Karlsruhe Institute of Technology

A local time–dependent Generalized Polynomial Chaos method for Stochastic Dynamical Systems

Vincent Heuveline
Michael Schick

No. 2011-04

Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)

KIT – University of the State of Baden-Wuerttemberg and
National Research Center of the Helmholtz Association

www.emcl.kit.edu

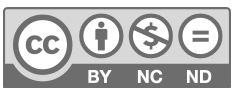
Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)
ISSN 2191–0693
No. 2011-04

Impressum

Karlsruhe Institute of Technology (KIT)
Engineering Mathematics and Computing Lab (EMCL)

Fritz-Erler-Str. 23, building 01.86
76133 Karlsruhe
Germany

KIT – University of the State of Baden Wuerttemberg and
National Laboratory of the Helmholtz Association



A LOCAL TIME–DEPENDENT GENERALIZED POLYNOMIAL CHAOS METHOD FOR STOCHASTIC DYNAMICAL SYSTEMS

VINCENT HEUVELINE AND MICHAEL SCHICK

Abstract. Generalized Polynomial Chaos (gPC) is known to fail for problems involving strong nonlinear dependencies on stochastic inputs, which especially arise in the context of long term integration or stochastic discontinuities. There are various attempts in the literature which address these difficulties, such as the time–dependent generalized Polynomial Chaos (TD-gPC) and the multi–element generalized Polynomial Chaos (ME-gPC) both leading to higher accuracies but higher numerical costs in comparison to the classical gPC approach. A combination of these methods is introduced, which leads to a powerful solution method since high accuracies can be maintained and computational cost can be distributed by utilizing parallel computation. However, to be able to apply the hybrid method to all types of ordinary differential equations subject to random inputs, new modifications with respect to TD-gPC are carried out by creating an orthogonal tensor basis consisting of the random input variable as well as the solution itself. Such modifications allow TD-gPC to capture the dynamics of the solution by increasing the approximation quality of its time derivatives.

1. Introduction

Polynomial Chaos, as initially introduced by Wiener in 1938 [17] is a spectral expansion method with application in the field of uncertainty quantification. It essentially utilizes Hermite–polynomials in terms of Gaussian uncorrelated random variables to decompose a stochastic process into deterministic and non–deterministic parts. Cameron and Martin [1] proved in 1947, that this expansion converges pointwise in mean–square for square–integrable random processes. Ghanem and Spanos [5] pioneered the application of Polynomial Chaos in context of the Finite–Element–Method in the field of solid mechanics in 1991. Later on the projection method became more popular, leading to a broader range of applications, such as CFD (e.g. [8, 10]). However, this classical approach is only capable of capturing the dynamics of Gaussian or close-to-Gaussian processes, since the decomposition of other distribution types converges very slowly or may even diverge. In their paper in 2002, Xiu and Karniadakis [18, 19] proposed a generalization of the Hermite–Chaos for other classes of probability distributions by establishing a correspondence between the probability density function and the weighting function of orthogonal polynomials by hypergeometric series. In 2010, Ernst et al. [3] proved the convergence of generalized Polynomial Chaos (gPC) for certain probability distributions.

In 2006, Wan and Karniadakis [14, 16, 15] developed a multi–element generalized Polynomial Chaos method (ME-gPC) to overcome weaknesses of gPC with respect to accuracy when dealing with strong nonlinear dependencies on the random input. These cases can occur, for example, when the application involves long term integration or stochastic discontinuities. Its basic idea lies in decomposing the probability space of the stochastic input and solve independent local problems. ME-gPC proved to be efficient in reducing the degree of the nonlinear dependencies but still suffers from the fact that after a certain simulation time the accuracy starts to deteriorate again. Alternatives without decomposing the probability space were proposed by Le Maître et

Key words and phrases. Polynomial Chaos, Uncertainty Quantification, Long term integration, Time dependence, Stochastic differential equations, Dynamical systems, Stochastic discontinuities.

al. in [9, 7] dealing with random oscillators with a stable limit–cycle. Here, an asynchronous time integration was developed to allow gPC to capture the dynamics of the solution. In their paper in 2010, Gerritsma et al. [4] introduced a discrete time–dependent approach (TD-gPC) for ordinary differential equations subject to uniformly distributed random inputs, leading to very high accuracies but high numerical costs. The goal of this method is to reduce nonlinear dependencies by a basis transformation to a set of new random variables which are defined by the solution itself at certain discrete time steps. In 2011, the authors proposed a hybrid combination of both ME-gPC and TD-gPC [6]. The key idea is to reduce the numerical cost by exploitation of the trivial parallelization structure of ME-gPC and usage of time dependent basis functionals in each element arising from the decomposition of the probability space of the random input. This work provides the further development and extension of this local time–dependent approach and is structured in the following way:

Section 2 gives a short review of gPC, an introduction to TD-gPC and an a priori error estimation for certain problem classes. Section 3 investigates the application of a modified TD-gPC to the problem class of a linear oscillator, which is also studied in [9]. Section 4 recapitulates the multi–element approach and introduces its combination with time dependent basis functionals. Numerical results for the hybrid method are provided in Section 5 followed by drawing conclusions from this work in Section 6.

2. Time–dependent generalized Polynomial Chaos

2.1. **Generalized Polynomial Chaos.** As developed by Xiu and Karniadakis [18], the generalized Polynomial Chaos method (gPC) represents an extension to the original Polynomial Chaos method, initially introduced by Wiener [17]. It is a stationary, i.e. time–independent projection method onto the space of square–integrable random variables.

