

Flow-driven spectral chaos (FSC) method for long-time integration of second-order stochastic dynamical systems

Hugo Esquivel^a, Arun Prakash^a, Guang Lin^{b,1}

^a*Lyles School of Civil Engineering, Purdue University, 550 W Stadium Ave, West Lafayette IN 47907, USA*

^b*Department of Mathematics, School of Mechanical Engineering, Purdue University, 150 N University St, West Lafayette IN 47907, USA*

Abstract

For decades, uncertainty quantification techniques based on the spectral approach have been demonstrated to be computationally more efficient than the Monte Carlo method for a wide variety of problems, particularly when the dimensionality of the probability space is relatively low. The time-dependent generalized polynomial chaos (TD-gPC) is one such technique that uses an evolving orthogonal basis to better represent the stochastic part of the solution space in time. In this paper, we present a new numerical method that uses the concept of *enriched stochastic flow maps* to track the evolution of the stochastic part of the solution space in time. The computational cost of this proposed flow-driven stochastic chaos (FSC) method is an order of magnitude lower than TD-gPC for comparable solution accuracy. This gain in computational cost is realized because, unlike most existing methods, the number of basis vectors required to track the stochastic part of the solution space, and consequently the computational cost associated with the solution of the resulting system of equations, does not depend upon the dimensionality of the probability space. Four representative numerical examples are presented to demonstrate the performance of the FSC method for long-time integration of second-order stochastic dynamical systems in the context of stochastic dynamics of structures.

Keywords: uncertainty quantification; long-time integration; stochastic flow map; stochastic dynamics of structures; flow-driven spectral chaos (FSC); TD-gPC.

Highlights

- A novel and efficient numerical method that uses an enriched stochastic flow map to track the evolution of the solution space is proposed for long-time integration of stochastic dynamics of structures.
- For the same level of accuracy, the computational cost associated with the proposed FSC method is an order of magnitude lower than other time-dependent, spectral-based methods in the current literature.
- It is partially insensitive to the curse of dimensionality compared to other spectral-based methods such as gPC and TD-gPC.
- It has the potential to be used in the context of large-scale structural engineering problems to quantify uncertainties of long-time response with high fidelity.
- Exact response expressions for the mean and variance of a single-degree-of-freedom system under free vibration and uniform stochastic stiffness are also provided.

Email addresses: hesquiv@purdue.edu (Hugo Esquivel), aprakas@purdue.edu (Arun Prakash), guanglin@purdue.edu (Guang Lin)

¹Department of Statistics, Department of Earth, Atmospheric, and Planetary Sciences (by courtesy), Purdue University.

1. Introduction

In the past few decades, the area of uncertainty quantification has received increasing attention, primarily in the fields of physics and engineering. This is not surprising since any mathematical description of a real-life physical system is always subject to the effects of input randomness. In structural engineering, for example, the randomness of a system may arise from different sources, such as variability of material properties, imperfections in geometry, loading scenarios, boundary conditions, etc. Though these physical quantities may be random, in most cases their stochastic characteristics can be identified and modeled mathematically using one or more of the numerous distributions available in statistics.

Due to the inherent complexity of real-life systems, closed-form solutions are not always possible. As an alternative, one uses accurate numerical methods that allow one to propagate and quantify the effects of input uncertainties in the system response efficiently. Historically, the polynomial chaos (PC) method developed by Wiener [1] and extended by Ghanem and Spanos [2] in the context of stochastic finite elements, has been used for uncertainty quantification. Xiu and Karniadakis [3] further developed the generalized-PC (gPC) method in which time-independent polynomials are used to decompose a stochastic process into deterministic and non-deterministic parts using the orthogonality property of the basis in the random function space (RFS). The benefit of using such orthogonal polynomials in the RFS is that when the underlying process (i.e. the solution of the ODE) is represented with them, the method leads to exponential convergence to the solution (provided the stochastic part of the solution space is not discretized).

The gPC method has undergone several modifications and extensions to improve its computational efficiency, its effectiveness for long-time integration, and its ability to handle stochastic discontinuities. The *multi-element gPC* and related methods, developed by Karniadakis and others [4–13], adaptively decompose the probability space into elements until a pre-specified threshold for the relative error in variance is reached. Then a stochastic spectral expansion on each element is used to advance the state of the system one-time step forward, and this process is repeated every time the threshold is exceeded during the simulation. Another approach is the *dynamically orthogonal PC* (DO-PC) method [14–18] where the time rate of change of the spatio-temporal function space is ensured to be orthogonal to itself. This condition, called the dynamically orthogonal (DO) condition, is enforced at every time step as the simulation proceeds. The DO-PC scheme is essentially a generalization of the POD (Proper Orthogonal Decomposition) method [19, 20] and the gPC method in the framework of continuous stochastic dynamical systems. An error analysis for the DO-PC method can be found in [21].

For long-time dynamical simulations, when the stochasticity of the system has developed significantly, the gPC method fails to capture the probability moments accurately because the probability distribution of the solution changes significantly with time. The time-dependent gPC (TD-gPC) method was proposed by Gerritsma et al. [22] to allow the basis to evolve in time so as to better represent the transient nature of the probability distribution of the solution during the simulation. Heuveline and Schick [23] modified the TD-gPC method (mTD-gPC) to account for systems governed by second-order ODEs and also improved the accuracy of the method. Generally speaking, the mTD-gPC method relies on spanning the stochastic part of the solution space, at certain time steps (aka *reset times*), by taking a full tensor product between an evolving RFS (that depends upon the evolution of the state variables of the system) and the original RFS. However, since a full tensor product is required to be conducted at the reset times, the method suffers from the curse of dimensionality. A *hybrid generalized polynomial chaos* was also developed in [23] to help alleviate the curse of dimensionality of mTD-gPC. This method, however, requires the use of an (h, p) -discretization for the stochastic part of the solution space in contrast to mTD-gPC which solely requires the use of a p -discretization.

More recently, a generalized PC method based on flow map composition was proposed by Luchtenburg et al. [24] to address the long-time uncertainty propagation in dynamical systems more effectively. The method fundamentally uses short-time flow maps based on spectral polynomial bases to account for the stretching and folding effect caused by the evolution of the system's state over time. As with mTD-gPC, this method suffers from the curse of dimensionality, because a tensor product is needed to construct the spectral polynomial bases during the simulation. Ozen and Bal [25] developed the *dynamical gPC* (DgPC) method to address the long-time behavior of stochastic dynamical systems via a generalization of the PCE

(Polynomial Chaos Expansion) framework. They demonstrated that results from DgPC match well with those obtained from other standard methods such as Monte Carlo.

In this paper, a flow-driven spectral chaos (FSC) method is proposed which tracks the evolution of the random basis via an *enriched stochastic flow map* of the state of the system.² The enriched flow map consists of the time derivatives of the solution up to a specified order. Unlike any gPC-based method, the number of basis vectors needed in FSC to represent the stochastic part of the solution space does not grow with the dimensionality of the probability space. However, as with any gPC-based method, the FSC method requires the computation of inner products using quadrature points distributed over the entire random domain, and the number of quadrature points can grow significantly with the dimensionality of the probability space. Nevertheless, FSC presents a significant advantage over gPC-based methods as its computational cost, for comparable accuracy, is an order of magnitude lower compared to existing TD-gPC methods. Conversely, for comparable computational cost, FSC is able to achieve an order of magnitude of better accuracy than TD-gPC.

This paper is organized as follows. Section 2 introduces the notation and definitions used in this paper and Section 3 presents the precise problem statement. The spectral approach for solving this problem is presented in Section 4 and Section 5 describes the proposed FSC method in detail. Three numerical examples are presented in Section 6 to demonstrate and compare the accuracy of FSC to other existing methods, such as mTD-gPC and Monte Carlo. The FSC method is then applied to quantify uncertainties in the structural dynamics of a 3-story building subject to an earthquake excitation in Section 7. In Appendix A, we define the random bases that we use to span the random function space of the problem in hand. Finally, in Appendix B, we provide expressions for the mean and variance of the exact response of a single-degree-of-freedom system under free vibration and uniformly-distributed stochastic stiffness, followed by a brief outline of the standard Monte Carlo method in Appendix C.

2. Setting and notation

Definition 1 (Temporal space). Let $(\mathfrak{T}, \mathcal{O})$ be a topological space, where $\mathfrak{T} = [0, T]$ is a closed interval representing the temporal domain of the system, T is a positive real number symbolizing the duration of the simulation, and $\mathcal{O} = \mathcal{O}_{\mathbb{R}} \cap \mathfrak{T}$ is the topology on \mathfrak{T} with $\mathcal{O}_{\mathbb{R}}$ denoting the standard topology over \mathbb{R} . In this paper, $(\mathfrak{T}, \mathcal{O})$ is called the *temporal space* of the system.

Definition 2 (Random space). Let $(\Omega, \Omega, \lambda)$ be a (complete) probability space, where Ω is the sample space, $\Omega \subset 2^{\Omega}$ is the σ -algebra on Ω (aka the collection of events in probability theory), and $\lambda : \Omega \rightarrow [0, 1]$ is the probability measure on Ω . Let $\xi : (\Omega, \Omega) \rightarrow (\mathbb{R}^d, \mathcal{B}_{\mathbb{R}^d})$ be a measurable function (aka random variable) given by $\xi = \xi(\omega)$, with $\mathcal{B}_{\mathbb{R}^d}$ denoting the Borel σ -algebra over \mathbb{R}^d . In this work, the measure space (Ξ, Ξ, μ) is called *random space*, where $\Xi = \xi(\Omega) \subset \mathbb{R}^d$ is a set representing the random domain of the system, d denotes the dimensionality of the random space, $\Xi = \mathcal{B}_{\mathbb{R}^d} \cap \Xi$ is the σ -algebra on Ξ , and $\mu : \Xi \rightarrow [0, 1]$ is the probability measure on Ξ defined by the pushforward of λ by ξ , that is $\mu = \xi_*(\lambda)$.

Note that more structure can be added to these spaces whenever they are needed in the analysis. However, in order to keep the above definitions as elementary as possible, we singled out those mathematical objects that did not play a crucial role in the development of this work, such as the definition of a metric, a norm or an inner product for the underlying set.

Definition 3 (Temporal function space). Let $\mathcal{T}(n) = C^n(\mathfrak{T}, \mathcal{O}; \mathbb{R})$ be a continuous n -differentiable function space. This *temporal function space* is the space of all functions $f : (\mathfrak{T}, \mathcal{O}) \rightarrow (\mathbb{R}, \mathcal{O}_{\mathbb{R}})$ that have continuous first n derivatives on $(\mathfrak{T}, \mathcal{O})$.

²A *generalized* version of this FSC method can already be found in [26]. In that work, the FSC scheme presented in Section 5.4 is simply denoted as FSC-1.

Definition 4 (Random function space). Let $\mathcal{X} = (L^2(\Xi, \mathbf{E}, \mu; \mathbb{R}), \langle \cdot, \cdot \rangle)$ be a Lebesgue square-integrable space equipped with its standard inner product

$$\langle \cdot, \cdot \rangle : L^2(\Xi, \mathbf{E}, \mu; \mathbb{R}) \times L^2(\Xi, \mathbf{E}, \mu; \mathbb{R}) \rightarrow \mathbb{R} \quad \Leftrightarrow \quad \langle f, g \rangle = \int f g \, d\mu.$$

This *random function space* (aka RFS in this paper) is the space of all measurable functions $f : (\Xi, \mathbf{E}) \rightarrow (\mathbb{R}, \mathcal{B}_{\mathbb{R}})$ that are square-integrable with respect to μ . (By f we actually mean an equivalence class of square-integrable functions that are equal μ -almost everywhere; usually denoted by $[f]$ in the literature.) This inner product space is known to form a Hilbert space because it is complete under the metric induced by the inner product. Furthermore, we define $\{\Psi_j : (\Xi, \mathbf{E}) \rightarrow (\mathbb{R}, \mathcal{B}_{\mathbb{R}})\}_{j=0}^{\infty}$ to be a complete orthogonal basis in \mathcal{X} , such that $\Psi_0(\xi) = 1$ for all $\xi \in \Xi$. It is worth noting that such a basis does not necessarily need to consist of d -variate polynomials as in Refs. [3, 27], but may also consist of more general functions (including non-elementary functions such as wavelets).

Therefore, any function $f \in \mathcal{X}$ can be represented in a Fourier series of the form:

$$f = \sum_{j=0}^{\infty} f^j \Psi_j,$$

where f^j denotes the j -th coefficient of the series with the superscript not symbolizing an exponentiation.

Moreover, the dual space of \mathcal{X} , which we denote by \mathcal{X}' , is simply the space spanned by the continuous linear functionals $\{\Psi^i : \mathcal{X} \rightarrow \mathbb{R}\}_{i=0}^{\infty}$ defined by:

$$\Psi^i[f] := [\Psi^i, f] = \frac{\langle \Psi_i, f \rangle}{\langle \Psi_i, \Psi_i \rangle} \equiv f^i,$$

where $[\cdot, \cdot] : \mathcal{X}' \times \mathcal{X} \rightarrow \mathbb{R}$ is the natural pairing map between \mathcal{X} and \mathcal{X}' . This continuous dual space is also known to form a Hilbert space, thanks to the Riesz representation theorem [28].

We recall that the orthogonality property of the basis $\{\Psi_j \in \mathcal{X}\}_{j=0}^{\infty}$ implies that:

$$\langle \Psi_i, \Psi_j \rangle := \int \Psi_i \Psi_j \, d\mu = \langle \Psi_i, \Psi_i \rangle \delta_{ij},$$

where δ_{ij} is the Kronecker delta.

Definition 5 (Solution space and root space). Let $\mathcal{U} = \mathcal{T}(2) \otimes \mathcal{X}$ and $\mathcal{V} = \mathcal{T}(0) \otimes \mathcal{X}$ be, respectively, the *solution space* and the *root space* of the system.

In what follows, we assume that the components of the d -tuple random variable $\xi = (\xi^1, \dots, \xi^d)$ are mutually independent, and as sketched in Fig. 1 that the random domain Ξ is a hypercube of d dimensions obtained by performing a d -fold Cartesian product of intervals $\bar{\Xi}_i := \xi^i(\Omega)$. It is for this reason that we define the probability measure in \mathcal{X} hereafter as

$$\mu = \bigotimes_{i=1}^d \mu^i, \quad \text{or equivalently, } d\mu \equiv \mu(d\xi) = \prod_{i=1}^d \mu^i(d\xi^i) \equiv d\mu^1 \cdots d\mu^d,$$

where $\mu^i(d\xi^i) =: d\mu^i$ represents the probability measure of $d\xi^i$ in the vicinity of $\xi^i \in \bar{\Xi}_i$.

3. Problem statement

In this work, we are interested in solving the non-autonomous, second-order ODE described below.

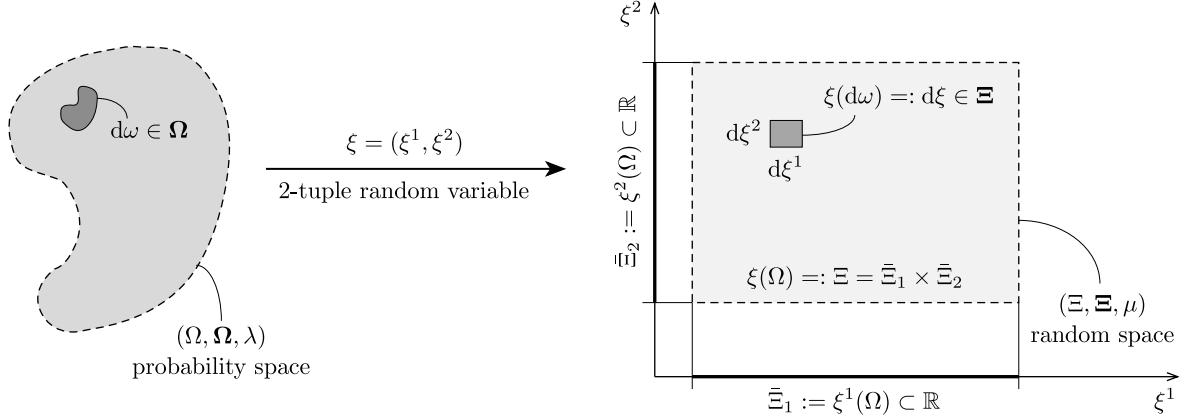


Figure 1: Relationship between probability space and random space for $d = 2$

Find the real-valued stochastic process $u : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}$ in \mathcal{U} , such that (μ -a.e.):

$$m\ddot{u} + \mathcal{F}[u, \dot{u}] = p \quad \text{on } \mathfrak{T} \times \Xi \quad (1a)$$

$$\{u(0, \cdot) = u, \dot{u}(0, \cdot) = v\} \quad \text{on } \{0\} \times \Xi, \quad (1b)$$

where $m, \mathcal{F}[u, \dot{u}], p : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}$ are elements of \mathcal{V} such that $m(t, \xi) \neq 0$ for all $(t, \xi) \in \mathfrak{T} \times \Xi$, and $u, v : \Xi \rightarrow \mathbb{R}$ are elements of \mathcal{Z} . Note that in (1), $\dot{u} := \partial_t u$ and $\ddot{u} := \partial_t^2 u$ denote, respectively, the first and second partial derivatives of u with respect to time. Thus, $\dot{u} : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}$ is an element of $\mathcal{T}(1) \otimes \mathcal{Z}$, and $\ddot{u} : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}$ is an element of $\mathcal{T}(0) \otimes \mathcal{Z} \equiv \mathcal{V}$ (the root space).

