https://doi.org/10.1016/j.jcp.2022.111456

A STOCHASTIC GALERKIN METHOD WITH ADAPTIVE TIME-STEPPING FOR THE NAVIER–STOKES EQUATIONS[*]

BEDŘICH SOUSEDíK[†] AND RANDY PRICE[‡]

Abstract. We study the time-dependent Navier–Stokes equations in the context of stochastic finite element discretizations. Specifically, we assume that the viscosity is a random field given in the form of a generalized polynomial chaos expansion, and we use the stochastic Galerkin method to extend the methodology from [D. A. Kay et al., *SIAM J. Sci. Comput.* 32(1), pp. 111–128, 2010] into this framework. For the resulting stochastic problem, we explore the properties of the resulting stochastic solutions, and we also compare the results with that of Monte Carlo and stochastic collocation. Since the time-stepping scheme is fully implicit, we also propose strategies for efficient solution of the stochastic Galerkin linear systems using a preconditioned Krylov subspace method. The effectiveness of the stochastic Galerkin method is illustrated by numerical experiments.

Key words. uncertainty quantification, spectral stochastic finite element methods, stochastic Galerkin method, Navier–Stokes equation, preconditioning

AMS subject classifications. 35R60, 60H15, 65N22, 65N30, 65N35

1. Introduction.

Models of mathematical physics are commonly based on partial differential equations (PDEs). In this study, we focus on the most popular PDE model in fluid mechanics, which is the Navier–Stokes equation [7, 23]. We consider a stochastic version of the model: we assume that the viscosity is given by a generalized polynomial chaos (gPC) expansion, we discretize the problem using spectral stochastic finite elements see, e.g., [13, 23, 25, 39], and we wish to find the gPC expansion of the solution. The steady-state version of this problem was studied in [24, 31, 37], and our focus here is on the time-dependent counterpart. Our approach to time discretization is built on the fully implicit scheme with adaptive time-stepping strategy, which was developed for the deterministic Navier–Stokes equation by Kay et al. [21], see also [15]. We extend their scheme in the stochastic Galerkin framework, and in particular we show that the physics inspired time-stepping strategy can be also adapted to this framework. The scheme is fully implicit, and so each time step entails a solve with the stochastic Galerkin matrix. This typically leads to very large systems of linear equations, for which use of direct solvers may be prohibitive, and therefore the method could potentially be quite computationally expensive. There are other approaches to time stepping see, e.g., [1, 7, 20] which may appear more appealing. Nevertheless, finally we also show that the iterative solvers by Sousedík and Elman [37], which are based on preconditioned Krylov subspace methods, are quite effective for the implicit time discretizations of the time-dependent Navier–Stokes problem as well.

Some aspects of the gPC methods for time-dependent problems were studied in literature see, e.g., [16, 41, 42]. In particular, long-term integration was addressed by Gerritsma et al. [10], Heuveline, Schick and Song [19, 34, 36], Wilkins [38], Özen

[*] This work was supported by the U. S. National Science Foundation under grant DMS1913201. Most of the computations were performed at the Center for Computational Mathematics, University of Colorado Denver, and the support of Prof. Jan Mandel is greatly appreciated. Part of the work was completed while Randy Price was student at the University of Maryland, Baltimore County.

[†] Department of Mathematics and Statistics, University of Maryland, Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250 (sousedik@umbc.edu).

[‡] The Center for Mathematics and Artificial Intelligence and the Center for Computational Fluid Dynamics, George Mason University, Fairfax, VA 22030 (rprice25@gmu.edu).

nad Bal [29, 30], and most recently by Esquivel et al. [9], among others. Methods for flows exhibiting uncertain periodic dynamics were proposed, e.g., by Bonnaire et al. [2], Lacour et al. [22] and Schick et al. [33]. These methods typically entail time-dependent or other variants of gPC expansions that are tailored to the changing character of the solution. Nevertheless, here we use a time-independent gPC basis, which turns out to be sufficient for the transient problems considered in our numerical experiments. Therefore, all these techniques can be viewed as complementary to the present study. We also note that Elman and Su [8] proposed a low-rank stochastic Galerkin solver based on monolithic (all-at-once) time discretization of the Navier–Stokes problem, however their scheme is based on a constant timestep.

Finally, we remark on possible interpretations of the Navier–Stokes problem with stochastic viscosity. In such case, the Reynolds number defined as

$$\mathrm{Re}(\xi) = \frac{UL}{\nu(\xi)},$$

where $\nu > 0$ is the viscosity, U is the characteristic velocity and L is the characteristic length, is also stochastic. The possible interpretations of such setup are discussed by Powell and Silvester in [31]: for example, assuming fixed geometry, the stochastic viscosity is equivalent to Reynolds number being stochastic, which may correspond to a scenario when the volume of fluid moving into the channel is uncertain.

The paper is organized as follows. In Section 2 we recall the algorithm for the deterministic problem, in Section 3 we formulate the algorithm for the stochastic problem using both the stochastic Galerkin and sampling methods, in Section 4 we report results of numerical experiments and provide details about the preconditioning of the Oseen problem, and finally in Section 5 we summarize and conclude our work.

2. Algorithm for the deterministic problem.

We first recall the algorithm for the deterministic problem following Kay et al. [21]. Let $D \subset \mathbb{R}^2$ be a physical domain, and let $T > 0$ denote a stopping time. We wish to solve the time-dependent Navier–Stokes equation in $D \times [0, T]$, where $(\vec{u}, p)$ denote the fluid velocity and pressure, and $\nu \equiv \nu(x) > 0$[1] is the viscosity parameter, written as

$$\frac{\partial \vec{u}}{\partial t} = f(\nu, \vec{u}, p), \qquad f(\nu, \vec{u}, p) = \nu \nabla^2 \vec{u} - \vec{u} \cdot \nabla \vec{u} - \nabla p, \tag{2.1}$$

$$-\nabla \cdot \vec{u} = 0, \tag{2.2}$$

with boundary and initial conditions given on $\partial D = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ as

$$\vec{u} = \vec{g}, \quad \text{on } \Gamma_D \times [0, T], \tag{2.3}$$

$$\nu \nabla \vec{u} \cdot \vec{n} - p\vec{n} = \vec{0}, \quad \text{on } \Gamma_N \times [0, T], \tag{2.4}$$

$$\vec{u}(\vec{x}, 0) = \vec{u}_0(\vec{x}), \quad \text{in } D. \tag{2.5}$$

The initial velocity field is assumed to satisfy the incompressibility constraint, that is $\nabla \cdot \vec{u}_0 = 0$. We also assume that Γ_N has nonzero measure so that the pressure is uniquely specified, and to this end we will use the outflow (*do-nothing*) boundary condition. We begin by recalling the implicit trapezoid rule (TR) as

$$\vec{u}_t \approx \frac{\vec{u}^{n+1} - \vec{u}^n}{k_{n+1}} = \frac{1}{2}\left[f_{n+1} + f_n\right],$$

[1] The assumption $\nu(x) \neq \text{const}$ is the only difference from the setup in [21] in this section.

where $k_{n+1} = t_{n+1} - t_n$. Then $2\vec{u}_t \approx 2\left(\vec{u}^{n+1} - \vec{u}^n\right)/k_{n+1} = f_{n+1} + f_n$ and (2.1)–(2.2) can be written as

$$\frac{2}{k_{n+1}}\vec{u}^{n+1} - \nu\nabla^2\vec{u}^{n+1} + \vec{u}^{n+1}\cdot\nabla\vec{u}^{n+1} + \nabla p^{n+1} = \frac{2}{k_{n+1}}\vec{u}^n + \frac{\partial\vec{u}^n}{\partial t}, \tag{2.6}$$

$$-\nabla\cdot\vec{u}^{n+1} = 0. \tag{2.7}$$

The nonlinear term is linearized as $\vec{u}^{n+1}\cdot\nabla\vec{u}^{n+1} \approx \vec{w}^{n+1}\cdot\nabla\vec{u}^{n+1}$. The linearization is based on extrapolation $\left(\vec{w}^{n+1} - \vec{u}^n\right)/k_{n+1} = \left(\vec{u}^n - \vec{u}^{n-1}\right)/k_n$, from which we find

$$\vec{w}^{n+1} = \left(1 + k_{n+1}/k_n\right)\vec{u}^n - \left(k_{n+1}/k_n\right)\vec{u}^{n-1}. \tag{2.8}$$

Next, let (V_D, Q_D) denote a pair of spaces satisfying the inf-sup condition and let V_E be an extension of V_D containing velocity vectors that satisfy the Dirichlet boundary conditions [4, 7, 14]. The mixed variational formulation of (2.6)–(2.7) is: find $(\vec{u}^{n+1}, p^{n+1}) \in V_E \times Q_D$, for a given pair $(\vec{u}^n, p^n)$, such that

$$\begin{aligned}\frac{2}{k_{n+1}}\int_D \vec{u}^{n+1}\vec{v} + \int_D \nu\,\nabla\vec{u}^{n+1} : \nabla\vec{v} + \int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{u}^{n+1}\right)\vec{v} - \int_D p^{n+1}\left(\nabla\cdot\vec{v}\right) \\ = \frac{2}{k_{n+1}}\int_D \vec{u}^n\vec{v} + \int_D \frac{\partial\vec{u}^n}{\partial t}\vec{v},\end{aligned} \tag{2.9}$$

$$-\int_D q\left(\nabla\cdot\vec{u}^{n+1}\right) = 0, \tag{2.10}$$

for all $(\vec{v}, q) \in V_D \times Q_D$. We note that p^{n+1} is not needed for subsequent time steps. Next, we recall the three ingredients of the algorithm as discussed in [21]: time integration, time-step selection and stabilization of the integrator.