Suppose X is a random variable defined on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$, whereas Ω denotes the sample space with samples $\omega \in \Omega$, $\mathcal{F} \subset 2^\Omega$ a σ–algebra defined on the power set of Ω and $\mathbb{P}$ a probability measure. Furthermore, let X be square–integrable, i.e.

$$\mathbb{E}(X^2) = \int_\Omega X^2 \, d\mathbb{P} < \infty. \tag{2.1}$$

As a generalization of the Cameron and Martin theorem [1], a L^2–decomposition of X is then given by

$$X(\omega) = \sum_{i=0}^{\infty} x_i \psi_i(\zeta(\omega)). \tag{2.2}$$

Here, ζ is a (possibly multi–dimensional) random variable according to the probability distribution of X, which defines the set of polynomials $\{\psi_i\}$ according to the Askey–scheme (see Table 1). For example, if ζ is a Gaussian distributed random variable, then Hermite polynomials are chosen, which represents the classical Polynomial Chaos approach. A uniformly distributed ζ leads to the choice of Legendre polynomials. The principle is to select L^2–basis functionals, which are orthogonal with respect to the probability density function f_X of X, i.e.

$$\langle \psi_i, \psi_j \rangle := \int_\Omega \psi_i(\zeta)\psi_j(\zeta) \, d\mathbb{P} = \int \psi_i(z)\psi_j(z) f_X(z) \, dz = \langle \psi_i, \psi_i \rangle \delta_{ij}. \tag{2.3}$$

whereas δ_{ij} denotes the Kronecker–Delta. Note that from here and in the following we do not explicitly denote the domain of integration in cases involving the probability density function for notational convenience. The advantage of gPC is that a functional dependency on the random variable is given a priori for the decomposed random variable X. The task left is to calculate the modes x_i of X, which is usually numerical expensive, especially when dealing with stochastic

Case	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 1. Askey–scheme for selecting polynomials corresponding to certain types of distributions.

processes. For those, the procedure described above is applied pointwise, i.e. given a stochastic process $X = X(t;\omega)$, dependent on some time variable $t \geq 0$, the gPC decomposition reads

$$X(t;\omega) = \sum_{i=0}^{\infty} x_i(t)\psi_i(\zeta(\omega)), \tag{2.4}$$

with time dependent modes $x_i = x_i(t)$. Of course, since infinite sums are numerically intractable, (2.4) needs to be truncated at some finite integer M, resulting in an approximation X^M of X defined by

$$X^M(t;\omega) := \sum_{i=0}^{M} x_i(t)\psi_i(\zeta(\omega)). \tag{2.5}$$

The truncation order M is dependent on two discretization parameters, which can be chosen a priori. The first one, denoted by P, represents the maximal total polynomial degree allowed. In a multi–dimensional case, the functionals ψ_i are constructed via a tensor product of corresponding one–dimensional polynomials, therefore, the total polynomial degree of some ψ_i equals to the sum of the degrees of the one–dimensional polynomials. The second parameter, denoted by L, represents the dimension of the random vector $\zeta = (\zeta_1, \ldots, \zeta_L)$, whereas ζ_i are scalar random variables which distribution is known a priori. Both parameters combined result in the corresponding truncation parameter M, via

$$M + 1 = \frac{(P + L)!}{P!L!}. \tag{2.6}$$

Note that even for low P and L the number of unknown variables $M + 1$ grows rapidly. This leads to a huge numerical drawback, since every single mode $x_i = x_i(t)$ has to be discretized further by some deterministic discretization method. This phenomena is often referred to as "curse of dimensionality".

2.2. **Application of gPC to ordinary differential equations.** The procedure outlined above decomposes a stochastic process into a (finite) summation of products between deterministic functions of time and chosen stochastic basis functionals, the so–called Chaos Polynomials. Next, we briefly review their application to ordinary differential equations subject to stochastic input via a standard Galerkin approach.

Let us consider, without loss of generality, the following scalar problem:
Seek $u = u(t;\omega)$ with $t \in [0, T] \subset \mathbb{R}$ and $\omega \in \Omega$, such that

$$\mathcal{L}(u, t;\omega) = f(t;\omega), \tag{2.7}$$

where $\mathcal{L}$ denotes some differential operator in the time variable t. The stochastic influence is assumed to be parametrized by some vector of independent random variables $\zeta = (\zeta_1, \zeta_2, \ldots)$. Hence, problem (2.7) can be reformulated to obtain the form:

Seek $u = u(t;\zeta) = u(t;\zeta(\omega))$ with $t \in [0,T]$ and $\omega \in \Omega$, such that

$$\mathcal{L}(u,t;\zeta) = f(t;\zeta), \tag{2.8}$$

whereas the notation of ω is dropped for notational convenience. At this stage, the gPC discretization of u is employed by truncation of ζ to a finite dimensional vector with dimension L and by choosing the maximal polynomial degree P. The approximation u^M of u is now inserted into the governing equation (2.8) leading to

$$\mathcal{L}(\sum_{i=0}^{M} u_i\psi_i, t;\zeta) = f(t;\zeta). \tag{2.9}$$

Now, there exist several possibilities of solving (2.9), e.g. a least–squares approach or collocation methods. Most commonly used is the Galerkin projection onto the space $\mathcal{V}$ spanned by the Chaos Polynomials, i.e. $\mathcal{V} := span\{\psi_0, \ldots, \psi_M\}$, which ensures the orthogonality of the residual to $\mathcal{V}$. Therefore, (2.9) is multiplied by ψ_j for every $j = 0, \ldots, M$ and the inner product $\langle\cdot\rangle$ on $\mathcal{V}$ is taken, resulting in

$$\langle\mathcal{L}(\sum_{i=0}^{M} u_i\psi_i, t;\zeta), \psi_j\rangle = \langle f, \psi_j\rangle, \quad j = 0, \ldots, M. \tag{2.10}$$

This leads to a coupled deterministic system of differential equations, which size is equal to $M+1$. Note, that if uncertainty is only involved in either the right hand side f or initial or boundary conditions, the system actually deteriorates to a stochastic decoupled structure for linear operators $\mathcal{L}$.

