Hybrid Data-Driven Closure Strategies for Reduced Order Modeling

Anna Ivagnes

SISSA, International School for Advanced Studies,

Mathematics Area, mathLab, Trieste, Italy.

aivagnes@sissa.it

Giovanni Stabile

SISSA, International School for Advanced Studies,

Mathematics Area, mathLab, Trieste, Italy.

gstabile@sissa.it

Andrea Mola

Multi-scale Analysis of Materials Unit,

Scuola IMT Alti Studi, Lucca, Italy.

andrea.mola@imtlucca.it

Traian Iliescu

Department of Mathematics, Virginia Tech,

Blacksburg, VA, USA.

iliescu@vt.edu

Gianluigi Rozza

SISSA, International School for Advanced Studies,

Mathematics Area, mathLab, Trieste, Italy.

grozza@sissa.it

Abstract

In this paper, we propose hybrid data-driven ROM closures for fluid flows. These new ROM closures combine two fundamentally different strategies: (i) purely data-driven ROM closures, both for the velocity and the pressure; and (ii) physically based, eddy viscosity data-driven closures, which model the energy transfer in the system. The first strategy consists in the addition of closure/correction terms to the governing equations, which are built from the available data. The second strategy includes turbulence modeling by adding eddy viscosity terms, which are determined by using machine learning techniques. The two strategies are combined for the first time in this paper to investigate a two-dimensional flow past a circular cylinder at $Re = 50\,000$. Our numerical results show that the hybrid data-driven ROM is more accurate than both the purely data-driven ROM and the eddy viscosity ROM.

1. Introduction

Full order models (FOMs) are computational models that are obtained by using classical numerical methods, such as the finite element or the finite volume methods. The FOM computational cost is prohibitively high for many important engineering, environmental, and biomedical applications that require repeated turbulent flow simulations, such as [53].

Reduced Order Models (ROMs), presented in [9, 10, 11, 45, 46, 48] and applied to finite-volume schemes in [16, 14], can reduce the FOM computational cost by orders of magnitude. ROMs can be built by using different methodologies, such as least-squares Petrov–Galerkin projection [13] or Galerkin projection [38, 12, 30, 6]. In this paper, we consider the class of Galerkin-ROMs (G-ROMs), which are ROMs constructed by using a Galerkin method. The standard methodology consists in the construction of a set of basis functions (modes) for the velocity and pressure fields,

1

$\{\varphi_1, \ldots, \varphi_r\}$ and $\{\chi_1, \ldots, \chi_q\}$, respectively, such that the unknown solution can be approximated as a linear combination of these basis functions: $u(x,t) = \sum_{i=1}^{r} a_i(t)\,\varphi_i(x)$, $p(x,t) = \sum_{i=1}^{q} b_i(t)\,\chi_i(x)$. The G-ROM consists in the resolution of a dynamical system, which is obtained by projecting the governing equations onto the space spanned by these modes. The ROM basis is a data-driven basis, which is built by available FOM snapshots. In this paper, we use the proper orthogonal decomposition (POD) [19] to build the ROM basis. We note, however, that other approaches can be used (see, e.g., [17, 42]). The resulting Galerkin ROM (G-ROM) is a system of equations in which the unknowns are the coefficients $(a_i)_{i=1}^{N_u}$ and $(b_i)_{i=1}^{N_p}$ appearing in the reduced fields expressions. For example, for the incompressible Navier-Stokes equations, the G-ROM can be written as follows:

$$\begin{cases} \boldsymbol{a}_t = \boldsymbol{f}(\boldsymbol{a}, \boldsymbol{b}), \\ \mathbf{h}(\boldsymbol{a}, \boldsymbol{b}) = \mathbf{0}, \end{cases} \tag{1}$$

where $\boldsymbol{a}(t) := (a_1(t), \ldots, a_r(t))^T$ and $\boldsymbol{b}(t) := (b_1(t), \ldots, b_q(t))^T$, and $\boldsymbol{f}$ and $\mathbf{h}$ denote the ROM operators.

ROMs are often built by using a few modes. In the *marginally-resolved* modal regime, where the number of ROM basis functions allows a moderately accurate representation of the main features of the underlying dynamics, G-ROMs can yield acceptable approximations. However, in the case of turbulent flow applications, hundreds or even thousands of ROM modes are often required to obtain a good approximation of the flow dynamics. Therefore, in those cases (i.e., in the *under-resolved* regime), low-dimensional G-ROMs generally yield inaccurate approximations.

Different stabilization approaches have been studied in the past years in order to avoid the stability issues and to increase the ROM accuracy [15, 12, 20, 26].

In this paper, in the marginally-resolved and, especially, under-resolved regimes, in order to increase the accuracy of the standard G-ROM (1) while maintaining an acceptable computational cost, the ROM approach is integrated with data-driven techniques. One popular approach is to add *closure/correction* terms to the reduced formulation. To this end, one can use a *purely data-driven* approach, which exploits only the available FOM data to build new terms that include the contribution of the modes neglected in the ROM formulation. The new reduced system with purely data-driven terms is the following:

$$\begin{cases} \boldsymbol{a}_t = \boldsymbol{f}(\boldsymbol{a}, \boldsymbol{b}) + \boldsymbol{\tau_u}(\boldsymbol{a}, \boldsymbol{b}), \\ \mathbf{h}(\boldsymbol{a}, \boldsymbol{b}) + \boldsymbol{\tau}_p(\boldsymbol{a}, \boldsymbol{b}) = \mathbf{0}. \end{cases} \tag{2}$$

The terms $\boldsymbol{\tau}_u$ and $\boldsymbol{\tau}_p$ are evaluated by solving a least square problem that minimizes the difference between the model form of the correction terms and the exact correction terms, which are evaluated by using the available FOM data [23, 57, 34, 36]. The technique used to compute the correction terms takes inspiration from the operator inference approach [41, 29, 8, 27]. In [23], the purely data-driven approach was applied for the first time to the pressure Poisson formulation (PPE-ROM), leading to the introduction of the novel pressure correction term, $\boldsymbol{\tau}_p$. This strategy has significantly increased the accuracy of the pressure field approximation.

To construct the ROM correction terms, a different strategy is the *physically-based data-driven* approach. This strategy consists of two steps: In the first step, one postulates a physical model form for the ROM correction term. For example, to approximate turbulent flows, one can postulate an eddy viscosity model. In the second step of the physically-based data-driven strategy, one solves a least squares problem to determine the optimal parameters in the ROM correction term, e.g., the eddy viscosity coefficient. The physically-based data-driven strategy was used in [18], where the FOM utilizes a Reynolds averaged Navier-Stokes (RANS) approach [44] and an eddy viscosity model is leveraged to close the FOM system [25, 28, 50]. To enforce FOM-ROM model

consistency, a turbulence model was also introduced at a ROM level in [43] and in [18], where an eddy viscosity reduced order model (EV-ROM) is built. In the EV-ROM, the eddy viscosity reduced field is expressed as a linear combination of eddy viscosity modes $\nu_t(x,t) = \sum_{i=1}^{N_{\nu_t}} g_i(t)\eta_i(x)$, where $\{\eta_1, \ldots, \eta_{N_{\nu_t}}\}$ are the eddy viscosity modes. The model introduced in [18] can be expressed as follows:

$$\begin{cases} \boldsymbol{a}_t = \boldsymbol{f}(\boldsymbol{a}, \boldsymbol{b}, \mathbf{g}), \\ \mathbf{h}(\boldsymbol{a}, \boldsymbol{b}, \mathbf{g}) = \mathbf{0}, \end{cases} \tag{3}$$

The vector of eddy viscosity coefficients $\mathbf{g} := (g_1(t), \ldots, g_{N_{\nu_t}})$ can be obtained from the velocity vector of coefficients $\boldsymbol{a}$ by using either interpolation or regression techniques.

The main aim of this paper is to propose a novel *hybrid data-driven* ROM that is constructed by combining the two fundamentally different strategies outlined above: purely data-driven and physically-based data-driven techniques. The new hybrid data-driven ROM can be written as follows:

$$\begin{cases} \boldsymbol{a}_t = \boldsymbol{f}(\boldsymbol{a}, \boldsymbol{b}, \mathbf{g}) + \boldsymbol{\tau_u}(\boldsymbol{a}, \boldsymbol{b}), \\ \mathbf{h}(\boldsymbol{a}, \boldsymbol{b}) + \boldsymbol{\tau}_p(\boldsymbol{a}, \boldsymbol{b}, \mathbf{g}) = \mathbf{0}. \end{cases} \tag{4}$$

Machine learning techniques are exploited to find the eddy viscosity coefficients in (4), training a fully-connected neural network from the full order data.

To our knowledge, this is the first time that a purely data-driven strategy for both the velocity and the pressure, and a physically-based eddy viscosity data-driven strategy are combined. A similar approach has been used in [35]. We note, however, that the approach in [35] is different from the approach used in the present study in two significant aspects: first, a data-driven correction term is used in this paper, but not in [35]; second, the physically-based data-driven correction terms are different: regression based mixing length in [35], and machine learning based eddy viscosity in the current study.

In this paper, the new hybrid data-driven ROM (4) is compared to (i) the purely data-driven ROM (2) proposed in [23], and (ii) the physically-based data-driven ROM (3) proposed in [18]. The three ROMs are tested and compared on the classical case study of a turbulent flow around a cylinder. The numerical results show that the new data-driven ROM (4) yields more accurate velocity and pressure approximations than both the purely data-driven ROM (2) and the physically-based data-driven ROM (3).

These results support the following two conclusions, which are similar to those in [35]: adding a physically-based (eddy viscosity) correction term improves the accuracy of the purely data-driven ROM. Furthermore, adding purely data-driven velocity and pressure correction terms improves the accuracy of the physically-based data-driven ROM. Thus, our numerical investigation suggests that a hybrid data-driven ROM closure strategy is more accurate than both a purely data-driven closure and a physically-based data-driven closure.

The rest of the paper is organized as follows: Section 2 is dedicated to a brief overview of the FOM used and implemented in the C++ open source software *OpenFOAM*. In section 3, the standard G-ROM framework is recalled, specifying the two approaches used in the numerical investigation (the supremizer and pressure Poisson approaches). Sections 4 and 5 summarize the purely data-driven ROMs and physically-based data-driven ROMs, respectively. Section 6 is dedicated to the presentation of the new hybrid data-driven ROM, whereas section 7 presents the results of the numerical simulations in terms of relative errors of the pressure and velocity reduced fields with respect to the full order results. Finally, section 8 presents the conclusions of our study and outlines future research directions.

2. Full Order Model (FOM)

As a mathematical model, we use the Navier-Stokes Equations (NSE) for incompressible flows. The fluid domain is $\Omega \in \mathbb{R}^d$ with $d = 2$ or 3, Γ the domain's boundary, $t \in [0, T]$ the time, $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ the flow velocity vector field, $p = p(\mathbf{x}, t)$ the pressure scalar field normalized by the fluid density, and ν the fluid kinematic viscosity. The strong form of the incompressible NSE is the following:

$$
\begin{cases}
\dfrac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) + \nabla \cdot \nu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) - \nabla p & \text{in } \Omega \times [0, T], & \text{(5a)} \\
\nabla \cdot \mathbf{u} = \mathbf{0} & \text{in } \Omega \times [0, T], & \text{(5b)} \\
+ \text{ boundary conditions} & \text{on } \Gamma \times [0, T], & \text{(5c)} \\
+ \text{ initial conditions} & \text{in } (\Omega, 0). & \text{(5d)}
\end{cases}
$$

The full order solutions of 5 are computed by using the software *OpenFOAM*, which employs a finite volume discretization of the system [37, 24].