Time integration. Substituting $\vec{u}^{n+1} = \vec{u}^n + k_{n+1}\vec{d}^n$ into (2.9)–(2.10), rearranging and using $\int_D q\left(\nabla\cdot\vec{u}^n\right) = 0$, we get the so-called *discrete Oseen problem*: given $\vec{u}^n$, $\partial\vec{u}^n/\partial t$ and the boundary update $\vec{g} := (\vec{g}^{n+1} - \vec{g}^n)/k_{n+1}$, we first compute $\left(\vec{d}^n, p^{n+1}\right) \in V_E \times Q_D$ such that

$$\begin{aligned}2\int_D \vec{d}^n\vec{v} + k_{n+1}\int_D \nu\,\nabla\vec{d}^n : \nabla v + k_{n+1}\int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{d}^n\right)v - \int_D p^{n+1}\left(\nabla\cdot\vec{v}\right) \\ = \int_D \frac{\partial\vec{u}^n}{\partial t}\vec{v} - \int_D \nu\,\nabla\vec{u}^n : \nabla\vec{v} - \int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{u}^n\right)\vec{v},\end{aligned} \tag{2.11}$$

$$\int_D q\left(\nabla\cdot\vec{d}^n\right) = 0, \tag{2.12}$$

for all $(\vec{v}, q) \in V_D \times Q_D$, and the TR velocity and acceleration are updated as

$$\vec{u}^{n+1} = \vec{u}^n + k_{n+1}\vec{d}^n, \qquad \frac{\partial\vec{u}^{n+1}}{\partial t} = 2\vec{d}^n - \frac{\partial\vec{u}^n}{\partial t}. \tag{2.13}$$

Time-step selection. The time step size is driven by the heuristic formula

$$k_{n+2} = k_{n+1}\left(\varepsilon/\left\|\vec{e}^{n+1}\right\|\right)^{1/3}. \tag{2.14}$$

The local truncation error $\vec{e}^{n+1}$ is estimated by

$$\vec{e}^{n+1} = \left(\vec{u}^{n+1} - \vec{u}_*^{n+1}\right)/\left[3\left(1 + k_n/k_{n+1}\right)\right], \tag{2.15}$$

where the TR velocity $\vec{u}^{n+1}$ is compared with the AB2 velocity $\vec{u}_*^{n+1}$, which is computed using the explicit formula

$$\vec{u}_*^{n+1} = \vec{u}^n + \frac{k_{n+1}}{2}\left[\left(2+\frac{k_{n+1}}{k_n}\right)\frac{\partial \vec{u}^n}{\partial t} - \left(\frac{k_{n+1}}{k_n}\right)\frac{\partial \vec{u}^{n-1}}{\partial t}\right]. \tag{2.16}$$

There are three issues that need to be addressed:

1. *The* AB2 *is not self-starting.* To start the simulation we require a function $\vec{u}^0$ with boundary data $\vec{g}^0$ such that
$$\int_D q\left(\nabla\cdot\vec{u}^0\right) = 0, \quad \forall q \in Q_D.$$
The initial acceleration (and pressure) is computed as follows: given the boundary update $\vec{g} := (\vec{g}^1 - \vec{g}^0)/k_1$, find the pair $\left(\frac{\partial \vec{u}^0}{\partial t}, p^0\right) \in V_E \times Q_D$ such that
$$\int_D \frac{\partial \vec{u}^0}{\partial t}\vec{v} - \int_D p^0\left(\nabla\cdot\vec{v}\right) = -\int_D \nu\,\nabla\vec{u}^0 : \nabla\vec{v} - \int_D\left(\vec{u}^0\cdot\nabla\vec{u}^0\right)\vec{v},$$
$$\int_D q\left(\nabla\cdot\frac{\partial \vec{u}^0}{\partial t}\right) = 0,$$
for all $(\vec{v}, q) \in V_D \times Q_D$. The discrete Oseen problem (2.11)–(2.12) is then constructed by setting $n = 0$ and defining $\vec{w}^1 = \vec{u}^0 + k_1\frac{\partial \vec{u}^0}{\partial t}$, and its solution $\left(\vec{u}^1, p^1\right)$ is used to compute the acceleration at time $t = k_1$ as
$$\frac{\partial \vec{u}^1}{\partial t} = \frac{2}{k_1}\left(\vec{u}^1 - \vec{u}^0\right) - \frac{\partial \vec{u}^0}{\partial t}, \tag{2.17}$$
and allows to compute the AB2 velocity at the second time step. The start-up is completed by switching on the time-step control at the third time step ($k_1 = k_0$).
2. *Choice of initial time step.* The strategy is to select a conservatively small value for k_0, say 10^{-8}. The time step then typically exhibits a rapid growth in the first few steps, roughly as $k_{n+1}/k_n = O\left(\left(\varepsilon/\mathtt{eps}\right)^{1/3}\right) \approx 10^4$, with $\varepsilon = 10^{-4}$ and considering the (double) machine precision $\mathtt{eps} \approx 10^{-16}$.
3. *Time-step rejection.* The new time step is proposed by formula (2.14). However, if the next time step is seriously reduced, i.e., $k_{n+2} < 0.7k_{n+1}$ (or equivalently $\left\|\vec{e}^{n+1}\right\| > (1/0.7)^3\,\varepsilon$), the next time step is rejected: the value of k_{n+1} is multiplied by $\left(\varepsilon/\left\|\vec{e}^{n+1}\right\|\right)^{1/3}$, and the current step is repeated with this new k_{n+1}.

Stabilization of the integrator. The numerical stabilization is implemented using *time-step averaging* with the purpose to annihilate any contribution of the form $(-1)^n$ to the solution and its time derivative, which is invoked periodically every n_* steps. For such a step the values of $t_* = t_n$ and $\vec{u}^* = \vec{u}^n$ are saved, we set $t_n = t_{n-1} + \frac{1}{2}k_n$, $t_{n+1} = t_* + \frac{1}{2}k_{n+1}$ and define the new "shifted" solution vectors as

$$\vec{u}^n = \frac{1}{2}\left(\vec{u}^* + \vec{u}^{n-1}\right), \qquad \frac{\partial \vec{u}^n}{\partial t} = \frac{1}{2}\left(\frac{\partial \vec{u}^n}{\partial t} + \frac{\partial \vec{u}^{n-1}}{\partial t}\right),$$
$$\vec{u}^{n+1} = \vec{u}^* + \frac{1}{2}k_{n+1}\vec{d}^n, \qquad \frac{\partial \vec{u}^{n+1}}{\partial t} = \vec{d}^n,$$

where $\vec{d}^n$ is the TR update computed via (2.11)–(2.12). In our implementation, the parameter n_* is fixed and the value is set to 10.

Finite element formulation. We consider the discretization of the Oseen problem (2.11)–(2.12) by a div-stable mixed finite element method; in the numerical experiments we use Taylor–Hood elements see, e.g., [7]. Let the bases for velocity and pressure spaces be denoted by $\{\phi_i\}_{i=1}^{n_u}$ and $\{\varphi_j\}_{j=1}^{n_p}$, respectively. In matrix terminology, the Oseen problem at time step n entails solving a linear system

$$\begin{bmatrix} \mathbf{F}^{n+1} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{d}^n \\ \mathbf{p}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_v^{n+1} \\ \mathbf{f}_p^{n+1} \end{bmatrix}, \tag{2.18}$$

where $\mathbf{F}^{n+1}$ is the velocity convection-diffusion matrix: a sum of the velocity mass matrix $\mathbf{M}$, diffusion matrix $\mathbf{A}$ and convection matrix $\mathbf{N}^{n+1}$, defined as

$$\mathbf{F}^{n+1} = 2\mathbf{M} + k_{n+1}\mathbf{A} + k_{n+1}\mathbf{N}^{n+1}, \tag{2.19}$$

where

$$\begin{aligned} \mathbf{A} &= [a_{ab}], & a_{ab} &= \int_D \nu\, \nabla\phi_b : \nabla\phi_a, \\ \mathbf{M} &= [m_{ab}], & m_{ab} &= \int_D \phi_b\phi_a, \\ \mathbf{N}^{n+1} &= \left[n_{ab}^{n+1}\right], & n_{ab}^{n+1} &= \int_D \left(\vec{w}^{n+1} \cdot \nabla\phi_b\right) \cdot \phi_a, \end{aligned}$$

and $\vec{w}^{n+1}$ is computed from (2.8). The divergence matrix $\mathbf{B}$ is defined as

$$\mathbf{B} = [b_{cd}], \qquad b_{cd} = -\int_D \varphi_c \left(\nabla \cdot \phi_d\right). \tag{2.20}$$

The right-hand side in (2.18) is constructed from the boundary data $\vec{g}^{n+1}$, the computed velocity $\vec{u}^n$ at the previous time level, and the acceleration $\frac{\partial \vec{u}^n}{\partial t}$.

3. Algorithms for the stochastic problem.

Let $(\Omega, \mathcal{F}, \mathcal{P})$ represent a complete probability space, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra on Ω and $\mathcal{P}$ is a probability measure. We assume that the randomness in the model is induced by a vector $\xi : \Omega \to \Gamma \subset \mathbb{R}^{m_\xi}$ of independent, identically distributed (i.i.d.) random variables $\xi_1(\omega), \dots, \xi_{m_\xi}(\omega)$, where $\omega \in \Omega$. Let $\mathcal{B}(\Gamma)$ denote the Borel σ-algebra on Γ induced by ξ, and μ denote the induced measure. The expected value of the product of measurable functions on Γ determines a Hilbert space $T_\Gamma \equiv L^2\left(\Gamma, \mathcal{B}(\Gamma), \mu\right)$ with inner product

$$\langle u, v \rangle = \mathbb{E}[uv] = \int_\Gamma u(\xi)\, v(\xi)\, d\mu(\xi), \tag{3.1}$$

where the symbol $\mathbb{E}$ denotes mathematical expectation.