2.3. **Time–dependent basis functionals.** A well known difficulty when employing gPC is the possible convergence breakdown in cases involving strong nonlinear dependencies on the random input ζ. These cases can occur for example when dealing with long term integration or stochastic discontinuities. Since a stationary, i.e. time–independent approach is used, the time evolution of the probability density function of u cannot be captured efficiently after some application dependent critical time. This can be seen clearly when taking a look at the discretization parameter P, which essentially only allows for nonlinear dependencies up to the order of P. This problem has been studied in various works, e.g. [4, 2, 13, 15, 14, 9], leading to promising modifications toward gPC to overcome the lack of convergence. Here, we want to recapitulate and extend one of the more recent approaches introduced by Gerritsma et al. [4], called the time–dependent generalized Polynomial Chaos (TD-gPC).

Recall the gPC approximation u^M of u given by

$$u^M(t;\zeta) = \sum_{i=0}^{M} u_i(t)\psi_i(\zeta). \tag{2.11}$$

At each fixed time t^*, u^M defines a random variable η depending on ζ via

$$\eta := u^M(t^*;\zeta) = \sum_{i=0}^{M} u_i(t^*)\psi_i(\zeta). \tag{2.12}$$

The key idea is to express u^M in terms of η for times $t \geq t^*$. Then, the nonlinear dependency on ζ is given implicitly through η, however, the solution's dependence on η is linear at $t = t^*$ and is expected to be close to linear for short times $t \geq t^*$, which is leading to an efficient representation of u via gPC. For a certain $t^{**} > t^*$ the change of variables is repeated once again to ensure low

nonlinear dependencies. This principle is then applied throughout the simulation interval $[0, T]$. To maintain optimal, i.e. orthogonal basis functionals the Chaos Polynomials are recomputed for every change of variables, such that

$$\int \psi_i(\eta)\psi_j(\eta) f_\eta(\eta)\, d\eta = \langle \psi_i, \psi_i \rangle \delta_{ij}, \tag{2.13}$$

whereas f_η denotes the probability density function of η. However, it is important to note that computing the probability density function f_η of η is numerically not feasible. Instead (2.13) is transformed to the original random variable ζ in the following way:

$$\int \psi_i(\eta)\psi_j(\eta) f_\eta(\eta)\, d\eta = \int \psi_i(u^M(t^*;\zeta))\psi_j(u^M(t^*;\zeta)) f_\zeta(\zeta)\, d\zeta, \tag{2.14}$$

whereas f_ζ denotes the probability density function of ζ. This has the advantage that since f_ζ and u^M at $t = t^*$ are both known, all integrals can be evaluated in terms of ζ and still represent the dependencies on η without explicit knowledge of f_η.

After having computed the new Chaos Polynomials the Galerkin projection is applied to the governing equations similar as described for the gPC. However, new initial conditions $u_j^{(new)}(t^*)$ for $j = 0, \ldots, M$ need to be provided. This can easily be achieved by an orthogonal projection of the current solution values $u_i^{(old)}(t^*)$ at time $t = t^*$ onto the new basis via

$$u_j^{(new)}(t^*) = \sum_{i=0}^{M} u_i^{(old)}(t^*) \frac{\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle}{\langle \psi_j^{(new)}, \psi_j^{(new)} \rangle}, \quad j = 0, \ldots, M, \tag{2.15}$$

whereas

$$\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle = \int \psi_i^{(old)}(\zeta)\psi_j^{(new)}(u^M(t^*;\zeta)) f_\zeta(\zeta)\, d\zeta. \tag{2.16}$$

The projection step (2.15) can be simplified further as stated by the following theorem:

Theorem 1. *The initial conditions $u_j^{(new)}(t^*)$ at time $t = t^*$ for u^M and $j = 0, \ldots, M$ are given by*

$$u_0^{(new)}(t^*) = u_0^{(old)}(t^*), \tag{2.17}$$

$$u_1^{(new)}(t^*) = 1, \tag{2.18}$$

$$u_j^{(new)}(t^*) = 0, \quad \text{for } j = 2, \ldots, M, \tag{2.19}$$

with the initial polynomial of degree 0 defined by $\psi_0^{(new)} := 1$ and polynomial $\psi_1^{(new)}$ defined with leading coefficient 1.

Proof. At time $t = t^*$ the following requirement holds:

$$\sum_{i=0}^{M} u_i^{(new)}(t^*)\psi_i^{(new)}(\eta) \stackrel{!}{=} \eta. \tag{2.20}$$

Since $\psi_i^{(new)}(\eta)$ is nonlinear for $i \geq 2$ it follows immediately that $u_i^{(new)} = 0$ for $i \geq 2$. Furthermore, it holds that $\psi_1^{(new)}(\eta) = \eta - \alpha$ for some $\alpha \in \mathbb{R}$. Since $\psi_1^{(new)}$ is required to be orthogonal to $\psi_0^{(new)} = 1$ it follows that

$$0 = \int (\eta - \alpha) f_\eta(\eta)\, d\eta = \int (u(t^*;\zeta) - \alpha) f_\zeta(\zeta)\, d\zeta, \tag{2.21}$$

and hence

$$\alpha = \mathbb{E}(u(t^*;\zeta)) = u_0^{(old)}(t^*). \tag{2.22}$$

Rewriting equation (2.20) we obtain

$$u_0^{(new)}(t^*) + u_1^{(new)}(t^*)(\eta - u_0^{(old)}(t^*)) = \eta, \tag{2.23}$$

which has the unique solution $u_0^{(new)}(t^*) = u_0^{(old)}(t^*)$ and $u_1^{(new)}(t^*) = 1$. □

The computation of the stochastic moments for $t \in [t^*, t^{**})$, here the mean $\bar{u}$ and the variance $\sigma^2(u)$ needs to be carried out according to η via

$$\bar{u}(t) = u_0(t), \tag{2.24}$$

$$\sigma^2(u)(t) = \sum_{i=1}^{M} u_i(t)^2 \langle \psi_i^{(new)}, \psi_i^{(new)} \rangle, \tag{2.25}$$

due to the orthogonality of the basis functionals $\psi_i^{(new)}$.