The full order problem uses the U-RANS (unsteady RANS equations) approach. This approach is based on the Reynolds decomposition, proposed in [44], which decomposes each flow field in its mean and its fluctuating part. The RANS equations are obtained by taking the time-average of the NSE in (5). In the resulting system, it is important to include a closure model for the Reynolds stress tensor. The closure model considered in this paper is an eddy viscosity model that is based on the Boussinesq hypothesis. Specifically, the SST $\kappa - \omega$ model is used to close the RANS system. This model is based on the inclusion of two transport equations that describe two additional variables: the kinetic energy, κ, and the specific turbulent dissipation rate, ω. The SST $\kappa - \omega$ model is presented in [28] in the standard formulation, and in [32] in the SST formulation. The extended RANS model including the SST $\kappa - \omega$ equations can be found in [18].

3. The POD-Galerkin ROMs

In this section, a brief overview of the standard POD-Galerkin ROM techniques is provided.

After the offline stage is performed, all the FOM snapshots, i.e., the FOM solutions for different time instants $\{t_j\}_{j=1}^{N_T}$, are collected. The POD is then applied on the full order snapshot matrices:

$$
\mathcal{S}_u = \{\mathbf{u}(\mathbf{x}, t_1), ..., \mathbf{u}(\mathbf{x}, t_{N_T})\} \in \mathbb{R}^{N_u^h \times N_T}, \quad \mathcal{S}_p = \{p(\mathbf{x}, t_1), ..., p(\mathbf{x}, t_{N_T})\} \in \mathbb{R}^{N_p^h \times N_T},
$$

where N_u^h and N_p^h are the numbers of degrees of freedom for the velocity and pressure fields, respectively.

The ROM construction uses the following different stabilization approaches for the velocity-pressure coupling [23]:

(1) the SUP-ROM approach, in which additional *supremizer* modes for the velocity space are introduced in order to fulfill the *inf-sup* condition [47, 7, 52, 5];

(2) the PPE-ROM approach, in which the pressure Poisson equation replaces the continuity equation [4, 51, 52, 39].

In the two formulations, the velocity and pressure POD spaces are assembled as follows:

$$
\mathbb{V}_{\text{POD}}^u = \text{span}\{[\boldsymbol{\phi}_i]_{i=1}^{N_u}, [\mathbf{s}(\chi_i)]_{i=1}^{N_{sup}}\}, \quad \mathbb{V}_{\text{POD}}^p = \text{span}\{[\chi_i]_{i=1}^{N_p}\}, \tag{6}
$$

where $N_u \ll N_u^h$ and $N_p \ll N_p^h$, and $[\boldsymbol{\phi}_i]_{i=1}^{N_u}$ and $[\chi_i]_{i=1}^{N_p}$ are the velocity and pressure POD modes, respectively. The supremizer modes $(\mathbf{s}_i)_{i=1}^{N_{sup}} = \mathbf{s}(\chi_i)_{i=1}^{N_{sup}}$ are additional modes introduced in the SUP-ROM approach in order to fulfill the *inf-sup* condition [52].

4

For each pressure basis function, the corresponding supremizer element can be found by solving the following problem:

$$\begin{cases} \Delta \mathbf{s}_i = -\nabla p_i \text{ in } \Omega, \\ \mathbf{s}_i = 0 \text{ on } \partial\Omega. \end{cases} \tag{7}$$

The velocity POD space can be enriched either with an *exact* or with an *approximated* approach [7]. In the exact approach, the problem (7) is solved for each pressure basis function χ_i and each solution is added to the velocity space. In the approximated approach, the problem (7) is solved for each pressure snapshot $p(\mathbf{x}, t_i)_{i=1}^{N_T}$, which yields the following supremizer snapshot matrix:

$$\mathcal{S}_{sup} = \{\mathbf{s}(\mathbf{x}, t_1), ..., \mathbf{s}(\mathbf{x}, t_{N_T})\} \in \mathbb{R}^{N_u^h \times N_T}.$$

A POD modal decomposition is then applied to the snapshot matrix in order to obtain the supremizer POD modes $(\boldsymbol{\eta}_i)_{i=1}^{N_{sup}}$ [52]. In this paper, we adopt the approximated procedure, since it significantly reduces the computational cost of the offline phase. The reduced velocity and pressure fields are expressed as follows:

$$\mathbf{u}(\mathbf{x}, t) \approx \mathbf{u}_r(\mathbf{x}, t) = \sum_{i=1}^{N_u + N_{sup}} a_i(t)\boldsymbol{\phi}_i(\mathbf{x}), \quad p(\mathbf{x}, t) \approx p_r(\mathbf{x}, t) = \sum_{i=1}^{N_p} b_i(t)\chi_i(\mathbf{x}), \tag{8}$$

where $N_{sup} = 0$ for the PPE-ROM approach.

In the remainder of this section, we outline the SUP-ROM (section 3.1) and PPE-ROM (section 3.2) approaches.

3.1. SUP-ROM

In the supremizer approach, performing a Galerkin projection of the momentum equation (5a) onto the velocity modes, and of the continuity equation (5b) onto the pressure modes, the following dynamical system is obtained:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T \mathbf{C}\mathbf{a} - \mathbf{H}\mathbf{b} + \tau\left(\sum_{k=1}^{N_{\text{BC}}}(U_{\text{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right), \\ \mathbf{Pa} = \mathbf{0}, \end{cases} \tag{9}$$

where $\mathbf{a}$ and $\mathbf{b}$ are the vectors of the coefficients associated to the velocity and pressure modes, respectively.

The matrices appearing in the system are defined as follows:

$$(\mathbf{M})_{ij} = (\boldsymbol{\phi}_i, \boldsymbol{\phi}_j)_{L^2(\Omega)}, \quad (\mathbf{P})_{ij} = (\chi_i, \nabla \cdot \boldsymbol{\phi}_j)_{L^2(\Omega)}, \quad (\mathbf{B})_{ij} = (\boldsymbol{\phi}_i, \nabla \cdot \nabla \boldsymbol{\phi}_j)_{L^2(\Omega)},$$
$$(\mathbf{B_T})_{ij} = (\boldsymbol{\phi}_i, \nabla \cdot (\nabla \boldsymbol{\phi}_j)^T)_{L^2(\Omega)}, \quad (\mathbf{C})_{ijk} = (\boldsymbol{\phi}_i, \nabla \cdot (\boldsymbol{\phi}_j \otimes \boldsymbol{\phi}_k))_{L^2(\Omega)}, \quad (\mathbf{H})_{ij} = (\boldsymbol{\phi}_i, \nabla\chi_j)_{L^2(\Omega)}.$$

Remark 3.1 *We highlight that the POD modes, as they are linear combination of snapshots, are piece-wise constant functions. However, it is possible to compute second order operators using the same FV procedure used at the FOM level. We remind to [37] for details.*

The term $\tau\left(\sum_{k=1}^{N_{\text{BC}}}(U_{\text{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right)$ in (9) is a term used to enforce the Dirichlet boundary conditions in the reduced order model [18, 54]. We call Γ_D the Dirichlet boundary, which is composed by different parts, $\Gamma_{D_1}, \Gamma_{D_2}, .., \Gamma_{D_K}$.

In particular, N_{BC} is the number of velocity boundary conditions we would like to impose on the parts of the Dirichlet boundary where velocity has at least one non-zero component; $U_{\text{BC},k}$ is the velocity non-zero component at the k-th part of the Dirichlet boundary.

5

τ is a penalization factor which is tuned by a sensitivity analysis on the specific problem considered [18, 54]. In general, bigger values of the penalization factor lead to a stronger enforcement of the boundary conditions.

The matrices $\mathbf{E}^k$ and vectors $\mathbf{D}^k$ are defined as follows:

$$(\mathbf{E}^k)_{ij} = (\phi_i, \phi_j)_{L^2(\Gamma_{D_k})}, \quad (\mathbf{D}^k)_i = (\phi_i)_{L^2(\Gamma_{D_k})}, \text{ for all } k = 1, ..., N_{\text{BC}}.$$

3.2. PPE-ROM

In the pressure Poisson approach, the continuity equation (9) is replaced by the Poisson equation for pressure, obtained by taking the divergence of the momentum equation. The dynamical system is obtained by projecting the momentum equation and the pressure Poisson equation on the velocity and pressure modes, respectively:

$$\begin{cases} \mathbf{M\dot{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T\mathbf{Ca} - \mathbf{Hb} + \tau\left(\sum_{k=1}^{N_{\text{BC}}}(U_{BC,k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right), \\ \mathbf{Db} + \mathbf{a}^T\mathbf{Ga} - \nu\mathbf{Na} - \mathbf{L} = \mathbf{0}. \end{cases} \tag{10}$$

The matrices $\mathbf{M}$, $\mathbf{B}$, $\mathbf{B_T}$, $\mathbf{C}$, $\mathbf{H}$, $\mathbf{D}^k$, and $\mathbf{E}^k$ in system (10) are the same as the corresponding matrices in (9). The additional matrices are defined as follows:

$$(\mathbf{D})_{ij} = (\nabla\chi_i, \nabla\chi_j)_{L^2(\Omega)}, \quad (\mathbf{G})_{ijk} = (\nabla\chi_i, \nabla \cdot (\phi_j \otimes \phi_k))_{L^2(\Omega)},$$
$$(\mathbf{N})_{ij} = (\mathbf{n} \times \nabla\chi_i, \nabla\phi_j)_{L^2(\Gamma)}, \quad (\mathbf{L})_{ij} = (\chi_i, \mathbf{n} \cdot \boldsymbol{R}_t)_{L^2(\Gamma)},$$

where vector $\mathbf{n}$ is the normal unitary vector to the domain boundary Γ.

4. DATA-DRIVEN VMS-ROMS

We consider the *data-driven variational multiscale ROM* (*DD-VMS-ROM*) framework [23], in which different correction/closure terms are introduced in the formulations of SUP-ROM(9) and PPE-ROM (10). The DD-VMS-SUP-ROM approach (section 4.1) includes only velocity correction terms, whereas the DD-VMS-PPE-ROM approach (section 4.2) includes both pressure and velocity correction terms.

For the sake of simplicity, in this section we will consider the following notation for the coefficient vectors for velocity and pressure:

$$\mathbf{a} = (a_i)_{i=1}^r, \quad \mathbf{b} = (b_i)_{i=1}^q,$$

where r is the reduced number of modes for velocity ($r = N_u$ in the PPE-ROM approach, $r = N_u + N_{sup}$ in the SUP-ROM approach), q is the reduced number of modes for pressure ($q = N_p$). We will also consider $r_{tot} = r + q$.