In computations, we will use a finite-dimensional subspace $T_p \subset T_\Gamma$ spanned by a set of multivariate polynomials $\{\psi_\ell(\xi)\}$ that are orthonormal with respect to the density function μ, that is $\mathbb{E}[\psi_k\psi_\ell] = \delta_{k\ell}$, and $\psi_1 = 1$. This will be referred to as the gPC basis [40]. The dimension of the space T_p, depends on the polynomial degree. For polynomials of total degree p_ξ, the dimension is $n_\xi = \binom{m_\xi + p_\xi}{p_\xi}$.

3.1. Navier–Stokes equation with stochastic viscosity. We use the same setup as in [37]. Specifically, we consider that the expansion of viscosity is given as

$$\nu \equiv \nu\,(x,\xi) = \sum_{\ell=1}^{n_\nu} \nu_\ell(x)\psi_\ell(\xi), \tag{3.2}$$

where $\nu_\ell(x)$ is a set of deterministic spatial functions, and index ℓ is related through a multi-index to the degrees of the random variables $\xi_1,\dots,\xi_{m_\xi}$ used in the construction of the gPC basis function $\psi_\ell(\xi)$ see, e.g., [13, Section 2.4.3] or [39, Section 5.2]. For simplicity, we will also assume that both the Dirichlet boundary conditions (2.3) and the initial condition (2.5) are deterministic. We seek a discrete approximation of the velocity in the form

$$\vec{u}\,(x,t,\xi) \approx \sum_{k=1}^{n_\xi}\sum_{i=1}^{n_u} u_{ik}(x,t)\phi_i(x)\psi_k(\xi) = \sum_{k=1}^{n_\xi} \vec{u}_k(x,t)\psi_k(\xi). \tag{3.3}$$

Remark 3.1. *In literature it is sometime recommended to use a time-dependent gPC basis, that is $\psi_k(\xi,t)$, to keep the stochastic dimension low in long-time integration. However, this is a complementary strategy to the present study. Since it is not needed in our numerical experiments, we use only a time-independent gPC basis.*

3.2. Stochastic Galerkin method. The stochastic Galerkin formulation of problem (2.9)–(2.10) consists of using the expansion (3.2) and performing a Galerkin projection on the space T_Γ using mathematical expectation in the sense of (3.1). That is, we seek velocity $\vec{u}^{n+1} \in T_\Gamma \otimes V_E$ and pressure $p^{n+1} \in T_\Gamma \otimes Q_D$ for a given pair $(\vec{u}^n, p^n)$, such that

$$\begin{aligned}\mathbb{E}\left[\frac{2}{k_{n+1}}\int_D \vec{u}^{n+1}\vec{v} + \int_D \nu\,\nabla\vec{u}^{n+1} : \nabla\vec{v} + \int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{u}^{n+1}\right)\vec{v} - \int_D p^{n+1}\left(\nabla\cdot\vec{v}\right)\right]& \\ = \mathbb{E}\left[\frac{2}{k_{n+1}}\int_D \vec{u}^n\vec{v} + \int_D \frac{\partial\vec{u}^n}{\partial t}\vec{v}\right], &\quad \forall\vec{v}\in T_\Gamma\otimes V_D, \\ \mathbb{E}\left[\int_D q\left(\nabla\cdot\vec{u}^{n+1}\right)\right] = 0, \quad \forall q\in T_\Gamma\otimes Q_D,&\end{aligned}$$

and the stochastic counterpart of the discrete Oseen problem (2.11)–(2.13) is: given $\vec{u}^n$, $\partial\vec{u}^n/\partial t$ and the boundary update $\vec{g} := (\vec{g}^{n+1} - \vec{g}^n)/k_{n+1}$, we first compute $\vec{d}^n \in T_\Gamma \otimes V_E$ and $p^{n+1} \in T_\Gamma \otimes Q_D$ such that

$$\begin{aligned}\mathbb{E}\left[2\int_D \vec{d}^n\vec{v} + k_{n+1}\int_D \nu\,\nabla\vec{d}^n : \nabla v + k_{n+1}\int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{d}^n\right)v - \int_D p^{n+1}\left(\nabla\cdot\vec{v}\right)\right]& \\ = \mathbb{E}\left[\int_D \frac{\partial\vec{u}^n}{\partial t}\vec{v} - \int_D \nu\,\nabla\vec{u}^n : \nabla\vec{v} - \int_D \left(\vec{w}^{n+1}\cdot\nabla\vec{u}^n\right)\vec{v}\right], &\quad \forall\vec{v}\in T_\Gamma\otimes V_D,\end{aligned} \tag{3.4}$$

$$\mathbb{E}\left[\int_D q\left(\nabla\cdot\vec{d}^n\right)\right] = 0, \quad \forall q\in T_\Gamma\otimes Q_D, \tag{3.5}$$

and the TR velocity and the acceleration are updated as in (2.13).

3.2.1. Stochastic Galerkin finite element formulation. The Galerkin projection leads to a large coupled system of equations with structure depending on the ordering of the unknown coefficientsts $\{u_{ik}\}$, $\{p_{jk}\}$. We will group velocity-pressure pairs for each k, the index of stochastic basis functions (and order equations in the same way), giving the ordered list of coefficients

$$u_{1:n_u,1}, p_{1:n_p,1},\ u_{1:n_u,2}, p_{1:n_p,2},\ \ldots,\ u_{1:n_u,n_\xi}, p_{1:n_p,n_\xi}. \tag{3.6}$$

The discrete stochastic Oseen operator is built as follows. First, we set up the discrete components of the diffusion matrix using the expansion of viscosity (3.2) as

$$\mathbf{A}_\ell = [a_{\ell,ab}], \quad a_{\ell,ab} = \left(\int_D \nu_\ell(x)\, \nabla\phi_b : \nabla\phi_a \right), \qquad \ell = 1, \ldots, n_\nu. \tag{3.7}$$

Next, let $\vec{w}_\ell^{n+1}(x)$ denote the ℓth term of the extrapolated velocity iterate (as in the expression on the right in (3.3) for $k = \ell$) at step n, and let

$$\mathbf{N}_\ell^{n+1} = \left[n_{\ell,ab}^{n+1} \right], \qquad n_{\ell,ab}^{n+1} = \int_D \left(\vec{w}_\ell^{n+1} \cdot \nabla\phi_b \right) \cdot \phi_a, \qquad \ell = 1, \ldots, n_\xi.$$

Let $\widehat{n} = \max(n_\nu, n_\xi)$ and, if needed, define $\mathbf{A}_\ell = \mathbf{0}$ for $n_\nu < \ell \leq \widehat{n}$ and $\mathbf{N}_\ell^{n+1} = \mathbf{0}$ for $n_\xi < \ell \leq \widehat{n}$. Then in analogue to (2.19) define matrices

$$\mathbf{F}_1^{n+1} = 2\mathbf{M} + k_{n+1}\mathbf{A}_1 + k_{n+1}\mathbf{N}_1^{n+1}, \tag{3.8}$$

$$\mathbf{F}_\ell^{n+1} = k_{n+1}\mathbf{A}_\ell + k_{n+1}\mathbf{N}_\ell^{n+1}, \quad \ell = 2, \ldots, \widehat{n}, \tag{3.9}$$

which are incorporated into the block matrices

$$\mathcal{F}_1^{n+1} = \begin{bmatrix} \mathbf{F}_1^{n+1} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix}, \qquad \mathcal{F}_\ell^{n+1} = \begin{bmatrix} \mathbf{F}_\ell^{n+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \ell = 2, \ldots, \widehat{n}. \tag{3.10}$$

These operators will be coupled with matrices arising from terms in T_p,

$$\mathbf{H}_\ell = [h_{\ell,jk}], \quad h_{\ell,jk} \equiv \mathbb{E}[\psi_\ell \psi_j \psi_k], \qquad \ell = 1, \ldots, n_\nu, \quad j, k = 1, \ldots, n_\xi. \tag{3.11}$$

Combining the expressions from (3.10) and (3.11), using the ordering (3.6) yields the discrete stochastic Oseen system

$$\left(\sum_{\ell=1}^{\widehat{n}} \mathbf{H}_\ell \otimes \mathcal{F}_\ell^{n+1} \right) \mathbf{v} = \mathbf{y}, \tag{3.12}$$

where $\otimes$ denotes the matrix Kronecker product. The entries of the vectors $\mathbf{v}$ and $\mathbf{y}$ are ordered as in (3.6). Note that $\mathbf{H}_1$ is the identity matrix of order n_ξ.

Remark 3.2. *With this ordering, which we used also in [37], the coefficient matrix contains a set of n_ξ block 2×2 matrices of saddle-point structure along its block diagonal, given by*

$$\mathcal{F}_1^{n+1} + \sum_{\ell=2}^{\widehat{n}} h_{\ell,jj} \mathcal{F}_\ell^{n+1}, \qquad j = 1, \ldots, n_\xi.$$

This enables the use of existing deterministic solvers for the individual diagonal blocks.

