

A hyper-reduced MAC scheme for the parametric Stokes and Navier-Stokes equations *

Yanlai Chen[†], Lijie Ji[‡], Zhu Wang[§]

Abstract

The need for accelerating the repeated solving of certain parametrized systems motivates the development of more efficient reduced order methods. The classical reduced basis method is popular due to an offline-online decomposition and a mathematically rigorous *a posteriori* error estimator which guides a greedy algorithm offline. For non-linear and nonaffine problems, hyper reduction techniques have been introduced to make this decomposition efficient. However, they may be tricky to implement and often degrade the online computation efficiency.

To avoid this degradation, reduced residual reduced over-collocation (R2-ROC) was invented integrating empirical interpolation techniques on the solution snapshots and well-chosen residuals, the collocation philosophy, and the simplicity of evaluating the hyper-reduced well-chosen residuals. In this paper, we introduce an adaptive enrichment strategy for R2-ROC rendering it capable of handling parametric fluid flow problems. Built on top of an underlying Marker and Cell (MAC) scheme, a novel hyper-reduced MAC scheme is therefore presented and tested on Stokes and Navier-Stokes equations demonstrating its high efficiency, stability and accuracy.

1 Introduction

Developing fast algorithms for parametrized systems is a critical topic for researchers in many fields, especially when the problems require repeated numerical simulations. If the dimension of parameter space is high, the computation cost tends to be unacceptable. The

*Y. Chen was partially supported by National Science Foundation grant DMS-1719698 and by the UMass Dartmouth MUST program established by Dr. Ramprasad Balasubramanian via sponsorship from the Office of Naval Research. L. Ji acknowledges the support from NSFC (grant Nos. 12071288 and 21773165). Z. Wang was partially supported by National Science Foundation DMS-1913073.

[†]Department of Mathematics, University of Massachusetts Dartmouth, 285 Old Westport Road, North Dartmouth, MA 02747, USA. Email: yanlai.chen@umassd.edu.

[‡]School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai 200240, China. Email: sjtujidreamer@sjtu.edu.cn.

[§]Department of Mathematics, University of South Carolina, Columbia, SC 29208, United States. Email: wangzhu@math.sc.edu.

reduced basis method (RBM) [38, 27] was developed for parametrized PDEs (pPDEs), which is computation efficient and can produce an accurate approximation of the original model. Many existing literature and mathematical analysis have proved its effectiveness and cost saving capability [7, 18]. The key of RBM to realize cost saving is the offline-online decomposition procedure [40, 25]. That is, the algorithm calculates several high fidelity solutions through the full order model to construct the reduced basis space and build the reduced order model during the offline process. The online process only needs solving the reduced system whose assembly and resolution are independent of the degrees of freedom of the full order model. The efficiency of this offline-online procedure is trivial for the linear and affine system. However, for a nonaffine and nonlinear system, *Empirical Interpolation Method* (EIM) or its discrete version (DEIM) [4, 23, 11, 37] have to be applied to achieve online efficiency.

To illustrate the EIM, we consider the following time dependent linear equation with a nonaffine parameter dependence:

$$\frac{\partial u}{\partial t} - \mathcal{P}(u, t, \mathbf{x}; \boldsymbol{\mu}) = f, \quad \mathbf{x} \in \Omega, \quad t \in [0, T].$$

Adopting a simple backward Euler scheme, we have the following linear system to solve for advancing from t_k to t_{k+1} :

$$(I + \tau A(X^{\mathcal{N}}; \boldsymbol{\mu})) \mathbf{u}^{k+1} = \mathbf{u}^k + \tau f^{\mathcal{N}},$$

where $\boldsymbol{\mu}$ is the parameter, $\mathbf{u}^k = u^{\mathcal{N}}(t_k, X^{\mathcal{N}})$ represents the discretization of u at time t_k , $f^{\mathcal{N}}$ is the discretization of f , $X^{\mathcal{N}}$ represents all the discretized nodes, $\mathcal{N}$ is the degrees of freedom of FDM, I is the identity matrix, τ is the time step size, and A is the discrete counterpart of operator $-\mathcal{P}$. EIM generates an approximation of the original nonaffine operator $\mathcal{P}(u, t, \mathbf{x}; \boldsymbol{\mu})$ by a linear combination of $\boldsymbol{\mu}$ -independent operators

$$\mathcal{P}(u, t, \mathbf{x}; \boldsymbol{\mu}) \approx \sum_{q=1}^Q \theta_q(\boldsymbol{\mu}) \mathcal{P}_q(u, t, \mathbf{x}).$$

where $\{\theta_q(\boldsymbol{\mu})\}_{q=1}^Q$ are parameter dependent coefficients, $\mathcal{P}_q(u, t, \mathbf{x}) := \mathcal{P}(u, t, \mathbf{x}; \boldsymbol{\mu}^q)$, and $\{\boldsymbol{\mu}^q\}_{q=1}^Q$ are determined through the greedy framework of EIM. Using this affine approximation and letting $V \mathbf{c}^{k+1}$ denote the reduced order approximation of $\mathbf{u}^{k+1}$, the reduced system can be written as

$$(V^T V + \sum_{q=1}^Q \theta_q(\boldsymbol{\mu}) \tau V^T A_q V) \mathbf{c}^{k+1} = V^T V \mathbf{c}^k + \tau V^T f^{\mathcal{N}},$$

where A_q is the discretized matrix of the parameter independent operator $-\mathcal{P}_q$. This reduced form is obtained from Galerkin projection, one of the commonly used variational approaches. Petrov-Galerkin projection [6, 9] can be adopted as well. Obviously, this reduced solver is not dependent on the dimension of the full order model. However, its complexity is linearly dependent on the number of EIM terms Q . If Q is too large, assembling this affine

approximation will lead to an efficiency degradation. This may happen when, e.g. Ω is a parametrized domain [15, 5, 45].

To overcome the degradation from EIM technique, we adopt a (over-)collocation approach as we did in [13, 16, 14]. Methods in [16, 14] are named as *over-collocation* method because they choose extra collocation points compared to the collocation methods in [13]. The reduced residual reduced over-collocation (R2-ROC) [14] integrates empirical interpolation techniques on the solution snapshots and well-chosen residuals, the collocation philosophy, and the simplicity of evaluating the hyper-reduced well-chosen residuals for a stable surrogate solver for many pPDEs while keeping the online computational complexity independent of Q . However R2-ROC, applied directly to the more complicated unsteady fluid flow problems, leads to surrogate solutions of oscillating accuracy especially when the RB dimension is low. Motivated by the facts that the instability / lack of robustness is usually ameliorated by adapting or enriching the test space in Finite Element Method and that the classical RBM adopts global basis functions, we design an adaptive strategy to enrich the set of collocation points during the Offline process. The idea is to control the accuracy oscillation in the surrogate solutions by monitoring a *robustness indicator* and making sure that it is essentially non-oscillatory. We systematically select more collocation points, from a well-chosen residual, when a new basis is determined until the robustness indicator falls below a prescribed tolerance. We note that, when the robustness indicator stays below this tolerance, the adaptively enriched R2-ROC degrades to the original R2-ROC algorithm. As our numerical results will show, the original R2-ROC suffices for the steady Stokes and Navier-Stokes equations. That is, a bare minimum of $2N - 1$ points for N -dimensional RB spaces provide a stable approximation for those cases. However, the new adaptive enrichment is essential in generating reliable surrogate solutions for the unsteady Navier-Stokes equations.

We end the introduction by noting that the collocation idea has also been explored by many other researchers. For instance, authors in [2, 1] chose collocations based on the missing point interpolation and reduced the amount of original equations that were then solved by the Galerkin method. In [42, 43], the reduced basis space was constructed through the proper orthogonal decomposition (POD) method, and collocation points were selected by an adaptive algorithm in the context of finite element method. It was also applied to nonlinear dynamical systems with randomly sampled collocation points [8]. More recently, the reduced quadrature scheme has been developed and the calculation of residuals has been accelerated with sampled integration points. This approach uses less integration points, together with new weights, to generate full residual approximations. Authors in [21] developed an Energy-conserving Sampling and Weighting hyper reduction (ECSW) method to approximate a vector function using only a reduced mesh, which is determined by solving a non-negative least squares problem. Empirical quadrature procedure (EQP) in [47] and Empirical cubature method in [39] were all designed with reducing the integration points to accelerate calculations of nonlinear residuals. Moreover, model reduction for fluid dynamical system has been investigated by many research groups. Authors in [46, 30] used the POD method with eddy viscosity closure models to generate reduced basis space of structurally dominated turbulent flows. Rozza and Stabile [44] also used the POD to construct the reduced basis

space of the incompressible Navier-Stokes equations with moderate Reynolds number. Ito and Ravindran presented the reduced-order modeling to approximate viscous incompressible flows [31], where the reduced basis is constructed based on a Lagrange approach. Authors in [22] generated the basis space through a greedy framework with a constrained Galerkin projection. Our proposed R2-ROC with the adaptive enrichment differentiate itself via its greedy framework for simultaneous basis and collocation point generation through empirical interpolation procedures [34, 33] and a novel adaptive mechanism for additional collocation points necessary for unsteady Navier-Stokes equations.

The rest of paper is organized as follows. In Section 2, we present the full order numerical schemes for the Stokes and Navier-Stokes equations. In Section 3, we introduce the adaptively enriched R2-ROC method through a generalized time dependent pPDE. In Section 4, numerical results are presented for Stokes equations and incompressible Navier-Stokes equations. Finally, concluding remarks are drawn in Section 5.

2 The full order model (FOM) for fluid dynamics

In this section, we introduce three parametric fluid dynamics equations, namely the Stokes equations, steady incompressible Navier-Stokes (NS) equations, and the unsteady incompressible NS equations. As they are well-known, we simply list the equations and present the full order numerical schemes for each of them.