When (1) is specialized to study the (nonlinear) behavior of a single-degree-of-freedom system, $m\ddot{u}$ represents the inertial force of the system with $m : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}^+$ symbolizing the mass of the system, $\mathcal{F}[u, \dot{u}]$ is the damping and resisting force of the system, and p is the external force acting on the system. Furthermore, in this case u , \dot{u} and \ddot{u} denote the displacement, the velocity and the acceleration response of the system, respectively.

When (1) is written in modeling notation, it becomes

$$y = \mathcal{M}[u][x] \quad \text{subject to initial condition } \mathcal{I}[u], \quad (1^*)$$

where $\mathcal{M}[u] : \mathcal{V}^3 \rightarrow \mathcal{V}^s$ represents the mathematical model of the system defined by (1a), $x = (x_1, x_2, x_3) : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}^3$ is the 3-tuple input of $\mathcal{M}[u]$, and $y = (y_1, \dots, y_s) : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}^s$ is the s -tuple output of $\mathcal{M}[u]$ (aka the s -tuple observable in physics or the s -tuple response in engineering). In addition, $\mathcal{I}[u]$ represents the initial condition for $\mathcal{M}[u]$ which is given by (1b). Thus, by comparing (1) to (1*), the components of x can be identified in the following way: $x_1 = m$, $x_2 = \mathcal{F}[u, \dot{u}]$ and $x_3 = p$. The objective of this mathematical model is to propagate and quantify the effects of input uncertainty x on system's output y . Therefore, besides seeking u in \mathcal{U} as mentioned earlier, it is also important to compute the probability moments of y as time progresses.

4. Solution based on the spectral approach

4.1. Discretization of random function space

Since by hypothesis u is an element of \mathcal{U} , then it can be represented by the Fourier series

$$u(t, \xi) = \sum_{j=0}^{\infty} u^j(t) \Psi_j(\xi), \quad (2)$$

where u^j is a temporal function in $\mathcal{T}(2)$ denoting the j -th random mode of u . This series, usually referred to as *stochastic spectral expansion* in the literature [29, 30], has the remarkable property that when u is sufficiently smooth in the solution space (and, of course, provided that the basis is orthogonal with respect to the probability measure defined in \mathcal{Z}), it leads to exponential convergence to the solution [3, 31].

For the purpose of this manuscript, let us simply consider a p -discretization of the random function space as follows. Let $\mathcal{Z}^{[P]} = \text{span}\{\Psi_j\}_{j=0}^P$ be a finite subspace of \mathcal{Z} with $P + 1 \in \mathbb{N}_1$ denoting the dimensionality of the subspace. If we let $u^{[P]}(t, \cdot)$ be an element of $\mathcal{Z}^{[P]}$, then it is evident from (2) that³:

$$u(t, \xi) \approx u^{[P]}(t, \xi) = \sum_{j=0}^P u^j(t) \Psi_j(\xi) \equiv u^j(t) \Psi_j(\xi), \quad (3)$$

where for notational convenience we have omitted the summation sign in the last equality (aka Einstein summation convention). Therefore, unless otherwise noted hereinafter, a summation sign will always be implied over the repeated index $j \in \{0, 1, \dots, P\}$.

Substituting (3) into (1) gives

$$m\ddot{u}^j \Psi_j + \mathcal{F}[u^j \Psi_j, \dot{u}^j \Psi_j] = p \quad \text{on } \mathfrak{T} \times \Xi \quad (4a)$$

$$\{u^j(0) \Psi_j = u, \dot{u}^j(0) \Psi_j = v\} \quad \text{on } \{0\} \times \Xi. \quad (4b)$$

Projecting (4) onto $\mathcal{Z}^{[P]}$ yields a system of $P + 1$ ordinary differential equations of second order in the variable t , where the unknowns are the random modes $u^j = u^j(t)$ and $\dot{u}^j = \dot{u}^j(t)$:

$$\langle \Psi_i, m\Psi_j \rangle \ddot{u}^j + \langle \Psi_i, \mathcal{F}[u^j \Psi_j, \dot{u}^j \Psi_j] \rangle = \langle \Psi_i, p \rangle \quad \text{on } \mathfrak{T} \quad (5a)$$

$$\{u^i(0) = \langle \Psi_i, u \rangle / \langle \Psi_i, \Psi_i \rangle, \dot{u}^i(0) = \langle \Psi_i, v \rangle / \langle \Psi_i, \Psi_i \rangle\} \quad \text{on } \{0\} \quad (5b)$$

with $i, j \in \{0, 1, \dots, P\}$. Note that in order to get (5), we simply applied on both sides of each equation the linear functionals $\{\Psi^i \in \mathcal{Z}'\}_{i=0}^P$ one by one, and then we simplified the resulting expressions. It is also worth noting that because the randomness of the stochastic system has effectively been absorbed by the application of the aforementioned functionals, the system of equations that we are dealing with at this point is no longer ‘stochastic’ but ‘deterministic’. In other words, the system now depends merely on the time variable t rather than on the tuple (t, ξ) .

System (5) can also be restated using multilinear and tensor algebra notation as follows:

$$m^i_j \ddot{u}^j + \mathcal{F}^i[u^j, \dot{u}^j] = p^i \quad \text{on } \mathfrak{T} \quad (6a)$$

$$\{u^i(0) = u^i, \dot{u}^i(0) = v^i\} \quad \text{on } \{0\}, \quad (6b)$$

where $i, j \in \{0, 1, \dots, P\}$ (summation sign implied over repeated index j), and:

$$m^i_j(t) = \langle \Psi_i, m(t, \cdot) \Psi_j \rangle / \langle \Psi_i, \Psi_i \rangle, \quad \mathcal{F}^i[u^j, \dot{u}^j](t) = \langle \Psi_i, \mathcal{F}[u^j \Psi_j, \dot{u}^j \Psi_j](t, \cdot) \rangle / \langle \Psi_i, \Psi_i \rangle,$$

$$p^i(t) = \langle \Psi_i, p(t, \cdot) \rangle / \langle \Psi_i, \Psi_i \rangle, \quad u^i = \langle \Psi_i, u \rangle / \langle \Psi_i, \Psi_i \rangle \quad \text{and} \quad v^i = \langle \Psi_i, v \rangle / \langle \Psi_i, \Psi_i \rangle,$$

whence $m^i_j, \mathcal{F}^i[u^j, \dot{u}^j], p^i \in \mathcal{T}(0)$ and $u^i, v^i \in \mathbb{R}$. To simplify notation, we have taken $\mathcal{F}^i[u^j, \dot{u}^j]$ as the short notation for $\mathcal{F}^i[u^0, \dots, u^j, \dots, u^P, \dot{u}^0, \dots, \dot{u}^j, \dots, \dot{u}^P]$.

To evaluate the inner products approximately, any integration technique of one’s choice can be used, including those addressed in [32–36]. If, for instance, a Gaussian quadrature rule is adopted, the inner products are computed with:

$$\langle f, g \rangle := \int f g \, d\mu \approx \mathcal{Q}^{[Q]}[fg] := \sum_{i=1}^Q f(\xi_i) g(\xi_i) w_i,$$

where $w_i \in \mathbb{R}^+$ denotes the quadrature weight associated with the Gaussian quadrature point $\xi_i \in \Xi$ (w.r.t. μ), and $Q \in \mathbb{N}_1$ represents the number of quadrature points involved in approximating the evaluation of the inner product.

³As long as we assume that $\{\Psi_j\}_{j=0}^\infty$ is well-graded to carry out the approximation of u this way.

4.2. Discretization of temporal function space

Once \mathcal{Z} has been discretized, the temporal function space can be discretized using an (h, p) -discretization for $\mathcal{T}(2)$. In the literature, there exists an extensive list of time integration techniques that one can employ in order to solve the ODE system given by (6) numerically. For example, the Runge-Kutta method [37] of fourth-order (aka RK4 method) or the Newmark- β method [38, 39].

4.3. Computation of probability moments

In this manuscript, the probability moments of interest are the mean and the variance of the system's response. For this reason, we define these objects below.

Suppose that $z := y_k$ is the k -th component of output $y = \mathcal{M}[u][x]$. If $z \in \mathcal{V}$, then it can approximately be expanded with a truncated Fourier series similar to the one set forth in (3) to obtain:

$$z(t, \xi) \approx z^{[P]}(t, \xi) = \sum_{j=0}^P z^j(t) \Psi_j(\xi) \equiv z^j(t) \Psi_j(\xi), \quad (7)$$

where the j -th random mode of z is given by

$$z^j(t) = \frac{\langle \Psi_j, z(t, \cdot) \rangle}{\langle \Psi_j, \Psi_j \rangle}.$$

(Note that P in expression (7) does not necessarily need to be the same as in (3).)

The expectation of z , $\mathbf{E}[z] : \mathfrak{T} \rightarrow \mathbb{R}$, is simply given by the first random mode of z :

$$\mathbf{E}[z](t) := \int z(t, \cdot) d\mu = z^0(t), \quad (8)$$

whereas the variance of z , $\text{Var}[z] : \mathfrak{T} \rightarrow \mathbb{R}_0^+$, is defined by the partial sum:

$$\text{Var}[z](t) := \int (z(t, \cdot) - \mathbf{E}[z](t))^2 d\mu = \sum_{j=1}^P \langle \Psi_j, \Psi_j \rangle z^j(t) z^j(t). \quad (9)$$

5. Flow-driven spectral chaos (FSC) method

5.1. Stochastic flow map

Observe that the stochastic system given by (1) can also be expressed as:

$$\partial_t^2 u(t, \xi) := \mathcal{f}(t, \xi, s(t, \xi)) = (p(t, \xi) - \mathcal{F}[u, \dot{u}](t, \xi)) / m(t, \xi) \quad \text{on } \mathfrak{T} \times \Xi \quad (10a)$$

$$\{u(0, \xi) = \mathbf{u}(\xi), \dot{u}(0, \xi) = \mathbf{v}(\xi)\} \quad \text{on } \{0\} \times \Xi, \quad (10b)$$

where $s = (u, \dot{u}) \in \prod_{j=1}^2 \mathcal{T}(3-j) \otimes \mathcal{Z}$ is the configuration state of the system over $\mathfrak{T} \times \Xi$, and $\mathcal{f} : \mathfrak{T} \times \Xi \times \mathbb{R}^2 \rightarrow \mathbb{R}$ is a noisy, non-autonomous function defining the response $\ddot{u} = \partial_t^2 u$.

Therefore, if the solution is analytic on \mathfrak{T} for all $\xi \in \Xi$, then it can be represented by the Taylor series:

$$u(t_i + h, \xi) = \sum_{j=0}^{\infty} \frac{h^j}{j!} \partial_t^j u(t_i, \xi) = \sum_{j=0}^M \frac{h^j}{j!} \partial_t^j u(t_i, \xi) + O(h^{M+1})(\xi), \quad (11)$$

where $h := t - t_i$ is the time-step size used for the simulation around t_i (once t is fixed), $t_i \in \mathfrak{T}$ is the time instant of the simulation, and $M \in \mathbb{N}_1$ is the order of the flow map we are interested to implement.

For this system, the *stochastic flow map* of order M , $\varphi(M) : \mathbb{R} \times \mathcal{X}^2 \rightarrow \mathcal{X}^2$, can be defined as a random map given by:

$$\begin{aligned}\varphi(M)(h, s(t_i, \cdot)) &=: s(t_i + h, \cdot) = (u(t_i + h, \cdot), \dot{u}(t_i + h, \cdot)) - O(h^{M+1}) \\ &= \left(\sum_{j=0}^M \frac{h^j}{j!} \partial_t^j u(t_i, \cdot), \sum_{j=0}^M \frac{h^j}{j!} \partial_t^{j+1} u(t_i, \cdot) \right),\end{aligned}\quad (12)$$

where this $s(t_i + h, \cdot)$ is the same as in (10) if $M \rightarrow \infty$. However, to avoid unnecessary complexity in notation, no distinction between these definitions will be made in this work. That is, from now on we will assume that there is an equivalence relation \sim between $s(t_i + h, \cdot) - O(h^{M+1})$ and the $s(t_i + h, \cdot)$ defined in (12). Notice that (12) now requires that $s = (u, \dot{u}) \in \prod_{j=1}^2 \mathcal{T}(M - j + 2) \otimes \mathcal{X}$.

For the sake of illustration, suppose $M = 4$. Differentiating (10a) with respect to time three times (i.e. $M - 1$ times) gives

$$\partial_t^3 u := D_t f = \partial_t f + \partial_u f \partial_t u + \partial_{u\dot{u}} f \partial_t^2 u \quad (13a)$$

$$\partial_t^4 u := D_t^2 f = \partial_t^2 f + 2 \partial_{tu}^2 f \partial_t u + (2 \partial_{tu}^2 f + \partial_u f) \partial_t^2 u + \partial_{u\dot{u}} f \partial_t^3 u \quad (13b)$$

$$\partial_t^5 u := D_t^3 f = \partial_t^3 f + 3 \partial_{ttu}^3 f \partial_t u + 3 (\partial_{ttu}^3 f + \partial_{tu}^2 f) \partial_t^2 u + (3 \partial_{tu}^2 f + \partial_u f) \partial_t^3 u + \partial_{u\dot{u}} f \partial_t^4 u. \quad (13c)$$

Hence, when $M = 4$, the stochastic flow map of the system, $\varphi(4) = (\varphi^1(4), \varphi^2(4))$, is given by:

$$\begin{aligned}\varphi(4)(h, s(t_i, \cdot)) &=: s(t_i + h, \cdot) = (u(t_i + h, \cdot), \dot{u}(t_i + h, \cdot)) - O(h^5) \\ &= \left(\sum_{j=0}^4 \frac{h^j}{j!} \partial_t^j u(t_i, \cdot), \sum_{j=0}^4 \frac{h^j}{j!} \partial_t^{j+1} u(t_i, \cdot) \right),\end{aligned}\quad (14)$$

where the second and higher time derivatives of u at $t = t_i$ are computed with (10a) and (13), respectively.

Note that if (10) is an autonomous ODE, the expressions prescribed by (13) reduce to:

$$\partial_t^3 u = \partial_u f \partial_t u + \partial_{u\dot{u}} f \partial_t^2 u, \quad \partial_t^4 u = \partial_u f \partial_t^2 u + \partial_{u\dot{u}} f \partial_t^3 u \quad \text{and} \quad \partial_t^5 u = \partial_u f \partial_t^3 u + \partial_{u\dot{u}} f \partial_t^4 u. \quad (13^*)$$

5.2. Enriched stochastic flow map

In this work, we define the *enriched stochastic flow map* of order M , $\hat{\varphi}(M) : \mathbb{R} \times \mathcal{X}^{M+2} \rightarrow \mathcal{X}^{M+2}$, such that its k -th component is given by:

$$\hat{\varphi}^k(M)(h, \hat{s}(t_i, \cdot)) =: \hat{s}^k(t_i + h, \cdot) = \begin{cases} \varphi^k(M)(h, s(t_i, \cdot)) & \text{for } k \in \{1, 2\} \\ D_t^{k-3} f(t_i + h, \cdot, s(t_i + h, \cdot)) & \text{otherwise,} \end{cases} \quad (15)$$

where $\hat{s} = (u, \dot{u}, \dots, \partial_t^{M+1} u) \in \prod_{j=1}^{M+2} \mathcal{T}(M - j + 2) \otimes \mathcal{X}$ is called the *enriched configuration state* of the system over $\mathcal{X} \times \Xi$, and $k \in \{1, 2, \dots, M+2\}$.

For instance, when $M = 4$, the components of the enriched stochastic flow map, $\hat{\varphi}(4)$, are

$$\hat{\varphi}^1(4)(h, s(t_i, \cdot)) := \varphi^1(4)(h, s(t_i, \cdot)) = s^1(t_i + h, \cdot) = u(t_i + h, \cdot) - O(h^5), \quad (16a)$$

$$\hat{\varphi}^2(4)(h, s(t_i, \cdot)) := \varphi^2(4)(h, s(t_i, \cdot)) = s^2(t_i + h, \cdot) = \dot{u}(t_i + h, \cdot) - O(h^5), \quad (16b)$$

$$\hat{\varphi}^3(4) := f = \partial_t^2 u, \quad \hat{\varphi}^4(4) := D_t f = \partial_t^3 u, \quad \hat{\varphi}^5(4) := D_t^2 f = \partial_t^4 u \quad \text{and} \quad \hat{\varphi}^6(4) := D_t^3 f = \partial_t^5 u, \quad (16c)$$

where $\hat{\varphi}^1(4)$ and $\hat{\varphi}^2(4)$ are computed with (14), $\hat{\varphi}^3(4)$ with (10a), and $\{\hat{\varphi}^k(4)\}_{k=4}^6$ with (13).