2.4. **A priori error estimates.** This section aims at providing some a priori estimates toward the error contribution arising from TD-gPC when progressing in time. Therefore, we focus on three example problem classes, which display different convergence behaviors. For this purpose, we restrict our attention to scalar equations represented by the form

$$\frac{d}{dt} u(t;\omega) = f(t, u; \zeta(\omega)), \tag{2.26}$$

where f denotes some right hand side depending on the time variable t, the solution u and some initial random variable ζ. Note that at time $t = t^*$, when a change of basis is performed from ζ to η, equation (2.26) can be written in the form

$$\frac{d}{dt} u(t^*;\eta) = f(t^*, \eta; \zeta). \tag{2.27}$$

For the further analysis, we expand u in its Taylor series representation w.r.t. some time step size $\Delta t > 0$:

$$u(t^* + \Delta t; \eta) = \eta + \Delta t \frac{d}{dt} u(t^*;\eta) + \frac{(\Delta t)^2}{2} \frac{d^2}{dt^2} u(t^*;\eta) + \dots. \tag{2.28}$$

It follows immediately from (2.28) that the error contribution of the basis in terms of η is at least of the order $O(\Delta t)$, since it depends on f how well the time derivatives can be approximated in terms of η. In the following, we will illustrate how three different types of f can effect the error contribution order.

Polynomial right hand side. Suppose $f = f(u) = u^n$ for some $n \in \mathbb{N} \setminus \{1\}$. Differentiating f with respect to t by applying the chain rule we obtain

$$\frac{d^i}{dt^i} f(u) = n(2n-1)\cdots(in-(i-1))u^{(i+1)n-i}, \quad i \geq 0. \tag{2.29}$$

It follows that a discretization employing a P^{th} order TD-gPC expansion is exact for $(i+1)n - i \leq P$ and $P \geq n$, i.e. all time derivatives can be expressed in terms of η within this restriction due to

$$u(t^* + \Delta t; \eta) = \eta + \sum_{j=1}^{i} \frac{(\Delta t)^j}{j!} \frac{d^{j-1}}{dt^{j-1}} f(\eta) + O((\Delta t)^{i+1}). \tag{2.30}$$

From this it is easy to show that the error contribution is of order $O((\Delta t)^{\frac{P-1}{n-1}+1})$.

Exponential right hand side. Suppose $f = f(u) = \exp(u)$. Differentiating f with respect to t by applying the chain rule we obtain

$$\frac{d^i}{dt^i} f(u) = \exp((i+1)u), \quad i \geq 0. \tag{2.31}$$

Therefore no discretization which employs a P^{th} order TD-gPC expansion can capture the evolution of the time derivatives of u in an exact way. Hence, it follows from the Taylor series representation (2.28) that the error contribution is of order $O(\Delta t)$.

Linear right hand side with explicit dependence on initial random input. Suppose $f = f(u;\zeta) = -\zeta u$ for some initial stochastic input random variable ζ with arbitrary probability distribution. Differentiating f with respect to t by applying the chain rule we obtain

$$\frac{d^i}{dt^i} f(u;\zeta) = (-1)^{i+1}\zeta^{i+1}u, \quad i \geq 0. \tag{2.32}$$

In this case 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 ζ. Therefore, TD-gPC as introduced is not feasible for this class of applications leading to an error contribution of order $O(\Delta t)$, which follows from the Taylor series representation in (2.28). However, it is possible to overcome this drawback by defining a new basis in terms of η and ζ via a Tensor product, such that u can be expressed via

$$u(t;\omega) = \sum_{i=0}^{P}\sum_{j=0}^{Q} u_{ij}(t)\psi_i(\eta)\phi_j(\zeta), \tag{2.33}$$

whereas $\{\psi_i\}_{i=0}^{P}$ and $\{\phi_j\}_{j=0}^{Q}$ are orthogonal polynomials with respect to the probability distribution of η and ζ, respectively. Such a basis is capable of representing the time derivatives of u to the order of $O((\Delta t)^{Q+1})$ in this case. The next section will demonstrate this basis extension and give numerical results, which display the improved convergence behavior.

3. LINEAR OSCILLATOR

For this problem it is known that the classical gPC expansion fails to capture the dynamics of the solution after some certain time [9]. To overcome this issue Le Maître et al. [9] introduced an asynchronous time integration method valid for problems involving stable limit cycles. Here, we show that employing TD-gPC leads to a powerful alternative when being modified accordingly.

3.1. **Problem definition.**

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

$$\frac{d}{dt} x_1(t) = x_2(t), \tag{3.1}$$

$$\frac{d}{dt} x_2(t) = -q x_1(t), \tag{3.2}$$

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 $x_2(t=0) = 0$. We will consider a random frequency, i.e. $q = q(\zeta) = q_0 + q_1\zeta$, with a uniformly distributed $\zeta \sim U(-1,1)$. The analytical solutions are given by

$$x_1(t;\zeta) = \cos(\sqrt{q(\zeta)}t), \tag{3.3}$$

$$x_2(t;\zeta) = -\sqrt{q(\zeta)}\sin(\sqrt{q(\zeta)}t). \tag{3.4}$$

3.2. **Discretization employing TD-gPC.** Since we are dealing with a vector–valued problem in two dimensions, we extend the procedure as described in Section 2.3 to this case, represented at some reset time $t = t^*$. We begin with the classical gPC discretization of x_1 and x_2 given by