We find it convenient to formulate the solvers in the so-called matricized format. To this end, we make use of isomorphism between $\mathbb{R}^{n_x n_\xi}$ and $\mathbb{R}^{n_x \times n_\xi}$ determined by the operators vec and mat. Let $n_x = n_u + n_p$ and consider writing the solution of (3.12) using the ordering (3.6) as $\mathbf{v} = [v_1^T, v_2^T, \dots, v_{n_\xi}^T]^T$, where $v_k = [\vec{u}_k^T, p_k^T]$ for $k = 1, \dots, n_\xi$ as in the expansions on the right in (3.3). Then we write $\mathbf{v} = \mathrm{vec}(\mathbf{V})$, $\mathbf{V} = \mathrm{mat}(\mathbf{v})$, where $\mathbf{v} \in \mathbb{R}^{n_x n_\xi}$, $\mathbf{V} \in \mathbb{R}^{n_x \times n_\xi}$ and the upper/lower case notation is assumed throughout the paper, so $\mathbf{Y} = \mathrm{mat}(\mathbf{y})$, etc. Specifically, we define the *matricized* coefficients of the solution expansion

$$\mathbf{V} = \mathrm{mat}(\mathbf{v}) = \left[v_1, v_2, \dots, v_{n_\xi}\right] \in \mathbb{R}^{n_x \times n_\xi}, \tag{3.13}$$

where the column k contains the coefficients associated with the basis function ψ_k. In this setting, since $(\mathbf{V} \otimes \mathbf{W})\,\mathrm{vec}\,(\mathbf{X}) = \mathrm{vec}\left(\mathbf{W}\mathbf{X}\mathbf{V}^T\right)$, the linear system (3.12) can be equivalently written as

$$\sum_{\ell=1}^{\widehat{n}} \mathcal{F}_\ell^{n+1} \mathbf{V} \mathbf{H}_\ell = \mathbf{Y}. \tag{3.14}$$

The time-step selection is driven by the formula (2.14), which we heuristically modify as follows. First, we run the deterministic solver with viscosity $\nu = \nu_1$ and record the set of time steps $0, t_1, t_2, , \dots, T$. Then, we divide size of each interval $[t_n, t_{n+1}]$ by n_ξ, and we further round down the time-step size to the nearest power of 10. This procedure yields a sequence of time steps, which is then used for evolution of the stochastic Galerkin method. We note that an alternative strategy could be utilized by using the gPC coefficients corresponding to the mean velocity directly in formula (2.14), that is without an a priori run of the deterministic solver.

3.3. Sampling methods. Both Monte Carlo and stochastic collocation methods are based on sampling. This entails the solution of a number of mutually independent deterministic problems at a set of sample points $\left\{\xi^{(q)}\right\}$, which give realizations of the viscosity (3.2). That is, a realization of viscosity $\nu\left(\xi^{(q)}\right)$ gives rise to deterministic functions $\vec{u}\left(\cdot, \cdot, \xi^{(q)}\right)$ and $p\left(\cdot, \cdot, \xi^{(q)}\right)$ on D that satisfy the standard deterministic Navier–Stokes equations, and to corresponding finite-element approximations.

In the Monte Carlo method, the n_{MC} sample points are generated randomly, following the distribution of the random variables ξ, and moments of the solution are obtained from ensemble averaging. In addition the coefficients in (3.3) could be determined at time t_b using[2]

$$u_{ik}(t_b) = \frac{1}{n_{MC}} \sum_{q=1}^{n_{MC}} \vec{u}^{\,(q)}\left(x_i, t_b\right) \psi_k\left(\xi^{(q)}\right),$$

where for t_b, $b = 1, \dots, n_b$, we will consider an a priori set of *time barriers*, which is used in implementation to enforce all n_{MC} instances of the deterministic solver to step through. For stochastic collocation, the sample points consist of a set of predetermined *collocation points*. This approach derives from a methodology for performing quadrature or interpolation in multidimensional space using a small number of points, a so-called sparse grid [11, 28]. There are several ways to implement stochastic collocation to obtain the coefficients in (3.3). In the basic variant of the method, it is

[2] In numerical experiments, we avoid this approximation of the gPC coefficients and directly work with the sampled quantities.

possible proceed either by constructing a Lagrange interpolating polynomial, or, in the so-called pseudospectral approach, by performing a discrete projection [39]. We use the pseudospectral approach because it facilitates a direct comparison with the stochastic Galerkin method, and we refer, e.g., to [23] for an overview and discussion of integration rules. In particular, the coefficients in (3.3) are determined at time t_b using a quadrature rule

$$u_{ik}(t_b) = \sum_{q=1}^{n_q} \vec{u}^{(q)}(x_i, t_b)\ \psi_k\left(\xi^{(q)}\right)\ w^{(q)},$$

where $\xi^{(q)}$ and $w^{(q)}$, $q = 1, \ldots, n_q$, are the collocation (quadrature) points and weights. Finally, we note that the other ways to perform stochastic collocation include the least-square approach and the compressed sensing approach see, e.g., [5, 17, 18, 27].

4. Numerical experiments. We implemented the method in MATLAB using the IFISS 3.5 package [6, 35], and in this section we present results of numerical experiments for a model problem given by a flow around an obstacle. The geometry of the problem is shown in Figure 4.1. The discretization of the physical space consists of $12,640$ velocity and 1640 pressure degrees of freedom. The viscosity was taken to be a lognormal process, and its representation was computed from an underlying Gaussian random process using the transformation described in [12]. That is, for $\ell = 1, \ldots, n_\nu$, $\psi_\ell(\xi)$ is the product of m_ξ univariate Hermite polynomials, and denoting the coefficients of the Karhunen-Loève expansion of the Gaussian process by $g_j(x)$ and $\eta_j = \xi_j - g_j$, $j = 1, \ldots, m_\xi$, the coefficients in expansion (3.2) are computed as

$$\nu_\ell(x) = \mathbb{E}\left[\psi_\ell(\eta)\right] \exp\left[g_0(x) + \frac{1}{2}\sum_{j=1}^{m_\xi}\left(g_j(x)\right)^2\right].$$

The covariance function of the Gaussian field, for points $X_i = (x_i, y_i) \in D$, $i = 1, 2$, was chosen to be

$$C(X_1, X_2) = \sigma_g^2 \exp\left(-\frac{|x_2 - x_1|}{L_x} - \frac{|y_2 - y_1|}{L_y}\right),$$

where L_x and L_y are the correlation lengths of the random variables ξ_i, $i = 1, \ldots, m_\xi$, in the x and y directions, respectively, and σ_g is the standard deviation of the Gaussian random field. The correlation lengths were set to be equal to 25% of the width and height of the domain, i.e. $L_x = 3$ and $L_y = 0.5$. The coefficient of variation of the lognormal field, defined as $CoV = \sigma_\nu/\nu_1$ where σ_ν is the standard deviation, was set to either 1% or 10%. The stochastic dimension was $m_\xi = 2$. The degree used for the polynomial expansion of the solution was $p_\xi = 3$, and the degree used for the expansion of the lognormal process was $2p_\xi$, which ensures a complete representation of the process in the discrete problem [26]. With these settings, $n_\xi = 10$ and $n_\nu = \widehat{n} = 28$, and $\mathbf{H}_\ell$ is of order 10 in (3.12). For the mean value of viscosity we used $\nu_1 = 0.02$, which corresponds to mean Reynolds number $\mathrm{Re}_1 = 100$, and $\nu_1 = 6.67 \times 10^{-3}$, which corresponds to mean Reynolds number $\mathrm{Re}_1 = 300$. We note that the steady-state case was studied in [37], and the setup for the (deterministic) time-dependent problem is the same to [7, Chapter 10] except that the length of the channel was set to 12. Specifically, the initial condition for velocity was taken zero, and the Dirichlet boundary condition on the inflow $\partial D_{\mathrm{Dir}}$ (the left side) was smoothly ramped up from zero to steady state as $\vec{u}\left(\cdot,t\right)=\left(1-e^{-5t}\right)\vec{w}$, where $\vec{w}$ is a Poiseuille (parabolic) flow profile, no-flow condition was prescribed on the top and bottom walls and natural 'do-nothing' condition was used on the outflow boundary (the right side). The initial time step was set to k_0=10^{-9}, and the problem was evolved from $t_0=0\,s$ to $T=10\,s$. The time-stepping method described in Section 2 was used for each sample of viscosity $\nu\left(x,\xi^{(q)}\right)$ for both Monte Carlo and the stochastic collocation methods, and the method from Section 3.2 was used for the stochastic Galerkin method. In order to compare the gPC coefficients at the a priori chosen set of times, which we will refer to as *time barriers*, we prescribed the stochastic Galerkin solver to step through certain times, and we used the same set also for the Monte Carlo simulation in order to compare probability density function estimates of the velocity obtained by using all three methods. Specifically, we used time barriers $t_b=\{0,0.1,0.2,0.5,1,2,5,6,8,10\}$.

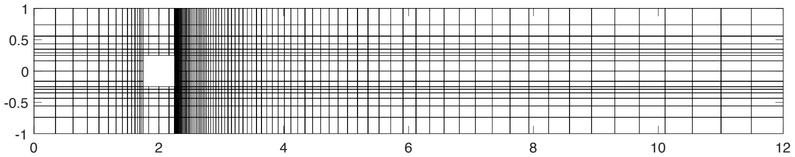

FIG. 4.1. *Finite element mesh for the flow around an obstacle problem.*

The evolution of the time step for the two deterministic cases with mean Reynolds numbers $\mathrm{Re}_1=100$ and $\mathrm{Re}_1=300$ and for the stochastic Galerkin method is shown in Figure 4.2. We note that the heuristic used for the stochastic Galerkin methods yields the same time-step selection for both values of the Reynolds number. Specifically, only three steps with size 10^{-9} are performed at the very beginning, then the step size increases to 10^{-6} and eventually to 10^{-5} for the most part of the first second. For the next two seconds it becomes 10^{-4} and eventually 10^{-3} for the rest of the time.

