AN EVOLVE-FILTER-RELAX STABILIZED REDUCED ORDER STOCHASTIC COLLOCATION METHOD FOR THE TIME-DEPENDENT NAVIER-STOKES EQUATIONS

M. GUNZBURGER *, T. ILIESCU †, M. MOHEBUJJAMAN ‡, AND M. SCHNEIER §

Abstract. In this paper, we propose a filter-based stabilization of reduced order models (ROMs) for uncertainty quantification (UQ) of the time-dependent Navier-Stokes equations in convection-dominated regimes. We propose a novel high-order ROM differential filter and use it in conjunction with an evolve-filter-relax algorithm to attenuate the numerical oscillations of standard ROMs. We also examine how stochastic collocation methods (SCMs) can be combined with the evolve-filter-relax algorithm for efficient UQ of fluid flows. We test the new framework in the numerical simulation of a two-dimensional flow past a circular cylinder with a random viscosity that yields a random Reynolds number with mean $Re = 100$.

Key Words: Reduced order model, stochastic collocation method, stabilization method, differential filter

Mathematical Subject Classifications (2000): 65M15, 65M60

1. Introduction. Mathematical models of realistic flows inherently rely on fundamental input quantities whose actual values are imperfectly measured or sensitive to variations in other parameters. These quantities include the initial conditions, forcing functions, and model coefficients. Robust flow models must identify and properly handle parameter influence whose sensitivity or uncertainty affects model behavior. Such robust flow models are vitally important for numerical weather forecasting, ocean modeling, aeronautical design, and hemodynamics. In this paper, we consider the uncertainty quantification (UQ) of a fluid flow model in the form of the incompressible Navier-Stokes Equations (NSE).

In order to recover accurate solution statistics for fluid flow models, several challenges must be tackled: (i) These flow models must be realized over an ensemble of parameters at a high spatial resolution. For many flows individual runs can take on the order of hours, which will often make this problem computationally intractable; (ii) Realistic flows generally occur in the convection-dominated regime, for which standard numerical methods usually yield numerical oscillations; and (iii) Legacy codes are often used in realistic fluid flow applications in engineering and geophysics.

A popular approach for dealing with issue (i) is stochastic collocation methods (SCMs) [2, 13, 22, 37, 49, 52]. These methods use global polynomial approximations, taking advantage of the regularity of the solution map with respect to the random variables. Compared to other sampling based UQ methods, such as the Monte Carlo algorithm, SCMs will require far fewer realizations in order to provide accurate results. A prominent feature of these schemes is that they are independent of the PDE solver so they may easily be combined with any legacy code.

A recent approach for cutting down on the computational cost of UQ schemes is reduced order models (ROMs). These methods seek a linear space which approximates the solution manifold well. At the discrete level, this results in the need to solve orders of magnitude smaller linear systems than those required by full order models.

In order to devise an efficient UQ algorithm to address these challenges, we propose combining SCMs with a new *modular* stabilized ROM scheme, i.e., stabilization that is independent of the solver. A major advantage of this approach is that it can be easily used in conjunction with legacy flow solvers. Additionally, the modular ROM stabilization significantly decreases the computational cost of the flow solver and is able to tackle flows in the convection-dominated regime.

Our framework has several novel components: To our knowledge, SCM-ROM for the time-dependent NSE is new. SCM-ROM has been previously used only for the steady-state NSE [14]. In fact, standard (i.e., without a ROM component) SCM investigations of parabolic PDEs are rather scarce: SCM have been used for the heat equation [61], the convection-diffusion equation [18] (see also [53] for a reduced basis method approach), and the time-dependent NSE [54].

The second novel component of our framework is the modular filter-based stabilization of the SCM-ROM. Specifically, at each timestep, given the flow variables at the current timestep (u^n, p^n), we use the following *evolve-filter-relax (EFR)* algorithm to find the flow variables at the new timestep: (1) evolve the flow variables one timestep using the given CFD code and call these new variables $(\tilde{u}^n, \tilde{p}^n)$; (2) use a spatial filter F to filter the flow variables obtained in step (1): $(u^{n+1}, p^{n+1}) = (F(\tilde{u}^n), F(\tilde{p}^n))$; (3) relax the numerical approximation obtained in (2). Using the spatial filter in step (2) decreases the numerical oscillations of the flow variables and, therefore, increases the numerical stability of the algorithm. The role of the relaxation step (3) is to control the amount of numerical dissipation introduced in step (2). The main advantage of the EFR algorithm over other stabilization algorithms is that it is *modular*, i.e., one can use the EFR algorithm *without modifying the CFD solver*. Thus, the EFR algorithm can be easily used in legacy codes: One can simply start with a legacy CFD code and, with minimal coding effort, can produce a code that can tackle moderate to high Reynolds number flows. Its modular character and simplicity have made the EFR algorithm extremely appealing for the numerical simulation of convection-dominated flows with standard methods, e.g., spectral methods [8, 9, 10, 23, 38], spectral element methods [17, 35], and finite element methods [15, 29]. However, we emphasize that the EFR algorithm has never been used in conjunction with ROMs; only an EF algorithm (without relaxation) has been used in [57]. In this paper, we build on the initial efforts made in [57] and propose two new developments: (i) we add a *relaxation* step to the EF-ROM introduced in [57]; and (ii) we propose a novel *higher-order ROM differential filter*.

The new stabilized SCM-ROM framework proposed above is an *efficient, practical solution for the UQ of realistic, convection-dominated fluid flows with legacy codes*. The new SCM-ROM tackles the three challenges mentioned in the beginning of this section: (i) It is computationally efficient; the computational cost of the ROM component is orders of magnitude lower than the cost of a brute-force DNS. (ii) Its filter-based stabilization allows the numerical simulation of realistic, convection-dominated flows. (iii) Both the SCM and ROM stabilization components are independent of the PDE solver, which allows the new framework to be successfully used in conjunction with legacy codes. We note that legacy codes are often used in the DNS of realistic flows and are consistently becoming more common in the ROM community (see, e.g.,

RBniCS [24] and pyMOR [33]).

The rest of the paper is organized as follows: In Section 2, we introduce the problem setting and present a brief overview of SCMs. In Section 3, we briefly present ROMs for fluid flows. In Section 4, we propose stabilized SCM-ROMs for the numerical investigation of UQ in the time-dependent NSE in the convection-dominated regime. In Section 5, we present numerical results for the new stabilized SCM-ROM for the two-dimensional (2D) flow past a circular cylinder with a random viscosity that yields a random Reynolds number with mean $Re = 100$. Finally, we draw conclusions in Section 6.

2. SCMs for Fluid Flows.

2.1. Problem Setting. Let $D \subset \mathbb{R}^s$, $s = 2, 3$, be an open regular domain with boundary ∂D and let $[0, T]$ be a given time interval. Denote by $(\Omega, \mathcal{F}, P)$ a complete probability space, with Ω the set of outcomes, $\mathcal{F} \subset 2^\Omega$ the σ-algebra of events, and $P : F \to [0, 1]$ a probability measure. The viscosity $\nu(x, \omega)$ is modeled as a random field, with $\omega \in \Omega$. Using no-slip boundary conditions, we consider the stochastic time-dependent Navier-Stokes equations:

$$\begin{cases} u_t + u \cdot \nabla u - \nu(x, \omega) \Delta u + \nabla p = f(x, t) & \forall (x, t) \in D \times (0, T] \\ \nabla \cdot u = 0 & \forall (x, t) \in D \times (0, T] \\ u = 0 & \forall (x, t) \in \partial D \times (0, T] \\ u(x, 0, \omega) = u_0(x) & \forall x \in D. \end{cases} \tag{2.1}$$

Here u and p denote the velocity and pressure of the flow, respectively, and f is the body force.

This problem can be put into a corresponding weak formulation. Let $(\cdot, \cdot)$ denote the $L^2(D)$ inner product. Then, defining the Hilbert spaces for velocity $X = H_0^1(D)$, pressure $Q = L_0^2(D)$, and stochastic space $W = L_P^2(\Omega)$, the weak formulation of (2.1) can be written as: Find $u \in X \otimes W$ and $p \in Q \otimes W$ which, for almost all $t \in (0, T]$, satisfy

$$\begin{cases} \mathbb{E}[(u_t, v)] + \mathbb{E}[(u \cdot \nabla u, v)] + \mathbb{E}[\nu(\nabla u, \nabla v)] - \mathbb{E}[(p, \nabla \cdot v)] = \mathbb{E}[(f, v)] & \forall v \in X \otimes W \\ \mathbb{E}[(\nabla \cdot u, q)] = 0 & \forall q \in Q \otimes W \\ u(x, 0, \omega) = u_0(x). \end{cases} \tag{2.2}$$

In a number of settings it is valid to assume that the randomness can be approximated well by a finite number of random variables. Letting $y = (y_1, \ldots, y_d) \in \Gamma \subset \mathbb{R}^d$ be a finite $d \in \mathbb{N}$ dimensional vector distributed according to a joint probability density function $\rho(y)$ in some parameter space $\Gamma = \prod_{\ell=1}^d \Gamma_\ell$, the random field $\nu(x, \omega)$ can then be written in terms of the finite random variable as $\nu(x, y)$. Denoting by $Y = L_\rho^2(\Gamma)$ the space of square integrable functions on Γ with respect to the weight $\rho(y)$, the weak formulation we will consider throughout the rest of this paper is: Find

$u \in X \otimes Y$ and $p \in Q \otimes Y$ which, for almost all $t \in (0, T]$, satisfy

$$
\begin{cases}
\int_\Gamma \int_D (u_t, v)\rho(y)dy + \int_\Gamma \int_D (u \cdot \nabla u, v)\rho(y)dy + \int_\Gamma \int_D \nu(x, y)(\nabla u, \nabla v)\rho(y)dy \\
\qquad - \int_\Gamma \int_D (p, \nabla \cdot v)\rho(y)dy = \int_\Gamma \int_D (f, v)\rho(y)dy \quad \forall v \in X \otimes Y \\
\qquad \int_\Gamma \int_D (\nabla \cdot u, q)\rho(y)dy = 0 \qquad\qquad\qquad \forall q \in Q \otimes Y \\
\qquad\qquad u(x, 0, \omega) = u_0(x).
\end{cases}
\tag{2.3}
$$

In order to ensure an efficient ROM scheme, we make an assumption of affine dependence on the random variable for the viscosity, i.e.,

$$
\nu(x, y) = \nu_0(x) + \sum_{\ell=1}^{d} \nu_\ell(x) y_\ell.
\tag{2.4}
$$

In the case that this assumption does not hold, empirical interpolation schemes [11] have been devised to provide approximations that do fulfill this assumption.

