e.g., drag minimisation, stall control and aerodynamic noise reduction [88–90]. A major bottleneck to the design of flow control devices is represented by the large number of simulations involved in the tuning of the control loop. In this section, the potential of the described nonintrusive PGD implementation in OpenFOAM is demonstrated for parametrised flow control problems. Two- and three-dimensional internal flows with jets are studied. Specifically, a parametric study involving the peak velocity of the jets as extra-coordinate of the problem is considered to test the proposed PGD methodology.

5.1 Lid-driven cavity with parametrised jet velocity

Consider the nonleaky lid-driven cavity problem introduced in Section 4.2. The lid velocity is defined with two linear ramps, increasing from 0 to 10 m/s on the top-left corner and decreasing correspondingly on the top-right one. Three jets of size 0.12 m are introduced on the vertical walls, two on the right wall and one on the left, respectively. The parametrised velocity of the jets is $\mathbf{u}_{\text{jet}}(\mathbf{x}, \mu) = \mu \mathbf{u}_{\text{jet}}(\mathbf{x})$, where the maximum velocity is controlled by the parameter $\mu \in [0, 1]$ and the profile $\mathbf{u}_{\text{jet}}(\mathbf{x})$ is defined as

$$\mathbf{u}_{\text{jet}}(x, y) = \begin{cases} \left(-1 + \cos(2\pi y/0.12), 0 \right) & \text{for } x=1, y \in [0, 0.12], \\ \left(1 - \cos(2\pi(y-0.88)/0.12), 0 \right) & \text{for } x=1, y \in [0.88, 1], \\ \left(1 - \cos(2\pi(y-0.88)/0.12), 0 \right) & \text{for } x=0, y \in [0.88, 1]. \end{cases} \quad (19)$$

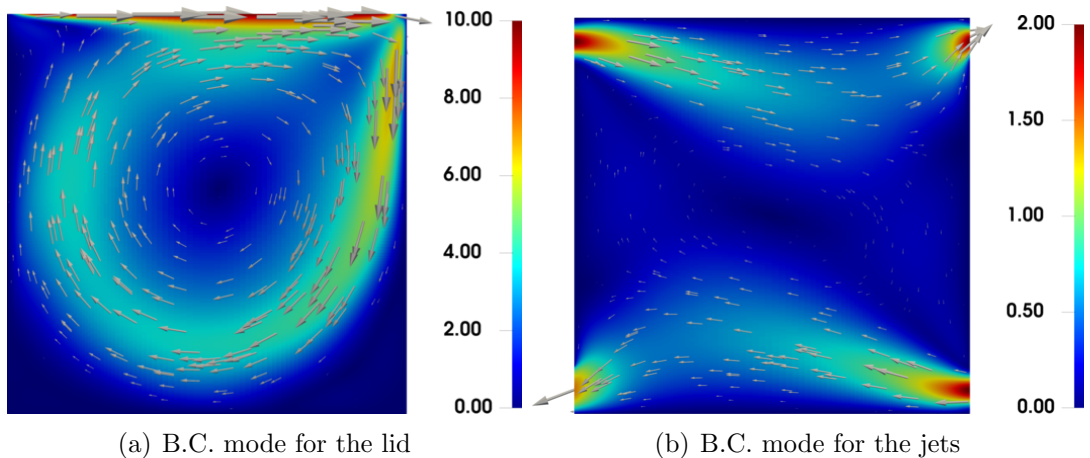


Figure 9: Cavity flow with parametrised jet velocity. Spatial boundary condition modes for velocity.

An outlet boundary is added on the left vertical wall for $y \in [0, 0.12]$ and a free-traction condition is enforced. The dynamic viscosity is set to $\nu=0.01 \text{ m}^2/\text{s}$, therefore the corresponding Reynolds number is $\text{Re}=1,000$.

The boundary conditions of the problem are enforced through two modes computed as full-order solutions via `simpleFoam` as shown in Figure 9: the first one, for $\mu=0$, corresponds to lid velocity of 10 m/s and inactive jets; the second one, for $\mu=1$, is associated with the maximum velocity of the jets and a zero velocity of the lid. The corresponding parametric modes for the boundary conditions are set to $\phi(\mu)=1$ and $\phi(\mu)=\mu$, respectively.

Following the rationale previously discussed, the PGD enrichment process is stopped when $\eta_{(u,p)} \leq 10^{-4}$. In Figure 10, the generalised solution computed by the PGD is interpolated in different points of the parametric space under analysis and compared with the corresponding full-order solutions provided by `simpleFoam`. The flows for $\mu=0.1$, $\mu=0.3$, $\mu=0.5$, $\mu=0.7$, $\mu=0.8$ and $\mu=1$ are displayed, covering a wide range of flow regimes in the cavity. It is worth noting that the discussed reduced-order strategy is able to capture the topological changes in the flow features and accurately reproduce the appearance and disappearance of vortices localised in different regions of the domain.

The accuracy of the PGD approximation with respect to the full-order solution is also verified by computing the relative $\mathcal{L}_2(\Omega)$ error of the spatial discretisation while enriching the modal description of the solution. More precisely, Figure 11 shows that using two boundary condition modes and six computed modes all approximations achieve a plateau with relative errors of 10^{-2} or lower.

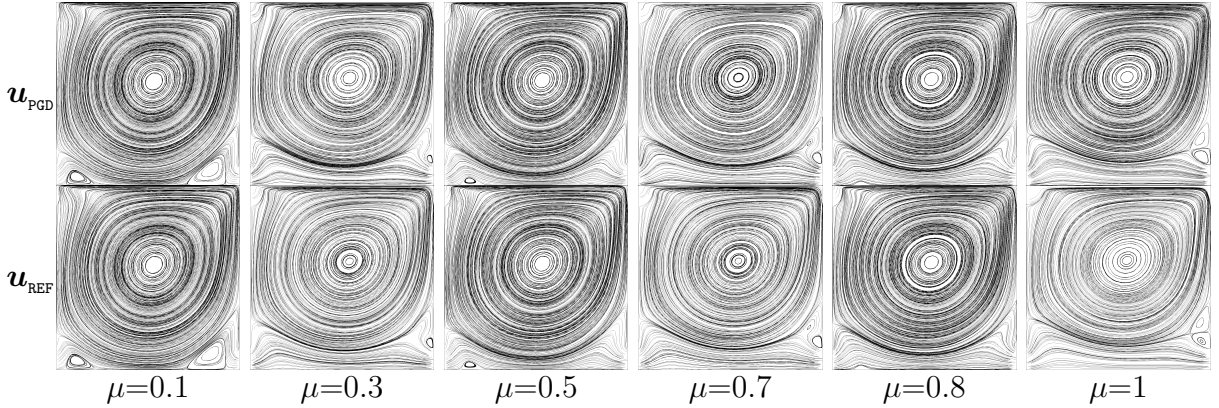


Figure 10: Comparison of the PGD approximation (top) and the full-order solution (bottom) of the lid-driven cavity with jets for $\mu=0.1$, $\mu=0.3$, $\mu=0.5$, $\mu=0.7$, $\mu=0.8$ and $\mu=1$, corresponding to a maximum velocity of the jets spanning from 0.2 to 2 m/s.

