Uncertainty Quantification for Stochastic Dynamical Systems

Spectral Methods using Generalized Polynomial Chaos

Zur Erlangung des akademischen Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

von der Fakultät für Mathematik des
Karlsruher Instituts für Technologie (KIT)
genehmigte

DISSERTATION

von
Dipl.-Math. techn. Michael Schick
aus
Bruchsal

Tag der mündlichen Prüfung: 21. Dezember 2011

Referent:	Prof.	Dr. Vincent Heuveline
Korreferent:	Prof.	Dr. Olivier Le Maître
Korreferentin:	PD	Dr. Gudrun Thäter

Abstract

Generalized Polynomial Chaos (gPC) presents an efficient spectral method for stochastic processes or fields. However, it is known that gPC exhibits a convergence breakdown in cases involving strong nonlinear dependencies on random parameters. The first part of this work describes the application of gPC in context of ordinary differential equations subject to random quantities and introduces new hybrid approaches involving time-dependent basis functionals in a decomposed probability space, which allow for a significant increase of the numerical accuracy and improvement of the convergence properties of gPC. The second part of this work is devoted to the computation of stochastic time-periodic solutions to the unsteady stochastic incompressible Navier-Stokes equations. Thereby, a new numerical algorithm has been developed based on Newton's method and optimization techniques, which transfers the uncertainty of the simulation time interval into the governing equations.

Zusammenfassung

Generalized Polynomial Chaos (gPC) stellt eine effiziente Spektralmethode für stochastische Prozesse dar. Es ist bekannt, dass gPC in Fällen starker nichtlinearer Abhängigkeiten bzgl. stochastischer Parameter ein schlechtes Konvergenzverhalten aufzeigt. Der erste Teil dieser Arbeit betrachtet gPC für gewöhnliche Differentialgleichungen unter Einfluss stochastischer Größen. Hierbei wird ein neues hybrides numerisches Verfahren vorgestellt, welches auf zeitabhängigen Basisfunktionalen auf einer Gebietszerlegung des Wahrscheinlichkeitsraumes basiert. Dieses ermöglicht es sehr hohe Genauigkeiten in der numerischen Lösung zu erzielen und die Konvergenz von gPC deutlich zu verbessern. Der zweite Teil dieser Arbeit betrachtet gPC zur Berechnung zeit-periodischer Strömungsvorgänge, welche mittels den instationären stochastischen inkompressiblen Navier-Stokes Gleichungen modelliert werden. Hierbei wird eine neuer numerischer Algorithmus basierend auf dem Newton- und Optimierungsverfahren entwickelt, welcher die stochastische Periodenlänge im zeitlichen Intervall der Simulation auf die zugrunde liegenden Differentialgleichungen überträgt.

Acknowledgments

I want to thank Prof. Dr. Vincent Heuveline for providing the framework and support for the development of this work. Especially, I want to thank him for many fruitful discussions throughout the past years. Furthermore, the continuous exchange with my friends and colleagues within the working group "Numerical Simulation, Optimization and High Performance Computing" has been a driving force at all times for which I am very thankful. I also want to thank Prof. Dr. Olivier Le Maître, who provided me with lots of feedback regarding this work and with the initial ideas for developing the methodology described in Chapter 5.

Contents

Chapter 1.

Introduction

Many physical phenomena can be modeled by systems of differential equations describing the interaction of system variables subject to predefined parameters. In practice, their solution often requires adequate numerical solvers, since for many applications no analytical solution can be derived. Thereby, the accuracy of a numerically computed solution is determined by three major classes of error contributions:

- **Model error:**
  In practice, one often relies on approximations of physical relations, e.g., by modeling not exactly known system characteristics or by simplifications of complex interactions within an exact model. In fluid flow problems, for example, the incompressible Navier-Stokes equations represent a reasonably simplified model of their compressible version within an appropriate framework, for which a constant fluid density can be assumed. However, such approximations need to be validated carefully a posteriori to ensure a small model error contribution, otherwise it cannot be guaranteed that the solution of the underlying model is actually a solution to the exact physical system.

- **Numerical error:**
  The numerical solution of the model equations is based on a finite dimensional discretization of the continuous quantities, e.g., by a polynomial approximation of smooth functions. This introduces an additional source of errors, whose influence on the solution process has to be controlled in an adequate way. This can be achieved, for example, by a priori and a posteriori error estimation techniques. Furthermore, the solution of a corresponding discrete system in high dimensions often employs iterative numerical solvers, which in practice can only evaluate a finite number of iterations. Since on standard computer architectures only a finite precision of the computed values is available, additional round-off errors are introduced, which especially for ill-conditioned problems can provide a significant impact on the solvability of the model equations.

- **Data error:**
  The third class of errors involves the data needed to determine system parameters a priori to accurately describe the underlying model. In practice, information about the data is often subject to noisy measurements or even not available at all. A deterministic approach is to neglect the noise within the data and approximate it by some deterministic reference value, which, for example, can be determined by filter techniques. However, if such an approach

is used, no sensitivity information with respect to the data error within the solution of the underlying model is available. Therefore, there exist many different techniques which try to capture the data error propagation to obtain reliable validity of the computed results. One example is to model the lack of information by an uncertainty model, which represents a stochastic approach to estimate the sensitivity subject to the probability distribution of the modeled data.

The main focus of this work lies on the quantification of error propagations resulting from uncertainty models of system parameters (data errors), which in literature is referred to as "Uncertainty Quantification" (UQ). Its aim is to provide feasible information regarding the sensitivity of a solution with respect to stochastic variations in system parameters, thereby taking into account the evolution of the probability distribution of the random input. An abstract view on UQ can be stated by the following:

Suppose one seeks a solution u of some model $\mathcal{M}$ subject to some data d satisfying:

$$\mathcal{M}(u; d) = 0. \tag{1.1}$$

Furthermore, suppose that the data d is random and parameterized by some vector-valued random variable denoted by ξ, i.e., $d = d(\xi)$. Hence, equation (1.1) implicitly introduces a functional dependency of a solution u on the random input ξ, i.e., $u = u(\xi)$. Therefore, the value of u depends on the realization of the random variable ξ. Usually one is interested in computing specific outcomes of u or determining its stochastic moments, e.g., its mean or variance. This can be accomplished by various numerical methods, for which two major representatives are highlighted in the following, the second one being elaborated on within this work.

1.1. Monte Carlo methods

Monte Carlo methods present a popular approach to compute stochastic moments of a solution by deterministic numerical simulations based on random realizations of the input data. The standard technique is to generate a finite sequence of random samples $\xi_1, \xi_2, \xi_3, \ldots, \xi_M$, $M \in \mathbb{N}$, by so-called "pseudo" random number generators. Afterwards, to each sample ξ_i a deterministic solution $u_i := u(\xi_i)$ is computed via:

$$\mathcal{M}(u_i; d(\xi_i)) = 0, \tag{1.2}$$

for $i = 1, \ldots, M$. The stochastic moments of u can be estimated based on the computed solutions $\{u_i\}_{i=1}^{M}$, e.g., for the expectation $\mathbb{E}(u)$ of u it holds:

$$\mathbb{E}(u) = \lim_{M \to \infty} \frac{1}{M} \sum_{i=1}^{M} u_i p_i, \tag{1.3}$$

whereas $p_i := \mathbb{P}(\xi = \xi_i)$ denotes the probability of the event $\{\xi = \xi_i\}$, $i = 1, \ldots, M$. This reflects the quite simple and powerful way of determining information about a stochastic quantity u without imposing specific regularity requirements, which

makes it also very robust. Since only independent resolutions of deterministic models are required, the deterministic systems can be computed in parallel in a trivial way. Furthermore, the fact that no changes in a deterministic model are required to determine a stochastic solution results in a very attractive feature of the method itself. However, the convergence rate of Monte Carlo methods is relative low and behaves asymptotically like $O(1/\sqrt{M})$ as $M \to \infty$ [25]. This can provide a huge drawback if the resolution of a single deterministic system is numerically expensive and a high accuracy of the resolved model is required. There exist techniques, e.g., "Quasi Monte Carlo", "Latin-Hypercube" or "variance reduction" methods [28, 29, 30], which improve the convergence rate, however, obtaining high accuracies still requires a large number of realizations. An alternative approach, which will be the main focus of this work, is given by spectral methods which may exhibit significantly higher convergence rates.

1.2. Spectral methods

Spectral methods provide a substantially different approach towards the resolution of the model equation (1.1). Their aim is to construct a functional dependency of the solution u on the random input ξ, such that:

$$u(\xi) = \sum_{i=0}^{\infty} u_i \psi_i(\xi), \tag{1.4}$$

whereas $\{\psi_i\}_{i=0}^{\infty}$ denotes a set of selected basis functionals and $\{u_i\}_{i=0}^{\infty}$ denotes deterministic "modes" of u. Note that usually, such a series representation will result in additional regularity requirements on the solution u, e.g., u being square-integrable with respect to the underlying probability space. If possible, the derivation of the modes can be carried out analytically, otherwise by utilizing sampling or projection techniques [25]. The two latter approaches mainly differ in the fact, that projection techniques require a change of the model $\mathcal{M}$ by a projection onto the space spanned by the basis functionals $\{\psi_i\}_{i=0}^{\infty}$. This is not the case for sampling based strategies, which only require multiple deterministic resolutions of the model $\mathcal{M}$.

The main focus of this work lies on a specific representative of such spectral methods, given by the so-called "generalized Polynomial Chaos" (gPC) expansion [47] based on the original "Homogeneous Chaos" introduced in [46]. It became increasingly popular within the last two decades and is based on choosing the basis $\{\psi_i\}_{i=0}^{\infty}$ to consist of orthogonal polynomial functionals exhibiting exponential convergence rates for square-integrable random variables. The gPC expansion has been successfully applied in various field, such as solid and fluid mechanics [15, 24, 27] and provides a powerful method to obtain high accuracies for low to moderate sized model equations. However, it is also known that the standard gPC approach exhibits a convergence breakdown in cases involving strong nonlinear dependencies on the random input, e.g., in context of stochastic discontinuities or long-term integration, especially arising in the field of stochastic dynamical systems [14, 19, 20, 26, 43]. In this work, these problem classes are being elaborated on by considering representatives of stochastic ordinary differential equations (SODEs) and the unsteady

stochastic incompressible Navier-Stokes equations (SNSE). For the SODEs, new modifications towards gPC are developed, which significantly increase the accuracy of gPC and therefore making it possible to apply gPC with respect to strong non-linear dependencies. Furthermore, a new efficient numerical method is developed for computing time-periodic stochastic solutions to the SNSE, therefore extending the field of application of gPC towards this complex problem class and overcoming the previously mentioned convergence breakdown of gPC.

1.3. Outline

This work is structured in the following way: Chapter 2 provides a detailed review of the generalized Polynomial Chaos expansion and corresponding Galerkin projected systems for differential equations subject to a random input. Thereby, especially the sparsity structure resulting from the projection of the model equations is being analyzed. Chaper 3 is focused on the application of gPC to stochastic ordinary differential equations, whereas new modifications of the standard gPC approach are introduced, which are able to capture the evolution of the probability distribution in time more efficiently. These are based on a change of the probability measure and a decomposition of the probability space. Chapter 4 introduces the application of gPC to the SNSE using the so-called "Spectral-Stochastic-Finite-Element-Method". Thereby, the governing equations are put into an appropriate analytical framework for which numerically stable stochastic Finite-Element discretization schemes are derived. In Chapter 5 the results of Chapter 4 are applied to time-periodic fluid flow problems to which end a new numerical method is developed, which enables the computation of a period and a corresponding initial condition in a stochastic framework based on an iterative approach. Conclusions from this work are drawn in Chapter 6.

Chapter 2.

Generalized Polynomial Chaos

In 1938, Norbert Wiener [46] introduced the so-called "Homogeneous Chaos" as the span of Hermite polynomial functionals of a Gaussian process. "Polynomial Chaos" (PC) is defined as a member of this set. It is a Fourier-Hermite series expansion, for which Cameron and Martin proved in 1947 [4], that the orthogonal representation converges to any square-integrable functional. In context of stochastic processes, this implies that the Homogeneous Chaos converges to any stochastic process of second order, which was put in a broader framework involving more general probability distributions, referred to as "generalized Polynomial Chaos" (gPC) by Xiu and Karniadakis in 2002 [47]. Ghanem and Spanos pioneered in 1991 [15] the application of the Polynomial Chaos expansion to problems arising in the field of solid mechanics employing a Finite-Element discretization in the space variable. This chapter will introduce the basic concepts of generalized Polynomial Chaos and analyze the system structure resulting from a stochastic Galerkin projection.

2.1. Polynomial Chaos

Consider, without loss of generality, a real-valued random variable $X = X(\omega)$ according to some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Here, Ω denotes the sample space, $\mathcal{F} \subset 2^{\Omega}$ a filtration, and $\mathbb{P}$ a probability measure. Furthermore, it is assumed that X is square-integrable, i.e., $X \in L^2(\Omega) = \{X : \mathbb{E}(X^2) < \infty\}$, whereas

$$\mathbb{E}(X) := \int_{\Omega} X \, d\mathbb{P} \tag{2.1}$$

denotes the expectation of X. For a sequence of centered, normalized and mutually orthogonal Gaussian random variables $\{\xi_i\}_{i=1}^{\infty}$ define $\hat{\Gamma}_p$ to be the space of polynomials in $\{\xi_i\}_{i=1}^{\infty}$ having a polynomial degree of less than or equal to $p \in \mathbb{N}$. Furthermore, define $\Gamma_p \subset \hat{\Gamma}_p$ to be the set of polynomials, which belong to $\hat{\Gamma}_p$ and which are orthogonal to $\hat{\Gamma}_{p-1}$. The space spanned by Γ_p shall be denoted by $\tilde{\Gamma}_p$. Then, the Cameron and Martin theorem [4] yields:

$$\hat{\Gamma}_p = \hat{\Gamma}_{p-1} \oplus \tilde{\Gamma}_p, \quad L^2(\Omega) = \bigoplus_{i=0}^{\infty} \tilde{\Gamma}_i. \tag{2.2}$$

Here, the subspace $\tilde{\Gamma}_p$ of $L^2(\Omega)$ is called the p-th Homogeneous Chaos, whereas Γ_p is called the Polynomial Chaos of order p. Therefore, the original Polynomial Chaos

expansion of X can be represented in the form [25]:

$$X(\omega) = a_0 H_0 + \sum_{i_1=1}^{\infty} a_{i_1} H_1(\xi_{i_1}(\omega)) + \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} a_{i_1 i_2} H_2(\xi_{i_1}(\omega), \xi_{i_2}(\omega)) + \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} \sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3} H_3(\xi_{i_1}(\omega), \xi_{i_2}(\omega), \xi_{i_3}(\omega)) + \dots, \tag{2.3}$$

whereas $H_n(\xi_{i_1}, \dots, \xi_{i_n})$ denotes the multi-dimensional Hermite polynomial of order n in terms of the multi-dimensional independent standard Gaussian random variable $\xi = (\xi_{i_1}, \dots, \xi_{i_n})$. Note that this expansion corresponds to the discrete version of the decomposition originally introduced by Wiener [46], which employs integrals instead of summations. Furthermore, the Polynomial Chaos expansion converges in mean-square to X, i.e.,

$$\lim_{p \to \infty} \mathbb{E}\left[\left(a_0 H_0 + \dots + \sum_{i_1=1}^{\infty} \cdots \sum_{i_n=1}^{i_{n-1}} a_{i_1 \dots i_n} H_n(\xi_{i_1}, \dots, \xi_{i_n}) - X \right)^2 \right] = 0. \tag{2.4}$$

The multi-dimensional Hermite polynomials are constructed via a tensor product of the corresponding one-dimensional Hermite polynomials, resulting in:

$$H_n(\xi_{i_1}, \dots, \xi_{i_n}) = (-1)^n \exp(\frac{1}{2}\xi^t \xi) \frac{\partial^n}{\partial \xi_{i_1} \cdots \partial \xi_{i_n}} \exp(-\frac{1}{2}\xi^t \xi). \tag{2.5}$$

To simplify the notation, equation (2.3) can be recasted to obtain the form:

$$X(\omega) = \sum_{i=0}^{\infty} x_i \psi_i(\xi(\omega)), \tag{2.6}$$

since there exists a one-to-one correspondence between the polynomials $\{\psi_i\}$ with coefficients $\{x_i\}$ and $\{H_n\}$ with coefficients $\{a_{i_1}, \dots, a_{i_n}\}$. The basis functionals indeed form an orthogonal and complete basis in $L^2(\Omega)$, whereas the orthogonality is expressed by:

$$\langle \psi_i, \psi_j \rangle = \langle \psi_i, \psi_i \rangle \delta_{ij}. \tag{2.7}$$

Here, δ_{ij} denotes the Kronecker–Delta and $\langle \cdot, \cdot \rangle$ denotes the inner product, defined by:

$$\langle f(\xi), g(\xi) \rangle := \int_{\text{supp } w(\cdot)} f(\xi) g(\xi) w(\xi) \, d\xi, \tag{2.8}$$

whereas the weight function w is given by:

$$w(\xi) := \frac{1}{\sqrt{(2\pi)^n}} \exp(-\frac{1}{2}\xi^t \xi) \tag{2.9}$$

as the joint probability density function of the multi-dimensional independent standard Gaussian random variable ξ. Taking the inner product is the same as taking the expectation of the functionals, i.e.,

$$\langle f(\xi), g\xi) \rangle = \mathbb{E}(f(\xi) g(\xi)) = \int_{\Omega} f(\xi) g(\xi) \, d\mathbb{P}. \tag{2.10}$$

Remark 2.1. The Hermite polynomials $\{\psi_i\}_{i=0}^{\infty}$ are ordered starting from lower polynomial degree up to the higher degrees. Therefore, for a two-dimensional random vector $\xi = (\xi_1, \xi_2)$ this expansion reads:

$$\begin{aligned} X &= a_0 H_0 + a_1 H_1(\xi_1) + a_2 H_1(\xi_2) \\ &\quad + a_{11} H_2(\xi_1, \xi_1) + a_{12} H_2(\xi_1, \xi_2) + a_{22} H_2(\xi_2, \xi_2) + \dots \qquad (2.11) \\ &= x_0\psi_0 + x_1\psi_1 + x_2\psi_2 + x_3\psi_3 + x_4\psi_4 + x_5\psi_5 + \dots \qquad (2.12) \\ &= x_0 + x_1\xi_1 + x_2\xi_2 + x_3(\xi_1^2 - 1) + x_4\xi_1\xi_2 + x_5(\xi_2^2 - 1) + \dots . \qquad (2.13) \end{aligned}$$

2.2. Generalization

In practical applications, it is desirable that the series in (2.6) converges fast, since especially for computational purposes, the series needs to be truncated at some finite integer, which preferably should be of small value. Being a spectral polynomial expansion, an exponential convergence rate is optimal [47]. Although for the decomposition of a Gaussian random variable a first order expansion is exact, other types of non-Gaussian random variables may exhibit low convergence rates and hence require a large truncation order. Therefore, Xiu and Karniadakis [47] introduced the so-called "Askey-Chaos" and with it the "generalized Polynomial Chaos", which essentially replaces the Hermite polynomials $\{H_i\}$ by other polynomials, denoted by $\{I_i\}$, which are orthogonal with respect to the underlying probability density function of the random vector ξ.

To distinguish between the classical Hermite-Chaos in terms of a Gaussian random variable ξ the notation for the random vector is changed to ζ within this section to emphasize the possibility of other types of probability distributions. Therefore, the generalization of the Polynomial Chaos expansion reads:

$$\begin{aligned} X(\omega) = a_0 I_0 &+ \sum_{i_1=1}^{\infty} a_{i_1} I_1(\zeta_{i_1}(\omega)) + \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} a_{i_1 i_2} I_2(\zeta_{i_1}(\omega), \zeta_{i_2}(\omega)) \\ &+ \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} \sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3} I_3(\zeta_{i_1}(\omega), \zeta_{i_2}(\omega), \zeta_{i_3}(\omega)) + \dots . \qquad (2.14) \end{aligned}$$

For example, if X is chosen to be represented in terms of uniformly distributed random variables, then the $\{I_i\}$ are chosen to be Legendre polynomials. The "Askey-Chaos" is listed in Table 2.1 and gives an overview which polynomials correspond to certain types of distributions (continuous and discrete ones).

Remark 2.2. For discrete distributions, the integral in (2.8) is replaced by a sum, which yields:

$$\langle f(\zeta), g(\zeta) \rangle = \sum_{\zeta} f(\zeta) g(\zeta) w(\zeta). \qquad (2.15)$$

It is possible to extend gPC even further, by allowing arbitrary probability densities and hence construct corresponding orthogonal polynomials. Another possibility is to even allow for non-polynomial representations, e.g., by using trigonometric functionals. However, there exist probability distributions, such as the log-normal

	Probability distribution	Askey-Chaos	Support
Continuous	Gaussian	Hermite-Chaos	$(-\infty, \infty)$
	Gamma	Laguerre-Chaos	$[0, \infty)$
	Beta	Jacobi-Chaos	$[a, b]$
	Uniform	Legendre-Chaos	$[a, b]$
Discrete	Poisson	Charlier-Chaos	$\{0, 1, 2, \ldots\}$
	Binomial	Krawtchouk-Chaos	$\{0, 1, 2, \ldots, N\}$
	Negative binomial	Meixner-Chaos	$\{0, 1, 2, \ldots\}$
	Hypergeometric	Hahn-Chaos	$\{0, 1, 2, \ldots, N\}$

Table 2.1.: Askey-Chaos corresponding to certain types of probability distributions.

distribution, for which the series in (2.14) does not converge to any second order random variable, since a momentum condition does not hold, which was shown to be a necessary requirement for convergence [11], based on a generalization of the Cameron and Martin theorem [4]. Since a log-normal distributed random variable can be recast in terms of a Gaussian random variable, a Hermite-Chaos can provide an alternative to achieve convergence for this case.

2.3. Spectral Galerkin projection of differential equations

Suppose, a solution u is sought to some differential equation with stochastic parameters, given by:

$$\mathcal{L}(u, x, t; \omega) = f(x, t; \omega), \quad a.s. \text{ in } \Omega, \tag{2.16}$$

whereas $\mathcal{L}$ denotes some differential operator involving derivatives of u in the time variable $t \in [0, T] \subset \mathbb{R}$ and space variable $x \in \mathcal{D} \subset \mathbb{R}^d$ in an open and bounded domain $\mathcal{D}$ for some $d \in \mathbb{N}$. Here, for simplicity it is assumed that there exists a unique and scalar valued solution u to equation (2.16) with finite second order moments according to some prescribed boundary and initial conditions. Furthermore, $\omega \in \Omega$ denotes a sample from the sample space Ω and represents the stochastic nature of the differential equation. Suppose, that the stochastic influence can be parameterized and approximated by some finite dimensional random vector $\xi = (\xi_1, \ldots, \xi_L)$ of dimension $L \in \mathbb{N}$. One possibility and popular technique to derive such a finite approximation is given by the so-called "Karhunen-Loève decomposition" [25], based on a spectral expansion of the covariance function of the random vector ξ. Therefore, equation (2.16) can be recast in a parameterized form:

$$\mathcal{L}(u, x, t; \xi) = f(x, t; \xi). \tag{2.17}$$

Because of the stochastic dependency of the differential equation on ξ, the solution u itself is a random field in space and time, i.e., $u = u(x, t; \xi)$. Since u is assumed to be square-integrable for all $(x, t) \in \mathcal{D} \times [0, T]$, it has a convergent gPC expansion pointwise in $\mathcal{D} \times [0, T]$. Depending on the probability distribution of ξ the expansion

polynomials $\{\psi_i\}$ are chosen according to the Askey-Chaos (see Table 2.1) and u is approximated via a finite dimensional gPC expansion of order $No \in \mathbb{N}$:

$$u(x,t;\xi) \approx u^P(x,t;\xi) := \sum_{i=0}^{P} u_i(x,t)\psi_i(\xi), \tag{2.18}$$

whereas for the actual number of terms $P+1$ employed in (2.18) it holds:

$$P+1 = \frac{(No+L)!}{No!L!}. \tag{2.19}$$

Here, for simplicity the notation of ω is dropped and $\psi_i(\xi) := \psi_i(\xi(\omega))$ almost surely in Ω. Note, that in this context the gPC expansion has been extended to the case of stochastic processes and random fields by a pointwise gPC expansion of u in Ω leading to a separation of the functional dependency of u on the deterministic space and time variable $(x,t) \in \mathcal{D}\times[0,T]$ and the stochastic random vector ξ. For the rest of this work the deterministic coefficients $\{u_i\}_{i=0}^P$ arising in a gPC expansion will be referred to as "stochastic modes" or simply "modes". Furthermore, the notation u^P will be dropped for notational convenience such that u itself may denote its gPC expansion employing a finite number of terms, therefore also dropping the approximation symbol "$\approx$". It will be explicitly stated if u and u^P should need to be distinguished within some context.

Next, the discretization (2.18) is inserted into the governing equations (2.16) resulting in:

$$\mathcal{L}(\sum_{i=0}^{P} u_i\psi_i, x, t; \xi) = f(x,t;\xi). \tag{2.20}$$

At this point, a Galerkin projection is employed, ensuring that the residual of (2.16) is orthogonal to the space spanned by the Chaos Polynomials. This is achieved by multiplying equation (2.20) with ψ_j, $j = 0,\dots,P$ and taking the inner product $\langle\cdot,\cdot\rangle$, leading to:

$$\left\langle \mathcal{L}(\sum_{i=0}^{P} u_i\psi_i, x, t; \xi), \psi_j \right\rangle = \langle f(x,t;\xi), \psi_j\rangle\,, \quad \forall j = 0,\dots,P. \tag{2.21}$$

The resulting coupled system (2.21) is deterministic and of size $P+1$. Hence, taking the stochastic dependency into account leads to a blown up system size by a factor of $P+1$, with respect to a corresponding deterministic problem. In literature, this phenomenon is often referred to as "curse of dimensionality". It is important to note, that when no explicit stochastic influence is given by $\mathcal{L}$, i.e., $\mathcal{L} \neq \mathcal{L}(\cdot;\xi)$ and furthermore, $\mathcal{L}$ is linear in u, the system deteriorates to a fully decoupled structure, leading to $P+1$ independent computable problems. This can easily be verified by observing that the following relation holds for a linear operator $\mathcal{L}$ for all $j = 0,\dots,P$:

$$\left\langle \mathcal{L}(\sum_{i=0}^{P} u_i\psi_i, x, t; \xi), \psi_j \right\rangle = \sum_{i=0}^{P} \langle\psi_i,\psi_j\rangle \mathcal{L}(u_i,x,t), \tag{2.22}$$

$$= \langle\psi_j,\psi_j\rangle \mathcal{L}(u_j,x,t), \tag{2.23}$$

$No\backslash L$	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	6	10	15	21	28
3	4	10	20	35	56	84
4	5	15	35	70	126	210
5	6	21	56	126	252	462
6	7	28	84	210	462	924

Table 2.2.: Number of unknown modes $P+1$ in gPC. No denotes the maximum polynomial degree of the Chaos Polynomials and L the dimension of the random vector ξ.

by the orthogonality of the Chaos Polynomials $\{\psi_i\}_{i=0}^{P}$.

At this point any appropriate deterministic solver can be employed when solving the system of differential equations (2.21) for the modes u_i, $i = 0, \ldots, P$. However, as can be seen from Table 2.2 the system size grows extremely fast due to the factorial dependency of the size $P+1$ on the parameters No and L (cf. Eq. (2.19)). This illustrates a huge numerical challenge, especially when the discretization of the deterministic part is numerically expensive itself.

The stochastic moments of u, e.g., the mean and the variance, can be calculated by a weighted sum of the computed modes u_i. For the mean it holds:

$$\begin{aligned}\mathbb{E}(u)(x,t) &= \left\langle \sum_{i=0}^{P} u_i(x,t)\psi_i, 1 \right\rangle \\ &= \sum_{i=0}^{P} u_i(x,t)\langle \psi_i, 1\rangle = u_0(x,t),\end{aligned} \tag{2.24}$$

since $\langle \psi_i, 1\rangle = 0$ for $i > 0$ due to the orthogonality of the polynomials to ψ_0, for which, without loss of generality, can be assumed that $\psi_0 \equiv 1$. For the variance σ^2 it holds:

$$\begin{aligned}\sigma^2(u)(x,t) &= \mathbb{E}(u^2)(x,t) - (\mathbb{E}(u)(x,t))^2 \\ &= \left\langle \sum_{i=0}^{P}\sum_{j=0}^{P} u_i(x,t)u_j(x,t)\psi_i, \psi_j \right\rangle - u_0^2(x,t) \\ &= \sum_{i=0}^{P}\sum_{j=0}^{P} u_i(x,t)u_j(x,t)\langle \psi_i, \psi_j\rangle - u_0^2(x,t) \\ &= \sum_{i=0}^{P} u_i^2(x,t)\langle \psi_i, \psi_i\rangle - u_0^2(x,t) \\ &= \sum_{i=1}^{P} u_i^2(x,t)\langle \psi_i, \psi_i\rangle.\end{aligned} \tag{2.25}$$

2.4. Sparsity structure of a Galerkin projected system

This section aims at analyzing the sparsity structure of systems arising in the context of stochastic Galerkin projections. Especially, bilinear and trilinear forms with a dependency on some random vector ξ are addressed, since these can be used as an abstract representation of stochastic weak formulations typically arising in context of Finite-Element discretizations of stochastic partial differential equations.

2.4.1. Linear projected systems

Consider the abstract stochastic problem:

Find $u \in L^2(\Omega) \otimes V$ such that

$$a(u, v; \xi) = l(v), \quad \forall v \in L^2(\Omega) \otimes W, \tag{2.26}$$

for some given bilinear form a, linear form l and vector spaces V and W. Since this section is only concerned with the structure of the corresponding Galerkin projected system, it is assumed that there exists some unique solution $u \in L^2(\Omega) \otimes V$, which solves (2.26). This problem class includes, for example, a weak formulation of the stochastic Poisson problem, given by:

Find $u \in L^2(\Omega) \otimes H_0^1(\mathcal{D})$ such that

$$\nu(\xi)(\nabla u, \nabla v) = (f, v), \quad \forall v \in L^2(\Omega) \otimes H_0^1(\mathcal{D}), \tag{2.27}$$

for some open bounded Lipschitz domain $\mathcal{D} \subset \mathbb{R}^2$, a given right hand side f and some stochastic diffusion coefficient $\nu \in L^2(\Omega)$, whereas $(\cdot, \cdot)$ denotes the inner product on $L^2(\mathcal{D})$.