2.2. Stochastic Collocation Methods (SCMs). SCMs, which take advantage of the regularity of the solution map with respect to the random variables, have recently been developed for the UQ of quantities of interest (QOI) of stochastic PDE solutions, $\psi(u)$. Some common choices for ψ in the case of fluids include the drag, lift, and energy. The goal of SCMs is to obtain statistical information about this QOI, i.e., the expectation

$$
\mathbb{E}[\psi(u)] = \int_\Gamma \psi(u, y)\rho(y)dy.
\tag{2.5}
$$

Two of the most popular SCMs are the discrete least squares collocation method and the sparse grid algorithm [13, 31, 37]. In either of these schemes, for a given time t_n and a set of sample points $y_{sc} = \{y^j\}_{j=1}^{N_{sc}} \subset \Gamma$, the PDE is solved using some spatial approximation, giving the solution denoted by $u(x, t_n, y^j)$. Then, given a basis $\{\phi_\ell\}_{\ell=1}^{p}$ for the space $L_\rho^2(\Gamma)$, a discrete approximation of the form

$$
u^{sc}(x, t_n, y) := \sum_{\ell=1}^{p} c_\ell(x, t_n)\phi_\ell(y)
\tag{2.6}
$$

is constructed with the coefficients $c_\ell(x, t_n)$.

There is a wide range of options for both the approximation points y_{sc} and polynomial basis $\{\phi_\ell\}_{\ell=1}^{p}$. A common choice for the basis functions will be polynomials which are orthogonal with respect to the PDF ρ. The sample points will depend upon the SCM used. For discrete least squares algorithms, common approaches are to take random samples with respect to ρ. When a sparse grid approach is taken, (2.6) becomes an interpolant. Some commonly used interpolations points include nested rules, such as Leja and Clenshaw Curtis points. An advantage of these interpolation schemes is that they have associated quadrature weights $\{w^j\}_{j=1}^{N_{sc}}$. In this case, an approximation to (2.5) will be given as

$$
\mathbb{E}[\psi(u)] = \int_\Gamma \psi(u, y)\rho(y)dy \approx \sum_{j=1}^{N_{sc}} w^j \psi(u, y^j).
\tag{2.7}
$$

2.3. SCMs for Fluid Flows. To our knowledge, there are relatively few investigations of SCMs for the NSE: In [14], SCMs were combined with a reduced basis method and applied to the steady NSE. In [45], SCMs were applied to the time-dependent NSE and fast convergence was demonstrated for short term time evolutions. In [55], the long term behavior of SCMs for time-dependent NSE was studied using a polynomial chaos method for random frequency stochastic processes. Finally, the numerical investigation in [55] was generalized in [54] to a wider range of stochastic processes; theoretical foundations for the numerical results were also proven in [54].

3. ROMs for Fluid Flows. In this section, the standard Galerkin ROM for fluid flows is presented succinctly.

3.1. ROM Basis. As a ROM basis, we use the proper orthogonal decomposition (POD) [25, 36, 47], which we briefly describe next. We emphasize that the new framework could be extended to other ROM bases, e.g., the dynamic mode decomposition (DMD) [28, 41, 46]. The POD starts with the snapshots $\{u_h^1, \ldots, u_h^N\}$, which are numerical approximations of (2.1) at N different time instances. The finite element (FE) solutions of (2.1) are considered as snapshots in this section. We emphasize, however, that other numerical methods can be used instead. The POD seeks a low-dimensional basis that approximates the snapshots optimally with respect to a certain norm. In this paper, the commonly used L^2-norm will be chosen. The solution of the minimization problem is equivalent to the solution of the eigenvalue problem $\mathbb{A}\mathbb{A}^\top M_h \varphi_j = \lambda_j \varphi_j$, $j = 1, \ldots, \mathcal{N}$, where φ_j and λ_j denote the vector of the FE coefficients of the POD basis functions and the POD eigenvalues, respectively, $\mathbb{A}$ denotes the snapshot matrix, whose columns correspond to the FE coefficients of the snapshots, M_h denotes the FE mass matrix, and $\mathcal{N}$ is the dimension of the FE space X^h. The eigenvalues are real and non-negative, so they can be ordered as follows: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_R > \lambda_{R+1} = \ldots = \lambda_\mathcal{N} = 0$. The POD basis consists of the normalized functions $\{\varphi_j\}_{j=1}^r$, which correspond to the first $r \leq \mathcal{N}$ largest eigenvalues. Thus, the POD space is defined as $X^r := \text{span}\{\varphi_1, \ldots, \varphi_r\}$.

3.2. Galerkin ROM (G-ROM) for Fluid Flows. To develop the standard Galerkin ROM for fluid flows, we start by considering the POD approximation of the velocity

$$u_r(x, t, y) \equiv \sum_{j=1}^{r} a_j(t, y)\varphi_j(x), \qquad (3.1)$$

where $\{a_j(t, y)\}_{j=1}^r$ are the sought time-varying coefficients that represent the POD-Galerkin trajectories. Since the POD basis functions are simply linear combinations of the snapshots, we assume that each POD basis function is weakly divergence-free and the compressibility constraint is automatically satisfied (see, however, [24, 39] for alternatives). By using the POD basis in a Galerkin approximation of the NSE, the standard *Galerkin ROM (G-ROM)* is obtained: $\forall i = 1, \ldots, r$,

$$\left(\frac{\partial u_r(y)}{\partial t}, \varphi_i \right) + \nu(y) \left(\nabla u_r(y), \nabla \varphi_i \right) + \left((u_r(y) \cdot \nabla) u_r(y), \varphi_i \right) = 0. \qquad (3.2)$$

The G-ROM (3.2) yields the following autonomous dynamical system for the vector of time coefficients, $a(t)$:

$$\dot{a} = Aa + a^\top B a, \qquad (3.3)$$

where A and B correspond to the linear and quadratic terms in the numerical discretization of the NSE (2.1), respectively. The finite dimensional system (3.3) can be written componentwise as follows: For all $i = 1, \dots, r$,

$$\dot{a}_i(t, y) = \sum_{m=1}^{r} A_{im} a_m(t, y) + \sum_{m=1}^{r} \sum_{n=1}^{r} B_{imn} a_n(t, y) a_m(t, y), \qquad (3.4)$$

where $A_{im} = -\nu(x, y) \left(\nabla \varphi_m, \nabla \varphi_i \right)$ and $B_{imn} = -\left(\varphi_m \cdot \nabla \varphi_n, \varphi_i \right).$

3.3. Stabilized ROMs for Convection-Dominated Flows. The numerical discretization of convection-dominated flows with standard Galerkin methods (e.g., FEs) generally displays spurious numerical oscillations [27]. Thus, stabilized numerical discretizations are used instead [27].

Since G-ROM (3.2) is a Galerkin method, it also yields numerical oscillations when used in the numerical simulation of convection-dominated flows [25]. Thus, many stabilized ROMs have been proposed over the years; see, e.g., [3, 4, 5, 6, 16, 21, 40, 48, 57] for recent work on this topic. Among these stabilization methods, we mention the filter based stabilization proposed in [57] (see also [21, 26, 42, 58] for related work), in which spatial filtering is used to smooth flow variables and alleviate the G-ROM's numerical instability. In the next section, we propose a new filter-based stabilization for convection-dominated flows.

4. Evolve-Filter-Relax SCM-ROM. In this section, we propose a new stabilized SCM-ROM for the numerical investigation of UQ in convection-dominated flows. In Section 4.1, we first review the current spatial ROM filters and then propose a novel ROM spatial filter: The higher-order ROM differential filter. In Section 4.2, we use the higher-order ROM differential filter to develop a new stabilized ROM, the evolve-filter-relax ROM. Finally, in Section 4.3, we combine the new evolve-filter-relax ROM with the SCM to obtain a stabilized SCM-ROM for convection-dominated flows.

4.1. ROM Spatial Filters. To build the evolve-filter-relax ROM in Section 4.2, we use explicit ROM spatial filtering to smooth the flow variables and increase the numerical stability of the models. Thus, we need to address the following problem: Given the current ROM approximation at timestep n

$$u_r^n = \sum_{j=1}^{r} a_j^n \varphi_j, \qquad (4.1)$$

define the filtered ROM approximation at timestep n, $\overline{u_r^n}$.