$$x_1(t;\zeta) = \sum_{i=0}^{P} x_i^{(1)}(t) L_i(\zeta), \tag{3.5}$$

$$x_2(t;\zeta) = \sum_{i=0}^{P} x_i^{(2)}(t) L_i(\zeta), \tag{3.6}$$

with L_i denoting the Legendre polynomials in terms of the uniformly distributed random variable ζ. Here, $M = P$ since we are dealing with a one–dimensional random input. Therefore, the index i of L_i equals to the degree of the considered Legendre polynomial. At some reset time $t = t^*$ we define two new random variables corresponding to the solution components via

$$\eta^{(1)}(\zeta) := \sum_{i=0}^{P} x_i^{(1)}(t^*) L_i(\zeta), \tag{3.7}$$

$$\eta^{(2)}(\zeta) := \sum_{i=0}^{P} x_i^{(2)}(t^*) L_i(\zeta). \tag{3.8}$$

From this point on, we are dealing with a multi–dimensional stochastic input given by $\eta^{(1)}$ and $\eta^{(2)}$. This needs to be taken into account when employing gPC in terms of the new random variables. However, since $\eta^{(1)}$ and $\eta^{(2)}$ are dependent random variables via ζ, we suggest a modification to the classical approach to maintain orthogonality of the multi–dimensional basis functionals, which has proved to be more numerically stable. We start with computing orthogonal Chaos Polynomials $\psi_j^{(i)}$, $i = 1, 2$, $j = 0, \ldots, P$ with respect to each random variable $\eta^{(i)}$, $i = 1, 2$, such that

$$\int \psi_s^{(i)}(\eta)\psi_r^{(i)}(\eta) f_{\eta^{(i)}}(\eta)\, d\eta = \langle \psi_s^{(i)}, \psi_s^{(i)} \rangle \delta_{sr}, \quad s, r = 0, \ldots, P,\ i = 1, 2. \tag{3.9}$$

This can be achieved for example by employing a Gram–Schmidt orthogonalization method. Note that computing the integral in (3.9) can be transformed to the original random variable ζ as described in Section 2.3 to avoid the explicit calculation of the probability density functions $f_{\eta^{(i)}}$ of $\eta^{(i)}$, $i = 1, 2$.

Next, we define a new temporary basis by a tensor product of the corresponding one–dimensional polynomials via

$$x_i(t;\eta^{(1)};\eta^{(2)}) = \sum_{0 \le i+j \le P} u_{ij}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)}), \quad t \ge t^*, i = 1, 2, \tag{3.10}$$

which alternatively can be expressed by

$$x_i(t;\eta^{(1)};\eta^{(2)}) = \sum_{j=0}^{M} x_j^{(i)}(t)\phi_j(\eta^{(1)}, \eta^{(2)}), \quad t \ge t^*, i = 1, 2, \tag{3.11}$$

by a one–to–one correspondence between the basis functionals and coefficients, whereas for the number $M + 1$ of terms in (3.11) it holds

$$M + 1 = \frac{(P+2)!}{P!2!} = \frac{(P+1)(P+2)}{2}. \tag{3.12}$$

Note that at this stage, the basis polynomials ϕ_i are not orthogonal to each other because of the dependency of $\eta^{(i)}$, $i = 1, 2$ introduced through ζ. Therefore, we orthogonalize the basis via a

Gram–Schmidt method in two dimensions, maintaining an orthogonal projection of x_1 and x_2. To this end we define an orthogonal basis via

$$\psi_0 := 1, \tag{3.13}$$

$$\psi_i(\eta^{(1)}, \eta^{(2)}) := \phi_i(\eta^{(1)}, \eta^{(2)}) - \sum_{j=0}^{i-1} \frac{\langle \phi_i, \psi_j \rangle}{\langle \psi_j, \psi_j \rangle} \psi_j(\eta^{(1)}, \eta^{(2)}), \quad i = 1, \dots, M. \tag{3.14}$$

Since we now employ an orthogonal basis, it is straight forward to calculate the required initial conditions at $t = t^*$ by a projection similar to the one introduced in Section 2.3:

$$x_j^{(new)}(t^*) = \sum_{i=0}^{M} x_i^{(old)}(t^*) \frac{\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle}{\langle \psi_j^{(new)}, \psi_j^{(new)} \rangle}, \quad \text{for } j = 0, \dots, M. \tag{3.15}$$

Note, that in case of the first reset within the simulation time interval all (dummy) modes of the "old" solution with index $i > P$ are set to zero. The procedure described above is then repeated at every necessary time step, which can be identified by some error estimation criteria or chosen to be each time step within the time discretization procedure.

For postprocessing 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 in the classical gPC via

$$\bar{x}_i(t) = x_0^{(i)}(t), \tag{3.16}$$

$$\sigma^2(x_i)(t) = \sum_{j=1}^{M} \left(x_j^{(i)} \right)^2 \langle \psi_j, \psi_j \rangle, \tag{3.17}$$

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

3.3. **Modified TD-gPC.** Before we state the numerical results, one major drawback of TD-gPC is analyzed as already described in Section 2.4, 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 we take a look at the second equation of the problem:

$$\frac{d}{dt} x_2 = -q(\zeta) x_1. \tag{3.18}$$

Employing any deterministic time–discretization scheme represented by some function g, this results in

$$x_2(t + \Delta t; \zeta) = g(x_1(t; \zeta), x_1(t + \Delta t; \zeta), t, \zeta), \tag{3.19}$$

for some time step size $\Delta t > 0$. Here the explicit dependency of g on ζ is crucial. If changing the variables from ζ to $\eta^{(i)}$, $i = 1, 2$ due to TD-gPC, we arrive at an optimal representation of the solution x_1, x_2 itself at every time step but we are unable to capture the solution's time evolution, i.e. its time derivative in terms of the new random variables, since g is still depending on the initial random variable ζ. Hence, when progressing in time, the error made because of a poor representation of the time derivatives increases steadily, leading to unfeasible results similar to the case demonstrated in Section 2.4. Therefore, the temporary basis $\psi_i^{(1)}, \psi_i^{(2)}$ is modified to take into