4.1. DD-VMS-SUP-ROM

The reduced formulation for the supremizer approach adopted in [23] is the following:

$$\begin{cases} \mathbf{M\dot{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T\mathbf{Ca} - \mathbf{Hb} + \tau\left(\sum_{k=1}^{N_{BC}}(U_{BC,k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right) + \boldsymbol{\tau}^u(\boldsymbol{a}), \\ \mathbf{Pa} = \mathbf{0}. \end{cases} \tag{11}$$

The velocity correction term appearing in (11) is modeled as follows:

$$\boldsymbol{\tau}^u(\mathbf{a}) = \tilde{A}\mathbf{a} + \mathbf{a}^T\tilde{B}\mathbf{a}, \tag{12}$$

where $\tilde{A}$ is a matrix and $\tilde{B}$ is a three-dimensional tensor. The operators $\tilde{A}$ and $\tilde{B}$ are constructed by solving an optimization problem that minimizes the difference between the exact correction term, $\boldsymbol{\tau}_u^{\text{exact}}$, and a correction term ansatz, $\boldsymbol{\tau}_u^{\text{ansatz}}$:

$$\min_{\substack{\tilde{A} \in \mathbb{R}^{r \times r}, \\ \tilde{B} \in \mathbb{R}^{r \times r \times r}}} \sum_{j=1}^{M} ||\boldsymbol{\tau}_u^{\text{exact}}(t_j) - \boldsymbol{\tau}_u^{\text{ansatz}}(t_j)||_{L^2(\Omega)}^2 , \tag{13}$$

where M time instances are considered to build the correction term and the term $\boldsymbol{\tau}^{\text{exact}}(t_j)$ is computed starting from the snapshots coefficient vectors $\mathbf{a}_d^{snap}(t_j)$ and $\mathbf{a}_r^{snap}(t_j)$, defined as follows:

$$a_{d_i}^{snap}(t_j) = (\mathbf{u}_d(t_j), \boldsymbol{\phi}_i)_{L^2(\Omega)} \quad \forall i = 1, ..., d \gg r.$$
$$a_{r_i}^{snap}(t_j) = (\mathbf{u}_r(t_j), \boldsymbol{\phi}_i)_{L^2(\Omega)} \quad \forall i = 1, ..., r.$$

The value d is the number of modes used to build the exact correction term.

Remark 4.1 *We remark that the value of d can be chosen as the rank of the snapshots matrix. However, in order to decrease the computational cost of the offline stage, we chose a value which is smaller than the rank of the snapshots matrix, but considerably bigger than the reduced number of modes considered, as pointed out in [57].*

The exact correction term for velocity is evaluated as follows:

$$\boldsymbol{\tau}_u^{\text{exact}}(t_j) = \left(-\overline{(\mathbf{a}_d^{snap}(t_j))^T \mathbf{C_d} \mathbf{a}_d^{snap}(t_j)}^r \right) - \left(-(\mathbf{a}_r^{snap}(t_j))^T \mathbf{C} \mathbf{a}_r^{snap}(t_j) \right) ,$$

where the tensor $\mathbf{C_d} \in \mathbb{R}^{d \times d \times d}$ is defined in the following way:

$$\mathbf{C_d}_{ijk} = \left(\boldsymbol{\phi}_i, \nabla \cdot (\boldsymbol{\phi}_j \otimes \boldsymbol{\phi}_k) \right)_{L^2(\Omega)} .$$

The correction term ansatz is evaluated as in (12), but starting from $\mathbf{a}_r^{snap}(t_j)$ at each time step j:

$$\boldsymbol{\tau}_u^{\text{ansatz}}(t_j) = \tilde{A} \mathbf{a}_r^{snap}(t_j) + (\mathbf{a}_r^{snap}(t_j))^T \tilde{B} \mathbf{a}_r^{snap}(t_j). \tag{14}$$

The optimization problem (13) is rewritten as a least squares problem following a procedure similar to that used in [41] and presented in detail in [23] and [22].

A different way to find $\tilde{A}$ and $\tilde{B}$, introduced in [34], is that of solving a constrained least squares problem, inheriting the physical properties of the exact tensors:

$$\min_{\substack{\tilde{A} \in \mathbb{R}^{r \times r}, \\ \tilde{B} \in \mathbb{R}^{r \times r \times r}, \\ \mathbf{a}^T \tilde{A} \mathbf{a} \leqslant 0, \\ \mathbf{a}^T (\mathbf{a}^T \tilde{B} \mathbf{a}) = 0}} \sum_{j=1}^{M} ||\boldsymbol{\tau}_u^{\text{exact}}(t_j) - \boldsymbol{\tau}_u^{\text{ansatz}}(t_j)||_{L^2(\Omega)}^2 . \tag{15}$$

As shown in [23, 34], the constrained method can yield better results than the unconstrained method in the marginally-resolved regime.

4.2. DD-VMS-PPE-ROM

In this section, the pressure data-driven model developed in [23] is briefly recalled. The continuity equation at the reduced level is replaced by the pressure Poisson equation. This formulation allows for the introduction of novel pressure correction terms in the reduced system:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T \mathbf{C} \mathbf{a} - \mathbf{H}\mathbf{b} + \tau \left(\sum_{k=1}^{N_{BC}} (U_{BC,k} \mathbf{D}^k - \mathbf{E}^k \mathbf{a}) \right) + \boldsymbol{\tau}^u(\boldsymbol{a}, \boldsymbol{b}) , \\ \mathbf{D}\mathbf{b} + \mathbf{a}^T \mathbf{G} \mathbf{a} - \nu \mathbf{N} \mathbf{a} - \mathbf{L} + \boldsymbol{\tau}^p(\boldsymbol{a}, \boldsymbol{b}) = \mathbf{0} . \end{cases} \tag{16}$$

7

In [23, 22], different ansatzes are analyzed for the correction terms $\boldsymbol{\tau}^u$ and $\boldsymbol{\tau}^p$. The formulation which is chosen in this paper is the one in which a unique least squares problem is solved to find all the purely data-driven terms.

As the exact correction term in the optimization problem, we consider the following:

$$\boldsymbol{\tau}_{\text{tot}}^{\text{exact}}(t_j) = \left(\boldsymbol{\tau}_u^{\text{exact}}(t_j), \boldsymbol{\tau}_p^{\text{exact}}(t_j) \right) \; \forall j = 1, ..., M \,. \tag{17}$$

The exact correction corresponding to $\boldsymbol{\tau}_p$ includes the contribution of two different terms:

$$\boldsymbol{\tau}_p^{\text{exact}}(t_j) = \boldsymbol{\tau}_D^{\text{exact}}(t_j) + \boldsymbol{\tau}_G^{\text{exact}}(t_j) =$$
$$= \left(\overline{\mathbf{D_d} \mathbf{b}_d^{\text{snap}}(t_j)^q} \right) - \mathbf{D} \mathbf{b}_q^{\text{snap}}(t_j) + \overline{(\boldsymbol{a}_d^{\text{snap}}(t_j))^T \mathbf{G_d} \boldsymbol{a}_d^{\text{snap}}(t_j)}^r - (\boldsymbol{a}_r^{\text{snap}}(t_j))^T \mathbf{G} \boldsymbol{a}_r^{\text{snap}}(t_j) \,,$$

where the matrix $\mathbf{D_d}$ and tensor $\mathbf{G_d}$ are defined as follows:

$$\mathbf{D_d}_{ij} = (\nabla \chi_i, \nabla \chi_j)_{L^2(\Omega)}, \quad \mathbf{G_d}_{ijk} = (\nabla \chi_i, \nabla \cdot (\boldsymbol{\phi}_j \otimes \boldsymbol{\phi}_k))_{L^2(\Omega)}, \; i, j, k = 1, \dots, d \,.$$

The ansatz considered in this paper is the following:

$$\boldsymbol{\tau}_{\text{tot}}^{\text{ansatz}}(t_j) = \tilde{J}_A \mathbf{ab}^{\text{snap}}(t_j) + (\mathbf{ab}^{\text{snap}}(t_j))^T \tilde{J}_B \mathbf{ab}^{\text{snap}}(t_j) \,, \tag{18}$$

where the matrices $\tilde{J}_A \in \mathbb{R}^{(r+q) \times (r+q)}$ and $\tilde{J}_B \in \mathbb{R}^{(r+q) \times (r+q) \times (r+q)}$, and the vector $\mathbf{ab}^{\text{snap}}(t_j) = (\boldsymbol{a}_r^{\text{snap}}(t_j), \boldsymbol{b}_q^{\text{snap}}(t_j)) \in \mathbb{R}^{r+q}$. The final correction which is inserted in the reduced system (16) is divided into two vectors:

$$\tilde{J}_A \mathbf{ab} + \mathbf{ab}^T \tilde{J}_B \mathbf{ab} = (\boldsymbol{\tau}_u, \boldsymbol{\tau}_p) \,, \; \text{where} \; \boldsymbol{\tau}_u \in \mathbb{R}^{N_u}, \boldsymbol{\tau}_p \in \mathbb{R}^{N_p} \,. \tag{19}$$

Finally, the following optimization problem is solved:

$$\min_{\substack{\tilde{J}_A \in \mathbb{R}^{r_{\text{tot}} \times r_{\text{tot}}}; \\ \tilde{J}_B \in \mathbb{R}^{r_{\text{tot}} \times r_{\text{tot}} \times r_{\text{tot}}}}} \sum_{j=1}^{M} ||\boldsymbol{\tau}_{\text{tot}}^{\text{exact}}(t_j) - \boldsymbol{\tau}_{\text{tot}}^{\text{ansatz}}(t_j)||_{L^2(\Omega)}^2 \,. \tag{20}$$

5. Eddy Viscosity ROMs

The ROMs described in section 4 do not include turbulence treatment inside the formulation. Different models have been used in fluid applications to simulate the turbulent behaviour. In this paper, turbulence is included at the full order level by using a particular type of *eddy viscosity* model, the SST $\kappa - \omega$ model, which adds to the RANS equations the transport equations for κ and ω.

At the reduced order level, an approximation of the eddy viscosity terms can be included in equations by introducing a reduced order version of the eddy viscosity [18], as follows:

$$\nu_t(\mathbf{x}, t) \approx \nu_{t_r}(\mathbf{x}, t) = \sum_{i=1}^{N_{\nu_t}} g_i(t) \eta_i(\mathbf{x}) \,,$$

where $\eta_i(\mathbf{x})$ is the i-th eddy viscosity mode evaluated through a POD procedure and $g_i(t)$ the corresponding coefficient.