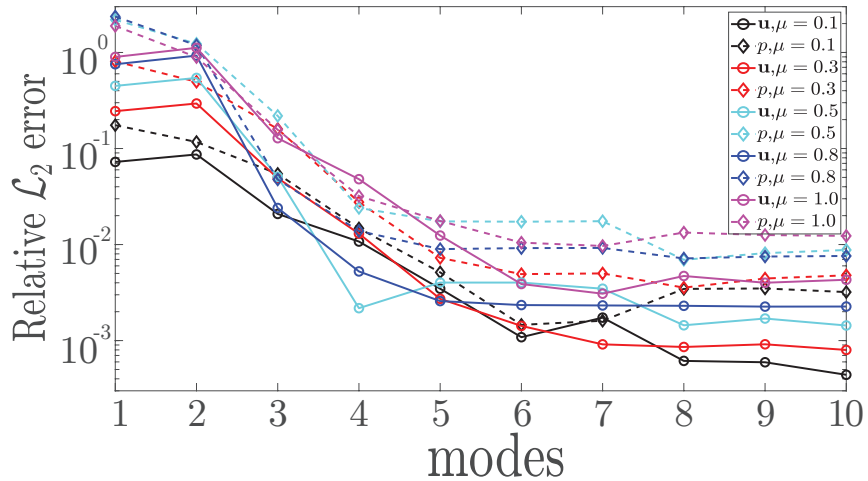


Figure 11: Relative $\mathcal{L}_2(\Omega)$ error of the PGD approximation of the cavity flow with parametrised jet velocity with respect to the full-order solution as a function of the global number of modes (i.e. boundary conditions and computed) utilised in the PGD expansion, for $\mu=0.1$, $\mu=0.5$, $\mu=0.7$ and $\mu=1$.

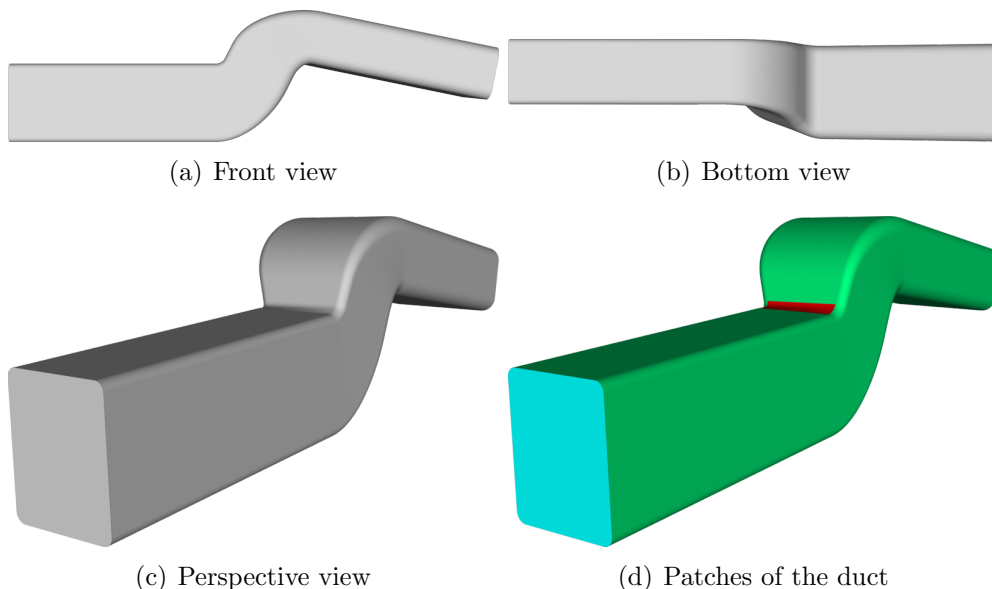


Figure 12: Geometrical model of the S-Bend. On the bottom-right image, the jet patch is highlighted in red.

5.2 S-Bend with flow control driven by a jet

In this section, the proposed PGD methodology is applied to a flow control problem using a three-dimensional geometry of industrial interest. The model of a heating, ventilation and air conditioning (HVAC) duct section provided by Volkswagen AG is shown in Figure 12. A jet is introduced on the red patch, at the first bend of the duct. The velocity profile of the jet is a sinusoidal function defined on the reference planar square $[0, 1]^2$ as

$$u_{\hat{y}}(\hat{x}, \hat{z}) = 0.0375(1 - \cos(-2\pi\hat{x}))(1 - \cos(2\pi\hat{z})) \quad (20)$$

and pointing in the direction \hat{y} orthogonal to the plane (\hat{x}, \hat{z}) . The parametrisation is constructed as a scaling of the jet velocity from $u_y = -0.015$ m/s, i.e. blowing, to suction with $u_y = 0.15$ m/s. A single parameter μ is introduced and the parametric domain considered for the analysis is $\mathcal{I} = [-0.1, 1]$. Note that this problem is especially challenging due to the change of sign in the interval of parametric values considered leading to different physical phenomena. The remaining boundary conditions feature homogeneous velocity on all the lateral walls, a parabolic velocity profile with mean value $\mathbf{u} = (0.83, 0, 0)$ m/s on the inlet and a free-traction on the outlet. The dynamic viscosity is set to $\nu = 1.588 \times 10^{-5}$ m²/s and the corresponding value of the Reynolds number is $\text{Re} = 280$. The quantity of interest in this problem is the pressure drop computed along the duct.

As previously done for the lid-driven cavity with jets, two modes to account for the boundary conditions are computed using `simpleFoam`. The first mode is a full-order solution corresponding to the case of inactive jet and given inlet parabolic profile; the second one, is obtained setting a zero inlet velocity and a jet of maximum velocity $u_y = 0.15$ m/s.