In this section, we present the ROM differential filter [57, 60] (Section 4.1.1) and propose a *novel higher-order ROM differential filter* (Section 4.1.2); see, e.g., [59] for alternative ROM spatial filters.

4.1.1. Differential Filter (DF). For a given $u_r^n \in X^r$ and a given radius δ, the *differential filter (DF)* computes $\overline{u_r^n} \in X^r$ as follows:

$$\left(\left(I - \delta^2 \Delta \right) \overline{u_r^n}, \varphi_j \right) = (u_r^n, \varphi_j), \quad \forall j = 1, \dots r. \qquad (4.2)$$

We emphasize that the DF uses an *explicit length scale* δ (i.e., the radius of the filter). Differential filters have been used in the simulation of convection-dominated flows with standard numerical methods [19, 20]. In reduced order modeling, the DF was used in [21, 26, 42, 57, 60].

The DF (4.2) has several appealing properties: (i) It acts as a spatial filter, since it eliminates the small scales (i.e., high frequencies) from the input. Indeed, the DF (4.2) uses an elliptic operator to smooth the input variable. (ii) The DF has a low computational overhead, since it amounts to either solving a linear system with a very small $r \times r$ matrix that is precomputed or precomputing and storing the filtered ROM modes. (iii) The DF preserves incompressibility in the NSE, since it is a linear operator.

There are two types of DF discretization that are currently in use: A ROM version and an FE version [56]. In our numerical investigation, we use the ROM discretization of the DF; we emphasize, however, that the new stabilized ROM that we propose can also be used with the FE discretization of the DF. The ROM discretization of the DF is defined as follows: Let δ be the radius of the ROM-DF. For all $u_r \in X^r$, find $\overline{u_r}^r \in X^r$ such that

$$-\delta^2 \, \Delta \overline{u_r^n}^r + \overline{u_r^n}^r = u_r^n \qquad \text{in } \Omega \tag{4.3}$$

$$\overline{u_r^n}^r = 0 \qquad \text{on } \partial\Omega \,, \tag{4.4}$$

which yields the following linear system:

$$\left(I + \delta^2 \, S_r\right) \overline{a_r^n}^r = a_r^n \,, \tag{4.5}$$

where $I \in \mathbb{R}^{r \times r}$ is the identity matrix, $S_r \in \mathbb{R}^{r \times r}$ is the ROM stiffness matrix with entries $(S_r)_{ij} = (\nabla \varphi_j, \nabla \varphi_i)$, $\overline{a_r^n}^r \in \mathbb{R}^r$ is the vector of ROM coefficients of the output filtered variable $\overline{u_r^n}^r$, and $a_r^n \in \mathbb{R}^r$ is the vector of ROM coefficients of the input variable u_r^n. Note that we exploited the fact that the ROM mass matrix $(M_r)_{ij} = (\varphi_j, \varphi_i) = I$ by the $L^2(\Omega)$ orthonormality of the POD basis.

We emphasize that the computational cost of the ROM-DF is much lower than the computational cost of the FE-DF, since the former yields a very small $r \times r$ linear system, whereas the latter yields a much larger linear system.

4.1.2. Higher-Order Differential Filter (HODF). In this section, we propose a novel *higher-order ROM differential filter (HODF)*.

The need for higher-order filters has been emphasized in [23] for spectral methods, in [35] for SEM, and in [44] for finite difference methods. For example, the left panel of Figure 1 in [35] shows that the transfer function of the DF does not have a very sharp decay, whereas the higher-order filters in the right panel of Figure 1 do have a sharp decay that approaches the decay of the sharp cut-off filter (see also Figures. 4-6 in [44]).

In this section, we extend to ROM arena the high-order SEM filters proposed in [35].

HODF – Version 1:. The easiest way to enforce a sharp decay of the transfer function of the DF is to apply repeatedly the DF filter (see left panel of Figure 1 in [35]). This amounts to replacing (4.5) with the following:

$$\left(I + \delta^2 \, S_r\right)^m \overline{a_r^n}^r = a_r^n \,, \tag{4.6}$$

where m is the order of the HODF. The major criticism in [35] was that the transfer function of the HODF (4.6) does not have a sharp decay (see left panel of Figure 1 in [35]).

HODF – Version 2:. To enforce an even sharper decay of the transfer function of the HODF (4.6), we propose the following:

$$\left(I + \delta^2 \, S_r^m\right) \overline{a_r^n}^r = a_r^n \,, \tag{4.7}$$

where m is the order of the HODF. This effectively amounts to replacing the Laplacian Δ in the standard DF (4.2) with an m-th order Laplacian Δ^m.

The transfer function of the HODF (4.7) is significantly sharper than the transfer function of the HODF (4.6) (compare left and right panels of Figure 1 in [35] and also Figures 3-6 in [44]). However, the authors in [35] emphasize that the condition number of S_r^m is extremely high. Thus, they propose to use a modified CG algorithm.

REMARK 4.1 (Previous Relevant Work). *The first HODF was proposed in [17] for spectral element methods. For finite difference methods, HODFs were investigated in [44]. To our knowledge, the HODFs in (4.6) and (4.7) are novel in a ROM setting.*

REMARK 4.2 (Computational Cost). *We emphasize that the computational cost of the HODF (and DF, for that matter) is negligible (at least for their ROM versions), since this filter involves only low-dimensional ($\mathcal{O}(10)$) matrices. This is in stark contrast with the SEM setting, where HODF's computational cost is prohibitively high.*

4.2. Evolve-Filter-Relax ROM (EFR-ROM). In this section, we put forth an *evolve-filter-relax ROM (EFR-ROM)* as a stabilization method for ROMs of convection dominated flows. We use the DF and HODF filters proposed in Section 4.1 as ROM spatial filters for the EFR-ROM. Since the computational cost of the DF and HODF is negligible, the computational cost of the EFR-ROM will be very low, similar to the computational cost of the G-ROM. The EFR-ROM allows one to simply start with a legacy CFD/ROM code and, with minimal coding effort, produce a ROM code that can tackle moderate to high Reynolds number flows.

The EFR algorithm has been used in conjunction with classical numerical methods, e.g., spectral methods [8, 9, 10, 38], spectral element methods [17, 35], finite difference methods [32], and finite element methods [15, 29]. To our knowledge, the EFR model has not been used in a ROM setting. Indeed, the only current evolve-filter ROM (EF-ROM) is that proposed in [57]; this EF-ROM, however, did not include a relaxation step and, as such, was overly-dissipative.

The algorithm for the new EFR-ROM reads: $\forall\, n = 1, \ldots, N$ and $\forall\, i = 1, \ldots, r$,

Evolve:

$$\left(\frac{3w_r^{n+1} - 4u_r^n + u_r^{n-1}}{2\Delta t}, \varphi_i\right) + \nu\left(\nabla w_r^{n+1}, \nabla \varphi_i\right) + \left((2u_r^n - u_r^{n-1}) \cdot \nabla w_r^{n+1}, \varphi_i\right)$$

$$= \left(f^{n+1}, \varphi_i\right) \tag{4.8}$$

Filter: Compute $\overline{w_r^{n+1}}$, $\tag{4.9}$

Relax: $u_r^{n+1} = (1 - \chi)\, w_r^{n+1} + \chi\, \overline{w_r^{n+1}}$, $\tag{4.10}$

where Δt is the timestep and $\chi \in [0, 1]$ is the relaxation parameter. The "evolve" step in the EFR-ROM (i.e., equation (4.8)) is just one step of the time discretization of the standard G-ROM (3.2). The "filter" step in the EFR-ROM (i.e., equation (4.9))

consists of filtering of the intermediate solution obtained in the "evolve" step with the DF and HODF proposed in Section 4.1. Finally, the "relax" step in the EFR-ROM (i.e., equation (4.10)) averages the unfiltered and filtered flow variables w_r^{n+1} and $\overline{w_r^{n+1}}$, respectively.

We note that a BDF2 time discretization was used in the EFR-ROM (4.8)–(4.10), but other time discretizations are possible [15].

As pointed out in [15], if $\chi = 1$ (i.e., there is no relaxation) and an explicit time-discretization is used, the EFR-ROM used with the DF reduces to the standard G-ROM plus numerical diffusion of magnitude $\frac{\delta^2}{\Delta t}$. (We note, however, that if an implicit time discretization is used or if one does not use the DF, the relaxation effect is not that clear [15]. In fact, the time-relaxation seems to be quite different from the DF; Leo Rebholz, personal communication.) Thus, to diminish the magnitude of the numerical diffusion, an extra relaxation step is generally used, i.e., $\chi < 1$ [15, 17]. The relaxation step decreases the amount of numerical diffusion and increases the accuracy; see, e.g., the numerical results in [7, 17] and the theoretical results in [15]. The scaling $\chi \sim \Delta t$ is commonly used in EFR models [15]. In [7], however, the authors advocate higher values for χ.