Now, let us consider first the case of $\mathrm{Re}_1=100$ and $CoV=10\%$. Figure 4.3 shows the evolution of the gPC coefficients of the horizontal velocity, and the symbols $\Box$ and $\times$ represent the results of Monte Carlo and stochastic collocation at some of the time barriers. It can be seen that all methods are in agreement. Figure 4.4 shows the mean horizontal velocity, Figure 4.5 the variance of the horizontal velocity, and Figure 4.6 the variance of the vertical velocity at times 0.1s, 1s and 10s. From these figures it can be seen that the flow quickly evolves during the first second, and the later changes are relatively less dramatic. It can be seen that there is symmetry in all the quantities, the mean values are essentially the same as we would expect in the deterministic case [7], and the variance of the horizontal velocity component is evolving to be concentrated in two "eddies" and it is larger than the variance of the vertical velocity component. In fact, it appears that all quantities are already at time 10s close to the steady state, see also Figures 4.13 and 4.14. A different perspective on the solution is given by Figure 4.7, which displays evolution of the probability density function (pdf) estimates in several points of the domain at times 0.1s, 1s and 10s. The left panels show the pdf estimates of the velocity in x direction at points with coordinates $(4.0100,-0.4339)$ (top), $(4.0100,0.4339)$ (bottom), where the variance of the velocity is relatively large cf. Figure 4.5. The right panels show the estimates at point $(3.6436,0)$ which is slightly downstream from the obstacle: the estimate in the x direction in the top panel and the estimate in the y direction in the bottom panel. The results were obtained using MATLAB's `ksdensity` function. It can be seen that the changes of the mean values of the pdf estimates are relatively large during the first second, and then the uncertainty gradually increases and the supports of the pdf estimates grow as the solution evolves away from the deterministic initial condition and the effect of the stochastic viscosity becomes evident.

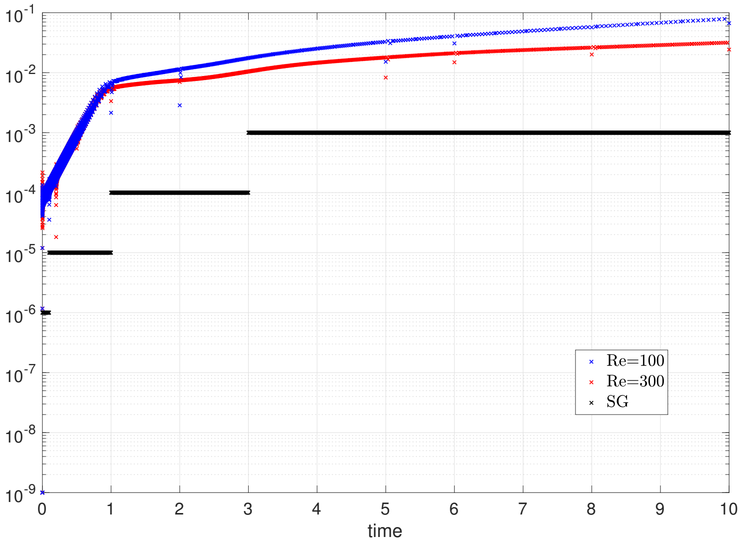

FIG. 4.2. *Evolution of the time-step size for the deterministic problems with Reynolds numbers* $\mathrm{Re}_1 = 100$ *and* $\mathrm{Re}_1 = 300$ *and for the stochastic Galerkin method (SG).*

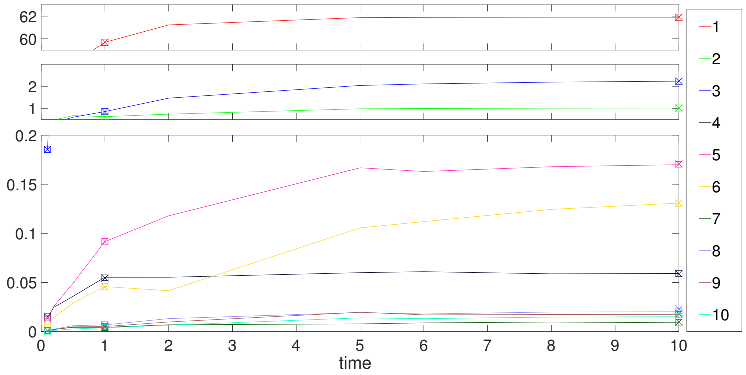

FIG. 4.3. *Evolution of the gPC coefficients corresponding to the horizontal velocity in terms of* ℓ_2*-norm for mean Reynolds number* $\mathrm{Re}_1 = 100$ *and* $CoV = 10\%$*. The symbols* $\square$ *and* $\times$ *represent the results of the Monte Carlo and stochastic collocation, respectively, at times* $0.1s$*,* $1s$ *and* $10s$*.*

Next, let us consider the case of $\mathrm{Re}_1 = 300$ and $CoV = 1\%$. Figure 4.8 shows the evolution of the gPC coefficients of the horizontal velocity. It can be seen that with increased Re_1 it takes more time for the flow to develop, including the stochastic components of the solution despite lower CoV than in the previous problem. Again, all methods are in agreement. Figure 4.9 then shows the mean horizontal velocity, Figure 4.10 the variance of the horizontal velocity, and Figure 4.11 the variance of the

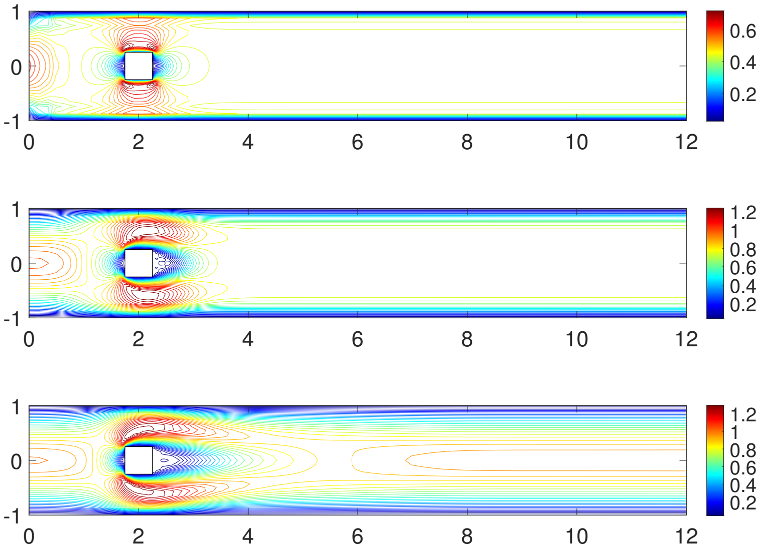

FIG. 4.4. *Mean horizontal velocity at times* 0.1*s (top),* 1*s (center) and* 10*s (bottom) for mean Reynolds number* $\mathrm{Re}_1 = 100$ *and CoV* $= 10\%$.

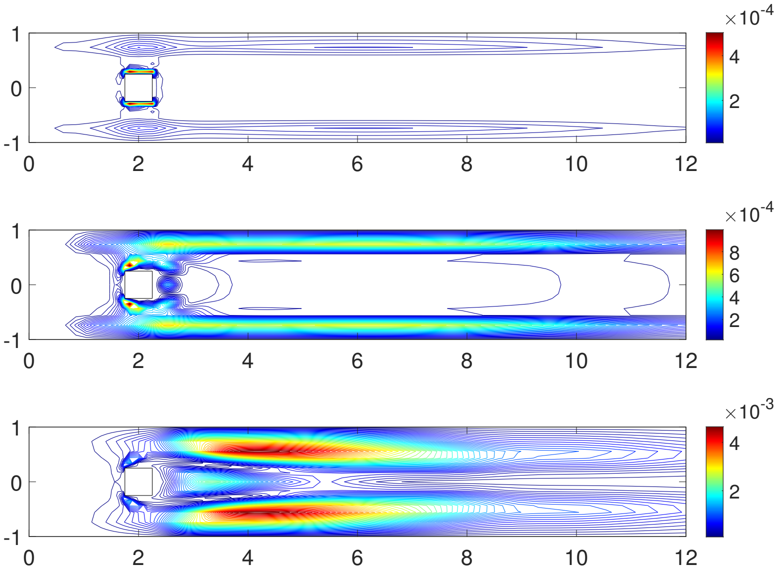

FIG. 4.5. *Variance of the horizontal velocity at times* 0.1*s (top),* 1*s (center) and* 10*s (bottom) for mean Reynolds number* $\mathrm{Re}_1 = 100$ *and CoV* $= 10\%$.

vertical velocity, all at times 0.1s, 1s and 10s. The mean quantities are quite similar to what would be expected in the deterministic case, and the variances reflect on more complex behavior of the fluid at the higher value of Re_1. Finally, Figure 4.12 displays evolution of the probability density function (pdf) estimates at the same set of points of the domain and times as Figure 4.7, and all three methods are again in agreement.

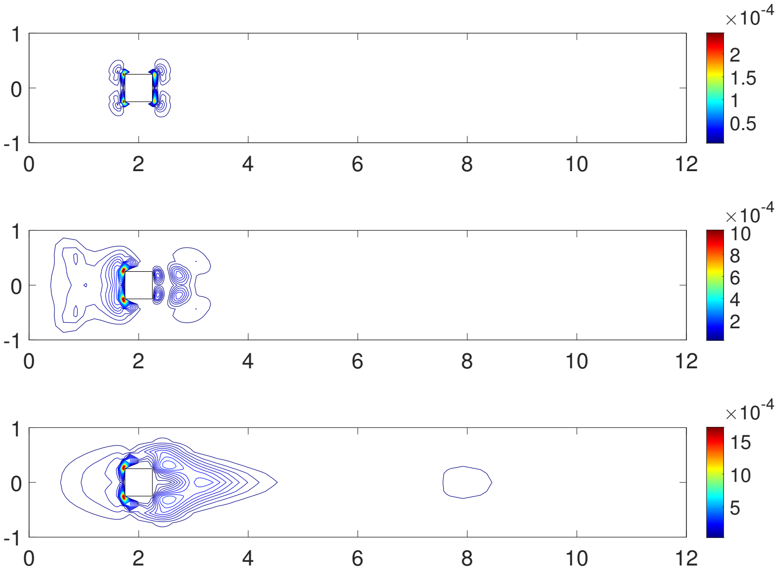

FIG. 4.6. *Variance of the vertical velocity at times* 0.1*s* *(top),* 1*s* *(center) and* 10*s* *(bottom) for mean Reynolds number* $\mathrm{Re}_1 = 100$ *and* $CoV = 10\%$.