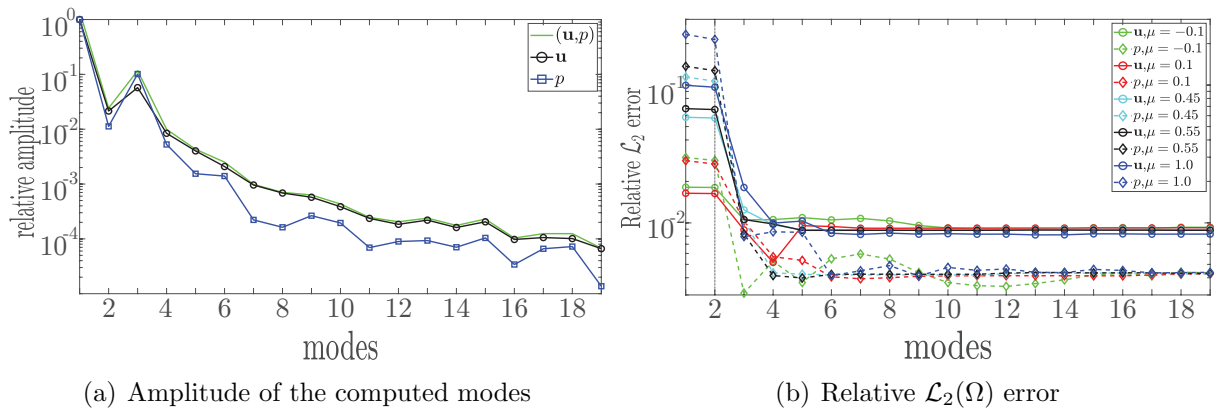


Figure 13: Internal flow in the S-Bend with parametrised jet velocity. (a) Relative amplitude of the computed modes \mathbf{f}_u^m (black), \mathbf{f}_p^m (blue) and the combined amplitude of $(\mathbf{f}_u^m, \mathbf{f}_p^m)$ according to Equation (18). (b) Relative $\mathcal{L}_2(\Omega)$ error of the PGD approximations of pressure and velocity with respect to the full-order solutions for different values of μ .

The corresponding parametric modes are $\phi(\mu)=1$ and $\phi(\mu)=\mu$, respectively.

Setting a tolerance of 10^{-3} , `pgdFoam` computes five modes before fulfilling the stopping criterion for $\eta_{(u,p)}$, see Equation (18), whereas seventeen computed modes are required for the amplitude to drop at 10^{-4} , as displayed in Figure 13(a).

The PGD approximation obtained is compared with the full-order solutions given by `simpleFoam` for the values $\mu=-0.1$, $\mu=0.45$ and $\mu=1$ of the parameter under analysis. In Figure 13(b), the relative $\mathcal{L}_2(\Omega)$ error for these configurations is reported. The numerical experiments confirm that an accuracy of 10^{-2} is achieved, for all parameters, using two computed modes additionally to the two terms accounting for the boundary conditions. Note that the first computed mode is one order of magnitude more relevant than the following ones (Fig. 13(a)) and after two computed modes only limited corrections are introduced to the existing PGD approximation. It is worth observing that a comparable accuracy of the PGD solution is achieved throughout the whole parametric space for both velocity and pressure.

A qualitative comparison of the pressure and velocity fields computed using the PGD solution interpolated in different points of the parametric interval \mathcal{I} and the corresponding full-order discretisations is presented in Figures 14 and 15.

As mentioned at the beginning of this section, the quantity of engineering interest in the analysis of this problem is the pressure drop computed along the duct. Considering a zero reference pressure on the outlet, the weighted average pressure drop is defined as

$$p_{\text{drop}} := \frac{1}{A_{\text{in}}} \sum_{i=1}^{N_{\text{in}}} A_i p_i, \quad (21)$$

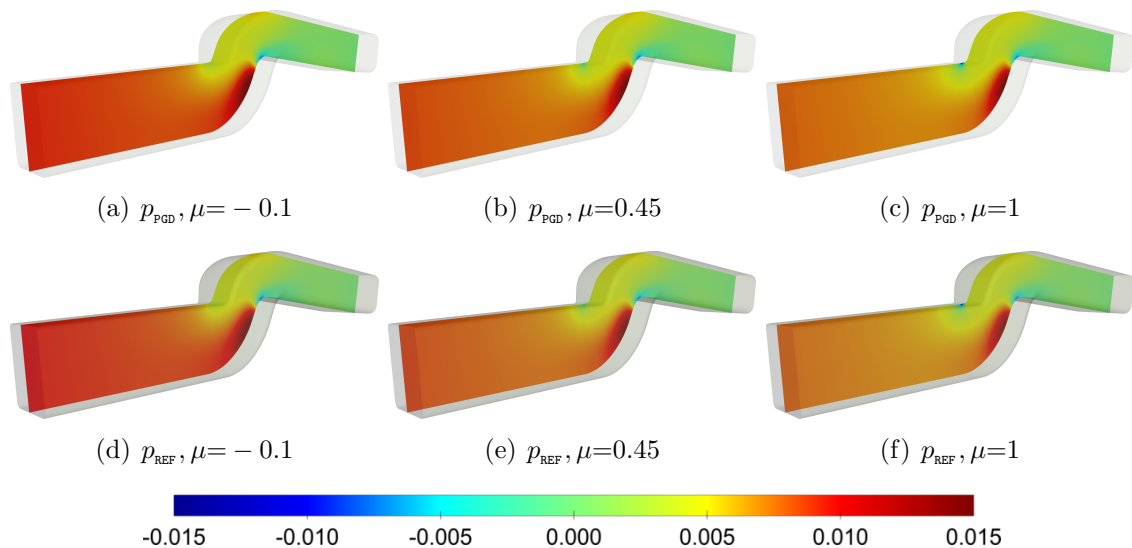


Figure 14: Comparison of the PGD approximation (top) and the full-order solution (bottom) of the pressure field of the internal flow in the S-Bend with a jet configuration of $\mu = -0.1$, $\mu = 0.45$ and $\mu = 1$, corresponding to a jet which spans from blowing at maximum velocity 0.015 m/s to suction with peak velocity 0.15 m/s.