REMARK 4.3. *We emphasize that the EFR-ROM is fundamentally different from the preprocessing utilized to filter out the noise in the snapshot data [1] (which is a G-ROM that uses the filtered basis functions, $\overline{\varphi_i}$). The reason is that the EFR-ROM, spatial filtering is performed at each timestep, which amounts to filtering the ROM basis functions repeatedly. Since $\overline{\varphi} \neq \overline{\overline{\varphi}}$, we conclude that EFR-ROM is different from the preprocessing approach.*

4.3. EFR-SCM-ROM. In this section, we provide a detailed description of the combination of the EFR-ROM scheme given in Section 4.2 with the SCMs described in Section 2.2. The resulting EFR-SCM-ROM algorithm can be split into an offline and online phase. The overall goal of the new EFR-SCM-ROM is to evaluate the approximation $u_h(x, t, y)$ at a sufficiently large number of times t_n and sample points N_{sc} generated by a chosen SCM scheme so that the approximation (2.6) is sufficiently accurate. However, due to a lack of computational resources, we will only be able to obtain full order solution trajectories for a smaller number of sample points, N_{train}. In the offline phase, we generate discrete solutions (known as snapshots) for these N_{train} sample points. Using POD, we build a reduced basis from this snapshot set, which should accurately approximate the data present in the snapshot set. In the online phase, the POD basis will now be used to generate approximations for other parameter values determined by the targeted N_{sc} collocation points.

In the rest of this subsection, we provide details on each phase, as well as a full algorithm for the EFR-SCM-ROM. For other examples of SCM and ROM combinations, see [14, 12].

4.3.1. Offline Phase. In the offline phase, we are interested in constructing a reduced basis in both time and stochastic space. Let $y_{sc} = \{y^j\}_{j=1}^{N_{sc}}$ denote the N_{sc} total stochastic collocation points we will compute with in the online phase and let $y_{train} = \{y^j\}_{j=1}^{N_{train}}$ denote the training set which we will use in the offline phase (where $N_{train} << N_{sc}$ for the out of sample case but $N_{train} = N_{sc}$ for the in-sample case). Inputting y_{train} into the parameterized viscosity will result in a total of $\nu_{train} = \{\nu_k\}_{k=1}^{N_{train}}$ viscosities. For a positive integer N', we then let $0 = t_0 < t_1 < \cdots < t_m < \cdots < t_{N'} = T_{train}$ denote a uniform partition of the time interval $[0, T_{train}]$, with $T_{train} \leq T$. Denoting by $\vec{u}_S^{k,m}$ a finite dimensional solution vector to the NSE

(i.e., FE solution) at time t_m for viscosity ν_k, we construct the snapshot matrix

$$\mathbb{A} = \left(\vec{u}_S^{1,0}, \vec{u}_S^{1,1}, \ldots, \vec{u}_S^{1,N_T}, \vec{u}_S^{2,0}, \vec{u}_S^{2,1}, \ldots, \vec{u}_S^{2,N_T}, \ldots, \vec{u}_S^{N_{train},0}, \vec{u}_S^{N_{train},1}, \ldots, \vec{u}_S^{N_{train},N'} \right).$$

Solving the constrained minimization problem outlined in Section 3.1, we then construct the reduced basis $\{\varphi_j\}_{j=1}^r$.

4.3.2. Online Phase. In the online phase, we compute the approximation (2.6) using the targeted sample points $\{y^j\}_{j=1}^{N_{sc}}$. Inputting y_{train} into the parameterized viscosity will result in a total of $\nu_{sc} = \{\nu_k\}_{k=1}^{N_{sc}}$ viscosities. We now consider the time interval $0 = t_0 < t_1 < \cdots < t_m < \cdots < t_N = T$. Fixing the parameters δ and χ for the filter and relaxation step respectively, we compute solutions to the NSE at each timestep $u_{r,k}^{n+1}$ using the EFR scheme for each individual viscosity. This reduces to solving at each timestep $\forall\, k = 1, \ldots, N_{sc}$ and $\forall\, i = 1, \ldots, r$,

Evolve:

$$\left(\frac{3w_{r,k}^{n+1} - 4u_{r,k}^n + u_{r,k}^{n-1}}{2\Delta t}, \varphi_i \right) + \nu_k \left(\nabla w_{r,k}^{n+1}, \nabla \varphi_i \right) + \left((2u_{r,k}^n - u_{r,k}^{n-1}) \cdot \nabla w_{r,k}^{n+1}, \varphi_i \right)$$

$$= \left(f^{n+1}, \varphi_i \right) \quad (4.11)$$

Filter: Compute $\overline{w_{r,k}^{n+1}}$, $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ (4.12)

Relax: $u_{r,k}^{n+1} = (1 - \chi)\, w_{r,k}^{n+1} + \chi\, \overline{w_{r,k}^{n+1}}$, $\qquad\qquad\qquad\qquad\qquad$ (4.13)

For each solution, we then compute the corresponding quantity of interest $\psi(u_{r,k}^{n+1})$. Applying the quadrature formula (2.7) gives an approximation to $\mathbb{E}[\psi(u^{n+1})]$. A full outline of the EFR-SCM-ROM scheme is given in Algorithm 1.

5. Numerical Results. In this section, we perform a numerical investigation of the new EFR-SCM-ROM equipped with the DF (4.2) and the HODF (4.6). Specifically, we investigate whether the new EFR-ROM is more accurate than the standard G-ROM (3.2) in the SCM setting. As a benchmark for our numerical investigation, we use the FE approximation. Thus, we compare four models, which we denote as follows: (i) EFR-ROM, which is EFR-SCM-ROM with the DF (4.2); (ii) HODF-ROM-m, which is EFR-SCM-ROM with the HODF and; (iii) G-ROM, which is SCM-ROM (3.2); and (iv) DNS, which, with an abuse of notation, is the FE approximation used to generate the snapshots. For the HODF-ROM-m model, we investigate both HODF versions proposed in Section 4.1.2: version 1 (i.e., equation (4.6)) and version 2 (i.e., equation (4.7)). We compare the four models (i)–(iv) in the numerical investigation of a 2D flow past a circular cylinder with a 5D random viscosity that yields a random Reynolds number with mean $Re = 100$. To compare the four models, we use the accuracy of the mean energy at a time $t = t^n$, which is defined as the weighted average of $\frac{1}{2}\|u(x, t^n, y^j)\|^2$ for all sample points, where $\|\cdot\|$ denotes the norm induced by the $L^2(D)$ inner product defined in Section 2.1.

The SCM used throughout this section is a Clenshaw Curtis sparse grid generated via the software package TASMANIAN [50, 51].

Algorithm 1 EFR-SCM-ROM

1: **procedure** O(F)FLINE construction
2:     **Initialization:** Mesh, FE functions, T_{train}, N_{train}, u_h^0, snapshot matrix $\mathbb{A}$, $\{y^j\}_{j=1}^{N_{train}}$, $\{\nu_k\}_{k=1}^{N_{train}}$
3:     **for** $n = 0 \ldots N'$ **do**
4:         **for** $k = 1 \ldots N_{train}$ **do**
5:             Compute $u_{h,k}^{n+1}$
6:             Store: $\mathbb{A} = [\mathbb{A}, u_{h,k}^{n+1}]$
7:         **end for**
8:     **end for**
9:     Solve for the reduced basis $\{\varphi_j\}_{j=1}^r : \mathbb{A}\mathbb{A}^\top M_h \varphi_j = \lambda_j \varphi_j$ $j = 1, \ldots, r$
10: **end procedure**
11: **procedure** O(N)LINE Construction
12:     **Initialization:** T, N_{sc}, u_r^0, $\{y^j\}_{j=1}^{N_{sc}}$, $\{w^j\}_{j=1}^{N_{sc}}$, $\{\nu_k\}_{k=1}^{N_{sc}}$, δ, χ
13:     **Pre-compute:** S_r, A_r, B_r
14:     **for** $n = 0 \ldots N$ **do**
15:         **for** $k = 1 \ldots N_{sc}$ **do**
16:             Evolve: $w_{r,k}^{n+1}$ according to (4.11)
17:             Filter: $\overline{w_r^{n+1}}$ according to (4.12)
18:             Relax: u_r^{n+1} according to (4.13)
19:             Calculate $\psi(u_{r,k}^{n+1})$
20:         **end for**
21:         Estimate $\mathbb{E}[\psi(u^{n+1})] \approx \sum_{k=1}^{N_{sc}} w^k \psi(u_{r,k}^{n+1})$
22:     **end for**
23: **end procedure**

The rest of this section is organized as follows: In Section 5.1, we describe the test problem setup. In Section 5.2, we present details of the DNS discretization and in Section 5.3 we outline the ROM construction. In Section 5.4, we define the 5D random variable viscosity we will use for the numerical investigations. In Section 5.5 we conduct a numerical investigation for the algorithm with optimally tuned fixed parameters. Finally, in Section 5.6 we perform a sensitivity analysis of our scheme with respect to the possible tuning parameters.

5.1. Test Problem Setup. The domain is a 2.2×0.41 rectangular channel with a radius $= 0.05$ cylinder, centered at $(0.2, 0.2)$, see Figure 5.1. No slip boundary conditions are prescribed for the walls and on the cylinder, and the inflow and outflow profiles are given by [34, 59] $u_1(0, y, t) = u_1(2.2, y, t) = \frac{6}{0.41^2} y(0.41 - y)$, $u_2(0, y, t) = u_2(2.2, y, t) = 0$. The kinematic viscosity $\nu(x, \omega)$ is random, there is no forcing, and the flow starts from rest.