Since all quantities are assumed to be square-integrable in Ω, they posses a convergent gPC expansion, which approximation employing $P + 1$ terms is given by:

$$u = \sum_{i=0}^{P} u_i \psi_i, \tag{2.28}$$

$$v = \sum_{i=0}^{P} v_i \psi_i, \tag{2.29}$$

$$a = \sum_{i=0}^{P} a_i \psi_i. \tag{2.30}$$

Therefore, the Galerkin projected system reads:

$$\sum_{k=0}^{P} \sum_{j=0}^{P} a_k(u_j, v_i) c_{kji} = l(v_i), \tag{2.31}$$

for all $i = 0, \ldots, P$, whereas the third order tensor $\mathbb{T}_3 = (c_{ijk})_{i,j,k=0}^{P}$ is defined by

$$c_{ijk} := \frac{\langle \psi_i \psi_j, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}. \tag{2.32}$$

Equation (2.31) can be recast in operator form, which reads:

$$\sum_{j=0}^{P} A_{ij} u_j c_{kji} = l(v_i), \quad \forall i = 0, \dots, P, \tag{2.33}$$

whereas

$$A_{ij} u_j := \sum_{k=0}^{P} a_k(u_j, v_i) c_{kji}, \quad \forall i = 0, \dots, P. \tag{2.34}$$

Therefore, (2.33) represents a linear system of equations given by:

$$\begin{bmatrix} A_{00} & \cdots & A_{0P} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ A_{P0} & \cdots & A_{PP} \end{bmatrix} \begin{bmatrix} u_0 \\ \vdots \\ \vdots \\ u_P \end{bmatrix} = \begin{bmatrix} l(v_0) \\ \vdots \\ \vdots \\ l(v_P) \end{bmatrix}. \tag{2.35}$$

The main focus of this section is to analyze the sparsity structure of the system operator $A = (A_{ij})_{i,j=0}^{P}$ defined in (2.35). To this purpose, a sparsity ratio S is being defined by:

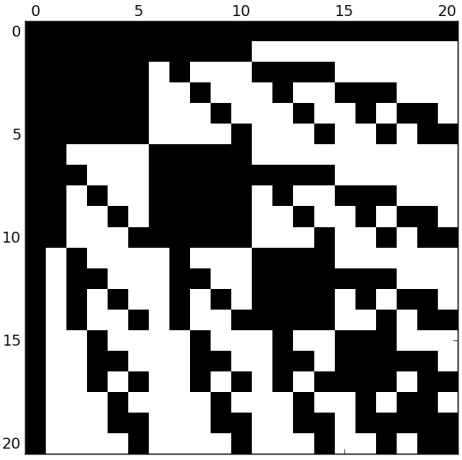
$$S := \frac{\#\{(i,j) : A_{ij} \neq 0\}}{(P+1)^2} \leq 1, \tag{2.36}$$

which represents a measure for the general number of nonzero blocks in A divided by the total number of blocks $(P+1)^2$. It is important to note, that this ratio is defined for a generic operator A resulting from (2.33), i.e., the sparsity ratio S is independent of the concrete definition of the bilinear form a, instead it measures the contribution of the stochastic tensor $\mathbb{T}_3$ on the structure of system (2.35) in a generic way. The calculation of S can be carried out easily by identification of tuples (i,j), $i,j = 0, \dots, P$, for which $c_{kji} = 0$ for all $k = 0, \dots, P$.

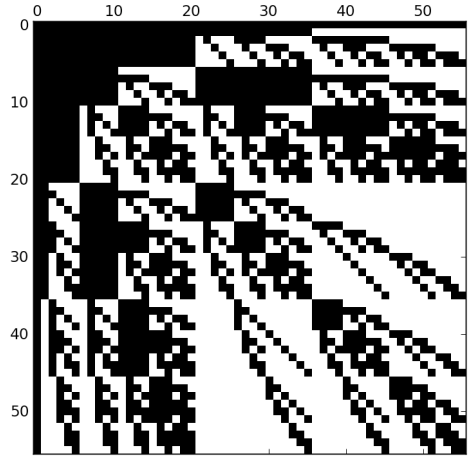
Table 2.3, Fig. 2.1 and Fig. 2.2 illustrate the dependency of S on the dimension L of the random vector ξ and the employed polynomial degree No in the gPC approximation of the bilinear form a. Here, black blocks represent block entries which can be nonzero in general. It can be observed that for increasing dimension L the sparsity ratio of A decreases almost linear in L, whereas for increasing polynomial degree No the sparsity ratio S is subject to a small variation. Note, that for $L = 1$ there do not exist any general nonzero blocks, leading to a sparsity ratio of $S = 1$. However, the blocks A_{ij}, $i,j = 0, \dots, P$ are often sparse themselves, e.g., due to a Finite-Element discretization of the deterministic variables, introducing an additional sparsity.

2.4.2. Quadratic nonlinearity

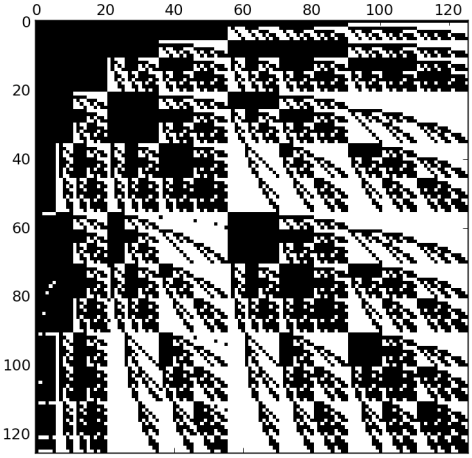
Similar to the previous section, the system structure resulting from a Galerkin projection of a trilinear form shall be analyzed in the following. This is motivated by the problem definition stated in Chapter 5, which involves a quadratic nonlinearity resulting from a stochastic convection term of the Navier-Stokes equations. Therefore, the model problem for this section reads:



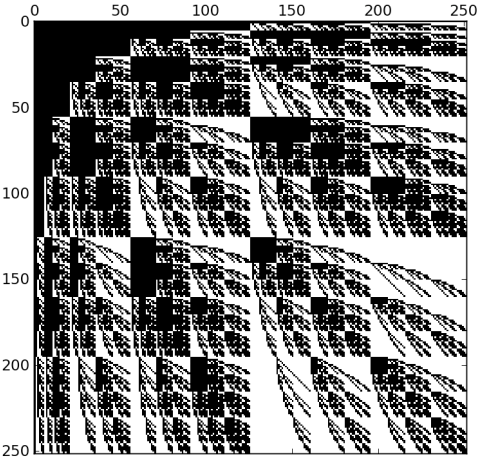
(a) Sparsity structure for $No = 2$, $L = 5$. Sparsity ratio $S = 0.524$. $(P+1)^2 = 441$.



(b) Sparsity structure for $No = 3$, $L = 5$. Sparsity ratio $S = 0.49$. $(P+1)^2 = 3136$.



(c) Sparsity structure for $No = 4$, $L = 5$. Sparsity ratio $S = 0.553$. $(P+1)^2 = 15876$.

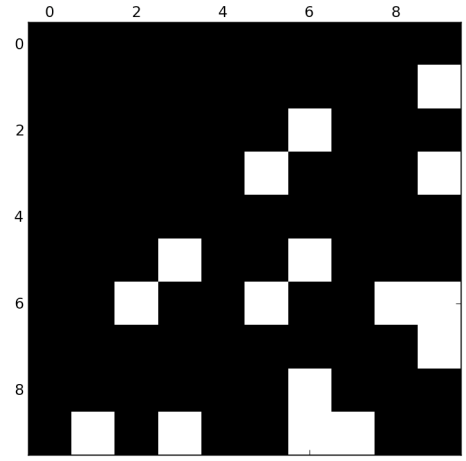


(d) Sparsity structure for $No = 5$, $L = 5$. Sparsity ratio $S = 0.547$. $(P+1)^2 = 63504$.

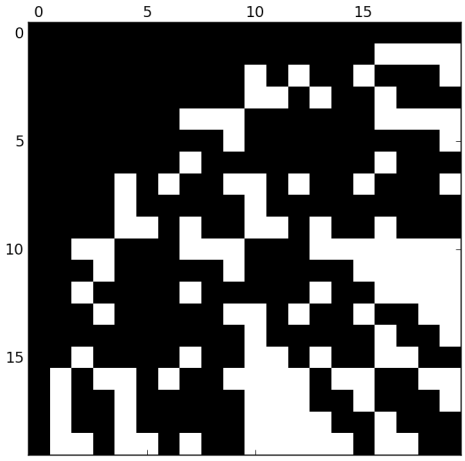
Figure 2.1.: Sparsity structure plots for varying polynomial order No and fixed dimension L of the random vector ξ.

$L \backslash No$	1	2	3	4	5
1	1	1	1	1	1
2	0.778	0.833	0.84	0.867	0.864
3	0.625	0.7	0.7	0.748	0.746
4	0.52	0.6	0.584	0.643	0.64
5	0.444	0.524	0.49	0.553	0.547

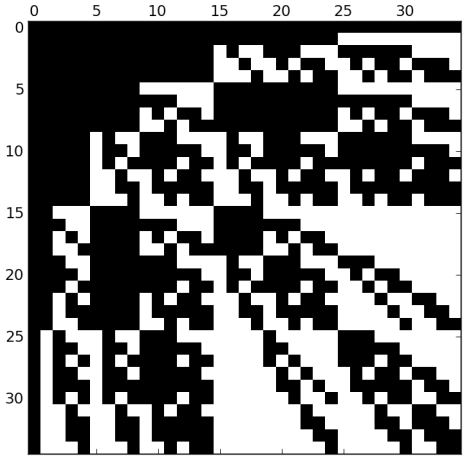
Table 2.3.: Sparsity ratio S for varying polynomial degree No and dimension L of the random vector ξ.



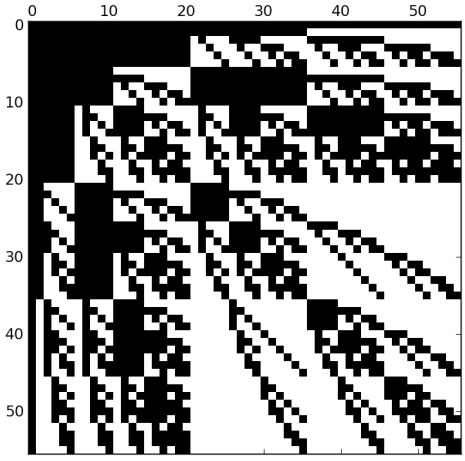
(a) Sparsity structure for $No = 3$, $L = 2$. Sparsity ratio $S = 0.84$. $(P+1)^2 = 100$.



(b) Sparsity structure for $No = 3$, $L = 3$. Sparsity ratio $S = 0.7$. $(P+1)^2 = 400$.



(c) Sparsity structure for $No = 3$, $L = 4$. Sparsity ratio $S = 0.584$. $(P+1)^2 = 1225$.



(d) Sparsity structure for $No = 3$, $L = 5$. Sparsity ratio $S = 0.49$. $(P+1)^2 = 3163$.

Figure 2.2.: Sparsity structure plots for varying dimension L of the random vector ξ and fixed polynomial order No.

Find $u \in L^2(\Omega) \otimes V$ such that

$$n(u, u, v; \xi) = l(v), \quad \forall v \in L^2(\Omega) \otimes W, \tag{2.37}$$

whereas n defines some trilinear form according to the vector spaces $L^2(\Omega) \otimes V$ and $L^2(\Omega) \otimes W$ subject to some random vector ξ of dimension L. One example is given by a stochastic convection term (cf. Chapter 5), for which n is being defined by:

$$n(u, \bar{u}, v; \xi) := T(\xi)((u \cdot \nabla \bar{u}), v) + a(u, v), \tag{2.38}$$

for $T \in L^2(\Omega)$, a being a bilinear form and $u, \bar{u}, v \in L^2(\Omega) \otimes H_0^1(\mathcal{D})^2 \cap \{u : \nabla \cdot u = 0\}$ on some open, bounded and Lipschitz spatial domain $\mathcal{D} \subset \mathbb{R}^2$.

Following an analog derivation of the corresponding Galerkin projected system as for the linear case, one obtains the following system of equations:

$$\sum_{k=0}^{P}\sum_{j=0}^{P}\sum_{l=0}^{P} n_k(u_j, u_l, v_i)c_{kjli} = l(v_i), \quad \forall i = 0, \ldots, P, \tag{2.39}$$

whereas the corresponding gPC expansions employing a finite number of terms are given by:

$$n(u, \bar{u}, v; \xi) = \sum_{k=0}^{P} n_k \psi_k(\xi), \tag{2.40}$$

$$u = \sum_{i=0}^{P} u_i \psi_i, \tag{2.41}$$

$$v = \sum_{i=0}^{P} v_i \psi_i. \tag{2.42}$$

Note, that the main difference to the linear case is reflected by the fact that the third order tensor $\mathbb{T}_3$ is replaced by a corresponding fourth order tensor $\mathbb{T}_4 = (c_{ijlk})_{i,j,l,k=0,\ldots,P}$ given by

$$c_{ijlk} := \frac{\langle \psi_i \psi_j \psi_l, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}. \tag{2.43}$$

Again, equation (2.39) can be recast in operator form, such that

$$\begin{bmatrix} N_{00} & \cdots & N_{0P} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ N_{P0} & \cdots & N_{PP} \end{bmatrix} \begin{bmatrix} u_0 \\ \vdots \\ \vdots \\ u_P \end{bmatrix} = \begin{bmatrix} l(v_0) \\ \vdots \\ \vdots \\ l(v_P) \end{bmatrix}, \tag{2.44}$$

whereas the nonlinear block operators N_{ij}, $i, j = 0, \ldots, P$ are defined by:

$$N_{ij}u_j := \sum_{k=0}^{P}\sum_{l=0}^{P} n_k(u_j, u_l, v_i)c_{kjli}, \quad \forall i = 0, \ldots, P. \tag{2.45}$$

However, there is no corresponding general sparsity structure observable as being the case for the linear operator involving the third order tensor $\mathbb{T}_3$, resulting in a sparsity ratio $S = 1$ for the nonlinear system given in (2.44). Therefore, the Galerkin projection does not introduce additional sparsity in general. However, the fourth order tensor $\mathbb{T}_4$ is sparse itself [25], which also plays an important role, since many of its entries are equal to zero resulting in a significant reduction of the summands in (2.39). Especially, if the assembly of the operators N_{ij}, $i, j = 0, \ldots, P$ is numerically costly, the total assembly of the system matrix $N = (N_{ij})_{i,j=0}^{P}$ benefits heavily from the sparsity of $\mathbb{T}_4$. A similar result is valid regarding the third order tensor $\mathbb{T}_3$ [25].

Chapter 3.

Time-dependent generalized Polynomial Chaos for ordinary differential equations

A well known difficulty when employing gPC is the possible convergence breakdown in cases involving strong nonlinear dependencies on the random vector ξ. These cases can occur, for example, when dealing with long term integration or stochastic discontinuities in context of differential equations. Since a stationary, i.e., time-independent approach is used, the time evolution of the probability distribution of the solution to some underlying differential equation cannot be captured efficiently after some application dependent critical time. This can be seen clearly when taking a look at the discretization parameter No (cf. Chapter 2), which only allows for nonlinear dependencies up to the order of No, since only polynomials of maximum order No are employed in a corresponding gPC expansion. This problem has been studied in various works, e.g., [8, 14, 26, 38, 44, 45], leading to promising modifications towards gPC to overcome the mentioned convergence breakdown. This chapter will recapitulate and extend one of the more recent approaches introduced by Gerritsma et al. [14], termed the "time-dependent generalized Polynomial Chaos" (TD-gPC), in context of ordinary differential equations subject to a random input. Furthermore, the statements of this chapter are based on the published work of the author in [19, 20] of the year 2011.

3.1. Nonlinear dependencies on the random input

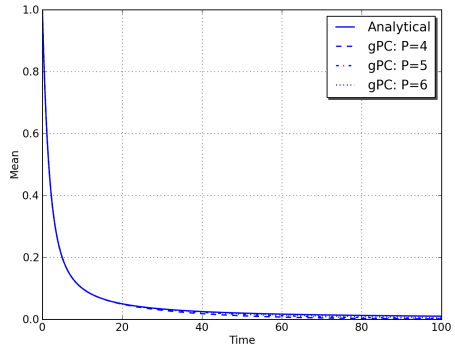
Although gPC exhibits exponential convergence for a lot of numerical applications, the question arises if an optimal convergence property can be maintained throughout a time- and space-dependent numerical simulation. For elaboration on this question, a simple one-dimensional ordinary differential equation will be considered, showing that the standard gPC expansion is failing quickly to approximate the exact solution accurately. Therefore, the model equations are given by:

$$\frac{d}{dt}u(t;\xi) = -k(\xi)u(t;\xi), \quad \text{for } t \in (0,T], \tag{3.1}$$

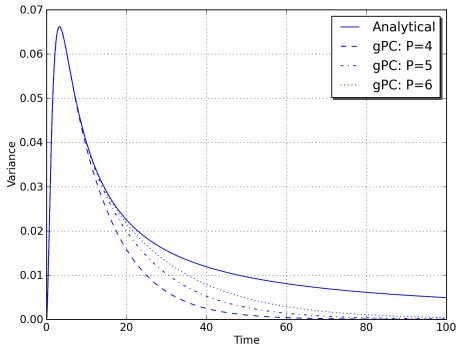
$$u(0;\xi) = 1, \tag{3.2}$$

whereas $\xi \in L^2(\Omega)$ and $k(\xi) := \frac{1}{2}(1+\xi)$. The reason why this problem is being considered lies in its simple structure and the fact that its exact solution u_{exact} can be calculated analytically and is given by:

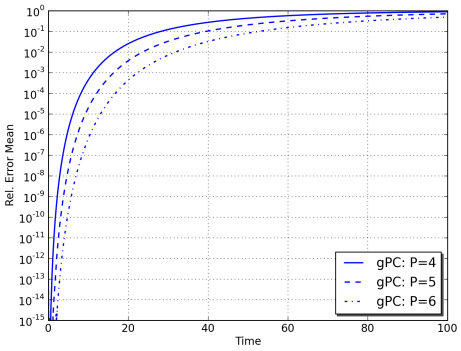
$$u_{\text{exact}}(t;\xi) = \exp(-k(\xi)t), \quad t \in [0,T]. \tag{3.3}$$



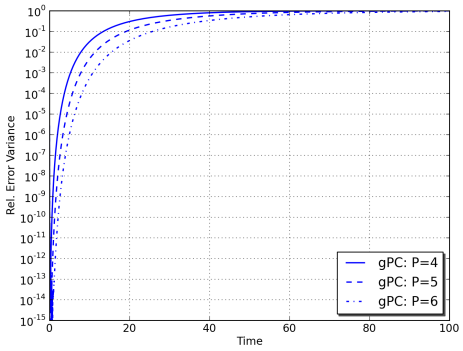
(a) Analytical and computed mean.



(b) Analytical and computed variance.



(c) Relative error mean.



(d) Relative error variance.

Figure 3.1.: Calculated mean and variance employing gPC for $P = 4, 5, 6$ and their corresponding relative errors subject to a uniformly distributed input.

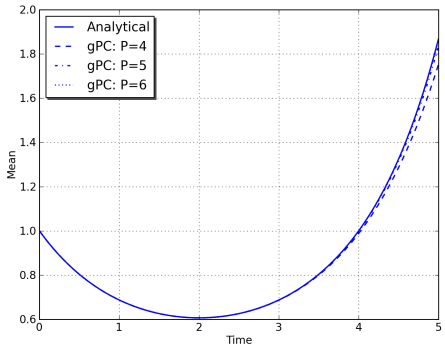
Taking a closer look on u_{exact} by expressing it through its series representation

$$u_{\text{exact}}(t;\xi) = \sum_{n=0}^{\infty} (-1)^n \frac{(1+\xi)^n t^n}{2^n n!}, \tag{3.4}$$

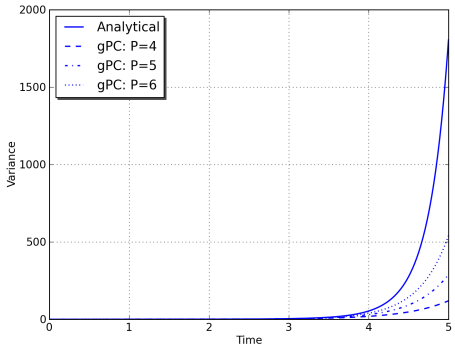
it is observable that if $t > 0$ gets large the nonlinear dependencies on ξ become more dominant. However, a gPC expansion of order No, i.e.,

$$u(t;\xi) = \sum_{i=0}^{P} u_i(t)\psi_i(\xi), \tag{3.5}$$

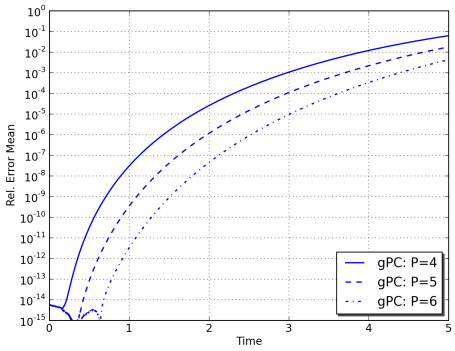
whereas $P = No$ (note that $L = 1$ for this case, cf. Chapter 2), is restricted to a time-independent fixed maximum polynomial degree of order P. Therefore (3.5) is only capable of representing nonlinear dependencies on ξ up to the order of P. This leads to an increasingly poor approximation quality for large $t > 0$, since the decomposition is only accurate for early times $t < t_{\text{crit}}$. This convergence breakdown is illustrated by a numerical computation of the mean and the variance of a solution to (3.1), (3.2), for which the relative errors between the exact u_{exact}



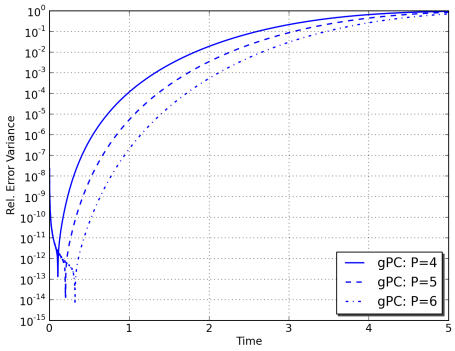
(a) Analytical and computed mean.



(b) Analytical and computed variance.



(c) Relative error mean.



(d) Relative error variance.

Figure 3.2.: Calculated mean and variance employing gPC for $P = 4, 5, 6$ and their corresponding relative errors subject to a Gaussian distributed input.

and computed solution u_P are defined by:

$$\epsilon_{\text{mean}} := \frac{|\mathbb{E}(u_{\text{exact}}) - \mathbb{E}(u_P)|}{|\mathbb{E}(u_{\text{exact}})|}, \tag{3.6}$$

$$\epsilon_{\text{variance}} := \frac{|\sigma^2(u_{\text{exact}}) - \sigma^2(u_P)|}{|\sigma^2(u_{\text{exact}})|}. \tag{3.7}$$

Fig. 3.1 and Fig. 3.2 depict the mean and the variance of an exact and computed solution to the model problem (3.1), (3.2) and their corresponding relative errors arising from the numerical discretization employing a 4^{th} order Runga-Kutta scheme with a homogeneous time step size $\Delta t = 0.001$. Fig 3.1 considers a Uniform distributed $\xi \sim U(-1, 1)$, while Fig. 3.2 considers a Gaussian distributed $\xi \sim \mathcal{N}(0, 1)$, both employing a gPC expansion of order $P = 4, 5, 6$. It can be observed, that for increasing time $t > 0$ the gPC expansion is not capable of representing the solution's moments accurately, leading to quickly growing unfeasible relative error levels of the order of $O(1)$ for both the mean and the variance.

Summing up, due to the fixed order P of the gPC the errors are growing in time. This can be attributed to the fact that only nonlinear dependencies on ξ up to the

order P can be approximated accurately. Since the solution u_{exact} is of exponential type, this restriction does not suffice anymore if $t > 0$ gets large enough leading to unacceptable error levels. Hence, the generalized Polynomial Chaos expansion is only efficient for times $t < t_{\text{crit}}$ for which the nonlinear dependencies do not have a huge impact on the solution's stochastic dynamic behavior.

3.2. Time-dependent basis functionals

This section is focused on the illustration of the basic principles of the time-dependent gPC (TD-gPC) for a scalar valued stochastic process $u = u(t;\xi)$ subject to a random input ξ with some arbitrary probability distribution. Generalizations to systems of differential equations will be given in Section 3.3.

It is assumed that $u \in L^2(\Omega)$ for all times $t \geq 0$, such that a standard gPC expansion can be employed at each time t. Recapitulate, that a P^{th} order decomposition of u reads:

$$u(t;\xi) = \sum_{i=0}^{P} u_i(t)\psi_i(\xi), \tag{3.8}$$

whereas ψ_i, $i = 0, \ldots, P$, denote the Chaos Polynomials according to the Askey-Chaos (cf. Table 2.1). Note, that since ξ is assumed to be scalar valued, the maximum polynomial degree No employed in the gPC expansion of u is equal to the number of terms minus one within equation (3.8), i.e., $No = P$. At this stage, the gPC expansion is only capable of representing nonlinear dependencies on ξ up to the order P (cf. Section 3.1). Therefore, it is desirable to reduce the degree of the nonlinear dependency ideally towards a linear dependency at each time $t > 0$, which would yield an optimal (exact) expansion, since all coefficients corresponding to the nonlinear parts would vanish. Therefore, the gPC expansion would be capable of representing u accurately at each time $t \geq 0$.

The questions at this point are: Is a decrease in the degree of the nonlinear dependency possible? and if yes, then how? An answer to these questions will be given in the following subsections.

3.2.1. Transformation of the probability measure

For the subsequent analysis the time t is fixed at $t = t^* > 0$. It is observed that a new random variable η can be defined by:

$$\eta(\xi) := u(t^*;\xi) = \sum_{i=0}^{P} u_i(t^*)\psi_i(\xi). \tag{3.9}$$

The properties of η are that it is a functional depending on ξ and representing the discretized solution at time $t = t^*$. Furthermore, since it is assumed that u is square-integrable at all times $t \geq 0$, the same holds for η, i.e., $\eta(\xi) \in L^2(\Omega)$. The goal at this stage is to express u via a transformed gPC expansion with respect to η, which is trivial at $t = t^*$. However, for $t > t^*$ this will result in nonlinear dependencies on η of lower order compared to the dependencies on ξ leading to a

better approximation quality of the stochastic dynamics of u. Carrying out this step, a "transformed" decomposition of u in terms of η is given by:

$$u(t;\eta) = \sum_{i=0}^{P} \tilde{u}_i(t)\tilde{\psi}_i(\eta), \quad \text{for } t \geq t^*. \tag{3.10}$$

Here, $\{\tilde{u}_i\}_{i=0}^P$ denote the modes with respect to the new Chaos Polynomial basis functionals $\{\tilde{\psi}_i\}_{i=0}^P$, which satisfy the orthogonality condition:

$$\int_{\text{supp } F_\eta} \tilde{\psi}_i(\eta)\tilde{\psi}_j(\eta)\ dF_\eta(\eta) = \langle \tilde{\psi}_i, \tilde{\psi}_i \rangle \delta_{ij}, \quad i,j = 0,\dots,P, \tag{3.11}$$

whereas F_η denotes the probability distribution function of η. Note, that this expansion is a further extension of the Askey-Chaos (cf. Table 2.1), with respect to the probability distribution of u at $t = t^*$. However, in order to be able to determine the new basis $\{\tilde{\psi}_i\}_{i=0}^P$, knowledge about F_η is required. In practical calculations, computing F_η is often not feasible, therefore the integration in (3.11) is transformed back to ξ in order to be able to use the knowledge about the initial probability distribution on ξ, resulting in:

$$\int_{\text{supp } F_\eta} \tilde{\psi}_i(\eta)\tilde{\psi}_j(\eta)\ dF_\eta(\eta) = \int_{\mathbb{R}} \tilde{\psi}_i(u(t^*;\xi))\tilde{\psi}_j(u(t^*;\xi)) f_\xi(\xi)\ d\xi, \tag{3.12}$$

for $i,j = 0,\dots,P$, with f_ξ denoting the probability density function of ξ.

3.2.2. Projection onto the new basis

When dealing with ordinary differential equations, one needs to specify new initial conditions at $t = t^*$ such that a Galerkin projection in terms of η can be applied for $t > t^*$. This can be achieved by a projection of u onto the new basis via the following relations, which hold almost surely in Ω:

$$\sum_{i=0}^{P} u_i(t^*)\psi_i(\xi) = \sum_{i=0}^{P} \tilde{u}_i(t^*)\tilde{\psi}_i(\eta) \tag{3.13}$$

$$\Rightarrow \sum_{i=0}^{P} u_i(t^*)\langle \psi_i, \tilde{\psi}_j \rangle = \sum_{i=0}^{P} \tilde{u}_i(t^*)\langle \tilde{\psi}_i, \tilde{\psi}_j \rangle, \quad j = 0,\dots,P \tag{3.14}$$

$$\Rightarrow \tilde{u}_j(t^*) = \sum_{i=0}^{P} u_i(t^*)\frac{\langle \psi_i, \tilde{\psi}_j \rangle}{\langle \tilde{\psi}_j, \tilde{\psi}_j \rangle}, \quad j = 0,\dots,P \quad \text{by orthogonality.} \tag{3.15}$$

The inner product $\langle \psi_i, \tilde{\psi}_j \rangle$ taking into account ψ_i and $\tilde{\psi}_j$ can be calculated by:

$$\langle \psi_i, \tilde{\psi}_j \rangle = \int_{\mathbb{R}} \psi_i(\xi)\tilde{\psi}_j(u(t^*;\xi)) f_\xi(\xi)\ d\xi. \tag{3.16}$$

Usually, the first functional $\tilde{\psi}_0$ is being defined as a constant functional equal to 1. For a polynomial basis the next functional can be defined as a linear polynomial with

leading coefficient equal to 1, namely, $\tilde{\psi}_1(\eta) := \eta - \alpha$ for some $\alpha \in \mathbb{R}$. Recapitulate, that at $t = t^*$ it holds:

$$\eta = \sum_{i=0}^{P} \tilde{u}_i(t^*)\tilde{\psi}_i(\eta), \tag{3.17}$$

which can be simplified to the conditions

$$\tilde{u}_0(t^*) + \tilde{u}_1(t^*)(\eta - \alpha) = \eta \tag{3.18}$$

$$\tilde{u}_i(t^*) = 0, \quad i \geq 2, \tag{3.19}$$

leading to $\tilde{u}_0(t^*) = \alpha$ and $\tilde{u}_1(t^*) = 1$. By taking the expectation in (3.13) and interchanging integration and summation one obtains:

$$\sum_{i=0}^{P} u_i(t^*)\langle\psi_i\rangle = \sum_{i=0}^{P} \tilde{u}_i(t^*)\langle\tilde{\psi}_i\rangle. \tag{3.20}$$

Due to the orthogonality conditions for both ψ_i and $\tilde{\psi}_i$ with respect to the constant functional 1, i.e., $\langle\psi_i, 1\rangle = 0$ and $\langle\tilde{\psi}_i, 1\rangle = 0$ for $i > 0$, it holds:

$$u_0(t^*) = \tilde{u}_0(t^*). \tag{3.21}$$

Therefore, in the case of a scalar valued ordinary differential equation subject to a random input, the modes after the "transformation of measure" at time $t = t^*$ are analytically given by:

$$\tilde{u}_0(t^*) = u_0(t^*), \tag{3.22}$$

$$\tilde{u}_1(t^*) = 1, \tag{3.23}$$

$$\tilde{u}_i(t^*) = 0, \quad \text{for } i \geq 2, \tag{3.24}$$

if a polynomial basis is employed with $\tilde{\psi}_0 = 1$ and a normalized leading coefficient of $\tilde{\psi}_1$. This allows for a numerically cost efficient way of transforming the solution u onto the new basis functionals in the random space. However, Section 3.2.3 will introduce modifications to the time-dependent gPC expansion, for which the analytical expressions will not hold, requiring the solution of the projected system given by equation (3.15). Note, that the computation of the mean and the variance of a solution u can be carried out analog to the case employing the standard gPC expansion, i.e.,

$$\bar{u}(t) = u_0(t), \tag{3.25}$$

$$\sigma^2(u)(t) = \sum_{i=1}^{P} u_i(t)^2\langle\tilde{\psi}_i, \tilde{\psi}_i\rangle, \tag{3.26}$$

due to the orthogonality of the Chaos Polynomials $\{\tilde{\psi}_i\}_{i=0}^{P}$.