Adding the turbulence terms to the SUP-ROM (9), the new dynamical system can be written as follows:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T \mathbf{C}\mathbf{a} + \mathbf{g}^T(\mathbf{C}_{\text{T1}} + \mathbf{C}_{\text{T2}})\mathbf{a} - \mathbf{H}\mathbf{b} + \tau \left(\sum_{k=1}^{N_{\text{BC}}} (U_{\text{BC},k} \mathbf{D}^k - \mathbf{E}^k \mathbf{a}) \right) \,, \\ \mathbf{P}\mathbf{a} = \mathbf{0} \,, \end{cases} \tag{21}$$

where the new matrices appearing are defined as:

$$(\mathbf{C}_{\mathrm{T1}})_{ijk} = (\boldsymbol{\phi}_i, \eta_j \nabla \cdot \nabla \boldsymbol{\phi}_k)_{L^2(\Omega)},$$

$$(\mathbf{C}_{\mathrm{T2}})_{ijk} = (\boldsymbol{\phi}_i, \nabla \cdot \eta_j (\nabla \boldsymbol{\phi}_k)^T)_{L^2(\Omega)}.$$

When a PPE approach is considered, the FOM momentum and Poisson equation, according to the RANS turbulent model, are written in the following way:

$$\begin{cases} \dfrac{\partial \overline{\mathbf{u}}}{\partial t} + \nabla \cdot (\overline{\mathbf{u}} \otimes \overline{\mathbf{u}}) = \nabla \cdot \left[-\overline{p}\mathbf{I} + (\nu + \nu_t)\left(\nabla\overline{\mathbf{u}} + (\nabla\overline{\mathbf{u}})^T\right)\right] & \text{in } \Omega \times [0, T], \\ \Delta \overline{p} = -\nabla \cdot (\nabla \cdot (\overline{\mathbf{u}} \otimes \overline{\mathbf{u}})) + \nabla \cdot \left[\nabla \cdot \left(\nu_t\left(\nabla\overline{\mathbf{u}} + (\nabla\overline{\mathbf{u}})^T\right)\right)\right] & \text{in } \Omega, \\ + \text{ Boundary Conditions} & \text{on } \Gamma \times [0, T], \\ + \text{ Initial Conditions} & \text{in } (\Omega, 0). \end{cases}$$

Consequently, the dynamical system (10) takes the following form:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T\mathbf{C}\mathbf{a} + \mathbf{g}^T(\mathbf{C}_{\mathrm{T1}} + \mathbf{C}_{\mathrm{T2}})\mathbf{a} - \mathbf{Hb} + \tau\left(\sum_{k=1}^{N_{\mathrm{BC}}}(U_{\mathrm{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right), \\ \mathbf{Db} + \mathbf{a}^T\mathbf{Ga} - \mathbf{g}^T(\mathbf{C}_{\mathrm{T3}} + \mathbf{C}_{\mathrm{T4}})\mathbf{a} - \nu\mathbf{Na} - \mathbf{L} = \mathbf{0}, \end{cases} \tag{22}$$

where:

$$(\mathbf{C}_{\mathrm{T3}})_{ijk} = (\nabla \chi_i, \eta_j \nabla \cdot \nabla \boldsymbol{\phi}_k)_{L^2(\Omega)}, \quad (\mathbf{C}_{\mathrm{T4}})_{ijk} = (\nabla \chi_i, \nabla \cdot \eta_j (\nabla \boldsymbol{\phi}_k)^T)_{L^2(\Omega)}.$$

In the dynamical systems defined in (21) and (22), the number of unknowns is $N_u + N_{sup}$ (in (21)) and N_u (in (22)) for velocity, N_p for pressure, and N_{ν_t} for the eddy viscosity. However, the number of equations is $N_u + N_{sup} + N_p$ in (21) and $N_u + N_p$ in (22). Thus, there are more unknowns than equations and the systems are not closed for both the supremizer and the Poisson approach. In order to close the systems, the eddy viscosity coefficients $[g_i(t)]_{i=1}^{N_{\nu_t}}$ can be computed considering the mapping $\mathbf{g} = f(\mathbf{a})$ through either *interpolation* or *regression* techniques.

In the first approach, the reduced eddy viscosity coefficients are interpolated with radial basis functions [31, 33]. This technique was exploited in [18], following the POD-I approach [56, 55, 49]. In the regression approach, which is the one used in this paper, the reduced eddy viscosity coefficients are computed starting from the velocity coefficients $[a_i]_{i=1}^{N_u}$ through a feed-forward fully-connected neural network [22]. The neural network considered has two hidden layers of sizes 256 and 64, the activation function is *ReLU*, and the learning rate used for training is $1e-5$. The loss function minimized in the training process is the difference between the output of the neural network ($\mathbf{g}_{\mathrm{NN}} = \mathbf{f}_{\mathrm{NN}}(\mathbf{a})$) and the values of the known coefficients of the eddy viscosity field ($\mathbf{g} = (g_i)_{i=1}^{N_{\nu_t}}$), which are found from the POD procedure.

6. HYBRID DATA-DRIVEN ROMs

This section introduces the hybrid data-driven approach. The new formulation fuses the purely data-driven and physically-based data-driven strategies presented in sections 4 and 5, respectively. The reduced system following the supremizer approach is expressed as follows:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T\mathbf{C}\mathbf{a} + \mathbf{g}^T(\mathbf{C}_{\mathrm{T1}} + \mathbf{C}_{\mathrm{T2}})\mathbf{a} - \mathbf{Hb} + \tau\left(\sum_{k=1}^{N_{\mathrm{BC}}}(U_{\mathrm{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right) + \boldsymbol{\tau}_u(\boldsymbol{a}), \\ \mathbf{Pa} = \mathbf{0}, \end{cases}$$
$$\tag{23}$$

where:

$$\boldsymbol{\tau}_u(\boldsymbol{a}) = \tilde{A}\boldsymbol{a} + \boldsymbol{a}^T\tilde{B}\boldsymbol{a}. \tag{24}$$

9

Considering the pressure Poisson approach, the reduced system is written in the following way:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}} = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a} - \mathbf{a}^T\mathbf{Ca} + \mathbf{g}^T(\mathbf{C}_{T1} + \mathbf{C}_{T2})\mathbf{a} - \mathbf{Hb} + \tau\left(\sum_{k=1}^{N_{BC}}(U_{BC,k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a})\right) + \boldsymbol{\tau}_u(\boldsymbol{a}, \boldsymbol{b})\,, \\ \mathbf{Db} + \mathbf{a}^T\mathbf{Ga} - \mathbf{g}^T(\mathbf{C}_{T3} + \mathbf{C}_{T4})\mathbf{a} - \nu\mathbf{Na} - \mathbf{L} + \boldsymbol{\tau}_p(\boldsymbol{a}, \boldsymbol{b}) = \mathbf{0}\,, \end{cases}$$

$$(25)$$

where, as described in section 4.2, the correction terms are:

$$\boldsymbol{\tau}_{\text{tot}} = (\boldsymbol{\tau}_u(\boldsymbol{a}, \boldsymbol{b}), \boldsymbol{\tau}_p(\boldsymbol{a}, \boldsymbol{b})) = \tilde{J}_A\mathbf{ab} + \mathbf{ab}^T\tilde{J}_B\mathbf{ab}\,. \tag{26}$$

Remark 6.1 *We highlight that there is an interplay between the two data-driven techniques used in the hybrid model:*

- *the coefficient vector* $\mathbf{a}(t)$ *is computed at time* t *by solving a system in which both purely and physically data-driven closure terms appear. For example, in the hybrid data-driven PPE-ROM model:*

$$\begin{cases} \dot{\mathbf{a}}(t) = \mathbf{f}(\mathbf{a}(t), \mathbf{b}(t), \mathbf{g}(t)) + \boldsymbol{\tau_u}(\mathbf{a}(t), \mathbf{b}(t)) \\ \mathbf{h}(\mathbf{a}(t), \mathbf{b}(t), \mathbf{g}(t)) + \boldsymbol{\tau_p}(\mathbf{a}(t), \mathbf{b}(t)) = 0. \end{cases} \tag{27}$$

  The computed coefficient vector $\mathbf{a}$ *is then given as input to the neural network and used to predict the eddy viscosity reduced field* $(\mathbf{g}(t + 1) = f(\mathbf{a}(t)))$.

- *the computed* $\mathbf{g}(t + 1)$ *is then used to solve* (27) *at time* $t + 1$ *and find* $\mathbf{a}(t + 1)$ *and* $\mathbf{b}(t + 1)$, *and so on.*

7. NUMERICAL RESULTS

The case study considered to test the data-driven ROMs described in sections 4, 5 and 6 is an unsteady flow past a circular cylinder. The case is studied in two dimensions and the mesh used has 11644 cells. The mesh and the boundary conditions set for velocity and pressure are displayed in Figure 1 [18]. The diameter of the cylinder is $D = 1\,\mathrm{m}$, the fluid kinematic viscosity $\nu = 1 \times 10^{-4}\,\mathrm{m^2\,s^{-1}}$, and the velocity at the inlet is horizontal and fixed: $U_{in} = 5\,\mathrm{m\,s^{-1}}$.
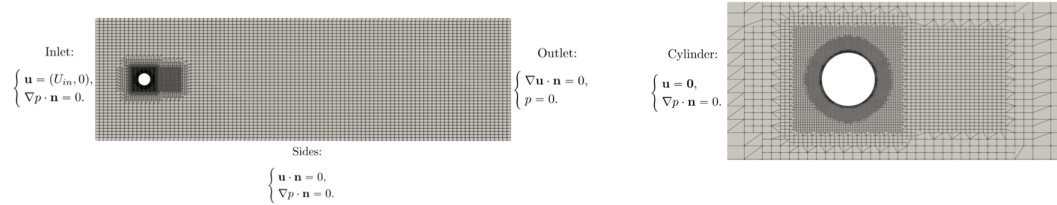

Figure 1: *The mesh used in simulations (left) and the mesh zoomed around the cylinder (right), with the corresponding boundary conditions.*

For the offline stage, the software *OpenFOAM* is used to obtain the full order fields, using the unsteady solver pimpleFoam and the $\kappa - \omega$ model for the turbulence treatment. The pimpleFoam solver is based on a PIMPLE approach for pressure velocity coupling with under-relaxation technique, which consists of the coupling of a SIMPLE [40] and a PISO strategy [21]. The number of time instances considered in the offline phase is 5000, considering one FOM snapshot every 0,004 s

and starting from the second 20 from the beginning of the simulation. Since the time step of the full order simulation is set to 0,0002 s, we are using *undersampling*, taking one snapshots every 20 time steps.

Then, the POD is performed. The POD modes for the velocity, pressure, and supremizer fields are extracted from the numerical data, making use of the library *ITHACA-FV* [1, 52, 51].

The simulations for the online ROM procedure are computed in a dedicated *Python* script, and both the SUP-ROM and the PPE-ROM approaches are tested.

The results of the high-fidelity simulations are compared to the results obtained by solving the reduced order dynamical systems with or without the data-driven terms and considering the parameter $\tau = 1000$, which was introduced in system (9). More specifically, the results are evaluated in terms of the percentage errors of the reduced pressure and velocity fields with respect to the correspondent high-fidelity fields, evaluated at each time step, in the L^2 norm. Since the projection of the full order solution on the reduced POD space is the best possible result which can be achieved with a given number of modes, the solution of the reduced system cannot improve with respect to that projection. For this reason, the percentage errors with respect to the projection of the full order solution on the reduced POD space are used in the numerical investigation. The percentage errors with respect to the full order fields at each j-th time step are evaluated in the following way:

$$\varepsilon_u(t_j) = \frac{||\mathbf{u}_r^{abs}(\mathbf{x}, t_j) - \mathbf{u}_d^{abs}(\mathbf{x}, t_j)||_{L^2(\Omega)}}{||\mathbf{u}_d^{abs}(\mathbf{x}, t_j)||_{L^2(\Omega)}}, \quad \varepsilon_p(t_j) = \frac{||p_r(\mathbf{x}, t_j) - p_d(\mathbf{x}, t_j)||_{L^2(\Omega)}}{||p_d(\mathbf{x}, t_j)||_{L^2(\Omega)}}, \quad (28)$$

where the following quantities are considered:

- the reduced velocity and pressure fields:

$$\mathbf{u}_r(\mathbf{x}, t_j) = \sum_{i=1}^{r} a_i(t_j)\boldsymbol{\phi}_i(\mathbf{x}), \quad p_r(\mathbf{x}, t_j) = \sum_{i=1}^{q} b_i(t_j)\chi_i(\mathbf{x}),$$

where the coefficients $a_i(t_j)$ and $b_i(t_j)$ are the solutions of the dynamical systems (29) (in the supremizer approach) and (30) (in the Poisson approach);

- the approximated full order fields of velocity and pressure, which are evaluated starting from the first d modes, where $d = 100$ when a supremizer approach is considered, and $d = 50$ when a pressure Poisson approach is considered. Such a term reads

$$\mathbf{u}_d(\mathbf{x}, t_j) = \sum_{i=1}^{d} a_i^{\text{snap}}(t_j)\boldsymbol{\phi}_i(\mathbf{x}), \quad p_d(\mathbf{x}, t_j) = \sum_{i=1}^{d_p} b_i^{\text{snap}}(t_j)\chi_i(\mathbf{x}).$$

We remark that when the supremizer approach is considered, $\{\boldsymbol{\phi}_i\}_{i=51}^{100} = \{\mathbf{s}_i(\chi_i)\}_{i=1}^{50}$ are the supremizer modes.