5.2. DNS. In this section, we present details of the algorithm outlined in Section 4.3 that is used to construct the benchmark DNS results. To generate the DNS data, we run the time-dependent NSE (2.1) up to the simulation time $t = 10$ for each of the collocation points $y_i \in \Omega$ and use the SCM to compute the quantity of interest ψ, i.e., the mean energy. For the spatial discretization of the NSE, we use the $(X_h, Q_h) = (P_2, P_1^{disc})$ Scott-Vogelius (SV) element, which satisfies the mass conservation point-wise, on a barycenter refined regular triangular mesh providing 35020 velocity and 25974 pressure degrees of freedom. For the temporal discretization, we
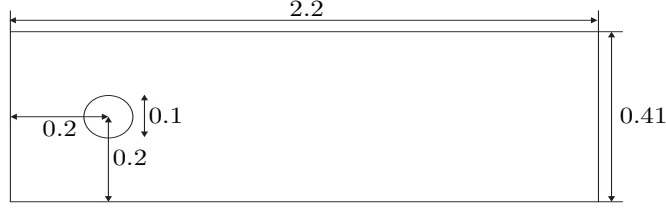
Fig. 5.1: The physical model of the domain for the channel flow around a cylinder.

use a linearized BDF2 temporal discretization with a timestep size $\Delta t = 0.002$. On the first timestep, we use a backward Euler temporal discretization to generate the second initial approximations required by the BDF2 scheme. The time discretization reads: For $n = 1, 2, \cdots$, find $(u_h^{n+1}, p_h^{n+1}) \in (X_h, Q_h)$ satisfying for every $(v_h, q_h) \in (X_h, Q_h)$,

$$\left(\frac{3u_h^{n+1} - 4u_h^n + u_h^{n-1}}{2\Delta t}, v_h \right) + ((2u_h^n - u_h^{n-1}) \cdot \nabla u_h^{n+1}, v_h) - (p_h^{n+1}, \nabla \cdot v_h)$$
$$+ \nu(x, y_i)(\nabla u_h^{n+1}, \nabla v_h) = 0,$$
$$(\nabla \cdot u_h, q_h) = 0.$$

5.3. ROMs. In this section, we present details of the EFR-SCM-ROM algorithm presented in Section 4.3, which is used to construct the DF-ROM and HODF-ROM-m. In Section 5.3.1, we present details of the offline phase and in Section 5.3.2, we present details of the online phase. In Section 5.3.3, we comment on the computational efficiency of the DF-ROM and HODF-ROM-m.

5.3.1. Offline Phase. In the offline phase of the EFR-SCM-ROM algorithm, we construct the DF-ROM and HODF-ROM-m (see Section 4.3).

To this end, we generate a number of N_{train} collocation points $\{y_i\}_{i=1}^{N_{train}}$, where $N_{train} << N_{sc}$. We run our simulations for all these N_{train} collocation points up to $t = 10$. We collect a total of 2500 snapshots at each timestep from $t = 5$ to $t = 10$ for each of the collocation points. We put together the resulting $2500 \times N_{train}$ snapshots in a snapshot matrix, which we then use to generate the ROM basis functions (modes), as described in Section 3.1. The snapshot average is used as the first mode, which satisfies the boundary conditions. We subtract the first mode from the snapshots, and solve an eigenvalue problem to find the dominant modes of these adjusted snapshots. These modes are used to generate the ROM-mass, ROM-stiffness, and ROM-nonlinear matrices.

5.3.2. Online Phase. In the online phase of the EFR-SCM-ROM algorithm, we run the DF-ROM and HODF-ROM-m that were built in Section 5.3.1. To avoid any significant temporal error in the ROM discretization, in all our tests we choose the same timestep as that used in the DNS, i.e., $\Delta t = 0.002$. As discussed in Section 4.3, we use the same number of collocation points and corresponding weights as those used in the DNS phase. For each sample point, the ROM initial condition at $t = 7$ is the L^2 projection into the ROM space of the corresponding FE solution at $t = 7$. We also use the backward Euler method to construct the ROM initial condition at $t = 7.002$. For each collocation point, we apply the DF-ROM and HODF-ROM-m algorithms described in Section 4.3. We start the ROMs from the initial time $t = 7.002$ (now

called $t = 0$) and we run them up to $t = 10$. Since the ROM initial condition was taken at $t = 7$, we plot all the ROM results on the time interval $t \in [7, 17]$. For each collocation point, we compute the energy as our ROM quantity of interest.

5.3.3. Computational Efficiency. As described in Section 5.2, to generate the DNS data we integrate in time a system of equations with 35020 velocity and 25974 pressure degrees of freedom. We need to integrate this high-dimensional system of equations for each collocation point: $N_{sc} = 801$ for the variable 5D random viscosity case (see Section 5.4).

As described in Section 5.3, to generate the DF-ROM and HODF-ROM-m (i.e., the ROM offline phase in which the DF-ROM and HODF-ROM-m are built), we integrate in time the same high-dimensional system as that used to generate the DNS data. We emphasize, however, that in this ROM offline phase we use significantly fewer collocation points: $N_{train} = 11$ (instead of $N_{sc} = 801$) for the variable 5D random viscosity case (see Section 5.4).

In conclusion, both the DF-ROM and the HODF-ROM-m provide considerable computational savings compared to the brute force DNS.

5.4. Variable 5D Random Viscosity. In this section, we consider the time-dependent incompressible stochastic NSE (2.1) with a random viscosity $\nu(x, y)$, where $y = (y_1, y_2, \cdots, y_d) \in \Gamma \subset \mathbb{R}^d$ is a higher-dimensional random variable, $\mathbb{E}[\nu](x) = \frac{c}{1000}$ for a suitable $c > 0$, $\mathbb{C}ov[\nu](x, x') = \frac{1}{1000^2} exp\left(-\frac{(x-x')^2}{l^2}\right)$, and l is the correlation length. This random field can be represented by the Karhunen-Loéve expansion

$$\nu(x, y) = \frac{1}{1000}\left(c + \left(\frac{\sqrt{\pi}l}{2}\right)^{\frac{1}{2}} y_1(\omega) + \sum_{j=1}^{q} \sqrt{\xi_j}\left(\sin\left(\frac{j\pi x_1}{2.2}\right)\sin\left(\frac{j\pi x_2}{0.41}\right) y_{2j}(\omega)\right.\right.$$
$$\left.\left. + \cos\left(\frac{j\pi x_1}{2.2}\right)\cos\left(\frac{j\pi x_2}{0.41}\right) y_{2j+1}(\omega)\right),\right.$$
$$(5.1)$$

in which the infinite series is truncated up to the first q terms. The uncorrelated random variables y_j have zero mean and unit variance, and the eigenvalues are equal to

$$\sqrt{\xi_j} = (\sqrt{\pi}l)^{\frac{1}{2}} exp\left(-\frac{(j\pi l)^2}{8}\right).$$

For our test problem, we consider the random variables $y_j(\omega) \in [-\sqrt{3}, \sqrt{3}]$, the correlation length $l = 0.01$, $d = 5$, $c = 1$, and $q = 2$.

To generate the DNS data, we use $N_{sc} = 801$ collocation points and their corresponding weights.

We run the DNS for each of the collocation points and we compute the energy (which is shown in Figure 5.2) as our DNS quantity of interest.

To generate the ROM data, we use $N_{train} = 11$ collocation points. The singular values of the snapshot matrix are plotted in a semi-log scale in Figure 5.3. We run the ROMs for each of the collocation points and we compute the mean energy.

5.5. Numerical Experiments I: Fixed Parameters. For the first set of numerical experiments, we demonstrate the efficacy of the algorithm for fixed parameter
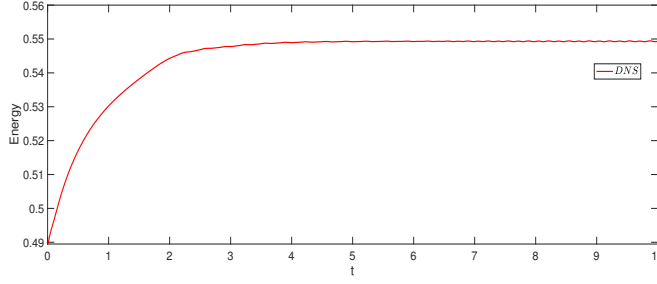
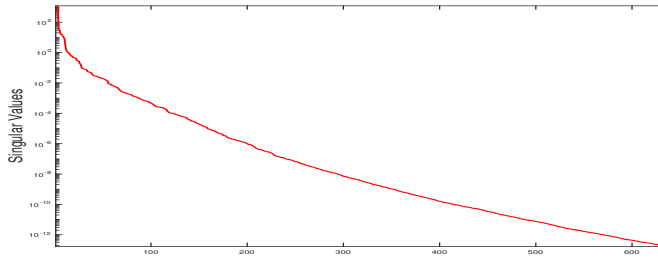Fig. 5.2: Variable 5D random viscosity: Plot of DNS mean energy vs. time.