5.3. Derivation of the FSC method

According to Section 5.1, the state of a system driven by a stochastic flow map of order M is:

$$u(t, \xi) = \sum_{j=0}^M \frac{(t - t_i)^j}{j!} \partial_t^j u(t_i, \xi) \quad \text{and} \quad \dot{u}(t, \xi) = \sum_{j=0}^M \frac{(t - t_i)^j}{j!} \partial_t^{j+1} u(t_i, \xi), \quad (17)$$

with the provision that the stochastic process u is analytic on the temporal domain.

From these two expressions, it can be seen that the state of the system has been decomposed effectively into deterministic and non-deterministic parts. That is, the deterministic part $(t - t_i)^j/j!$ consisting of a temporal function in \mathcal{T} , and the non-deterministic part $\partial_t^j u(t_i, \xi)$ consisting of a random function in \mathcal{Z} . If the set of functions associated with the non-deterministic part, i.e. $\{\partial_t^j u(t_i, \cdot)\}_{j=0}^{M+1}$, is orthogonalized with respect to the measure in \mathcal{Z} , then (17) can also be written in the following way:

$$u(t, \xi) = \sum_{j=1}^{M+2} u^j(t) \Psi_j(\xi) \quad \text{and} \quad \dot{u}(t, \xi) = \sum_{j=1}^{M+2} \dot{u}^j(t) \Psi_j(\xi). \quad (18)$$

Hence, if the space associated with the stochastic part of the solution space were to be spanned with $\{\Psi_j\}_{j=1}^{M+2}$, then $u(t, \cdot)$ and $\dot{u}(t, \cdot)$ would be elements of that space around $t = t_i$. However, since one cannot always guarantee that the constant functions are in such a space construction, we prefer to write (18) in the following final form instead:

$$u(t, \xi) = \sum_{j=0}^{M+2} u^j(t) \Psi_j(\xi) \quad \text{and} \quad \dot{u}(t, \xi) = \sum_{j=0}^{M+2} \dot{u}^j(t) \Psi_j(\xi), \quad (19)$$

where $\Psi_0 \equiv 1$ is the identically-equal-to-one function as per Definition 4.

Therefore, for a system driven by a stochastic flow map of order M , the maximum number of basis vectors to use in a simulation with FSC is bounded from above by $M + 3$. Hence, regardless of the dimensionality of the random space, the probability information of the system's state can be completely captured in $\mathcal{Z}^{[P]}$ if $P = M + 2$. It is for this reason that our FSC scheme is superior in terms of efficiency in comparison to mTD-gPC which uses a combination of full and total-order tensor products to construct a suitable basis for $\mathcal{Z}^{[P]}$ around $t = t_i$. We emphasize, however, that the FSC scheme does not address by itself the curse of dimensionality at the random-space level, since we still have the issue that the bigger the random space is ($d \gg 1$), the more difficult it is to compute the inner products accurately in (6) and (9). This is still an open area of research and there are several approaches available in the literature for dealing with this issue [32–36].

Moreover, to reduce the computational cost associated with orthogonalizing $M + 2$ basis vectors, it is sometimes convenient to start the FSC analysis with the smallest value for M (i.e. $M = 1$), and then to progressively increment it if more accurate results are desired for the simulation. Therefore, the minimum number of basis vectors to use in a simulation with FSC is bounded from below by 4.

5.4. FSC scheme

Suppose that a stochastic system such as (1) has been given. Let $\{\mathfrak{T}_i\}_{i=0}^{N-1}$ be a partition of the temporal domain, where $\mathfrak{T}_i \neq \emptyset$ represents the i -th interval of the partition, and define $s_{\cdot i} = s|_{\text{cl}(\mathfrak{T}_i) \times \Xi}$ to be the restriction of s to $\mathfrak{R}_i := \text{cl}(\mathfrak{T}_i) \times \Xi$. (Recall that $s = (u, \dot{u})$ represents the configuration state of the system over $\mathfrak{T} \times \Xi$.) Then, if the system is driven by a stochastic flow map of order M (Fig. 2), proceed as follows:

1. Loop across the temporal domain from $i = 0$ to $i = N - 1$.

(a) Define a solution representation for the configuration state $s_{\cdot i}$ in the following way.

- Take $\{\Phi_{j,i} := \hat{\varphi}^j(M)(0, \hat{s}(t_i, \cdot))\}_{j=1}^P$ to be an ordered set of linearly independent functions in \mathcal{Z} with $3 \leq P \leq M + 2$, and define $\Phi_{0,i} \equiv 1$. Observe that $\hat{\varphi}(M)(0, \hat{s}(t_i, \cdot)) \equiv \hat{s}_{\cdot i}(t_i, \cdot) = \hat{s}_{\cdot i-1}(t_i, \cdot)$ for $i \geq 1$. However, if $i = 0$, then $\hat{\varphi}(M)(0, \hat{s}(t_0, \cdot)) \equiv \hat{s}(0, \cdot)$. (Note: When the initial conditions are deterministic or linearly dependent, please see Remark 1.)

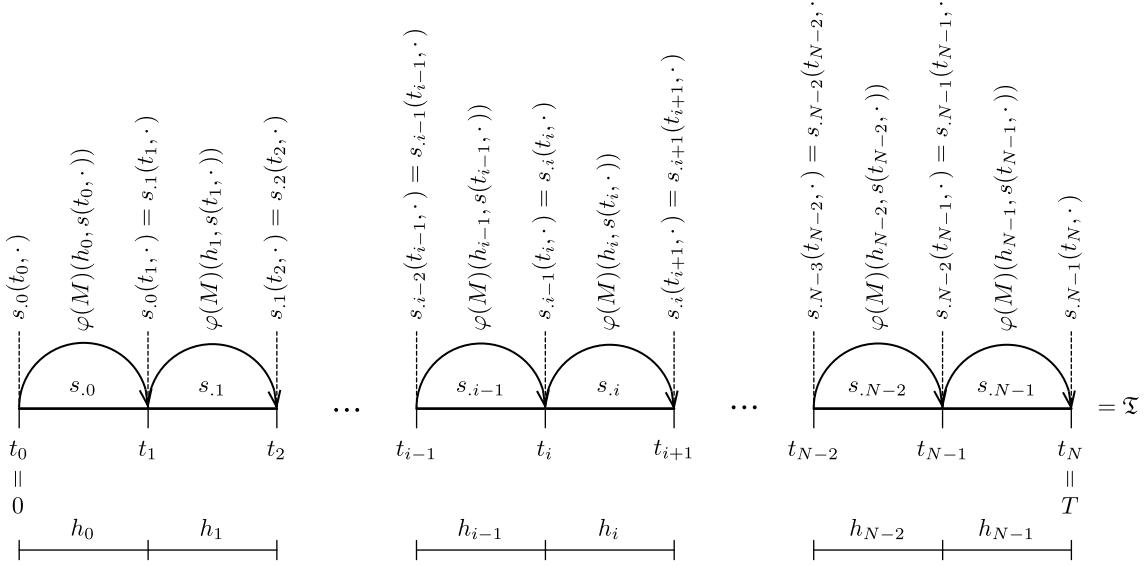


Figure 2: Evolution of a dynamical system via a stochastic flow map of order M , provided that h_i is taken sufficiently small and M sufficiently large so as to have $s(t_N, \cdot) \approx s_{N-1}(t_N, \cdot)$ at the end of the simulation. Then, as a means to avoid complexity in notation, we take $s(t_i + h, \cdot) \sim s_{i-1}(t_i + h, \cdot)$ for $h \geq 0$, and $s(t_i + h, \cdot) \sim s_{i-1}(t_i + h, \cdot)$ for $h \leq 0$

- Orthogonalize the set $\{\Phi_{j,i}\}_{j=0}^P$ using the Gram-Schmidt process [40], so that the resulting set $\{\Psi_{j,i}\}_{j=0}^P$ is an orthogonal basis in \mathfrak{Z} . That is, for $j \in \{0, 1, \dots, P\}$:

$$\Psi_{j,i} := \Phi_{j,i} - \sum_{k=0}^{j-1} \frac{\langle \Phi_{j,i}, \Psi_{k,i} \rangle}{\langle \Psi_{k,i}, \Psi_{k,i} \rangle} \Psi_{k,i}. \quad (20)$$

- Define $\mathfrak{Z}_i^{[P]} = \text{span}\{\Psi_{j,i}\}_{j=0}^P$ to be a p -discretization of \mathfrak{Z} over the region \mathfrak{R}_i . Since $\mathfrak{Z}_i^{[P]}$ is an evolving function space, expansion (3) is now to be read as:

$$u_{.i}(t, \xi) \approx u_{.i}^{[P]}(t, \xi) = \sum_{j=0}^P u_{.i}^j(t) \Psi_{j,i}(\xi) \equiv u_{.i}^j(t) \Psi_{j,i}(\xi). \quad (3^*)$$

Hence, $\dot{u}_{.i} = \partial_t u_{.i}$.

(b) Transfer the random modes of $s_{i-1} = (u_{.i-1}, \dot{u}_{.i-1})$ to $s_i = (u_{.i}, \dot{u}_{.i})$ at $t = t_i$, given that $i \geq 1$.

One way to achieve this is to ensure that the probability information of the system's state is transferred in the mean-square sense. Put differently, we wish to make sure that the equalities shown below hold in the mean-square sense (summation sign implied only over repeated index k):

$$u_{.i}(t_i, \xi) = u_{.i-1}(t_i, \xi) \iff u_{.i}^k(t_i) \Psi_{k,i}(\xi) = u_{.i-1}^k(t_i) \Psi_{k,i-1}(\xi) \quad (21a)$$

$$\dot{u}_{.i}(t_i, \xi) = \dot{u}_{.i-1}(t_i, \xi) \iff \dot{u}_{.i}^k(t_i) \Psi_{k,i}(\xi) = \dot{u}_{.i-1}^k(t_i) \Psi_{k,i-1}(\xi). \quad (21b)$$

Projecting (21) onto $\mathfrak{Z}_i^{[P]}$ gives:

$$u_{.i}^k(t_i) \langle \Psi_{j,i}, \Psi_{k,i} \rangle = u_{.i-1}^k(t_i) \langle \Psi_{j,i}, \Psi_{k,i-1} \rangle \quad (22a)$$

$$\dot{u}_{.i}^k(t_i) \langle \Psi_{j,i}, \Psi_{k,i} \rangle = \dot{u}_{.i-1}^k(t_i) \langle \Psi_{j,i}, \Psi_{k,i-1} \rangle. \quad (22b)$$

Thus, upon simplification yields the random modes of $s_{.i} = (u_{.i}, \dot{u}_{.i})$ at $t = t_i$:

$$u^j_{.i}(t_i) = \frac{\langle \Psi_{j.i}, \Psi_{k.i-1} \rangle}{\langle \Psi_{j.i}, \Psi_{j.i} \rangle} u^k_{.i-1}(t_i) \quad \text{and} \quad \dot{u}^j_{.i}(t_i) = \frac{\langle \Psi_{j.i}, \Psi_{k.i-1} \rangle}{\langle \Psi_{j.i}, \Psi_{j.i} \rangle} \dot{u}^k_{.i-1}(t_i), \quad (6b^*)$$

where $j \in \{0, 1, \dots, P\}$ (summation sign implied over repeated index k). These are to be interpreted as the initial conditions of the system over the region \mathfrak{R}_i .

If $i = 0$, the initial conditions are computed with (6b) directly.

- (c) Substitute (3*) into (1) to obtain (4).
- (d) Project (4a) onto $\mathfrak{X}_i^{[P]}$ to obtain (6a) subject to (6b*). Note that if $i = 0$, (6a) is subject to (6b).
- (e) Integrate (6) over time, as long as a suitable time integration method has been selected for solving the resulting system of equations. This step requires to find the random modes $\{u^j_{.i}(t_{i+1})\}_{j=0}^P$ and $\{\dot{u}^j_{.i}(t_{i+1})\}_{j=0}^P$.
- (f) Compute both the mean and the variance of each of the components of output $y = \mathcal{M}[u][x]$ over \mathfrak{R}_i , by recurring to the formulas prescribed by (8) and (9).

2. Post-process results.

Remark 1. Any of the following two approaches can be carried out at the start of the simulation ($i = 0$) to address the case when the initial conditions are deterministic or, more generally, linearly dependent:

- When the initial conditions are deterministic, the first two vectors are required to be removed from the set $\{\Phi_{j.i} := \hat{\varphi}^j(M)(0, \hat{s}(t_i, \cdot))\}_{j=1}^P$ for they are constant, and when the initial conditions are stochastic but linearly dependent, only one of them needs to be removed from the set.
- When the initial conditions are deterministic or linearly dependent, the gPC method [3, 27] can be employed instead to advance the state of the system one-time step forward; that is, from $s(t_0 = 0, \cdot)$ to $s(t_1, \cdot)$. After this, the gPC method can be switched over FSC to continue pushing the system's state forward in time.

Remark 2. Compared to the standard TD-gPC by Gerritsma et al. [22], in our FSC scheme we update the stochastic part of the solution space at every time step to minimize the error propagation over time. We do this without loss of generality since the scheme can conveniently be modified to incorporate a stopping criterion of one's choice and update the random basis only when the criterion is met.

6. Numerical results

We demonstrate and compare the performance of the FSC scheme to the mTD-gPC scheme using two numerical examples for the dynamical system described in Section 3. We also define the local error, $\epsilon : \mathcal{T} \rightarrow \mathcal{T}$, and the global error, $\epsilon_G : \mathcal{T} \rightarrow \mathbb{R}$, with the following expressions:

$$\epsilon[f](t) = |f(t) - f_{\text{exact}}(t)| \quad (23a)$$

$$\epsilon_G[f] = \frac{1}{T} \int_{\mathfrak{T}} |f(t) - f_{\text{exact}}(t)| dt \approx \frac{\Delta t}{T} \sum_{i=0}^N |f(t_i) - f_{\text{exact}}(t_i)|, \quad (23b)$$

where Δt is the time-step size used for the simulation, $t_i \in \mathfrak{T}$ is the time instant of the simulation, and N denotes the number of time steps employed in the simulation (with $t_0 = 0$ and $t_N = N \Delta t = T$).

Table 1: Case studies considered for a single-degree-of-freedom system under free vibration and stochastic stiffness

Case	Probability distribution*	Probability moments	Quadrature rule
1	Uniform $\sim \xi \in \Xi = [a, b]$ $a = 340 \text{ N/m}$, $b = 460 \text{ N/m}$	$\mathbf{E}[\xi] = 400 \text{ N/m}$ $\text{Var}[\xi] = 1200 \text{ N}^2/\text{m}^2$	Gauss-Legendre (100 points)
2	Beta(α, β) $\sim \xi \in \Xi = [a, b]$ $\alpha = 2$, $\beta = 5$ $a = 340 \text{ N/m}$, $b = 460 \text{ N/m}$	$\mathbf{E}[\xi] \approx 374.3 \text{ N/m}$ $\text{Var}[\xi] \approx 367.3 \text{ N}^2/\text{m}^2$	Gauss-Jacobi (95 points) $\alpha_J := \beta - 1 = 4$ $\beta_J := \alpha - 1 = 1$
3	Gamma(α, β) $\sim \xi \in \Xi = [a, \infty)$ $\alpha = 10$, $\beta = 1/10$ $a = 340 \text{ N/m}$	$\mathbf{E}[\xi] = 440 \text{ N/m}$ $\text{Var}[\xi] = 1000 \text{ N}^2/\text{m}^2$	Gauss-Laguerre (135 points) $\alpha_L := \alpha - 1 = 9$

*Probability density function, $f : \Xi \rightarrow \mathbb{R}_0^+$:

$$\text{Uniform} \sim f(\xi) = \frac{1}{b-a}, \quad \text{Beta} \sim f(\xi) = \frac{(\xi-a)^{\alpha-1} (b-\xi)^{\beta-1}}{(b-a)^{\alpha+\beta-1} B(\alpha, \beta)} \quad \text{and} \quad \text{Gamma} \sim f(\xi) = \frac{\beta^\alpha}{\Gamma(\alpha)} (\xi-a)^{\alpha-1} \exp(-\beta(\xi-a)).$$

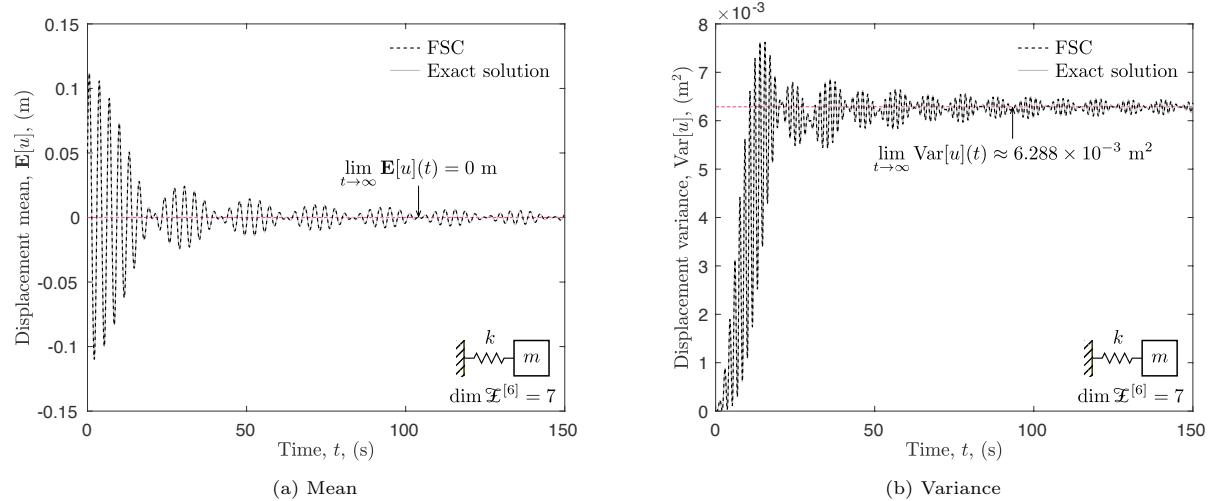


Figure 3: Evolution of $\mathbf{E}[u]$ and $\text{Var}[u]$ for the case when the p -discretization level of RFS is $\Xi^{[6]}$ and $\mu \sim \text{Uniform}$

6.1. Single-degree-of-freedom system under free vibration

We consider an undamped single-degree-of-freedom system governed by $m\ddot{u} + ku = 0$ with mass $m = 100 \text{ kg}$ and stochastic stiffness $k(\xi) = \xi$ subjected to free vibration. Three different cases of stochasticity are considered as listed in Table 1. The system has an initial displacement of $u(0, \cdot) \equiv 0.05 \text{ m}$, and an initial velocity of $\dot{u}(0, \cdot) \equiv 0.20 \text{ m/s}$. The duration of the simulation is set to $T = 150 \text{ s}$. To minimize the errors coming from the discretization of $\mathcal{T}(2)$, the time-step size is taken as $\Delta t = 0.005 \text{ s}$, meaning that a total of $N = 30\,000$ time steps are employed in the simulation. To integrate (6) over time⁴, we use the RK4 method, and as described in Section 5.4, the stochastic part of the solution space is updated at every time step in order to obtain accurate results. Lastly, because the initial conditions of the system are deterministic, we opt to employ the gPC method (with $P = 6$) [3] for the first 5 seconds of the simulation, in an effort to ensure that the stochasticity of the system's state is well developed for the analysis with FSC or mTD-gPC.