3.2.3. Modified TD-gPC

When dealing with a system of ordinary differential equations $\frac{d}{dt}u = f(u,t;\xi)$, which involves an explicit dependence of the right hand side f on the random input ξ, the basis as being constructed in the previous section might not be sufficient. This issue is being illustrated in the following example:

Suppose $f = f(u;\xi) := -\xi u$ for some initial scalar stochastic input random variable ξ subject to some arbitrary probability distribution. Differentiating f with respect to t by applying the chain rule, yields:

$$\frac{d^i}{dt^i} f(u;\xi) = (-1)^{i+1}\xi^{i+1}u, \quad i \geq 0. \tag{3.27}$$

Furthermore, the Taylor expansion of u is given by:

$$u(t^* + \Delta t;\eta) = u(t^*;\eta) + \Delta t \frac{d}{dt}u(t^*;\eta) + \frac{(\Delta t)^2}{2}\frac{d^2}{dt^2}u(t^*;\eta) + \ldots, \tag{3.28}$$

$$= \eta - \Delta t\xi\eta + \frac{(\Delta t)^2}{2}\xi^2\eta - \ldots. \tag{3.29}$$

From this it can be seen that a basis in terms of η cannot approximate the time derivatives of u in an exact way due to the explicit dependence of f on ξ. Therefore, TD-gPC as introduced in the previous section is not feasible for this class of applications leading to an error contribution of order $O(\Delta t)$. However, it is possible to overcome this drawback by defining a new basis in terms of η and ξ via a Tensor product, such that u can be expressed by the expansion:

$$u(t;\omega) = \sum_{i=0}^{P}\sum_{j=0}^{Q} u_{ij}(t)\psi_i(\eta(\omega))\phi_j(\xi(\omega)), \tag{3.30}$$

whereas $\{\psi_i\}_{i=0}^{P}$ and $\{\phi_j\}_{j=0}^{Q}$ are orthogonal polynomials with respect to the probability distribution of η and ξ, respectively. By a technical calculation it can be easily shown that such a basis is capable of representing the time derivatives of u to the order of $O((\Delta t)^{Q+1})$. The next section will demonstrate this basis extension with respect to a system of differential equations and give numerical results, which display the improved convergence behavior.

3.3. System of differential equations - A linear oscillator

To the present knowledge of the author, the modifications of TD-gPC described in Section 3.2.3 have not been yet considered for systems of differential equations, except in the work of the author himself [19]. In the original work of Gerritsma et al. [14] only the non-modified version of TD-gPC has been applied to a system of differential equations, yet subject to a variation of the approach outlined in the following. The results presented in [19] are being elaborated on within the next sections, showing how the modifications in Section 3.2.3 can be adapted in that context. This is exemplified considering a linear oscillator exhibiting stable limit cycles. For this problem it is well known that the classical gPC expansion fails to capture the dynamics of the solution after some certain critical time [26].

3.3.1. Model equations

Consider the equations of motion of a linear oscillator in two dimensions:

$$\frac{d}{dt}x_1(t) = x_2(t), \tag{3.31}$$

$$\frac{d}{dt}x_2(t) = -qx_1(t), \tag{3.32}$$

for $t \in [0,T] \subset \mathbb{R}$ with $q > 0$, position x_1 and impulse $x_2 = \dot{x}_1$. The frequency of the system is $\sqrt{q}/2\pi$ and the initial conditions are set to $x_1(t=0) = 1$ and $x_2(t=0) = 0$. In the following a random frequency, i.e., $q = q(\xi) = q_0 + q_1\xi$, shall be considered subject to a scalar random variable ξ, which is uniformly distributed within the real interval $(-1,1)$, i.e., $\xi \sim U(-1,1)$. The analytical solutions can easily be calculated and are given by:

$$x_1(t;\xi) = \cos(\sqrt{q(\xi)}t), \tag{3.33}$$

$$x_2(t;\xi) = -\sqrt{q(\xi)}\sin(\sqrt{q(\xi)}t). \tag{3.34}$$

Here, it can already be seen that the solution (x_1, x_2) has a strong nonlinear dependency on the random frequency $\sqrt{q}/2\pi$.

3.3.2. Discretization employing TD-gPC

Since in contrast to the previous sections a vector-valued problem in two dimensions is being considered, the procedure, as described in Section 3.2.3, is extended to this case and analyzed at some reset time $t = t^* > 0$. The starting point is the standard gPC discretization of x_1 and x_2 given by:

$$x_1(t;\xi) = \sum_{i=0}^{P} x_{1,i}(t)L_i(\xi), \tag{3.35}$$

$$x_2(t;\xi) = \sum_{i=0}^{P} x_{2,i}(t)L_i(\xi), \tag{3.36}$$

whereas $\{L_i\}_{i=0}^{P}$ denotes the Legendre polynomials in terms of the uniformly distributed random variable ξ. Here it holds $No = P$, since a one-dimensional random input is being considered (cf. Chapter 2). Therefore, the index i of L_i, $i = 0,\dots,P$, equals to the degree of the considered Legendre polynomial. At some reset time $t = t^*$ two new random variables are defined corresponding to the solution components:

$$\eta^{(1)}(\xi) := \sum_{i=0}^{P} x_{1,i}(t^*)L_i(\xi), \tag{3.37}$$

$$\eta^{(2)}(\xi) := \sum_{i=0}^{P} x_{2,i}(t^*)L_i(\xi). \tag{3.38}$$

From this point on, a multi-dimensional stochastic input, given by $\eta^{(1)}$ and $\eta^{(2)}$, has to be considered. This needs to be taken into account when employing a gPC

expansion in terms of the new random variables. However, since $\eta^{(1)}$ and $\eta^{(2)}$ are dependent random variables via ξ, a modification to the standard approach is suggested to maintain orthogonality of the multi-dimensional basis functionals. Therefore, first orthogonal Chaos Polynomials $\psi_j^{(i)}$, $i = 1, 2$, $j = 0, \ldots, P$, are being computed with respect to each random variable $\eta^{(i)}$, $i = 1, 2$, such that:

$$\int \psi_s^{(i)}(\eta)\psi_r^{(i)}(\eta) f_{\eta^{(i)}}(\eta)\, d\eta = \langle \psi_s^{(i)}, \psi_s^{(i)} \rangle \delta_{sr}, \quad s, r = 0, \ldots, P,\ i = 1, 2. \tag{3.39}$$

This can be achieved, for example, by employing a Gram-Schmidt orthogonalization method, subject to a "twice is enough" modification [16], which improves the numerical stability. Note that the computation of the integral in (3.39) can be transformed to the original random variable ξ as described in Section 3.2.1 to avoid the explicit calculation of the probability density functions $f_{\eta^{(i)}}$ of $\eta^{(i)}$, $i = 1, 2$.

Next, a new temporary basis is defined by a tensor product of the corresponding Chaos Polynomials via:

$$x_k(t; \eta^{(1)}; \eta^{(2)}) = \sum_{0 \le i+j \le P} \tilde{x}_{k,ij}(t) \psi_i^{(1)}(\eta^{(1)}) \psi_j^{(2)}(\eta^{(2)}), \quad t \ge t^*,\ k = 1, 2, \tag{3.40}$$

which equivalently can be expressed by:

$$x_k(t; \eta^{(1)}; \eta^{(2)}) = \sum_{j=0}^{M} x_{k,j}(t) \phi_j(\eta^{(1)}, \eta^{(2)}), \quad t \ge t^*,\ k = 1, 2, \tag{3.41}$$

via a one-to-one correspondence between the basis functionals and coefficients in equations (3.40) and (3.41), whereas for the number of terms $M + 1$ in (3.41) it holds:

$$M + 1 = \frac{(P + 2)!}{P!2!} = \frac{(P + 1)(P + 2)}{2}. \tag{3.42}$$

Note that at this stage, the basis polynomials $\{\phi_i\}_{i=0}^{M}$ are not orthogonal to each other because of the dependency of $\eta^{(1)}$ and $\eta^{(2)}$, introduced through ξ. Therefore, the basis is once again orthogonalized via a Gram-Schmidt method in two dimensions, maintaining an orthogonal projection of x_1 and x_2. To this end an orthogonal basis is defined by:

$$\psi_0 := 1, \tag{3.43}$$

$$\psi_i(\eta^{(1)}, \eta^{(2)}) := \phi_i(\eta^{(1)}, \eta^{(2)}) - \sum_{j=0}^{i-1} \frac{\langle \phi_i, \psi_j \rangle}{\langle \psi_j, \psi_j \rangle} \psi_j(\eta^{(1)}, \eta^{(2)}), \quad i = 1, \ldots, M. \tag{3.44}$$

Since now an orthogonal basis is employed, it is straight forward to calculate the required initial conditions at $t = t^*$ by a projection similar to the one introduced in Section 3.2.3:

$$x_{k,j}^{(new)}(t^*) = \sum_{i=0}^{M} x_{k,i}^{(old)}(t^*) \frac{\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle}{\langle \psi_j^{(new)}, \psi_j^{(new)} \rangle}, \quad \text{for } j = 0, \ldots, M,\ k = 1, 2. \tag{3.45}$$

Here, $x_{k,j}^{(old)}$ and $x_{k,j}^{(new)}$, $j = 0, \ldots, P$, $k = 1, 2$ denote the modes of the solution (x_1, x_2) before and after the reset step at $t = t^*$, respectively. Similary, $\psi_j^{(old)}$ and $\psi_j^{(new)}$ denote the corresponding Chaos Polynomials before and after the reset step, respectively. Note, that in case of the first reset within the simulation time interval $[0, T]$ all modes of the "old" solution with index $i > P$ are set to zero. However, it is also possible to start with a lower expansion order and add the new required terms initialized with zero when needed. The procedure described above is then repeated at every qualified time step, which can be identified by some error estimation criteria [14] or chosen to be each time step within the time discretization procedure.

For post processing purposes the mean $\bar{x}_1, \bar{x}_2$ and the variances $\sigma^2(x_1), \sigma^2(x_2)$ can be calculated in the same manner as for the standard gPC expansion via:

$$\bar{x}_i(t) = x_{i,0}(t), \tag{3.46}$$

$$\sigma^2(x_i)(t) = \sum_{j=1}^{M} (x_{i,j})^2 \langle \psi_j, \psi_j \rangle, \tag{3.47}$$

for $i = 1, 2$, due to the orthogonal nature of the projection.

3.3.3. Extension of the modified TD-gPC

Before stating the numerical results, one major drawback of TD-gPC is analyzed as already described in Section 3.2.3, namely a convergence breakdown when the uncertain parameter is explicitly involved within the differential equation (this is the case here) and not exclusively in initial conditions. For this purpose the second equation of the model problem (3.32) is investigated in more detail:

$$\frac{d}{dt} x_2 = -q(\xi) x_1. \tag{3.48}$$

Employing any deterministic time-discretization scheme represented by some function g, w.l.o.g. subject to a homogeneous time step size $\Delta t > 0$, results in:

$$x_2(t + \Delta t; \xi) = g(x_1(t; \xi), x_1(t + \Delta t; \xi), t; \xi). \tag{3.49}$$

Here the explicit dependency of g on ξ is crucial. If changing the variables from ξ to $\eta^{(i)}$, $i = 1, 2$ due to TD-gPC, an optimal representation of the solution x_1, x_2 itself is obtained at every time step, however, the solution's time evolution cannot be captured accurately, i.e., its time derivative in terms of the new random variables cannot be represented efficiently in a TD-gPC expansion, since g is still depending on the initial random variable ξ. Hence, when progressing in time, the error made because of a poor representation of the time derivative increases steadily, leading to unfeasible results similar to the case exemplified in Section 3.2.3. Therefore, the temporary basis $\psi_i^{(1)}, \psi_i^{(2)}$ is modified to take into account the initial random

variable ξ, resulting in:

$$x_1(t;\eta^{(1)},\eta^{(2)},\xi) = \sum_{0\leq i+j\leq P}^{P}\sum_{k=0}^{Q} x_{ijk}^{(1)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\xi), \tag{3.50}$$

$$x_2(t;\eta^{(1)},\eta^{(2)},\xi) = \sum_{0\leq i+j\leq P}^{P}\sum_{k=0}^{Q} x_{ijk}^{(2)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\xi), \tag{3.51}$$

whereas L_k, $k = 0,\ldots,Q$, denote the Legendre polynomials in terms of ξ according to a gPC expansion of order Q. Next a new orthogonal basis $\{\psi_j\}_{j=0}^{M}$ is constructed analog to the procedure described in the previous section, resulting in:

$$x_i(t;\eta^{(1)},\eta^{(2)},\xi) = \sum_{j=0}^{M} x_j^{(i)}(t)\psi_j(\eta^{(1)},\eta^{(2)},\xi), \quad t \geq t^*,\ i = 1,2, \tag{3.52}$$

with the number of terms given by:

$$M+1 = \frac{(P+1)(P+2)(Q+1)}{2}. \tag{3.53}$$

Note, that the number of terms $M+1$ employed in the modified TD-gPC is of factor $(Q+1)$ larger then a corresponding standard TD-gPC expansion as described in Section 3.3.2. The calculation of the initial values at time $t = t^*$ is carried out using the projection described in (3.45). The same holds regarding the calculation of the mean and the variance.

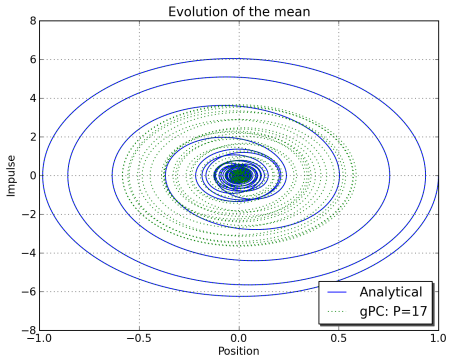
3.3.4. Numerical results

Next, numerical results are presented with respect to various TD-gPC expansion orders. The random frequency is defined to be:

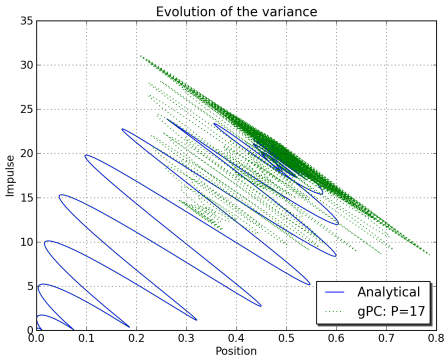
$$q(\xi) := 4\pi^2(1+0.2\xi). \tag{3.54}$$

An explicit Runge-Kutta scheme of order 4 is employed for time discretization with a homogeneous time step size $\Delta t = 0.001$ to minimize the error contributions introduced by the time discretization. Furthermore, a reset is carried out in every time step throughout the simulation interval $[0, 75]$. To reduce the errors arising from the numerical integration a Gauss-Legendre quadrature rule with 100 quadrature points is employed. The results concerning the absolute errors of the time trajectories of the mean and the variance as well as their relative errors regarding the first solution component x_1 are presented in Fig. 3.3 and Fig. 3.4, respectively. Since a discretization employing the time-dependent approach results in some certain total number of modes $M+1$, the results of TD-gPC are compared to the standard gPC using the same number of modes $M+1$, e.g., $P = 2$ and $Q = 2$ for a TD-gPC expansion equals a total number of $M+1 = 18$ modes, i.e., $P = 17$ for a standard gPC expansion.

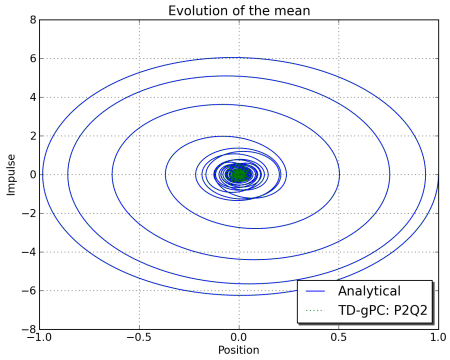
As expected, the standard gPC employing Legendre polynomials is only capable of following the solution for early times even for a large number of modes. The



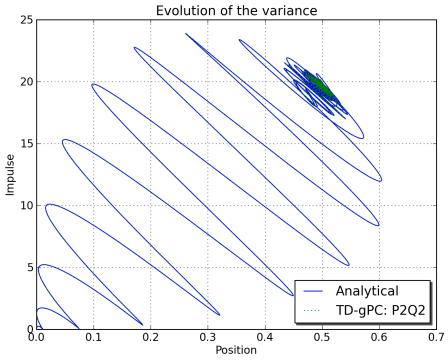
(a) Standard Legendre Chaos $P = 17$.



(b) Standard Legendre Chaos $P = 17$.



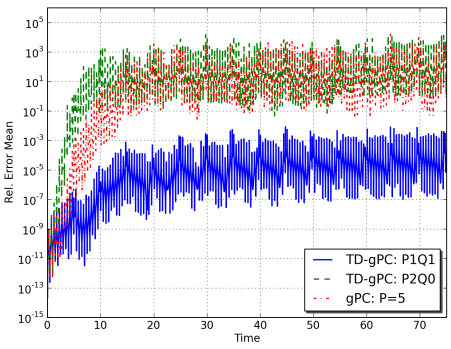
(c) Time-dependent variant $P = 2, Q = 2$.



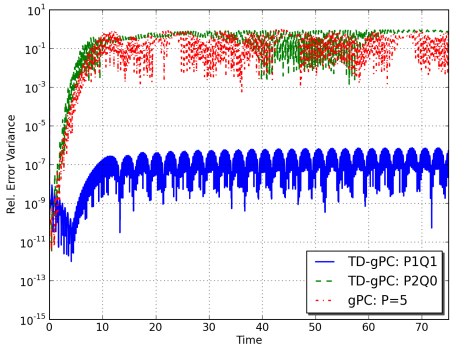
(d) Time-dependent variant $P = 2, Q = 2$.

Figure 3.3.: Evolution of the trajectories corresponding to the mean and the variance for the standard gPC of order $P = 17$ and the time-dependent variant employing $P = 2$, $Q = 2$, both equal to a total number of 18 modes.

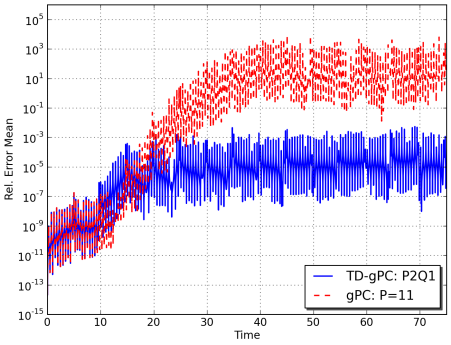
time-dependent approach, however, is performing slightly worse if $Q = 0$, i.e., the errors arising from a poor representation of the time derivative start to dominate quickly. If this is taken into account by increasing the expansion order Q to $Q = 1$ and $Q = 2$, TD-gPC converges to almost exact results w.r.t. the relative errors. Optimal results are achieved employing $P = 2$ and $Q = 2$, which lead to an optimal basis to represent the solution itself and its time derivative. It is interesting to point out the convergence property regarding P and Q in context of TD-gPC. If comparing the results for $P1Q1$ and $P2Q1$ there are no significant errors improvements achieved. In contrast, comparing the results for $P2Q0$, $P2Q1$ and $P2Q2$ an exponential convergence property is observable with respect to Q. However, using $P = 0$ and $Q > 0$, TD-gPC is actually equal to the standard gPC and therefore leading to unfeasible results. This emphasizes the importance of choosing an optimal basis both for the solution itself as well as for its time derivative.



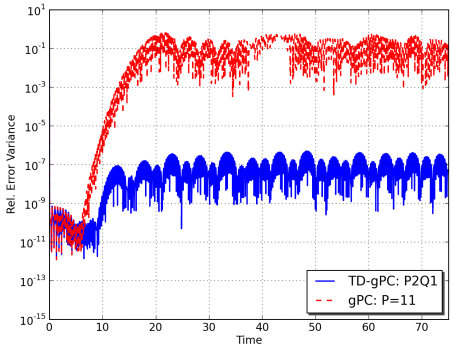
(a) Relative error of the mean with a total of 6 modes each.



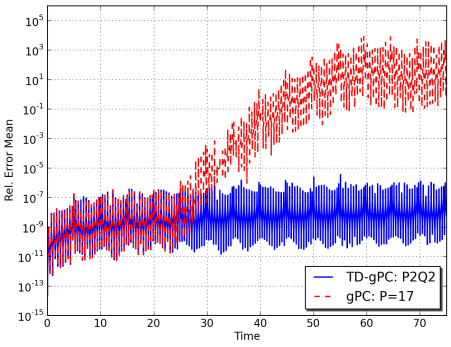
(b) Relative error of the variance with a total of 6 modes each.



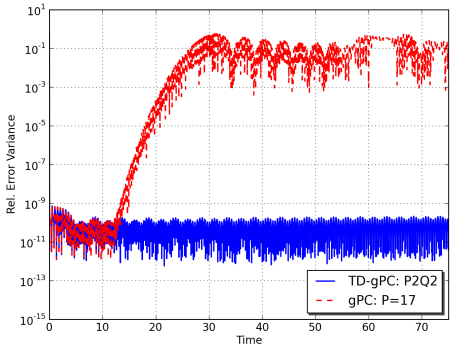
(c) Relative error of the mean with a total of 12 modes each.



(d) Relative error of the variance with a total of 12 modes each.



(e) Relative error of the mean with a total of 18 modes each.



(f) Relative error of the variance with a total of 18 modes each.

Figure 3.4.: Relative errors of the mean and the variance of x_1 corresponding to various discretization parameters.

3.4. Analysis of the reset steps

The key point of applying the (modified) TD-gPC approach to systems of differential equations lies in the shift of the nonlinear complexity towards the numerical evaluation of the integrals involved when computing the inner-product of the Chaos basis functionals $\{\psi_i\}_{i=0}^{M}$. For simplicity, the analysis is elaborated on the first reset step at time $t = t^*$ and only the non-modified TD-gPC version is being considered.

The goal is to evaluate the inner-product:

$$\langle \psi_i \psi_j, \psi_k \rangle = \int \psi_i(\eta)\psi_j(\eta)\psi_k(\eta) f_\eta(\eta)\, d\eta, \tag{3.55}$$

for some $i, j, k \in \{0, \ldots, M\}$. As described in the previous sections, the integration in (3.55) is transformed to the original random variable ξ via the relation:

$$\langle \psi_i \psi_j, \psi_k \rangle = \int \psi_i(u(t^*;\xi))\psi_j(u(t^*;\xi))\psi_k(u(t^*;\xi)) f_\xi(\xi)\, d\xi, \tag{3.56}$$

whereas u denotes the solution of the underlying system of ordinary differential equations, u being expressed by:

$$u(t;\xi) = \sum_{i=0}^{M} u_i(t)\tilde{\psi}_i(\xi), \tag{3.57}$$

for $t < t^*$, where $\{\tilde{\psi}_i\}_{i=0}^{M}$ shall denote the Chaos Polynomials before the reset. Here, without loss of generality, the same expansion order M is assumed before and after the reset step at time $t = t^*$ (note that unnecessary modes can be set equal to zero). The determination of (3.56) is carried out by utilizing an appropriate Gaussian quadrature rule according to the probability distribution represented by f_ξ of ξ. Therefore, (3.56) is approximated by a finite number N_q of terms:

$$\langle \psi_i \psi_j, \psi_k \rangle \approx \sum_{n=1}^{N_q} w_n (\psi_i \circ u)(\xi_n)(\psi_j \circ u)(\xi_n)(\psi_k \circ u)(\xi_n), \tag{3.58}$$

whereas N_q denotes the number of quadrature points, w_n the corresponding weights and ξ_n the corresponding quadrature points. Since u and ψ_M are both polynomials of degree M, $(\psi_M \circ u)$ is a polynomial of degree M^2. To ensure an exact integration of (3.56), recall that Gaussian quadratures are exact for polynomials of degree $2N_q - 1$ [37], leading to the condition:

$$N_q = \lceil \frac{3M^2 + 1}{2} \rceil, \tag{3.59}$$

whereas $\lceil \cdot \rceil$ denotes the ceiling function. Applying this analysis to further reset times yields:

$$N_q = \lceil \frac{3M^{N_r} + 1}{2} \rceil, \tag{3.60}$$

whereas N_r denotes the current number of the reset step. This, of course, is a worst-case estimate leading to the requirement of a high number N_q of quadrature points

when progressing in time by reseting the basis, even for low expansion orders M. This presents a major drawback of the TD-gPC approach, however, since at every reset step the solution u is transformed to obtain a linear expansion, it is expected that the contribution of higher order modes remains low for a certain time period. Therefore, the high nonlinear orders in (3.56) only have a small contribution to the integral, leading to a much more accurate numerical integration than the estimate suggests. However, this observation relies on numerical results and is certainly application dependent. Therefore, it is still an open question if there exist more accurate numerical integration approaches which remain numerically cost effective.

3.5. Local approach for time-dependent generalized Polynomial Chaos

One major drawback of the (modified) TD-gPC is the fast growing number of modes resulting for even low expansion orders P and Q. Therefore, although the method itself is leading to accurate results, the numerical cost which comes along with solving a coupled system of differential equations for a high number of modes increases significantly, especially when the discretization of the deterministic part of the system is quite expensive. Hence, it is necessary to think about possibilities of reducing the numerical cost and making the computation of the modes feasible. To achieve this goal this section introduces a local approach to TD-gPC, motivated by the "multi–element generalized Polynomial Chaos" introduced by Wan and Karniadakis [44, 45] and based on the published work of the author in [19], resulting in a domain decomposition of the probability space employing time-dependent basis functionals in each element. This leads to the task of solving N independent problems, whereas N denotes the number of elements used, employing a smaller number of basis functionals in each sub-problem compared to solving the global problem.

3.5.1. Domain decomposition

Following the procedure described in [44, 45] the sample space Ω is decomposed implicitly by decomposition of the range of ξ. Note, that here it is not assumed that ξ is scalar valued, instead it holds $\text{range}(\xi) =: B \subset (\mathbb{R} \cup \{-\infty, \infty\})^d$ for some $d \in \mathbb{N}$. Furthermore, ξ denotes the original stochastic input to the system before being transformed via the reset steps in context of TD-gPC. The decomposition of Ω is carried out in the following way:

Let $\{B_j\}_{j=1}^N$ be a disjoint interval decomposition of B, such that:

$$B = \bigcup_{j=1}^{N} B_j, \quad B_{j_1} \cap B_{j_2} = \emptyset \quad \text{for } j_1 \neq j_2, \tag{3.61}$$

$$B_j := [a_1^j, b_1^j) \times [a_2^j, b_2^j) \times \cdots \times [a_d^j, b_d^j]. \tag{3.62}$$

Therefore, B_j defines a multi-dimensional interval of dimension d for every $j = 1, \ldots, N$. Note, that if $\pm\infty \in B_j$ for some j, as this is the case for the Gaussian distribution, usually B_j is decomposed by $B_j = (-\infty, a_*^j) \dot\cup [a_*^j, b_*^j] \dot\cup (b_*^j, \infty)$ and

all refinements are carried out on the middle element $[a_*^j, b_*^j]$. The choice of a_*^j and b_*^j is distribution and application dependent.

To achieve a decomposition of the sample space Ω the indicator function I_j is introduced, which is defined by:

$$I_j = \begin{cases} 1 & \text{if } \xi \in B_j, \\ 0 & \text{otherwise.} \end{cases} \tag{3.63}$$

Hence, a decomposition of Ω is given by $\Omega = \bigcup_{j=1}^{N} I_j^{-1}(1)$, since $I_i^{-1}(1) \cap I_j^{-1}(1) = \emptyset$ for $i \neq j$. Now in each element B_j a local random variable ξ^j is defined via $\xi^j = (\xi_1^j, \xi_2^j, \ldots, \xi_d^j) : I_j^{-1}(1) \mapsto B_j$ for $j = 1, \ldots, N$ subject to the corresponding conditional probability distribution function f_j defined by:

$$f_j(\xi^j | I_j = 1) = \frac{f(\xi^j)}{\mathbb{P}(I_j = 1)}, \tag{3.64}$$

whereas $f(\cdot)$ denotes the probability density function of the globally defined random variable ξ. In practice the random variable ξ_k^j restricted to the element $[a_k^j, b_k^j)$ is rescaled by the transformation:

$$\xi_k^j = \frac{b_k^j - a_k^j}{2} Y_k^j + \frac{b_k^j + a_k^j}{2}, \tag{3.65}$$

subject to a new random variable Y_k^j defined in $(-1, 1)$. Therefore, the probability density function $\bar{f}_j(\cdot)$ of the vector $Y^j = (Y_1^j, \ldots, Y_d^j)$ is given by:

$$\bar{f}_j(y_j) = \det \left| \left(\frac{\partial \xi^j}{\partial y_j} \right)_{j=1}^{N} \right| f_j(\xi^j(y_j) \,|\, I_j = 1) = \frac{f(\xi^j(y_j))}{\mathbb{P}(I_j = 1)} \prod_{k=1}^{d} \frac{b_k^j - a_k^j}{2}. \tag{3.66}$$

3.5.2. Problem structure

The strength in decomposing the probability space lies in the independency of the resulting local problems [44, 45]. Therefore, after decomposing the probability space into N elements, N independent problems on the corresponding probability spaces $(I_j^{-1}(1), \mathcal{F} \cap I_j^{-1}(1), \mathbb{P}(\cdot | I_j^{-1}(1))$ for $j = 1, \ldots, N$ have to be solved. Now the time-dependent approach can be applied in each single element, i.e., given a solution $u^{(j)} = u^{(j)}(\xi^j)$ in each element j, a new random variable η_k^j is introduced by:

$$\eta_k^j := \sum_{i=0}^{M} u_i^{(j)}(t_k) \psi_i(\eta_{k-1}^j, \xi^j), \tag{3.67}$$

for a reset time step $t = t_k$, whereas $\eta_0^j := \xi^j$. The solution $u^{(j)}$ is then expressed in terms of η_k^j and ξ^j as described in Section 3.3 and the procedure is repeated at every time step qualifying for a reset, e.g., every time step or a time step defined by some error estimation criteria.

3.5.3. Calculation of the stochastic moments

Since a change of variables is performed in each element, the stochastic moments such as the mean and the variance need to be calculated independently in each element and combined afterwards. Therefore, first the mean and the variance have to be calculated locally according to Section 3.3, denoted by $\bar{u}^{(j)}$ and $\sigma^2(u^{(j)})$, $j = 1, \ldots, N$. Next, according to Bayes's theorem and the law of total probability [45], the global stochastic moments of order m, denoted by μ_m, can be calculated via:

$$\mu_m(u)(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\mu_m(u^{(j)})(t). \tag{3.68}$$

Hence, the global mean is approximated by the weighted sum of the local mean values, i.e.,

$$\bar{u}(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\bar{u}^{(j)}(t). \tag{3.69}$$

For the variance it holds:

$$\begin{aligned}
\sigma^2(u)(t) &= \mu_2(u)(t) - \mu_1(u)(t)^2 \\
&\approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\mu_2(u^{(j)})(t) \\
&\quad - \sum_{j=1}^{N}\sum_{k=1}^{N} \mathbb{P}(I_j = 1)\mathbb{P}(I_k = 1)\mu_1(u^{(j)})(t)\mu_1(u^{(k)})(t) \\
&= \sum_{j=1}^{N} \mathbb{P}(I_j = 1) \left(\sigma^2(u^{(j)})(t) + (\bar{u}^{(j)})^2 - \bar{u}^{(j)}(t) \sum_{k=1}^{N} \mathbb{P}(I_k = 1)\bar{u}^{(k)}(t) \right) \\
&= \sum_{j=1}^{N} \mathbb{P}(I_j = 1) \left(\sigma^2(u^{(j)})(t) + \bar{u}^{(j)}(t)(\bar{u}^{(j)}(t) - \bar{u}(t)) \right).
\end{aligned} \tag{3.70}$$

3.5.4. Local TD-gPC algorithm

If no adaptive refinement of the probability space with respect to the number of elements N is employed, the numerical implementation of the local TD-gPC is carried out straight forward if a global TD-gPC solver is available. The numerical cost involved is due to the orthogonalization of the basis functionals in each element for every reset step. The number of reset steps can be reduced when employing suitable reset criteria, e.g., the observation of the magnitudes of modes representing nonlinear dependencies as introduced in [14] can lead to a significant reduction of the numerical cost. However, defining a reset criteria introduces an extra source of errors, therefore, here it is chosen to apply TD-gPC in every time step. A summary of the local TD-gPC algorithm is given in Algorithm 3.1.