Considering the supremizer approach, the most general form for the dynamical system solved at each time step is the following:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}}^i = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a}^i - (\mathbf{a}^i)^T \mathbf{C}\mathbf{a}^i - \mathbf{H}\mathbf{b}^i + c_u \boldsymbol{\tau}_u(\mathbf{a}^i, \mathbf{b}^i) & + \\ +c_t \left((\mathbf{g}^i)^T (\mathbf{C}_{T1} + \mathbf{C}_{T2})\mathbf{a}^i\right) + \tau \left(\sum_{k=1}^{N_{\text{BC}}} (U_{\text{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a}^i)\right) & \text{at each } i = 1, ..., M, \\ \mathbf{P}\mathbf{a}^i = \mathbf{0} & \text{at each } i = 1, ..., M, \end{cases} \quad (29)$$

where M is the total number of time steps in the online phase.

Considering the pressure Poisson approach, the reduced dynamical system can be written as follows:

$$\begin{cases} \mathbf{M}\dot{\mathbf{a}}^i = \nu(\mathbf{B} + \mathbf{B_T})\mathbf{a}^i - (\mathbf{a}^i)^T\mathbf{C}\mathbf{a}^i - \mathbf{H}\mathbf{b}^i + c_u\boldsymbol{\tau}^u(\mathbf{a}^i, \mathbf{b}^i) & + \\ +c_t\left((\mathbf{g}^i)^T(\mathbf{C}_{T1} + \mathbf{C}_{T2})\mathbf{a}^i\right) + \tau\left(\sum_{k=1}^{N_{\text{BC}}}(U_{\text{BC},k}\mathbf{D}^k - \mathbf{E}^k\mathbf{a}^i)\right) & \text{at each } i = 1, ..., M, \\ \mathbf{D}\mathbf{b}^i + (\mathbf{a}^i)^T\mathbf{G}\mathbf{a}^i - \nu\mathbf{N}\mathbf{a}^i - \mathbf{L} + c_t\left((\mathbf{g}^i)^T(\mathbf{C}_{T3} + \mathbf{C}_{T4})\mathbf{a}^i\right) & + \\ +c_p\boldsymbol{\tau}^p(\mathbf{a}^i, \mathbf{b}^i) = \mathbf{0} & \text{at each } i = 1, ..., M. \end{cases} \tag{30}$$

The matrices and tensors appearing in the two formulations are specified in sections 3, 4, and 5. In systems (29) and (30), the parameters c_u, c_p, and c_t are introduced to either include or combine the different data-driven strategies in the reduced formulation.

For the time integration, two different approaches are used. Specifically, the time derivative $\dot{\mathbf{a}}^i$ appearing in the momentum equation in (29) and (30) is computed following (i) an Euler first order time scheme, and (ii) an implicit second order time scheme. It is worth remarking that the second order time scheme corresponds to the scheme implemented in *OpenFOAM* and used to solve the full order problem.

In this section, the numerical results are analyzed with respect to three different criteria:

- in the first part (section 7.1), the accuracy of the data-driven ROMs is analyzed by varying the number of ROM modes between 1 and 10;

- the second and the third sections (7.2 and 7.3) investigate the accuracy of the data-driven ROMs by first fixing the number of modes to 5 for the velocity, pressure, and supremizer fields, and then varying the time. Sections 7.2 and 7.3 investigate the supremizer and the Poisson approaches, respectively;

- the last section (7.4) is dedicated to the graphical comparison of the velocity and pressure fields produced by the data-driven ROMs.

The following cases are analyzed and compared, for both the first and the second order time schemes, and for both SUP-ROM and PPE-ROM approaches:

- $c_t = c_u = 0$ (and $c_p = 0$ for PPE-ROM), i.e., standard ROM without the addition of any data-driven term;

- $c_t = 0$, $c_u = 1$ (and $c_p = 1$ in PPE-ROM), i.e., purely data-driven model;

- $c_t = 1$, $c_u = 0$ (and $c_p = 0$ in PPE-ROM), i.e., physically-based data-driven model;

- $c_t = c_u = 1$ (and $c_p = 1$ in PPE-ROM), i.e., hybrid data-driven model.

In sections 7.2, 7.3, and 7.4, the results provided by the combination of the two different data-driven techniques are compared to those obtained in the previous works [23] and [18], where only one of these two techniques is used.

7.1. Hybrid data-driven approach in different modal regimes

In this section, a comparison of different modal regimes is carried out for both the supremizer and the pressure Poisson approaches. The models considered for the investigation are:

- the SUP-ROM, with and without the closure turbulence model and the extra velocity correction term;

- the PPE-ROM, with and without the turbulence model, extra velocity correction term (in the momentum equation), and pressure correction term (in the pressure Poisson equation).

The accuracy is evaluated in terms of percentage integral errors for the absolute value of velocity and pressure on a time window of 500 time instants. The expressions for the overall time window L^2 errors are the following:

$$\frac{\int_0^T ||\mathbf{u}_r^{abs}(\mathbf{x},t_j) - \mathbf{u}_d^{abs}(\mathbf{x},t_j)||_{L^2(\Omega)}\, dt}{\int_0^T ||\mathbf{u}_d^{abs}(\mathbf{x},t_j)||_{L^2(\Omega)}\, dt} \times 100\,, \quad \frac{\int_0^T ||p_r(\mathbf{x},t_j) - p_d(\mathbf{x},t_j)||_{L^2(\Omega)}\, dt}{\int_0^T ||p_d(\mathbf{x},t_j)||_{L^2(\Omega)}\, dt} \times 100$$

where T corresponds to $2\,$s. The errors defined above are analyzed for all the cases for both the first and the second order time schemes.

The graphical results are showed in semi-logarithmic plots in Figure 2 for the supremizer approach, and in Figure 3 for the Poisson approach.

In Figure 2, the number of modes considered in the supremizer approach satisfies $N_{sup} > N_p$ when $N_u = N_p$ is equal to 8, 9, and 10, in order to avoid stability issues. From Figures 2 and 3, one can note that the data-driven corrections have a more significant positive effect when a small number of modes is considered; the improvement with respect to the standard Galerkin-ROM, especially for the pressure field, is not visible when the number of modes is larger than 8.

In both the supremizer and the Poisson approaches, the time integration scheme plays an important role in the evaluation of the data-driven ROM accuracy. In fact, when a first order time scheme is used, the introduction of the turbulence treatment in the formulation does not improve the results with respect to the standard ROM and the results obtained in [23] with the DD-VMS-ROM. This fact is particularly evident for a number of modes between 7 and 10 (Figures 2(a),(b) and 3(a),(b)).

When the standard ROM is considered, the results obtained with a first-order time scheme outperform the results obtained with a second-order time scheme: this is likely due to the reduced numerical dissipation associated with the second-order time scheme, which makes the system more exposed to numerical instabilities.

When the turbulence modeling is added to the reduced formulation, results improve only if we consider a second-order time scheme, i.e., if we have consistency with respect to the full order model.

The analysis in the following section focuses on $N_u = N_p = 5$ (and $N_{sup} = 5$ for the supremizer enrichment), i.e., in the marginally-resolved regime.

7.2. Comparison of data-driven VMS-SUP-ROMs

In this section, the results of different data-driven techniques are compared for the supremizer approach for a first and a second order time schemes.

For the simulations of the reduced systems (21) and (23), the coefficients of the reduced eddy viscosity field $(g_i)_{i=1}^{N_{\nu_t}}$ are computed by using a fully-connected neural network, starting from the velocity coefficients $(a_i)_{i=1}^{N_u}$. The network is composed of two hidden layers, the ReLU function is used as an activation function in the network, and the learning rate is set to 10^{-5}.

The momentum equation correction term is obtained by solving the constrained optimization problem (15), since it provides the best performance with respect to the velocity accuracy for a low number of modes, as pointed out in section 4.1.

Figures 4 (a) and (b) display the results obtained using a first order time scheme. In this case, the inclusion of a turbulence model does not appear to have a completely positive effect on accuracy. In addition, coupling the turbulence and correction strategies does not lead to higher
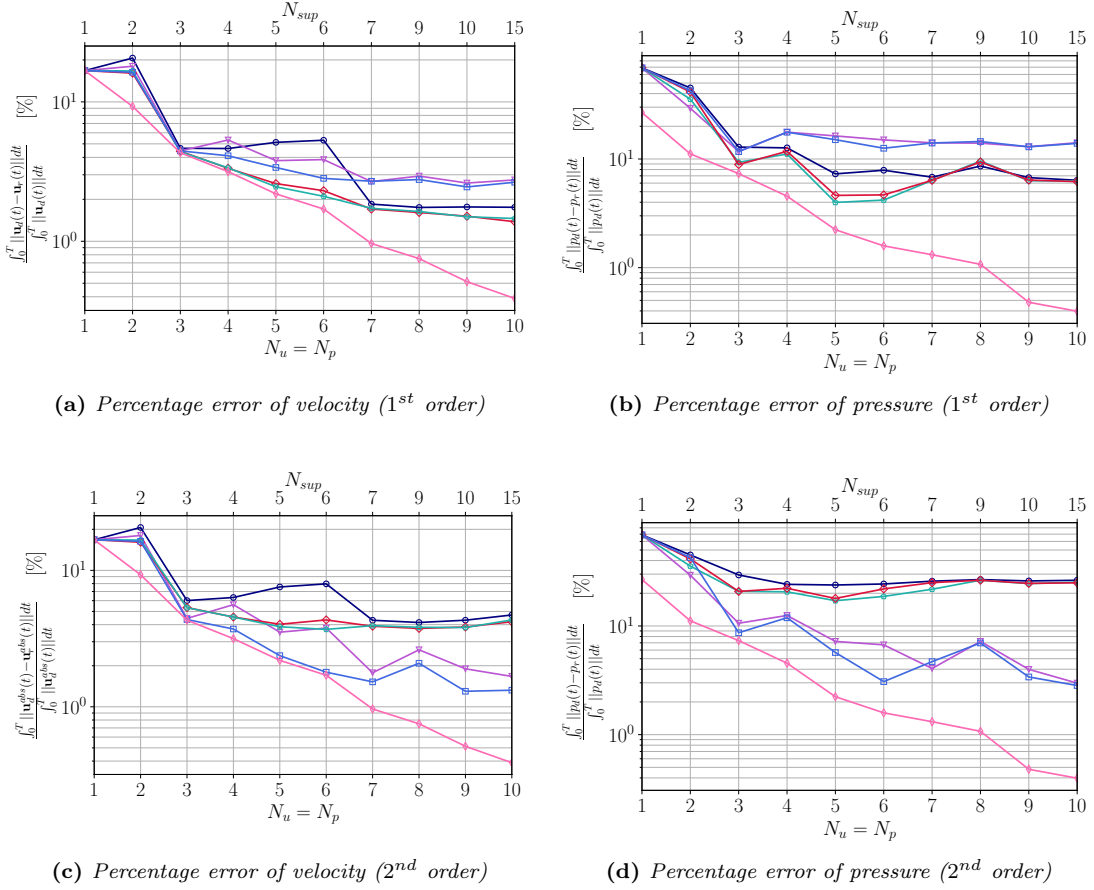
13

(a) *Percentage error of velocity (1st order)*

(b) *Percentage error of pressure (1st order)*

(c) *Percentage error of velocity (2nd order)*

(d) *Percentage error of pressure (2nd order)*

Figure 2: *Percentage integral errors of the absolute value of velocity and pressure, varying the number of modes. The model is the SUP-ROM with a first ((a), (b)) and second ((c), (d)) order time scheme. The cases represented are the following: without any data-driven term (──◦──); physically-based data-driven model (──▽──); purely data-driven model for the velocity (──◇── and ──◦── unconstrained and constrained, respectively); hybrid model (──□──); projection (──◇──).*

accuracy, especially when the pressure field is considered. In such a case in fact, the hybrid method leads to worse results than in the no-correction case, as already pointed out in section 7.1.