Figs. 3 and 4 show the evolution of the mean and variance of the system's state. From these figures, it is observed that the response obtained with FSC using only 7 basis vectors has the ability to reproduce

⁴In this problem we have taken $\mathcal{F}[u, \dot{u}] = ku$, and thus $\mathcal{F}^i[u^j, \dot{u}^j](t) = k^i_j u^j(t)$ with $k^i_j = \langle \Psi_i, k\Psi_j \rangle / \langle \Psi_i, \Psi_i \rangle$.

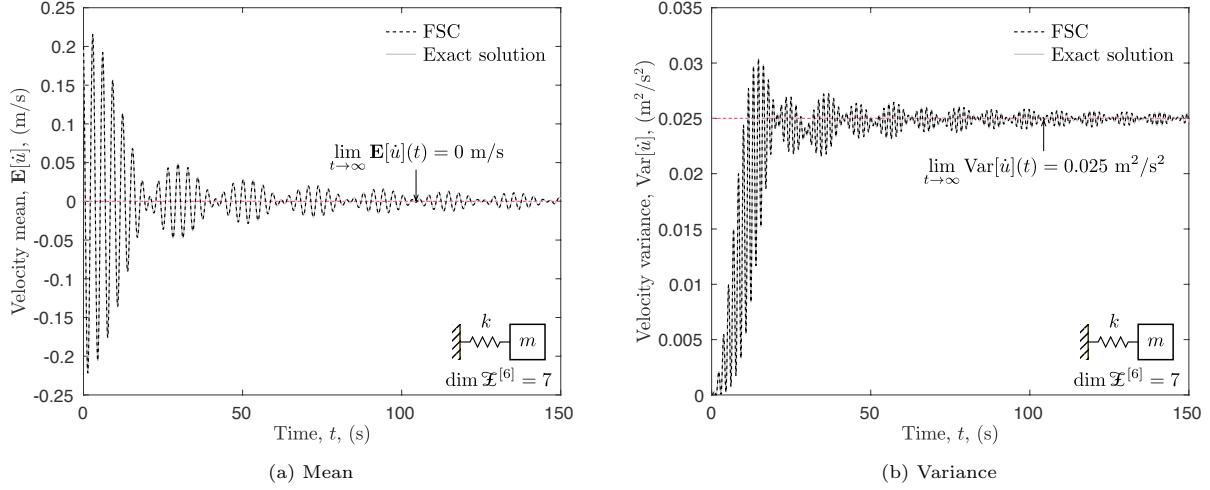


Figure 4: Evolution of $E[\dot{u}]$ and $\text{Var}[\dot{u}]$ for the case when the p -discretization level of RFS is $\mathbb{Z}^{[6]}$ and $\mu \sim \text{Uniform}$

the exact response (from Appendix B) with high fidelity. This is the reason why the two plots appear to be indistinguishable from each other. The figures also show the limit values for each response computed using the exact expressions given by (30) and (32).

Fig. 5 presents the local errors in mean and variance of the system's state using different choices of number of basis vectors ranging from 3 to 7. Note that even though the FSC scheme requires P to be greater than or equal to 3, here we also study the case when $P = 2$ for sake of comparison. Fig. 5 shows that as the number of basis vectors increases, so does the accuracy of the results. In particular, by increasing the number of basis vectors from 3 to 5, the accuracy of the results improves significantly by an order of magnitude of 6 (approximately from 10^{-1} to 10^{-7} for the mean), whereas when the number of basis vectors is increased from 5 to 7, the improvement in error is more moderate (to approximately 10^{-8} for the mean). However, for the 7-basis-vector case we do see an improvement in the accuracy of the solution as time progresses due to the increase in the number of basis vectors used. Fig. 6 presents the convergence of global errors as a function of the number of basis vectors and the different distributions used to define the stochasticity of k . The FSC scheme achieves exponential convergence when 3, 4 and 5 basis vectors are used, but adding more basis vectors does not improve the accuracy of the response. The primary reason for this slow-down in convergence is that the accuracy of the solution is limited by machine precision and the fact that the probability information is being transferred in the mean-square sense at every time step (for P is finite). In fact, these plots indicate that there is no reason to implement more than 5 basis vectors into the simulation as it does not improve the accuracy of the results significantly. It is also apparent from this figure that when k is assumed gamma-distributed, the results are not as accurate as those obtained from the uniform and beta distributions. The reason behind this outcome is that when k is gamma-distributed its support is unbounded, which from a numerical viewpoint leads to the dreaded case of unbounded basis vectors.

Fig. 7 plots the global errors as a function of computational cost⁵ of the FSC and mTD-gPC schemes expressed in terms of the wall-clock time taken to complete the computation. The implementation of both schemes was optimized as much as reasonably possible, and the labels $P2Q0$, $P2Q1$ and $P2Q2$ are defined in Ref. [23] (Pg. 45). The comparison of computational cost is shown here only for Case 1 of Table 1 (for sake of brevity), but similar trends are observed for cases 2 and 3 as well. Note that FSC is much faster

⁵All problems in this work were run in MATLAB R2016b [41] on a 2017 MacBook Pro with quad-core 3.1 GHz Intel Core i7 processor (hyper-threading technology enabled), 16 GB 2133 MHz LPDDR3 memory and 1 TB PCI-Express SSD storage (APFS-formatted), running macOS Mojave (version 10.14.6).

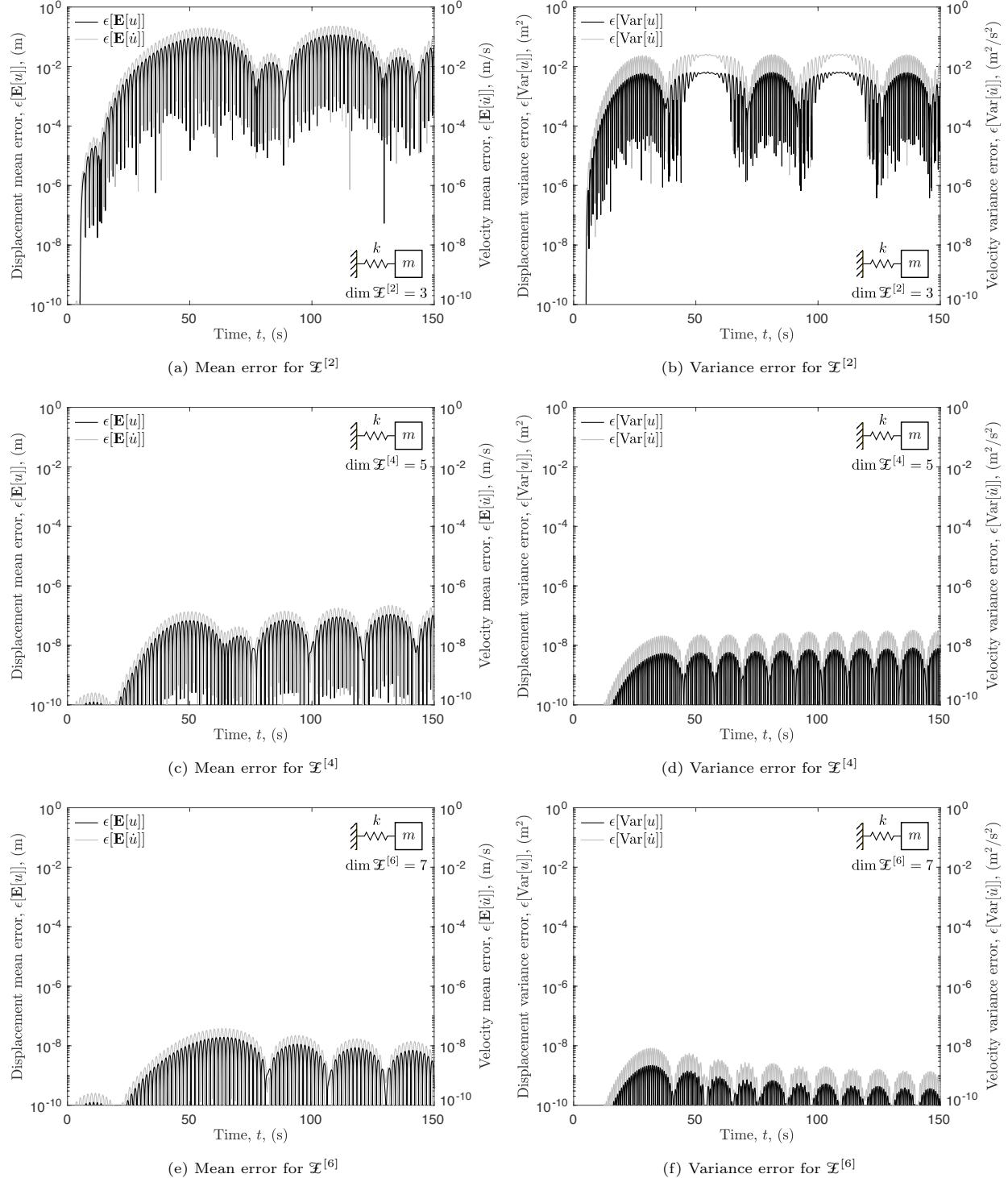


Figure 5: Local error evolution of $\mathbf{E}[u]$, $\text{Var}[u]$, $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for different p -discretization levels of RFS and for $\mu \sim \text{Uniform}$

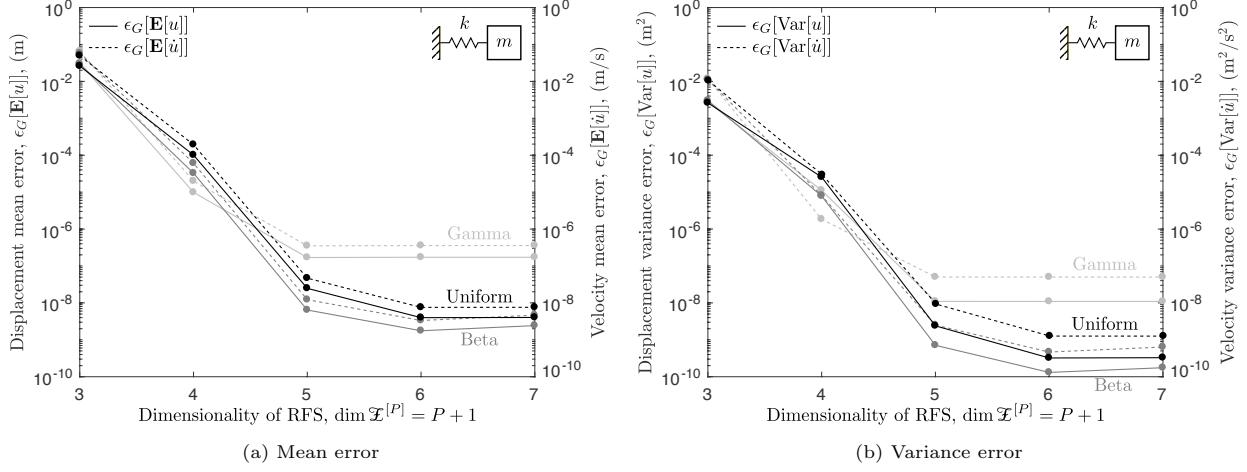


Figure 6: Global error of $\mathbf{E}[u]$, $\text{Var}[u]$, $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for different p -discretization levels of RFS

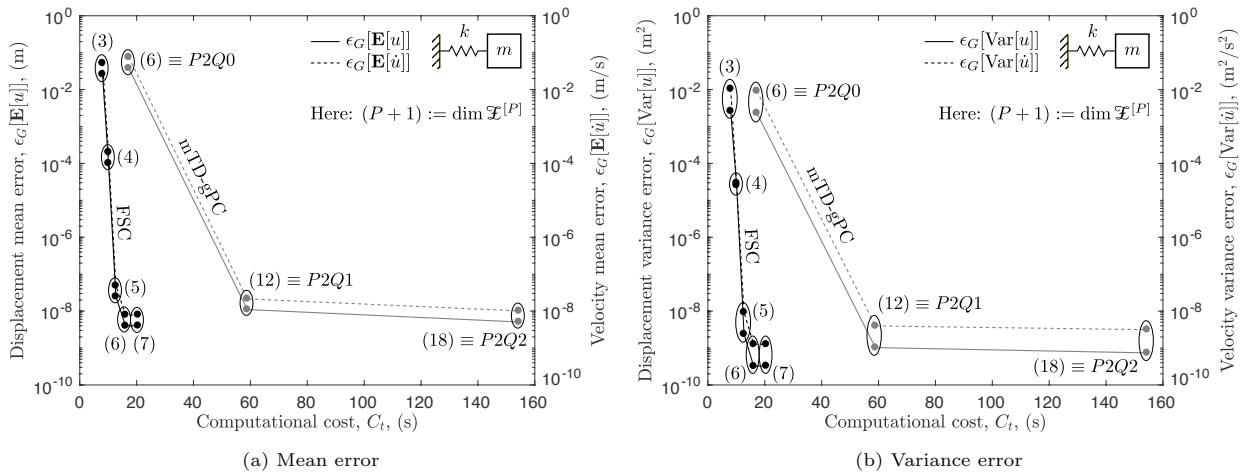


Figure 7: Global error versus computational cost for $\mu \sim \text{Uniform}$

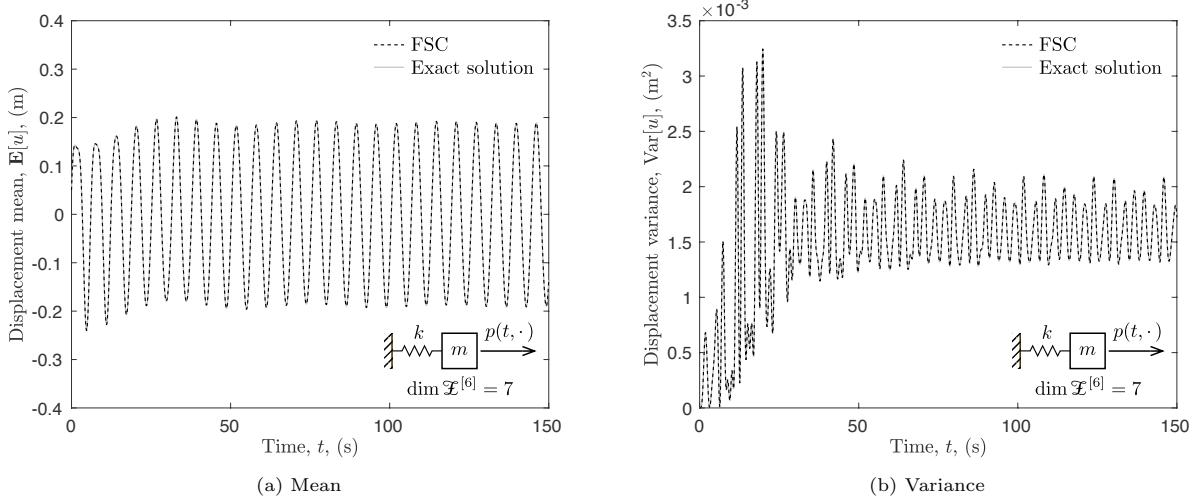


Figure 8: Evolution of $\mathbf{E}[u]$ and $\text{Var}[u]$ for the case when the p -discretization level of RFS is $\mathcal{E}^{[6]}$ and $\mu \sim \text{Uniform} \otimes \text{Beta}$

in comparison to mTD-gPC for a similar level of error. For instance, in order to attain a global error of approximately 10^{-8} , FSC runs about 3.5 times faster than mTD-gPC. This is because, in general, FSC requires much fewer basis vectors than mTD-gPC—to achieve an error of about 10^{-8} , FSC requires only 6 basis vectors in comparison to 12 for mTD-gPC. Another reason for the superior efficiency of FSC is that, for mTD-gPC, the orthogonalization process needs to be conducted three times when the random basis is demanded to be updated during the simulation (one time for the monomials of u , another time for the monomials of \dot{u} , and one more time after performing the tensor product)⁶. This also explains why using 6 basis vectors in both methods, FSC runs slightly faster than mTD-gPC. Furthermore, we see that the probability information is better encoded in FSC because it uses a fewer number of basis vectors to achieve the same level of accuracy. Fig. 7 also reveals that increasing the number of basis vectors from 6 to 7 for FSC and 12 to 18 for mTD-gPC does not improve the accuracy of the results significantly. This, again, is because of the limited precision of the machine and the fact that the probability information is being transferred in the mean-square sense.