Fig. 5.3: Plot of singular values vs. index in a semi-log scale.

values. The goal is to show the potential of the algorithm when hyper-parameters can be sufficiently tuned. We consider the case of $\delta = 9.6 \times 10^{-3}$, $r = 4$, $\chi = \Delta t$, and version 1 of HODF-ROM-m with $m = 4$. In Figure 5.4 we plot the time evolution of the mean energy for the in-sample parameter case and in Figure 5.5 we plot the out of sample case. These two plots show that version 1 of the HODF-ROM-m with $m = 4$ is dramatically more accurate than the DF-ROM and G-ROM for both the in-sample case and the out of sample case. This conclusion is also supported by the results in Table 5.1 and Table 5.2.
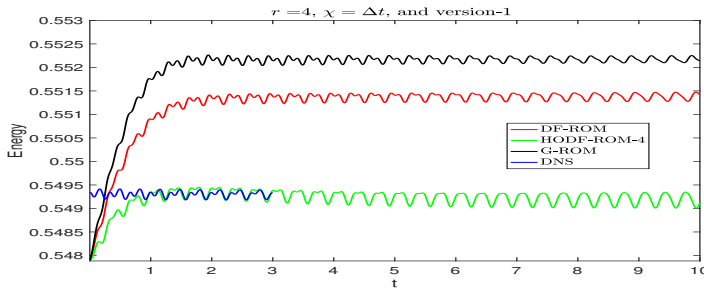

Fig. 5.4: Variable 5D random viscosity: Plot of mean energy vs. time for the in-sample case with $\delta = 9.6 \times 10^{-3}$.
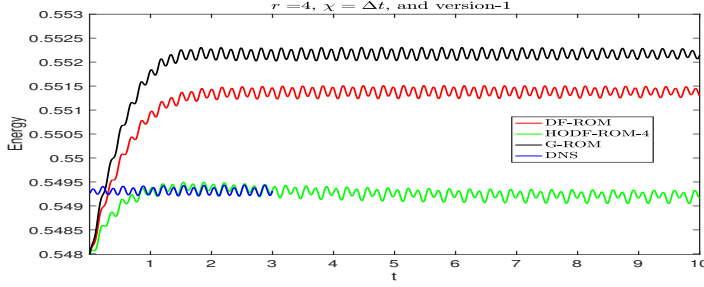
Fig. 5.5: Variable 5D random viscosity: Plot of mean energy vs. time for the out of sample case with $\delta = 9.6 \times 10^{-3}$.

5.6. Numerical Experiments II: Sensitivity Analysis. It was shown in the previous section that the DF-ROM and HODF-ROM-m schemes outperformed the G-ROM scheme after the parameters had been properly tuned. In practice, selection of hyper-parameters is a difficult task, and if done improperly, it can actually result in worse performance of the algorithm. To obtain a better understanding of the robustness of the EFR-SCM-ROM scheme, we study in this section the sensitivity of the scheme with respect to the relaxation parameter χ, filtering parameters δ and m, as well as the ROM dimension r. In order to evaluate the sensitivity, we again consider the error in the energy time series with respect to the first and second order moments.

In Table 5.1 and 5.2, we consider the in-sample and out of sample parameter cases with fixed $\delta = 9.60 \times 10^{-3}$ and $\chi = \Delta t$, but with varying r and m. In terms of the first-order moment, we see that the HODF-ROM-m version 1 with $m = 4$ performs the best. In particular, for $r = 4$ and $r = 5$, the error for the HODF-ROM-m version 1 with $m = 4$ is *more than an order of magnitude lower* than the errors for the DF-ROM and G-ROM. We also notice that, overall, the DF-ROM consistently performs better than the G-ROM. Finally, we see that the HODF-ROM-m version 2 always performs worse than the other methods. In terms of the second-order moments, we see that the DF-ROM or the G-ROM perform the best.

In Table 5.3 and 5.4, we repeat the same process, except that we consider a larger δ value, $\delta = 1.18 \times 10^{-2}$. In terms of the first order moments, we see that, this time, DF-ROM performs the best. In particular, for $r = 4$ and $r = 5$, the error for the DF-ROM is *more than an order of magnitude lower* than the errors for the HODF-ROM-m and G-ROM. We also notice that the HODF-ROM-m version 2 always performs worse than the other methods (see Figure 5.6 and Figure 5.7). In terms of the second-order moments, we see that the DF-ROM and the G-ROM perform the best.

In Table 5.5 - 5.10, we consider the in-sample and out of sample parameter case with $r = 3, 4, 5$, $\delta = 1.18 \times 10^{-2}$, and varying values of χ. We see in this case that the DF-ROM scheme generally performs, as well as or better than the HODF-ROM-m scheme. We also note that, choosing $\chi = \Delta t$ yields the most accurate results for DF-ROM and HODF-ROM-m. Furthermore, other χ values can significantly degrade the accuracy of DF-ROM and HODF-ROM-m.

From this study, we draw the following conclusions: In many cases, DF-ROM seems to perform comparably to or better than HODF-ROM-m, especially for larger δ values. This is notable as DF-ROM has one fewer parameter to tune. Additionally,
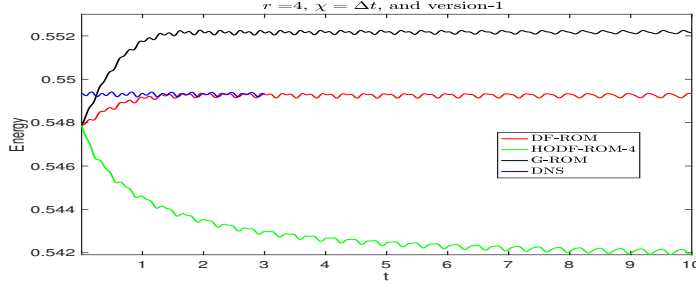
Fig. 5.6: Variable 5D random viscosity: Plot of mean energy vs. time for the in-sample case with $\delta = 1.18 \times 10^{-2}$.
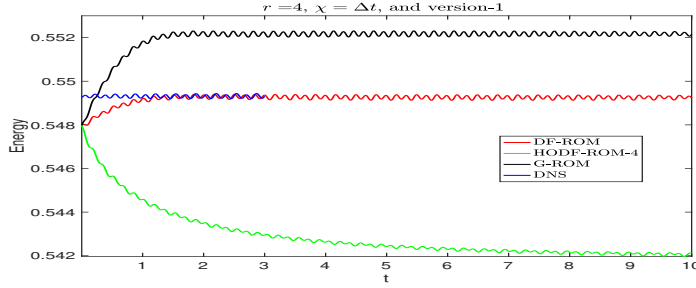

Fig. 5.7: Variable 5D random viscosity: Plot of mean energy vs. time for the out of sample case with $\delta = 1.18 \times 10^{-2}$.

version 2 of HODF-ROM-m performs poorly, even compared to the G-ROM. Finally, we also conclude that the DF-ROM and the HODF-ROM-m are sensitive to parameter changes. We emphasize that this is expected, since a similar parameter sensitivity is generally noticed in regularization methods for standard spatial discretizations, e.g., finite element methods [30, 43].

$\chi = \Delta t$	$\delta = 9.60 \times 10^{-3}$	Error in first-order moment			Error in second-order moment		
	r	3	4	5	3	4	5
Time Avg Error	G-ROM	4.60e-03	2.90e-03	2.80e-03	1.33e-05	4.05e-07	6.33e-07
	DF-ROM	3.90e-03	2.10e-03	2.00e-03	1.31e-05	1.29e-06	6.97e-07
V-1	HODF-ROM2	3.20e-03	1.30e-03	1.30e-03	1.30e-05	2.78e-06	1.92e-06
	HODF-ROM3	2.50e-03	5.77e-04	6.59e-04	1.29e-05	4.01e-06	2.97e-06
	HODF-ROM4	1.90e-03	6.74e-05	8.17e-05	1.28e-05	4.95e-06	3.81e-06
V-2	HODF-ROM2	1.68e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05
	HODF-ROM3	1.69e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05
	HODF-ROM4	1.69e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05

Table 5.1: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 9.60 \times 10^{-3}$, $\chi = \Delta t$, and in-sample parameters.

6. Conclusions. We proposed a modular filter-based stabilization of ROMs for UQ of the time-dependent NSE in convection-dominated regimes. For the ROM stabilization, we put forth a new evolve-filter-relax (EFR) algorithm to attenuate the

$\chi = \Delta t$	$\delta = 9.60 \times 10^{-3}$	Error in first-order moment			Error in second-order moment		
Time Avg Error	r	3	4	5	3	4	5
	G-ROM	4.70e-03	2.90e-03	2.80e-03	1.38e-05	8.85e-07	1.10e-06
	DF-ROM	3.90e-03	2.10e-03	2.00e-03	1.36e-05	7.77e-07	2.11e-07
V-1	HODF-ROM2	3.20e-03	1.30e-03	1.30e-03	1.35e-05	2.22e-06	1.40e-06
	HODF-ROM3	2.50e-03	5.79e-04	6.61e-04	1.34e-05	3.37e-06	2.39e-06
	HODF-ROM4	1.90e-03	6.18e-05	8.87e-05	1.33e-05	4.19e-06	3.14e-06
V-2	HODF-ROM2	1.68e-02	1.84e-02	1.84e-02	1.46e-05	1.45e-05	1.45e-05
	HODF-ROM3	1.69e-02	1.84e-02	1.85e-02	1.46e-05	1.45e-05	1.45e-05
	HODF-ROM4	1.69e-02	1.84e-02	1.85e-02	1.46e-05	1.45e-05	1.45e-05

Table 5.2: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 9.60 \times 10^{-3}$, $\chi = \Delta t$, and out of sample parameters.

$\chi = \Delta t$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	r	3	4	5	3	4	5
	G-ROM	4.60e-03	2.90e-03	2.80e-03	1.38e-05	4.05e-07	6.33e-07
	DF-ROM	4.60e-03	4.59e-05	4.98e-05	1.36e-05	1.99e-06	1.24e-06
V-1	HODF-ROM2	4.60e-03	2.70e-03	2.60e-03	1.35e-05	3.43e-06	2.48e-06
	HODF-ROM3	4.70e-03	5.10e-03	4.90e-03	1.34e-05	3.26e-06	2.50e-06
	HODF-ROM4	4.80e-03	6.90e-03	6.60e-03	1.34e-05	1.20e-06	9.64e-07
V-2	HODF-ROM2	1.68e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05
	HODF-ROM3	1.69e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05
	HODF-ROM4	1.69e-02	1.84e-02	1.84e-02	1.41e-05	1.41e-05	1.41e-05

Table 5.3: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $\chi = \Delta t$, and in-sample parameters.