2.1 Stokes equations

We consider the Stokes problem [32, 35, 41] in a two-dimensional domain $\Omega = [0, 1] \times [0, 1]$:

$$\begin{aligned} -\frac{1}{\text{Re}}(u_{xx} + u_{yy}) + \partial_x p &= 0, \text{ in } \Omega, \\ -\frac{1}{\text{Re}}(v_{xx} + v_{yy}) + \partial_y p &= 0, \text{ in } \Omega, \\ u_x + v_y &= 0, \text{ in } \Omega. \end{aligned} \tag{1}$$

The first two are (x - and y -coordinate) momentum equations and the last one is the so-called continuity equation. Here, p denotes the pressure and u, v represent the velocity components in the x - and y -directions, respectively. We consider the Reynolds number Re a parameter, and introduce a second parameter in a parametric boundary condition on the top side of the domain while setting the rest of the boundary homogeneous:

$$\begin{aligned} u &= 1 + \sin\left(\frac{\pi}{2}\nu x\right), \text{ on } \Gamma_{\text{top}} = \{(x, y) : y = 1\} \\ u &= 0, \text{ on } \partial\Omega \setminus \Gamma_{\text{top}} \\ v &= 0, \text{ on } \partial\Omega. \end{aligned} \tag{2}$$

This paper concerns how to solve (1) efficiently when $\boldsymbol{\mu} = (\text{Re}, \nu)$ changes value in a

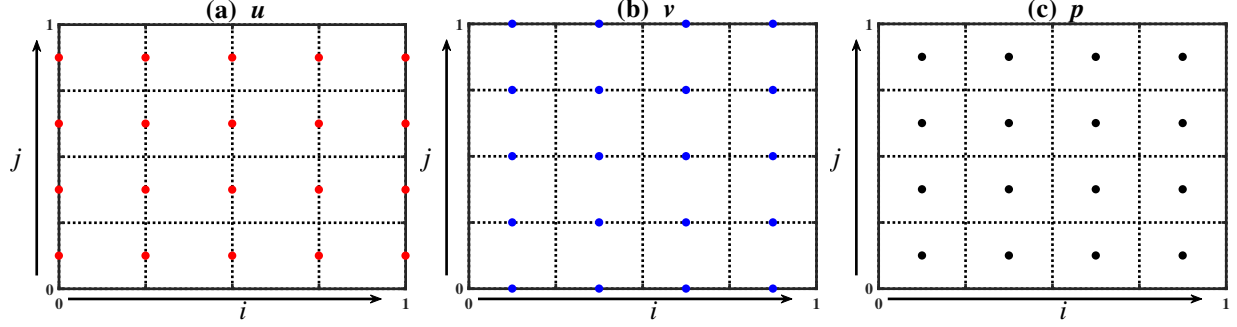


Figure 1: Location and indices of (u, v, p) variables in the MAC scheme. [12].

prescribed domain. Next, we first present the finite difference scheme, Marker and Cell (MAC) [26, 12, 36], for solving (1) with any given $\boldsymbol{\mu}$ value. Its idea is to place the unknowns (u, v, p) at different locations in a staggered fashion, see Figure 1, and then discretize the x -coordinate momentum equation at vertical edges, y -coordinate momentum equation at horizontal edges, and the continuity equation at cell centers. Namely, we have

$$\begin{aligned}
-\frac{1}{\text{Re}} \left(\frac{u_{i-1,j}^{\mathcal{N}} - 2u_{i,j}^{\mathcal{N}} + u_{i+1,j}^{\mathcal{N}}}{h_x^2} + \frac{u_{i,j-1}^{\mathcal{N}} - 2u_{i,j}^{\mathcal{N}} + u_{i,j+1}^{\mathcal{N}}}{h_y^2} \right) + \frac{p_{i+1,j}^{\mathcal{N}} - p_{i,j}^{\mathcal{N}}}{h_x} &= 0, \\
-\frac{1}{\text{Re}} \left(\frac{v_{i-1,j}^{\mathcal{N}} - 2v_{i,j}^{\mathcal{N}} + v_{i+1,j}^{\mathcal{N}}}{h_x^2} + \frac{v_{i,j-1}^{\mathcal{N}} - 2v_{i,j}^{\mathcal{N}} + v_{i,j+1}^{\mathcal{N}}}{h_y^2} \right) + \frac{p_{i,j+1}^{\mathcal{N}} - p_{i,j}^{\mathcal{N}}}{h_y} &= 0, \\
\frac{u_{i,j}^{\mathcal{N}} - u_{i-1,j}^{\mathcal{N}}}{h_x} + \frac{v_{i,j}^{\mathcal{N}} - v_{i,j-1}^{\mathcal{N}}}{h_y} &= 0.
\end{aligned} \tag{3}$$

where $\{u, v, p\}_{i,j}^{\mathcal{N}}$ denote each variable's value at the (i, j) -location of the corresponding grid, see Figure 1. The superscript $\mathcal{N}$, equals to the total number of degrees of freedom, dictates that the quantity is a numerical approximation as opposed to the exact solution. Concatenating all unknowns of the velocities and pressure and denoting the resulting vector by $\mathbf{u}(\boldsymbol{\mu}) = [u^{\mathcal{N}}; v^{\mathcal{N}}; p^{\mathcal{N}}]$, we can rewrite the numerical scheme (3) in a matrix form as:

$$A_S^{\mathcal{N}}(\boldsymbol{\mu})\mathbf{u}(\boldsymbol{\mu}) = \mathbf{f}^{\mathcal{N}}(\boldsymbol{\mu}), \tag{4}$$

where $\mathbf{f}^{\mathcal{N}}$ encodes the boundary treatment of u, v, p . We end by a remark that, in a multi-query or real-time setting, a system (4) of size $\mathcal{N} \times \mathcal{N}$ instantiated by a particular $\boldsymbol{\mu}$ value will have to be solved for each $\boldsymbol{\mu}$ value.

2.2 Steady-state Navier-Stokes equations

Next, we consider the steady-state incompressible NS equations [19, 3]. The system becomes nonlinear, but other than that the setup is the same as the Stokes equation. It reads:

$$\begin{aligned} -\frac{1}{\text{Re}}(u_{xx} + u_{yy}) + (u^2)_x + (uv)_y + \partial_x p &= 0, \text{ in } \Omega, \\ -\frac{1}{\text{Re}}(v_{xx} + v_{yy}) + (uv)_x + (v^2)_y + \partial_y p &= 0, \text{ in } \Omega, \\ u_x + v_y &= 0, \text{ in } \Omega. \end{aligned} \quad (5)$$

Homogeneous boundary conditions are considered on all but the top side of the boundary where x -velocity u is parameterized by the boundary control parameter ν , $u = 1 + \nu x$. The MAC scheme is applied in the same fashion on a staggered grid. We adopt the Picard's iteration to solve the resulting nonlinear system. The detailed linearized iterative scheme is as follows

$$\begin{aligned} &-\frac{1}{\text{Re}} \left(\frac{u_{i-1,j}^{\mathcal{N},\ell} - 2u_{i,j}^{\mathcal{N},\ell} + u_{i+1,j}^{\mathcal{N},\ell}}{h_x^2} + \frac{u_{i,j-1}^{\mathcal{N},\ell} - 2u_{i,j}^{\mathcal{N},\ell} + u_{i,j+1}^{\mathcal{N},\ell}}{h_y^2} \right) + \frac{p_{i+1,j}^{\mathcal{N},\ell} - p_{i,j}^{\mathcal{N},\ell}}{h_x} \\ &+ u_{i,j}^{\mathcal{N},\ell-1} \frac{u_{i+1,j}^{\mathcal{N},\ell} - u_{i-1,j}^{\mathcal{N},\ell}}{2h_x} + \frac{v_{i,j}^{\mathcal{N},\ell-1} + v_{i,j-1}^{\mathcal{N},\ell-1} + v_{i+1,j}^{\mathcal{N},\ell-1} + v_{i+1,j-1}^{\mathcal{N},\ell-1}}{4} \frac{u_{i,j+1}^{\mathcal{N},\ell} - u_{i,j-1}^{\mathcal{N},\ell}}{2h_y} = 0, \\ &-\frac{1}{\text{Re}} \left(\frac{v_{i-1,j}^{\mathcal{N},\ell} - 2v_{i,j}^{\mathcal{N},\ell} + v_{i+1,j}^{\mathcal{N},\ell}}{h_x^2} + \frac{v_{i,j-1}^{\mathcal{N},\ell} - 2v_{i,j}^{\mathcal{N},\ell} + v_{i,j+1}^{\mathcal{N},\ell}}{h_y^2} \right) + \frac{p_{i,j+1}^{\mathcal{N},\ell} - p_{i,j}^{\mathcal{N},\ell}}{h_y} \\ &+ \frac{u_{i,j}^{\mathcal{N},\ell-1} + u_{i-1,j}^{\mathcal{N},\ell-1} + u_{i-1,j+1}^{\mathcal{N},\ell-1} + u_{i,j+1}^{\mathcal{N},\ell-1}}{4} \frac{v_{i+1,j}^{\mathcal{N},\ell} - v_{i-1,j}^{\mathcal{N},\ell}}{2h_x} + v_{i,j}^{\mathcal{N},\ell-1} \frac{v_{i,j+1}^{\mathcal{N},\ell} - v_{i,j-1}^{\mathcal{N},\ell}}{2h_y} = 0, \\ &\frac{u_{i,j}^{\mathcal{N},\ell} - u_{i-1,j}^{\mathcal{N},\ell}}{h_x} + \frac{v_{i,j}^{\mathcal{N},\ell} - v_{i,j-1}^{\mathcal{N},\ell}}{h_y} = 0, \end{aligned} \quad (6)$$

where $\{u, v, p\}_{i,j}^{\mathcal{N},\ell}$ denote each variable's value at the (i, j) -location of the corresponding grid during the ℓ^{th} Picard iteration. Concatenating the high-fidelity solutions at the ℓ^{th} iteration by $\mathbf{u}^\ell = [u^{\mathcal{N},\ell}; v^{\mathcal{N},\ell}; p^{\mathcal{N},\ell}]$, we denote the matrix form of this numerical scheme by

$$A_{\text{NS}}^{\mathcal{N}}(\mathbf{u}^{\ell-1}; \boldsymbol{\mu}) \mathbf{u}^\ell = f^{\mathcal{N}}(\mathbf{u}^{\ell-1}; \boldsymbol{\mu}), \quad (7)$$

here $\boldsymbol{\mu} = (\text{Re}, \nu)$. The stopping criterion of the Picard iteration is that the norm of the full residual $\|A_{\text{NS}}^{\mathcal{N}}(\mathbf{u}^\ell; \boldsymbol{\mu}) \mathbf{u}^\ell - f^{\mathcal{N}}(\mathbf{u}^\ell; \boldsymbol{\mu})\|$ is below a prescribed tolerance.

2.3 Time-dependent Navier-Stokes equations

Finally, we consider the time-dependent NS equations [46, 44] in the two dimensional domain $\Omega = [0, 1] \times [0, 1]$,

$$\begin{aligned} \frac{\partial u}{\partial t} - \frac{1}{\text{Re}}(u_{xx} + u_{yy}) + (u^2)_x + (uv)_y + \partial_x p &= 0, \text{ in } \Omega, \\ \frac{\partial v}{\partial t} - \frac{1}{\text{Re}}(v_{xx} + v_{yy}) + (uv)_x + (v^2)_y + \partial_y p &= 0, \text{ in } \Omega, \\ u_x + v_y &= 0, \text{ in } \Omega, \end{aligned} \quad (8)$$

with a given initial condition. As in previous cases, homogeneous boundary conditions are considered on all but the top of the boundary where x -velocity u is designated to be $u = 1 + \nu x$. The spatial MAC scheme for this case is identical to (6) which means we are solving

$$\begin{pmatrix} u^{\mathcal{N},\ell} \\ v^{\mathcal{N},\ell} \\ 0 \end{pmatrix}_t + A_{\text{NS}}^{\mathcal{N}}(\mathbf{u}^{\ell-1}; \boldsymbol{\mu}) \mathbf{u}^\ell = f^{\mathcal{N}}(\mathbf{u}^{\ell-1}; \boldsymbol{\mu}), \quad (9)$$

at each Picard iteration. With backward Euler being our time discretization for simplicity, we have the following fully discretized scheme

$$\begin{pmatrix} u_{(k+1)}^{\mathcal{N},\ell} \\ v_{(k+1)}^{\mathcal{N},\ell} \\ 0 \end{pmatrix} + \tau A_{\text{NS}}^{\mathcal{N}}(\mathbf{u}_{(k+1)}^{\ell-1}; \boldsymbol{\mu}) \mathbf{u}_{(k+1)}^\ell = \tau f^{\mathcal{N}}(\mathbf{u}_{(k+1)}^{\ell-1}; \boldsymbol{\mu}) + \begin{pmatrix} u_{(k)}^{\mathcal{N},\ell} \\ v_{(k)}^{\mathcal{N},\ell} \\ 0 \end{pmatrix}. \quad (10)$$

Here τ is the time stepsize, subscript (k) denotes the k^{th} timestep value of the corresponding variable while the superscript ℓ continues to index the Picard iteration.