6.2. Single-degree-of-freedom system under forced vibration

In this example, we show that the number of basis vectors needed in the simulation does not increase when the dimensionality of the random space increases. For this, we consider the same system described in the Example 6.1 (including the same deterministic initial conditions), with the only difference being that the system is subjected to a stochastic external force given by $p(t, \cdot) = q \sin(t)$. That is, the system is now governed by $m\ddot{u} + ku = p$. Here the stiffness $k(\xi) = \xi^1$ is taken to be the same as Case 1 of Table 1 with $\bar{\Xi}_1 = [340, 460]$ N/m. The amplitude of the external force $q(\xi) = \xi^2$ is assumed beta-distributed with parameters $\alpha = 2$ and $\beta = 5$ in $\bar{\Xi}_2 = [51, 69]$ N, giving thereby probability moments: $\mathbf{E}[\xi^2] \approx 56.14$ N and $\text{Var}[\xi^2] \approx 8.265$ N². Because two random variables are present in the mathematical model, the random domain of the system is 2-dimensional, and thus, it is defined by $\Xi = \bar{\Xi}_1 \times \bar{\Xi}_2$ with $\mu \sim \text{Uniform} \otimes \text{Beta}$. For this example, the inner products are computed using a quadrature rule constructed by performing a cartesian product between 100 Gauss-Legendre points distributed along the ξ^1 -axis and 95 Gauss-Jacobi points distributed along the ξ^2 -axis. Finally, the gPC method (with $P = 8$) is used for the first 0.5 seconds of the simulation to allow the stochasticity of the system to develop sufficiently before using the FSC scheme.

⁶We tested the method by orthogonalizing only once—namely, after performing the tensor product between the monomials of u and \dot{u} —, and found that the accuracy of the results degrades noticeably.

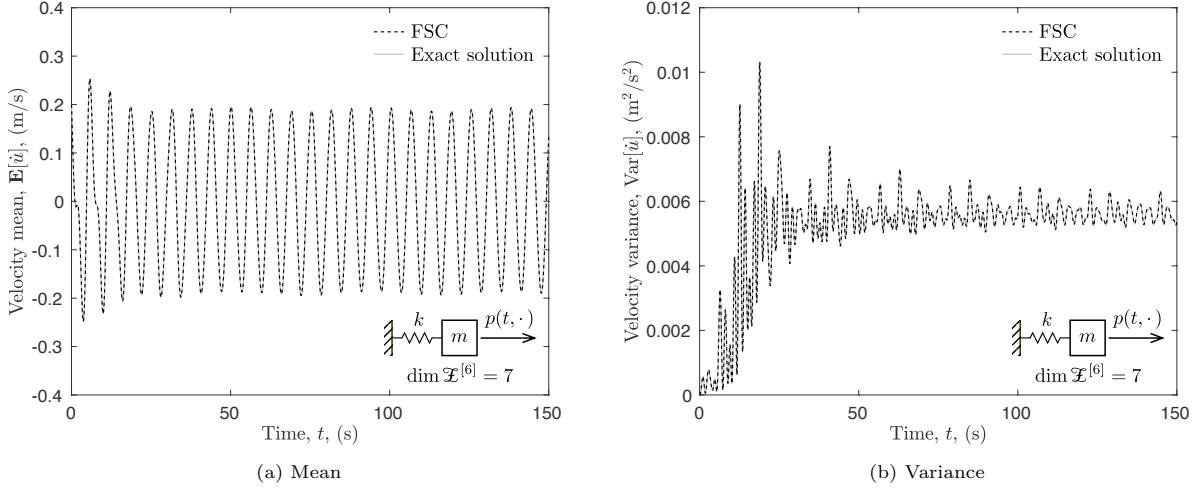


Figure 9: Evolution of $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for the case when the p -discretization level of RFS is $\Xi^{[6]}$ and $\mu \sim \text{Uniform} \otimes \text{Beta}$

Figs. 8 and 9 show the evolution of the mean and variance of the system's state. As in the previous example, the numerical solution obtained using FSC with 7 basis vectors is indistinguishable from the exact response⁷. Figs. 10 and 11 depict the local and global errors in mean and variance of the system's state. Here we also notice the same trend found in Figs. 5 and 6. That is, as the number of basis vectors increases, so does the accuracy of the results. Moreover, when the number of basis vectors increases from 3 to 5, the error for the mean drops down from approximately 10^{-1} to 10^{-7} , whereas increasing the number of basis vectors from 5 to 7 does not result in a noticeable improvement in the computation of the probability moments. Note that using the same number of basis vectors as in Example 6.1 led to similar levels of error, even though the dimensionality of the random space in this example is twice that of Example 6.1. However, despite the number of basis vectors being the same, the computational cost of this example is much higher than that of Example 6.1 (as per Fig. 12) because of the increase in the number of quadrature points needed to compute the inner products accurately.

6.3. Nonlinear single-degree-of-freedom system under free vibration

In this last example, we explore the nonlinear behavior of a single-degree-of-freedom system in order to test the ability of the FSC method to solve nonlinear problems. The governing differential equation for this system is set to be given by

$$m\ddot{u} + (1 + \rho u^2)ku = 0, \quad (24)$$

where $m = 100$ kg is the mass of the system, $k(\xi) = \xi^1$ is a beta-distributed random variable representing the strength of the stiffness which is given by Case 2 of Table 1 with $\bar{\Xi}_1 = [340, 460]$ N/m, and $\rho(\xi) = \xi^2$ is a uniformly-distributed random variable denoting the contributing factor to the nonlinearity of the system in $\bar{\Xi}_2 = [-20, -30]$ m⁻². The probability moments for ξ^2 are thus: $\mathbf{E}[\xi^2] = -25$ m⁻² and $\text{Var}[\xi^2] \approx 8.333$ m⁻⁴. The system has an initial displacement of $u(0, \cdot) \equiv 0.05$ m and an initial velocity of $\dot{u}(0, \cdot) \equiv 0.20$ m/s. As in the previous example, the random domain of the system is two-dimensional and defined by $\Xi = \bar{\Xi}_1 \times \bar{\Xi}_2$ with $\mu \sim \text{Beta} \otimes \text{Uniform}$. The inner products are again computed with a Gaussian quadrature rule using the same number of points indicated in the previous section, and the gPC method (with $P = 8$) is used for the first second of the simulation. The system is integrated over time using the RK4 method with a time-step size of $\Delta t = 0.005$ s, and the simulation is set to last $T = 150$ s.

⁷To obtain the ‘exact’ solution for $\mathbf{E}[u]$, $\text{Var}[u]$, $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$, the corresponding values for $\mathbf{E}[u(t_i, \cdot)]$, $\text{Var}[u(t_i, \cdot)]$, $\mathbf{E}[\dot{u}(t_i, \cdot)]$ and $\text{Var}[\dot{u}(t_i, \cdot)]$ were computed at each instant of time $t_i \in \mathfrak{T}$ using the `vpaintegral` command (provided in the MATLAB’s Symbolic Math Toolbox [41]) with `RelTol` set equal to 10^{-14} . The exact displacement response, u , is well known and can be found in any structural dynamics textbook, e.g. [39, 42].

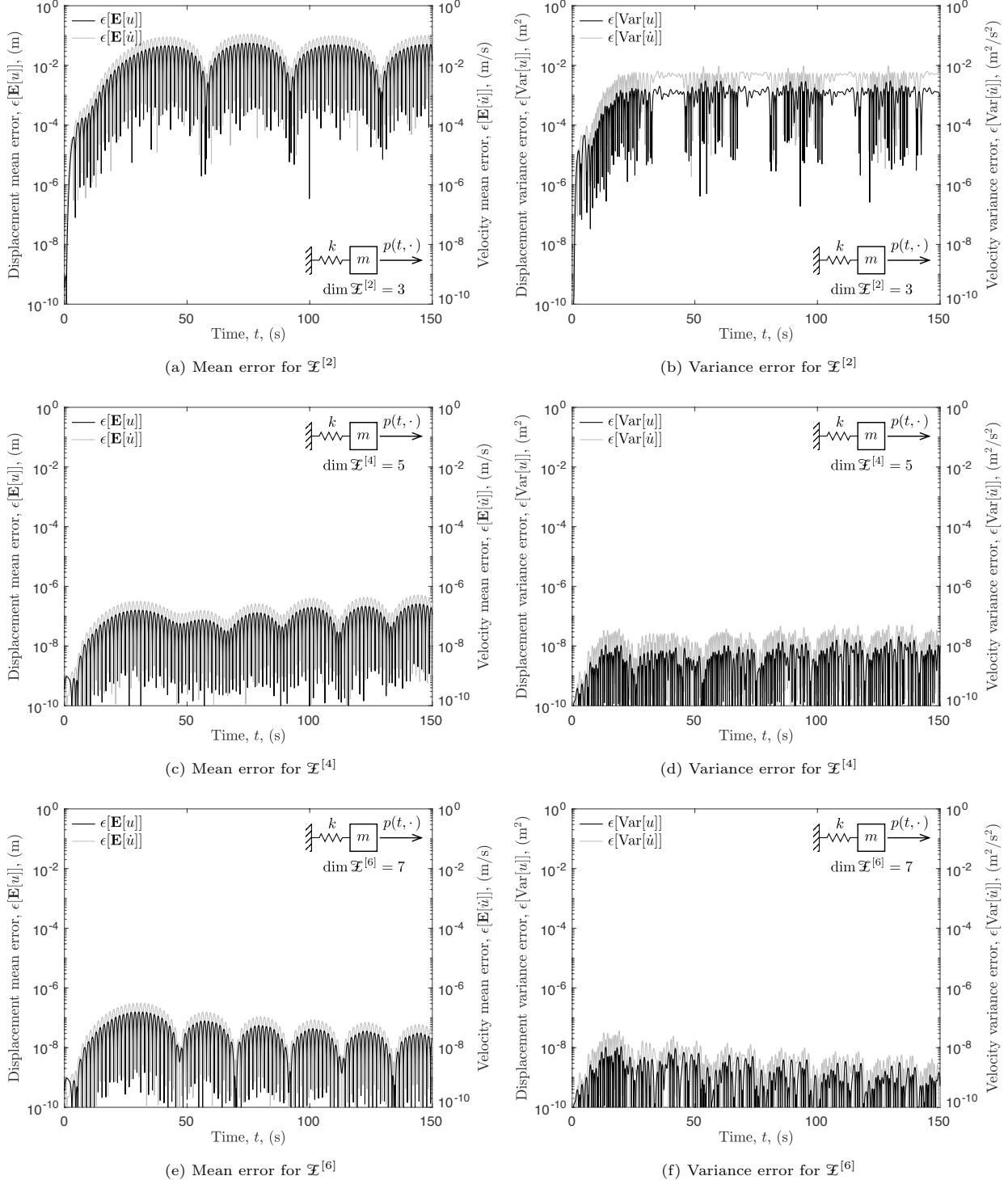


Figure 10: Local error evolution of $\mathbf{E}[u]$, $\text{Var}[u]$, $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for different p -discretization levels of RFS and for $\mu \sim \text{Uniform} \otimes \text{Beta}$

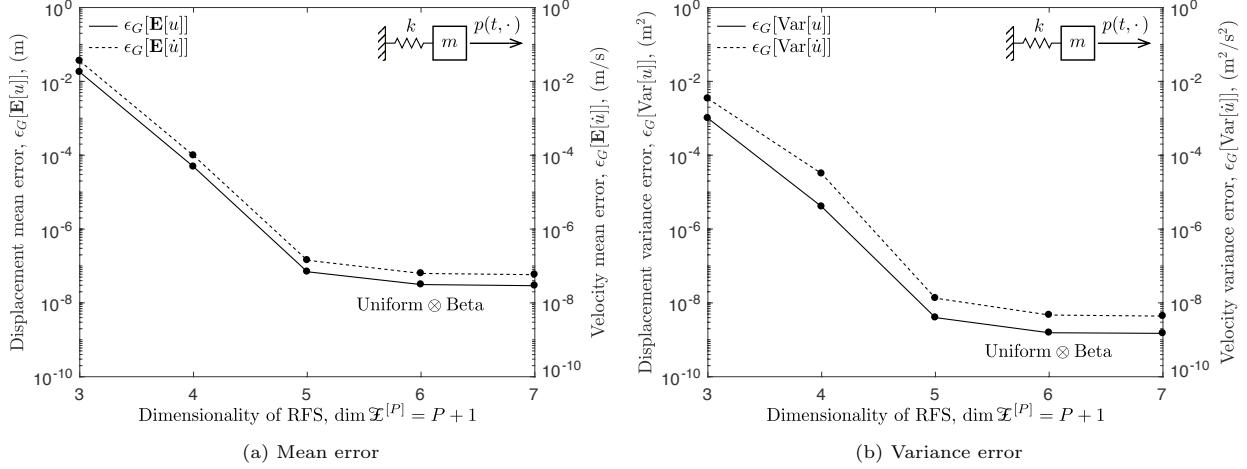


Figure 11: Global error of $\mathbf{E}[u]$, $\text{Var}[u]$, $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for different p -discretization levels of RFS

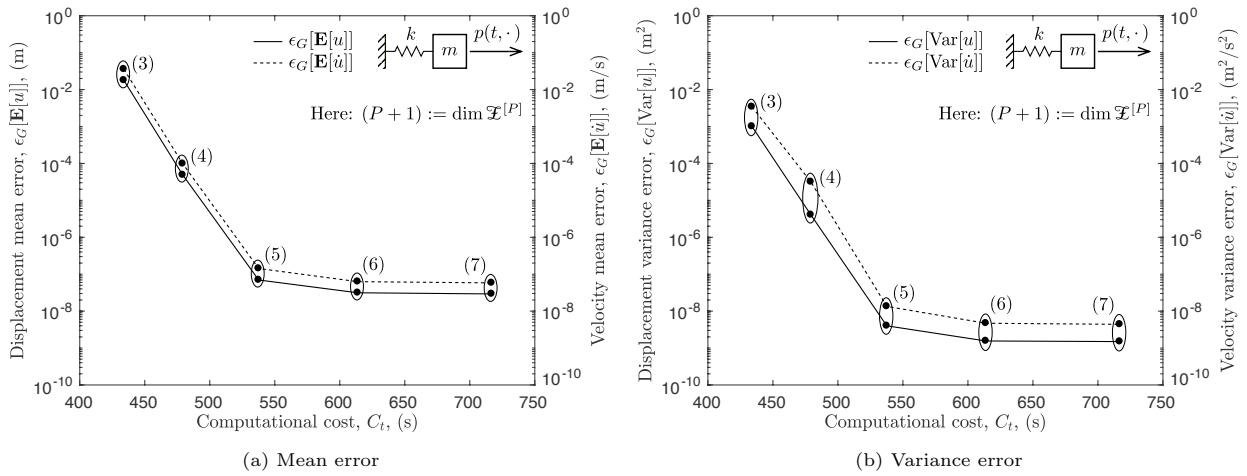


Figure 12: Global error versus computational cost for $\mu \sim \text{Uniform} \otimes \text{Beta}$

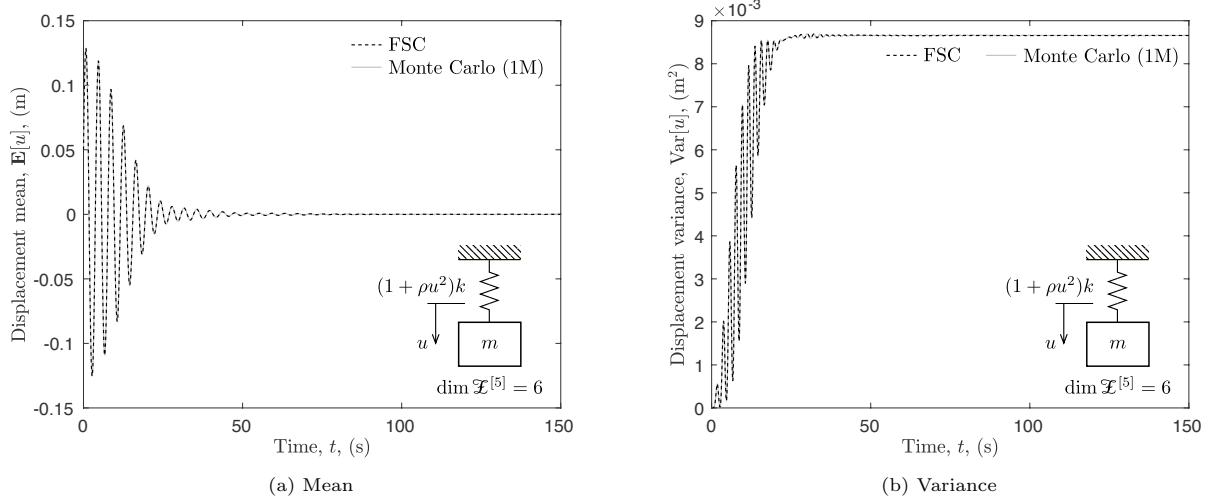


Figure 13: Evolution of $\mathbf{E}[u]$ and $\text{Var}[u]$ for the case when the p -discretization level of RFS is $\mathbb{Z}^{[5]}$ and $\mu \sim \text{Beta} \otimes \text{Uniform}$

Remark 3. According to (6), the temporal function $\mathcal{F}^i[u^j, \dot{u}^j]$ associated with $\mathcal{F}[u, \dot{u}] = (1 + \rho u^2)ku$ is given by:

$$\mathcal{F}^i[u^j, \dot{u}^j](t) = \frac{\langle \Psi_i, k\Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} u^j(t) + \frac{\langle \Psi_i, \rho k\Psi_j\Psi_k\Psi_l \rangle}{\langle \Psi_i, \Psi_i \rangle} u^j(t) u^k(t) u^l(t).$$

Note that a summation sign is implied over every repeated index j , k and l .