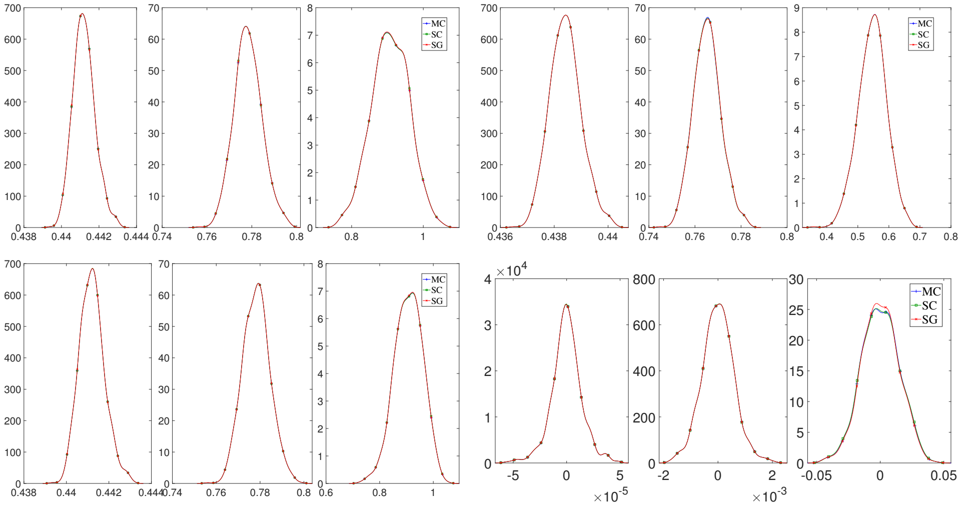

FIG. 4.7. *Estimated probability density functions at times* 0.1*s,* 1*s and* 10*s (left to right, all panels) of the horizontal velocity at points with coordinates* $(4.0100, -0.4339)$ *(top left),* $(4.0100, 0.4339)$ *(bottom left), and of the horizontal (top right) and vertical (bottom right) velocities at the point* $(3.6436, 0)$ *for mean Reynolds number* $\mathrm{Re}_1 = 100$ *and* $CoV = 10\%$.

Finally, we compare the results of the stochastic Galerkin method applied to the steady-state problem with mean Reynolds number $\mathrm{Re}_1 = 100$ and $CoV = 10\%$, which was studied by Sousedík and Elman in [37], and the results of the long-term integration at time 100s. Specifically, a comparison of the mean horizontal velocity is shown in Figure 4.13, and Figure 4.14 displays the variance of the horizontal velocity. By comparing the two figures, it can be seen that the results are virtually identical.

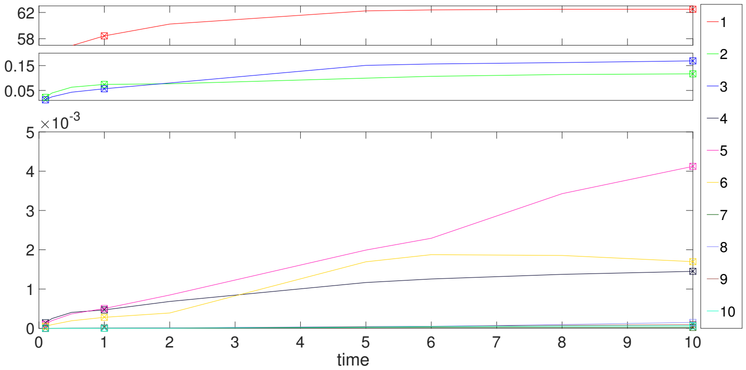

FIG. 4.8. *Evolution of the gPC coefficients corresponding to the horizontal velocity in terms of* ℓ_2*-norm for mean Reynolds number* $\mathrm{Re}_1 = 300$ *and CoV* $= 1\%$*. The symbols* □ *and* × *represent the results of the Monte Carlo and stochastic collocation, respectively, at times* 0.1*s,* 1*s and* 10*s.*

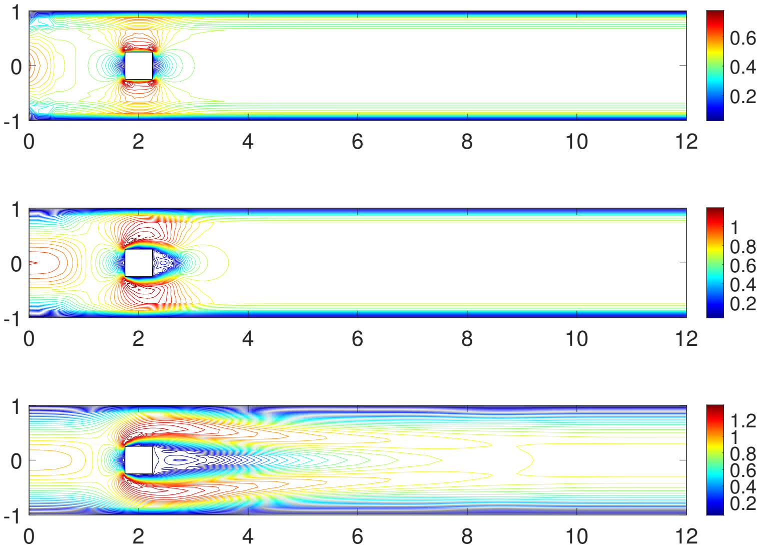

FIG. 4.9. *Mean horizontal velocity at times* 0.1*s (top),* 1*s (center) and* 10*s (bottom) for mean Reynolds number* $\mathrm{Re}_1 = 300$ *and CoV* $= 1\%$*.*

4.1. Preconditioning of the Oseen problem. The solution of the Oseen problem (3.12) in each time step of the stochastic Galerkin method is a computationally expensive task. Therefore, use of a preconditioned Krylov subspace method may be preferred over a direct solver. To this end, we used the right-preconditioned flexible GMRES (fGMRES) method [32] with the so-called mean-based preconditioner $\mathcal{M}_1^{-1} : \mathbf{R} \longmapsto \mathbf{V}$, which entails solving a linear system

$$\mathcal{M}_1 \mathbf{V} = \mathbf{R}, \tag{4.1}$$

where $\mathbf{R}$ and $\mathbf{V}$ are the matricized coefficients of the gPC expansions, cf. (3.13). Specifically, $\mathcal{M}_1^{-1}$ denotes an action of the pressure convection-diffusion (PCD) preconditioner, see [21, Section 3] and [7, Section 9.2.2], which is motivated by the block

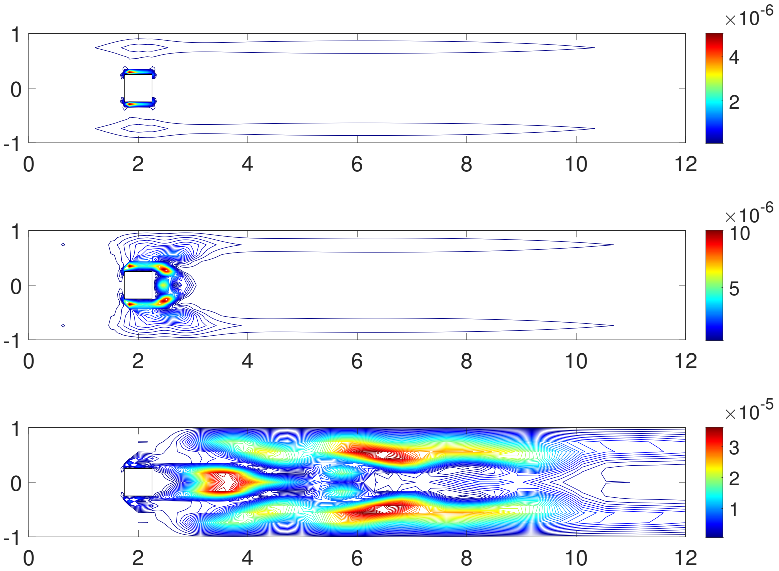

Fig. 4.10. *Variance of the horizontal velocity at times* 0.1*s (top),* 1*s (center) and* 10*s (bottom) for mean Reynolds number* $\mathrm{Re}_1 = 300$ *and CoV* $= 1\%$*.*

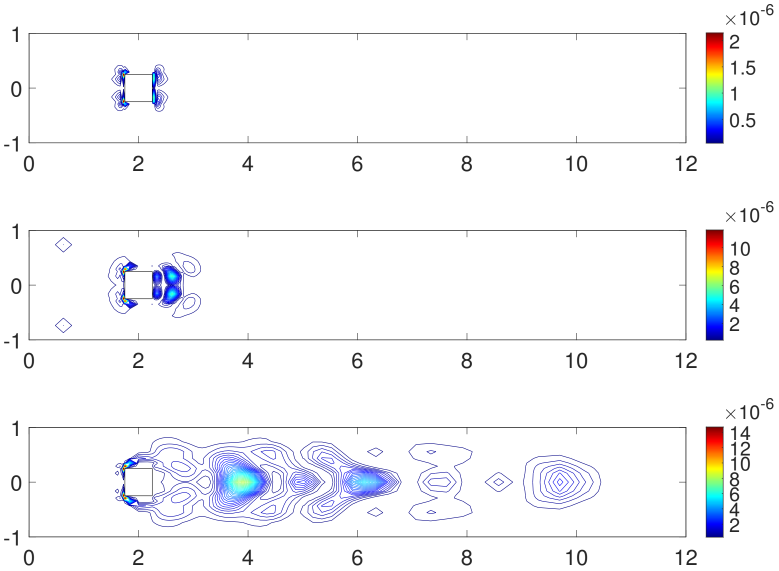

Fig. 4.11. *Variance of the vertical velocity at times* 0.1*s (top),* 1*s (center) and* 10*s (bottom) for mean Reynolds number* $\mathrm{Re}_1 = 300$ *and CoV* $= 1\%$*.*

inverse of the matrix $\mathcal{F}_1^{n+1}$ in (3.10). It can be specifically written as

$$\mathcal{M}_1^{-1} = \begin{bmatrix} \left(\mathbf{F}_1^{n+1}\right)^{-1} & \left(\mathbf{F}_1^{n+1}\right)^{-1}\mathbf{B}^T\mathbf{X}^{-1} \\ \mathbf{0} & -\mathbf{X}^{-1} \end{bmatrix}, \tag{4.2}$$