3 The reduced over-collocation method with adaptive enrichment

Following the FOMs presented in the last section, we introduce our proposed Reduced Order Model (ROM) for these three parametric fluid dynamics equations. It is an augmentation of the Reduced Over Collocation (ROC) approach introduced in [16, 14] that was developed for steady-state and time-dependent nonlinear problems. The trademark feature of ROC is the immunity of the degradation in online efficiency suffered by regular RBM as a result of the EIM-like expansion of the nonlinear and nonaffine terms. In what follows, we provide a review of R2-ROC following [14] in Section 3.1 and then describe in Section 3.2 the proposed augmentation, an adaptive enrichment of the collocation points that is automatically turned for the time-dependent case.

3.1 Reduced residual reduced over-collocation (R2-ROC) Method

The R2-ROC algorithm is applicable to both steady-state and time-dependent equations. To illustrate it, we consider the following general time dependent nonlinear and nonaffine PDE:

$$\frac{\partial u}{\partial t} - \mathcal{P}(u, t, \mathbf{x}; \boldsymbol{\mu}) = g, \quad \mathbf{x} \in \Omega, \quad t \in [0, T] \quad (11)$$

with necessary boundary and initial conditions. Here $\mathcal{P}$ is a nonlinear differential operator of u that is parameterized, perhaps in a nonaffine fashion, by $\boldsymbol{\mu}$, and g is a forcing term. We assume access to a high fidelity solver for (11) that is written in a nodal form. To fix the notation, we use $X^{\mathcal{N}}$ to denote all the grid points, $\mathcal{P}_{\mathcal{N}}$ and $g^{\mathcal{N}}$ represent their discretized counterparts of $\mathcal{P}$ and g respectively. The FOM solution corresponding to parameter $\boldsymbol{\mu}$ at time t is denoted by $u^{\mathcal{N}}(t, \mathbf{x}; \boldsymbol{\mu})$ which we assume is close enough to the exact solution $\mathbf{u}(t, \mathbf{x}; \boldsymbol{\mu})$ for us to adopt as a reference for the ROM. Lastly, we will be referring to the (full or reduced version of the) residual when a surrogate solution v is found. This residual is also written in a nodal form.

$$r(\mathbf{v}) = \mathbf{v}_t - \mathcal{P}_{\mathcal{N}}(\mathbf{v}, t; \boldsymbol{\mu}) - g^{\mathcal{N}}. \quad (12)$$

Now we are ready to briefly review the R2-ROC algorithm. It has two components: an online (reduced) solver of size n that is between 1 and N with N usually much smaller than $\mathcal{N}$, and an offline training component which repeatedly calls the online solver of increasing size n to build up a surrogate solution space from scratch dimension-by-dimension.

Online solver

Given the reduced space W_n and a collocation set $X^m (\subset X^{\mathcal{N}})$ of cardinality m that is comparable to n , R2-ROC identifies a surrogate solution for any specific parameter $\boldsymbol{\mu}$ in the following form

$$\hat{\mathbf{u}}_n(\boldsymbol{\mu}, t) = W_n \mathbf{c}_n(\boldsymbol{\mu}, t). \quad (13)$$

Here, for simplicity of notation, we also adopt W_n for the snapshot matrix whose column space forms the reduced space W_n . The unknown coefficients $\mathbf{c}_n(\boldsymbol{\mu}, t) \in \mathbb{R}^{n \times 1}$ is obtained by minimizing a subsampled residual

$$\mathbf{c}_n(\boldsymbol{\mu}, t) = \underset{\omega}{\operatorname{argmin}} \left\| P_* \left((W_n \omega)_t - \mathcal{P}_{\mathcal{N}}(W_n \omega, t; \boldsymbol{\mu}) - g^{\mathcal{N}} \right) \right\|_{\ell^2(\mathbb{R}^m)}. \quad (14)$$

The subsampling matrix $P_* \in \mathbb{R}^{m \times \mathcal{N}}$, RB space W_n , and the reduced collocation set X^m will be generated in the offline process that is described next.

Offline training

The offline component utilizes a parameter-time greedy framework [23, 24] to iteratively construct the reduced basis space W_n and subsequently enrich the collocation set X^m which determines the subsampling matrix P_* . To describe this iterative procedure, we need to fix more notations. We discretize the time domain by $\mathcal{T}_f := \{t_i : i = 0, \dots, \mathcal{N}_t\}$ with $t_0 = 0$ being the initial time and $\mathcal{N}_t = T/\tau$ where τ is the temporal step-size and T the final time. The set of reduced time nodes, denoted by $\mathcal{T}_r$, is a subset of $\mathcal{T}_f$ and will be gradually enriched as the new bases are generated. In addition, we need the GEIM [33] interpolating functional as described in [14]. It is defined for any admissible function $v(\mathbf{x})$ any $\mathbf{x} \in \Omega$, $t \in [0, T]$, and $\boldsymbol{\mu} \in \mathcal{D}$

$$\sigma_{\mathbf{x},t}^{\boldsymbol{\mu}}(v) = v_t - \mathcal{P}(v(t, \mathbf{x}; \boldsymbol{\mu}), t; \boldsymbol{\mu}). \quad (15)$$

Here, we will also adopt $\sigma_{\mathbf{x},t}^{\boldsymbol{\mu}}(\cdot)$ for its discretized form. The algorithm judiciously identifies different parameter-time pairs

$$\left\{ (\boldsymbol{\mu}^1, t_{\boldsymbol{\mu}^1}^1), \dots, (\boldsymbol{\mu}^1, t_{\boldsymbol{\mu}^1}^{k_{\boldsymbol{\mu}^1}}); \dots (\boldsymbol{\mu}^n, t_{\boldsymbol{\mu}^n}^1), \dots, (\boldsymbol{\mu}^n, t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}}) \right\}$$

one-by-one (but not necessarily in the order above), and construct the reduced basis space via the corresponding snapshots. With these notations set, we start the greedy procedure with a randomly chosen $\boldsymbol{\mu}^1$ and obtain the snapshots $\{\mathbf{u}(t_i, X^{\mathcal{N}}; \boldsymbol{\mu}^1)\}_{i=0}^{\mathcal{N}_t}$ by the high fidelity algorithm. $\mathcal{T}_r$ is initiated by the time instant when the corresponding snapshot has the largest variation. That is,

$$\mathcal{T}_r = \{t_{\boldsymbol{\mu}^1}^1\} \text{ where } t_{\boldsymbol{\mu}^1}^1 = \operatorname{argmax}_{t \in \mathcal{T}_f} \left(\max_{x \in X^{\mathcal{N}}} \mathbf{u}(t, X^{\mathcal{N}}; \boldsymbol{\mu}^1) - \min_{x \in X^{\mathcal{N}}} \mathbf{u}(t, X^{\mathcal{N}}; \boldsymbol{\mu}^1) \right).$$

The RB space W_1 is then set as $W_1 = \{u_1\} = \{\mathbf{u}(t_{\boldsymbol{\mu}^1}^1, X^{\mathcal{N}}; \boldsymbol{\mu}^1)\}$, and the first collocation point chosen as the GEIM point of the first basis $\mathbf{x}_*^1 = \operatorname{argmax}_{x \in X^{\mathcal{N}}} |\sigma_{\mathbf{x}, t_{\boldsymbol{\mu}^1}^1}^{\boldsymbol{\mu}^1}(u_1)|$ [14]. When the first pair $(\boldsymbol{\mu}^1, t_{\boldsymbol{\mu}^1}^1)$ is determined, we use the online procedure described above to obtain an RB approximation $\hat{\mathbf{u}}_n(\boldsymbol{\mu}, t)$ for each parameter $\boldsymbol{\mu}$ in Ξ_{train} (a discretization of the parameter domain $\mathcal{D}$) and compute its error estimator $\Delta_n^{RRt}(\boldsymbol{\mu})$.

$$\Delta_n^{RRt}(\boldsymbol{\mu}) := \sum_{t \in \mathcal{T}_f} \varepsilon^{RR}(t; \boldsymbol{\mu}) \quad \text{with} \quad \varepsilon^{RR}(t; \boldsymbol{\mu}) := \|P_* \mathbf{r}_n(t; \boldsymbol{\mu})\|_{\infty}. \quad (16)$$

Here,

$$\mathbf{r}_n(t; \boldsymbol{\mu}) = W_n \mathbf{c}_t(\boldsymbol{\mu}) - \mathcal{P}_{\mathcal{N}}(W_n \mathbf{c}(\boldsymbol{\mu}), t; \boldsymbol{\mu}) - g^{\mathcal{N}}, \quad (17)$$

is the *full* residual for the current RB approximation $\hat{\mathbf{u}}_n(\boldsymbol{\mu}, t)$ at time t of parameter $\boldsymbol{\mu}$. $P_* \mathbf{r}_n(t; \boldsymbol{\mu}) \in \mathbb{R}^{m \times 1}$ then represents its *reduced* (subsamped) version.¹ Then the next $(\boldsymbol{\mu}, t)$ pair will be determined in the following greedy fashion [14]:

¹The conventional error estimate calculates the negative-order norm of the residual and scales it by the (parametric) stability factor. It is challenging to compute for the nonlinear and nonaffine case with EIM expansion, the involvement of the successive constraint method [29, 28] used to efficiently estimate the parametric stability factor, and the delicacy of evaluating the residual norm even for the linear problem [10, 17]. This simple error estimator based on the reduced residual [14] has shown to be promising for nonlinear and nonaffine problems without the need of EIM expansion.

1. **Greedy in μ :** The greedy choice for the μ -component of the (μ, t) pair is through maximizing $\Delta_n^{RRt}(\mu)$ over the training set Ξ_{train} :

$$\mu^{n+1} = \operatorname{argmax}_{\mu \in \Xi_{\text{train}}} \Delta_n^{RRt}(\mu), \quad \Delta_n = \Delta_n^{RRt}(\mu^{n+1}). \quad (18)$$

2. **Greedy in t :** Once we have a new μ^{n+1} , we calculate the ROC approximations $\hat{u}_n(\mu^{n+1}, t) = W_n \mathbf{c}_n(\mu^{n+1}, t)$ for all time levels $t \in \mathcal{T}_f$, and then the greedy t -choice is given as

$$t_{\mu^{n+1}}^{k_{\mu^{n+1}}} := \operatorname{argmax}_{t \in \mathcal{T}_f} \{ \varepsilon^{RR}(t; \mu) := \|P_* \mathbf{r}_n(t; \mu^{n+1})\|_\infty \}, \text{ and } \mathcal{T}_r = \mathcal{T}_r \cup \{t_{\mu^{n+1}}^{k_{\mu^{n+1}}}\}. \quad (19)$$

X^m expansion: With the new selected $(\mu^{n+1}, t_{\mu^{n+1}}^{k_{\mu^{n+1}}})$, we solve for the truth approximations $\mathbf{u}(t, X^N; \mu^{n+1})$ for $t \leq t_{\mu^{n+1}}^{k_{\mu^{n+1}}}$. We then obtain the first additional collocation point $\{\mathbf{x}_*^{n+1}\}$ from the GEIM process of $\mathbf{u}(t_{\mu^{n+1}}^{k_{\mu^{n+1}}}, X^N; \mu^{n+1})$, see Algorithm 1, and the second additional point $\{\mathbf{x}_{**}^n\}$ by the EIM process of the full residual $\mathbf{r}_n(t_{\mu^{n+1}}^{k_{\mu^{n+1}}}; \mu^{n+1})$, see Algorithm 2. The analysis in [14] demonstrates the importance of retaining these two sets of points for producing accurate approximations for both the reduced solution and the residual corresponding to each parameter.