Algorithm 3.1 Local TD-gPC for calculating the mean and the variance

1: Choose the number of elements N and discretization orders P and Q
2: Choose an appropriate time discretization method and let N_T denote the number of time steps
3: **for** $j = 1 \rightarrow N$ **do**
4: Construct the local conditional random variable $\eta_0^{(j)} := \xi^{(j)}$ w.r.t. the initial stochastic input
5: Transform $\eta_0^{(j)}$ to a random variable $Y^{(j)}$ defined on $(-1, 1)$ via (3.65) and set $\eta_0^{(j)} = Y^{(j)}$
6: **for** $i = 1 \rightarrow N_T$ **do**
7: Construct a new random variable $\eta_i^{(j)}$ according to TD-gPC via $\eta_{i-1}^{(j)}$, $\xi^{(j)}$ and the local solution $u^{(j)}$
8: Construct a new set of orthogonal basis functionals $\{\psi_i^{(j)}\}$ depending on $\eta_i^{(j)}$ and ξ^j w.r.t. P and Q
9: Generate new local initial conditions according to TD-gPC
10: **end for**
11: Store the calculated local mean $\bar{u}^{(j)}$ and variance $\sigma^2(u^{(j)})$
12: **end for**
13: Post processing of the stored local quantities

3.6. Numerical results for the local TD-gPC

In this section the effect of employing the local TD-gPC is demonstrated in context of a simple one-dimensional ordinary differential equation, representing the class of long term integration related problems, and the more challenging Kraichnan-Orszag three mode problem [31], which is representing the class of stochastic discontinuities. The focus lies on the illustration of the convergence properties and the analysis of the trade off between solving N independent local problems and employing an $(M + 1)$-dimensional TD-gPC in each element.

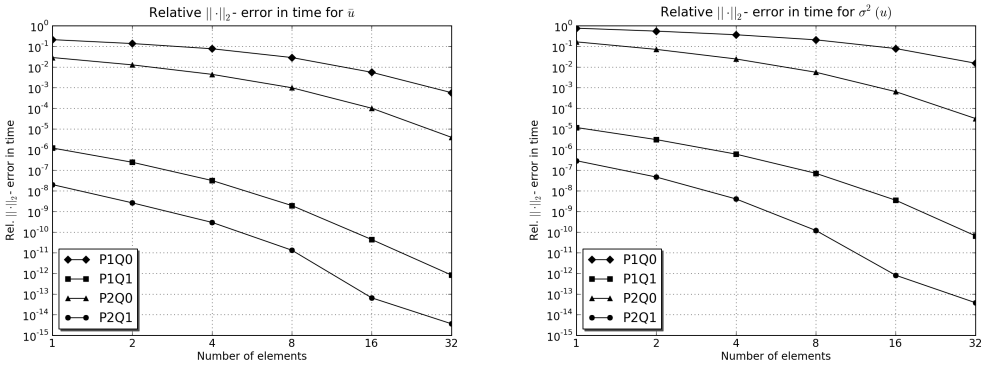
3.6.1. A simple one-dimensional ODE

This problem has been studied in various work, e.g., [14, 44], and also in Section 3.1 having the advantage that its simplicity allows to calculate an exact solution analytically. The governing equations are given by:

$$\frac{du}{dt} = -k(\xi)u, \tag{3.71}$$

$$u(0) = 1, \tag{3.72}$$

subject to a uniformly distributed random variable $\xi \sim U(-1, 1)$, where it is further assumed that $k(\xi) = \frac{1}{2}(1 + \xi)$. Therefore, the exact solution, its mean and variance



(a) Mean (b) Variance

Figure 3.5.: Relative $\|\cdot\|_2$-error in time with respect to various orders of Chaos expansions with varying number of elements.

are given by:

$$u(t;\xi) = \exp(-k(\xi)t), \tag{3.73}$$

$$\bar{u}(t) = \frac{1-\exp(-t)}{t}, \tag{3.74}$$

$$\sigma^2(u)(t) = \frac{1}{2t}(1-\exp(-2t)) - \left(\frac{1-\exp(-t)}{t}\right)^2, \tag{3.75}$$

respectively. This clearly shows the increasing nonlinear dependency of u on ξ for increasing time t due to the exponential type of the solution, which leads to a poor convergence property when employing the standard gPC (cf. Section 3.1). Here, the focus is restricted on the convergence behavior of the local time-dependent gPC. For the numerical computation a Runge-Kutta scheme of 4^{th} order is employed with a homogeneous time step size $\Delta t = 0.001$ to minimize the error contributions arising from the time discretization. For numerical integration a Gauss-Legendre quadrature rule with 100 quadrature points is employed. Furthermore, the probability space is decomposed employing equidistant elements.

The absolute errors are measured in the discrete euclidean norm $\|\cdot\|_2$, i.e.,

$$\|\bar{u} - \bar{u}_{exact}\|_2 = \sqrt{\left(\sum_n \left(\bar{u}(t_n) - \bar{u}_{exact}(t_n)\right)^2\right)}, \tag{3.76}$$

$$\|\sigma^2(u) - \sigma^2(u_{exact})\|_2 = \sqrt{\left(\sum_n \left(\sigma^2(u)(t_n) - \sigma^2(u_{exact})(t_n)\right)^2\right)}, \tag{3.77}$$

for all discrete time steps $t_n = n\Delta t$ within the simulation interval $[0, 100]$. For computation of the associated relative errors, the absolute errors are divided by the euclidean norm of the exact mean or variance, respectively. Figure 3.5 depicts the relative error evolution for this problem. It clearly displays an exponential convergence behavior with increasing convergence rate when refining the elements,

which is in good agreement with the results shown for the multi-element gPC in [44]. Therefore, if high accuracy is desired there is the possibility of choosing between a specific high order P and Q or a high number of elements to be used. This is important, since due to the independence of the local problems it is possible to obtain a trivial parallelization when computing the results in parallel, which leads to a very efficient solver. Hence, a small expansion order, e.g., $P = 1$ and $Q = 1$, which equals a total number of modes $M+1 = 4$, is already sufficient to achieve high accuracies with respect to the whole simulation time interval $[0, 100]$. Of course, this model problem is small with respect to its dimension, but it serves the purpose of demonstrating the fast convergence property of the local time-dependent approach to the exact solution.

3.6.2. The Kraichnan-Orszag three mode problem

Problem definition

The Kraichnan–Orszag three mode problem [31] is known to fail in a short time when employing gPC due to a discontinuous dependency on the initial conditions. It therefore represents a challenging benchmark problem, which has been studied in various contexts, such as adaptive multi-element gPC in [44, 45] and TD-gPC in [14]. It is a nonlinear three-dimensional system of ordinary differential equations given by:

$$\frac{dx_1}{dt} = x_2 x_3, \tag{3.78}$$

$$\frac{dx_2}{dt} = x_3 x_1, \tag{3.79}$$

$$\frac{dx_3}{dt} = -2 x_1 x_2, \tag{3.80}$$

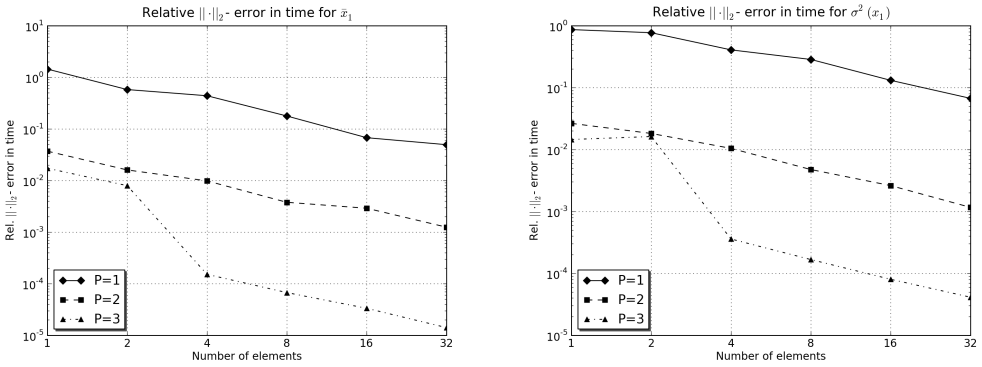
with $x_1(t = 0) = \alpha + 0.01\xi$, $x_2(t = 0) = 1.0$ and $x_3(t = 0) = 1.0$, subject to a uniformly distributed random variable $\xi \sim U(-1, 1)$. It is known [14, 44] that the critical range of α for which there is a strong dependency on the initial conditions is given by $(0.9, 1)$. Therefore, the following analysis is focused on $\alpha = 0.995$.

Numerical results

When employing TD-gPC in each element, the discretization parameter Q is set to $Q = 0$, since no direct stochastic input is given within the system of differential equations. Still, the size of the system is quite large and given by:

$$M + 1 = \frac{(P + 3)!}{P!3!} = \frac{(P + 1)(P + 2)(P + 3)}{6} = O(P^3). \tag{3.81}$$

This illustrates the necessity of keeping the order P of the expansion small, such that the resulting number of modes $M+1$ remains low. For the numerical computations a Runge-Kutta solver of 4^{th} order is used to solve the deterministic part with a homogeneous time step size $\Delta t = 0.001$. Furthermore, an equidistant refinement of the interval $(-1, 1)$ is employed resulting in $N = 2^i$ elements for each refinement level i. Relative errors are measured in the L^2-norm defined in (3.76),(3.77).



(a) Mean (b) Variance

Figure 3.6.: Relative $\|\cdot\|_2$-error in time with respect to various orders of Chaos expansions with varying number of elements.

For numerical integration a Gauss-Legendre quadrature rule with 100 quadrature points is employed. Since there is no analytical solution available for this problem, the results are compared to a discretization employing $N = 64$ elements and an expansion order $P = 3$.

Fig. 3.6 shows exponential convergence behavior in P as well as in N (with an exception for $N \leq 2$) for both the mean and the variance of the first component x_1 of the solution. This is in good agreement with the results obtained for the one-dimensional problem given in Section 3.6.1. From this it follows that when utilizing parallel computation of the sub-problems on each element (which is possible in a trivial way due to the independence of the sub-problems) it is possible to achieve same accuracies by either refining P or N, e.g., $P1N32$ is almost as accurate as $P3N1$ but only requires $M+1 = 4$ modes for the $P = 1$ case instead of $M+1 = 20$ modes for the $P = 3$ case. Since the size $M + 1$ is depending on the size of the corresponding deterministic system (here this equals 3), the local time-dependent method is expected to increase its efficiency in reducing the numerical cost in trade off to parallel computation of the sub-problems even further for larger systems of differential equations.

Chapter 4.

Spectral-Stochastic-Finite-Element-Method for the unsteady stochastic incompressible Navier-Stokes equations

The previous chapter is focused on the generalized Polynomial Chaos expansion in context of systems of ordinary differential equations subject to random parameters and initial conditions. The second part of this work, starting with this chapter, is devoted to the application of the gPC to systems of partial differential equations given by the unsteady stochastic incompressible Navier-Stokes equations subject to uncertainties introduced in initial and boundary conditions or right hand sides. Galerkin projected systems arising from a gPC expansion of the stochastic Navier-Stokes equations have been analyzed in various works, see for example [1, 24, 25, 27, 43, 48]. This chapter will introduce the governing equations and demonstrate how the probability and deterministic spaces can be discretized appropriately. Therefore, first the discretization is outlined for the deterministic equations and afterwards transfered to the stochastic case. Furthermore, numerically stable Finite-Element spaces are derived for the stochastic Galerkin projected system.

4.1. The unsteady deterministic incompressible Navier-Stokes equations

Many physical systems describing a fluid flow in some domain $\mathcal{D} \times [0,T] \subset \mathbb{R}^d \times \mathbb{R}$, $d = 2, 3$ can be modeled by the unsteady incompressible Navier–Stokes equations [36], whereas the following, so-called "primitive" variables play an important role:

- Density $\rho : \mathcal{D} \times [0,T] \to \mathbb{R}^+$,
- Velocity $u : \mathcal{D} \times [0,T] \to \mathbb{R}^d$,
- Pressure $p : \mathcal{D} \times [0,T] \to \mathbb{R}$.

Since, an incompressible flow shall be considered, the density ρ is assumed to be constant, i.e., $\rho \equiv \rho_0 \in \mathbb{R}^+$. Furthermore, only Newtonian fluids will be considered, resulting in a linear stress tensor model introducing the scalar kinematic viscosity $\nu \in \mathbb{R}^+$. Therefore, for an open domain $\mathcal{D}$ with Lipschitz-continuous boundary

$\Gamma = \partial\mathcal{D}$ the unsteady incompressible Navier-Stokes equations (NSE) read:

$$\partial_t u + (u \cdot \nabla)u - \nu\Delta u + \frac{1}{\rho}\nabla p = f, \qquad \text{in } \mathcal{D} \times (0,T), \tag{4.1}$$

$$\nabla \cdot u = 0, \qquad \text{in } \mathcal{D} \times (0,T), \tag{4.2}$$

$$u = u^D, \quad \text{on } \Gamma \times [0,T], \tag{4.3}$$

$$u(t=0) = u_I, \qquad \text{in } \mathcal{D}, \tag{4.4}$$

for u^D defined on $\Gamma \times [0,T]$, f defined on $\mathcal{D} \times (0,T)$ and u_I defined on $\mathcal{D}$. Note, that Γ denotes the so-called "Dirichlet" boundary, since the constraint is directly prescribed for the velocity values. The following definition is taken from [40]:

Definition 4.1. A solution (u,p) is called a classical solution to the Navier-Stokes equations, if it satisfies (4.1)–(4.4) and the regularity conditions:

$$u \in [C^2(\overline{\mathcal{D} \times (0,T)})]^d, \tag{4.5}$$

$$p \in C^1(\overline{\mathcal{D} \times (0,T)}). \tag{4.6}$$

Usually one doesn't seek for a classical solution to the Navier-Stokes equations due to the high regularity requirements on (u,p). A common way to reduce those is by introducing the concept of a weak solution. Therefore, the equations (4.1) and (4.2) are expressed in a variational formulation. To this end the following functional spaces are defined [12, 13, 35, 40]:

- $Q := \{p \in L^2(\mathcal{D}) : \int_{\mathcal{D}} p \, dx = 0\}$,
- $V := [H_0^1(\mathcal{D})]^d$,
- $V_{div} := \{v \in V : \ \nabla \cdot v = 0, \text{weak sense}\}$,
- $H_{div} := \{v \in [L^2(\mathcal{D})]^d : \ \nabla \cdot v = 0 \text{ in } \mathcal{D}, \ v \cdot \vec{n} = 0 \text{ on } \Gamma, \ \text{trace sense}\}$,

whereas $\vec{n}$ denotes the unit outward normal vector on Γ.

Before stating the variational (weak) formulation of the NSE consider the following calculation:

For every $v \in V$ it holds via integration by parts:

$$\int_{\mathcal{D}} \Delta u \cdot v \, dx = -\int_{\mathcal{D}} \nabla u : \nabla v \, dx + \underbrace{\int_{\partial\mathcal{D}} v \cdot \nabla u \cdot \vec{n} \, d\Gamma}_{=0, \text{ since } v \in V} = -\int_{\mathcal{D}} \nabla u : \nabla v \, dx, \tag{4.7}$$

$$\int_{\mathcal{D}} \nabla p \cdot v \, dx = -\int_{\mathcal{D}} p \nabla \cdot v \, dx + \underbrace{\int_{\partial\mathcal{D}} pv \cdot \vec{n} \, d\Gamma}_{=0, \text{ since } v \in V} = -\int_{\mathcal{D}} p \nabla \cdot v \, dx. \tag{4.8}$$

Therefore, the variational (weak) formulation of the NSE is given by [32, 40]:

For given $f \in L^2(0,T; H_{div})$ and $u_I \in H_{div}$, find $u(t) \in u^D + V$ and $p(t) \in Q$ such that for almost every $t \in (0,T)$

$$(\partial_t u, v) + ((u \cdot \nabla)u, v) + \nu(\nabla u, \nabla v) - (p, \nabla \cdot v) = (f, v), \quad \forall v \in V, \tag{4.9}$$

$$(\nabla \cdot u, q) = 0, \qquad \forall q \in Q, \tag{4.10}$$

$$u(t=0) = u_I, \qquad \text{in } \mathcal{D}, \tag{4.11}$$

whereas $(\cdot,\cdot)$ denotes the inner-product on $L^2(\mathcal{D})$.

Definition 4.2. A solution (u, p) is called a weak solution to the Navier-Stokes equations if it satisfies (4.9)–(4.11) and the regularity conditions:

$$u(t) \in u^D + V, \tag{4.12}$$

$$p(t) \in Q, \tag{4.13}$$

for almost every $t \in (0, T)$.

It can also be shown that u is almost everywhere equal to a continuous function, so that (4.4) remains meaningful [40]. Note that the functions v, q arising in the variational formulation are referred to as "test functions" and u, p as "trial functions". An alternative weak formulation of the NSE reads [32]:

Given $f \in L^2(0, T; H_{div})$ and $u_I \in H_{div}$, find $u \in L^2(0, T; V_{div}) \cap L^\infty(0, T; H_{div})$ such that

$$(\partial_t u, v) + ((u \cdot \nabla)u, v) + \nu(\nabla u, \nabla v) = (f, v), \quad \forall v \in V, \tag{4.14}$$

$$u(t = 0) = u_I, \qquad \text{in } \mathcal{D}, \tag{4.15}$$

for homogeneous Dirichlet boundary conditions, i.e., $u^D \equiv 0$. For this formulation the existence and uniqueness of a solution satisfying (4.14) and (4.15) in the two dimensional case has been proven [32, 40]. However, for the three dimensional case, it is still an open question whether the solution is unique.

Any solution of (4.9)–(4.10) is also a solution to (4.14), (4.15). The converse is true provided the solution fulfills specific regularity requirements given in [32].

For discretizations employing Finite-Elements, as will be discussed in the following section, it is numerically more feasible to use the weak formulation involving the pressure variable, the so-called "mixed-type formulation", since defining an appropriate discretization space, which ensures the divergence-free condition can become quite complicated.

4.2. Space and time discretization of the deterministic equations

Solving the weak formulation of the NSE (4.9)–(4.11) in practical computations requires an appropriate discretization employing a finite number of unknowns both in the space and time variable. Here, the so-called Rothe method [7] is chosen, referring to a Finite–Difference approximation of the time variable and a Finite–Element approximation of the space variable for the time discretized system, which will be explained in the following subsections. Furthermore, it is notationally convenient for the numerical analysis to consider the following notation of the NSE:

$$(\partial_t u, v) + n(u, u, v) + a(u, v) + b(p, v) = (f, v), \tag{4.16}$$

$$b(q, u) = 0, \tag{4.17}$$

with the bilinear forms a and b and the trilinear form n defined by

$$n(u, \bar{u}, v) := ((u \cdot \nabla)\bar{u}, v), \tag{4.18}$$

$$a(u, v) := \nu(\nabla u, \nabla v), \tag{4.19}$$

$$b(p, v) := -(p, \nabla \cdot v), \tag{4.20}$$

for $u, \bar{u}, v \in H^1(\mathcal{D}),\ p, q \in Q$. Note, that this variational formulation reflects a saddle-point structure, which plays an important role for numerical stability of the corresponding discretized systems. This issue will be addressed later within this section.

4.2.1. Time discrectization by a Crank-Nicolson scheme

The continuous system (4.16) and (4.17) can be time discretized by various discretization methods. Here and for the rest of this work, the focus is on the so-called Crank-Nicolson scheme [5] belonging to the class of Finite-Difference approximations. Let a partition of the time interval $[0, T]$ consisting of N_T elements be given by:

$$[0, T] = \bigcup_{n=0}^{N_T - 1} [t_n, t_{n+1}], \tag{4.21}$$

$$(t_i, t_{i+1}) \cap (t_j, t_{j+1}) = \emptyset, \quad i \neq j. \tag{4.22}$$

A homogeneous discretization can be parameterized by a step length $\Delta t > 0$, resulting in $t_n := n\Delta t$ for $n = 0, \ldots, N_T$. Otherwise, for given time steps t_n the time step length Δt_n is defined by $\Delta t_n := t_n - t_{n-1}$. Therefore, the Crank-Nicolson scheme for the unsteady incompressible Navier-Stokes equations reads:

$$\left(\frac{u^n - u^{n-1}}{\Delta t_n}, v\right) + \frac{1}{2}(n(u^n, u^n, v) + n(u^{n-1}, u^{n-1}, v)) \tag{4.23}$$
$$+ \frac{1}{2}(a(u^n, v) + a(u^{n-1}, v)) + b(p^n, v) = \frac{1}{2}((f^n, v) + (f^{n-1}, v)),$$

$$b(q, u^n) = 0, \tag{4.24}$$

for all $v \in V$ and $q \in Q$, whereas $u^n := u(t = t_n) \in u^D + V,\ p^n := p(t = t_n) \in Q,\ f^n := f(t = t_n) \in L^2(H_{div})$ for $n = 0, \ldots, N_T$. Note that the treatment of the pressure term in (4.23) and the divergence–free condition (4.24) is different from the time discretization of the velocity variable. This is due to the fact that the divergence–free condition (4.24) is required to hold for every time step, which influence on (4.23) is given by the pressure variable allowing for an implicit enforcement of (4.24) at every time step. It has been shown, that the Crank-Nicolson method is A-stable and of second order in time [41, 42], however it also belongs to the class of implicit time discretizations and therefore requires the solution of a PDE system in the space variable for each time step.

4.2.2. Linearization by Newton's method

The time-discretized system (4.23), (4.24) requires an appropriate space discretization, such that the resulting system becomes finite dimensional. However, before introducing adequate discretized function spaces the nonlinearity of (4.23) arising in the trilinear form n is being linearized. This comes into play when the nonlinear system (4.23), (4.24) is solved by Newton's method [6], which is an iterative method requiring the solution of the linearized weak formulation of the NSE in every iteration.

To simplify the notation for the derivation of the linearized system, (4.23) and (4.24) are denoted by:

$$F(u^n, p^n, v, q) = [g^{n-1}, 0]^t, \tag{4.25}$$

whereas F and g^{n-1} are defined by:

$$F(u, p, v, q) := [F_1(u, p, v), F_2(u, q)]^t, \tag{4.26}$$

$$F_1(u, p, v) := (u, v) + \frac{\Delta t_n}{2} n(u, u, v) + \frac{\Delta t_n}{2} a(u, v) + \Delta t_n b(p, v) - \frac{\Delta t_n}{2}(f^n, v), \tag{4.27}$$

$$F_2(u, q) := b(q, u), \tag{4.28}$$

$$g^{n-1} := (u^{n-1}, v) - \frac{\Delta t_n}{2} n(u^{n-1}, u^{n-1}, v) - \frac{\Delta t_n}{2} a(u^{n-1}, v) + \frac{\Delta t_n}{2}(f^{n-1}, v). \tag{4.29}$$

Note, that $F : ((u^D + V) \times Q \times V \times Q) \longrightarrow \mathbb{R}^2$ and $g^{n-1} \in \mathbb{R}$. The linearization of (4.25) requires the computation of the Gâteaux directional derivative of F w.r.t. u and p in a direction denoted by $\bar{u} \in u^D + V$ and $\bar{p} \in Q$ respectively. The Gâteaux directional derivative (if it exists) denoted by $J(u, p, v, q) : (u^D + V) \times Q \longrightarrow \mathbb{R}^2$ is defined by [49]:

$$J(u, p, v, q)[\bar{u}, \bar{p}] := \lim_{h \to 0} \frac{F(u + h\bar{u}, p + h\bar{p}, v, q) - F(u, p, v, q)}{h}. \tag{4.30}$$

An easy calculation shows that the Gâteaux directional derivative exists and is given by:

$$J(u, p, v, q)[\bar{u}, \bar{p}] = \begin{bmatrix} (\bar{u}, v) + \frac{\Delta t_n}{2}(n(u, \bar{u}, v) + n(\bar{u}, u, v)) + \frac{\Delta t_n}{2} a(\bar{u}, v) + \Delta t_n b(\bar{p}, v) \\ b(q, \bar{u}) \end{bmatrix}, \tag{4.31}$$

which is representing the left hand side of a linear system of PDEs with respect to $\bar{u}$ and $\bar{p}$.

As being mentioned in the beginning of this subsection, Newton's method shall be applied to solve the time-discretized system (4.25) for every time step. In this context, Newton's method starts with an initial guess $[u_0^n, p_0^n]$ and iteratively computes the following iterates via the relation:

$$[u_{k+1}^n, p_{k+1}^n] = [u_k^n + \bar{u}_k^n, p_k^n + \bar{p}_k^n], \tag{4.32}$$

$$-J(u_k^n, p_k^n, v, q)[\bar{u}_k^n, \bar{p}_k^n] = F(u_k^n, p_k^n, v, q) - [g^{n-1}, 0]^t \quad \forall v \in V,\ q \in Q, \tag{4.33}$$

whereas k denotes the iteration index. It is well-known that if the initial guess is "close enough" to a solution, Newton's method is of quadratic convergence order [6]. However, ensuring that the initial guess is "close enough" to a solution can become a complicated task, which might not be solvable at all in practice. In those cases one often relies on modifications of the Newton method, such as so-called "Inexact Newton" [6, 10] and "Quasi Newton" [6, 23] approaches. One natural, not

necessary optimal, choice of the initial guess is given by the solution of the previous time step, i.e., $[u_0^n, p_0^n] = [u^{n-1}, p^{n-1}]$.

The so-called Newton step (4.33) requires the solution of a linearized system of PDEs for the velocity and pressure updates $\bar{u}_k^n$, $\bar{p}_k^n$. Since this system is still infinite dimensional, an appropriate space discretization needs to be employed, which is described in the following subsection.

4.2.3. Space discretization by the Finite-Element method

The starting point for employing the Finite-Element method [2, 3] is the system of PDEs given by the Newton step (4.33). Since the application of the Finite-Element method shall be independent of the time step t^n, it suffices to consider one specific Newton iteration with index k in (4.33) for which the following simplified notation is employed:

Seek $u \in u^D + V$ and $p \in Q$ such that

$$-J_k(v,q)[u,p] = F_k(v,q), \quad \forall v \in V,\ q \in Q, \tag{4.34}$$

whereas

$$J_k(v,q)[u,p] := J(u_k^n, p_k^n, v, q)[u,p], \tag{4.35}$$

$$F_k(v,q) := F(u_k^n, p_k^n, v, q) - [g^{n-1}, 0]^t, \tag{4.36}$$

for a given Newton iterate u_k^n and p_k^n and a right hand side g^{n-1} resulting from the previous time step.

The task is to construct finite dimensional vector spaces $V_h \subset V$ and $Q_h \subset Q$ for the trial and finite dimensional vector spaces $W_h \subset V$ and $R_h \subset Q$ for the test functions. Note, that the requirement that the discretized spaces are subspaces of the continuous spaces represent the so-called "conforming" method. It is possible to violate this condition, e.g., by allowing $V_h \not\subset V$, leading to so-called "non-conforming" methods [2, 3]. However, the focus of this work is restricted on conforming approaches. Furthermore, by substitution of the continuous spaces V and Q by their corresponding discrete subspaces V_h, Q_h, W_h and R_h in (4.34), the solution is projected via the L^2-inner-product onto the discrete spaces of the test functions W_h and R_h, which in literature is referred to as being a Galerkin or Petrov-Galerkin projection [2, 3], depending if the discrete spaces for the solution variables and test functions do coincide or not. However, in this work only Galerkin projections are being considered leading to the requirement $V_h = W_h$ and $Q_h = R_h$. With these restrictions the discrete problem reads:

Seek $u_h \in u_h^D + V_h$ and $p_h \in Q_h$ such that

$$-J_k(v_h,q_h)[u_h,p_h] = F_k(v_h,q_h), \quad \forall v_h \in V_h,\ q_h \in Q_h, \tag{4.37}$$

whereas u_h^D is a L^2-projection of u^D onto V_h.