Since a closed-form solution for (24) does not exist, we use the standard Monte Carlo method described in Appendix C to compare the accuracy of the FSC results against it. To this end, one million realizations are randomly sampled from the random domain to conduct the Monte Carlo simulation. The evolution of the mean and variance of the system's displacement is depicted in Figs. 13 and 14. These figures show that by using only 5 basis vectors, the FSC results can effectively reproduce the Monte Carlo results. In fact, Figs. 15 and 16 further indicate that when $P = 4, 5$, the FSC results are 4-order-of-magnitude accurate for the mean and about 5-order-of-magnitude accurate for the variance. However, this is not the case for $P = 3$, which overall is one order of magnitude less accurate and displays a nearly linear drift after 50 s. Moreover, it can be seen that the FSC results with $P = 4$ and $P = 5$ are indistinguishable from each other, chiefly because the one-million Monte Carlo simulation used as the reference solution is an approximate version of the solution. This explains why the accuracy of the results did not increase when P was set equal to 5. Therefore, comparable results are achievable for this nonlinear problem if FSC is run with $P = 4$.

7. Application to structural dynamics

In structural dynamics, real-life systems are commonly modeled as multiple-degree-of-freedom systems. In order to demonstrate how FSC can be utilized in a more general setting, in this section we quantify the response uncertainties of a 3-story building (Fig. 17a) excited by the effects of a ground motion. The ground motion is taken here to be one of the ground accelerations recorded from the 1940 *El Centro Earthquake*⁸ event. A plot of this ground motion is depicted in Fig. 17b for sake of reference.

The governing differential equation of motion for this system is:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = -\mathbf{M}\boldsymbol{\epsilon}\dot{\mathbf{u}}_g \quad (= \mathbf{p}), \quad (25)$$

⁸This ground acceleration was obtained from the PEER Ground Motion Database [43]. Website: <https://ngawest2.berkeley.edu>. Event's name: Imperial Valley-02. Station's name: El Centro Array #9. File's name: RSN6_IMPVALI.I.I-ELC180.AT2.

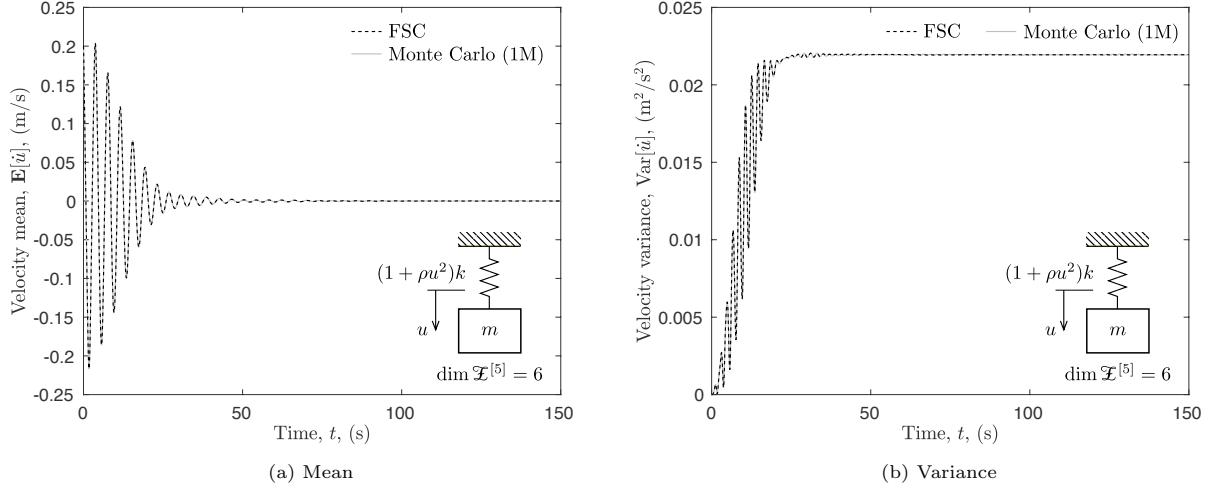


Figure 14: Evolution of $\mathbf{E}[\dot{u}]$ and $\text{Var}[\dot{u}]$ for the case when the p -discretization level of RFS is $\mathcal{Z}^{[5]}$ and $\mu \sim \text{Beta} \otimes \text{Uniform}$

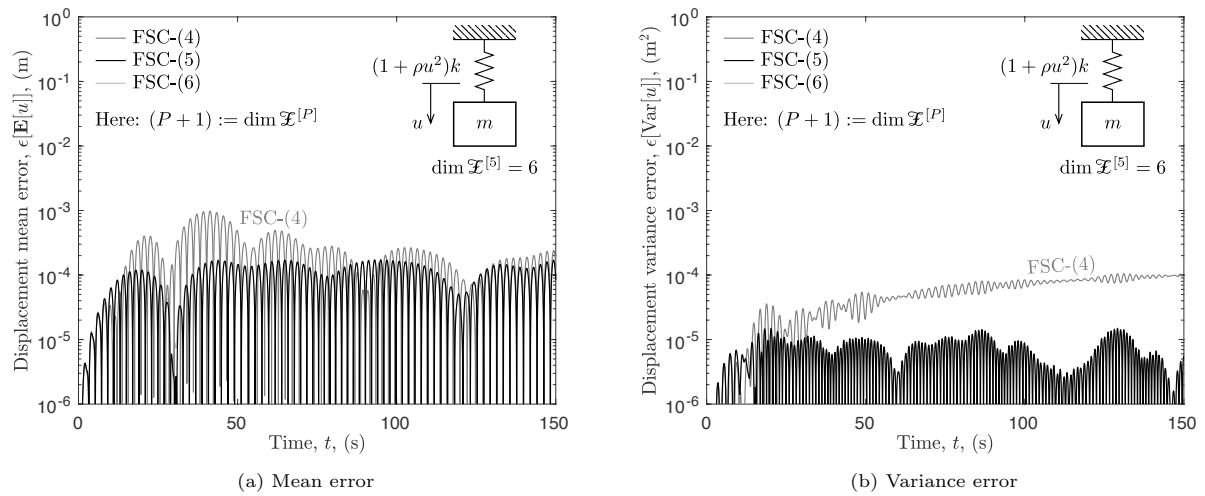


Figure 15: Local error evolution of $\mathbf{E}[u]$ and $\text{Var}[u]$ for different p -discretization levels of RFS with respect to the 1-million Monte Carlo simulation ($\mu \sim \text{Beta} \otimes \text{Uniform}$)

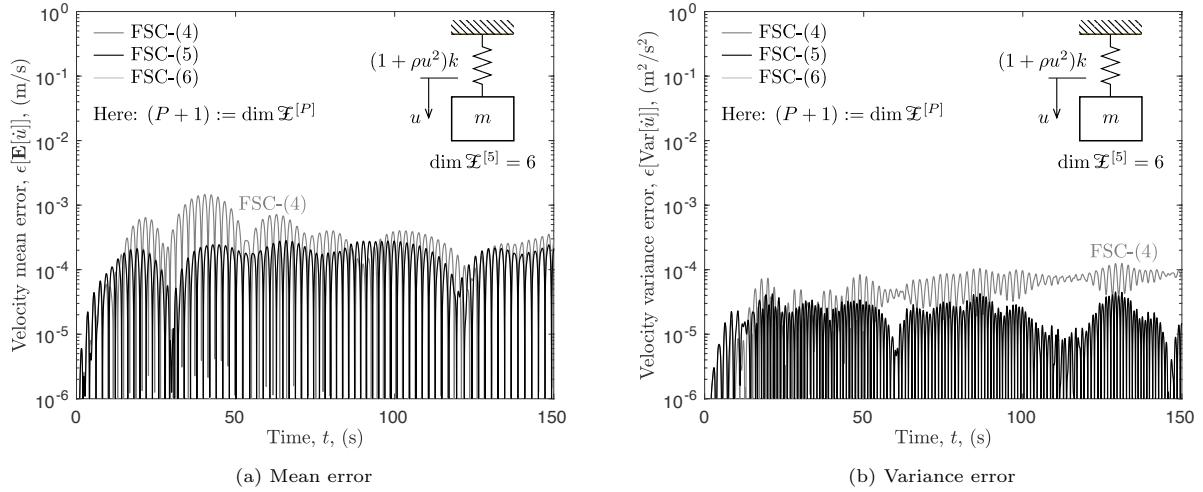
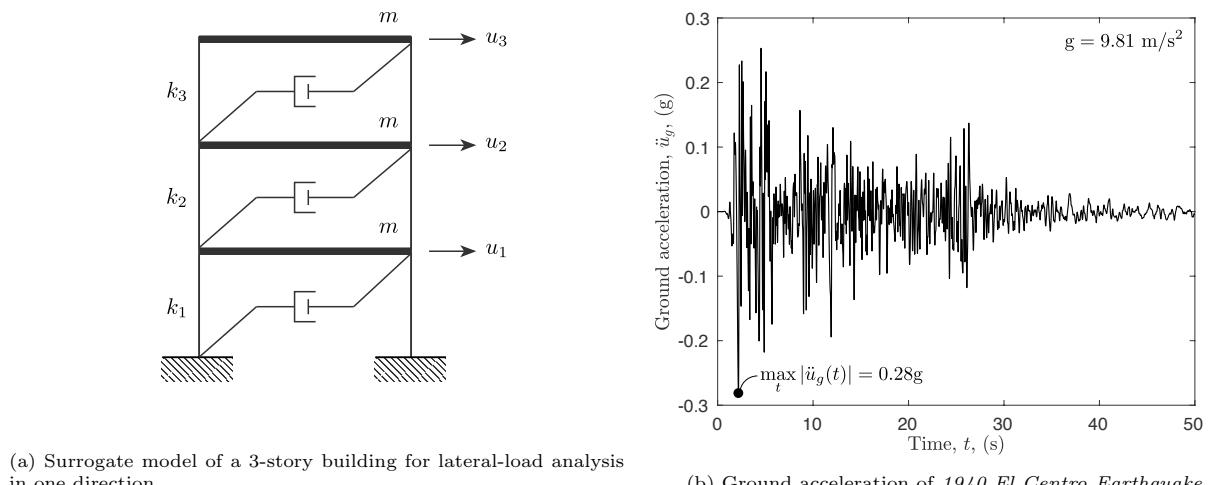


Figure 16: Local error evolution of $E[\dot{u}]$ and $\text{Var}[\dot{u}]$ for different p -discretization levels of RFS with respect to the 1-million Monte Carlo simulation ($\mu \sim \text{Beta} \otimes \text{Uniform}$)



(a) Surrogate model of a 3-story building for lateral-load analysis in one direction

(b) Ground acceleration of 1940 *El Centro* Earthquake

Figure 17: Surrogate model of a 3-story building to investigate its lateral behavior under an earthquake scenario

where $\mathbf{M} \in \mathbf{L}(\mathbb{R}^3, \mathbb{R}^3)$ is the mass matrix, $\mathbf{C} : \Xi \rightarrow \mathbf{L}(\mathbb{R}^3, \mathbb{R}^3)$ is the damping matrix, $\mathbf{K} : \Xi \rightarrow \mathbf{L}(\mathbb{R}^3, \mathbb{R}^3)$ is the stiffness matrix, $\boldsymbol{\iota} \in \mathbb{R}^3$ is the influence vector, and $\ddot{u}_g : \mathfrak{T} \rightarrow \mathbb{R}$ is the ground acceleration characterized by a real-valued function of time. The vectors $\mathbf{u}, \dot{\mathbf{u}} := \partial_t \mathbf{u}, \ddot{\mathbf{u}} := \partial_t^2 \mathbf{u} : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}^3$ represent, respectively, the displacement, the velocity and the acceleration of the system, where $\mathbf{u}^T = [u_1, u_2, u_3]$ is the unknown vector sought, and u_3 denotes the roof displacement of the 3-story building (our response of interest here).

The parameters of this system are defined as

$$\mathbf{M} = m \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix}, \quad \mathbf{K}(\xi) = \begin{bmatrix} k_1(\xi) + k_2(\xi) & -k_2(\xi) & \\ -k_2(\xi) & k_2(\xi) + k_3(\xi) & -k_3(\xi) \\ & -k_3(\xi) & k_3(\xi) \end{bmatrix},$$

and $\mathbf{C}(\xi) = \alpha(\xi) \mathbf{M} + \beta(\xi) \mathbf{K}(\xi)$, where $m = 500 \text{ Mg}$, $k_1(\xi) = \xi^1 \sim \text{Beta}(2, 5)$ in $[850 \times 10^3, 1150 \times 10^3] \text{ kN/m}$, $k_2(\xi) = \xi^2 \sim \text{Beta}(2, 5)$ in $[680 \times 10^3, 920 \times 10^3] \text{ kN/m}$, $k_3(\xi) = \xi^3 \sim \text{Beta}(2, 5)$ in $[680 \times 10^3, 920 \times 10^3] \text{ kN/m}$, $\alpha(\xi) = \xi^4 \sim \text{Uniform}$ in $[0.4, 0.7] \text{ s}^{-1}$, and $\beta(\xi) = \xi^5 \sim \text{Uniform}$ in $[0.4 \times 10^{-3}, 0.7 \times 10^{-3}]$. Furthermore, $\boldsymbol{\iota}^T = [1, 1, 1]$, and \ddot{u}_g is defined according to Fig. 17b. The random domain for this system is thus 5-dimensional:

$$\Xi = \prod_{i=1}^5 \bar{\Xi}_i \equiv [850 \times 10^3, 1150 \times 10^3] \times [680 \times 10^3, 920 \times 10^3]^2 \times [0.4, 0.7] \times [0.4 \times 10^{-3}, 0.7 \times 10^{-3}],$$

and we assume that the initial state of the system is at rest, i.e. $\mathbf{u}(0) = \dot{\mathbf{u}}(0) = \mathbf{0}$. Note that when the expected values of k_1 , k_2 and k_3 are utilized to define \mathbf{K} , the fundamental period of the system is approximately 0.33 s, which is consistent with a typical 3-story building found in practice featuring a damping ratio of about 2% for the first two modal frequencies.