We end this subsection by noting that the GEIM interpolating functional for the steady-state problems is defined as: for any admissible function $v(\mathbf{x})$, any $\mathbf{x} \in \Omega$, and $\mu \in \mathcal{D}$: $\sigma_{\mathbf{x}}^\mu(v) = \mathcal{P}(v(\mathbf{x}; \mu); \mu)$. The algorithm now only needs to identify different parameters $\{\mu^1, \dots, \mu^n\}$ and the error estimator is calculated by $\Delta_n^{RR}(\mu) := \|P_* \mathbf{r}_n(\mu)\|_\infty$ with $\mathbf{r}_n(\mu) = \mathcal{P}_N(W_n \mathbf{c}(\mu); \mu) - g^N$ being the *full* residual for the current RB approximation $\hat{u}_n(\mu)$ of parameter μ . The $(n+1)$ -th parameter is still chosen by maximizing $\Delta_n^{RR}(\mu)$ over the training set.

Algorithm 1 GEIM algorithm for the snapshots

- 1: **Input:** current basis space W_n , interpolation operators $\{\sigma_1, \dots, \sigma_n\}$, collocation set X^m , current subsampling operator matrix P_* , parameter-time pair $(\mu^{n+1}, t_{\mu^{n+1}}^{k_{\mu^{n+1}}})$ and new basis ξ_{n+1} .
 - 2: Compute a generalized interpolatory residual for vector ξ_{n+1} :
Find $\xi_{n+1} = \xi_{n+1} - W_n \alpha$ with $\{\alpha_i\}$ determined by requiring that $\sigma_i(\xi_{n+1}) = 0$ for $i = 1, \dots, n$.
Find $\mathbf{x}_*^{n+1} = \operatorname{argmax}_{\mathbf{x} \in X^N/X^m} \left| \sigma_{\mathbf{x}, t_{\mu^{n+1}}}^{\mu^{n+1}}(\xi_{n+1}) \right|$.
Define $\sigma_{n+1}(\cdot) := \sigma_{\mathbf{x}_*^{n+1}, t_{\mu^{n+1}}}^{\mu^{n+1}}(\cdot)$. $\xi_{n+1} = \xi_{n+1} / \sigma_{n+1}(\xi_{n+1})$ and let i_1 be the index of $\mathbf{x}_*^{n+1}$ in X^N .
 - 3: **Output:** $\{\sigma_1, \dots, \sigma_{n+1}\}$, i_1 , ξ_{n+1} , $\{\mathbf{x}_*^{n+1}\}$.
-

Algorithm 2 EIM algorithm for the residuals

- 1: **Input:** current basis space W_n , current subsampling operator matrix P_* , interpolatory residual vectors $\{r_1, \dots, r_{n-1}\}$, collocation set X^m , residual-based collocation set X_r^{n-1} , collocation $\mathbf{x}_*^{n+1}$ and parameter-time pair $(\boldsymbol{\mu}^{n+1}, t_{\boldsymbol{\mu}^{n+1}}^{k_{\boldsymbol{\mu}^{n+1}}})$.
 - 2: Calculate the full residual $r_n(t_{\boldsymbol{\mu}^{n+1}}^{k_{\boldsymbol{\mu}^{n+1}}}, \boldsymbol{\mu}^{n+1})$ by eq. (17).
 - 3: Compute an interpolatory residual r_n :
 Find $r_n = r_n - \sum_{j=1}^{n-1} \alpha_j r_j$ with $\{\alpha_j\}$ determined by requiring that $r_n(X_r^{n-1}) = 0$.
 Find $\mathbf{x}_{**}^n = \operatorname{argmax}_{\mathbf{x} \in X^{\mathcal{N}} / \{X^m, \mathbf{x}_*^{n+1}\}} |r_n|$.
 Calculate $r_n = r_n / r_n(\mathbf{x}_{**}^n)$, and let i_2 be the index of $\mathbf{x}_{**}^n$ in $X^{\mathcal{N}}$.
 - 4: **Output:** $\{r_1, \dots, r_{n-1}, r_n\}$, i_2 , $\{\mathbf{x}_{**}^n\}$.
-

3.2 Adaptive enrichment of R2-ROC

The original R2-ROC algorithm, if applied directly to the more complicated unsteady fluid flow problems targeted in this paper, leads to surrogate solutions of oscillating accuracy. This oscillation is usually more pronounced when the RB dimension is low. The instability / lack of robustness is usually ameliorated by enriching the test space in Finite Element Method. This observation and the fact that the classical RBM and other linear MOR methods adopt global basis functions motivate us to design an adaptive strategy to enrich the set of collocation points during the Offline process. This idea is realized by calculating a *robustness indicator* and then systematically selecting more collocation points in a try-and-error fashion when a new basis is determined until the robustness indicator falls below a prescribed tolerance.

To present the detail of our adaptive enrichment, we first recall that the n th parameter-time pair is determined based on Δ_{n-1} . Once it is obtained, we substitute the $(n-1)$ dimensional ROC approximation, at the chosen parameter-time pair, into the full order model for its residual $\mathbf{r}_{n-1}$,

$$\mathbf{r}_{n-1} = (\widehat{\mathbf{u}}_{n-1})_t(t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}}; \boldsymbol{\mu}^n) - \mathcal{P}_{\mathcal{N}}(\widehat{\mathbf{u}}_{n-1}(t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}}; \boldsymbol{\mu}^n), t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}}; \boldsymbol{\mu}^n) - g^{\mathcal{N}},$$

After an EIM orthonormalization with previous such residuals, the (original) ROC algorithm only selects one point, the maximizer of $|r_{n-1}|$, in addition to the one point from the GEIM process for the solutions. It then builds the n -dimensional ROC space and proceed to select the $(n+1)$ th parameter-time pair, based on Δ_n .

The essence of our augmentation is to define a *robustness indicator*

$$\rho_{\Delta}^n = \frac{\Delta_n}{\Delta_{n-1}},$$

and control the accuracy oscillation in the surrogate solutions by making sure that ρ_{Δ}^n is essentially non-oscillatory. We note that this is a natural choice whenever the effectivity index of Δ_n remains close to being a constant. This is achieved by revisiting the EIM-orthonormalized residual $\mathbf{r}_{n-1}$ and extract additional collocation points which, in turn, refines the n -dimensional ROC solver. We then update error indicator Δ_n and the robustness

indicator ρ_Δ^n . This process is repeated until ρ_Δ^n is smaller than a prescribed tolerance γ or the total number of additional collocation points surpasses a prescribed cap n_{adap}^{max} . These *adaptively enriched* points are sampled according to an inverse CDF (Cumulative Distribution Function) strategy. Specifically, we generate the CDF $\mathcal{F}_{|r_{n-1}|}$ of the (discretized) domain of $r_{n-1}(\mathbf{x})$ according to the function's absolute value and then perform a uniform inverse sampling of n_{adap} collocation points from the top p_{adap} percentile. Denoting these additional collocation points by $\mathbf{x}_+^{n-1}$, we have

$$\mathbf{x}_+^{n-1} = \mathcal{F}_{|r_{n-1}|}^{-1} \left(1 - p_{adap} : \frac{p_{adap}}{n_{adap} - 1} : 1 \right), \quad (20)$$

where $a : h : b$ is the Matlab notation of sampling the interval $[a, b]$ with a stepsize h .

The second set of residual-based collocation points is then updated as $X_r^{n-1} = X_r^{n-1} \cup \{\mathbf{x}_+^{n-1}\}$, and the whole collocation set updated as $X^m = X^m \cup \{\mathbf{x}_+^{n-1}\}$. Correspondingly, a new n th parameter will be determined based on this updated error estimator Δ_n as a consequence of the reduced collocation set being enriched. The adaptive algorithm is displayed in Algorithm 3 and the complete R2-ROC with adaptive enrichment for time dependent problems is shown in Algorithm 4. We note that, when the robustness indicator ρ_Δ^n stays below γ , the adaptively enriched R2-ROC will degrade to the original R2-ROC algorithm. As our numerical results will show in the next section, the original R2-ROC suffices for the steady-state Stokes and Navier-Stokes equations. That is, $2N - 1$ points for N -dimensional RB spaces produce stable surrogate solutions for those equations. We also note that γ represents the allowable oscillation and should be set by the practitioners for their specific problems. We take the following piecewise function in this paper

$$\gamma = \begin{cases} 5 & n < 6, \\ 2 & n \geq 6. \end{cases}$$

Algorithm 3 Adaptive enrichment algorithm

- 1: **Input:** control parameters γ , n_{adap}^{max} , n_{adap} and p_{adap} , error estimators Δ_{n-1} and Δ_n , full residual r_{n-1} , residual-based collocation set X_r^{n-1} , solution-based collocation set X_s^n , current collocation set X^m , interpolatory operator matrix P_* .
 - 2: **While** $\Delta_n > \gamma \Delta_{n-1}$ & $n_{adap} < n_{adap}^{max}$
 - 3: $n_{adap} = n_{adap} + 10$.
 - 4: Select n_{adap} elements from the top p_{adap} percentile of the domain of $|r_{n-1}|$, $\mathbf{x}_+^{n-1}$, according to (20).
 - 5: Update collocation set $X_r^{n-1} = X_r^{n-1} \cup \mathbf{x}_+^{n-1}$, $X^m = X_s^n \cup X_r^{n-1}$, $m = \#\{X_s^n, X_r^{n-1}\}$, and the operator matrix $P_* = [P_*; e_{i_+}^T]$, i_+ represents the index of $\mathbf{x}_+^{n-1}$ in X^N .
 - 6: Solve the reduced problem for $\mathbf{c}_n(\boldsymbol{\mu}, t_k)$ by eq. (14). Obtain an updated error estimator Δ_n and a new $\boldsymbol{\mu}^{n+1}$ by eq. (18).
 - 7: **End While**
 - 8: **Output:** X_r^{n-1} , X^m , P_* , Δ_n , $\boldsymbol{\mu}^{n+1}$.
-

Algorithm 4 R2-ROC algorithm with adaptive enrichment for time dependent problems