When considering a second order time discretization scheme (Figures 4 (c) and (d)), the results obtained with turbulence modeling or with both correction closure terms and turbulence terms are more accurate than the results of the standard ROM, and are close to the projection of the full order solution on the reduced POD space. In particular, in terms of accuracy of the velocity reduced field, the results are very similar to the projected field.

7.3. Comparison of data-driven VMS-PPE-ROMs

In this section, the combined effect of data-driven terms and turbulence modeling is evaluated for the PPE approach for a simulation lasting 8 seconds in Figure 5. The correction terms are built starting from the first 2 seconds and all corrections are constructed by using the approach detailed
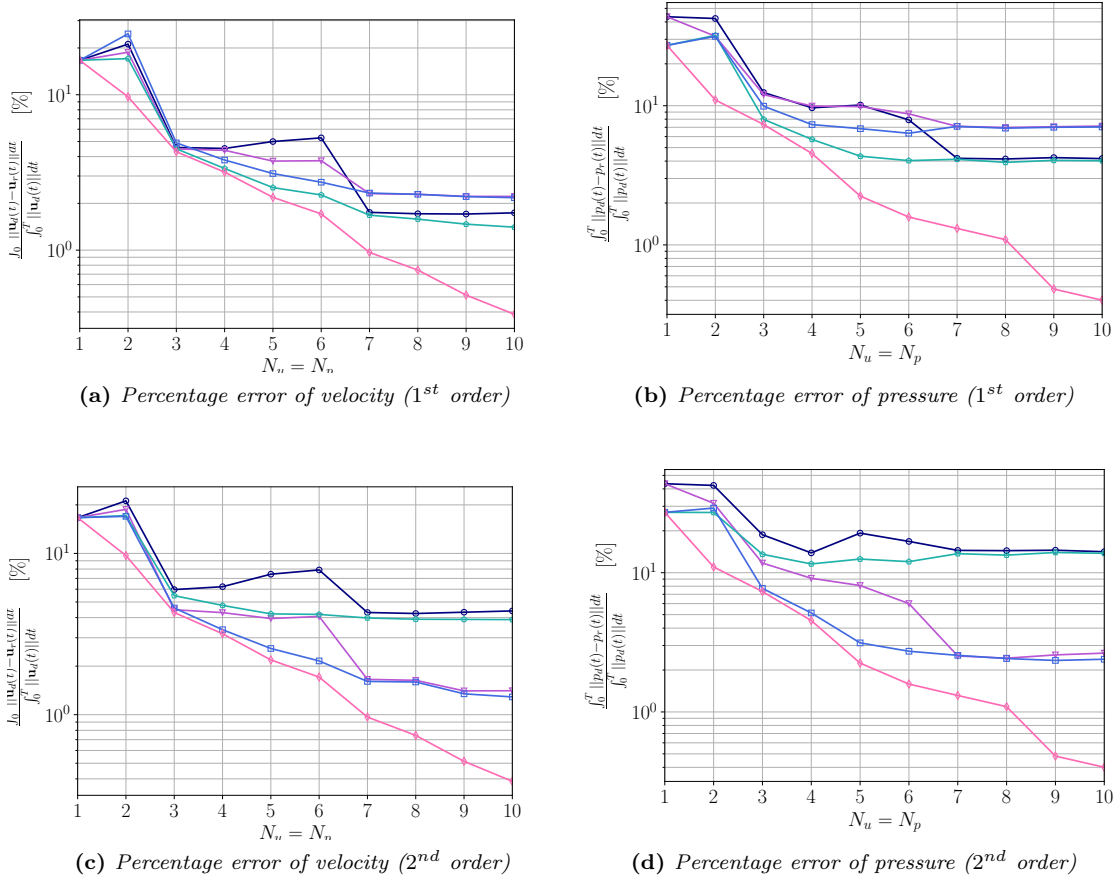
(a) *Percentage error of velocity (1st order)*

(b) *Percentage error of pressure (1st order)*

(c) *Percentage error of velocity (2nd order)*

(d) *Percentage error of pressure (2nd order)*

Figure 3: *Percentage integral errors of the absolute value of velocity and of pressure, varying the number of modes. The model is the PPE-ROM with a first ((a), (b)) and second ((c), (d)) order time scheme. The cases represented are the following: without any data-driven term (——○——); physically-based data-driven model (——▽——); purely data-driven model (——○——); hybrid data-driven model (——□——); projection (——◇——).*

in Section 4.2.

The differences between Figures 5 (a),(b) and 5 (c),(d) confirm what observed in the SUP-ROM investigation. When a first order scheme is used (Figure 5(a),(b)), the hybrid data-driven approach does not significantly improve the accuracy of the eddy viscosity or purely data-driven approaches. However, when a second order time scheme is used (Figure 5 (c),(d)), the combination of turbulence modeling and corrections leads to a significant accuracy improvement.

Remark 7.1 *Different behaviours of the reduced system in Figures 2, 3, 4 and 5 when using a first or second order time scheme, can be justified with the different dissipation associated with the two numerical time integration schemes.*

In fact, from the numerical examples, we deduce that, when considering a ROM which includes all the stabilizations of the FOM (i.e. data-driven + eddy viscosity), it is important to use the same temporal scheme employed at the FOM level (i.e. second order scheme). In the case of a ROM with missing stabilization terms (i.e. missing eddy viscosity for example) a first order scheme introduces additional numerical stabilization and gives a beneficial contribution.
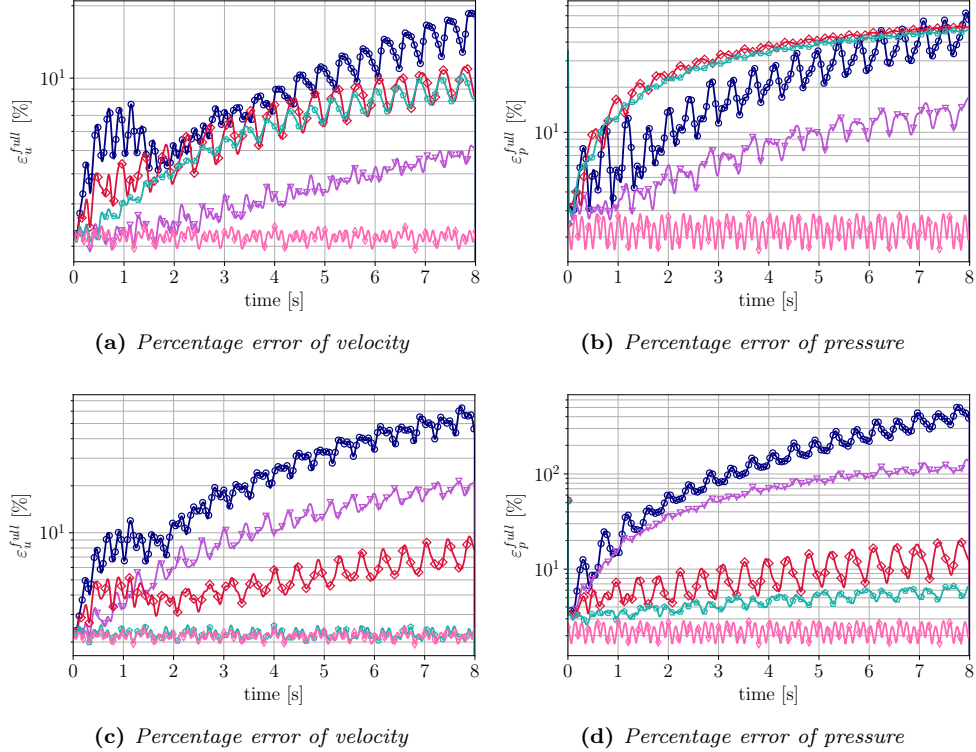
15

(a) *Percentage error of velocity*

(b) *Percentage error of pressure*

(c) *Percentage error of velocity*

(d) *Percentage error of pressure*

Figure 4: *Percentage errors of the absolute value of velocity and pressure, considering $N_u = N_p = N_{sup} = 5$. The model is the SUP-ROM with a first ((a), (b)) and second ((c), (d)) order time scheme. Results include the following cases: without any data-driven term (⎯○⎯); purely data-driven constrained model for the velocity (⎯▽⎯); physically-based data-driven model (⎯◇⎯); hybrid data-driven model (⎯⬠⎯); projection (⎯◇⎯).*

We also note that the model has an excellent extrapolation efficiency. Indeed, although the corrections are constructed with data from the first 2 seconds, they increase the ROM accuracy on the interval [2, 8] seconds. Finally, we point out that the instability of the second order integration scheme is damped by the addition of the turbulence model.

Overall, the numerical results in this section show that adding the data-driven velocity and pressure corrections proposed in [23] can significantly increase the accuracy of the turbulence ROMs proposed in [18].

7.4. Qualitative results

The inclusion of correction terms and turbulence modeling in the reduced formulations is also examined from a graphical point of view for the SUP-ROM and PPE-ROM approaches. The results are graphically represented on the test case grid by using the open-source application *Paraview* and the results are compared to those obtained in [23] and [18].

The second order time integration scheme is used since it provides the best results in Sections 7.2 and 7.3.