account the initial random variable ζ, resulting in

$$x_1(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{0\le i+j\le P}^{P}\sum_{k=0}^{Q} x_{ijk}^{(1)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\zeta), \tag{3.20}$$

$$x_2(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{0\le i+j\le P}^{P}\sum_{k=0}^{Q} x_{ijk}^{(2)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\zeta), \tag{3.21}$$

whereas L_k, $k = 0,\dots,Q$ again denote the Legendre polynomials in terms of ζ according to gPC. Next we construct a new orthogonal basis $\{\psi_j\}_{j=0}^{M}$ analog to the procedure described above, resulting in

$$x_i(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{j=0}^{M} x_j^{(i)}(t)\psi_j(\eta^{(1)},\eta^{(2)},\zeta), \quad t \ge t^*, i = 1,2, \tag{3.22}$$

with the number of terms given by

$$M+1 = \frac{(P+1)(P+2)(Q+1)}{2}. \tag{3.23}$$

The calculation of the initial values at time $t = t^*$ is carried out using the projection described in (3.15). The same holds concerning the calculation of the mean and the variances.

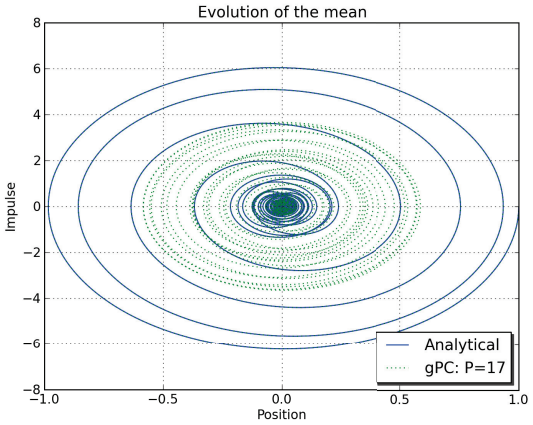
3.4. **Numerical results.**

Next we present numerical results with respect to various TD-gPC expansion orders. The random frequency is defined to be

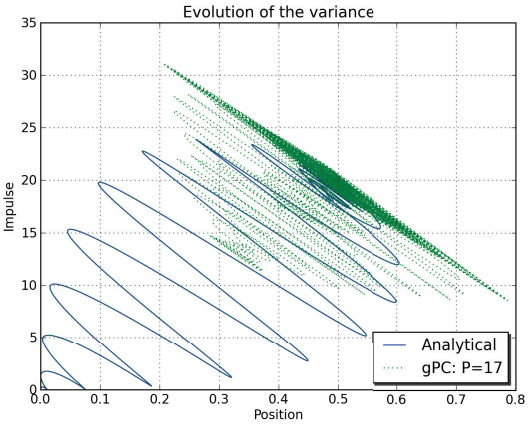
$$q(\zeta) = 4\pi^2(1+0.2\zeta). \tag{3.24}$$

We employ an explicit Runge–Kutta scheme of order 4 with a time step of $\Delta t = 0.001$ to minimize the error contributions introduced by the time discretization. Furthermore, a reset was carried out in every time step throughout the simulation interval $[0, 75]$. 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. 3.1 and Fig. 3.2, 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 classical gPC approach using the same number of modes $M+1$, e.g. for $P = 2$ and $Q = 2$ TD-gPC this equals 18 modes, i.e. $P = 17$ for the gPC.

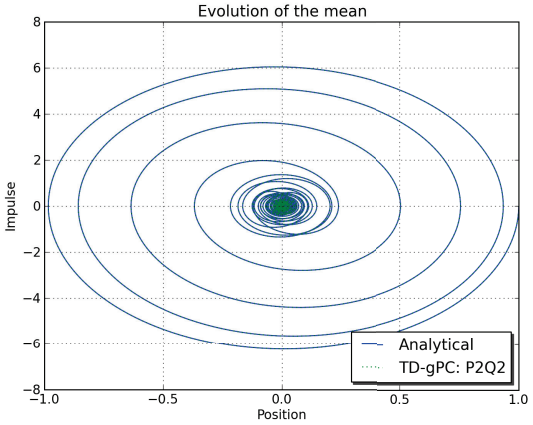
As expected, the classical gPC employing Legendre polynomials is only capable of following the solution for early times even for a large number of modes. The time–dependent approach, however, is performing slightly worse if $Q = 0$, i.e. the errors arising from a poor representation of the time derivative 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 achieved with respect to Q. However, using $P = 0$ and $Q > 0$, TD-gPC actually would be equal to the classical gPC and therefore leading to unfeasible results. This emphasizes the importance of an optimal basis both for the solution itself as well as for its time derivative.



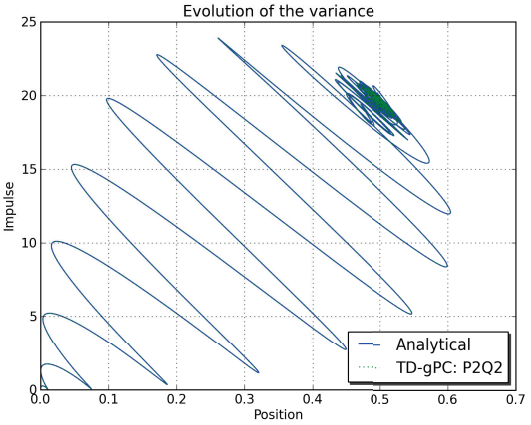
(A) Classical Legendre Chaos $P = 17$.



(B) Classical Legendre Chaos $P = 17$.