- 1: **Input:** control parameters γ , n_{adap}^{max} , and p_{adap} .
 - 2: Choose $\boldsymbol{\mu}^1$, and set $k_{\boldsymbol{\mu}^1} = 1$. $t_{\boldsymbol{\mu}^1}^{k_{\boldsymbol{\mu}^1}} = \operatorname{argmax}_{t \in \mathcal{T}_f} (\max_x \mathbf{u}(t, X^{\mathcal{N}}; \boldsymbol{\mu}^1) - \min_x \mathbf{u}(t, X^{\mathcal{N}}; \boldsymbol{\mu}^1))$ be the first temporal node. Define $\xi_1 := \mathbf{u}(t_{\boldsymbol{\mu}^1}^{k_{\boldsymbol{\mu}^1}}, X^{\mathcal{N}}; \boldsymbol{\mu}^1)$.
 - 3: Find $\mathbf{x}_*^1 = \operatorname{argmax}_{\mathbf{x} \in X^{\mathcal{N}}} \left| \sigma_{\mathbf{x}, t_{\boldsymbol{\mu}^1}^{k_{\boldsymbol{\mu}^1}}}^{\boldsymbol{\mu}^1}(\xi_1) \right|$. Define $\sigma_1(\cdot) := \sigma_{\mathbf{x}_*^1, t_{\boldsymbol{\mu}^1}^{k_{\boldsymbol{\mu}^1}}}^{\boldsymbol{\mu}^1}(\cdot)$ and calculate $\xi_1 = \xi_1 / \sigma_1(\xi_1)$.
Let $P_* = [e_{i_1}]^T$, where i_1 is the index of $\mathbf{x}_*^1$ in $X^{\mathcal{N}}$.
 - 4: Initialize $m = n = 1$, $X^m = X_s^n = \{\mathbf{x}_*^1\}$, $W_1 = \{\xi_1\}$, $n_{adap} = 0$ and $X_r^0 = \emptyset$.
 - 5: **For** $n = 2, \dots, N$
 - 6: Solve the reduced problem for $\mathbf{c}_{n-1}(\boldsymbol{\mu}, t_k)$ by eq. (14).
 - 7: Find $\boldsymbol{\mu}^n = \operatorname{argmax}_{\boldsymbol{\mu} \in \Xi_{\text{train}}} \Delta_{n-1}^{RR_t}(\boldsymbol{\mu})$, $t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}} = \operatorname{argmax}_{t \in \mathcal{T}_f} \varepsilon^{RR}(t; \boldsymbol{\mu}^n)$ and $\Delta_{n-1} = \Delta_{n-1}^{RR_t}(\boldsymbol{\mu}^n)$.
 - 8: If $\rho_{\Delta}^{n-1} > \gamma$, $n_{adap} = 0$, perform Algorithm 3 to obtain a rectified error estimator Δ_{n-1} and determine a new $\boldsymbol{\mu}^n$.
 - 9: Solve $\xi_n = \mathbf{u}(t_{\boldsymbol{\mu}^n}^{k_{\boldsymbol{\mu}^n}}, X^{\mathcal{N}}; \boldsymbol{\mu}^n)$.
 - 10: Perform Algorithms 1-2 and update $W_n = \{W_{n-1}, \xi_n\}$, $X_s^n = X_s^{n-1} \cup \mathbf{x}_*^n$, $X_r^{n-1} = X_r^{n-2} \cup \mathbf{x}_{**}^{n-1}$, $m = \#\{X_s^n, X_r^{n-1}\}$, $X^m = X_s^n \cup X_r^{n-1}$, $P_* = [P_*; (e_{i_1})^T; (e_{i_2})^T]$.
 - 11: **End For**
 - 12: **Output:** W_n , X_s^n , X_r^{n-1} , X^m , P_* .
-

4 Numerical results

In this section, we test the proposed R2-ROC method with adaptive enrichment on the three fluid problems described in Section 2. In order to evaluate the accuracy of our scheme, we measure the relative error $E(n)$ when n -dimensional RB space is used for parameters from a given testing set Ξ_{test} . It is defined as follows for the steady state problems:

$$E(n) = \max_{\boldsymbol{\mu} \in \Xi_{\text{test}}} \left\{ \frac{\|\mathbf{u}(\boldsymbol{\mu}) - \widehat{\mathbf{u}}_n(\boldsymbol{\mu})\|_{\infty}}{\|\mathbf{u}(\boldsymbol{\mu})\|_{L^{\infty}(\Omega)}} \right\} \text{ where } \|\mathbf{u}\|_{L^{\infty}(\Omega)} = \|\mathbf{u}(\boldsymbol{\mu})\|_{\infty}. \quad (21)$$

For the time dependent case, we take the L^{∞} -norm of the quantity above over a discrete time domain $\mathcal{T}_e$. That is,

$$E(n) = \max_{t \in \mathcal{T}_e} \max_{\boldsymbol{\mu} \in \Xi_{\text{test}}} \left\{ \frac{\|\mathbf{u}(\boldsymbol{\mu}, t) - \widehat{\mathbf{u}}_n(\boldsymbol{\mu}, t)\|_{\infty}}{\|\mathbf{u}(\boldsymbol{\mu})\|_{L^{\infty}(\Omega)}} \right\} \text{ where } \|\mathbf{u}\|_{L^{\infty}(\Omega)} = \max_{t \in \mathcal{T}_e} \|\mathbf{u}(\boldsymbol{\mu}, t)\|_{\infty}. \quad (22)$$

Here, $\mathcal{T}_e$ can be the full time grid $\mathcal{T}_f$ or a sub-sampled version. In addition, we measure the error of the streamline approximation by:

$$E_q(n) = |q^{\mathcal{N}} - \widehat{q}_n|, \quad (23)$$

where $q^{\mathcal{N}}$ is the truth streamline function calculated by the full FDM and $\widehat{q}_n$ is its ROC approximation using n basis. Note that the streamline q is related with the velocities u and v via $-\Delta q = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}$.

4.1 Steady-state Stokes and Navier-Stokes equations

We first consider the (linear) Stokes and (nonlinear) incompressible Navier-Stokes equations presented in Sections 2.1 and 2.2. The parameter dependence is linear in Reynolds number Re and nonlinear in the boundary control ν . The parameter domain for $\boldsymbol{\mu} = (\text{Re}, \nu)$ is $\Omega_p = [10, 2000] \times [0, 4]$ and $[10, 1000] \times [0, 2]$ respectively. We first validate our computational model by performing an accuracy test on a sequence of uniform grids for the equation with reference solution as a manufactured solution $u(x, y) = \sin(\pi x)$, $v(x, y) = -\pi \cos(\pi x)y$ for Stokes and a highly refined FDM solution for Navier-Stokes ($n_x = 256, n_y = 256$). These reference solutions are marked with a superscript $^{\text{acc}}$ and the L^2 norm of the numerical errors are listed in Table 1, where the second order convergence is observed for both equations.

Equation	n_x, n_y	$\ u^{\text{acc}} - u^{\text{fdm}}\ _2$	$\ v^{\text{acc}} - v^{\text{fdm}}\ _2$	$\ p^{\text{acc}} - p^{\text{fdm}}\ _2$
Stokes	8	0.5771	0.5771	0.0258
	16	0.0709	0.0709	0.0501
	32	0.0175	0.0175	0.0125
	64	0.0044	0.0044	0.0031
	128	0.0011	0.0011	0.000784
Navier-Stokes	8	0.3101	0.3389	1.0508
	16	0.2166	0.2108	0.4416
	32	0.0642	0.0650	0.1749
	64	0.0159	0.0159	0.0517
	128	0.0032	0.0032	0.0112

Table 1: Accuracy of the MAC scheme for the Stokes ($\text{Re} = 10$) and incompressible Navier-Stokes ($\nu = 1, \text{Re} = 20$) equations, on staggered grids with $n_x \times n_y$ cells.

For the ROC approximation, we fix the mesh with $n_x = n_y = 200$ ($n_x = 120, n_y = 120$ for Navier-Stokes) and perform the offline procedure on a training set obtained by discretizing the parameter domain Ω_p by a 50×20 (10) uniform grid for Stokes (Navier-Stokes). We then test the ROC model on 20×10 (5) uniform grids from $[10 + 2.2, 2000 (1000) - 2.2] \times [0 + 0.22, 4 (2) - 0.22]$ for Stokes (Navier-Stokes) that, in particular, are non-overlapping with the training sets. The parameter values chosen by the ROC algorithm during the training process are displayed in Figure 2 Right, where parameters selected earlier are presented by larger diamonds. It is seen that the selected parameters concentrate in a low-Reynolds-number regime but distribute relatively evenly in the range of ν for the (linear) Stokes problem. The distribution is more delicate for the Navier-Stokes equation. The error and error estimator curves are plotted in Figure 2 Left, both showcasing exponential convergence with respect to basis number n . It underscores the effectiveness of our error estimator and the resulting greedy algorithm.

Specifically, the error curves of Stokes decrease much faster and attain the accuracy of our high fidelity solution with a small number of basis. This is consistent with other results in the literature, such as a parameter dependent Stokes problem studied in [32]. The spatial domain therein is $\Omega = [0, 1] \times [0, 1]$ and the two parameters vary in $[1, 8] \times [1, 8]$. Relative

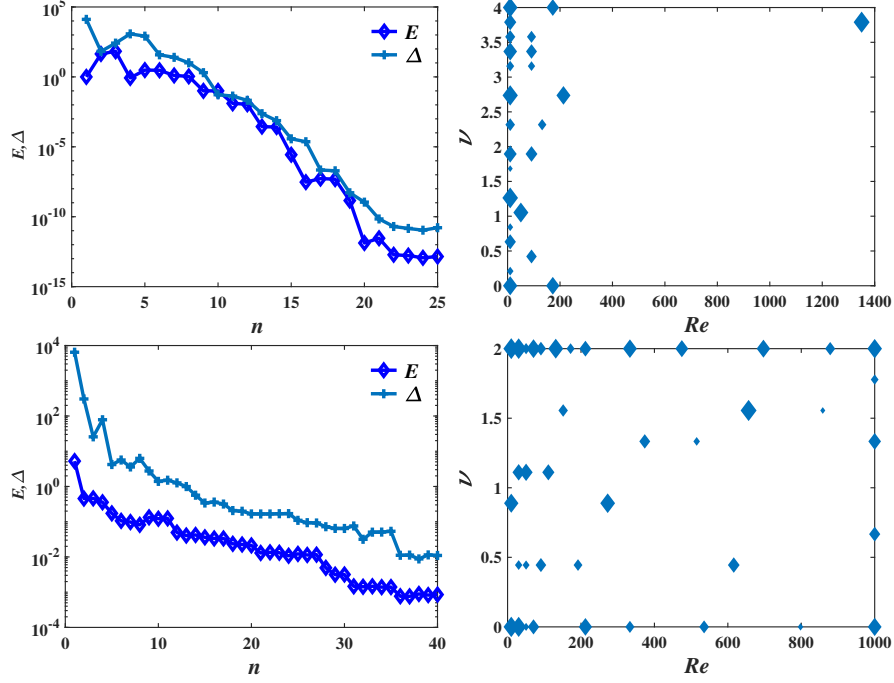


Figure 2: History of convergence for the relative error and error estimator (left) and the selected parameter values by the ROC algorithm (right). Parameter pair selected earlier has larger symbols. The first parameter is chosen randomly. Top is for Stokes and bottom Navier-Stokes.