$\chi = \Delta t$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	r	3	4	5	3	4	5
	G-ROM	4.70e-03	2.90e-03	2.80e-03	1.38e-05	8.85e-07	1.10e-06
	DF-ROM	4.59e-03	4.81e-05	5.19e-05	1.36e-05	1.42e-06	1.50e-07
V-1	HODF-ROM2	4.62e-03	2.74e-03	2.60e-03	1.35e-05	2.65e-06	7.28e-07
	HODF-ROM3	4.68e-03	5.05e-03	4.90e-03	1.34e-05	2.22e-06	2.82e-07
	HODF-ROM4	4.76e-03	6.87e-03	6.60e-03	1.34e-05	1.52e-07	1.26e-06
V-2	HODF-ROM2	1.68e-02	1.84e-02	1.84e-02	1.46e-05	1.45e-05	1.45e-05
	HODF-ROM3	1.69e-02	1.84e-02	1.85e-02	1.46e-05	1.45e-05	1.45e-05
	HODF-ROM4	1.69e-02	1.84e-02	1.85e-02	1.46e-05	1.45e-05	1.45e-05

Table 5.4: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $\chi = \Delta t$, and out of sample parameters.

$r = 3$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	4.59e-03	4.66e-03	4.84e-03	1.31e-05	1.30e-05	1.29e-05
V-1	HODF-ROM2	4.61e-03	4.95e-03	5.56e-03	1.30e-05	1.29e-05	1.28e-05
	HODF-ROM3	4.68e-03	5.33e-03	6.38e-03	1.30e-05	1.28e-05	1.28e-05
	HODF-ROM4	4.76e-03	5.70e-03	7.08e-03	1.29e-05	1.28e-05	1.29e-05
V-2	HODF-ROM2	1.68e-02	2.14e-02	2.32e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM3	1.69e-02	2.14e-02	2.32e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM4	1.69e-02	2.14e-02	2.32e-02	1.41e-05	1.45e-05	1.46e-05

Table 5.5: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 3$, and in-sample parameters.

numerical oscillations that standard ROMs generally display in convection-dominated flows. Furthermore, we proposed a novel high-order ROM differential filter (HODF)

$r = 4$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	4.59e-05	3.30e-03	6.62e-03	1.99e-06	3.63e-06	2.38e-06
V-1	HODF-ROM2	2.75e-03	8.53e-03	1.21e-02	3.43e-06	1.37e-06	1.11e-05
	HODF-ROM3	5.06e-03	1.13e-02	1.30e-02	3.26e-06	9.85e-06	1.32e-05
	HODF-ROM4	6.87e-03	1.21e-02	1.30e-02	1.20e-06	1.24e-05	1.35e-05
V-2	HODF-ROM2	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM3	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM4	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05

Table 5.6: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 4$, and in-sample parameters.

$r = 5$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	4.98e-05	3.18e-03	6.40e-03	1.24e-06	2.66e-06	1.90e-06
V-1	HODF-ROM2	2.64e-03	8.29e-03	1.19e-02	2.48e-06	1.17e-06	1.08e-05
	HODF-ROM3	4.87e-03	1.11e-02	1.29e-02	2.50e-06	9.40e-06	1.32e-05
	HODF-ROM4	6.64e-03	1.20e-02	1.30e-02	9.64e-07	1.23e-05	1.35e-05
V-2	HODF-ROM2	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM3	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05
	HODF-ROM4	1.84e-02	2.17e-02	2.33e-02	1.41e-05	1.45e-05	1.46e-05

Table 5.7: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 5$, and in-sample parameters.

$r = 3$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	4.60e-03	4.66e-03	4.83e-03	1.36e-05	1.35e-05	1.34e-05
V-1	HODF-ROM2	4.62e-03	4.95e-03	5.56e-03	1.35e-05	1.33e-05	1.33e-05
	HODF-ROM3	4.68e-03	5.33e-03	6.38e-03	1.34e-05	1.33e-05	1.33e-05
	HODF-ROM4	4.76e-03	5.71e-03	7.08e-03	1.34e-05	1.33e-05	1.34e-05
V-2	HODF-ROM2	1.68e-02	2.14e-02	2.32e-02	1.46e-05	1.50e-05	1.51e-05
	HODF-ROM3	1.69e-02	2.14e-02	2.32e-02	1.46e-05	1.50e-05	1.51e-05
	HODF-ROM4	1.69e-02	2.14e-02	2.32e-02	1.46e-05	1.50e-05	1.51e-05

Table 5.8: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 3$, and out of sample parameters.

$r = 4$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	4.81e-05	3.30e-03	6.60e-03	1.42e-06	2.79e-06	1.29e-06
V-1	HODF-ROM2	2.74e-03	8.60e-03	1.21e-02	2.65e-06	2.16e-06	1.13e-05
	HODF-ROM3	5.05e-03	1.13e-02	1.30e-02	2.22e-06	9.98e-06	1.37e-05
	HODF-ROM4	6.87e-03	1.21e-02	1.30e-02	1.52e-06	1.28e-05	1.40e-05
V-2	HODF-ROM2	1.84e-02	2.17e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05
	HODF-ROM3	1.84e-02	2.17e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05
	HODF-ROM4	1.84e-02	2.17e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05

Table 5.9: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 4$, and out of sample parameters.

to perform the spatial filtering in the evolve-filter-relax algorithm. For the UQ component, we used the SCM. We emphasize that the entire stabilized SCM-ROM framework that we proposed can be easily used in conjunction with legacy flow solvers. We also note that, to our knowledge, the SCM-ROM for time-dependent NSE, the EFR

$r = 5$	$\delta = 1.18 \times 10^{-2}$	Error in first-order moment			Error in second-order moment		
	χ	Δt	$2\Delta t$	$3\Delta t$	Δt	$2\Delta t$	$3\Delta t$
	DF-ROM	5.19e-05	3.18e-03	6.39e-03	7.12e-07	1.93e-06	8.84e-07
V-1	HODF-ROM2	2.64e-03	8.30e-03	1.19e-02	1.78e-06	2.03e-06	1.10e-05
	HODF-ROM3	4.86e-03	1.11e-02	1.29e-02	1.57e-06	9.54e-06	1.37e-05
	HODF-ROM4	6.63e-03	1.20e-02	1.30e-02	4.98e-08	1.26e-05	1.40e-05
V-2	HODF-ROM2	1.84e-02	2.17e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05
	HODF-ROM3	1.85e-02	2.18e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05
	HODF-ROM4	1.85e-02	2.18e-02	2.33e-02	1.45e-05	1.50e-05	1.51e-05

Table 5.10: Variable 5D random viscosity: Time average error in first and second energy moment with $\delta = 1.18 \times 10^{-2}$, $r = 5$, and out of sample parameters.

algorithm for ROMs, and the HODF are new.

We performed a numerical investigation of the new EFR-SCM-ROM equipped with the DF (4.2) and the HODF (4.6) with $m = 2, 3$, and 4. As a test problem, we considered the 2D flow past a circular cylinder with a random viscosity that yields a random Reynolds number with mean $Re = 100$. Our numerical investigation yielded the following general conclusions: For small δ values, version 1 of HODF-ROM-m (with high m values) was the most accurate. Version 2 of HODF-ROM-m performed worse than G-ROM. We also concluded that DF-ROM and HODF-ROM-m were sensitive to parameter changes. We emphasize that a similar parameter sensitivity is generally noticed in regularization methods for other spatial discretizations, e.g., finite element methods [30, 43].

These first steps in the numerical investigation of the new EFR-SCM-ROM framework are encouraging. There are, however, several research avenues that we plan to pursue for a better understanding of the capabilities and limitations of the EFR-SCM-ROM.

First, we plan to continue the comparison of the DF vs. HODF, both in a deterministic and a stochastic setting. In particular, we will investigate the effect of the two ROM spatial filters on the ROM spatial structures. To this end, in addition to the time evolution of the mean energy, we will consider alternative criteria in our numerical investigation, such as the time evolution of the ROM coefficients, a_i.

We also plan to investigate the new EFR-SCM-ROM framework in more challenging flows, such as the NSE at higher Reynolds numbers. In particular, to study the limitations of the proposed EFR-SCM-ROMs, we will gradually increase the Reynolds number until the accuracy of the ROM approximation exceeds the prescribed tolerance. Aditionally, we plan to continue the sensitivity analysis started in this paper and perform a sensitivity study of the ROM spatial filter radius δ with respect to the Reynolds number.

Lastly, we plan to examine the effect of the spatial filters on the accuracy of the SCM for long time evolutions. It is known that as the final time grows, the error from the SCM will also grow [54, 55]. Whether or not the spatial filters mitigate this effect is a subject of future work.

REFERENCES

[1] S. Aradag, S. Siegel, J. Seidel, K. Cohen, and T. McLaughlin. Filtered POD-based low-dimensional modeling of the 3D turbulent flow behind a circular cylinder. *Int. J. Num. Meth. Fluids*, 66(1):1–16, 2011.

[2] I. Babuska, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM J. Num. Anal.*, 45(3):1005–1034, 2007.

[3] M. J. Balajewicz, I. Tezaur, and E. H. Dowell. Minimal subspace rotation on the Stiefel manifold for stabilization and enhancement of projection-based reduced order models for the compressible Navier–Stokes equations. *J. Comput. Phys.*, 321:224–241, 2016.