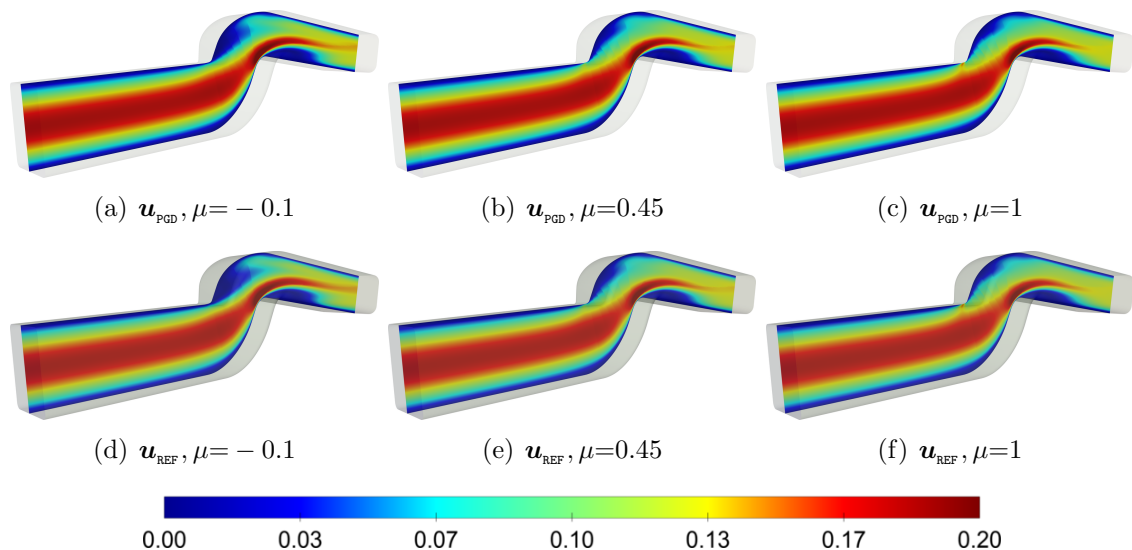


Figure 15: Comparison of the PGD approximation (top) and the full-order solution (bottom) of the velocity field of the internal flow in the S-Bend with a jet configuration of $\mu = -0.1$, $\mu = 0.45$ and $\mu = 1$, corresponding to a jet which spans from blowing at maximum velocity 0.015 m/s to suction with peak velocity 0.15 m/s.

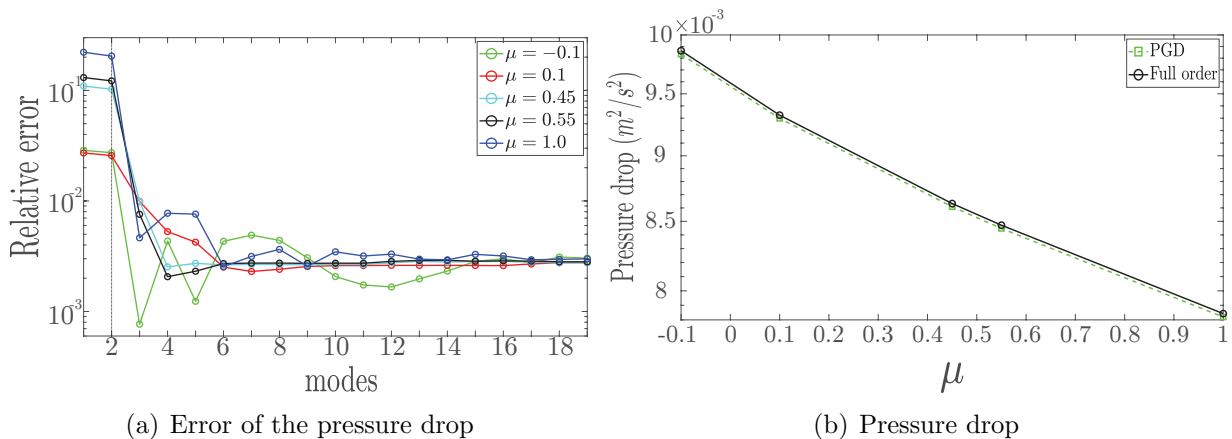


Figure 16: PGD approximation of the internal flow in the S-Bend with parametrised jet velocity. (a) Relative error of the pressure drop enriching the PGD modal approximation. (b) Pressure drop with respect to the maximum jet velocity.

where A_{in} is the area of the inlet surface, N_{in} the number of faces S_i , $i=1, \dots, N_{\text{in}}$ on the inlet patch and p_i , A_i are the pressure and area on the face S_i , respectively. For $\mu=-0.1$, $\mu=0.1$, $\mu=0.45$, $\mu=0.55$ and $\mu=1$, the pressure drop is evaluated interpolating the generalised PGD solution in the corresponding values of the parametric space and using the full-order solver `simpleFoam`. Figure 16(a) presents the convergence history of the error in the pressure drop as a function of the number of modes in the PGD approximation. It is straightforward to observe again that using the modes accounting for the boundary conditions and two computed modes is sufficient to capture the flow features of a wide range of parameters, with a maximum error in the pressure drop below 10^{-2} . Moreover, by comparing the pressure drop with respect to the maximum velocity of the jet for different configurations with the corresponding values provided by the full-order solver, the capability of the discussed reduced-order strategy to accurately capture the evolution of a quantity of interest throughout the range of values of the parameter μ is confirmed (Fig. 16(b)). Finally, these experiments highlight that the proposed PGD algorithm is able to provide an accurate approximation of the solution both in terms of a global measure of the error in the domain, see the \mathcal{L}_2 error in Figure 13(b), and of a localised quantity of interest as the pressure drop in Figure 16(a).

6 Concluding remarks

A nonintrusive PGD implementation in OpenFOAM has been proposed in the context of parametrised incompressible laminar flows. The main novelty of such approach is represented by the seamless exploitation of OpenFOAM native SIMPLE solver, making the resulting reduced-order strategy suitable for application in a daily industrial environment. The `pgdFoam` algorithm relies on the industrially-validated solver `simpleFoam` to compute

the spatial modes of the solution, whereas the parametric ones are determined via the solution of a linear system of algebraic equations.

The developed strategy has been validated using a manufactured solution to verify the optimal order of convergence of the PGD-FV approximation and the classical benchmark test of an incompressible flow in a nonleaky lid-driven cavity parametrising the Reynolds number. Moreover, the capability of the proposed `pgdFoam` algorithm for fast and accurate simulation of flow control problems has been verified, introducing parametric jets in a 2D cavity flow and in a 3D geometrical model of an air duct provided by Volkswagen AG.