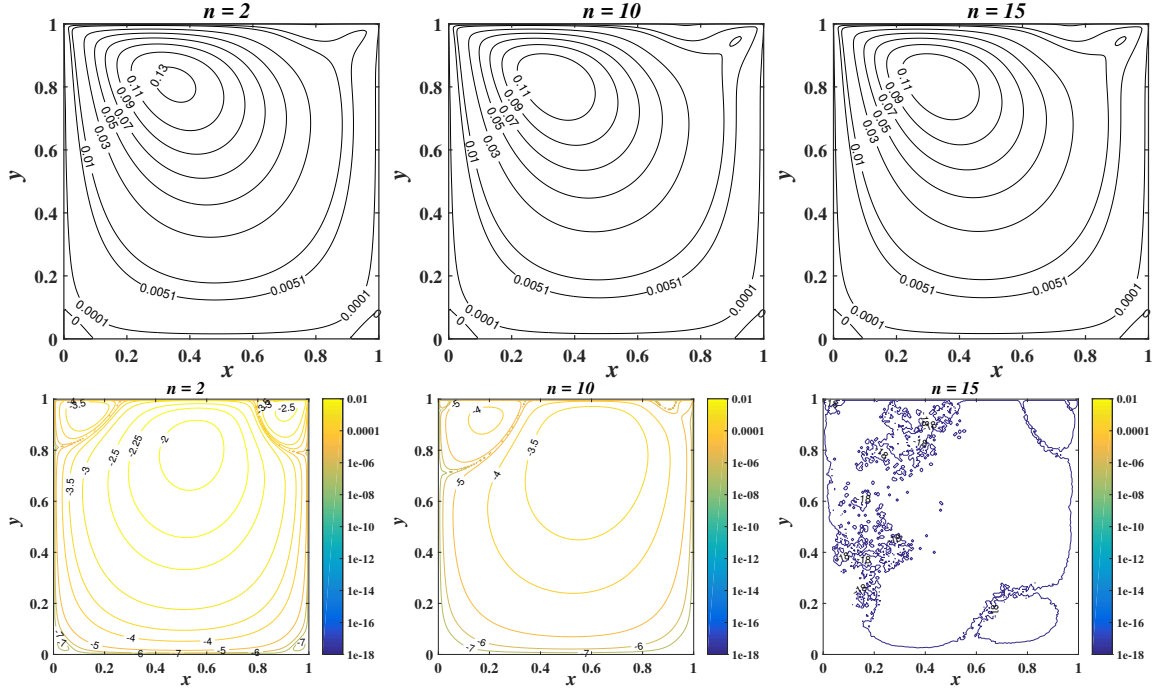


Figure 3: Streamlines of the ROC approximations with a given RB dimension n (first row), and the corresponding absolute streamline errors $E_q(n)$ in logarithmic scales (second row) for Stokes.

test errors for both pressure and velocity fields decay exponentially while stagnating at about $M = 25$ interpolatory terms with the accuracy of the underlying high fidelity solution being $10^{-5} \sim 10^{-6}$. Authors in [20] use proper generalized decomposition (PGD) method with 40 PGD terms to approximate solutions of the Stokes flow defined on a parametric domain. The final relative error stagnates at 10^{-6} for the velocity and 10^{-4} for the pressure. For the nonlinear Navier-Stokes problem with a two-dimensional parameter, the error curve decrease, as expected, slower compared to the Stokes problem above. This is also in line with other existing literature for steady-state Navier-Stokes equations with similar settings. For example, authors of [3] approximate the steady-state incompressible NS equations with low Reynolds number by the POD-Galerkin method. Their surrogate solutions for some representative parameters $Re = 30, 90$ attain $10^{-3} \sim 10^{-4}$ accuracy with about $30 \sim 40$ basis before stagnating.

We now present the streamline for both equations. For Stokes, the streamline of the ROC approximations of the FDM solution for $(Re, \nu) = (1000, 4)$ with $n = 2, 10, 15$ and their corresponding error $E_q(n)$ are in Figure 3. Those quantities for the Navier-Stokes with two sets of parameter values and $n = 2, 10, 40$ are in Figure 4. We see that for both equations, even with a bare minimum of two basis, the ROC streamline is quite accurate. As n increases, the error quickly decreases until reaching machine accuracy with 15 basis for both equations when Reynolds number is 1000, part of the training grid. We test the more challenging Navier-Stokes equation on the parameter value $(500, 2)$ that is outside of the training grid. We see that, as n increases from 2 to 40, the largest error decreases from 10^{-3} to 10^{-5} . To investigate the distribution of selected collocation points, streamlines corresponding to different Reynolds numbers together with the selected collocation points are shown in Figure 5. As n increases, more collocation points are identified. It is interesting to note that most points concentrate on the upper-right corner and some points are scattered around the upper-left corner. More interestingly, points are only selected on the edges of MAC cells, that is, the collocation points are all from the velocities. This is because the reduced basis are divergence free, thus, the residual of continuity equation is negligible. To showcase this phenomena more intuitively, we show two examples in the third and fourth columns in Figure 5.

4.2 Time-dependent Navier-Stokes equations

Finally, we consider the nonlinear time-dependent Navier-Stokes equations described in Section 2.3. We vary the two parameters one at a time. That is, $\boldsymbol{\mu} = (Re, \nu) \in \Omega_p = [10, 500] \times \{0\}$ or $\{500\} \times [0, 2]$. The high fidelity solutions are obtained with $n_x = n_y = 100, \tau = 0.002, T = 35$ adopting backward Euler as the time evolution scheme. The training set of the ROC algorithm is trained by uniformly sampling the parameter domain 21 times. The testing set consists of 5 uniformly sampled points on $[10 + 2.2, 500 - 2.2] \times \{0\}$, or $\{500\} \times [0 + 0.22, 2 - 0.22]$, which does not intersect with the training set. For the control parameter, we set $p_{adap} = 40\%, n_{adap}^{max} = 300$ (one percent of the degrees of freedom in the underlying FDM scheme). Under this problem setting, the offline process generates 70 (90) basis and 344 (835) collocation points .

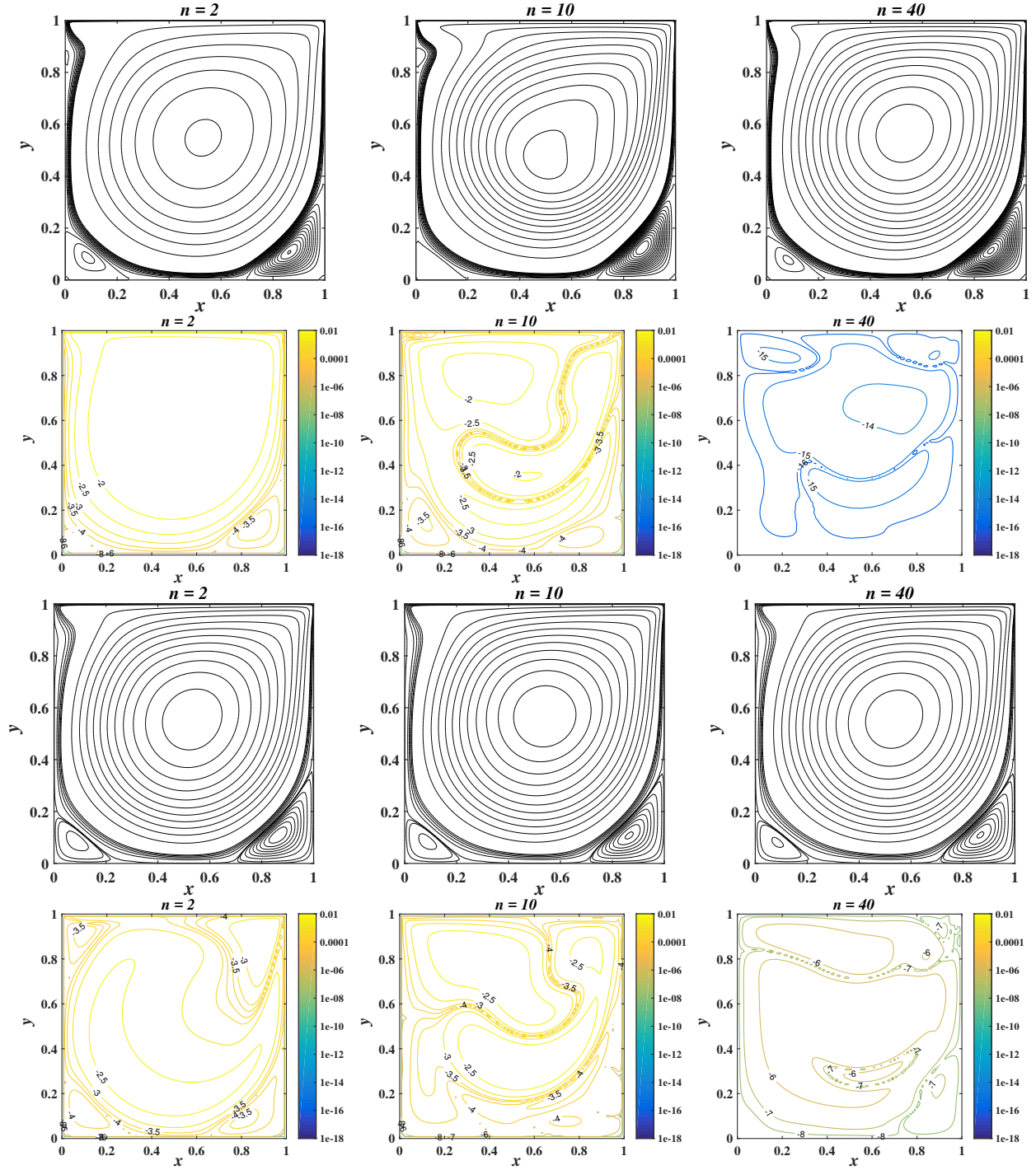


Figure 4: Streamlines of the ROC approximations with a given RB dimension n (Rows 1 and 3), and the absolute streamline errors $E_q(n)$ in logarithmic scale (Rows 2 and 4) for Navier-Stokes with parameter values $(Re, \nu) = (1000, 0)$ on the top and $(500, 2)$ at the bottom.