(C) Time–dependent variant $P = 2, Q = 2$.

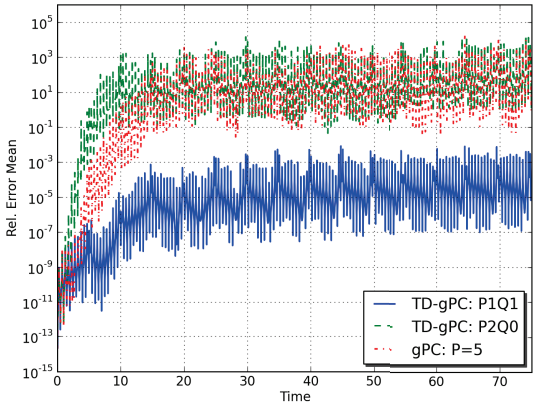


(D) Time–dependent variant $P = 2, Q = 2$.

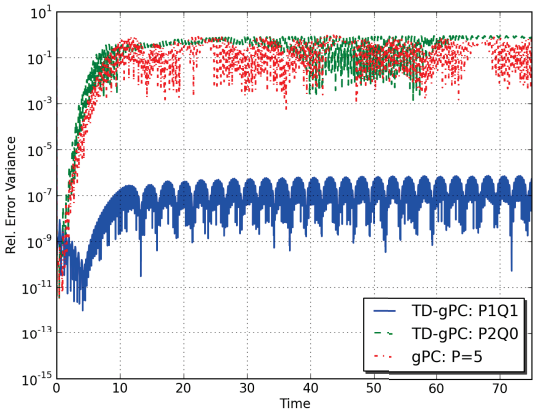
FIGURE 3.1. Evolution of the trajectories corresponding to the mean and the variance for the classical gPC employing $P = 17$ modes (total 18) and the time–dependent variant employing $P = 2, Q = 2$.

4. Local approach for time–dependent generalized polynomial chaos

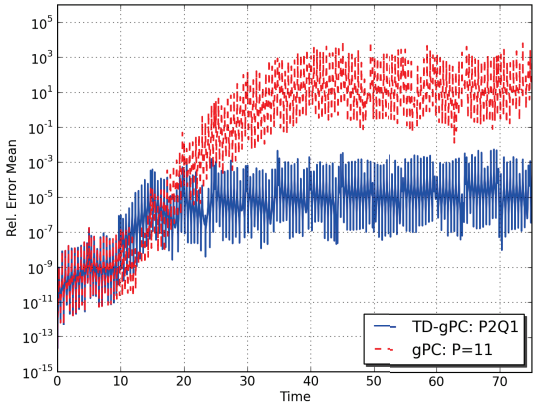
One major drawback of TD-gPC is the fast growing number of modes resulting for even low expansion orders P and Q. Therefore, although the method itself is leading to accurate results, the numerical cost which comes along with solving a coupled system of differential equations for a high number of modes increases drastically, 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, we want to introduce a local approach to TD-gPC motivated by the multi–element generalized Polynomial Chaos introduced by Wan and Karniadakis [14, 16], 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.



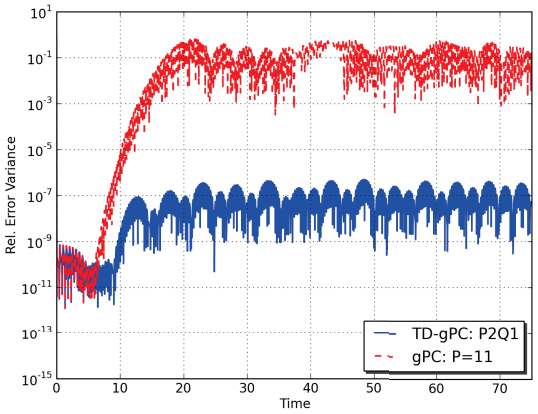
(A) Relative error mean with a total of 6 modes each.



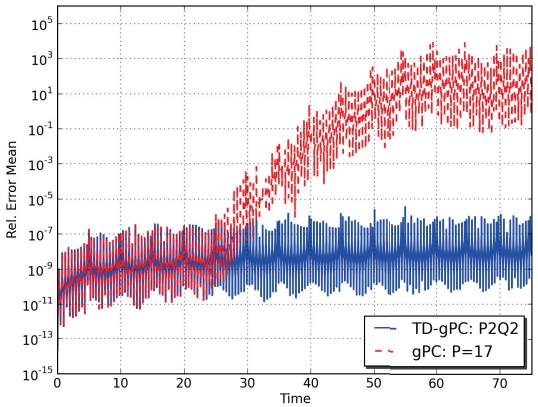
(B) Relative error variance with a total of 6 modes each.



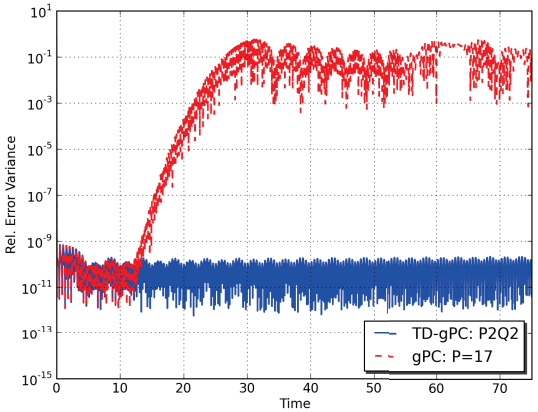
(C) Relative error mean with a total of 12 modes each.



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



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



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

FIGURE 3.2. Relative errors of mean and variance of $x_1$corresponding to various discretization parameters.