The POD is performed on the snapshots evaluated at the time instances within the interval [79.992, 99.992] seconds and the reduced order systems (29) and (30) are solved in the interval
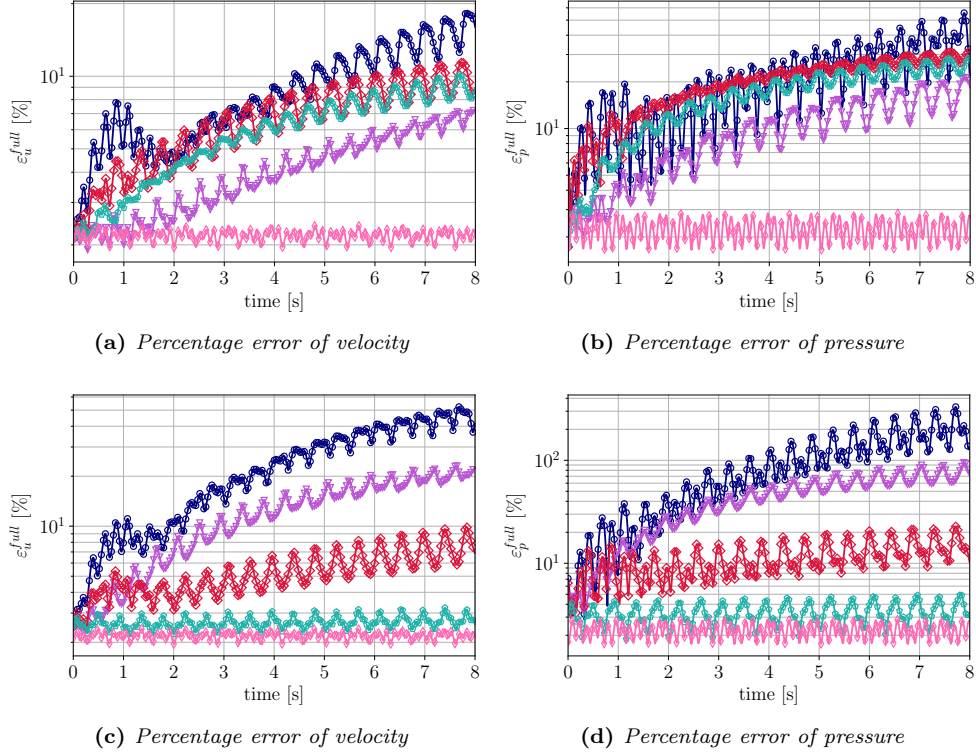
(a) *Percentage error of velocity*

(b) *Percentage error of pressure*

(c) *Percentage error of velocity*

(d) *Percentage error of pressure*

Figure 5: *Percentage errors of the absolute value of velocity and pressure, considering $N_u = N_p = N_{sup} = 5$. The model is the PPE-ROM, with a first ((a), (b)) and second ((c), (d)) order time scheme. Results include the following cases: without any data-driven term (———◦———); purely data-driven model, for both velocity and pressure (———▽———); physically-based data-driven model (———◇———); hybrid data-driven model (———◦———); projection (———◇———).*

$[79.992, 87.992]$ seconds, since the maximum length of the online simulations carried out is 8 seconds. For this reason, all the fields are captured at the final time step of online simulations, i.e., at 87.992 seconds.

The pressure and the velocity magnitude fields are displayed in Figures 6 and 7, respectively, for different SUP-ROM and PPE-ROM simulations. The fields computed with the standard ROMs and those coming from the systems including the data-driven terms are different. In particular, the fields in panels (e) and (f) of Figures 6 and 7 are closer to the full order fields displayed in panel (g) of Figures 6 and 7, especially in the region around the cylinder. The improvement of the accuracy nearby the circular cylinder is an important gain as it leads to a better reconstruction of the ROM lift coefficient.

8.   CONCLUSIONS AND OUTLOOK

One popular way to increase the accuracy of Galerkin ROMs in the under-resolved or marginally-resolved regimes is adding closure or correction terms [3]. In this paper, the data-driven modeling of these correction terms, deeply analyzed in [23], was combined with turbulence modeling, using the eddy viscosity formulation developed in [18]. This yielded a novel hybrid data-driven closure strategy, which we tested numerically. In our numerical investigation, we considered the two-
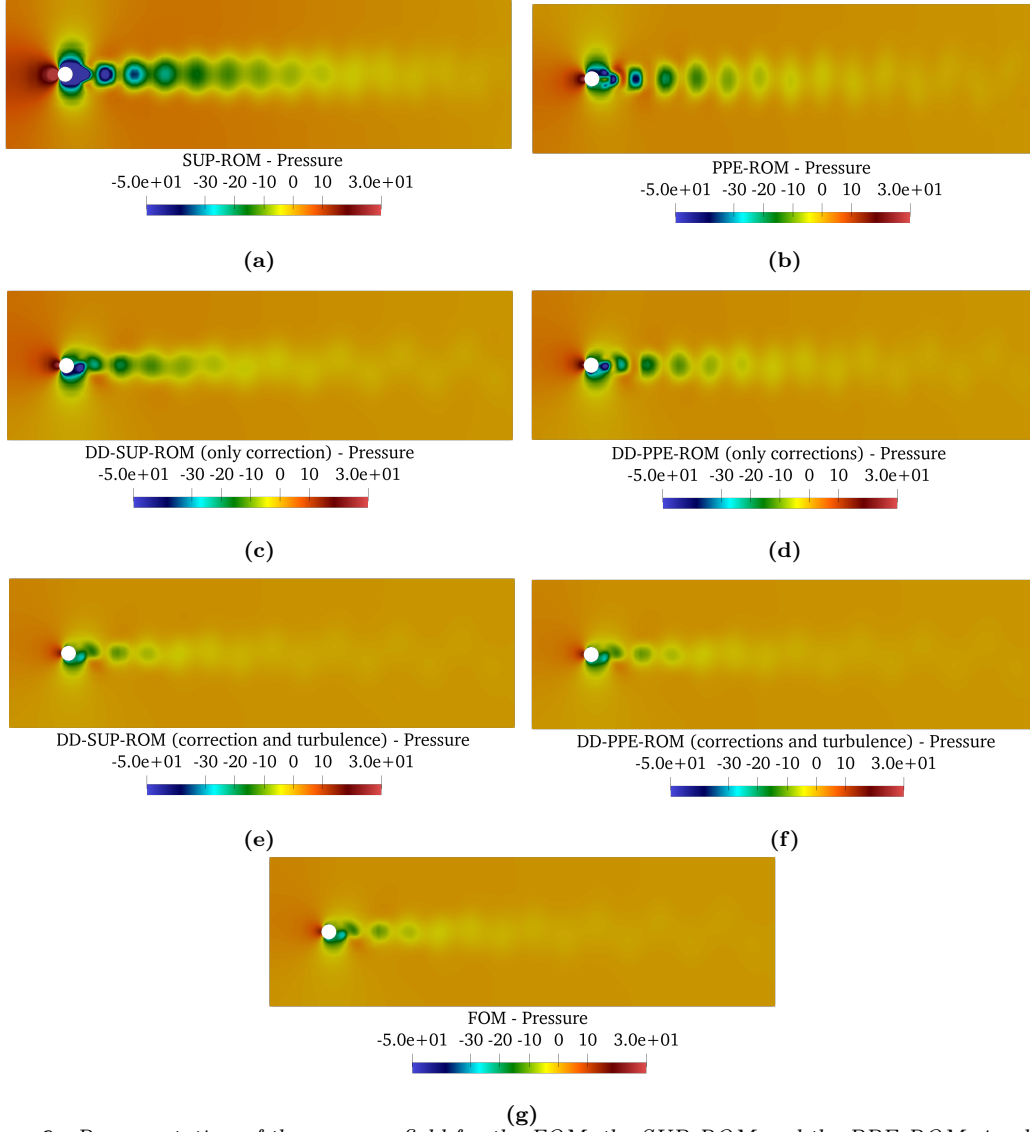
Figure 6: *Representation of the pressure field for the FOM, the SUP-ROM and the PPE-ROM simulations with and without the data-driven terms.*

dimensional flow past a circular cylinder at $Re = 50\,000$ in the marginally-resolved regime. We also considered several model configurations.

First, we tested two fundamentally different ROM pressure formulations: (i) The supremizer ROM (SUP-ROM), in which additional (supremizer) modes for the velocity approximation are used in order to satisfy the inf-sup condition. (ii) The pressure Poisson equation (PPE-ROM), in which the pressure approximation is determined by solving a Poisson equation instead of the continuity equation.

Secondly, a comparison between a first and a second order integration time scheme, in the resolution of the reduced system, was performed. In our numerical investigation, we observed that the numerical dissipation associated with the second order scheme leads to a better reconstruction
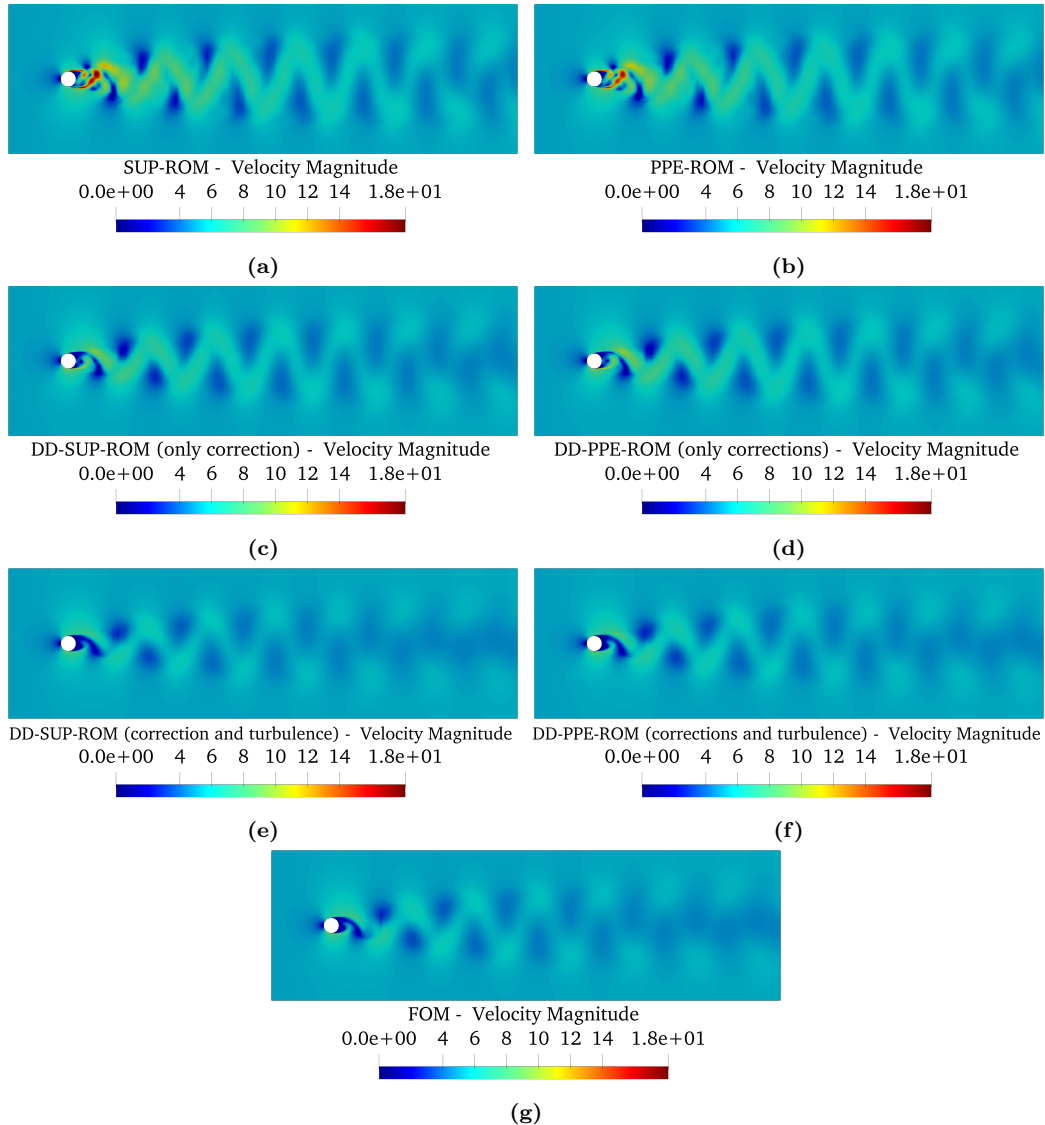
18

Figure 7: *Representation of the velocity magnitude field for the FOM, the SUP-ROM and the PPE-ROM simulations with and without the data-driven terms.*

of the pressure and velocity fields when the novel data-driven terms are added.