Following the formulation presented in Section 4, the system of equations (6) takes the form:

$$\mathbf{M}^i_j \ddot{\mathbf{u}}^j + \mathbf{C}^i_j \dot{\mathbf{u}}^j + \mathbf{K}^i_j \mathbf{u}^j = \mathbf{p}^i \quad \text{on } \mathfrak{T} = [0, 50] \text{ s} \quad (26a)$$

$$\{\mathbf{u}^i(0) = \mathbf{0}, \dot{\mathbf{u}}^i(0) = \mathbf{0}\} \quad \text{on } \{0\}, \quad (26b)$$

where a summation sign is implied over the repeated index j , $\mathbf{M}^i_j = \mathbf{M} \delta^i_j$, $\mathbf{p}^i(t) = -\mathbf{M} \boldsymbol{\iota} \ddot{u}_g(t) \delta^i_0$,

$$\begin{aligned} \mathbf{C}^i_j = m \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix} \frac{\langle \Psi_i, \alpha \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} + \begin{bmatrix} 1 & & \\ & 0 & \\ & & 0 \end{bmatrix} \frac{\langle \Psi_i, \beta k_1 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} \\ + \begin{bmatrix} 1 & -1 & \\ -1 & 1 & \\ & & 0 \end{bmatrix} \frac{\langle \Psi_i, \beta k_2 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} + \begin{bmatrix} 0 & & \\ & 1 & -1 \\ & -1 & 1 \end{bmatrix} \frac{\langle \Psi_i, \beta k_3 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle}, \end{aligned}$$

and

$$\mathbf{K}^i_j = \begin{bmatrix} 1 & & \\ & 0 & \\ & & 0 \end{bmatrix} \frac{\langle \Psi_i, k_1 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} + \begin{bmatrix} 1 & -1 & \\ -1 & 1 & \\ & & 0 \end{bmatrix} \frac{\langle \Psi_i, k_2 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle} + \begin{bmatrix} 0 & & \\ & 1 & -1 \\ & -1 & 1 \end{bmatrix} \frac{\langle \Psi_i, k_3 \Psi_j \rangle}{\langle \Psi_i, \Psi_i \rangle}$$

with $i, j \in \{0, 1, \dots, P\}$, and δ^i_j denoting the Kronecker delta.

For this problem, we take $P = 9$ for the set of linearly independent functions $\{\Phi_{j,i} := \hat{\varphi}^j(0, \hat{\mathbf{s}}(t_i, \cdot))\}_{j=1}^P$, where $\hat{\varphi} : \mathbb{R} \times \mathfrak{X}^9 \rightarrow \mathfrak{X}^9$ is given by

$$\hat{\varphi}(h, \hat{\mathbf{s}}(t_i, \cdot)) =: \hat{\mathbf{s}}(t_i + h, \cdot) = (\mathbf{u}(t_i + h, \cdot), \dot{\mathbf{u}}(t_i + h, \cdot), \ddot{\mathbf{u}}(t_i + h, \cdot)),$$

and $\hat{\mathbf{s}} = (\mathbf{u}, \dot{\mathbf{u}}, \ddot{\mathbf{u}}) \equiv (u_1, u_2, u_3, \dot{u}_1, \dot{u}_2, \dot{u}_3, \ddot{u}_1, \ddot{u}_2, \ddot{u}_3)$.

Remark 4. Notice that this P is the smallest value we can choose for a problem featuring three degrees of freedom and whose governing stochastic differential equation is of second-order in time. This is because for

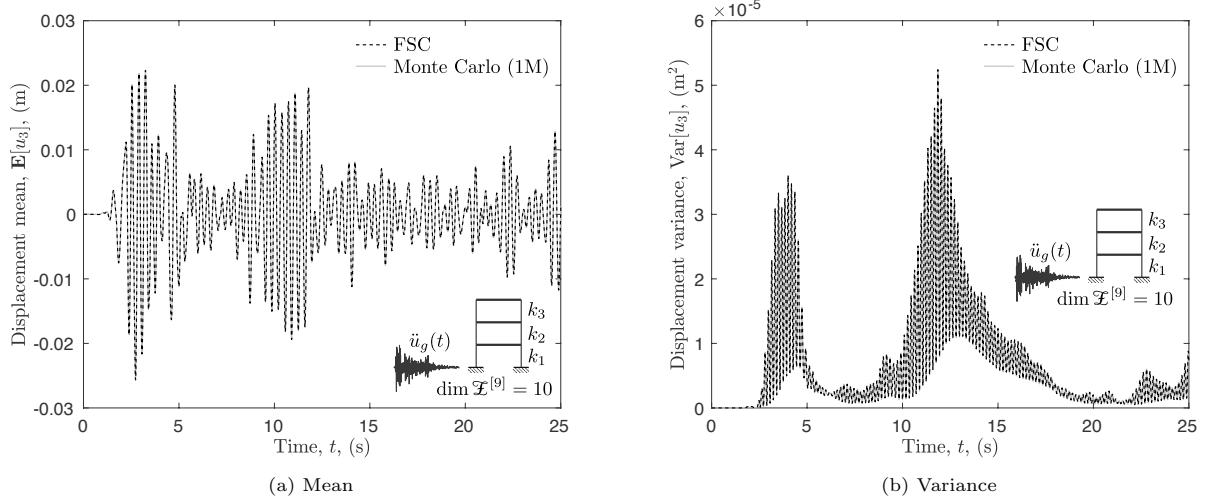


Figure 18: Evolution of $\mathbf{E}[u_3]$ and $\text{Var}[u_3]$ for the case when the p -discretization level of RFS is $\mathbb{Z}^{[9]}$ and $\mu \sim \text{Beta}^{\otimes 3} \otimes \text{Uniform}^{\otimes 2}$

each degree of freedom, the smallest RFS that one can construct using the FSC method is one whose P is equal to 3. However, we emphasize that this issue is not particular to the FSC method. It is well recognized that the more degrees of freedom a dynamical system has, the more basis vectors needed to construct a suitable random function space for the system’s state at any given time. For example, in TD-gPC-based methods, this would be equivalent to perform a full tensor product between all the RFS’s generated at each degree of freedom.

To integrate (26) numerically, we employ the RK4 method with a time-step size of $\Delta t = 0.01$ s (which is concordant with the sample frequency of the ground acceleration record). To evaluate the inner products approximately, we use 15 Gaussian quadrature points on each random axis, resulting in $15^5 = 759\,375$ quadrature points distributed across the entire random domain.

The results in Figs. 18 and 19 depict the solutions obtained from employing FSC and the standard Monte Carlo method (as described in Appendix C) to quantify the uncertainties of the response. One million realizations were randomly sampled from the random domain to conduct the Monte Carlo simulation. Once again, it is apparent that FSC is able to capture the system’s uncertainties with high fidelity and its solution is indistinguishable from that of the Monte Carlo method. For clarity, only the first 25 s of the solution are presented in Figs. 18 and 19, however, the conclusion drawn above applies to the last 25 s of the solution as well.

8. Conclusion

A novel numerical method, called the *flow-driven spectral chaos* (FSC) method, is presented for capturing uncertainties in structural dynamics using the spectral approach. The FSC method uses the concept of enriched stochastic flow maps to track the evolution of the system’s state efficiently in an augmented random phase space. The method is not only computationally more efficient than the TD-gPC approach but also easy to implement, since the flow map that we use in the scheme is nothing but the time derivatives of the solution up to a specific order. Moreover, since the random basis is defined with these time derivatives, the number of basis vectors required to characterize the stochastic part of the solution space does not depend upon the dimensionality of the probability space. This remarkable property opens up the possibility of investigating systems with high-dimensional probability spaces at low computational cost—an issue that has plagued the spectral approach since the introduction of the PC method.

The three numerical examples presented in Section 6 show that the FSC scheme is able to capture the response of the system with high accuracy using a small number of basis vectors and at a relatively low

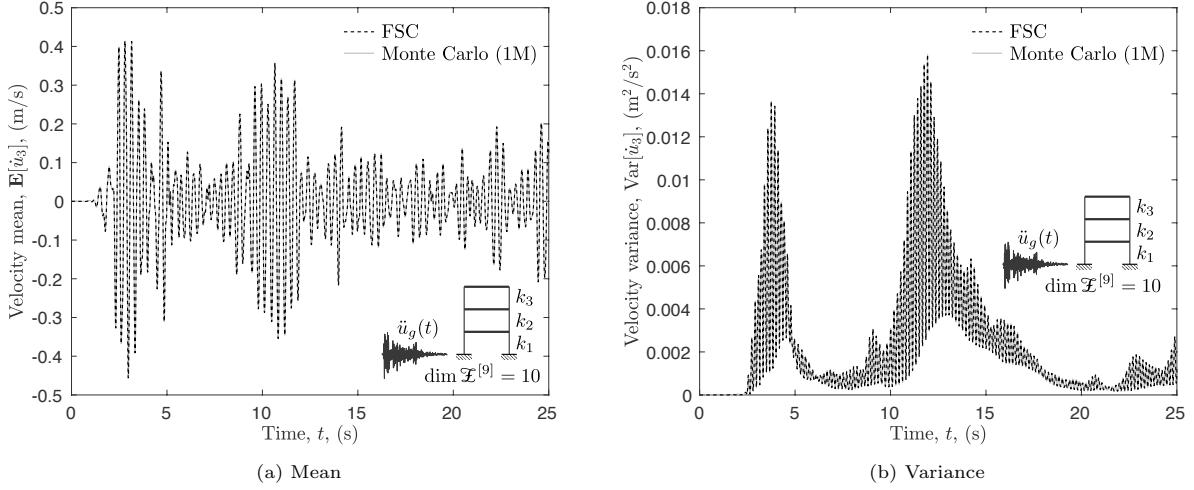


Figure 19: Evolution of $\mathbf{E}[\dot{u}_3]$ and $\text{Var}[\dot{u}_3]$ for the case when the p -discretization level of RFS is $\mathbb{Z}^{[9]}$ and $\mu \sim \text{Beta}^{\otimes 3} \otimes \text{Uniform}^{\otimes 2}$

computational cost. The illustrative problem described in Section 7 also shows that the FSC method can be readily applied to real-world structures involving multiple degrees of freedom. As a result, the FSC method has the potential to be used in the context of large-scale structural engineering problems to quantify the uncertainties of long-time response with high fidelity.

A. Random bases for illustrative examples

Table 2 presents the non-orthogonalized version of the random bases that we use in this manuscript to solve the examples described in Section 6.

B. Undamped single-degree-of-freedom system under free vibration

The objective of this section is to provide the exact response expressions of an undamped single-degree-of-freedom system subjected to free vibration for the case when the stiffness is assumed uniformly distributed with parameters k_a and k_b . The problem is stated formally as follows.

Problem statement. Consider a stochastic, undamped single-degree-of-freedom system with mass $m \in \mathbb{R}^+$, and stiffness $k : \Xi \rightarrow \mathbb{R}^+$ given by $k(\xi) = \xi \sim \text{Uniform}$ in $[k_a, k_b]$, subjected to free vibration. Note that: $k_b > k_a > 0$.

The *first problem* is to find the displacement of the system $u : \mathfrak{T} \times \Xi \rightarrow \mathbb{R}$ in \mathcal{U} , such that:

$$m\ddot{u} + ku = 0 \quad \text{on } \mathfrak{T} \times \Xi \quad (27a)$$

$$\{u(0, \cdot) = \mathbf{u}, \dot{u}(0, \cdot) = \mathbf{v}\} \quad \text{on } \{0\} \times \Xi \quad (27b)$$

where $\mathbf{u}, \mathbf{v} \in \mathbb{R}$, and $\dot{u} := \partial_t u$ and $\ddot{u} := \partial_t^2 u$ are the velocity and acceleration of the system, respectively.

The *second problem* is to find the expectation and variance of u , \dot{u} and \ddot{u} as a function of time.

Exact solution. The solution of (27) is well-known [39, 42], and it is given by

$$u(t, \xi) = \mathbf{u} \cos((\omega \circ k)(\xi) t) + \frac{\mathbf{v}}{(\omega \circ k)(\xi)} \sin((\omega \circ k)(\xi) t), \quad (28)$$

Table 2: Non-orthogonalized version of the random bases used in Section 6

Single-degree-of-freedom system under **free vibration** (Section 6.1)*:

$$\begin{aligned}
 \Phi_{0,i}(\xi) &:= 1 \\
 \Phi_{1,i}(\xi) &:= \hat{\varphi}^1(M)(0, \hat{s}(t_i, \xi)) = u_{.i}(t_i, \xi) = u_{.i-1}(t_i, \xi) \\
 \Phi_{2,i}(\xi) &:= \hat{\varphi}^2(M)(0, \hat{s}(t_i, \xi)) = \dot{u}_{.i}(t_i, \xi) = \dot{u}_{.i-1}(t_i, \xi) \\
 \Phi_{3,i}(\xi) &:= \hat{\varphi}^3(M)(0, \hat{s}(t_i, \xi)) = \partial_t^2 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} u_{.i}(t_i, \xi) \\
 \Phi_{4,i}(\xi) &:= \hat{\varphi}^4(M)(0, \hat{s}(t_i, \xi)) = \partial_t^3 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} \dot{u}_{.i}(t_i, \xi) \\
 \Phi_{5,i}(\xi) &:= \hat{\varphi}^5(M)(0, \hat{s}(t_i, \xi)) = \partial_t^4 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} \partial_t^2 u_{.i}(t_i, \xi) \\
 \Phi_{6,i}(\xi) &:= \hat{\varphi}^6(M)(0, \hat{s}(t_i, \xi)) = \partial_t^5 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} \partial_t^3 u_{.i}(t_i, \xi) \\
 &\vdots \quad (\text{until } P = M + 2 \text{ if needed})
 \end{aligned}$$

Single-degree-of-freedom system under **forced vibration** (Section 6.2)*:

$$\begin{aligned}
 \Phi_{0,i}(\xi) &:= 1 \\
 \Phi_{1,i}(\xi) &:= \hat{\varphi}^1(M)(0, \hat{s}(t_i, \xi)) = u_{.i}(t_i, \xi) = u_{.i-1}(t_i, \xi) \\
 \Phi_{2,i}(\xi) &:= \hat{\varphi}^2(M)(0, \hat{s}(t_i, \xi)) = \dot{u}_{.i}(t_i, \xi) = \dot{u}_{.i-1}(t_i, \xi) \\
 \Phi_{3,i}(\xi) &:= \hat{\varphi}^3(M)(0, \hat{s}(t_i, \xi)) = \partial_t^2 u_{.i}(t_i, \xi) = \frac{1}{m} (q(\xi) \sin(t_i) - k(\xi) u_{.i}(t_i, \xi)) \\
 \Phi_{4,i}(\xi) &:= \hat{\varphi}^4(M)(0, \hat{s}(t_i, \xi)) = \partial_t^3 u_{.i}(t_i, \xi) = \frac{1}{m} (q(\xi) \cos(t_i) - k(\xi) \dot{u}_{.i}(t_i, \xi)) \\
 \Phi_{5,i}(\xi) &:= \hat{\varphi}^5(M)(0, \hat{s}(t_i, \xi)) = \partial_t^4 u_{.i}(t_i, \xi) = \frac{1}{m} (-q(\xi) \sin(t_i) - k(\xi) \partial_t^2 u_{.i}(t_i, \xi)) \\
 \Phi_{6,i}(\xi) &:= \hat{\varphi}^6(M)(0, \hat{s}(t_i, \xi)) = \partial_t^5 u_{.i}(t_i, \xi) = \frac{1}{m} (-q(\xi) \cos(t_i) - k(\xi) \partial_t^3 u_{.i}(t_i, \xi)) \\
 &\vdots \quad (\text{until } P = M + 2 \text{ if needed})
 \end{aligned}$$

Nonlinear single-degree-of-freedom system under **free vibration** (Section 6.3)*:

$$\begin{aligned}
 \Phi_{0,i}(\xi) &:= 1 \\
 \Phi_{1,i}(\xi) &:= \hat{\varphi}^1(M)(0, \hat{s}(t_i, \xi)) = u_{.i}(t_i, \xi) = u_{.i-1}(t_i, \xi) \\
 \Phi_{2,i}(\xi) &:= \hat{\varphi}^2(M)(0, \hat{s}(t_i, \xi)) = \dot{u}_{.i}(t_i, \xi) = \dot{u}_{.i-1}(t_i, \xi) \\
 \Phi_{3,i}(\xi) &:= \hat{\varphi}^3(M)(0, \hat{s}(t_i, \xi)) = \partial_t^2 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} (1 + \rho(\xi) u_{.i}^2(t_i, \xi)) u_{.i}(t_i, \xi) \\
 \Phi_{4,i}(\xi) &:= \hat{\varphi}^4(M)(0, \hat{s}(t_i, \xi)) = \partial_t^3 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} (1 + 3 \rho(\xi) u_{.i}^2(t_i, \xi)) \dot{u}_{.i}(t_i, \xi) \\
 \Phi_{5,i}(\xi) &:= \hat{\varphi}^5(M)(0, \hat{s}(t_i, \xi)) = \partial_t^4 u_{.i}(t_i, \xi) = -\frac{k(\xi)}{m} (1 + 3 \rho(\xi) u_{.i}^2(t_i, \xi)) \partial_t^2 u_{.i}(t_i, \xi) - \frac{6 \rho(\xi) k(\xi)}{m} \dot{u}_{.i}^2(t_i, \xi) u_{.i}(t_i, \xi) \\
 &\vdots \quad (\text{until } P = M + 2 \text{ if needed})
 \end{aligned}$$

*The random basis is defined over the region $\mathfrak{R}_i = \text{cl}(\mathfrak{T}_i) \times \Xi$.