[4] 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.

[5] M. Benosman, J. Borggaard, O. San, and B. Kramer. Learning-based robust stabilization for reduced-order models of 2D and 3D Boussinesq equations. *Appl. Math. Model.*, 49:162–181, 2017.

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

[7] L. Bertagna, A. Quaini, and A. Veneziani. Deconvolution-based nonlinear filtering for incompressible flows at moderately large Reynolds numbers. *Int. J. Num. Meth. Fluids*, 81(8):463–488, 2016.

[8] J. P. Boyd. Two comments on filtering (artificial viscosity) for Chebyshev and Legendre spectral and spectral element methods: preserving boundary conditions and interpretation of the filter as a diffusion. *J. Comput. Phys.*, 143(1):283–288, 1998.

[9] J. P. Boyd. *Chebyshev and Fourier spectral methods*. Dover Publications, 2001.

[10] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral methods in fluid dynamics*. Springer Series in Computational Physics. Springer-Verlag, New York, 1988.

[11] S. Chaturantabut and D. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737–2764, 2010.

[12] P. Chen, A. Quarteroni, and G. Rozza. Comparison between reduced basis and stochastic collocation methods for elliptic problems. *J. Sci. Comput.*, 59(1):187–216, 2014.

[13] A. Chkifa, A. Cohen, G. Migliorati, F. Nobile, and R. Tempone. Discrete least squares polynomial approximation with random evaluations- application to parametric and stochastic elliptic pdes. *ESAIM: Math. Model. Num. Anal.*, 2015.

[14] H. C. Elman and Q. Liao. Reduced basis collocation methods for partial differential equations with random coefficients. *SIAM/ASA J. Uncertain. Quantif.*, 1(1):192–217, 2013.

[15] V. J. Ervin, W. J. Layton, and M. Neda. Numerical analysis of filter-based stabilization for evolution equations. *SIAM J. Num. Anal.*, 50(5):2307–2335, 2012.

[16] L. Fick, Y. Maday, A. T. Patera, and T. Taddei. A stabilized POD model for turbulent flows over a range of Reynolds numbers: Optimal parameter sampling and constrained projection. *J. Comp. Phys.*, 371:214–243, 2018.

[17] P. F. Fischer and J. Mullen. Filter-based stabilization of spectral element methods. *C. R. Acad. Sci. Paris Sér. I Math.*, 332(3):265–270, 2001.

[18] B. Ganapathysubramanian and N. Zabaras. Sparse grid collocation schemes for stochastic natural convection problems. *J. Comput. Phys.*, 225(1):652 – 685, 2007.

[19] M. Germano. Differential filters for the large eddy numerical simulation of turbulent flows. *Phys. Fluids*, 29(6):1755–1757, 1986.

[20] M. Germano. Differential filters of elliptic type. *Phys. Fluids*, 29(6):1757–1758, 1986.

[21] M. Gunzburger, T. Iliescu, and M. Schneier. A Leray regularized ensemble-proper orthogonal decomposition method for parameterized convection-dominated flows. *IMA Journal of Numerical Analysis*, 01 2019.

[22] M. D. Gunzburger, C. G. Webster, and G. Zhang. Stochastic finite element methods for partial differential equations with random input data. *Acta Numerica*, 23:521–650, 2014.

[23] J. Hesthaven and R. Kirby. Filtering in Legendre spectral methods. *Math. Comput.*, 77(263):1425–1452, 2008.

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

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

[26] T. Iliescu, H. Liu, and X. Xie. Regularized reduced order models for a stochastic Burgers equation. *Int. J. Numer. Anal. Mod.*, 15(4–5):594–607, 2018.

[27] V. John. *Finite element methods for incompressible flow problems*. Springer, 2016.

[28] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor. *Dynamic Mode Decomposition:*

Data-Driven Modeling of Complex Systems. SIAM, 2016.

[29] W. Layton, N. Mays, M. Neda, and C. Trenchea. Numerical analysis of modular regularization methods for the BDF2 time discretization of the Navier-Stokes equations. *ESAIM: Math. Model. Num.. Anal.*, 48(3):765–793, 2014.

[30] W. J. Layton and L. G. Rebholz. *Approximate Deconvolution Models of Turbulence: Analysis, Phenomenology and Numerical Analysis*, volume 2042. Springer Berlin Heidelberg, 2012.

[31] K. Lee, K. Carlberg, and H. C. Elman. Stochastic least-squares Petrov–Galerkin method for parameterized linear systems. *SIAM/ASA J. Uncertain. Quantif.*, 6(1):374–396, 2018.

[32] J. Mathew, R. Lechner, H. Foysi, J. Sesterhenn, and R. Friedrich. An explicit filtering method for large eddy simulation of compressible flows. *Phys. Fluids*, 15(8):2279–2289, 2003.

[33] R. Milk, S. Rave, and F. Schindler. pyMOR–generic algorithms and interfaces for model order reduction. *SIAM J. Sci. Comput.*, 38(5):S194–S216, 2016.

[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*, 89:103–122, 2019.

[35] J. S. Mullen and P. F. Fischer. Filtering techniques for complex geometry fluid flows. *Commun. Numer. Meth. Engng.*, 15(1):9–18, 1999.

[36] B. R. Noack, M. Morzynski, and G. Tadmor. *Reduced-Order Modelling for Flow Control*, volume 528. Springer Verlag, 2011.

[37] F. Nobile, R. Tempone, and C. G Webster. An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Num. Anal.*, 46(5):2411–2442, 2008.

[38] R. Pasquetti and C. J. Xu. Comments on Filter-based stabilization of spectral element methods. *J. Comput. Phys.*, 182(2):646–650, 2002.

[39] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced Basis Methods for Partial Differential Equations: An Introduction*, volume 92. Springer, 2015.

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

[41] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral analysis of nonlinear flows. *J. Fluid Mech.*, 641:115–127, 2009.

[42] F. Sabetghadam and A. Jafarpour. α regularization of the POD-Galerkin dynamical systems of the Kuramoto–Sivashinsky equation. *Appl. Math. Comput.*, 218(10):6012–6026, 2012.

[43] P. Sagaut. *Large Eddy Simulation for Incompressible Flows*. Scientific Computation. Springer-Verlag, Berlin, third edition, 2006.

[44] O. San, A. E. Staples, and T. Iliescu. A posteriori analysis of low-pass spatial filters for approximate deconvolution large eddy simulations of homogeneous incompressible flows. *Int. J. Comput. Fluid Dyn.*, 29(1):40–66, 2015.

[45] S. Sankaran and A. L. Marsden. A stochastic collocation method for uncertainty quantification and propagation in cardiovascular simulations. *J. Biomech. Eng.*, 133 3:031001, 2011.

[46] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.*, 656:5–28, 2010.

[47] L. Sirovich. Turbulence and the dynamics of coherent structures. Parts I–III. *Quart. Appl. Math.*, 45(3):561–590, 1987.

[48] G. Stabile and G. Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier-Stokes equations. *Comput. & Fluids*, 2018.

[49] M. Stoyanov. Hierarchy-direction selective approach for locally adaptive sparse grids. Technical Report ORNL/TM-2013/384, Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN, 2013.

[50] M. Stoyanov. User manual: Tasmanian sparse grids. Technical Report ORNL/TM-2015/596, Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN, 2015.

[51] M. Stoyanov, D. Lebrun-Grandie, J. Burkardt, and D. Munster. Tasmanian, 9 2013.

[52] M. K. Stoyanov and C. G. Webster. A dynamically adaptive sparse grids method for quasi-optimal interpolation of multidimensional functions. *Comput. Math. Appl.*, 71(11):2449–2465, 2016.

[53] D. Torlo, F. Ballarin, and G. Rozza. Stabilized weighted reduced basis methods for parametrized advection dominated problems with random inputs. *arXiv preprint, http://arxiv.org/abs/1711.11275*, 2017.

[54] H. Tran, C. Trenchea, and C. Webster. A convergence analysis of stochastic collocation method for Navier-Stokes equations with random input data. *ORNL Technical Report*, 2014.

[55] X. Wan and G. Karniadakis. Long-term behavior of polynomial chaos in stochastic flow simulations. *Comput. Meth. Appl. Mech. Eng.*, 195(41):5582 – 5596, 2006. John H. Argyris Memorial Issue. Part II.

[56] D. Wells. *Stabilization of POD-ROMs*. PhD thesis, Virginia Tech, http://vtechworks.lib.vt.

edu/bitstream/handle/10919/52960/Wells_DR_D_2015.pdf?sequence=1&isAllowed=y, 2015.

[57] D. Wells, Z. Wang, X. Xie, and T. Iliescu. An evolve-then-filter regularized reduced order model for convection-dominated flows. *Int. J. Num. Meth. Fluids*, 84:598–615, 2017.

[58] X. Xie, F. Bao, and C. G. Webster. Evolve then filter regularization for stochastic reduced order modeling. *Preprints.org*, 2018.

[59] 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.

[60] X. Xie, D. Wells, Z. Wang, and T. Iliescu. Approximate deconvolution reduced order modeling. *Comput. Methods Appl. Mech. Engrg.*, 313:512–534, 2017.

[61] G. Zhang and M. Gunzburger. Error analysis of a stochastic collocation method for parabolic partial differential equations with random input data. *SIAM J. Num. Anal.*, 50(4):1922–1940, 2012.