The proposed PGD methodology has proved to be able to compute an accurate reduced basis for the problems under analysis with no *a priori* knowledge of the expected solutions. Accuracy of 10^{-2} or higher was achieved for the \mathcal{L}_2 error of the generalised PGD solution with respect to the full-order one, in all computations using between ten and twenty modes. Moreover, the `pgdFoam` algorithm has shown robustness dealing with a large range of values of the parameters (e.g., Reynolds number from 1,000 to 4,000) and spanning different physical configurations (e.g., control jets from suction to blowing). On the one hand, the generalised PGD solution has proved to be able to qualitatively capture significant topological changes as well as localised features of the flow in an accurate way. On the other hand, it has shown reliability in the quantitative evaluation of engineering outcomes providing a maximum relative error of less than 10^{-2} , for all values of user-defined parameters, with a computing effort between ten and twenty runs of the full-order solver.

Acknowledgements

This work was partially supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Actions (Grant agreement No. 675919) that financed the Ph.D. fellowship of the first author. The second, third and last author were also supported by the Spanish Ministry of Economy and Competitiveness (Grant agreement No. DPI2017-85139-C2-2-R). The second and last authors are grateful for the financial support provided by the Generalitat de Catalunya (Grant agreement No. 2017-SGR-1278).

References

- [1] R. J. LeVeque. *Finite volume methods for hyperbolic problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 2002.
- [2] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics*. Springer-Verlag, Berlin, third edition, 2009. A practical introduction.

- [3] K. W. Morton and T. Sonar. Finite volume methods for hyperbolic conservation laws. *Acta Numer.*, 16:155–238, 2007.
- [4] T. Barth, R. Herbin, and M. Oehlberger. Finite Volume Methods: Foundation and Analysis. In *Encyclopedia of Computational Mechanics Second Edition*, pages 1–60. American Cancer Society, 2017.
- [5] R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of Numerical Analysis*, 7:713 – 1018, 2000. Solution of Equation in \mathbb{R}^n (Part 3), Techniques of Scientific Computing (Part 3).
- [6] Ruben Sevilla, Matteo Giacomini, and Antonio Huerta. A face-centred finite volume method for second-order elliptic problems. *Int. J. Numer. Methods Eng.*, 115(8):986–1014, 2018.
- [7] L. M. Vieira, M. Giacomini, R. Sevilla, and A. Huerta. A second-order face-centred finite volume method for elliptic problems. *Comput. Methods Appl. Mech. Eng.*, 2020. to appear.
- [8] F. Chinesta, A. Huerta, G. Rozza, and K. Willcox. Model Reduction Methods. In Erwin Stein, Rene de Borst, and Thomas J. R. Hughes, editors, *Encyclopedia of Computational Mechanics Second Edition*, volume Part 1 Solids and Structures, chapter 3, pages 1–36. John Wiley & Sons, Ltd., Chichester, 2017.
- [9] B. Peherstorfer, K. Willcox, and M. Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Rev.*, 60(3):550–591, 2018.
- [10] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Ser. I-Math.*, 339(9):667 – 672, 2004.
- [11] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: M2AN*, 39(1):157–181, 2005.
- [12] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: M2AN*, 41(3):575–605, 2007.
- [13] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Arch. Comput. Methods Eng.*, 15(3):229, May 2008.
- [14] S. Chaturantabut and D. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737–2764, 2010.

- [15] L. Iapichino, A. Quarteroni, and G. Rozza. A reduced basis hybrid method for the coupling of parametrized domains represented by fluidic networks. *Comput. Methods Appl. Mech. Eng.*, 221-222:63 – 82, 2012.
- [16] I. Martini, G. Rozza, and B. Haasdonk. Reduced basis approximation and a-posteriori error estimation for the coupled Stokes-Darcy system. *Adv. Comput. Math.*, 41(5):1131–1157, Oct 2015.
- [17] A. Iollo, S. Lanteri, and J.-A. Désidéri. Stability Properties of POD-Galerkin Approximations for the Compressible Navier-Stokes Equations. *Theor. Comput. Fluid Dyn.*, 13(6):377–396, Mar 2000.
- [18] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM J. Numer. Anal.*, 40(2):492–515, 2002.
- [19] M. Bergmann, C.-H. Bruneau, and A. Iollo. Enablers for robust POD models. *J. Comput. Phys.*, 228(2):516 – 538, 2009.
- [20] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa. A numerical investigation of velocity-pressure reduced order models for incompressible flows. *J. Comput. Phys.*, 259:598 – 616, 2014.
- [21] N. Akkari, A. Hamdouni, E. Liberge, and M. Jazar. A mathematical and numerical study of the sensitivity of a reduced order model by POD (ROM-POD), for a 2d incompressible fluid flow. *Journal of Computational and Applied Mathematics*, 270:522 – 530, 2014. Fourth International Conference on Finite Element Methods in Engineering and Sciences (FEMTEC 2013).
- [22] F. Ballarin, A. Manzoni, A. Quarteroni, and G. Rozza. Supremizer stabilization of POD-Galerkin approximation of parametrized steady incompressible Navier-Stokes equations. *Int. J. Numer. Methods Eng.*, 102(5):1136–1161, 2015.
- [23] E. Longatte, E. Liberge, M. Pomarede, J.-F. Sigrist, and A. Hamdouni. Parametric study of flow-induced vibrations in cylinder arrays under single-phase fluid cross flows using POD-ROM. *Journal of Fluids and Structures*, 78:314 – 330, 2018.
- [24] S. Perotto, A. Ern, and A. Veneziani. Hierarchical Local Model Reduction for Elliptic Problems: A Domain Decomposition Approach. *Multiscale Model. Simul.*, 8(4):1102–1127, 2010.
- [25] S. Perotto and A. Veneziani. Coupled model and grid adaptivity in hierarchical reduction of elliptic problems. *J. Sci. Comput.*, 60(3):505–536, Sep 2014.
- [26] M. C. Aletti, S. Perotto, and A. Veneziani. Himod reduction of advection–diffusion–reaction problems with general boundary conditions. *J. Sci. Comput.*, 76(1):89–119, Jul 2018.