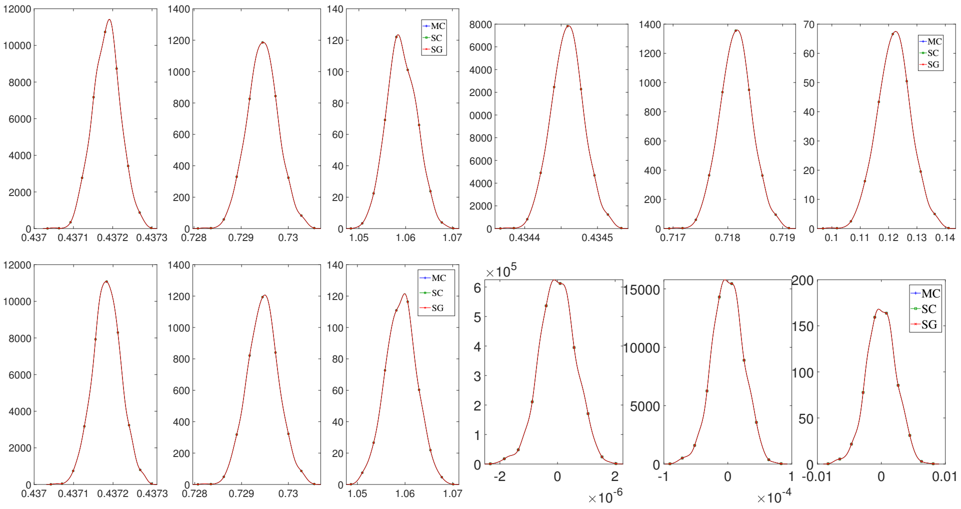

FIG. 4.12. *Estimated probability density functions at times* $0.1s$*,* $1s$ *and* $10s$ *(left to right, all panels) of the horizontal velocity at points with coordinates* $(4.0100, -0.4339)$ *(top left),* $(4.0100, 0.4339)$ *(bottom left), and of the horizontal (top right) and vertical (bottom right) velocities at the point* $(3.6436, 0)$ *for mean Reynolds number* $\mathrm{Re}_1 = 300$ *and CoV* $= 1\%$.

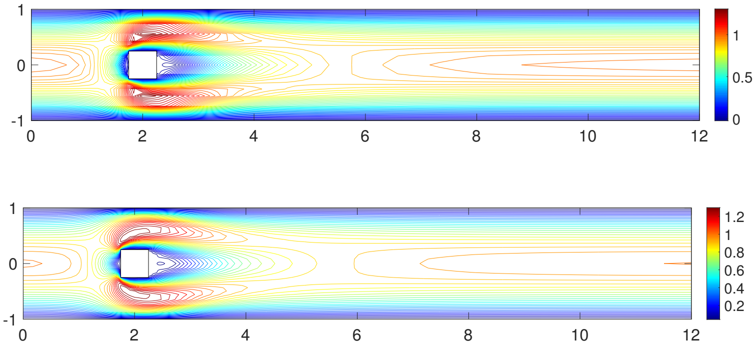

FIG. 4.13. *Mean horizontal velocity obtained using the stochastic Galerkin methods for the steady-state problem (top), and at time* $100s$ *(bottom) with mean Reynolds number* $\mathrm{Re}_1 = 100$ *and CoV* $= 10\%$.

where $\mathbf{F}_1^{n+1}$ is the matrix from (3.8), and $\mathbf{X}^{-1}$ is the pressure convection-diffusion term

$$\mathbf{X}^{-1}= \mathbf{A}_p^{-1}\mathbf{F}_p^{n+1}\mathbf{M}_p^{-1}.$$

First, we used LU factorizations of the matrices from (4.2), which are updated in each time step. Since the solves with the (mean) matrix $\mathcal{M}_1$ are thus exact, this illustrates the approximation properties of the mean-based preconditioner. Then, we also used the IFISS implementation of the PCD iterated preconditioner, in which the solves involving both $\mathbf{F}_1^{n+1}$ and $\mathbf{A}_p = \mathbf{B}\mathbf{T}^{-1}\mathbf{B}^T$, where $\mathbf{T}$ is the diagonal of the velocity mass matrix, are replaced by a single V-cycle of AMG using the IFISS default parameters, and the solve with the pressure matrix $\mathbf{M}_p$ is effected by five Chebyshev iterations, see [7, Section 10.3]. The construction of the matrix $\mathbf{F}_p^{n+1}$ is described

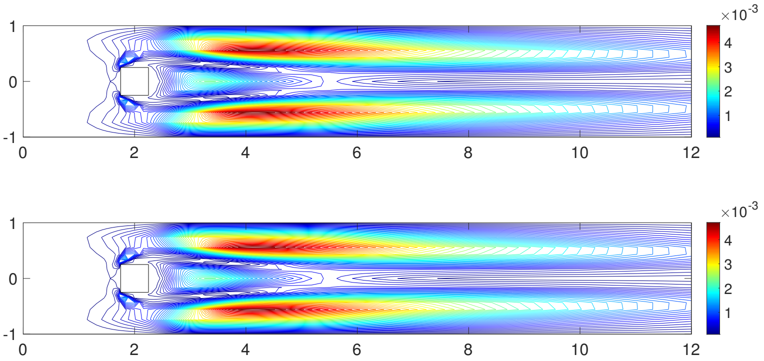

FIG. 4.14. *Variance of the horizontal velocity obtained using the stochastic Galerkin methods for the steady-state problem (top), and at time* $100s$ *(bottom) with mean Reynolds number* $\mathrm{Re}_1 = 100$ *and CoV* $= 10\%$.

in [7, Chapter 9]. We note that the AMG implementation is based on HSL_MI20 [3]. All tests started with a zero initial iterate and stopped when the relative residual was reduced to 10^{-8} in the Euclidean norm. The numbers of fGMRES iterations for solves in the time interval $[0, 10s]$ with the exact mean-based preconditioner (4.1) are shown in Figure 4.15. It can be seen that at most three iterations were needed in all steps. A comparison of the exact mean-based preconditioner (LU) and its PCD iterated variant (AMG) is illustrated by Figure 4.16. It can be seen that the numbers of iterations are the same in most cases, or it takes at most one extra step for the PCD iterated variant to converge. Thus both exact and iterated versions of the mean-based preconditioner are suitable for the problems studied in our numerical experiments.

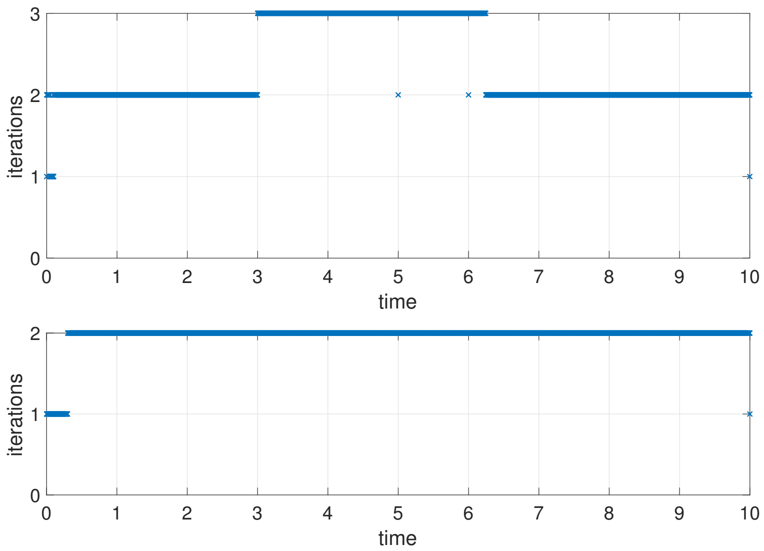

FIG. 4.15. *Numbers of fGMRES iterations with the exact mean-based preconditioner for mean Reynolds number* $\mathrm{Re}_1 = 100$, *CoV* $= 10\%$ *(top), and* $\mathrm{Re}_1 = 300$, *CoV* $= 1\%$ *(bottom).*

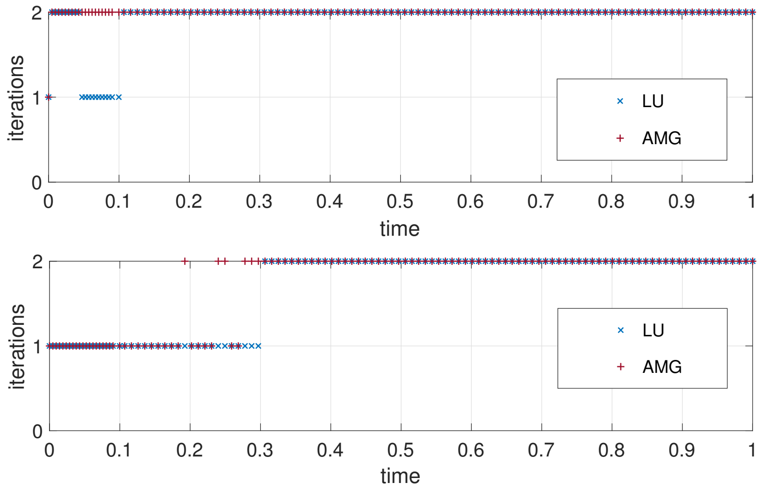

FIG. 4.16. *A comparison of the exact mean-based preconditioner (LU) and its PCD iterated variant (AMG) in terms of the numbers of fGMRES iterations for mean Reynolds number* $\mathrm{Re}_1 = 100$*, CoV* $= 10\%$ *(top), and* $\mathrm{Re}_1 = 300$*, CoV* $= 1\%$ *(bottom).*