In conclusion, the hybrid data-driven closure strategy provides the best performance, leading to an increased accuracy of the reduced pressure and velocity field with respect to the high-fidelity solution, in both supremizer and Poisson approaches with a second order time integration scheme.

Outlook There are several research directions that can be investigated:

- The combined effect of correction terms and eddy viscosity modeling is highly influenced by

19

the nature of the test case taken into account. The numerical method proposed was tested in this paper only in the case of a 2D turbulent flow around a circular cylinder. Further tests with unsteady problems, with a more complex computational setting or with 3D grids will be analysed in future works.

Moreover, in this paper we focused on a single configuration for the training dataset: the snapshots are collected in a time interval of 20 seconds, the correction terms are built from data of 2 seconds and the online simulations last 8 seconds. In particular, the purely data-driven technique has a good extrapolation efficiency since an interval of 2 seconds (corresponding to 2 flow periods) collects enough information for the construction of the purely data-driven terms.

Different settings would intuitively influence the performance of our numerical method and will be addressed in future works.

- In this paper, the only parameter considered in the reduced order simulations is time. Thus, the matrices appearing in the correction terms' ansatzes are parameter independent. An interesting task for the future would be the introduction of a parameter in the reduced formulation — for instance the velocity at the inlet of the domain — as in [18]. In that case, the goal would be to express the data-driven terms as a function of both the parameter considered and the time.

9. ACKNOWLEDGEMENTS

References

[1] *ITHACA-FV|mathLab innovating with mathematics* website. `https://mathlab.sissa.it/ithaca-fv`.

[2] *OpenFOAM* website. `https://openfoam.org/`.

[3] S. E. Ahmed, S. Pawar, O. San, A. Rasheed, T. Iliescu, and B. R. Noack. On closures for reduced order models—a spectrum of first-principle to machine-learned avenues. *Phys. Fluids*, 33(9):091301, 2021.

[4] I. Akhtar, A. H. Nayfeh, and C. J. Ribbens. On the stability and extension of reduced-order Galerkin models in incompressible flows. *Theor. Comp. Fluid Dyn.*, 23(3):213–237, 2009.

[5] S. Ali, F. Ballarin, and G. Rozza. Stabilized reduced basis methods for parametrized steady Stokes and Navier–Stokes equations. *Computers & Mathematics with Applications*, 80(11):2399–2416, 2020.

[6] M. Azaïez, T. C. Rebollo, and S. Rubino. Streamline derivative projection-based POD-ROM for convection-dominated flows. Part I: Numerical Analysis. *arXiv preprint* `http://arxiv.org/abs/1711.09780`, 2017.

[7] 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. Meth. Engng.*, 102:1136–1161, 2015.

[8] P. Benner, P. Goyal, B. Kramer, B. Peherstorfer, and K. Willcox. Operator inference for non-intrusive model reduction of systems with non-polynomial nonlinear terms. *Computer Methods in Applied Mechanics and Engineering*, 372:113433, 2020.

[9] P. Benner, S. Grivet-Talocia, A. Quarteroni, G. Rozza, W. Schilders, and L. Miguel Silveira. Model Order Reduction: Basic Concepts and Notation. In *Model Order Reduction: Volume 1: System-and Data-Driven Methods and Algorithms*, pages 1–14. De Gruyter, 2021.

[10] P. Benner, W. Schilders, S. Grivet-Talocia, A. Quarteroni, G. Rozza, and L. Miguel Silveira. *Model Order Reduction: Volume 2: Snapshot-Based Methods and Algorithms*. De Gruyter, 2020.

[11] P. Benner, W. Schilders, S. Grivet-Talocia, A. Quarteroni, G. Rozza, and L. Miguel Silveira. *Model order reduction: Volume 3: applications*. De Gruyter, 2020.

[12] M. Bergmann, C. H. Bruneau, and A. Iollo. Enablers for robust POD models. *J. Comput. Phys.*, 228(2):516–538, 2009.

[13] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *Int. J. Num. Meth. Eng.*, 86(2):155–181, 2011.

[14] Kevin Carlberg, Youngsoo Choi, and Syuzanna Sargsyan. Conservative model reduction for finite-volume models. *Journal of Computational Physics*, 371:280–314, 2018.

[15] S. Grimberg, C. Farhat, and N. Youkilis. On the stability of projection-based model order reduction for convection-dominated laminar and turbulent flows. 419:109681, October 2020.

[16] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(02):277–302, 2008.

[17] J. S. Hesthaven, G. Rozza, and B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer, 2015.

[18] S. Hijazi, G. Stabile, A. Mola, and G. Rozza. Data-Driven POD–Galerkin reduced order model for turbulent flows. *Journal of Computational Physics*, 416:109513, 2020.

[19] P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge, 1996.

[20] A. Iollo, S. Lanteri, and J. A. Désidéri. Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations. *Theoret. Comput. Fluid Dyn.*, 13(6):377–396, 2000.

[21] R. I Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of computational physics*, 62(1):40–65, 1986.

[22] A. Ivagnes. Data Enhanced Reduced Order Methods for Turbulent Flows, Master Thesis, Politecnico di Torino, 2021.

[23] A. Ivagnes, G. Stabile, A. Mola, T. Iliescu, and G. Rozza. Pressure data-driven variational multiscale reduced order models. *arXiv preprint arXiv:2205.15118*, 2022.

[24] H. Jasak. Error analysis and estimation for the finite volume method with applications to fluid flows. 1996.

[25] W. P. Jones and B. E. Launder. The prediction of laminarization with a two-equation model of turbulence. *International journal of heat and mass transfer*, 15(2):301–314, 1972.

[26] I. Kalashnikova, M. F. Barone, S. Arunajatesan, and B. G. van Bloemen Waanders. Construction of energy-stable projection-based reduced order models. *Applied Mathematics and Computation*, 249:569–596, 2014.

[27] B. Karasözen, S. Yıldız, and M. Uzunca. Intrusive and data-driven reduced order modelling of the rotating thermal shallow water equation. *Applied Mathematics and Computation*, 421:126924, 2022.

[28] Andrej Nikolaevich Kolmogorov. Equations of turbulent motion in an incompressible fluid. In *Dokl. Akad. Nauk SSSR*, volume 30, pages 299–303, 1941.

[29] B. Kramer and K. E. Willcox. Nonlinear model order reduction via lifting transformations and proper orthogonal decomposition. 57(6):2297–2307, June 2019.

[30] Karl Kunisch and Stefan Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical analysis*, 40(2):492–515, 2002.

[31] D. Lazzaro and L. B. Montefusco. Radial basis functions for the multivariate interpolation of large scattered data sets. *Journal of Computational and Applied Mathematics*, 140(1-2):521–536, 2002.

[32] F. R Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA journal*, 32(8):1598–1605, 1994.

[33] Charles A Micchelli. Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive approximation*, 2(1):11–22, 1986.

[34] M. Mohebujjaman, L. G. Rebholz, and T. Iliescu. Physically-constrained data-driven correction for reduced order modeling of fluid flows. *Int. J. Num. Meth. Fluids*, 89(3):103–122, 2019.

[35] C. Mou. *Data-Driven Variational Multiscale Reduced Order Modeling of Turbulent Flows*. PhD thesis, Virginia Tech, 2021.

[36] C. Mou, B. Koc, O. San, L. G. Rebholz, and T. Iliescu. Data-driven variational multiscale reduced order models. *Computer Methods in Applied Mechanics and Engineering*, 373:113470, 2021.

[37] F. Moukalled, L Mangani, M. Darwish, et al. *The finite volume method in computational fluid dynamics*, volume 113. Springer, 2016.

[38] B. R. Noack and H. Eckelmann. A low-dimensional galerkin method for the three-dimensional flow around a circular cylinder. *Physics of Fluids*, 6(1):124–143, 1994.

[39] B. R. Noack, P. Papas, and P. A. Monkewitz. The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows. *J. Fluid Mech.*, 523:339–365, 2005.

[40] S. V. Patankar. *Numerical heat transfer and fluid flow*. CRC press, 2018.

[41] B. Peherstorfer and K. Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Comput. Methods Appl. Mech. Engrg.*, 306:196–215, 2016.

[42] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.

[43] T. C. Rebollo, E. D. Ávila, M. G. Mármol, F. Ballarin, and G. Rozza. On a certified Smagorinsky reduced basis turbulence model. *SIAM J. Numer. Anal.*, 55(6):3047–3067, 2017.

[44] Osborne Reynolds. IV. On the dynamical theory of incompressible viscous fluids and the determination of the criterion. *Philosophical transactions of the royal society of London*, (186):123–164, 1895.

[45] 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. Method. E.*, 15(3):229–275, 2008.

[46] G. Rozza, DB 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(1):115–152, 2013.

[47] G. Rozza and K. Veroy. On the stability of the reduced basis method for Stokes equations in parametrized domains. *Comput. Methods Appl. Mech. Engrg.*, 196(7):1244–1260, 2007.

[48] Gianluigi Rozza, Giovanni Stabile, and Francesco Ballarin. *Advanced Reduced Order Methods and Applications in Computational Fluid Dynamics*. Society for Industrial & Applied Mathematics, 2022.

[49] F. Salmoiraghi, A. Scardigli, H. Telib, and G. Rozza. Free-form deformation, mesh morphing and reduced-order methods: enablers for efficient aerodynamic shape optimisation. *International Journal of Computational Fluid Dynamics*, 32(4-5):233–247, 2018.

[50] D. B. Spalding. The numerical computation of turbulent flow. *Comp. Methods Appl. Mech. Eng.*, 3:269, 1974.

[51] 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(1):210–236, 2017.

[52] 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.

[53] G. Stabile, M. Zancanaro, and G. Rozza. Efficient geometrical parametrization for finite-volume-based reduced order methods. *International journal for numerical methods in engineering*, 121(12):2655–2682, 2020.

[54] S. K. Star, G. Stabile, F. Belloni, G. Rozza, and J. Degroote. A novel iterative penalty method to enforce boundary conditions in Finite Volume POD-Galerkin reduced order models for fluid dynamics problems. *Communications in Computational Physics*, 30(1):34–66, 2021.

[55] S Walton, O Hassan, and K Morgan. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20-21):8930–8945, 2013.

[56] Yi Wang, Bo Yu, Zhizhu Cao, Weizhong Zou, and Guojun Yu. A comparative study of POD interpolation and POD projection methods for fast and accurate prediction of heat transfer problems. *International Journal of Heat and Mass Transfer*, 55(17-18):4827–4836, 2012.

[57] X. Xie, M. Mohebujjaman, L. G. Rebholz, and T. Iliescu. Data-driven filtered reduced order modeling of fluid flows. *SIAM J. Sci. Comput.*, 40(3):B834–B857, 2018.