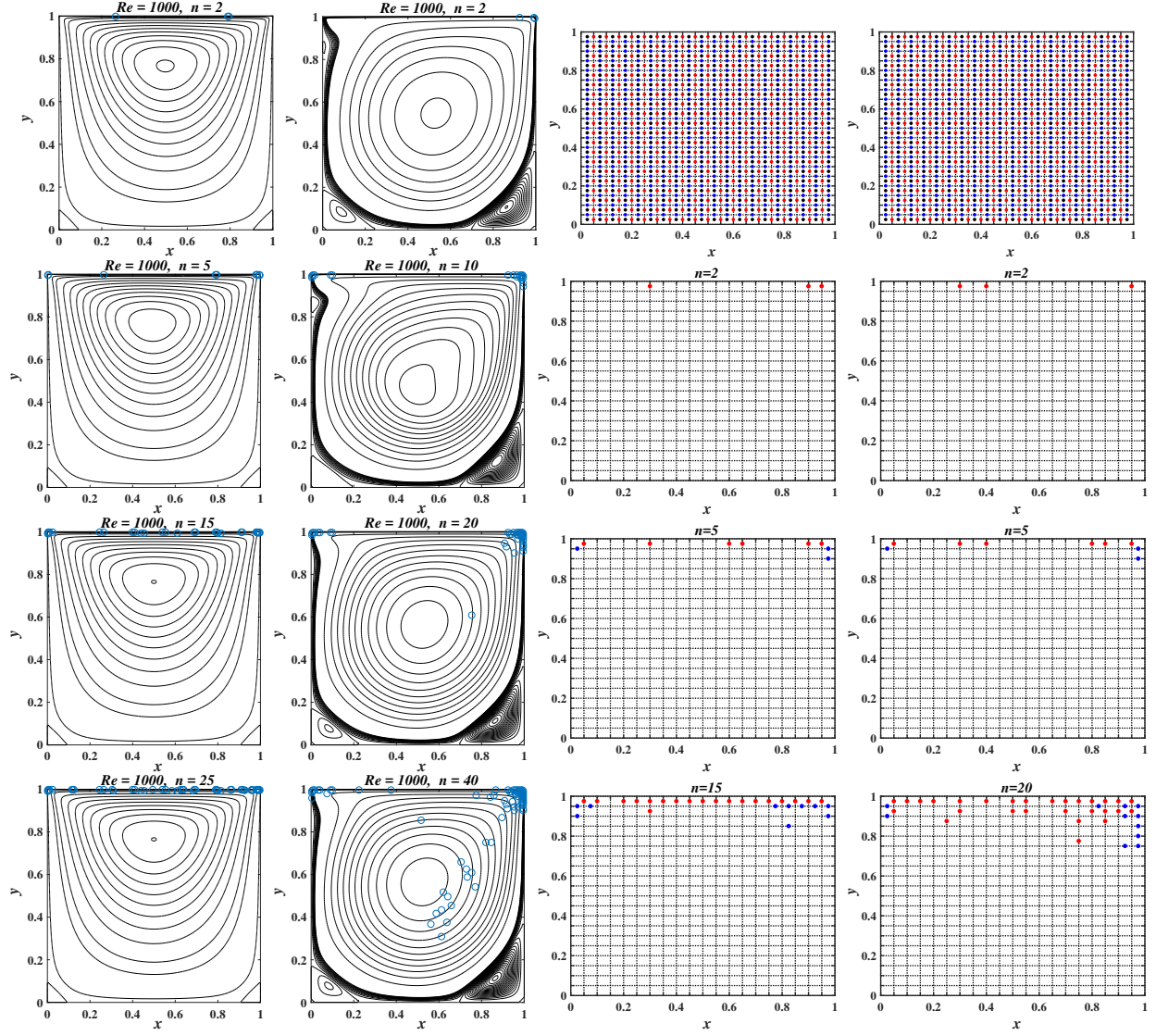


Figure 5: Stokes (Column 1) and Navier-Stokes (Column 2) streamlines of ROC approximations at $Re = 1000, \nu = 0$ with $n_x = n_y = 200$ (NS: $n_x = n_y = 120$) and various RB dimension n . Column 3 and 4: selected collocation points for Stokes and Navier-Stokes cases with training set $[10, 50] \times [0, 1]$, and $n_x = n_y = 20$. The corresponding $2n - 1$ collocation points are showed at each subplot with different n and the whole MAC points are also showed in the first subplots in Column 3 and 4, where red dots correspond to u -components and blue ones correspond to v -components.

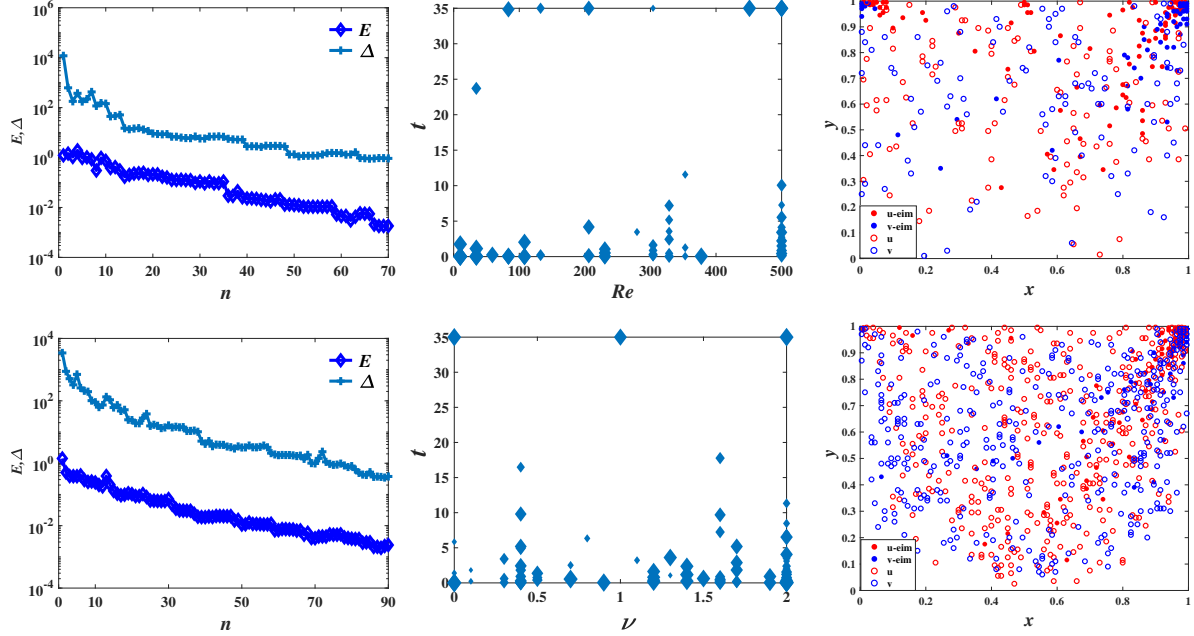


Figure 6: History of convergence for the relative error and error estimator (left), the selected parameter-time pairs (middle), and selected collocation points (right) by adaptive ROC algorithm. Top is for the training set $\Omega_p = [10, 500] \times \{0\}$, and bottom $\{500\} \times [0, 2]$.

The error and error estimator curves are shown in Figure 6 left. Compared with the Stokes and steady-state Navier-Stokes equations, these curves decrease slower due to the nonlinearity and time evolution. We note that this is consistent with the literature. For example, authors of [44] proposed a stabilized Galerkin projection based reduced basis method of unsteady incompressible NS equations with moderate Reynolds number. The relative error in L^2 norm at different time nodes for the velocity is at the $10^{-2} \sim 10^{-3}$ level, while that of pressure is around 10^{-2} . The chosen parameter-time pairs are displayed in Figure 6 center. Those with larger symbols are selected earlier. We show the selected collocation points in Figure 6 right which continue to not include any center of the MAC cells. Though much more collocation points are selected compared to the steady state NS and Stokes equations, collocation points in the upper left and upper right corners are still more likely to be chosen than those from other parts of the domain. To show the effect of the adaptive enrichment, we denote the collocation points from the adaptive enrichment with circles while the rest with filled diamonds. The distribution of the latter is similar with that in the steady state NS equations.

Streamlines and its ROC approximation errors at $(Re = 255, \nu = 0)$ ($(Re = 500, \nu = 1.78)$) with different basis numbers $n = 5, 35, 70$ ($n = 5, 45, 90$) at $T = 35$ are shown in Figure 7. As n increases from 5 to 70 (90), the largest error decreases from 10^{-2} to 10^{-4} and this is consistent with the decreasing rate of the error curve in Figure 6 left. Also, we investigate the streamlines at different time nodes $t = 0.5, 5, 10$ with different basis numbers

n in Figure 8, which indicates that our ROC algorithm can give an accurate approximation for the whole time evolution history as the basis number n is increased.

5 Conclusion

This paper proposes a novel reduced over-collocation method with adaptive enrichment for efficiently solving parametrized fluid systems. EIM/GEIM hyper reduction techniques produce two sets of collocation points whose combined cardinality is much lower than the size of the full model, while an adaptive reduced residual gives the greedy algorithm an error indicator. The resulting adaptive R2-ROC is online efficient, and stabilizes the time dependent Navier-Stokes equations with our adaptive enrichment strategy judiciously adding points on the physical domain for the reduced solver to collocate on.

References

- [1] P. Astrid. Fast reduced order modeling technique for large scale LTV systems. In *Proceedings of the 2004 American Control Conference*, volume 1, pages 762–767. IEEE, 2004.
- [2] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251, 2008.
- [3] 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.
- [4] 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. Math.*, 339(9):667–672, 2004.
- [5] A. Benaceur, V. Ehrlacher, A. Ern, and S. Meunier. A progressive reduced basis/empirical interpolation method for nonlinear parabolic problems. *SIAM J. Sci. Comput.*, 40(5):A2930–A2955, 2018.
- [6] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015.
- [7] P. Binev, A. Cohen, W. Dahmen, R. Devore, G. Petrova, and P. Wojtaszczyk. Convergence Rates for Greedy Algorithms in Reduced Basis Methods. *SIAM J. MATH. ANAL.*, pages 1457–1472, 2011.

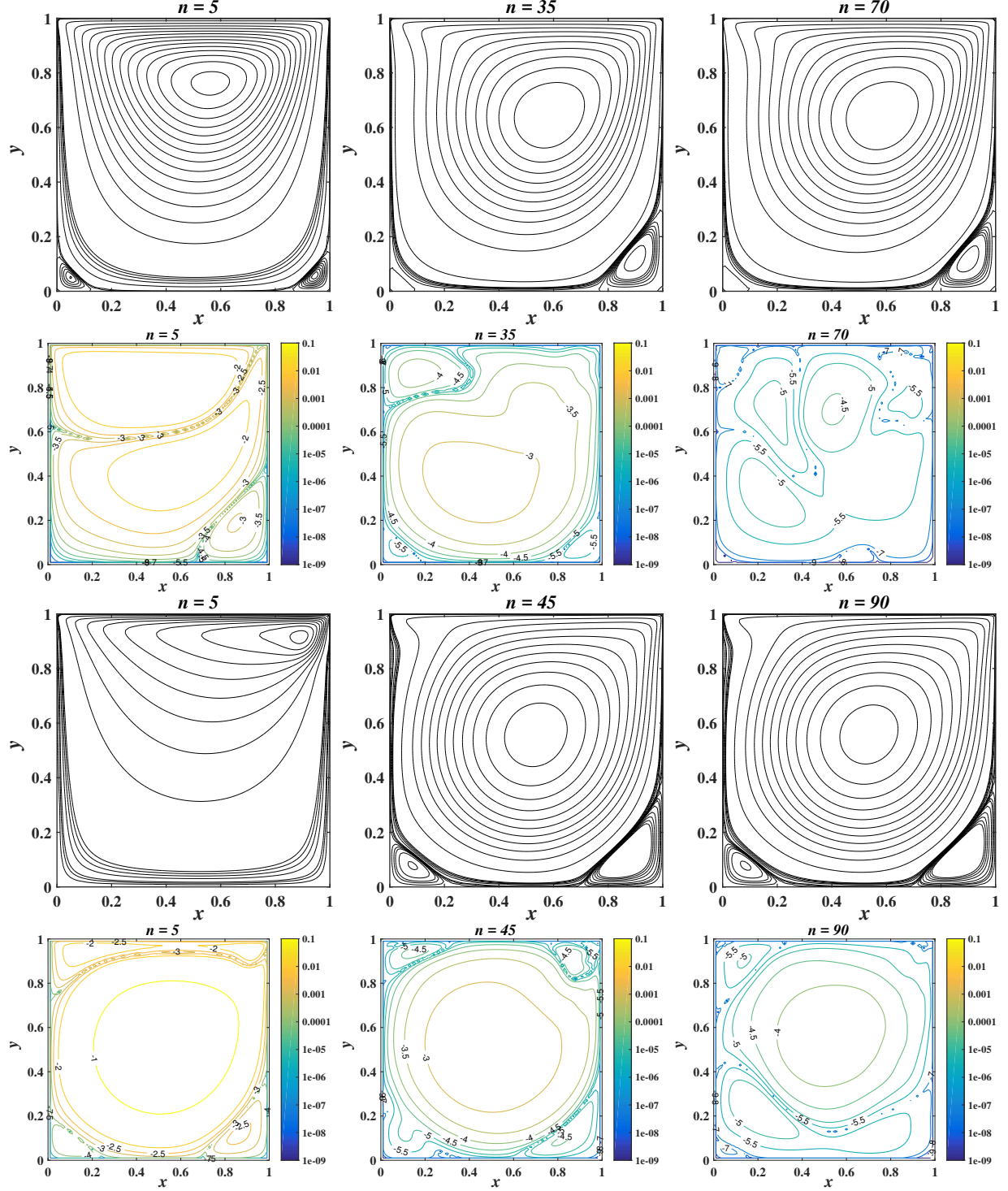


Figure 7: Streamlines of ROC approximations with the denoted RB dimensions n , and the absolute streamline errors $E_q(n)$ in logarithmic scales for time dependent Navier-Stokes equation at $T = 35$ (Rows 1,2 ($\text{Re} = 255, \nu = 0$), rows 3,4 ($\text{Re} = 500, \nu = 1.78$)).