Choosing appropriate discretization spaces V_h and Q_h is a non-trivial task. Since (4.37) represents a mixed-type approach [2, 3, 17], i.e., the velocity and pressure variables are sought in different vector spaces, special care needs to be considered to ensure numerical stability of the discretization. In fluid flow problems it is

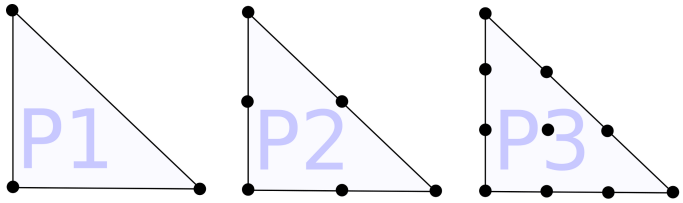


Figure 4.1.: Degrees of freedom of the Taylor-Hood elements defined on triangles for polynomial degrees $P1$ (linear), $P2$ (quadratic) and $P3$ (cubic).

known that the critical condition for numerical stability is the requirement that V_h and Q_h fulfill the so-called "Ladyshenskaja-Babuška-Brezzi" condition (LBB) [2], since for the linearized system uniform ellipticity can be proved based on Gårding's inequality provided that the time step sizes Δt_n are sufficiently small [3]. The LBB condition is given by:

$$\exists \beta > 0 : \sup_{v_h \in V_h \backslash \{0\}} \frac{b(q_h, v_h)}{\|v_h\|} \geq \beta \|q_h\|, \quad \forall q_h \in Q_h \backslash \{0\}. \tag{4.38}$$

One well established choice of V_h and Q_h satisfying the LBB condition (4.38) is given by the so-called "Taylor-Hood" elements [2]. For simplicity it is assumed that the spacial domain $\mathcal{D}$ is polygonal such that there exists a triangulation $\mathcal{T}_h$ of $\overline{\mathcal{D}}$ employing triangles, rectangles, tetrahedrons or hexahedrons, depending on the space dimension d of $\mathcal{D}$. Then the Taylor-Hood spaces of degree $n \in \mathbb{N}$ are defined by [2, 17, 39]:

$$V_h := \{v_h \in C(\overline{\mathcal{D}})^d \cap H_0^1(\mathcal{D})^d; v_{h|T} \in \mathcal{P}_n^d(T), \quad \forall T \in \mathcal{T}_h\}, \tag{4.39}$$

$$Q_h := \{q_h \in C(\mathcal{D}) \cap L^2(\mathcal{D}); q_{h|T} \in \mathcal{P}_{n-1}(T), \quad \forall T \in \mathcal{T}_h\}, \tag{4.40}$$

whereas

$$\mathcal{P}_n(T) := \{u \in C(T) : u(x_1, x_2) = \sum_{i+j \leq n, i,j \geq 0} c_{ij} x_1^i x_2^j, \quad T = \text{triangle}\}, \tag{4.41}$$

$$\mathcal{P}_n(T) := \{u \in C(T) : u(x_1, x_2) = \sum_{i,j \leq n, i,j \geq 0} c_{ij} x_1^i x_2^j, \quad T = \text{rectangle}\}, \tag{4.42}$$

denote the sets of polynomials of degree n defined on the domain $T \in \mathcal{T}_h$ with analog definitions for tetrahedrons and hexahedrons. Therefore, V_h and Q_h are finite dimensional vector spaces, whereas each can be represented by a set of basis functions being piecewise polynomials themselves, notated by:

$$V_h = \text{span}\{\phi_1, \ldots, \phi_N\}, \tag{4.43}$$

$$Q_h = \text{span}\{\chi_1, \ldots, \chi_M\}. \tag{4.44}$$

Here, the space dimensions N and M are defined by: $N := \dim\{V_h\}$ and $M := \dim\{Q_h\}$ such that $v_h(x) = \sum_{i=0}^N v_h^i \phi_i(x)$ and $q_h(x) = \sum_{i=0}^M q_h^i \chi_i(x)$ for all $v_h \in V_h$, $q_h \in Q_h$ and $x \in \mathcal{D}$ with some coefficients $v_h^i \in \mathbb{R}$ for $i = 1, \ldots, N$ and $q_h^i \in \mathbb{R}$ for $i = 1, \ldots, M$. Fig. 4.1 depicts the corresponding degrees of freedom for the Taylor-Hood elements defined on a triangle for varying polynomial degrees. Applying these

discretizations within the Newton step (4.37) one has to solve a linear system of equations for the unknowns $[\vec{u}_h, \vec{p}_h]^t$ in each Newton iteration k, given by:

$$\begin{bmatrix} A & B \\ B^t & 0 \end{bmatrix} \begin{bmatrix} \vec{u}_h \\ \vec{p}_h \end{bmatrix} = \begin{bmatrix} \vec{F}_1 \\ \vec{F}_2 \end{bmatrix}, \tag{4.45}$$

whereas $\vec{u}_h = [u_h^1, \ldots, u_h^N]^t \in \mathbb{R}^N$, $\vec{p}_h = [p_h^1, \ldots, p_h^M]^t \in \mathbb{R}^M$, $A \in \mathbb{R}^{N,N}$, $B \in \mathbb{R}^{N,M}$, $\vec{F}_1 \in \mathbb{R}^N$, and $\vec{F}_2 \in \mathbb{R}^M$, which structure does reflect the saddle-point nature of the problem. Here, the matrices $A = (a_{ij})_{i,j=1,\ldots,N}$ and $B = (b_{ij})_{i=1,\ldots,N,j=1,\ldots,M}$ and the right hand sides $\vec{F}_1 = (F_1^i)_{i=1,\ldots,N}$ and $\vec{F}_2 = (F_2^i)_{i=1,\ldots,M}$ are defined by:

$$a_{ij} = (\phi_j, \phi_i) + \frac{\Delta t_n}{2}(n(u_k^n, \phi_j, \phi_i) + n(\phi_j, u_k^n, \phi_i)) + \frac{\Delta t_n}{2} a(\phi_j, \phi_i), \tag{4.46}$$

$$b_{ij} = \Delta t_n b(\chi_j, \phi_i), \tag{4.47}$$

$$F_1^i = F_1(u_k^n, p_k^n, \phi_i) - g^{n-1}, \tag{4.48}$$

$$F_2^i = \Delta t_n F_2(u_k^n, \chi_i), \tag{4.49}$$

within the Newton iteration k.

4.3. The unsteady stochastic incompressible Navier-Stokes equations

Based on the approaches outlined for the deterministic case in the previous section, the mathematical framework is extended towards the stochastic context in the following, whereas the uncertainties are introduced via boundary conditions, initial conditions or right hand sides. The consideration of an uncertain viscosity term is not within the focus of this section, however, its treatment can be carried out in an analog way.

4.3.1. Strong and weak formulation

It is assumed that there exists a stochastic model, which appropriately describes the underlying uncertainties within the considered system. Furthermore, it is assumed that the uncertainties can be parameterized via some random vector $\xi = (\xi_1, \ldots, \xi_L) \in \mathbb{R}^L$ of dimension $L \in \mathbb{N}$, whereas ξ_i, $i = 1, \ldots, L$, are uncorrelated real-valued random variables, ξ belonging to some underlying probability space $(\Omega, \mathcal{F}, \mathbb{P})$, whereas Ω is a set of samples, $\mathcal{F} \subset 2^\Omega$ a filtration and $\mathbb{P}$ some probability measure (cf. Chapter 2). Since the uncertainties shall be introduced via boundary conditions, initial conditions or right hand sides, the parameterized unsteady stochastic incompressible Navier-Stokes equations (SNSE) read [25]:

$$\partial_t u(\xi) + (u(\xi) \cdot \nabla) u(\xi) - \nu \Delta u(\xi) + \frac{1}{\rho} \nabla p(\xi) = f, \qquad \text{in } \mathcal{D} \times (0, T), \tag{4.50}$$

$$\nabla \cdot u(\xi) = 0, \qquad \text{in } \mathcal{D} \times (0, T), \tag{4.51}$$

$$u(\xi) = u^D(\xi), \quad \text{on } \Gamma \times (0, T), \tag{4.52}$$

$$u(t = 0; \xi) = u_I(\xi), \qquad \text{in } \mathcal{D}, \tag{4.53}$$

whereas the equations (4.50)–(4.53) are required to hold almost surely in Ω. Note, that in contrast to the deterministic case, $u = u(x, t; \xi)$ is a random field itself due to the implicit dependency on the random vector ξ through the partial differential equation. Adapting the definitions given for the deterministic case pointwise in the probability space motivates the following definition for classical solutions to the SNSE:

Definition 4.3. A solution (u, p) is called a classical solution to the stochastic Navier-Stokes equations, if it is satisfies (4.50)–(4.53) almost surely in Ω and

$$u \in [L^2(\Omega)]^d \otimes [C^2(\overline{\mathcal{D} \times (0,T)})]^d, \tag{4.54}$$

$$p \in L^2(\Omega) \otimes C^1(\overline{\mathcal{D} \times (0,T)}). \tag{4.55}$$

As being the case in the deterministic framework, usually one seeks a solution which satisfies a weak formulation due to the high regularity requirements for a classical solution. The derivation of such a weak formulation can be approached by different strategies. Here, first the underlying probability space is being discretized, which will result in a deterministic system of PDEs for which a similar treatment as in the previous section will be employed. An alternative approach is given by defining test and trial functions on the tensor space of $L^2(\Omega)$ and the corresponding deterministic spaces, as introduced in Section 4.2.3 and employing a corresponding variational formulation [25]. However, both approaches are equivalent, provided that a gPC expansion for representing $L^2(\Omega)$ is employed.

Since the focus of this work is on applying a stochastic Galerkin projection employing generalized Polynomial Chaos, recall that the gPC expansion of a classical solution to the Navier-Stokes equations is given by (cf. Chapter 2):

$$u(x, t; \xi) = \sum_{i=0}^{\infty} u_i(x, t)\psi_i(\xi), \tag{4.56}$$

$$p(x, t; \xi) = \sum_{i=0}^{\infty} p_i(x, t)\psi_i(\xi), \tag{4.57}$$

pointwise in $\overline{\mathcal{D} \times (0,T)}$, whereas $\{\psi_i\}_{i=0}^{\infty}$ is an orthogonal polynomial basis of $L^2(\Omega)$ with deterministic modes u_i and p_i, $i = 0, \ldots, \infty$ defined on $\overline{\mathcal{D} \times (0,T)}$. The discretization of the probability space is carried out via truncation of the series by restricting the maximum polynomial degree to $No \in \mathbb{N}$, such that the total number $P + 1$ of modes is given by:

$$P + 1 = \frac{(No + L)!}{No!L!}, \tag{4.58}$$

whereas L denotes the dimension of the random vector ξ.

This defines a stochastic approximation employing a finite number of terms, i.e.,

$u \approx \sum_{i=0}^{P} u_i\psi_i$ and $p \approx \sum_{i=0}^{P} p_i\psi_i$, which by insertion into (4.50)–(4.53) yields:

$$\sum_{i=0}^{P} \partial_t u_i\psi_i + \sum_{i=0}^{P}\sum_{j=0}^{P}(u_i\cdot\nabla)u_j\psi_i\psi_j - \nu\sum_{i=0}^{P}\Delta u_i\psi_i + \frac{1}{\rho}\sum_{i=0}^{P}\nabla p_i\psi_i = f, \tag{4.59}$$

$$\sum_{i=0}^{P}\nabla\cdot u_i\psi_i = 0, \tag{4.60}$$

$$\sum_{i=0}^{P} u_i\psi_i = u^D, \tag{4.61}$$

$$\sum_{i=0}^{P} u_i(t=0)\psi_i = u_I, \tag{4.62}$$

almost surely in Ω and in $\mathcal{D}\times(0,T)$, $\Gamma\times[0,T]$ and $\mathcal{D}$, respectively. The Galerkin projection is carried out by multiplying these equations by some ψ_k for all $k = 0,\dots,P$, and taking the expectation, i.e., the inner-product on $L^2(\Omega)$, denoted by $\langle\cdot,\cdot\rangle$ such that due to the orthogonality of the Chaos Polynomials $\{\psi_k\}_{k=0}^{P}$ the resulting coupled deterministic system of PDEs reads:

$$\partial_t u_k + \sum_{i=0}^{P}\sum_{j=0}^{P}(u_i\cdot\nabla)u_j c_{ijk} - \nu\Delta u_k + \frac{1}{\rho}\nabla p_k = \frac{\langle f,\psi_k\rangle}{\langle\psi_k,\psi_k\rangle}, \quad \text{in } \mathcal{D}\times(0,T), \tag{4.63}$$

$$\nabla\cdot u_k = 0, \quad \text{in } \mathcal{D}\times(0,T), \tag{4.64}$$

$$u_k = \frac{\langle u^D,\psi_k\rangle}{\langle\psi_k,\psi_k\rangle}, \quad \text{on } \Gamma\times[0,T], \tag{4.65}$$

$$u_k(t=0) = \frac{\langle u_I,\psi_k\rangle}{\langle\psi_k,\psi_k\rangle}, \quad \text{in } \mathcal{D}, \tag{4.66}$$

for all $k = 0,\dots,P$, whereas

$$c_{ijk} := \frac{\langle\psi_i\psi_j,\psi_k\rangle}{\langle\psi_k,\psi_k\rangle} \in \mathbb{R},\ i,j,k = 0,\dots,P, \tag{4.67}$$

defines a tensor of third order (cf. Chapter 2), which accounts for the coupling terms in equation (4.63). Due to the Galerkin projection, the resulting system (4.63)–(4.66) is deterministic, whereas the solution is represented by a finite number of modes. Analog to the variational formulation outlined in Section 4.1, the corresponding variational formulation for (4.63)–(4.66) reads:

For given $f \in [L^2(\Omega)]^d \otimes L^2(0,T;H_{div})$ and $u_I \in [L^2(\Omega)]^d \otimes H_{div}$, find $u_k(t) \in u_k^D + V$ and $p_k(t) \in Q$, for $k = 0,\dots,P$, such that for almost every $t \in (0,T)$:

$$(\partial_t u_k, v) + \sum_{i=0}^{P}\sum_{j=0}^{P}((u_i\cdot\nabla)u_j, v)c_{ijk} + \nu(\nabla u_k,\nabla v) - (p_k,\nabla\cdot v) = (f_k,v), \quad \forall v\in V, \tag{4.68}$$

$$(\nabla\cdot u_k, q) = 0, \quad \forall q\in Q, \tag{4.69}$$

$$u_k(t=0) = u_{I,k}, \quad \text{in } \mathcal{D}, \tag{4.70}$$

for all $k = 0, \ldots, P$, whereas:

$$u_k^D := \frac{\langle u^D, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}, \tag{4.71}$$

$$f_k := \frac{\langle f, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}, \tag{4.72}$$

$$u_{I,k} := \frac{\langle u_I, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}. \tag{4.73}$$

Note, that in theory one could also allow for varying test function spaces with respect to the mode number $k = 0, \ldots, P$. However, this work is restricted to the case of a single space V for the velocity test functions and a single space Q for the pressure test functions.

Definition 4.4. A solution $\{(u_k, p_k)\}_{k=0}^P$ is called a weak solution to the stochastic Navier-Stokes equations if it satisfies (4.68)–(4.70) and the regularity conditions:

$$u_k(t) \in u_k^D + V, \tag{4.74}$$

$$p_k(t) \in Q, \tag{4.75}$$

for almost every $t \in (0, T)$ and $k = 0, \ldots, P$.

The definition of a weak solution to the stochastic Navier-Stokes equations implies that the continuous trial spaces are all the same for the velocity and pressure modes, respectively. However, the discretized trial spaces are allowed to vary for each stochastic mode, which requires an appropriate numerically stable choice due to the saddle-point structure of the problem. This will be addressed in detail in the following section.

The existence and uniqueness of a weak solution to the stochastic Navier-Stokes equations is a critical issue. Even in the deterministic case, at least the uniqueness of a weak solution is still an open question in three space dimensions. Of course, the results of Section 4.1 can be applied pointwise in Ω, however, to the present knowledge of the author a proof of existence and uniqueness of a weak solution to the stochastic Navier-Stokes equations as stated in this work is still effort of ongoing research.

4.3.2. LBB condition for the stochastic weak formulation

The space and time discretization of (4.68)–(4.70) can be carried out very similar to the approach outlined in Section 4.2, i.e., first the time variable is discretized via a Crank-Nicolson scheme and the resulting semi-discrete system is further discretized by employing Finite-Elements. However, for numerical stability it should also be ensured that the LBB condition (4.38) is satisfied within the stochastic context, when defining appropriate Finite-Element spaces for the velocity and pressure modes. The following corollary will show, that numerically stable Finite-Elements are given by a tensor product of the Taylor-Hood elements (cf. Section 4.2.3) for every mode.

Corollary 4.5. *Let $\mathcal{T}_h$ be a triangulation of a polygonal domain $\overline{\mathcal{D}}$ consisting of triangles, rectangles, tetrahedrons or hexahedrons and*

$$V_h := \{v_h \in C(\overline{\mathcal{D}})^d \cap H_0^1(\mathcal{D})^d; v_{h|T} \in \mathcal{P}_n^d(T), \quad \forall T \in \mathcal{T}_h\}, \tag{4.76}$$

$$Q_h := \{q_h \in C(\mathcal{D}) \cap L^2(\mathcal{D}); q_{h|T} \in \mathcal{P}_{n-1}(T), \quad \forall T \in \mathcal{T}_h\}, \tag{4.77}$$

with respect to some degree $n \in \mathbb{N}$. Furthermore, let $\tilde{b} : [Q_h]^{P+1} \times [V_h]^{P+1} \longrightarrow \mathbb{R}$ be defined by

$$\tilde{b}(\vec{q}_h, \vec{v}_h) := \sum_{i=0}^{P} b(q_{i,h}, v_{i,h}), \tag{4.78}$$

with $\vec{q}_h = [q_{0,h}, \ldots, q_{P,h}]^t \in [Q_h]^{P+1}$ and $\vec{v}_h = [v_{0,h}, \ldots, v_{P,h}]^t \in [V_h]^{P+1}$. Then it holds:

$$\exists \beta > 0 : \sup_{\vec{v}_h \in [V_h]^{P+1} \backslash \{0\}} \frac{\tilde{b}(\vec{q}_h, \vec{v}_h)}{\|\vec{v}_h\|_1} \geq \beta \|\vec{q}_h\|_1, \quad \forall \vec{q}_h \in [Q_h]^{P+1} \backslash \{0\}, \tag{4.79}$$

whereas

$$\|\vec{x}_h\|_1 := \sum_{i=0}^{P} \|x_{i,h}\|, \ \vec{x}_h \in [X_h]^{P+1}, \tag{4.80}$$

with X_h being V_h or Q_h, respectively.

Proof. It is known from Section 4.2.3 that for the chosen discretization spaces V_h and Q_h the LBB condition

$$\exists \beta > 0 : \sup_{v_h \in V_h \backslash \{0\}} \frac{b(q_h, v_h)}{\|v_h\|} \geq \beta \|q_h\|, \quad \forall q_h \in Q_h \backslash \{0\}, \tag{4.81}$$

holds. Furthermore, let $\epsilon > 0$, then for a given arbitrary $q_h \in Q_h \backslash \{0\}$ there exists some $\bar{v}_h \in V_h$, $\bar{v}_h \neq 0$, such that:

$$\frac{b(q_h, \bar{v}_h)}{\|\bar{v}_h\|} > \sup_{v_h \in V_h \backslash \{0\}} \frac{b(q_h, v_h)}{\|v_h\|} - \epsilon. \tag{4.82}$$

Next, define $\vec{\bar{v}}_h := [\bar{v}_h, \ldots, \bar{v}_h]^t \in [V_h]^{P+1} \backslash \{0\}$. Hence, for a given arbitrary $\vec{q}_h := [q_{0,h}, \ldots, q_{P,h}]^t \in [Q_h]^{P+1} \backslash \{0\}$ it holds:

$$\begin{aligned}
\sup_{\vec{v}_h \in [V_h]^{P+1} \backslash \{0\}} \frac{\tilde{b}(\vec{q}_h, \vec{v}_h)}{\|\vec{v}_h\|} &\geq \frac{\tilde{b}(\vec{q}_h, \vec{\bar{v}}_h)}{\|\vec{\bar{v}}_h\|} \\
&= \frac{\sum_{i=0}^{P} b(q_{i,h}, \bar{v}_h)}{\sum_{i=0}^{P} \|\bar{v}_h\|} \\
&= \frac{1}{P+1} \sum_{i=0}^{P} \frac{b(q_{i,h}, \bar{v}_h)}{\|\bar{v}_h\|} \\
&> \frac{1}{P+1} \sum_{i=0}^{P} \left(\sup_{v_h \in V_h \backslash \{0\}} \frac{b(q_{i,h}, v_h)}{\|v_h\|} - \epsilon \right), \quad \text{by (4.82).}
\end{aligned}$$

From equation (4.81) it follows:
$\exists \beta > 0 \; \forall \vec{q}_h = [q_{0,h}, \ldots, q_{P,h}]^t \in [Q_h]^{P+1} \backslash \{0\}$:

$$\sup_{\vec{v}_h \in [V_h]^{P+1} \backslash \{0\}} \frac{\tilde{b}(\vec{q}_h, \vec{v}_h)}{\|\vec{v}_h\|} > \frac{1}{P+1} \left(\sum_{i=0}^{P} \|q_{i,h}\| \beta - \epsilon \right) = \frac{1}{P+1} \left(\|\vec{q}_h\| \beta - \epsilon \right), \tag{4.83}$$

since β is independent of the choice of $\vec{q}_h \in [Q_h]^{P+1} \backslash \{0\}$. Letting $\epsilon \to 0$ completes the proof. □

Corollary 4.5 basically states that if for each mode of the trial and test functions the Taylor-Hood elements are employed, the corresponding LBB condition to the SNSE holds, which suggests a numerically stable Finite-Element discretization.

4.3.3. Structure of the discretized system

When the discretization spaces V_h and Q_h are fixed for every mode, as introduced in the previous section, there is an equivalent way of determining the discretized system [25]. One could also start with a variational formulation only considering the deterministic parts of the solution, such that by applying the procedure described in Section 4.2 one obtains a stochastic linear system, similar to (4.45):

$$\begin{bmatrix} A(\xi) & B(\xi) \\ B^t(\xi) & 0 \end{bmatrix} \begin{bmatrix} \vec{u}_h(\xi) \\ \vec{p}_h(\xi) \end{bmatrix} = \begin{bmatrix} \vec{F}_1(\xi) \\ \vec{F}_2(\xi) \end{bmatrix}, \tag{4.84}$$

whereas $\vec{u}_h(\xi) = [u_h^1(\xi), \ldots, u_h^N(\xi)]^t \in L^2(\Omega; \mathbb{R}^N)$, $\vec{p}_h(\xi) = [p_h^1(\xi), \ldots, p_h^M(\xi)]^t \in L^2(\Omega; \mathbb{R}^M)$, $A(\xi) \in L^2(\Omega; \mathbb{R}^{N,N})$, $B(\xi) \in L^2(\Omega; \mathbb{R}^{N,M})$, $\vec{F}_1(\xi) \in L^2(\Omega; \mathbb{R}^N)$, and $\vec{F}_2(\xi) \in L^2(\Omega; \mathbb{R}^M)$, which exact definitions can be obtained by an analog stochastic representation of the variables defined in (4.45). Note that $N = \dim\{V_h\}$ and $M = \dim\{Q_h\}$. The P^{th} order gPC expansion of A, B, $\vec{u}_h$, $\vec{p}_h$, $\vec{F}_1$ and $\vec{F}_2$ reads:

$$A(\xi) = \sum_{i=0}^{P} A_i \psi_i(\xi), \; B(\xi) = \sum_{i=0}^{P} B_i \psi_i(\xi), \tag{4.85}$$

$$\vec{u}_h = \sum_{i=0}^{P} \vec{u}_{i,h} \psi_i(\xi), \quad \vec{p}_h = \sum_{i=0}^{P} \vec{p}_{i,h} \psi_i(\xi), \tag{4.86}$$

$$\vec{F}_1 = \sum_{i=0}^{P} \vec{F}_{i,1} \psi_i(\xi), \quad \vec{F}_2 = \sum_{i=0}^{P} \vec{F}_{i,2} \psi_i(\xi), \tag{4.87}$$

whereas $A_i \in \mathbb{R}^{N,N}$, $B_i \in \mathbb{R}^{N,M}$, $\vec{u}_{i,h} \in \mathbb{R}^N$, $\vec{p}_{i,h} \in \mathbb{R}^M$, $\vec{F}_{i,1} \in \mathbb{R}^N$ and $\vec{F}_{i,2} \in \mathbb{R}^M$ for $i = 0, \ldots, P$. Introducing these gPC expansions into equation (4.84) and applying a stochastic Galerkin projection on the space spanned by the Chaos Polynomials $\text{span}\{\psi_i, i = 0, \ldots, P\}$ (cf. Section 4.3.1) one obtains the fully discretized system:

$$\sum_{i=0}^{P} \sum_{j=0}^{P} \begin{bmatrix} A_i & B_i \\ B_i^t & 0 \end{bmatrix} \begin{bmatrix} \vec{u}_{j,h} \\ \vec{p}_{j,h} \end{bmatrix} c_{ijk} = \begin{bmatrix} \vec{F}_{k,1} \\ \vec{F}_{k,2} \end{bmatrix}, \quad \forall k = 0, \ldots, P, \tag{4.88}$$

whereas c_{ijk} denotes the entries of the third order tensor defined in Section 4.3.1. Equivalently, one can recast (4.88) such that the fully discretized system has the following structure:

$$\sum_{i=0}^{P} \underbrace{\begin{bmatrix} \begin{bmatrix} A_i c_{i00} & B_i c_{i00} \\ B_i^t c_{i00} & 0 \end{bmatrix} & \cdots & \begin{bmatrix} A_i c_{iP0} & B_i c_{iP0} \\ B_i^t c_{iP0} & 0 \end{bmatrix} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ \begin{bmatrix} A_i c_{i0P} & B_i c_{i0P} \\ B_i^t c_{i0P} & 0 \end{bmatrix} & \cdots & \begin{bmatrix} A_i c_{iPP} & B_i c_{iPP} \\ B_i^t c_{iPP} & 0 \end{bmatrix} \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M),(P+1)(N+M)}} \underbrace{\begin{bmatrix} \vec{u}_{0,h} \\ \vec{p}_{0,h} \\ \vdots \\ \vdots \\ \vdots \\ \vec{u}_{P,h} \\ \vec{p}_{P,h} \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M)}} = \underbrace{\begin{bmatrix} \vec{F}_{0,1} \\ \vec{F}_{0,2} \\ \vdots \\ \vdots \\ \vdots \\ \vec{F}_{P,1} \\ \vec{F}_{P,2} \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M)}} . \tag{4.89}$$

Also, it is possible to reorder the solution vector, such that the saddle-point structure is reflected more concretely. This results in the equivalent system:

$$\sum_{i=0}^{P} \underbrace{\begin{bmatrix} \begin{bmatrix} A_i c_{i00} & \cdots & A_i c_{iP0} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ A_i c_{i0P} & \cdots & A_i c_{iPP} \end{bmatrix} & \begin{bmatrix} B_i c_{i00} & \cdots & B_i c_{iP0} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ B_i c_{i0P} & \cdots & B_i c_{iPP} \end{bmatrix} \\ \begin{bmatrix} B_i^t c_{i00} & \cdots & B_i^t c_{iP0} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ B_i^t c_{i0P} & \cdots & B_i^t c_{iPP} \end{bmatrix} & 0 \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M),(P+1)(N+M)}} \underbrace{\begin{bmatrix} \vec{u}_{0,h} \\ \vdots \\ \vdots \\ \vec{u}_{P,h} \\ \vec{p}_{0,h} \\ \vdots \\ \vdots \\ \vec{p}_{P,h} \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M)}} = \underbrace{\begin{bmatrix} \vec{F}_{0,1} \\ \vdots \\ \vdots \\ \vec{F}_{P,1} \\ \vec{F}_{0,2} \\ \vdots \\ \vdots \\ \vec{F}_{P,2} \end{bmatrix}}_{\in \mathbb{R}^{(P+1)(N+M)}} . \tag{4.90}$$

Therefore, it can be observed more clearly, that the saddle-point structure of the deterministic problem is preserved within the stochastic context, requiring the use of numerically stable Finite-Element spaces, e.g., the Taylor-Hood elements introduced in Section 4.3.2. Furthermore, the size of the system matrix is a factor of $(P+1)^2$ larger than the corresponding system matrix of a deterministic system. Therefore, one often relies on numerical strategies, which try to solve (4.90) in a matrix-free approach to avoid assembly time and reduce the memory requirements. However, the third order tensor defined by its entries c_{ijk}, $i, j, k = 0, \ldots, P$, is sparse (cf. Chapter 2), such that many entries of the system matrix do not need to be computed and stored, resulting in a sparse structure of the system matrix itself with additional sparsity introduced through the Finite-Element discretization.

Chapter 5.

Periodic orbits of the unsteady stochastic incompressible Navier-Stokes equations

Solutions to fluid flow problems can be categorized roughly by three classes, namely steady state, time-periodic and unsteady without specific repeating patterns. In the stochastic context, a gPC expansion exhibits a similar convergence breakdown as being elaborated on in Chapter 3 for unsteady problems, which makes the use of gPC for the solution of the SNSE only applicable for limited time intervals. Therefore, this Chapter restricts its attention towards the determination of time-periodic solutions to the SNSE using gPC as the next hierarchy step to the steady state case. Such solutions are usually referred to as "periodic orbits" in context of dynamical systems, which are characterized by some period and initial condition exhibiting stable limit cycles. Their determination often plays an important role in transition phases between laminar and turbulent states arising in fluid flow problems. If uncertainties are involved, the period and initial condition become random quantities themselves such that a deterministic time discretization scheme can only be applied in sampling based strategies without imposing specific modifications. Therefore, this chapter introduces a new numerical method for determining solutions of the SNSE, which are almost surely periodic subject to a stochastic period, by extending a deterministic approach introduced by Duguet et al. [9] to the unsteady stochastic incompressible Navier-Stokes equations using the Spectral-Stochastic-Finite-Element-Method introduced in Chapter 4. This makes it possible to characterize all trajectories of the random events by a functional representation of the stochastic period and initial condition with respect to the random input.

5.1. Problem definition

Recall the unsteady stochastic incompressible Navier-Stokes equations (SNSE) (cf. Chapter 4), which are given by:

$$\partial_t u(\xi) + (u(\xi)\cdot\nabla)u(\xi) - \nu\Delta u(\xi) + \frac{1}{\rho}\nabla p(\xi) = 0, \qquad \text{in } \mathcal{D}\times(0,\tilde{T}), \tag{5.1}$$

$$\nabla\cdot u(\xi) = 0, \qquad \text{in } \mathcal{D}\times(0,\tilde{T}), \tag{5.2}$$

$$u(\xi) = u^D(\xi), \quad \text{on } \Gamma\times(0,\tilde{T}), \tag{5.3}$$

$$u(t=0;\xi) = u_I(\xi), \qquad \text{in } \mathcal{D}, \tag{5.4}$$

almost surely in Ω subject to an open and bounded domain $\mathcal{D}\subset\mathbb{R}^d$, a Lipschitz-continuous Dirichlet boundary $\Gamma\subset\partial\mathcal{D}$ and time interval $[0,\tilde{T}]$, whereas ξ represents a finite dimensional random vector defined on some underlying probability

space $(\Omega, \mathcal{F}, \mathbb{P})$. Note, that no external forces are being considered resulting in a zero valued right hand side in equation (5.1). The uncertainty is introduced by boundary u^D and initial conditions u_I leading to an uncertain period $T = T(\xi)$ for almost surely time-periodic solutions. For the rest of this chapter a fundamental assumption shall hold:

Assumption 5.1. There exists a time-periodic solution u to (5.1)–(5.3), characterized by some initial solution $u(t = 0) = u_I \in [L^2(\Omega)]^d \otimes [C^2(\overline{\mathcal{D}})]^d$ satisfying (5.2)–(5.3) and period $T \in L^2(\Omega)$, $\tilde{T} \geq T \geq \alpha > 0$ almost surely in Ω, for some $\alpha \in \mathbb{R}$, such that for all $t \in [0, T(\xi)]$:

$$u(x, t + T(\xi); \xi) = u(x, t; \xi), \quad \forall x \in \overline{\mathcal{D}}, \tag{5.5}$$

almost surely in Ω, whereas $u \in [L^2(\Omega)]^d \otimes [C^2(\overline{\mathcal{D} \times (0, \tilde{T})})]^d)$ satisfies (5.1)–(5.4).