5. Conclusion. We studied the time-dependent Navier–Stokes equations with stochastic viscosity, which was given in terms of a polynomial chaos expansion. For this problem, we developed a stochastic Galerkin method with adaptive, mean-informed time stepping. We applied the method to a popular benchmark problem given by a flow around an obstacle, and we compared the solution of the time-dependent problem after the transient to that of the corresponding steady-state problem. Next, since the time-stepping scheme is fully implicit, a linear solve with the stochastic Galerkin matrix is required in each time step. Use of direct solvers may be prohibitive due to the large size of the systems, and in fact it is even not desirable to form the matrices explicitly. Therefore, we also formulated a preconditioner, which is used by the right-preconditioned flexible GMRES method, and allows to solve the stochastic Galerkin systems efficiently. We studied two variants of the preconditioner. The first variant is based on exact factorization of the matrix corresponding to the underlying mean problem, and the second one was an iterated variant by means of an algebraic multigrid solver. In the numerical experiments we observed that the performance of the exact and iterated variants of the preconditioner was virtually identical, and only a couple of GMRES iterations were needed for convergence in all time steps. Therefore, the proposed stochastic Galerkin method is designed as a wrapper around an existing code for the corresponding deterministic problem, and in fact an efficient solver for the deterministic problem is the essential component also for the method presented in this study. Finally, we also compared the stochastic Galerkin solution with the stochastic collocation and Monte Carlo solutions, and we observed an excellent agreement for all problems studied in our numerical experiments.

REFERENCES

[1] A. Almgren, J. Bell, and W. Szymczak. A numerical method for the incompressible Navier-Stokes equations based on an approximate projection. *SIAM Journal on Scientific Computing*, 17(2):358–369, 1996.

[2] P. Bonnaire, P. Pettersson, and C.F. Silva. Intrusive generalized polynomial chaos with asynchronous time integration for the solution of the unsteady Navier-Stokes equations. *Computers & Fluids*, page 104952, 2021.

[3] Jonathan Boyle, Milan Mihajlović, and Jennifer Scott. HSL_MI20: An efficient AMG preconditioner for finite element problems in 3D. *International Journal for Numerical Methods in Engineering*, 82(1):64–98, 2010.

[4] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer-Verlag, New York – Berlin – Heidelberg, 1991.

[5] Chkifa, Abdellah, Cohen, Albert, Migliorati, Giovanni, Nobile, Fabio, and Tempone, Raul. Discrete least squares polynomial approximation with random evaluations - application to parametric and stochastic elliptic PDEs. *ESAIM: M2AN*, 49(3):815–837, 2015.

[6] Howard C. Elman, Alison Ramage, and David J. Silvester. IFISS: A computational laboratory for investigating incompressible flow problems. *SIAM Review*, 56(2):261–273, 2014.

[7] Howard C. Elman, David J. Silvester, and Andrew J. Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics.* Numerical Mathematics and Scientific Computation. Oxford University Press, New York, second edition, 2014.

[8] Howard C. Elman and Tengfei Su. A low-rank solver for the stochastic unsteady Navier–Stokes problem. *Computer Methods in Applied Mechanics and Engineering*, 364:112948, 2020.

[9] Hugo Esquivel, Arun Prakash, and Guang Lin. Flow-driven spectral chaos (FSC) method for simulating long-time dynamics of arbitrary-order non-linear stochastic dynamical systems. *Journal of Computational Physics*, 430:110044, 2021.

[10] Marc Gerritsma, Jan-Bart van der Steen, Peter Vos, and George Karniadakis. Time-dependent generalized polynomial chaos. *Journal of Computational Physics*, 229(22):8333–8363, 2010.

[11] Thomas Gerstner and Michael Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18(3-4):209–232, 1998.

[12] Roger Ghanem. The nonlinear Gaussian spectrum of log-normal stochastic processes and variables. *J. Appl. Mech.*, 66(4):964–973, 1999.

[13] Roger G. Ghanem and Pol D. Spanos. *Stochastic Finite Elements: A Spectral Approach.* Springer-Verlag New York, Inc., New York, NY, USA, 1991. (Revised edition by Dover Publications, 2003).

[14] Vivette Girault and Pierre-Arnaud Raviart. *Finite element methods for Navier-Stokes equations.* Springer-Verlag, Berlin, 1986.

[15] Philip M. Gresho, David F. Griffiths, and David J. Silvester. Adaptive time-stepping for incompressible flow part I: Scalar advection-diffusion. *SIAM Journal on Scientific Computing*, 30(4):2018–2054, 2008.

[16] M. Gunzburger, T. Iliescu, M. Mohebujjaman, and M. Schneier. An evolve-filter-relax stabilized reduced order stochastic collocation method for the time-dependent Navier-Stokes equations. *SIAM/ASA Journal on Uncertainty Quantification*, 7(4):1162–1184, 2019.

[17] Ling Guo, Akil Narayan, and Tao Zhou. Constructing least-squares polynomial approximations. *SIAM Review*, 62(2):483–508, 2020.

[18] Jerrad Hampton and Alireza Doostan. Compressive sampling of polynomial chaos expansions: Convergence analysis and sampling strategies. *Journal of Computational Physics*, 280:363–386, 2015.

[19] Vincent Heuveline and Michael Schick. A hybrid generalized polynomial chaos method for stochastic dynamical systems. *International Journal for Uncertainty Quantification*, 4(1):37–61, 2014.

[20] George Em Karniadakis, Moshe Israeli, and Steven A Orszag. High-order splitting methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 97(2):414–443, 1991.

[21] David A. Kay, Philip M. Gresho, David F. Griffiths, and David J. Silvester. Adaptive time-stepping for incompressible flow part II: Navier-Stokes equations. *SIAM Journal on Scientific Computing*, 32(1):111–128, 2010.

[22] Maxime Lacour, Guillaume Bal, and Norman Abrahamson. Dynamic stochastic finite element method using time-dependent generalized polynomial chaos. *International Journal for Numerical and Analytical Methods in Geomechanics*, 45(3):293–306, 2021.

[23] Olivier Le Maître and Omar M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics.* Scientific Computation. Springer, 2010.

[24] Kookjin Lee, Howard. C. Elman, and Bedřich Sousedík. A low-rank solver for the Navier-Stokes equations with uncertain viscosity. *SIAM/ASA Journal on Uncertainty Quantification*, 7(4):1275–1300, 2019.

[25] Gabriel J. Lord, Catherine E. Powell, and Tony Shardlow. *An Introduction to Computational Stochastic PDEs.* Cambridge Texts in Applied Mathematics. Cambridge University Press,

2014.

[26] Hermann G. Matthies and Andreas Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Comput. Meth. Appl. Mech. Eng.*, 194(12–16):1295–1331, 2005.

[27] Akil Narayan, John D. Jakeman, and Tao Zhou. A Christoffel function weighted least squares algorithm for collocation approximations. *Mathematics of Computation*, 86(306):1913–1947, 2017.

[28] Erich Novak and Klaus Ritter. High dimensional integration of smooth functions over cubes. *Numer. Math.*, 75(1):79–97, 1996.

[29] H. Çağan Özen and Guillaume Bal. Dynamical polynomial chaos expansions and long time evolution of differential equations with random forcing. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):609–635, 2016.

[30] H. Çağan Özen and Guillaume Bal. A dynamical polynomial chaos approach for long-time evolution of SPDEs. *Journal of Computational Physics*, 343:300–323, 2017.

[31] Catherine E. Powell and David J. Silvester. Preconditioning steady-state Navier-Stokes equations with random data. *SIAM J. Sci. Comput.*, 34(5):A2482–A2506, 2012.

[32] Yousef Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.

[33] M. Schick, V. V. Heuveline, and O. P. Le Maître. A Newton–Galerkin method for fluid flow exhibiting uncertain periodic dynamics. *SIAM Review*, 58(1):119–140, 2016.

[34] Michael Schick. *Uncertainty Quantification for Stochastic Dynamical Systems: Spectral Methods using Generalized Polynomial Chaos.* PhD thesis, Karlsruher Instituts für Technologie (KIT), 2011.

[35] David Silvester, Howard Elman, and Alison Ramage. Incompressible Flow and Iterative Solver Software (IFISS), version 3.5, 2016. (available online from http://www.manchester.ac.uk/ifiss/).

[36] Chen Song and Vincent Heuveline. Multilevel preconditioning of polynomial chaos method for quantifying uncertainties in a blood pump. In G. Stefanou M. Papadrakakis, V. Papadopoulos, editor, *UNCECOMP 2017 2nd ECCOMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering*, 2017. Rhodes Island, Greece, 1517 June 2017.

[37] Bedřich Sousedík and Howard C. Elman. Stochastic Galerkin methods for the steady-state Navier-Stokes equations. *Journal of Computational Physics*, 316:435–452, 2016.

[38] Gautam Andrew Wilkins. *An Empirical Chaos Expansion Method for Uncertainty Quantification.* PhD thesis, University of California San Diego, 2016. (See also preprint at https://arxiv.org/abs/1709.08668).

[39] Dongbin Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach.* Princeton University Press, 2010.

[40] Dongbin Xiu and George Em Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.

[41] Dongbin Xiu and George Em Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *J. Comput. Phys.*, 187:137–167, 2003.

[42] Dongbin Xiu and George Em Karniadakis. A new stochastic approach to transient heat conduction modeling with uncertainty. *Int. J. Heat Mass Trans.*, 46:4681–4693, 2003.