- [27] S. Guzzetti, S. Perotto, and A. Veneziani. Hierarchical model reduction for incompressible fluids in pipes. *Int. J. Numer. Methods Eng.*, 114(5):469–500, 2018.
- [28] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *J. Non-Newton. Fluid*, 139(3):153 – 176, 2006.
- [29] F. Chinesta, P. Ladevèze, and E. Cueto. A Short Review on Model Order Reduction Based on Proper Generalized Decomposition. *Arch. Comput. Methods Eng.*, 18(4):395, Oct 2011.
- [30] F. Chinesta, A. Leygue, F. Bordeu, J. V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. PGD-Based Computational Vademecum for Efficient Design, Optimization and Control. *Arch. Comput. Methods Eng.*, 20(1):31–59, Mar 2013.
- [31] Francisco Chinesta, Elias Cueto, and Antonio Huerta. PGD for solving multidimensional and parametric models. In F. Chinesta and P. Ladevèze, editors, *Separated representations and PGD-based model reduction*, volume 554 of *CISM Courses and Lectures*, pages 27–89. Springer, Vienna, 2014.
- [32] Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The proper generalized decomposition for advanced numerical simulations. A primer*. Springer Briefs in Applied Sciences and Technology. Springer, Cham, 2014.
- [33] D. Ryckelynck, F. Chinesta, E. Cueto, and A. Ammar. On the a priori model reduction: Overview and recent developments. *Archives of Computational Methods in Engineering*, 13(1):91–128, Mar 2006.
- [34] N. Verdon, C. Allery, C. Béghein, A. Hamdouni, and D. Ryckelynck. Reduced-order modelling for solving linear and non-linear equations. *International Journal for Numerical Methods in Biomedical Engineering*, 27(1):43–58, 2011.
- [35] C. Allery, A. Hamdouni, D. Ryckelynck, and N. Verdon. A priori reduction method for solving the two-dimensional Burgers’ equations. *Applied Mathematics and Computation*, 217(15):6671 – 6679, 2011.
- [36] A. Dumon, C. Allery, and A. Ammar. Proper Generalized Decomposition method for incompressible flows in stream-vorticity formulation. *European Journal of Computational Mechanics*, 19(5-7):591–617, 2010.
- [37] A. Dumon, C. Allery, and A. Ammar. Proper general decomposition (PGD) for the resolution of Navier-Stokes equations. *Journal of Computational Physics*, 230(4):1387 – 1407, 2011.

- [38] C. Leblond and C. Allery. A priori space-time separated representation for the reduced order modeling of low Reynolds number flows. *Computer Methods in Applied Mechanics and Engineering*, 274:264 – 288, 2014.
- [39] D. González, L. Debeugny, E. Cueto, F. Chinesta, P. Díez, and A. Huerta. SUPG-based stabilization using a separated representations approach. *Int. J. Mater. Form.*, 3(1):883–886, 2010.
- [40] D. González, E. Cueto, F. Chinesta, P. Díez, and A. Huerta. Streamline upwind/Petrov-Galerkin-based stabilization of proper generalized decompositions for high-dimensional advection-diffusion equations. *Int. J. Numer. Methods Eng.*, 94(13):1216–1232, 2013.
- [41] D. Modesto, S. Zlotnik, and A. Huerta. Proper generalized decomposition for parameterized Helmholtz problems in heterogeneous and unbounded domains: Application to harbor agitation. *Comput. Methods Appl. Mech. Eng.*, 295:127 – 149, 2015.
- [42] Pedro Díez, Sergio Zlotnik, and Antonio Huerta. Generalized parametric solutions in Stokes flow. *Comput. Methods Appl. Mech. Eng.*, 326:223–240, 2017.
- [43] A. Huerta, E. Nadal, and F. Chinesta. Proper generalized decomposition solutions within a domain decomposition strategy. *Int. J. Numer. Methods Eng.*, 113(13):1972–1994, 2018.
- [44] A. Sibileau, A. García-González, F. Auricchio, S. Morganti, and P. Díez. Explicit parametric solutions of lattice structures with Proper Generalized Decomposition (PGD): Applications to the design of 3D-printed architected materials. *Comput. Mech.*, 62(4):871–891, 2018.
- [45] Amine Ammar, Antonio Huerta, Francisco Chinesta, Elías Cueto, and Adrien Leygue. Parametric solutions involving geometry: a step towards efficient shape optimization. *Comput. Methods Appl. Mech. Eng.*, 268:178–193, 2014.
- [46] Sergio Zlotnik, Pedro Díez, David Modesto, and Antonio Huerta. Proper Generalized Decomposition of a geometrically parametrized heat problem with geophysical applications. *Int. J. Numer. Methods Eng.*, 103(10):737–758, 2015.
- [47] M. Signorini, S. Zlotnik, and P. Díez. Proper generalized decomposition solution of the parameterized Helmholtz problem: application to inverse geophysical problems. *Int. J. Numer. Methods Eng.*, 109(8):1085–1102, 2017.
- [48] R. Sevilla, S. Zlotnik, and A. Huerta. Solution of geometrically parametrised problems within a CAD environment via model order reduction. *Comput. Methods Appl. Mech. Eng.*, 358:112631, 2020.

- [49] A. E. Deane, I. G. Kevrekidis, G. E. Karniadakis, and S. A. Orszag. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Phys. Fluids A: Fluid Dynamics*, 3(10):2337–2354, 1991.
- [50] X. Ma and G.E. Karniadakis. A low-dimensional model for simulating three-dimensional cylinder flow. *J. Fluid Mech.*, 458:181–190, 2002.
- [51] R. Zimmermann, A. Vendl, and S. Görtz. Reduced-order modeling of steady flows subject to aerodynamic constraints. *AIAA Journal*, 52(2):255–266, 2014.
- [52] P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge Monographs on Mechanics. Cambridge University Press, 1996.
- [53] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: M2AN*, 42(2):277–302, 2008.
- [54] M. Drohmann, B. Haasdonk, and M. Ohlberger. Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. *SIAM J. Sci. Comput.*, 34(2):A937–A969, 2012.
- [55] S. Lorenzi, A. Cammi, L. Luzzi, and G. Rozza. POD-Galerkin method for finite volume approximation of Navier-Stokes and RANS equations. *Comput. Methods Appl. Mech. Eng.*, 311:151 – 179, 2016.
- [56] G. Stabile and G. Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier-Stokes equations. *Comput. Fluids*, 173:273 – 284, 2018.
- [57] A. Tallet, C. Allery, C. Leblond, and E. Liberge. A minimum residual projection to build coupled velocity-pressure POD-ROM for incompressible Navier-Stokes equations. *Communications in Nonlinear Science and Numerical Simulation*, 22(1):909 – 932, 2015.
- [58] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, 2011.
- [59] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem. The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623 – 647, 2013.
- [60] K. Carlberg, M. Barone, and H. Antil. Galerkin v. least-squares Petrov-Galerkin projection in nonlinear model reduction. *Journal of Computational Physics*, 330:693 – 734, 2017.