The main problem in computing stochastic periodic orbits arises from the stochastic nature of the period T. A standard gPC computation, based on a deterministic time-discretization method (cf. Chapter 4) will have difficulties in capturing the stochastic variations of the period, since the trajectories of a solution to the SNSE depend on a random event $\omega \in \Omega$. Therefore, the approach outlined in the following aims at representing the stochastic period as some additional random input within the SNSE, introducing a new random variable, which computation requires an additional condition to the SNSE. Therefore, this allows the use of a deterministic simulation time interval to compute a pair of initial condition and period, since the uncertainty within the time interval is transfered towards the system equations. This is achieved by introducing a new scaled time variable λ which is defined by:

$$\lambda := \frac{t}{T(\xi)}, \quad \text{pointwise in } \Omega, \ t \in [0, (1 + \delta T)T(\xi)], \tag{5.6}$$

whereas $\delta T \in \mathbb{R}$ is some predefined deterministic time length, which is allowed to become positive as well as negative valued. Note that λ is a random variable itself, however, its range is deterministic and restricted to the scaled time interval $[0, 1 + \delta T]$ almost surely in Ω. Furthermore, $\lambda \in L^2(\Omega)$, provided that the period T satisfies $T \geq \alpha > 0$ almost surely in Ω, which is assumed in Assumption 5.1. The rescaling with respect to λ has an effect on the time derivative within equation (5.1) for which it holds by applying the chain rule:

$$\begin{aligned} \partial_t u(x, \lambda; \xi) &= \frac{\partial}{\partial t}(u(x, \lambda; \xi)), && (5.7)\\ &= \frac{\partial \lambda}{\partial t}\frac{\partial}{\partial \lambda} u(x, \lambda; \xi), && (5.8)\\ &= \frac{1}{T(\xi)} \partial_\lambda u(x, \lambda; \xi) && (5.9) \end{aligned}$$

almost surely in Ω. Introducing the scaled time λ into (5.1)–(5.3) results in a scaled version of the unsteady stochastic incompressible Navier-Stokes equations:

$$\begin{aligned} &\partial_\lambda u(\xi) + T(\xi)(u(\xi) \cdot \nabla) u(\xi) \\ &\quad -\nu T(\xi) \Delta u(\xi) + \nabla \tilde{p}(\xi) = 0, && \text{in } \mathcal{D} \times (0, 1 + \delta T), && (5.10)\\ &\quad \nabla \cdot u(\xi) = 0, && \text{in } \mathcal{D} \times (0, 1 + \delta T), && (5.11)\\ &\quad u(\xi) = u^D(\xi), && \text{on } \Gamma \times (0, 1 + \delta T), && (5.12) \end{aligned}$$

almost surely in Ω. For the rest of this chapter equations (5.10)–(5.12) are being referred to as scaled Navier-Stokes equations, or shortly S-SNSE, for notational convenience. Note that the pressure variable has been redefined by:

$$\tilde{p}(\xi) := \frac{T(\xi)}{\rho} p, \tag{5.13}$$

almost everywhere in $\overline{\mathcal{D} \times (0, 1 + \delta T)}$ and almost surely in Ω, which reduces the computational complexity due to a simplified system structure (cf. Section 5.2). In the following the tilde notation for the pressure variable $\tilde{p}$ will be dropped for notational convenience.

To formulate the problem definition for this chapter, the following operator is defined:

$$I(u_I, T, \lambda) := u_I + \int_0^\lambda \partial_\lambda u, \tag{5.14}$$

whereas $u \in [L^2(\Omega)]^d \otimes [C^2(\overline{\mathcal{D} \times (0, 1 + \delta T)})]^d)$ satisfies (5.10)–(5.12) up to the scaled time λ for a given initial condition u_I and period T as stated in Assumption 5.1. Therefore, the problem definition for finding stochastic periodic orbits of the stochastic Navier-Stokes equations reads:

Find some $u_I \in [L^2(\Omega)]^d \otimes [C^2(\overline{\mathcal{D}})]^d$ satisfying (5.11)–(5.12) and a corresponding $T \in L^2(\Omega)$, $T \geq \alpha > 0$ almost surely in Ω, such that:

$$\|I(u_I, T, 1) - u_I\|_{L^2(\mathcal{D})} = 0, \quad a.s. \quad \Leftrightarrow \quad \sqrt{\mathbb{E}\|I(u_I, T, 1) - u_I\|^2_{L^2(\mathcal{D})}} = 0. \tag{5.15}$$

Note that $\sqrt{\mathbb{E}\|\cdot\|^2_{L^2(\mathcal{D})}}$ defines a norm on $\Omega \times \mathcal{D}$, since $\sqrt{\mathbb{E}\|\cdot\|^2_{L^2(\mathcal{D})}} = \|\cdot\|_{L^2(\Omega \times \mathcal{D})}$, which in the following shall be denoted by $\|\cdot\|$. Furthermore, it is important to emphasize that no uniqueness of an initial condition satisfying (5.15) is to be expected.

5.2. Variational formulation

As introduced in Chapter 4 the system in (5.10)–(5.12) shall be expressed in a variational formulation, enabling the use of Finite-Elements in the spatial domain and reducing the regularity requirements on a solution to the scaled Navier-Stokes equations. Therefore, first u, p, T, u^D are approximated by their corresponding gPC expansion employing a finite number of terms:

$$T(\xi) = \sum_{i=0}^{P^T} T_i \psi_i(\xi), \tag{5.16}$$

$$u(x, \lambda; \xi) = \sum_{i=0}^{P} u_i(x, \lambda) \psi_i(\xi), \tag{5.17}$$

$$p(x, \lambda; \xi) = \sum_{i=0}^{P} p_i(x, \lambda) \psi_i(\xi), \tag{5.18}$$

$$u^D(x; \xi) = \sum_{i=0}^{P} u_i^D(x) \psi_i(\xi). \tag{5.19}$$

Here, the truncation parameters P^T for the period and P for the velocity and pressure variables satisfy:

$$(P^T + 1) = \frac{(No^T + L)!}{No^T!L!}, \tag{5.20}$$

$$(P + 1) = \frac{(No + L)!}{No!L!}, \tag{5.21}$$

whereas $L \in \mathbb{N}$ denotes the dimension of the random vector $\xi = (\xi_1, \ldots, \xi_L)$, $No^T \in \mathbb{N}$ the maximum polynomial degree employed for the period and $No \in \mathbb{N}$ the maximum polynomial degree employed for the velocity and pressure variables. Next, these discretizations are inserted into the scaled Navier-Stokes equations (5.10)–(5.12) and the stochastic Galerkin projection onto the space spanned by the Chaos Polynomials, i.e., onto $\text{span}\{\psi_i, i = 0, \ldots, P\}$, is applied as described in Section 4.3.1, resulting in:

$$\begin{aligned} &\partial_\lambda u_k + \sum_{i=0}^{P^T}\sum_{j=0}^{P}\sum_{l=0}^{P} T_i (u_j \cdot \nabla) u_l c_{ijlk} \\ &\quad -\nu \sum_{i=0}^{P^T}\sum_{j=0}^{P} T_i \Delta u_j c_{ijk} + \nabla p_k = 0, \qquad \text{in } \mathcal{D} \times (0, 1 + \delta T), \end{aligned} \tag{5.22}$$

$$\nabla \cdot u_k = 0, \qquad \text{in } \mathcal{D} \times (0, 1 + \delta T), \tag{5.23}$$

$$u_k = u_k^D, \quad \text{on } \Gamma \times (0, 1 + \delta T), \tag{5.24}$$

for $k = 0, \ldots, P$, whereas the corresponding 4^{th} and 3^{rd} order tensors are defined by:

$$c_{ijlk} := \frac{\langle \psi_i \psi_j \psi_l, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}, \tag{5.25}$$

$$c_{ijk} := \frac{\langle \psi_i \psi_j, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}. \tag{5.26}$$

for $i = 0, \ldots, P^T$ and $j, l, k = 0, \ldots, P$. Note that both tensors have a sparse structure (cf. Chapter 2), which leads to many zero valued coupling terms in (5.22). Furthermore, the redefinition of the pressure variable in (5.13) plays an important role, since the pressure term appears completely decoupled in (5.22), which would not be the case if the pressure term would still involve a product with the stochastic period $T(\xi)$, as being the case for the viscosity term. However, the number of coupling terms within (5.22) is significantly increased in comparison to the standard unscaled formulation stated in Section 4.3.1 due to the nonlinear convective term. This has a significant impact on the computational complexity when assembling the stiffness-matrix and residuals by a numerical solver.

Having stated the stochastic Galerkin projected system (5.22)–(5.23), a variational formulation can be derived analog to the variational formulation outlined in Section 4.3.1, resulting in:

For given period modes T_k, $k = 0, \ldots, P^T$ find $u_k \in u_k^D + V$ and $p_k \in Q$, $k = 0, \ldots, P$ such that:

$$(\partial_\lambda u_k, v) + \sum_{i=0}^{P^T} \sum_{j=0}^{P} \sum_{l=0}^{P} T_i((u_j \cdot \nabla)u_l, v)c_{ijlk}$$
$$+\nu \sum_{i=0}^{P^T} \sum_{j=0}^{P} T_i(\nabla u_j, \nabla v)c_{ijk} - (p_k, \nabla \cdot v) = 0, \quad \forall v \in V, \tag{5.27}$$

$$(\nabla \cdot u_k, q) = 0, \quad \forall q \in Q, \tag{5.28}$$

almost everywhere in $(0, 1 + \delta T)$ for $k = 0, \ldots, P$ subject to a given initial condition u_I with test function spaces V and Q defined in Section 4.1. Note that the variational formulation implies a predefined value for the stochastic period T, which suggest an iterative approach when computing a pair of period T and corresponding initial condition u_I. This will be explained in detail within the following section. Furthermore, the integration operator I introduced in (5.14) is adapted to the variational formulation, i.e.,

$$I(u_I, T, \lambda) := u_I + \int_0^\lambda \partial_\lambda u, \tag{5.29}$$

whereas $u = \sum_{i=0}^{P} u_i \psi_i$, with $u_i \in u_i^D + V$ almost everywhere in $(0, 1 + \delta T)$ for $i = 0, \ldots, P$, satisfies (5.27)–(5.28) up to the scaled time λ for a given initial condition $u_I = \sum_{i=0}^{P} u_{I,i} \psi_i$, with $u_{I,i} \in u_i^D + V$, $i = 0, \ldots, P$, and period $T \in L^2(\Omega)$.

5.3. Solution procedure

In the following an iterative method, based on a deterministic approach introduced by Duguet et al. [9], will be described to compute a pair of initial condition u_I and period T satisfying (5.15), employing Newton's method, Finite-Elements and optimization techniques.

The iterative numerical algorithm starts by choosing some appropriate initial guesses $u_I^0 = \sum_{i=0}^{P} u_{I,i}^0 \psi_i$ and $T^0 = \sum_{i=0}^{P^T} T_i^0 \psi_i$ for the initial condition and period, respectively. The choice of these initial guesses has a strong impact on the convergence behavior of the iteration procedure, since it involves Newton's method as will be shown later within this section. Furthermore, for Newton's method it is known [6] that it only converges if the initial guess is "close enough" to a solution, which in general cannot be guaranteed.

Fixing some iteration index, say k, a distance vector $D^k : [0, 1+\delta T] \longrightarrow [L^2(\Omega)]^d \otimes [H^1(\mathcal{D})]^d$ is defined by:

$$D^k(\lambda) := u_I^k - I(u_I^k, T^k, \lambda), \quad \lambda \in [0, 1 + \delta T]. \tag{5.30}$$

Therefore, $\|D^k\|$ measures the distance between a current iterate u_I^k and its terminal state $I(u_I^k, T^k, \lambda)$ at time λ, subject to the current period iterate T^k. The goal of the iteration procedure is to obtain convergence such that $\|D^k\| \to 0$ as $k \to \infty$.

Achieving this goal is carried out by two essential steps:

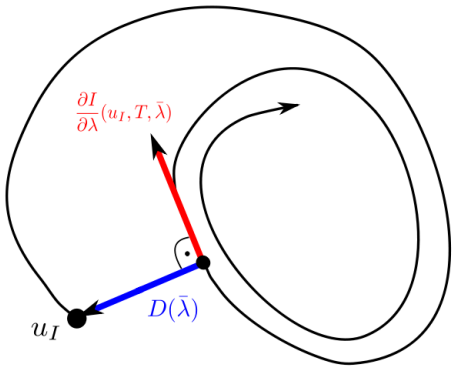


Figure 5.1.: Schematic view of a trajectory starting at some initial condition u_I with distance vector $D(\bar{\lambda})$ and time derivative $\frac{\partial I}{\partial \lambda}$ of the terminal state at minimization time $\bar{\lambda}$.

- A predictor–corrector step for determining a new period iterate T^{k+1},
- Newton's method for determining a new initial condition iterate u_I^{k+1}.

5.3.1. A predictor-corrector step for the period update

So-called "predictor-corrector" algorithms are a concept of algorithms which first try to estimate a solution to some given problem and afterwards aim to correct the error associated with the prediction. For the period update such a prediction can be obtained by determining some deterministic scaling time $\bar{\lambda} \in [0, 1 + \delta T]$, which minimizes $\lambda \mapsto \|D^k(\lambda)\|^2$ subject to the mean $\bar{\xi}$ of ξ. Therefore, $\bar{\lambda}$ provides an estimate for a new period iterate with respect to the mean $\bar{\xi}$ of the random input vector and serves as a global scaling information for every other realization of ξ. This leads to a deterministic optimization problem, for which the necessary optimality criteria reads:

$$\alpha^k(\lambda) := 2 \int_{\mathcal{D}} D^k(\lambda) \partial_\lambda I(u_I^k, T^k, \lambda)\, dx = 0, \quad \text{subject to } \bar{\xi}, \tag{5.31}$$

which can be easily derived by differentiating $\lambda \mapsto \|D^k(\lambda)\|^2$ with respect to λ and taking into account that the operators involved are deterministic with respect to $\bar{\xi}$. Equation (5.31) basically states that the critical points satisfying the optimality criteria require the distance vector D^k to be orthogonal to the tangential vector $\frac{\partial I}{\partial \lambda}$ of the time trajectory of the corresponding terminal state, which is depicted in Figure 5.1. Therefore, the minimization point ensures that the trajectory returns closest to the initial condition u_I^k at time $\bar{\lambda}$.

Furthermore, the predefined $\delta T \in \mathbb{R}$ has an effect on the number of critical points satisfying (5.31). Its choice is certainly application dependent and it should be ensured that the trajectory is able to return at least once closest to the initial condition.

A straight forward way of solving (5.31) is by monitoring $\alpha^k(\lambda)$ at every discrete time step, say λ_n, whereas n denotes some time iteration index. If a sign change of $\alpha^k(\lambda)$ is observed for two time steps λ_n and λ_{n+1}, then a critical point can be approximated by a linear interpolation between λ_n and λ_{n+1}, such that for a homogeneous time discretization, characterized by some constant time step length $\delta\lambda > 0$, the interpolation reads:

$$\lambda_n^{crit} := n\delta\lambda + \frac{-\delta\lambda\alpha^k(\lambda_n)}{\alpha^k(\lambda_{n+1}) - \alpha^k(\lambda_n)} > 0. \tag{5.32}$$

For a non-homogeneous time stepping an analog defined linear interpolation can be applied.

The exact minimization point $\bar{\lambda}$ can be approximated by λ_n^{crit}, whereas n is chosen such that:

$$\|D^k(\lambda_n)\| \text{ is minimal w.r.t. } n. \tag{5.33}$$

For $\delta\lambda$ sufficiently small, (5.33) provides an accurate approximation of the exact minimization point, having the advantage that the distance $\|D^k(\lambda)\|$ does not need to be approximated by some further interpolation, instead it suffices to monitor the quantities $\|D^k(\lambda_n)\|$.

In a deterministic framework, this prediction would be sufficient to obtain a new period iterate T^{k+1} [9]. However, since the underlying problem is of a stochastic nature, the calculated minimization time $\bar{\lambda}$ is only optimal for the mean $\bar{\xi}$ of ξ by construction. In general, other realizations of ξ will not satisfy the optimality criteria (5.31) at $\lambda = \bar{\lambda}$. Therefore, a stochastic correction step needs to be applied, which ensures that the distance between the initial condition and terminal state remains minimal almost surely in Ω. Hence, such an appropriate correction term, say $d\lambda \in L^2(\Omega)$, is required to satisfy:

$$d\lambda = \arg\min_{d\lambda} \|u_I^k - I(u_I^k, T^k, \bar{\lambda} + d\lambda)\|^2, \tag{5.34}$$

which leads to a stochastic nonlinear optimization problem. It is also possible to skip the deterministic prediction step and try to solve (5.34) directly. However, having a prediction $\bar{\lambda}$, the stochastic correction term $d\lambda$ can be assumed to be small in magnitude, provided that different realizations of ξ don't cause a large variation in the corresponding terminal states. This justifies a linearization approach, whereas the terminal state $I(u_I^k, T^k, \bar{\lambda} + d\lambda)$ is approximated by its first order Taylor series representation around $\bar{\lambda}$, i.e.,

$$I(u_I^k, T^k, \bar{\lambda} + d\lambda) \approx I(u_I^k, T^k, \bar{\lambda}) + \partial_\lambda I(u_I^k, T^k, \bar{\lambda})d\lambda, \quad a.s. \text{ in } \Omega. \tag{5.35}$$

Furthermore, since $d\lambda \in L^2(\Omega)$, $d\lambda$ can be approximated by a gPC expansion of order P^T given by:

$$d\lambda(\xi) \approx \sum_{i=0}^{P^T} d\lambda_i \psi_i(\xi), \tag{5.36}$$

with $d\lambda_i \in \mathbb{R}$, $i = 0, \ldots, P^T$. Inserting (5.35) and (5.36) in (5.34) one obtains a linearized stochastic optimization problem for the stochastic modes $d\lambda_i$, $i =$

$0, \dots, P^T$, of the correction term $d\lambda$:

$$\min_{d\lambda_0,\dots,d\lambda_{P^T}} \|D^k(\bar{\lambda}) - \partial_\lambda I(u_I^k, T^k, \bar{\lambda}) \sum_{j=0}^{P^T} d\lambda_j \psi_j\|^2. \tag{5.37}$$

Lemma 5.2. *The optimization problem, given in equation* (5.37)*, is convex, and the necessary and sufficient conditions for the minimizing stochastic mode vector* $\vec{d\lambda} = [d\lambda_0, \dots, d\lambda_{P^T}]^t \in \mathbb{R}^{P^T}$ *are given by:*

$$A\ \vec{d\lambda} = b, \tag{5.38}$$

whereas $A \in \mathbb{R}^{P^T, P^T}$ *and* $b \in \mathbb{R}^{P^T}$ *are defined by:*

$$A_{ml} := \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} c_{ijlm} (\partial_\lambda I(u_I^k, T^k, \bar{\lambda})_i, \partial_\lambda I(u_I^k, T^k, \bar{\lambda})_j), \tag{5.39}$$

$$b_m := \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} c_{ijm} (D^k(\bar{\lambda})_i, \partial_\lambda I(u_I^k, T^k, \bar{\lambda})_j), \tag{5.40}$$

with $(\cdot,\cdot)$ *denoting the* L^2*-inner-product on* $\mathcal{D}$*. The corresponding gPC coefficients of* $\partial_\lambda I$ *and* D^k *are given by:*

$$\partial_\lambda I(u_I^k, T^k, \bar{\lambda})_i = \frac{\langle \partial_\lambda I(u_I^k, T^k, \bar{\lambda}), \psi_i \rangle}{\langle \psi_i, \psi_i \rangle}, \tag{5.41}$$

$$D^k(\bar{\lambda})_i = \frac{\langle D^k(\bar{\lambda}), \psi_i \rangle}{\langle \psi_i, \psi_i \rangle}, \tag{5.42}$$

for $i = 0, \dots, P^T$*.*

Proof. For deriving the optimality criteria and showing that (5.37) is convex, it is sufficient to determine the Jacobian and Hessian of (5.37) with respect to the stochastic modes $d\lambda_i$, $i = 0, \dots, P^T$. For the Hessian it will be shown that it is positive semi-definite, which proves the convexity. From this it follows that setting the Jacobian equal to zero provides the necessary and sufficient condition for the minimization vector $\vec{d\lambda} = [d\lambda_0, \dots, d\lambda_{P^T}]^t \in \mathbb{R}^{P^T}$. Furthermore, for notational convenience, $\dot{I}^k$ and D^k are defined by $\dot{I}^k := \partial_\lambda I(u_I^k, T^k, \bar{\lambda})$ and $D^k := D^k(\bar{\lambda})$. Therefore, the optimality criteria can be derived by:

$$\frac{\partial}{\partial d\lambda_m} \|D^k(\bar{\lambda}) - \partial_\lambda I(u_I^k, T^k, \bar{\lambda}) \sum_{l=0}^{P^T} d\lambda_l \psi_l\|^2 = 0, \quad \forall m = 0, \dots, P^T, \tag{5.43}$$

$$\Leftrightarrow \quad -2\mathbb{E}[(D^k - \dot{I}^k \sum_{l=0}^{P^T} d\lambda_l \psi_l, \dot{I}^k \psi_m)] = 0, \quad \forall m = 0, \dots, P^T, \tag{5.44}$$

$$\Leftrightarrow \quad \mathbb{E}[(\dot{I}^k \sum_{l=0}^{P^T} d\lambda_l \psi_l, \dot{I}^k \psi_m)] = \mathbb{E}[(D^k, \dot{I}^k \psi_m)], \quad \forall m = 0, \dots, P^T. \tag{5.45}$$

Introducing the gPC expansions $\dot{I}^k = \sum_{i=0}^{P^T} \dot{I}_i^k \psi_i$ and $D^k = \sum_{i=0}^{P^T} D_i^k \psi_i$ into (5.45) one obtains for all $m = 0, \ldots, P^T$:

$$\mathbb{E}[(\sum_{i=0}^{P^T} \dot{I}_i^k \psi_i \sum_{l=0}^{P^T} d\lambda_l \psi_l, \sum_{j=0}^{P^T} \dot{I}_j^k \psi_j \psi_m)] = \mathbb{E}[(\sum_{i=0}^{P^T} D_i^k \psi_i, \sum_{j=0}^{P^T} \dot{I}_j^k \psi_j \psi_m)], \tag{5.46}$$

$$\Leftrightarrow \quad \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} \sum_{l=0}^{P^T} \langle \psi_i \psi_j \psi_l, \psi_m \rangle (\dot{I}_i^k, \dot{I}_j^k) = \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} \langle \psi_i \psi_j, \psi_m \rangle (D_i^k, \dot{I}_j^k). \tag{5.47}$$

Dividing by $\langle \psi_m, \psi_m \rangle$ gives the stated optimality criteria. To prove the convexity one has to determine the second partial derivatives of (5.37) with respect to some $d\lambda_m$ and $d\lambda_n$, $n, m = 0, \ldots, P^T$:

$$h_{mn} := \frac{\partial^2}{\partial d\lambda_n \partial d\lambda_m} \| D^k(\bar{\lambda}) - \partial_\lambda I(u_I^k, T^k, \bar{\lambda}) \sum_{l=0}^{P^T} d\lambda_l \psi_l \|^2 \tag{5.48}$$

$$= \frac{\partial}{\partial d\lambda_n} (-2\mathbb{E}[(D^k - \dot{I}^k \sum_{l=0}^{P^T} d\lambda_l \psi_l, \dot{I}^k \psi_m)]) \tag{5.49}$$

$$= 2\mathbb{E}[(\dot{I}^k \psi_n, \dot{I}^k \psi_m)]. \tag{5.50}$$

This defines the Hessian $H := (h_{mn})_{m,n=0}^{P^T}$, for which it needs to be shown that it is positive semi-definite such that the convexity of the optimization problem is proved. Therefore, consider some given vector $y = [y_0, \ldots, y_{P^T}]^t \in \mathbb{R}^{P^T}$, $y \neq 0$, and observe that the following relations hold:

$$y^t H y = \sum_{m=0}^{P^T} \sum_{n=0}^{P^T} y_m h_{mn} y_n, \tag{5.51}$$

$$= 2 \sum_{m=0}^{P^T} \sum_{n=0}^{P^T} y_m \mathbb{E}[(\dot{I}^k \psi_n, \dot{I}^k \psi_m)] y_n, \tag{5.52}$$

$$= 2\mathbb{E}[(\sum_{m=0}^{P^T} y_m \psi_m \dot{I}^k, \sum_{n=0}^{P^T} y_n \psi_n \dot{I}^k)], \tag{5.53}$$

$$= 2\| \sum_{m=0}^{P^T} y_m \psi_m \dot{I}^k \|^2 \geq 0. \tag{5.54}$$

Therefore, H is positive semi-definite, which completes the proof. □

Having determined the stochastic correction term $d\lambda$, the new period iterate T^{k+1} can be obtained by

$$(\bar{\lambda} + d\lambda) = \frac{T^{k+1}}{T^k} \Leftrightarrow T^{k+1} = (\bar{\lambda} + d\lambda) T^k, \tag{5.55}$$

by definition of the scaled time λ (set $t = T^{k+1}$, cf. Eq. (5.6)). Further calculation yields:

$$T_m^{k+1} = \bar{\lambda} T_m^k + \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} d\lambda_i T_j^k c_{ijm}, \quad m = 0, \ldots, P^T, \tag{5.56}$$

for the corresponding modes T_m^{k+1}, $m = 0, \ldots, P^T$, of the period iterate $T^{k+1} = \sum_{m=0}^{P^T} T_m^{k+1}\psi_m$.

Summing up, the determination of the new period update T^{k+1} requires the solution of a deterministic reference problem, given by the deterministic Navier-Stokes equations, for the mean $\bar{\xi}$ of the random input ξ to obtain a deterministic scaling information and the solution of a linear system of equations, as stated in Lemma 5.2, for P^T unknowns to determine a stochastic correction term, which can be carried out by employing standard direct numerical solvers.

5.3.2. Newton's method for updating the initial condition

Updating the period iterate ensures that the terminal state has a minimal distance with respect to the current initial condition iterate u_I^k. Ideally, this minimal distance would be equal to zero, for which the algorithm would terminate, however, in general the minimum is greater than zero, which necessitates a correction of u_I^k such that the distance can be decreased further. To this end, the updated distance vector, say $D^{k+1/2} : [L^2(\Omega)]^d \otimes [H^1(\mathcal{D})]^d \longrightarrow [L^2(\Omega)]^d \otimes [H^1(\mathcal{D})]^d$, subject to the new period iterate T^{k+1} is defined by:

$$D^{k+1/2}(u_I) := u_I - I(u_I, T^{k+1}, 1), \tag{5.57}$$

such that $\|D^{k+1/2}\|$ measures the distance between the variable initial condition u_I and its terminal state $I(u_I, T^{k+1}, 1)$ after one cycle with respect to the period T^{k+1}. Therefore, Newton's method applied to $D^{k+1/2}(u_I) = 0$ yields the desired update formula for the new iterate u_I^{k+1} for the initial condition. This update is given by one Newton step, which reads:

$$u_I^{k+1} = u_I^k + du_I^k, \tag{5.58}$$

$$-J_k[du_I^k] = D^{k+1/2}(u_I^k), \tag{5.59}$$

whereas $J_k[du_I^k]$ denotes the Jacobian of $D^{k+1/2}$ with respect to u_I in direction du_I^k at $u_I = u_I^k$. Since $J_k[du_I^k]$ involves the Jacobian of the Navier-Stokes equations in direction du_I^k, it provides a sensitivity information of the terminal state at $\lambda = 1$ with respect to the initial condition du_I^k.

Solving the linear system in (5.59) can be carried out by using the "Generalized Minimal Residual Method" (GMRES method, [33]), which belongs to the class of iterative solvers based on an approximation of the solution du_I^k within the Krylov-space $\mathcal{K}_{N_K}(du, J_k)$ of some given dimension N_K, defined by:

$$\mathcal{K}_{N_K}(du, J_k) := \text{span}\{du, J_k[du], J_k^2[du], \ldots, J_k^{N_K}[du]\}, \tag{5.60}$$

for the predefined initial Krylov-iterate $du := -D^{k+1/2}(u_I^k)$. The GMRES method iteratively constructs the Krylov-space $\mathcal{K}_{N_K}(du, J_k)$, whereas in each GMRES iteration one evaluation of $J_k[\delta u]$ for each GMRES iterate, say δu, needs to be performed. When no analytical expression for J_k is available, or the system size is too large to effectively compute J_k, one often relies on matrix-free approaches, which avoid the explicit calculation of the Jacobian J_k [6, 9]. One possibility is to approximate

$J_k[\delta u]$ by Finite-Differences [6, 9], e.g.,

$$J_k[\delta u] \approx \frac{D^{k+1/2}(u_I^k + \epsilon\delta u) - D^{k+1/2}(u_I^k)}{\epsilon}, \tag{5.61}$$

for some very small $\epsilon > 0$. However, the quality of the approximation strongly depends on the choice of ϵ. Furthermore, it requires to solve the scaled Navier-Stokes equations in their nonlinear form with respect to the initial conditions $u_I^k + \epsilon\delta u$ and u_I^k. As the following section will show, this is unnecessary since solving the linearized scaled Navier-Stokes equations will be sufficient, which also provide an exact evaluation of $J_k[\delta u]$.

5.3.3. Sensitivity of the terminal state with respect to the initial condition

The update formula given in (5.58) requires the solution of (5.59). As stated in the previous section, the GMRES method provides an iterative solution approach, whereas for some given iterate, say δu, the "effect" of the Jacobian $J_k[\delta u]$ has to be evaluated. This can be carried out by solving the linearized Navier-Stokes equations, which will be elaborated on in the following. This plays a very important role for the reduction of the overall numerical cost involved within the algorithm, since the solution of each Newton step (5.59) requires multiple solutions within the GMRES method such that there is a huge benefit in solving "only" linear unsteady problems.

Since the "implicit function theorem" plays an important role for proving the sufficiency of the linear equations, it shall be recapitulated, whereas its exact formulation is taken from [49] without proof:

Theorem 5.3 (Implicit Function Theorem)**.** *Let X, Y, and Z be Banach spaces over $\mathbb{R}$, and let*

$$F : U(u_I^*, u^*) \subset X \times Y \to Z \tag{5.62}$$

be a C^n-map on an open neighborhood of the point (u_I^*, u^*) such that $F(u_I^*, u^*) = 0$ holds and $1 \leq n \leq \infty$. Suppose that the operator (partial Fréchet differential)

$$F_v(u_I^*, u^*) : Y \to Z \quad \textit{is bijective.} \tag{5.63}$$

Then the following statements hold true:

(i) There exist numbers $r > 0$ and $\rho > 0$ such that, for each given $u_I \in U$, the equation $F(u_I, u) = 0$ has a unique solution $u \in Y$ with $\|u - u^*\| \leq r$. Denote this solution by $u(u_I)$.

(ii) The function $u_I \mapsto u(u_I)$ is C^n on U. In particular, for the Fréchet differential $u'(u_I)$ it holds

$$u'(u_I) = -F_u(u_I, u(u_I))^{-1} F_{u_I}(u_I, u(u_I)) \quad \textit{for all } u_I \in U. \tag{5.64}$$

The application of the Jacobian J_k to some δu can further be simplified by:

$$J_k[\delta u] = \delta u - J_k^{NS}[\delta u], \tag{5.65}$$

by definition of $D^{k+1/2}$, whereas J_k^{NS} denotes the Jacobian of the terminal state $I(u_I^k, T^{k+1}, 1)$ in direction δu. Therefore, the focus is shifted towards the computation of $J_k^{NS}[\delta u]$, which for the following analysis will be denoted by $J[\delta u]$ for notational convenience. Furthermore, to simplify the derivation of the linear model, divergence free vector spaces for the velocity u arising from the S-SNSE are assumed, such that the pressure variable can be neglected. Also, the strong formulation of the S-SNSE is being considered, provided that the velocity variable fulfills the regularity requirements of a classical solution to the stochastic Navier-Stokes equations. In an analog yet more technical way, the results of this section can be transfered to the mixed-type variational formulation involving the pressure variable with less regularity requirements on the velocity u.