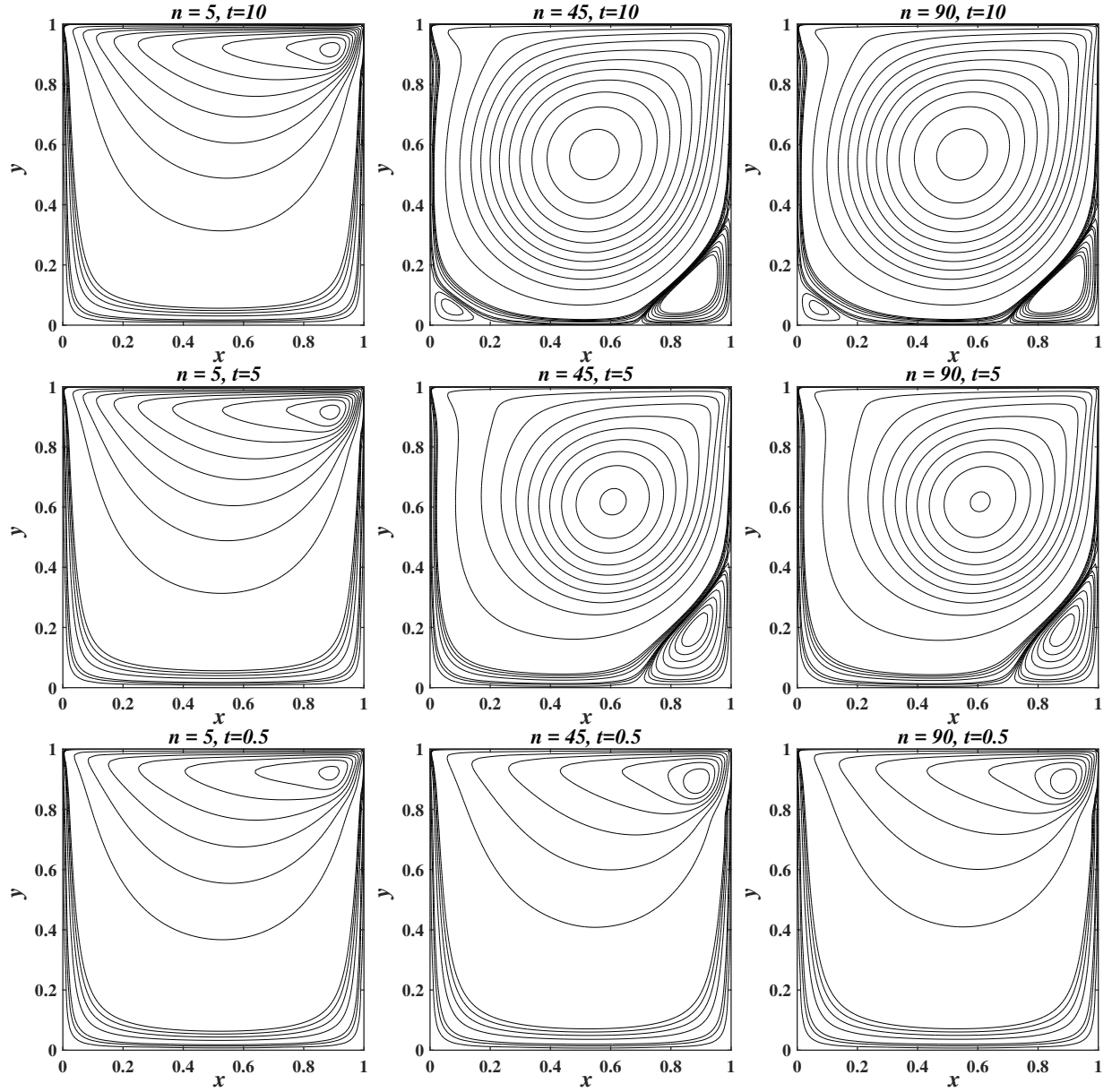


Figure 8: Streamlines of ROC approximations with the denoted RB dimensions n for time dependent Navier-Stokes equation at $t = 0.5, 5, 10$ with $(\text{Re} = 500, \nu = 1.78)$.

- [8] R. Bos, X. Bombois, and P. Van den Hof. Accelerating large-scale non-linear models for monitoring and control using spatial and temporal correlations. In *Proceedings of the 2004 American Control Conference*, volume 4, pages 3705–3710. IEEE, 2004.
- [9] 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. Numer. Methods Eng.*, 86(2):155–181, 2011.
- [10] F. Casenave, A. Ern, and T. Lelièvre. Accurate and online-efficient evaluation of the *a posteriori* error bound in the reduced basis method. *ESAIM: M2AN*, 48(1):207–229, 2014.
- [11] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737–2764, 2010.
- [12] L. Chen. Finite difference method for Stokes equations: MAC scheme. *University of California, Irvine*. <https://www.math.uci.edu/~chenlong/226/MACStokes.pdf>.
- [13] Y. Chen and S. Gottlieb. Reduced collocation methods: Reduced basis methods in the collocation framework. *J. Sci. Comput.*, 55(3):718–737, 2013.
- [14] Y. Chen, S. Gottlieb, L. Ji, and Y. Maday. An EIM-degradation free reduced basis method via over collocation and residual hyper reduction-based error estimation. *J. Comput. Phys.*, 444:110545, 2021.
- [15] Y. Chen, J. S. Hesthaven, Y. Maday, J. Rodriguez, and X. Zhu. Certified reduced basis method for electromagnetic scattering and radar cross section estimation. *Comput. Methods Appl. Mech. Eng.*, 233-236:92–108, 2012.
- [16] Y. Chen, L. Ji, A. Narayan, and Z. Xu. L1-based reduced over collocation and hyper reduction for steady state and time-dependent nonlinear equations. *J. Sci. Comput.*, 87:10, 2021.
- [17] Y. Chen, J. Jiang, and A. Narayan. A robust error estimator and a residual-free error indicator for reduced basis methods. *Computers & Mathematics with Applications*, 77:1963–1979, 2019.
- [18] A. Cohen and R. DeVore. Approximation of high-dimensional parametric {PDE}s. *Acta Numer.*, 24:1–159, 2015.
- [19] S. Deparis and G. Rozza. Reduced basis method for multi-parameter-dependent steady Navier–Stokes equations: applications to natural convection in a cavity. *J. Comput. Phys.*, 228(12):4359–4378, 2009.
- [20] P. Díez, S. Zlotnik, and A. Huerta. Generalized parametric solutions in Stokes flow. *Comput. Methods Appl. Mech. Eng.*, 326:223–240, 2017.

- [21] C. Farhat, P. Avery, T. Chapman, and J. Cortial. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *Int. J. Numer. Methods Eng.*, 98(9):625–662, 2014.
- [22] L. Fick, Y. Maday, A. T. Patera, and T. Taddei. A reduced basis technique for long-time unsteady turbulent flows. *arXiv preprint arXiv:1710.03569*, 2017.
- [23] M. A. Grepl, Y. Maday, N. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: M2AN*, 41(3):575–605, 2007.
- [24] 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.
- [25] B. Haasdonk. *Chapter 2: Reduced basis methods for parametrized PDEs—a tutorial introduction for stationary and instationary problems*, volume 15, pages 65–136. SIAM Philadelphia, 2017.
- [26] F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182–2189, 1965.
- [27] J. S. Hesthaven, G. Rozza, B. Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.
- [28] D. B. P. Huynh, D. J. Knezevic, Y. Chen, J. S. Hesthaven, and A. T. Patera. A natural-norm Successive Constraint Method for inf-sup lower bounds. *Comput. Methods Appl. Mech. Eng.*, 199:1963–1975, 2010.
- [29] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *C. R. Acad. Sci. Paris, Série I.*, 345:473–478, 2007.
- [30] T. Iliescu and Z. Wang. Variational multiscale proper orthogonal decomposition: Navier–Stokes equations. *Numer. Methods Partial Differ. Equ.*, 30(2):641–663, 2014.
- [31] K. Ito and S. S. Ravindran. A reduced-order method for simulation and control of fluid flows. *J. Comput. Phys.*, 143(2):403–425, 1998.
- [32] Y. Maday, O. Mula, A. Patera, and M. Yano. The Generalized Empirical Interpolation Method: Stability theory on Hilbert spaces with an application to the Stokes equation. *Comput. Methods Appl. Mech. Eng.*, 287:310 – 334, 2015.
- [33] Y. Maday, O. Mula, and G. Turinici. Convergence analysis of the generalized empirical interpolation method. *SIAM J. Numer. Anal.*, 54(3):1713–1731, 2016.

- [34] Y. Maday, N. C. Nguyen, A. T. Patera, and S. H. Pau. A general multipurpose interpolation procedure: The magic points. *Communications on Pure & Applied Analysis*, 8(1):383, 2009.
- [35] L. Mu and X. Ye. A simple finite element method for the Stokes equations. *Advances in Computational Mathematics*, 43(6):1305–1324, 2017.
- [36] R. A. Nicolaides. Analysis and convergence of the MAC scheme. I. the linear problem. *SIAM J. Numer. Anal.*, 29(6):1579–1591, 1992.
- [37] B. Peherstorfer, D. Butnaru, K. Willcox, and H. Bungartz. Localized discrete empirical interpolation method. *SIAM J. Sci. Comput.*, 36(1):A168–A192, 2014.
- [38] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced basis methods for partial differential equations: An introduction*, volume 92 of *Springer Series in Computational Mathematics*. Springer, 2015.
- [39] I. Rocha, F. van der Meer, and L. J. Sluys. An adaptive domain-based POD/ECM hyper-reduced modeling framework without offline training. *Computer Methods in Applied Mechanics and Engineering*, 358:112650, 2020.
- [40] 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):229–275, 2008.
- [41] G. Rozza and K. Veroy. On the stability of the reduced basis method for Stokes equations in parametrized domains. *Comput. Methods Appl. Mech. Eng.*, 196(7):1244–1260, 2007.
- [42] D. Ryckelynck. A priori hyperreduction method: An adaptive approach. *J. Comput. Phys.*, 202(1):346–366, 2005.
- [43] D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *Int. J. Numer. Methods Eng.*, 77(1):75–89, 2009.
- [44] G. Stabile and G. Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier–Stokes equations. *Computers & Fluids*, 173:273–284, 2018.
- [45] 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.
- [46] Z. Wang, I. Akhtar, J. Borggaard, and T. Iliescu. Proper orthogonal decomposition closure models for turbulent flows: A numerical comparison. *Comput. Methods Appl. Mech. Eng.*, 237:10–26, 2012.

- [47] M. Yano and A. T. Patera. An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs. *Computer Methods in Applied Mechanics and Engineering*, 344:1104–1123, 2019.