- [61] K. Carlberg, Y. Choi, and S. Sargsyan. Conservative model reduction for finite-volume models. *J. Comput. Phys.*, 371:280 – 314, 2018.
- [62] P. Destuynder and B. Métivet. Explicit error bounds in a conforming finite element method. *Math. Comp.*, 68(228):1379–1396, 1999.
- [63] A. Ern, A. F. Stephansen, and M. Vohralík. Guaranteed and robust discontinuous Galerkin a posteriori error estimates for convection-diffusion-reaction problems. *J. Comput. Appl. Math.*, 234(1):114 – 130, 2010.
- [64] A. Ern and M. Vohralík. Polynomial-Degree-Robust A Posteriori Estimates in a Unified Setting for Conforming, Nonconforming, Discontinuous Galerkin, and Mixed Discretizations. *SIAM J. Numer. Anal.*, 53(2):1058–1081, 2015.
- [65] N. Parés, P. Díez, and A. Huerta. Subdomain-based flux-free a posteriori error estimators. *Comput. Methods Appl. Mech. Eng.*, 195(4-6):297–323, 2006.
- [66] R. Cottreau, P. Díez, and A. Huerta. Strict error bounds for linear solid mechanics problems using a subdomain-based flux-free method. *Comput. Mech.*, 44(4):533–547, 2009.
- [67] N. Parés and P. Díez. A new equilibrated residual method improving accuracy and efficiency of flux-free error estimates. *Comput. Methods Appl. Mech. Eng.*, 313(1):785 – 816, 2017.
- [68] J.T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Comput. Math. Appl.*, 41(5):735 – 756, 2001.
- [69] N. Parés, J. Bonet, A. Huerta, and J. Peraire. The computation of bounds for linear-functional outputs of weak solutions to the two-dimensional elasticity equations. *Comput. Methods Appl. Mech. Eng.*, 195(4-6):406–429, 2006.
- [70] N. Parés, P. Díez, and A. Huerta. Exact bounds for linear outputs of the advection-diffusion-reaction equation using flux-free error estimates. *SIAM J. Sci. Comput.*, 31(4):3064–3089, 2009.
- [71] F. Larsson, P. Díez, and A. Huerta. A flux-free a posteriori error estimator for the incompressible Stokes problem using a mixed FE formulation. *Comput. Methods Appl. Mech. Eng.*, 199(37-40):2383–2402, 2010.
- [72] M. Ainsworth and R. Rankin. Guaranteed computable bounds on quantities of interest in finite element computations. *Int. J. Numer. Methods Eng.*, 89(13):1605–1634, 2012.
- [73] I. Mozolevski and S. Prudhomme. Goal-oriented error estimation based on equilibrated-flux reconstruction for finite element approximations of elliptic problems. *Comput. Methods Appl. Mech. Eng.*, 288:127 – 145, 2015.

- [74] Giacomini, M., Pantz, O., and Trabelsi, K. Certified descent algorithm for shape optimization driven by fully-computable a posteriori error estimators. *ESAIM: COCV*, 23(3):977–1001, 2017.
- [75] M. Giacomini. An Equilibrated Fluxes Approach to the Certified Descent Algorithm for Shape Optimization Using Conforming Finite Element and Discontinuous Galerkin Discretizations. *J. Sci. Comput.*, 75(1):560–595, Apr 2018.
- [76] X. Zou, M. Conti, P. Díez, and F. Auricchio. A nonintrusive proper generalized decomposition scheme with application in biomechanics. *Int. J. Numer. Methods Eng.*, 113(2):230–251, 2018.
- [77] Amaury Courard, David Néron, Pierre Ladevèze, and Ludovic Ballere. Integration of PGD-virtual charts into an engineering design process. *Comput. Mech.*, 57(4):637–651, 2016.
- [78] A. Bertram, C. Othmer, and R. Zimmermann. Towards real-time vehicle aerodynamic design via multi-fidelity data-driven reduced order modeling. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018.
- [79] The OpenFOAM foundation. OpenFOAM 6.0, 2019. [Accessed 5-February-2019].
- [80] S.V Patankar and D.B Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transfer*, 15(10):1787 – 1806, 1972.
- [81] Jean Donea and Antonio Huerta. *Finite element methods for flow problems*. John Wiley & Sons, Chichester, 2003.
- [82] A. Nouy. Generalized spectral decomposition method for solving stochastic finite element equations: Invariant subspace problem and dedicated algorithms. *Comput. Methods Appl. Mech. Eng.*, 197(51):4718 – 4736, 2008.
- [83] L. Tamellini, O. Le Maître, and A. Nouy. Model reduction based on proper generalized decomposition for the stochastic steady incompressible Navier-Stokes equations. *SIAM J. Sci. Comput.*, 36(3):A1089–A1117, 2014.
- [84] Anthony T. Patera and Gianluigi Rozza. Reduced Basis Approximation and A-Posteriori Error Estimation for Parametrized Partial Differential Equations. MIT Pappalardo Graduate Monographs in Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA, 2007.
- [85] Gianluigi Rozza. Fundamentals of reduced basis method for problems governed by parametrized PDEs and applications. In F. Chinesta and P. Ladevèze, editors, *Separated representations and PGD-based model reduction*, volume 554 of *CISM Courses and Lectures*, pages 153–227. Springer, Vienna, 2014.

- [86] L. I. G. Kovasznay. Laminar flow behind a two-dimensional grid. *Proc. Cambridge Phi. Sc.*, 44:58–62, 1947.
- [87] U. Ghia, K. N. Ghia, and C. T. Shin. High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. *J. Comput. Phys.*, 48:387–411, 1982.
- [88] R. Duvigneau and M. Visonneau. Optimization of a synthetic jet actuator for aerodynamic stall control. *Comput. Fluids*, 35(6):624 – 638, 2006.
- [89] L. Dedè. Optimal flow control for Navier-Stokes equations: drag minimization. *Int. J. Numer. Methods Fluids*, 55(4):347–366, 2007.
- [90] E. Guilmineau, R. Duvigneau, and J. Labroquère. Optimization of jet parameters to control the flow on a ramp. *C. R. Acad. Sci. Ser. II-Mec.*, 342(6):363 – 375, 2014. Flow separation control.
- [91] R. Temam. *Navier-Stokes equations. Theory and numerical analysis*. North-Holland Publishing Co., Amsterdam-New York-Oxford, 1977.
- [92] J.-L. Guermond and L. Quartapelle. On the approximation of the unsteady Navier-Stokes equations by finite element projection methods. *Numer. Math.*, 80(2):207–238, 1998.
- [93] J.-L. Guermond and L. Quartapelle. On stability and convergence of projection methods based on pressure Poisson equation. *Int. J. Numer. Methods Fluids*, 26(9):1039–1053, 1998.
- [94] A. Quarteroni. *Numerical models for differential problems*, volume 16 of *MS&A. Modeling, Simulation and Applications*. Springer, Cham, 2017.
- [95] H. Laval and L. Quartapelle. A fractional-step Taylor–Galerkin method for unsteady incompressible flows. *Int. J. Numer. Methods Fluids*, 11(5):501–513, 1990.
- [96] A. Quarteroni, F. Saleri, and A. Veneziani. Factorization methods for the numerical approximation of Navier-Stokes equations. *Comput. Methods Appl. Mech. Eng.*, 188(1–3):505–526, 2000.