Hence, the Navier-Stokes operator F is defined by:

$$F(u_I, u) := \begin{bmatrix} \partial_\lambda u + T(u \cdot \nabla)u - \nu T \Delta u \\ u_{|\Gamma} - u^D \\ u(\lambda = 0) - u_I \end{bmatrix}, \tag{5.66}$$

such that $F(u_I, u) = 0$ represents the S-SNSE in their strong formulation subject to a Dirichlet-condition u^D. Furthermore, let $u_I^* := u_I^k$, then there exists a solution u^* such that $F(u_I^*, u^*) = 0$ (cf. Chapter 4). It can be shown that F is C^∞-differentiable [22] in a neighborhood of (u_I^*, u^*). The directional Gâteaux derivative of F with respect to u in direction $\bar{u}$ can be easily calculated and reads:

$$F_u(u_I, u)[\bar{u}] = \begin{bmatrix} \partial_\lambda \bar{u} + T(\bar{u} \cdot \nabla)u + T(u \cdot \nabla)\bar{u} - \nu T \Delta \bar{u} \\ \bar{u}_{|\Gamma} \\ \bar{u}(\lambda = 0) \end{bmatrix}. \tag{5.67}$$

Assuming that the partial Fréchet derivative F_u is bijective in a neighborhood of (u_I^*, u^*) allows the use of the implicit function theorem 5.3 and therefore it holds:

$$-F_u(u_I, u)u'(u_I) = F_{u_I}(u_I, u), \tag{5.68}$$

for all (u_I, u) in a neighborhood of (u_I^*, u^*). Note, that $u'(u_I)[\delta u] = J[\delta u]$ for a direction δu. Therefore inserting this relation into equation (5.68) at $(u_I, u) = (u_I^*, u^*)$ with respect to the direction δu yields:

$$-\begin{bmatrix} \partial_\lambda v + T(v \cdot \nabla)u^* + T(u^* \cdot \nabla)v - \nu T \Delta v \\ v_{|\Gamma} \\ v(\lambda = 0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\delta u \end{bmatrix}, \tag{5.69}$$

for $v := J[\delta u]$.

Summing up, the computation of the directional derivative $J[\delta u]$ of the terminal state $I(u_I^k, T^{k+1}, 1)$ with respect to u_I in direction δu does not require a solution of the nonlinear form of the S-SNSE. Instead, given an iterate u_I^k and its corresponding terminal state $I(u_I^k, T^{k+1}, 1)$, it suffices to solve a linearized version of the S-SNSE subject to homogeneous Dirichlet boundary conditions on the boundary $\Gamma \subset \partial\mathcal{D}$ and initial condition δu, with linearization around the trajectory of the velocity between the initial condition u_I^k and its terminal state $I(u_I^k, T^{k+1}, 1)$ as stated in equation (5.69). Note that this provides a significant reduction in the computational cost of the overall iteration procedure compared to a Finite-Difference approximation of the Jacobian (cf. Eq. (5.61)), especially since the computation of the Jacobian in direction δu needs to be performed for every GMRES iteration δu.

Algorithm 5.1 Computation of stochastic periodic orbits

1: Choose initial guesses u_I^0 and T^0
2: Choose tolerances $\epsilon > 0$ and $k_{max} \in \mathbb{N}$
3: $k \leftarrow 0$
4: Compute terminal condition $u_T^0 \leftarrow I(u_I^0, T^0, 1)$ (nonlinear stochastic NS)
5: $r^0 \leftarrow \|u_I^0 - u_T^0\| / \|u_I^0\|$
6: **while** $k < k_{max}$ **and** $r^k > \epsilon$ **do**
7: $k \leftarrow k + 1$
8: Compute global scaling $\bar{\lambda} \in \mathbb{R}$ (nonlinear deterministic NS)
9: Compute stochastic correction $d\lambda = \sum_{i=0}^{P^T} d\lambda_i \psi_i$ (nonlinear stochastic NS)
10: Update period modes $T_m^k \leftarrow \bar{\lambda} T_m^{k-1} + \sum_{i=0}^{P^T} \sum_{j=0}^{P^T} d\lambda_i T_j^{k-1} c_{ijm}$ $m = 0, \ldots, P^T$
11: Compute terminal condition $u_T^k \leftarrow I(u_I^{k-1}, T^k, 1)$ (nonlinear stochastic NS)
12: Solve Newton step $-J_k[du_I^k] \leftarrow u_I^{k-1} - u_T^k$ (linear stochastic NS)
13: Update initial condition $u_I^k \leftarrow u_I^{k-1} + du_I^k$
14: Compute terminal condition $u_T^k \leftarrow I(u_I^k, T^k, 1)$ (nonlinear stochastic NS)
15: $r^k \leftarrow \|u_I^k - u_T^k\| / \|u_I^k\|$
16: **end while**
17: Postprocessing

5.3.4. The algorithm

This section shortly recapitulates the algorithm introduced in the previous sections. As can be seen from Algorithm 5.1, the computation of a periodic orbit starts by introducing initial guesses u_I^0 and T^0 for the initial condition and period, as well as the maximum number of iterations k_{max} and an error tolerance $\epsilon > 0$ for the relative distance error between a current iterate u_I^k and its corresponding terminal state $I(u_I^k, T^k, 1)$ with respect to the period iterate T^k within the iteration k.

The bulk of the computational cost is certainly the requirement of solving the stochastic Navier-Stokes equations and its linear and deterministic counterparts multiple times within each iteration, i.e., one iteration requires the solution of three nonlinear stochastic Navier-Stokes problems, one linearized stochastic Navier-Stokes problem and one deterministic nonlinear Navier-Stokes problem. Therefore, the numerical efficiency of this algorithm strongly depends on the numerical efficiency on the available numerical solvers for the stochastic Navier-Stokes equations. Especially, in context of high Reynolds number flows, a numerically stable Finite-Element discretization along with the stochastic Galerkin projection requires a large number of degrees of freedom to capture all dynamics of the flow. This necessitates the development and use of efficient parallel numerical solvers, employing memory distribution and thread parallelization.

5.4. Flow around a circular domain

The algorithm described in the previous sections shall be verified employing a benchmark problem originally introduced in [34]. It is a two-dimensional prob-

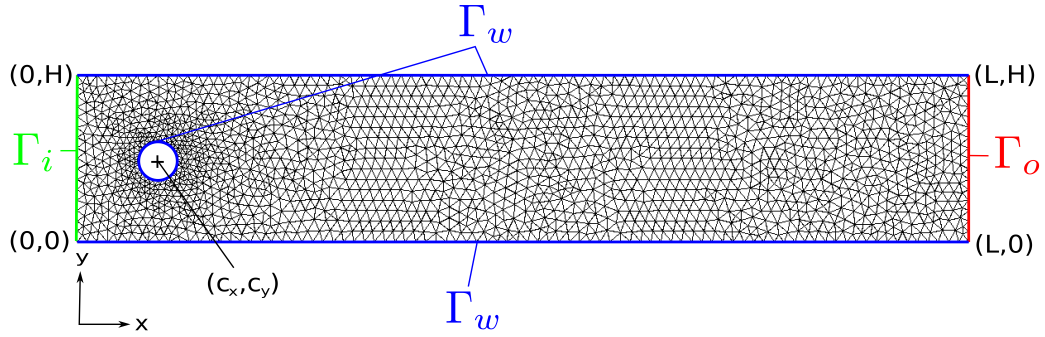


Figure 5.2.: Employed geometry and triangulation of a 2d-channel with a circular domain. Mesh refinement level $RL = 0$.

lem in space, which describes a flow of an incompressible fluid around a circular domain within a channel of length L and height H. Its exact geometrical data and a Finite-Element mesh consisting of triangles for a mesh refinement level $RL = 0$ is depicted in Fig. 5.2 with $L = 2.2$, $H = 0.41$. The circular domain has a diameter of $D = 0.1$ and its center-point has the coordinates $(c_x, c_y) = (0.2, 0.2)$. For the deterministic case it is known [34] that the problem exhibits a time-periodic solution for a Reynolds number $Re \approx 100$ resulting in a periodic vortex shedding scheme behind the circular domain (the flow is considered from left to right, cf. Fig. 5.2), also known as "Kármán vortex street" [36]. This suggests a similar time-periodic behavior of the stochastic Navier-Stokes equations making it a suitable benchmark problem for the algorithm.

No-slip boundary conditions are considered for $\Gamma_w \subset \partial\Omega = (0, L) \times (0, H)$. The inflow boundary condition at $\Gamma_i \subset \partial\Omega$ is set to be a stochastic parabolic profile, i.e.,

$$u_1(0, y, t; \xi) = 4u_1^{\text{inflow}}(\xi) y \frac{(H - y)}{H^2}, \quad y \in [0, H],\ t \geq 0, \tag{5.70}$$

$$u_2(0, y, t; \xi) = u_2^{\text{inflow}}(\xi), \qquad y \in [0, H],\ t \geq 0, \tag{5.71}$$

with $u = [u_1, u_2]$ denoting the components in x- and y-directions of the velocity u, respectively. In the following sections, a one-dimensional and a two-dimensional uniformly distributed random input ξ will be considered, whereas $u_1^{\text{inflow}}(\xi)$ and $u_2^{\text{inflow}}(\xi)$ will be defined later according to the numerical examples. For the outflow boundary condition at $\Gamma_o \subset \partial\Omega$ so-called "do-nothing" boundary conditions are applied [21]. These represent natural boundary conditions arising from the weak formulation of the SNSE by requiring all boundary integrals at Γ_o to vanish in their sum, i.e.,

$$\int_{\Gamma_o} \nabla u_i \cdot \vec{n} - p_i \vec{n}\, dx = 0, \quad i = 0, \ldots, P, \tag{5.72}$$

whereas u_i and p_i, $i = 0, \ldots, P$ denote the stochastic modes of the velocity and pressure variable, respectively and $\vec{n}$ denotes the outward unit normal vector on the boundary Γ_o. Since this boundary condition results in a unique pressure variable, no additional requirements, such as $\int_\Omega p\, dx = 0$, are necessary.

Furthermore, the kinematic viscosity ν is set to $\nu = 0.001$ and the density ρ is set to $\rho \equiv 1$ for all benchmark computations. For time integration a Crank-Nicolson scheme (cf. Section 4.2.1) with a homogeneous time step size $\Delta t = 0.01$ for the unscaled time variable t is employed. The according time step length $\delta\lambda$ for the scaled time variable λ is defined by:

$$\delta\lambda := \frac{\Delta t}{T^k(\bar{\xi})}, \tag{5.73}$$

with respect to a period iterate T^k at iteration k and the mean $\bar{\xi}$ of the random input ξ. Therefore, the time step size can vary for each iteration k of the algorithm, depending on the value $T^k(\bar{\xi})$ of the period iterate T^k at $\bar{\xi}$. The spatial variable is discretized employing the Finite-Element mesh depicted in Fig. 5.2 and Taylor-Hood elements of polynomial degree equal to 2 for each stochastic mode of the velocity variable and a corresponding polynomial degree equal to 1 for each pressure mode (cf. Section 4.2.3). Furthermore, the computations are carried out using the Finite-Element software *HiFlow*^{3} [18].

5.4.1. One-dimensional random input

In this section a one-dimensional random input $\xi \sim U(-1, 1)$, uniformly distributed in the interval $(-1, 1)$, is being considered. The random quantities $u_1^{\text{inflow}}(\xi)$ and $u_2^{\text{inflow}}(\xi)$ for the inflow boundary conditions (5.70) and (5.71) are set to:

$$u_1^{\text{inflow}}(\xi) := 1.5 + 0.1\xi, \tag{5.74}$$

$$u_2^{\text{inflow}}(\xi) := 0. \tag{5.75}$$

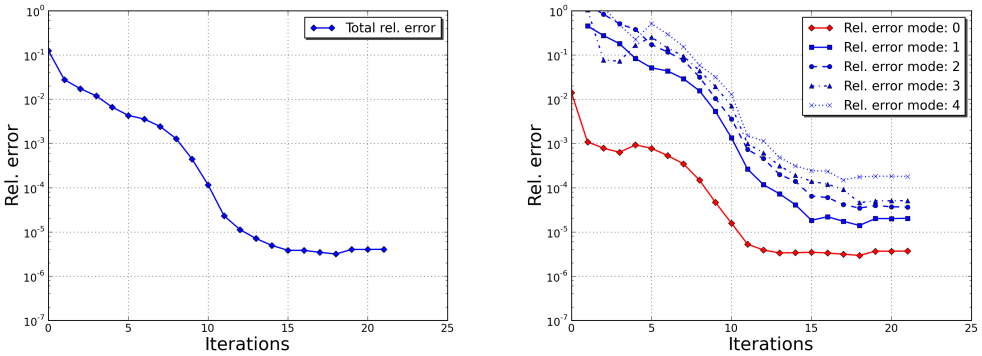
This results in a uniformly distributed Reynolds number $Re \sim U(93.\bar{3}, 106.\bar{6})$, whereas the Reynolds number is calculated by:

$$Re(\xi) = \frac{2}{3}\frac{u_1^{\text{inflow}}(\xi)D}{\nu}. \tag{5.76}$$

Therefore, a time-periodic solution of the SNSE can be expected. The numerical computations are carried out on the Finite-Element mesh depicted in Fig. 5.2 employing one larger mesh refinement level, i.e., $RL = 1$. The convergence of the iterative algorithm is measured by monitoring the total and mode-wise relative errors, ϵ_{total} and $\epsilon_{\text{mode},i}$, $i = 0, \ldots, P$, respectively. These measure the errors between the initial condition for the velocity and its corresponding terminal state with respect to the current period iterate and are defined by:

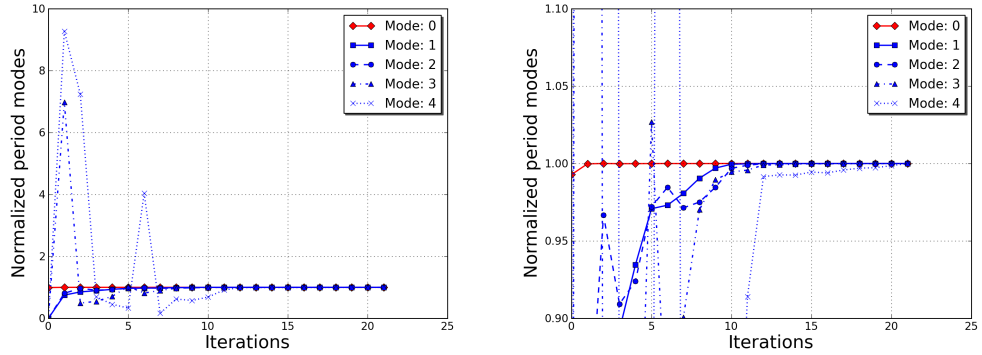
$$\epsilon_{\text{total}} := \sqrt{\left(\frac{\mathbb{E}\|u_I^k - I(u_I^k, T^k, 1)\|^2_{L^2(\mathcal{D})}}{\mathbb{E}\|u_I^k\|^2_{L^2(\mathcal{D})}}\right)}, \tag{5.77}$$

$$= \sqrt{\left(\frac{\sum_{i=0}^P \|u_{i,I}^k - I_i(u_I^k, T^k, 1)\|^2_{L^2(\mathcal{D})}\langle\psi_i, \psi_i\rangle}{\sum_{i=0}^P \|u_{i,I}^P\|^2_{L^2(\mathcal{D})}\langle\psi_i, \psi_i\rangle}\right)}, \tag{5.78}$$



(a) Total rel. error. (b) Mode-wise rel. error.

Figure 5.3.: Total and mode-wise relative error developments with respect to the number of iterations.



(a) Absolute view. (b) Zoom.

Figure 5.4.: Evolution of the normalized period modes with respect to the number of iterations.

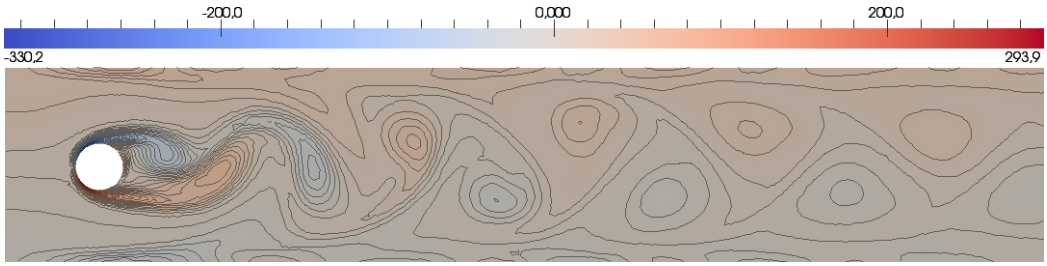
and

$$\epsilon_{\text{mode},i} = \sqrt{\left(\frac{\|u_{i,I}^k - I_i(u_I^k, T^k, 1)\|_{L^2(\mathcal{D})}^2}{\|u_{i,I}^k\|_{L^2(\mathcal{D})}^2} \right)}, \tag{5.79}$$

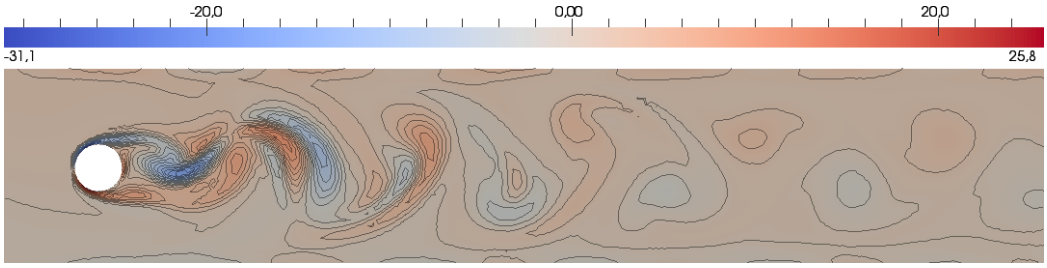
for $i = 0, \ldots, P$, both error terms are subject to some iteration index k. Note that $\{u_{i,I}^k\}_{i=0}^P$ and $\{I_i(u_I^k, T^k, 1)\}_{i=0}^P$ denote the modes of the corresponding gPC expansion of u_I and $I(u_I^k, T^k, 1)$ given by:

$$u_I^k = \sum_{i=0}^{P} u_{i,I}^k \psi_i, \tag{5.80}$$

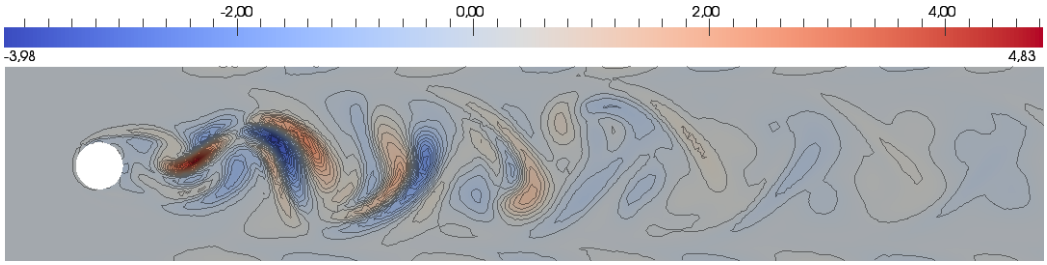
$$I(u_I^k, T^k, 1) = \sum_{i=0}^{P} I_i(u_I^k, T^k, 1) \psi_i. \tag{5.81}$$



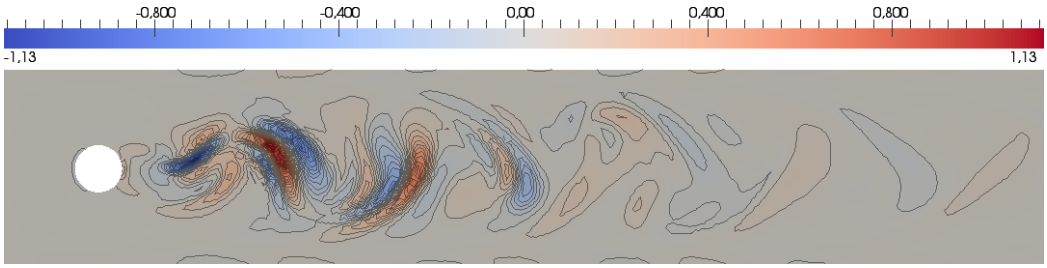
(a) Vorticity mode: 0. Mean.



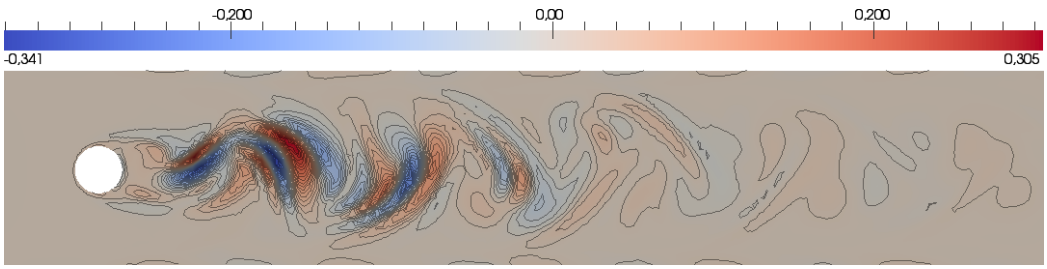
(b) Vorticity mode: 1. Linear influence of ξ.



(c) Vorticity mode: 2. Quadratic influence of ξ.



(d) Vorticity mode: 3. Cubic influence of ξ.



(e) Vorticity mode: 4. Fourth order influence of ξ.

Figure 5.5.: Computed vorticity of the velocity modes of the initial condition at iteration index $k = 21$.

For the computations an initial guess at iteration index $k = 0$ for the initial condition $u_I^{k=0}$ and period $T^{k=0}$ has to be defined. For the initial condition $u_I^{k=0}$ a fully developed deterministic flow computed by a standard deterministic unsteady Navier-Stokes solver has been taken as an initial guess, whereas all modes greater or equal to index 1 are set to zero. For the period $T^{k=0}$ the initial guess has been taken as $T_0^{k=0} = 0.33$ for the mean of the period and zero otherwise.

Fig. 5.3 depicts the evolution of the total and mode-wise relative errors with respect to the iterations of the algorithm for a gPC expansion order $No^T = No = 4$ leading to $P+1 = 5$ modes for the velocity, pressure and period variables. Furthermore, Fig. 5.4 presents the results concerning the evolution of the period modes, which have been normalized with respect to their value at iteration index $k = 21$ to illustrate their convergence behavior more clearly. Note, that the computed period modes at iteration index $k = 21$ are given by:

$$T_0^{k=21} = \quad 3.32 \cdot 10^{-1}, \tag{5.82}$$
$$T_1^{k=21} = -2.44 \cdot 10^{-2}, \tag{5.83}$$
$$T_2^{k=21} = \quad 1.21 \cdot 10^{-3}, \tag{5.84}$$
$$T_3^{k=21} = -5.72 \cdot 10^{-5}, \tag{5.85}$$
$$T_4^{k=21} = \quad 1.76 \cdot 10^{-6}. \tag{5.86}$$

In Fig. 5.3 a decrease in 5 orders of magnitude, both for the total and mode-wise relative errors, with respect to the errors introduced by the initial guess can be observed within 15 iterations. However, the total relative error begins to stagnate for iteration numbers $k > 15$, which is due to the fixed time step size $\Delta t = 0.01$ for the physical time variable t and the fixed gPC expansion orders No^T and No.

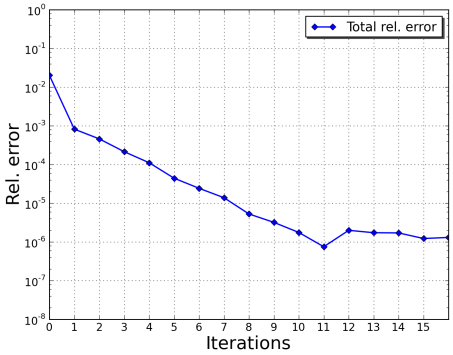
It can be expected, that a higher time discretization resolution along with a larger gPC expansion order are able to decrease the error levels even further. However, concerning the period solution at iteration index $k = 21$, it seems that the expansion order No^T for the period is large enough, due to the decreasing magnitude of the period modes.

In Fig. 5.5 the vorticity of each mode of the initial condition at iteration index $k = 21$ is depicted. Thereby, the vorticity $\mathrm{vort}(u)$ for a two-dimensional velocity field $u = [u_1, u_2]$ is defined by:

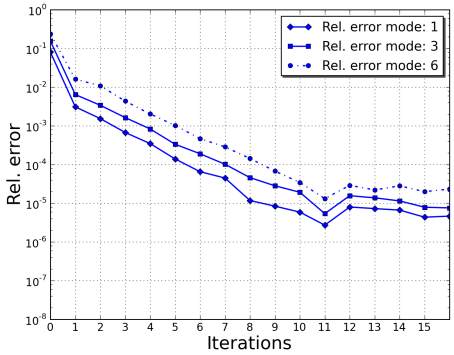
$$\mathrm{vort}(u) := \frac{\partial u_2}{\partial x} - \frac{\partial u_1}{\partial y}, \tag{5.87}$$

which does reflect the vortex shedding scheme of every mode of the initial condition more clearly. In addition the magnitudes of the vorticity of each mode exhibit a decrease, which reflects the convergence property of the gPC expansion of the velocity field. Note that the computation of the vorticity is carried out by a post processing step, since it is a derived quantity from the velocity field.

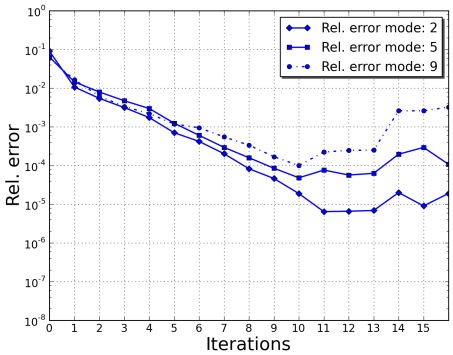
Further numerical simulations have been computed subject to different initial guesses for the iterative algorithm for lower inflow variability levels. Thereby, it was observed that the results exhibit quite varying behaviors, including steady state and periodic solutions similar to the analyzed case within this section. Hence, the algorithm strongly depends on the initial guess such that further constraints



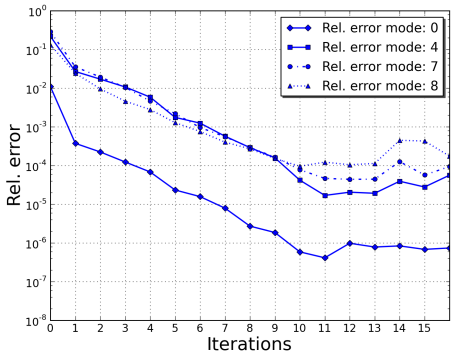
(a) Total rel. error.



(b) Mode-wise rel. error depending on ξ_1 only.



(c) Mode-wise rel. error depending on ξ_2 only.



(d) Mode-wise rel. error depending on mixed forms of ξ_1 and ξ_2.

Figure 5.6.: Total and mode-wise relative error developments with respect to the iterations. Plots show results concerning modes according to dependencies on ξ_1, ξ_2 only and on mixed forms of $\xi_1\xi_2$.

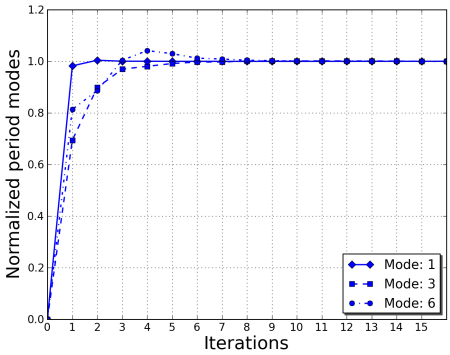
need to be prescribed a priori, e.g., by minimizing the norm of the initial condition reducing the number of possible solutions. These aspects, however, will be addressed in future work.

5.4.2. Two-dimensional random input

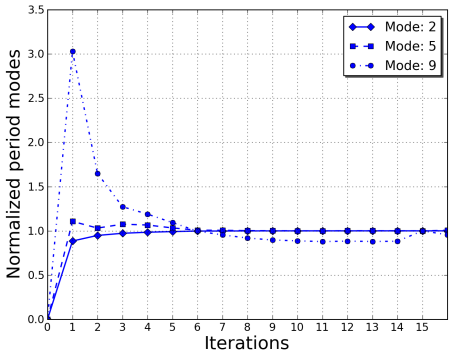
Next, the one-dimensional random input defined in the previous section is extended to a two-dimensional random vector $\xi = [\xi_1, \xi_2]$ to illustrate the convergence behavior of the algorithm in context of multi-dimensional random input. Therefore, to keep the necessary gPC expansion order low, only a slight variation on the x- and y-component of the velocity profile inflow boundary condition is introduced, such that:

$$u_1^{\text{inflow}} := 1.5 + 0.01\xi_1, \tag{5.88}$$

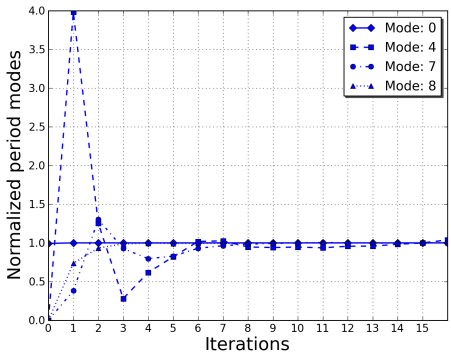
$$u_2^{\text{inflow}} := 0.01\xi_2. \tag{5.89}$$



(a) Modes according to ξ_1 only.



(b) Modes according to ξ_2 only.



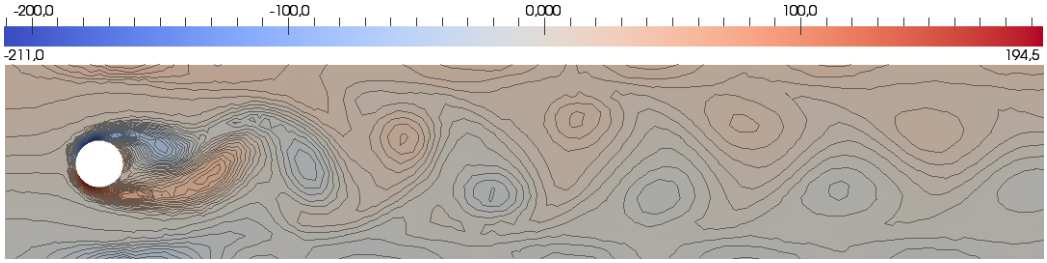
(c) Modes according to mixed forms of ξ_1 and ξ_2.

Figure 5.7.: Evolution of the normalized period modes with respect to the iterations. Plots show results concerning modes according to dependencies on ξ_1, ξ_2 only and on mixed forms of $\xi_1\xi_2$.

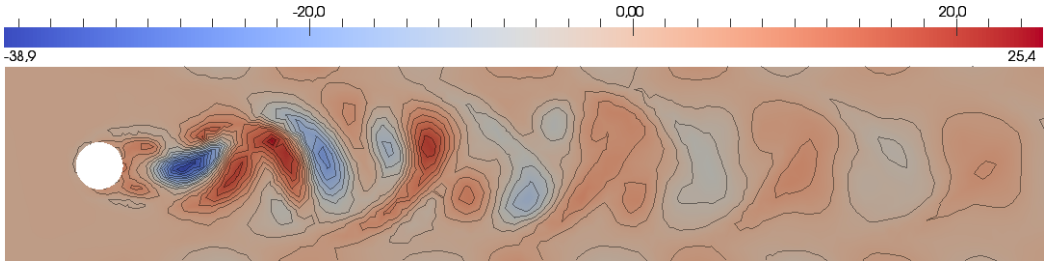
Here, ξ_1 and ξ_2 are assumed to be independent and uniformly distributed within the interval $(-1, 1)$.