Table 3: Definition of τ -functions for a single-degree-of-freedom system subjected to free vibration with $k \sim \text{Uniform in } [k_a, k_b]$

$\tau_u(t, k) = \{\omega(k) t \sin(\omega(k) t) + \cos(\omega(k) t)\} u - \cos(\omega(k) t) v t$
$\tau_v(t, k) = -\{2 \omega(k) t \sin(\omega(k) t) + (2 - \omega^2(k) t^2) \cos(\omega(k) t)\} u t^{-1} + \{\omega(k) t \sin(\omega(k) t) + \cos(\omega(k) t)\} v$
$\tau_a(t, k) = -\{(\omega^3(k) t^3 - 6 \omega(k) t) \sin(\omega(k) t) + 3(\omega^2(k) t^2 - 2) \cos(\omega(k) t)\} u t^{-2}$ $\quad \quad \quad - \{2 \omega(k) t \sin(\omega(k) t) + (2 - \omega^2(k) t^2) \cos(\omega(k) t)\} v t^{-1}$

where the natural circular frequency of the system, $\omega : \mathbb{R}^+ \rightarrow \mathbb{R}^+$, is defined by

$$\omega(k) = \sqrt{\frac{k}{m}}.$$

Then, the following exact expressions can be derived for the expectation and variance of u , \dot{u} and \ddot{u} .

Exact expectation. The expectation of u , \dot{u} and \ddot{u} are given by:

$$\mathbf{E}[u](t) = \kappa(t) (\tau_u(t, k_b) - \tau_u(t, k_a)) \quad (29a)$$

$$\mathbf{E}[\dot{u}](t) = \kappa(t) (\tau_v(t, k_b) - \tau_v(t, k_a)) \quad (29b)$$

$$\mathbf{E}[\ddot{u}](t) = \kappa(t) (\tau_a(t, k_b) - \tau_a(t, k_a)), \quad (29c)$$

where $\kappa : \mathfrak{T} \rightarrow \mathbb{R}^+$ is given by $\kappa(t) = 2m/((k_b - k_a)t^2)$, and $\tau_u, \tau_v, \tau_a : \mathfrak{T} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ are defined in Table 3.

A closer look at the above expressions indicates that in the long term the absolute mean of the response is dominated by $\kappa = \kappa(t)$ which is a function that tends to zero as time goes to infinity. For this reason,

$$\mathbf{E}[u], \mathbf{E}[\dot{u}], \mathbf{E}[\ddot{u}] \rightarrow 0 \quad \text{as } t \rightarrow \infty. \quad (30)$$

Exact variance. The variance of u , \dot{u} and \ddot{u} are given by:

$$\text{Var}[u](t) = \kappa(t) (\varrho_u(t, k_b) - \varrho_u(t, k_a)) - \mathbf{E}[u]^2(t) \quad (31a)$$

$$\text{Var}[\dot{u}](t) = \kappa(t) (\varrho_v(t, k_b) - \varrho_v(t, k_a)) - \mathbf{E}[\dot{u}]^2(t) \quad (31b)$$

$$\text{Var}[\ddot{u}](t) = \kappa(t) (\varrho_a(t, k_b) - \varrho_a(t, k_a)) - \mathbf{E}[\ddot{u}]^2(t), \quad (31c)$$

where $\varrho_u, \varrho_v, \varrho_a : \mathfrak{T} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ are defined in Table 4. In the expression for ϱ_u , the function $\text{Ci} : \mathbb{R}^+ \rightarrow \mathbb{R}$ is the cosine integral given by

$$\text{Ci}(x) = - \int_x^\infty \frac{\cos y}{y} dy.$$

We see that not only the variance of the response is bounded for all $t \in \mathfrak{T}$, but also as t goes to infinity:

$$\lim_{t \rightarrow \infty} \text{Var}[u](t) = \frac{1}{2} u^2 + \frac{1}{2} \ln\left(\frac{k_b}{k_a}\right) \left(\frac{m}{k_b - k_a}\right) v^2 \quad (32a)$$

$$\lim_{t \rightarrow \infty} \text{Var}[\dot{u}](t) = \frac{1}{4} \left(\frac{k_a + k_b}{m}\right) u^2 + \frac{1}{2} v^2 \quad (32b)$$

$$\lim_{t \rightarrow \infty} \text{Var}[\ddot{u}](t) = \frac{1}{6} \left(\frac{k_a^2 + k_a k_b + k_b^2}{m^2}\right) u^2 + \frac{1}{4} \left(\frac{k_a + k_b}{m}\right) v^2. \quad (32c)$$

Therefore, the underlying process u is naturally of second-order.

Table 4: Definition of ϱ -functions for a single-degree-of-freedom system subjected to free vibration with $k \sim \text{Uniform in } [k_a, k_b]$

$\varrho_u(t, k) = -\frac{1}{4} \{ \sin^2(\omega(k)t) - \omega(k)t \sin(2\omega(k)t) - \omega^2(k)t^2 \} u^2 - \frac{1}{2} \cos(2\omega(k)t) uv t$ $\quad \quad \quad + \frac{1}{4} \{ \ln(k) - 2 \text{Ci}(2\omega(k)t) \} v^2 t^2$
$\varrho_v(t, k) = \frac{1}{16} \{ 2(3\omega(k)t - 2\omega^3(k)t^3) \sin(2\omega(k)t) + 3(1 - 2\omega^2(k)t^2) \cos(2\omega(k)t) + 2\omega^4(k)t^4 \} u^2 t^{-2}$ $\quad \quad \quad - \frac{1}{4} \{ 2\omega(k)t \sin(2\omega(k)t) + (1 - 2\omega^2(k)t^2) \cos(2\omega(k)t) \} uv t^{-1}$ $\quad \quad \quad - \frac{1}{4} \{ \sin^2(\omega(k)t) - \omega(k)t \sin(2\omega(k)t) - \omega^2(k)t^2 \} v^2$
$\varrho_a(t, k) = \frac{1}{48} \{ 6(2\omega^5(k)t^5 - 10\omega^3(k)t^3 + 15\omega(k)t \sin(2\omega(k)t)$ $\quad \quad \quad + 15(2\omega^4(k)t^4 - 6\omega^2(k)t^2 + 3) \cos(2\omega(k)t) + 4\omega^6(k)t^6 \} u^2 t^{-4}$ $\quad \quad \quad + \frac{1}{8} \{ 4(2\omega^3(k)t^3 - 3\omega(k)t \sin(2\omega(k)t) - 2(2\omega^4(k)t^4 - 6\omega^2(k)t^2 + 3) \cos(2\omega(k)t) \} uv t^{-3}$ $\quad \quad \quad - \frac{1}{16} \{ 2(2\omega^3(k)t^3 - 3\omega(k)t \sin(2\omega(k)t) + 3(2\omega^2(k)t^2 - 1) \cos(2\omega(k)t) - 2\omega^4(k)t^4 \} v^2 t^{-2}$

C. Overview of standard Monte Carlo method

The Monte Carlo method is the most popular numerical technique used in stochastic modeling to quantify the effects of input uncertainty on system's outputs. It is basically a 'brute-force' method of attack that typically involves sampling a large number of realizations from the random space to estimate the statistics of the output. It is well-known that when N realizations are considered, the mean converges asymptotically as the square root of N^{-1} , and thus, it is remarkably independent of the dimensionality of the random space [44]. In this paper we use *standard Monte Carlo* to validate the FSC method in Sections 6.3 and 7.

The general procedure for conducting a standard Monte Carlo simulation is simple. Consider the stochastic system given by (1*):

$$y(t, \xi) = \mathcal{M}[u][x](t, \xi) \quad \text{subject to initial condition} \quad \mathcal{I}[u](\xi). \quad (1^{**})$$

Then:

1. Generate N realizations of the d -tuple random variable ξ in order to obtain the random set $\{\xi_i\}_{i=1}^N$. These N realizations are based on randomly sampling N points from the random domain Ξ according to the cumulative distribution function $F : \Xi \rightarrow [0, 1]$ given by

$$F(\xi) = \prod_{j=1}^d \mu^j((-\infty, \xi_j]), \quad \text{or equivalently,} \quad F(\xi) = \int_{-\infty}^{\xi^1} \cdots \int_{-\infty}^{\xi^d} d\mu^1 \cdots d\mu^d. \quad (33)$$

This way we can also obtain the input set $\{x(t, \xi_i)\}_{i=1}^N$ for reference purposes.

2. Solve (1^{**}) for each random point ξ_i to obtain the output set $\{y(t, \xi_i)\}_{i=1}^N$.
3. Aggregate results to estimate the statistics of output y as a function of time. For instance, if we let z denote the k -th component of $y = (y_1, \dots, y_s)$, then its statistical mean $\mathbf{E}^*[z] : \mathfrak{T} \rightarrow \mathbb{R}$ and statistical variance $\text{Var}^*[z] : \mathfrak{T} \rightarrow \mathbb{R}_0^+$ are given by:

$$\mathbf{E}^*[z](t) = \frac{1}{N} \sum_{i=1}^N z(t, \xi_i) \quad \text{and} \quad \text{Var}^*[z](t) = \frac{1}{N-1} \sum_{i=1}^N (z(t, \xi_i) - \mathbf{E}^*[z](t))^2. \quad (34)$$

Acknowledgements

The first author gratefully acknowledges the scholarship granted by Colciencias and Atlántico Department (Colombia) under Call 673 to pursue a Ph.D. degree in Structural Engineering at Purdue University, West

Lafayette, IN. The authors also gratefully acknowledge the support of the National Science Foundation (CNS-1136075, DMS-1555072, DMS-1736364, CMMI-1634832, and CMMI-1560834), Brookhaven National Laboratory subcontract 382247, ARO/MURI grant W911NF-15-1-0562, and U.S. Department of Energy (DOE) Office of Science Advanced Scientific Computing Research program DE-SC0021142.

Credit author statement

H.E. conceived the mathematical models, implemented the methods, designed the numerical experiments, interpreted the results, and wrote the paper. **A.P.** and **G.L.** supported the study, edited, and reviewed the final manuscript. All the authors gave their final approval for publication.

References

- [1] N. Wiener, The homogeneous chaos, *American Journal of Mathematics* 60 (4) (1938) 897–936.
- [2] R. Ghanem, P. D. Spanos, Polynomial chaos in stochastic finite elements, *Journal of Applied Mechanics* 57 (1) (1990) 197–202.
- [3] D. Xiu, G. E. Karniadakis, The Wiener-Askey polynomial chaos for stochastic differential equations, *SIAM journal on scientific computing* 24 (2) (2002) 619–644.
- [4] X. Wan, G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, *Journal of Computational Physics* 209 (2) (2005) 617–642.
- [5] X. Wan, G. E. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, *SIAM Journal on Scientific Computing* 28 (3) (2006) 901–928.
- [6] X. Wan, G. E. Karniadakis, Long-term behavior of polynomial chaos in stochastic flow simulations, *Computer methods in applied mechanics and engineering* 195 (41-43) (2006) 5582–5596.
- [7] J. Foo, X. Wan, G. E. Karniadakis, The multi-element probabilistic collocation method (ME-PCM): Error analysis and applications, *Journal of Computational Physics* 227 (22) (2008) 9572–9595.
- [8] N. Agarwal, N. R. Aluru, A domain adaptive stochastic collocation approach for analysis of MEMS under uncertainties, *Journal of Computational Physics* 228 (20) (2009) 7662–7688.
- [9] X. Ma, N. Zabaras, An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations, *Journal of Computational Physics* 228 (8) (2009) 3084–3113.
- [10] G. Kewlani, K. Iagnemma, A multi-element generalized polynomial chaos approach to analysis of mobile robot dynamics under uncertainty, in: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2009, pp. 1177–1182.
- [11] J. Foo, G. E. Karniadakis, Multi-element probabilistic collocation method in high dimensions, *Journal of Computational Physics* 229 (5) (2010) 1536–1557.
- [12] P. Prempraneeach, F. S. Hover, M. S. Triantafyllou, G. E. Karniadakis, Uncertainty quantification in simulations of power systems: Multi-element polynomial chaos methods, *Reliability Engineering & System Safety* 95 (6) (2010) 632–646.
- [13] J. D. Jakeman, A. Narayan, D. Xiu, Minimal multi-element stochastic collocation for uncertainty quantification of discontinuous functions, *Journal of Computational Physics* 242 (2013) 790–808.
- [14] T. P. Sapsis, P. F. Lermusiaux, Dynamically orthogonal field equations for continuous stochastic dynamical systems, *Physica D: Nonlinear Phenomena* 238 (23) (2009) 2347–2360.
- [15] M. Choi, T. P. Sapsis, G. E. Karniadakis, A convergence study for SPDEs using combined polynomial chaos and dynamically-orthogonal schemes, *Journal of Computational Physics* 245 (2013) 281–301.
- [16] M. Ueckermann, P. F. Lermusiaux, T. P. Sapsis, Numerical schemes for dynamically orthogonal equations of stochastic fluid and ocean flows, *Journal of Computational Physics* 233 (2013) 272–294.
- [17] M. Cheng, T. Y. Hou, Z. Zhang, A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations. I: Derivation and algorithms, *Journal of Computational Physics* 242 (2013) 843–868.
- [18] M. Cheng, T. Y. Hou, Z. Zhang, A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations. II: Adaptivity and generalizations, *Journal of Computational Physics* 242 (2013) 753–776.
- [19] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, 1965.
- [20] P. Holmes, J. L. Lumley, G. Berkooz, C. W. Rowley, *Turbulence, coherent structures, dynamical systems and symmetry*, Cambridge university press, 2012.
- [21] E. Mosharbash, F. Nobile, T. Zhou, Error analysis of the dynamically orthogonal approximation of time dependent random PDEs, *SIAM Journal on Scientific Computing* 37 (2) (2015) A776–A810.
- [22] M. Gerritsma, J.-B. Van der Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos, *Journal of Computational Physics* 229 (22) (2010) 8333–8363.
- [23] V. Heuveline, M. Schick, A hybrid generalized polynomial chaos method for stochastic dynamical systems, *International Journal for Uncertainty Quantification* 4 (1) (2014).
- [24] D. M. Luchtenburg, S. L. Brunton, C. W. Rowley, Long-time uncertainty propagation using generalized polynomial chaos and flow map composition, *Journal of Computational Physics* 274 (2014) 783–802.
- [25] H. C. Ozen, G. Bal, Dynamical polynomial chaos expansions and long time evolution of differential equations with random forcing, *SIAM/ASA Journal on Uncertainty Quantification* 4 (1) (2016) 609–635.

- [26] H. Esquivel, A. Prakash, G. Lin, Flow-driven spectral chaos (FSC) method for simulating long-time dynamics of arbitrary-order non-linear stochastic dynamical systems, *Journal of Computational Physics* (2020) 110044.
- [27] D. Xiu, *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press, 2010.
- [28] W. Rudin, *Real and complex analysis*, McGraw-Hill, 1987.
- [29] O. P. Le Maître, O. M. Knio, *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*, Springer Science & Business Media, 2010.
- [30] T. J. Sullivan, *Introduction to Uncertainty Quantification*, Vol. 63, Springer, 2015.
- [31] O. G. Ernst, A. Mugler, H.-J. Starkloff, E. Ullmann, On the convergence of generalized polynomial chaos expansions, *ESAIM: Mathematical Modelling and Numerical Analysis* 46 (2) (2012) 317–339.
- [32] P. J. Davis, P. Rabinowitz, *Methods of numerical integration*, Courier Corporation, 2007.
- [33] E. Novak, K. Ritter, High dimensional integration of smooth functions over cubes, *Numerische Mathematik* 75 (1) (1996) 79–97.
- [34] H.-J. Bungartz, M. Griebel, Sparse grids, *Acta numerica* 13 (2004) 147–269.
- [35] G. Leobacher, F. Pillichshammer, *Introduction to quasi-Monte Carlo integration and applications*, Springer, 2014.
- [36] L. Guo, A. Narayan, T. Zhou, Constructing least-squares polynomial approximations, *SIAM Review* 62 (2) (2020) 483–508.
- [37] J. C. Butcher, J. Butcher, *The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods*, Vol. 512, Wiley New York, 1987.
- [38] N. M. Newmark, A method of computation for structural dynamics, *Journal of the engineering mechanics division* 85 (3) (1959) 67–94.
- [39] A. K. Chopra, *Dynamics of Structures: Theory and Applications to Earthquake Engineering*, 4th Edition, Prentice Hall New Jersey, 2012.
- [40] W. Cheney, D. Kincaid, *Linear algebra: Theory and applications*, 2nd Edition, Jones & Bartlett Learning, 2010.
- [41] The MathWorks Inc., *MATLAB* (2016).
URL www.mathworks.com
- [42] J. L. Humar, *Dynamics of Structures*, 2nd Edition, CRC Press, 2012.
- [43] T. D. Ancheta, R. B. Darragh, J. P. Stewart, E. Seyhan, W. J. Silva, B. S.-J. Chiou, K. E. Wooddell, R. W. Graves, A. R. Kottke, D. M. Boore, et al., NGA-West2 database, *Earthquake Spectra* 30 (3) (2014) 989–1005.
- [44] G. Fishman, *Monte Carlo: Concepts, algorithms, and applications*, Springer Science & Business Media, 2013.