A Separated representation of the residuals

Consider a separable expression of the source term $\mathbf{s}(\mathbf{x}, \boldsymbol{\mu}) := \eta(\boldsymbol{\mu})\mathbf{S}(\mathbf{x})$. For the spatial iteration, the residuals in separated form read as

$$R_u^n := \int_{V_i} \alpha_4 \mathbf{S} dV - \sum_{m=1}^n \sum_{q=1}^n \alpha_5^{mq} \int_{V_i} \nabla \cdot (\sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^q \mathbf{f}_u^q) dV$$

$$+ \int_{V_i} \nabla \cdot \left(D \nabla \left(\sum_{m=1}^n \alpha_6^m \sigma_u^m \mathbf{f}_u^m \right) \right) dV - \int_{V_i} \nabla \cdot \left(\sum_{m=1}^n \alpha_7^m \sigma_p^m \mathbf{f}_p^m \right) dV, \quad (22a)$$

$$R_p^n := - \int_{V_i} \nabla \cdot \left(\sum_{m=1}^n \alpha_7^m \sigma_u^m \mathbf{f}_u^m \right) dV, \quad (22b)$$

where the following expressions for the coefficients are devised

$$\begin{cases} \alpha_4 := \int_{\mathcal{I}} \phi^n \eta d\mathcal{I}, & \alpha_5^{mq} := \int_{\mathcal{I}} \phi^n \phi^m \phi^q d\mathcal{I}, \\ \alpha_6^m := \int_{\mathcal{I}} \phi^n \phi^m \psi d\mathcal{I}, & \alpha_7^m := \int_{\mathcal{I}} \phi^n \phi^m d\mathcal{I}. \end{cases} \quad (23)$$

For the parametric iteration, the separated expression of the residuals is

$$r_u^n := a_4 \eta + \sum_{m=1}^n \left(- \sum_{q=1}^n a_5^{mq} \phi^q + a_6^m \psi - a_7^m \right) \phi^m, \quad (24a)$$

$$r_p := - \sum_{m=1}^n a_8^m \phi^m, \quad (24b)$$

where the coefficients depend solely on the spatial modes, namely

$$\begin{cases} a_4 := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot \mathbf{S} dV, \\ a_5^{mq} := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot [\nabla \cdot (\sigma_u^m \mathbf{f}_u^m \otimes \sigma_u^q \mathbf{f}_u^q)] dV, \\ a_6^m := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot [\nabla \cdot (D \nabla (\sigma_u^m \mathbf{f}_u^m))] dV, \\ a_7^m := \int_{V_i} \sigma_u^n \mathbf{f}_u^n \cdot \nabla (\sigma_p^m \mathbf{f}_p^m) dV, \\ a_8^m := \int_{V_i} \sigma_p^n \mathbf{f}_p^n \cdot \nabla (\sigma_u^m \mathbf{f}_u^m) dV. \end{cases} \quad (25)$$

B simpleFoam: the semi-implicit method for pressure linked equations in OpenFOAM

In OpenFOAM, the steady laminar Navier-Stokes equations are approximated by means of an iterative procedure, namely `simpleFoam`. This algorithm implements the SIMPLE method proposed in [80]. SIMPLE is a fractional-step Chorin-Temam projection method [91, Sect. 3.7] that has been extensively studied in the literature [92, 93]. A description of the relationship between SIMPLE and Chorin-Temam projection methods is detailed in [94, Sect. 17.7, 17.8], whereas the implementation details of `simpleFoam` are provided in the official OpenFOAM documentation [79].

First, an intermediate velocity \mathbf{u}^k is computed starting from the momentum equation and neglecting the contribution of pressure, see Equation (26a); second, the step involving the incompressibility constraint is rewritten in terms of a Poisson equation for the pressure p , see Equation (26b); eventually, a correction is applied to the intermediate velocity field to determine the final value \mathbf{u} in Equation (26c). Special attention is required to impose the correct set of boundary conditions in each step of the algorithm [95].

$$\left\{ \begin{array}{ll} \frac{\mathbf{u}^k - \mathbf{u}^{k-1}}{\Delta t} + \nabla \cdot (\mathbf{u}^k \otimes \mathbf{u}^{k-1}) - \nabla \cdot (\nu \nabla \mathbf{u}^k) = \mathbf{s} & \text{in } \Omega, \\ \mathbf{u}^k = \mathbf{u}_D & \text{on } \Gamma_D, \\ \mathbf{n} \cdot (\nu \nabla \mathbf{u}^k) = \mathbf{t} & \text{on } \Gamma_N, \end{array} \right. \quad (26a)$$

$$\left\{ \begin{array}{ll} \nabla \cdot (\nabla p) = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^k & \text{in } \Omega, \\ \mathbf{n} \cdot \nabla p = 0 & \text{on } \Gamma_D, \\ \mathbf{n} p = \mathbf{0} & \text{on } \Gamma_N, \end{array} \right. \quad (26b)$$

$$\mathbf{u} = \mathbf{u}^k - \Delta t \nabla p. \quad (26c)$$

Remark 5. A variant of the above numerical scheme, known as *incremental* Chorin-Temam projection method, is obtained by adding the term $-\nabla p^k$ to the right-hand side of Equation (26a) to compute the velocity prediction. Hence, the Poisson problem for pressure in Equation (26b) is solved to compute a pressure variation Δp , whereas the correction step in Equation (26c) remains unchanged, now being $p = p^k + \Delta p$.

Note that the algorithm in Equation (26) may also be rewritten in the framework of an algebraic splitting method [96]. For a complete introduction to the subject, interested readers are referred to [81, Sect. 6.7].