For the numerical computations a gPC expansion order $No^T = No = 3$ is employed resulting in $P + 1 = 10$ number of modes for the velocity, pressure and period variables. Furthermore, the numerical computations are carried out on the Finite-Element mesh depicted in Fig. 5.2. The total and mode-wise relative errors are measured in the same way as for the one-dimensional case and are depicted in Fig. 5.6. For this setting, a fully developed stochastic flow computed by a standard gPC solver of the unsteady equations has been taken as an initial guess for the initial condition. The initial guess for the period is the same as in the previous section and given by $T_0^{k=0} = 0.33$ and zero for all higher order modes.

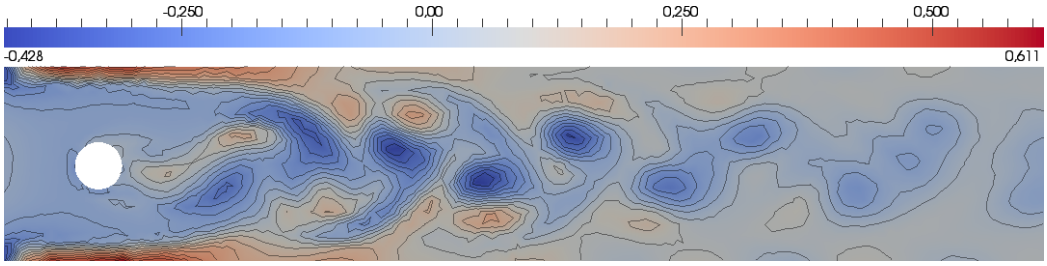
The evolution of the relative errors exhibit an exponential convergence rate. Thereby, the error levels begin to stagnate after $k > 10$ number of iterations similar to the one-dimensional case, due to the restriction given by the time-discretization step length Δt and the number of terms $P + 1$ employed in the gPC expansion. Furthermore, it can be seen in Fig. 5.7 that the evolution of the period modes



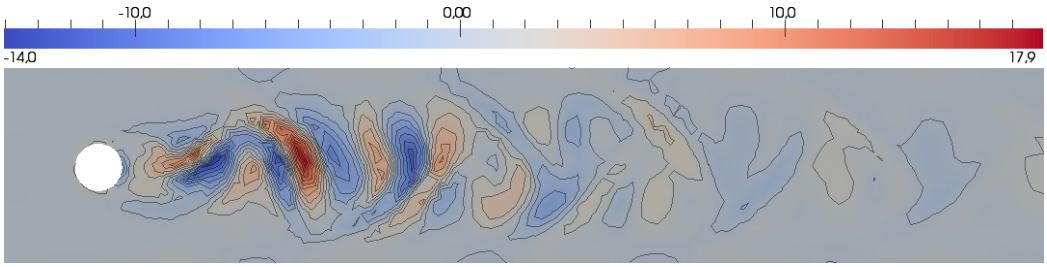
(a) Vorticity mode: 0. Mean.



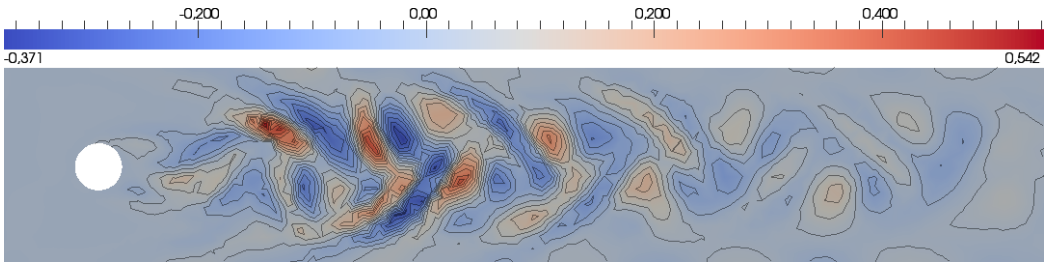
(b) Vorticity mode: 1. Single influence by ξ_1 only (linear).



(c) Vorticity mode: 2. Single influence by ξ_2 only (linear).

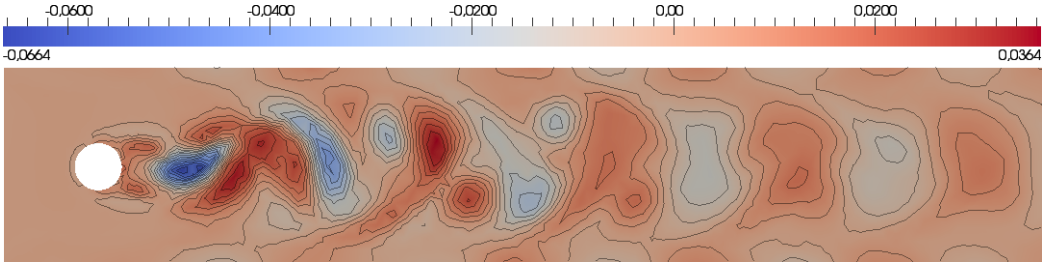


(d) Vorticity mode: 3. Single influence by ξ_1 only (quadratic).

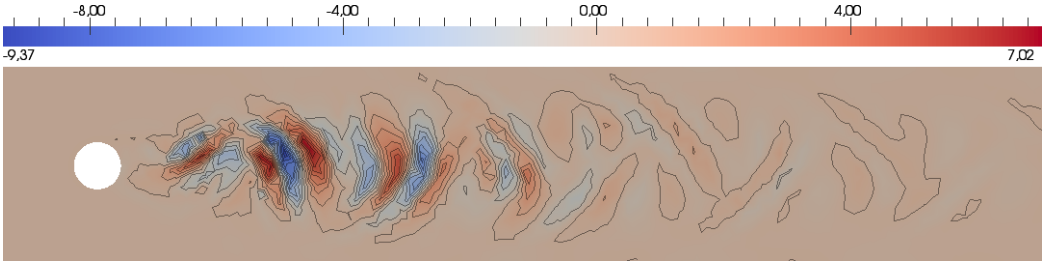


(e) Vorticity mode: 4. Mixed influence by ξ_1 and ξ_2 (both linear).

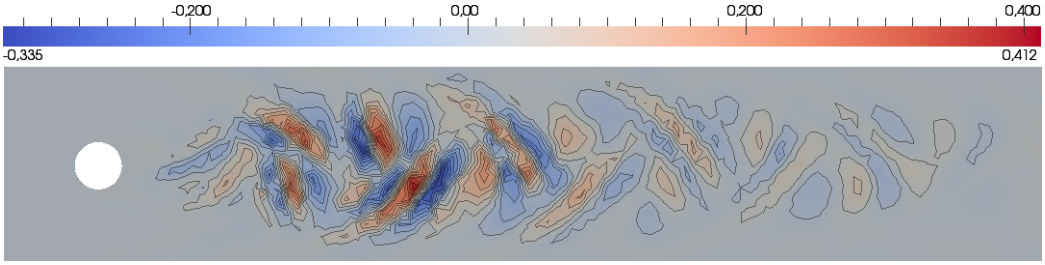
Figure 5.8.: Computed vorticity of the velocity modes of the initial condition. Modes 0 to 4 at iteration index $k = 15$.



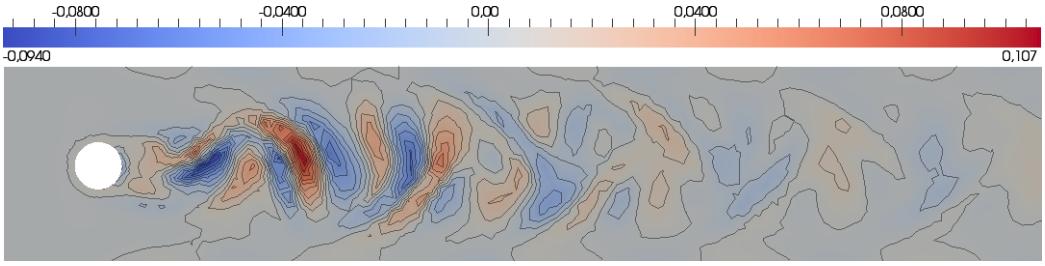
(a) Vorticity mode: 5. Single influence by ξ_2 (quadratic).



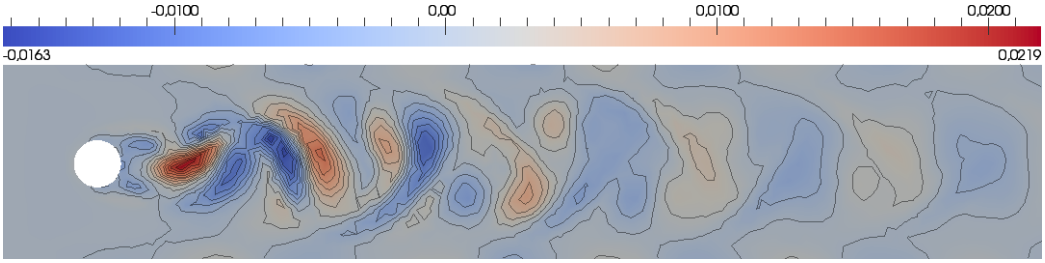
(b) Vorticity mode: 6. Single influence by ξ_1 (cubic).



(c) Vorticity mode: 7. Mixed influence by ξ_1 and ξ_2 (quadratic in ξ_1, linear in ξ_2).



(d) Vorticity mode: 8. Mixed influence by ξ_1 and ξ_2 (linear in ξ_1, quadratic in ξ_2)



(e) Vorticity mode: 9. Single influence by ξ_2 (cubic).

Figure 5.9.: Computed vorticity of the velocity modes of the initial condition. Modes 5 to 9 at iteration index $k = 15$.

depending on ξ_2 is subject to larger variations for the first few iterates compared to the modes only depending on ξ_1, however, all modes eventually stabilize after $k > 7$ number of iterations. Note that as in the previous section the period modes depicted in Fig. 5.7 are normalized with respect to their values at iteration index $k = 15$, which are given by:

$$T_0^{k=15} = \quad 3.32 \cdot 10^{-1}, \quad T_5^{k=15} \quad = 3.06 \cdot 10^{-6} \tag{5.90}$$

$$T_1^{k=15} = -2.42 \cdot 10^{-3}, \quad T_6^{k=15} = -2.02 \cdot 10^{-4} \tag{5.91}$$

$$T_2^{k=15} = \quad 9.73 \cdot 10^{-5}, \quad T_7^{k=15} = -1.43 \cdot 10^{-6} \tag{5.92}$$

$$T_3^{k=15} = -1.53 \cdot 10^{-4}, \quad T_8^{k=15} = -6.44 \cdot 10^{-6} \tag{5.93}$$

$$T_4^{k=15} = -3.23 \cdot 10^{-7}, \quad T_9^{k=15} = -1.83 \cdot 10^{-7}. \tag{5.94}$$

Furthermore, all modes of the initial condition (cf. Fig. 5.8 and Fig. 5.9) exhibit a time-periodic vortex shedding scheme behind the circular domain, which is in good agreement with the results regarding a deterministic reference problem.

In summery, both the one- and the two-dimensional case are able to demonstrate the convergence of the algorithm for different types of random input, which is the focus of this section. Future work will address the numerical results related to the approximation quality of limit-cycles for various realizations of the random input and provide a comparison between sampling and projection approaches for calculating the corresponding trajectories. Furthermore, stabilization techniques for the Newton increments by an optimal control framework will be analyzed.

Chapter 6.

Conclusions

Generalized Polynomial Chaos provides an efficient technique of representing square-integrable stochastic processes or fields by means of a spectral projection. However, it also suffers from a convergence breakdown when applied to problems involving nonlinear dependencies of a high degree on some random input, which, for example, can arise from long term integration or stochastic discontinuities. Therefore, this work is divided into two parts, whereas the first one is devoted to the application of gPC towards stochastic ordinary differential equations and the second one towards stochastic partial differential equations represented by the unsteady stochastic incompressible Navier-Stokes equations.

Within the first part of this work a new hybrid gPC approach is introduced, which relies on a local variant of a modified version of the "time-dependent generalized Polynomial Chaos" (TD-gPC), originally introduced by Gerritsma et al. [14]. Thereby, the mentioned modifications refer to an extension of TD-gPC involving an orthogonalized tensor product of two different types of basis functionals given by the standard gPC basis and the basis functionals computed by a transformation of the probability measure within TD-gPC. Hence, the modified variant is able to capture the evolution in time of the probability distribution of the solution itself and its time derivative, therefore significantly improving the convergence behavior of the standard gPC. This has been verified successfully on a system of stochastic ordinary differential equations given by a linear oscillator exhibiting stable limit-cycles. However, the modified TD-gPC variant introduces additional numerical cost, which is mainly related to a significantly increased system size arising from the number of terms within the expansion. To this end, a new local approach has been developed, which decomposes the underlying probability space into a finite number of stochastic independent elements in which the modified TD-gPC variant can be applied using a lower expansion order to achieve the same accuracies as compared to the global approach. This results in a trade-off between the number of elements and the expansion order employed allowing for a reduction of the numerical complexity if the independent problems are computed in parallel. The local TD-gPC has been numerically verified on two benchmark problems given by a simple scalar evolution equation and the more challenging Kraichnan-Orszag three mode problem.

However, there are still open questions regarding TD-gPC in general. The main issue is related to the transformation of the probability measure, which is based on a re-orthogonalization of the basis functionals. Thereby, the complexity is shifted towards the numerical evaluation of the inner products within the space of square integrable functionals on the probability space. In this work a Gaussian quadra-

ture rule has been applied to this purpose subject to a large number of quadrature points to maintain an accurate numerical integration, which introduces a significant increase in the numerical costs. Therefore, this topic needs further study to develop robust modifications of TD-gPC which require less basis transformations resulting in less evaluations of inner products or which utilize other numerical integration schemes. Nonetheless, TD-gPC along with its variants still provides a powerful method to solve problems involving strong nonlinearities with a high accuracy, for which the standard gPC approach will fail. Furthermore, future work will address the extension of TD-gPC to the class of stochastic partial differential equations which introduce additional complexity due to the dependency on a spatial variable.

The second part of this work is devoted to the application of the standard gPC to the unsteady stochastic incompressible Navier-Stokes equations (SNSE). The focus is on computing stochastic time-periodic solutions to the SNSE by means of a Galerkin projected system with respect to a gPC expansion of the random quantities. Therefore, a new iterative algorithm has been developed based on a deterministic version originally introduced by Duguet et al. in [9] employing optimization techniques as well as Newton's method. The algorithm is transfers the period variable from the time integration domain into the governing equations, resulting in a deterministic range for the time variable of the unsteady problem. To be able to compute a period iterate, a predictor-corrector step first computes an estimate based on a deterministic reference problem and afterwards corrects the error associated with the prediction for almost every stochastic realization. This minimizes the error between the initial condition of the SNSE and its corresponding terminal state. To be able to decrease the errors further, an inexact Newton method is applied to compute a new iterate for the initial condition. These steps are repeated until convergence is achieved, which is monitored by the distance between the initial condition and its terminal state.

The algorithm has been verified on a stochastic benchmark problem describing the fluid flow around a circular domain in two space dimensions. Therefore, a one-dimensional and a two-dimensional uniformly distributed random input introduced within the inflow boundary conditions have been considered. Both cases exhibit an exponential convergence rate of the relative errors with respect to the number of iterations of the algorithm. However, the accuracy still depends on the employed time discretization step size and the number of terms within the gPC expansion, such that the errors begin to stagnate after a certain number of iterations. Furthermore, the computation of a solution strongly depends on the initial guess of the initial condition and period such that further constraints need to be prescribed a priori which reduce the number of possible solutions. This can be achieved, for example, by requiring the norm of the initial condition to be minimal. Furthermore, the experience showed that the choice of forcing terms within the employed inexact Newton method plays a crucial role regarding the convergence behavior of the algorithm. Especially for high Reynolds number flows adequate numerical stabilization techniques need to be developed to increase the robustness of the underlying inexact Newton method, e.g., by adaption of so-called "globalization techniques" [9] to the stochastic context. Future work will address these open questions in detail and extend the algorithm to other problem classes involving time-periodic solutions.

Appendix A.

Inexact Newton method

The inexact Newton method (INM) is an extension of the classical Newton's method (NM), allowing each Newton iteration to solve the corresponding linear problem in a numerically inexact way [6]. Thereby, the approximation quality of each linear problem can be prescribed by relative error bounds in context of iterative linear solvers, such as the "Generalized Minimal Residual Method" (GMRES method) [33]. For illustration of the INM consider a continuously differentiable function $F : \mathbb{R}^n \to \mathbb{R}^n$ in a neighborhood of some $x^* \in \mathbb{R}^n$, for which it holds: $F(x^*) = 0$ and $F'(x^*) = \frac{d}{dx}F(x^*)$ is regular. Furthermore, assume that F' is Lipschitz-continuous at x^* with a constant $L > 0$, i.e.,

$$\|F'(x) - F'(x^*)\| \leq L\|x - x^*\|, \tag{A.1}$$

for every $x \in U_\epsilon(x^*) \subset \mathbb{R}^n$, whereas $U_\epsilon(x^*)$ denotes the ball with radius $\epsilon > 0$ and center point x^* in $\mathbb{R}^n$. The classical NM to determine a zero x^* of F is described in Algorithm A.1. It is well known that the convergence properties of the classical NM strongly depend on the choice of the initial guess x_0 [6].

The main computational cost involved within Algorithm A.1 is due to the solution of a linear system of equations in every Newton iteration (A.2). Especially for high dimensions n this constitutes a very challenging task. To reduce the numerical costs, Eisenstat and Walker [10] proposed an INM using so-called "forcing-terms", which determine the necessary approximation quality of a solution in each Newton iteration, such that convergence can be maintained. The algorithm for the INM is stated in Algorithm A.2. Employing an iterative linear solver in each inexact Newton step (A.3) will require less iterations compared to the same linear solver in a standard Newton step (A.2), since the forcing terms η_k, $k = 0, 1, 2, \ldots$, allow the relative error for an update value dx_k within (A.3) to be greater then zero in contrast to (A.2) leading to a reduction of the numerical cost involved in solving the linear system. However, this can result in a larger number of Newton iterations, since the quadratic convergence property of the classical NM in a neighborhood of the solution x^* can be violated [6]. Hence, there is a trade-off between undersolving the linear system and the additional required Newton iterations needed to solve the nonlinear problem.

A.1. Forcing terms

In [10] Eisenstat and Walker introduced two choices to determine the forcing terms η_k, $k = 0, 1, 2, \ldots$, within the INM. The first choice is given by:

Algorithm A.1 Classical Newton's method

1: Choose some initial guess $x_0 \in U_\epsilon(x^*)$
2: **for** $k = 0$ until convergence **do**
3: Solve

$$-F'(x_k)dx_k = F(x_k) \tag{A.2}$$

4: Update $x_{k+1} \leftarrow x_k + dx_k$
5: **end for**

Algorithm A.2 Inexact Newton method

1: Choose some initial guess $x_0 \in U_\epsilon(x^*)$
2: **for** $k = 0$ until convergence **do**
3: Find some $\eta_k \in [0, 1)$ and dx_k such that

$$\|F(x_k) + F'(x_k)dx_k\| \leq \eta_k \|F(x_k)\| \tag{A.3}$$

4: Update $x_{k+1} \leftarrow x_k + dx_k$
5: **end for**

Choice 1: Given $\eta_0 \in [0, 1)$, choose

$$\eta_k = \frac{\|F(x_k) - F(x_{k-1}) - F'(x_{k-1})dx_{k-1}\|}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots, \tag{A.4}$$

or

$$\eta_k = \frac{|\|F(x_k)\| - \|F(x_{k-1}) + F'(x_{k-1})dx_{k-1}\||}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots. \tag{A.5}$$

The second choice is given by:

Choice 2: Given $\gamma \in [0, 1]$, $\alpha \in (1, 2]$ and $\eta_0 \in [0, 1)$, choose

$$\eta_k = \gamma \left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|} \right)^\alpha, \quad k = 1, 2, \ldots. \tag{A.6}$$

The main difference between both choices lies in the fact that Choice 1 represents the agreement of the nonlinear model with respect to its local linear model, which is not the case for Choice 2. However, monitoring this agreement is not a necessary condition for achieving convergence of the INM, instead the optimal choice is certainly application dependent. Furthermore, Choice 2 requires the a priori definition of two parameters, namely γ and α, which have a strong impact on the convergence behavior of the INM [10].

In practice [10], it was observed that the forcing terms occasionally can become too small far away from the solution, e.g., by a coincidental good agreement between F and its local linear model in Choice 1. Therefore, it is possible to introduce additional safeguards, which prevent the forcing terms to become too small too quickly by monitoring the relation between η_k and η_{k-1}. The safeguards are defined by [10]:

- Choice 1 safeguard:
  Modify η_k by $\eta_k \leftarrow \max\{\eta_k, \eta_{k-1}^{(1+\sqrt{5})/2}\}$ whenever $\eta_{k-1}^{(1+\sqrt{5})/2} > 0.1$.

Algorithm A.3 Inexact Newton backtracking method

1: Choose some initial guess $x_0 \in U_\epsilon(x^*)$
2: Choose $t \in (0,1)$, $\eta_{\max}$ and $0 < \theta_{\min} < \theta_{\max} < 1$
3: **for** $k = 0$ until convergence **do**
4: Choose an initial $\eta_k \in [0, \eta_{\max}]$ and dx_k such that

$$\|F(x_k) + F'(x_k)dx_k\| \leq \eta_k \|F(x_k)\| \tag{A.7}$$

5: **while** $\|F(x_k + dx_k)\| > [1 - t(1 - \eta_k)]\|F(x_k)\|$ **do**
6: Choose $\theta \in [\theta_{\min}, \theta_{\max}]$
7: Update $dx_k \leftarrow \theta dx_k$ and $\eta_k \leftarrow 1 - \theta(1 - \eta_k)$
8: **end while**
9: Update $x_{k+1} \leftarrow x_k + dx_k$
10: **end for**

- Choice 2 safeguard:
  Modify η_k by $\eta_k \leftarrow \max\{\eta_k, \gamma\eta_{k-1}^\alpha\}$ whenever $\gamma\eta_{k-1}^\alpha > 0.1$.

Note, that the value $(1 + \sqrt{5})/2$ refers to a convergence rate of the INM described in [10]. Furthermore, it should be ensured that the forcing terms remain within $(0, 1]$ by introducing the requirement $\eta_k < \eta_{\max} \leq 1$ for all $k = 0, 1, 2, \ldots$.

A.2. Backtracking

As being mentioned earlier, the classical NM as well as the INM only exhibit local convergence, i.e, the sequence of iterates x_k, $k = 0, 1, 2, \ldots$, converges to a solution x^* only if the initial guess x_0 is sufficiently close to x^*. In many practical computations, this can provide a huge restriction since determining such an appropriate initial guess might be very difficult or its determination might not even be possible at all. Therefore, in practice one often relies on so-called "globalization" techniques [6]. This does not mean that the Newton methods converge for every initial guess x_0, instead one aims for a significant enlargement of the set of appropriate initial guesses for which convergence can be achieved. One popular example is given by the so-called "backtracking" method [10], which is stated in Algorithm A.3. The classical NM as well as its inexact version might suggest correction steps dx_k within some iteration k, which cause the updated iterate x_{k+1} to "jump" out of the convergence domain of x^*, which can happen, for example, if the gradient $F'(x_k)$ is close to singular. In this case, the backtracking method allows the update correction dx_k to be damped by some scalar in order to achieve a numerical stabilization of the INM. The determination of such a damping factor (cf. Algorithm A.3, step 6) can be accomplished by different strategies. A natural choice is to determine some $\theta \in [\theta_{\min}, \theta_{\max}]$ for which $\theta \mapsto \|F(x_k + \theta dx_k)\|^2$ is minimal. However, this also introduces additional numerical cost, since multiple evaluations of the residual $F(x_k + dx_k)$ need to be performed, which can be numerically expensive themselves.

References

[1] V. A. Badri Narayanan and N. Zabaras. Variational multiscale stabilized fem formulations for transport equations: stochastic advection-diffusion and incompressible stochastic navier-stokes equations. *J. Comput. Phys.*, 202:94–133, January 2005.

[2] D. Braess. *Finite Elemente : Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer-Lehrbuch. Springer, Berlin, 3rd edition, 2003.

[3] S. C. Brenner and L. R. Scott. *The mathematical theory of finite element methods*. Texts in applied mathematics. Springer, New York, 2nd edition, 2002.

[4] R. Cameron and W. Martin. The orthogonal development of non-linear functionals in series of fourier-hermite functionals. *The Annals of Mathematics*, 48(2):385–392, April 1947.

[5] J. Crank and P. Nicolson. A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Advances in Computational Mathematics*, 6:207–226, 1996.

[6] P. Deuflhard. *Newton methods for nonlinear problems : affine invariance and adaptive algorithms*. Springer series in computational mathematics. Springer, Berlin, corr. 2. print. edition, 2006.

[7] P. Deuflhard and M. Weiser. *Numerische Mathematik*, volume 3: Adaptive Lösung partieller Differentialgleichungen. De Gruyter, Berlin, 2011.

[8] M. Doi and T. Imamura. The wiener-hermite expansion with time-dependent ideal random function. *Progress of Theoretical Physics*, 41(2):358–366, February 1969.

[9] Y. Duguet, C. C. T. Pringle, and R. R. Kerswell. Relative periodic orbits in transitional pipe flow. *Physics of fluids*, 20, 2008.

[10] S. C. Eisenstat and H. F. Walker. Choosing the forcing terms in an inexact newton method. *SIAM J. Sci. Comput*, 17:16–32, 1994.

[11] O. G. Ernst, A. Mugler, H.-J. Starkloff, and E. Ullmann. On the convergence of generalized polynomial chaos expansions. Preprint 60, DFG-SPP 1324, August 2010.

[12] G. P. Galdi. *An introduction to the mathematical theory of the Navier-Stokes equations*, volume 1: Linearised steady problems of *Springer tracts in natural philosophy ; 38*. Springer, New York, rev. ed., corr. 2. print. edition, 1998.

[13] G. P. Galdi. *An introduction to the mathematical theory of the Navier-Stokes equations*, volume 2: Nonlinear steady problems of *Springer tracts in natural philosophy ; 39*. Springer, New York, rev. ed., corr. 2. print. edition, 1998.

[14] M. Gerritsma, J.-B. van der Steen, P. Vos, and G. E. Karniadakis. Time-dependent generalized polynomial chaos. *Journal of Computational Physics*, 229(22):8333–8363, November 2010.

[15] R. Ghanem and P. D. Spanos. *Stochastic finite elements: A spectral approach*. Springer-Verlag New York, NY, 1991.

[16] L. Giraud, J. Langou, and M. Rozložní. On the round-off error analysis of the gram-schmidt algorithm with reorthogonalization. CERFACS Technical Report No. TR/PA/02/33.

[17] V. Girault and P.-A. Raviart. *Finite element methods for Navier-Stokes equations : theory and algorithms*. Springer series in computational mathematics. Springer, Berlin, 1986.

[18] V. Heuveline. Hiflow3: a flexible and hardware-aware parallel finite element package. In *Proceedings of the 9th Workshop on Parallel/High-Performance Object-Oriented Scientific Computing*, POOSC '10, pages 4:1–4:6, New York, NY, USA, 2010. ACM.

[19] V. Heuveline and M. Schick. A local time-dependent Generalized Polynomial Chaos method for Stochastic Dynamical Systems. EMCL Preprint Series, 2011.

[20] V. Heuveline and M. Schick. Towards a hybrid numerical method using Generalized Polynomial Chaos for Stochastic Differential Equations. EMCL Preprint Series, 2011.

[21] J. G. Heywood, R. Rannacher, and S. Turek. Artificial boundaries and flux and pressure conditions for the incompressible navier-stokes equations. Technical report, 1992.

[22] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE Constraints*, volume 23 of *Mathematical Modelling: Theory and Applications*. Springer Science + Business Media B.V., 2009.

[23] F. Jarre and J. Stoer. *Optimierung*. Springer-Lehrbuch. Springer, Berlin, 2004.

[24] O. P. Le Maître, O. M. Kino, H. N. Najm, and R. G. Ghanem. A stochastic projection method for fluid flow. i: basic formulation. *J. Comput. Phys.*, 173:481–511, November 2001.

[25] O. P. Le Maître and O. M. Knio. *Spectral Methods for Uncertainty Quantification : With Applications to Computational Fluid Dynamics*. Springer Science+Business Media B.V, Dordrecht, 2010.

[26] O. P. Le Maître, L. Mathelin, O. M. Knio, and M. Y. Hussaini. Asynchronous time integration for polynomial chaos expansion of uncertain periodic dynamics. *Discrete and Continuous Dynamical Systems*, 28(1):199–226, September 2010.

[27] O. P. Le Maître, M. T. Reagan, H. N. Najm, R. G. Ghanem, and O. M. Knio. A stochastic projection method for fluid flow ii.: random process. *J. Comput. Phys.*, 181:9–44, September 2002.

[28] P. L'Ecuyer. *Monte Carlo and Quasi-Monte Carlo Methods.* Springer, Berlin, Heidelberg, 2009.

[29] J. S. Liu. *Monte Carlo strategies in scientific computing.* Springer series in statistic. Springer, New York NY, 1. softcover print. edition, 2008.

[30] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):pp. 55–61, 2000.

[31] S. A. Orszag and L. R. Bissonnette. Dynamical properties of truncated Wiener-Hermite expansions. *Physics of Fluids*, 10(12):2603–2613, 1967.

[32] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations.* Springer series in computational mathematics. Springer, Berlin, 1994.

[33] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.

[34] M. Schäfer and J. Turek. Benchmark computations of laminar flow around a cylinder, Januar 1996.

[35] H. Sohr. *The Navier-Stokes equations : an elementary functional analytic approach.* Birkhäuser advanced texts. Birkhäuser, Basel, 2001.

[36] J. H. Spurk and N. Aksel. *Strömungslehre : Einführung in die Theorie der Strömungen; mit Aufgaben und Übungsbeispielen auf CD-ROM.* Springer-Lehrbuch. Springer, Berlin, 8th edition, 2010.

[37] J. Stoer and R. Bulirsch. *Introduction to numerical analysis.* Texts in applied mathematics ; 12. Springer, New York, 3. ed. edition, 2002.

[38] S. Tanaka and T. Imamura. The wiener-hermite expansion with time-dependent ideal random function. ii. *Progress of Theoretical Physics*, 45(4):1098–1105, April 1971.

[39] C. Taylor and P. Hood. A numerical solution of the navier-stokes equations using the finite element technique. *Computers and Fluids*, 1(1):73 – 100, 1973.

[40] R. Temam. *Navier-Stokes equations : theory and numerical analysis.* AMS Chelsea Publ., Providence, RI, reprinted with corr. edition, 2001.

[41] J. W. Thomas. *Numerical partial differential equations*, volume [1]: Finite difference methods of *Texts in applied mathematics.* Springer, New York, 1995.

[42] V. Thomée. *Galerkin finite element methods for parabolic problems.* Springer series in computational mathematics. Springer, Berlin, 1997.

[43] X. Wan and G. Karniadakis. Long-term behavior of polynomial chaos in stochastic flow simulations. *Computer Methods in Applied Mechanics and Engineering*, 195(41-43):5582–5596, 2006.

[44] X. Wan and G. E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, 209:617–642, November 2005.

[45] X. Wan and G. E. Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM Journal on Scientific Computing*, 28(3):901–928, 2006.

[46] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, October 1938.

[47] D. Xiu and G. E. Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002.

[48] D. Xiu and G. E. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187(1):137–167, May 2003.

[49] E. Zeidler. *Applied functional analysis : applications to mathematical physics*. Applied mathematical sciences. Springer, New York, corr. 3. print. edition, 1999.