4.1. **Domain decomposition.** Following the procedure described in [14, 16] we decompose the sample space Ω implicitly by decomposing the range of ζ. Note, that here we do not assume a scalar valued ζ, instead it holds $range(\zeta) =: B \subset (\mathbb{R} \cup \{-\infty, \infty\})^d$ for some $d \in \mathbb{N}$. Furthermore, ζ denotes the original stochastic input to the system before being transformed via 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

$$B = \bigcup_{j=1}^{N} B_j, \quad B_{j_1} \cap B_{j_2} = \emptyset \quad \text{for } j_1 \neq j_2, \tag{4.1}$$

$$B_j := [a_1^j, b_1^j) \times [a_2^j, b_2^j) \times \cdots \times [a_d^j, b_d^j]. \tag{4.2}$$

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 into $(-\infty, a_*^j), [a_*^j, b_*^j], (b_*^j, \infty)$ 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 we introduce the indicator function I_j defined by

$$I_j = \begin{cases} 1 & \text{if } \zeta \in B_j, \\ 0 & \text{otherwise.} \end{cases} \tag{4.3}$$

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 ζ^j is defined via $\zeta^j = (\zeta_1^j, \zeta_2^j, \ldots, \zeta_d^j) : I_j^{-1}(1) \mapsto B_j$ for $j = 1, \ldots, N$ subject to the corresponding conditional probability distribution function f_j defined by

$$f_j(\zeta^j | I_j = 1) = \frac{f(\zeta^j)}{\mathbb{P}(I_j = 1)}, \tag{4.4}$$

whereas $f(\cdot)$ denotes the probability density function of the global random variable ζ. In practice the random variable ζ_k^j defined in the element $[a_k^j, b_k^j)$ is rescaled by the transformation

$$\zeta_k^j = \frac{b_k^j - a_k^j}{2} Y_k^j + \frac{b_k^j + a_k^j}{2}, \tag{4.5}$$

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

$$\bar{f}_j(y_j) = \det \left| \frac{\partial \zeta^j}{\partial y_j} \right| f_j(\zeta^j(y_j) \,|\, I_j = 1) = \frac{f(\zeta^j(y_j))}{\mathbb{P}(I_j = 1)} \prod_{k=1}^{d} \frac{b_k^j - a_k^j}{2}. \tag{4.6}$$

4.2. **Problem structure.** The strength in decomposing the probability space lies in the independency of the resulting local problems. Since the probability measure of an element boundary equals zero, no absolute continuity of the solution, denoted by u, on the boundaries must be guaranteed, i.e. the restriction

$$u_{B_1}(\zeta) = u_{B_2}(\zeta), \quad \zeta \in \bar{B}_1 \cap \bar{B}_2, \tag{4.7}$$

is not required. This is due to the fact that most statistics of u are integrations with respect to the underlying probability distribution function. Recapitulating, we have that 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 $u^{(j)} = u^{(j)}(\zeta^j)$ in each element j, a new random variable η_k^j is introduced via

$$\eta_k^j := \sum_{i=0}^{M} u_i^{(j)}(t_k)\psi_i(\eta_{k-1}^j, \zeta^j), \tag{4.8}$$

for a reset time step $t = t_k$, whereas $\eta_0^j := \zeta^j$. The solution $u^{(j)}$ is then expressed in terms of η_k^j and ζ^j as described in Section 2.3 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 criteria.

4.3. **Calculation of the stochastic moments.** Since a change of variables if 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, we first have to calculate the mean and the variance locally according to Section 2.3, denoted by $\bar{u}^{(j)}$ and $\sigma^2(u^{(j)})$, $j = 1, \dots, N$. Next, according to Bayes' theorem and the law of total probability [11], the global stochastic moments of order m, denoted by μ_m, can be calculated via

$$\mu_m(u)(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\mu_m(u^{(j)})(t). \tag{4.9}$$

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

$$\bar{u}(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\bar{u}^{(j)}(t). \tag{4.10}$$

For the variance it holds

$$\begin{aligned}
\sigma^2(u)(t) &= \mu_2(u)(t) - \mu_1(u)(t)^2 \\
&\approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\mu_2(u^{(j)})(t) - \sum_{j=1}^{N}\sum_{k=1}^{N} \mathbb{P}(I_j = 1)\mathbb{P}(I_k = 1)\mu_1(u^{(j)})(t)\mu_1(u^{(k)})(t) \\
&= \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\left(\sigma^2(u^{(j)})(t) + (\bar{u}^{(j)})^2 - \bar{u}^{(j)}(t)\sum_{k=1}^{N} \mathbb{P}(I_k = 1)\bar{u}^{(k)}(t)\right) \\
&= \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\left(\sigma^2(u^{(j)})(t) + \bar{u}^{(j)}(t)(\bar{u}^{(j)}(t) - \bar{u}(t))\right). \qquad (4.11)
\end{aligned}$$

4.4. **Implementation issues.** 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 [4] can lead to a significant lowering of the numerical cost. However, defining a reset criteria introduces an extra source of errors, therefore we choose to apply TD-gPC in every time step. A summary of the local TD-gPC algorithm is given in Algorithm 1.

5. Numerical results for the local time–dependent approach

In this section we demonstrate the effect of employing the local TD-gPC in context of a simple one–dimensional ordinary differential equation, representing the class of long term integration

Algorithm 1 Local TD-gPC.

Step 1: *Choose the number of elements* N
Step 2: *Loop over all elements* j*:*
  Step 2a: *Construct the local conditional random variable* $\eta_0^{(j)} := \zeta^{(j)}$ *w.r.t. the initial stochastic input*
  Step 2b: *Transform* $\eta_0^{(j)}$ *to a random variable* $Y^{(j)}$ *defined on* $(-1,1)$ *via* (4.5) *and set* $\eta_0^{(j)} = Y^{(j)}$
  Step 2c: *Loop over all time steps* i*:*
    - *Construct a new random variable* $\eta_i^{(j)}$ *according to TD-gPC via* $\eta_{i-1}^{(j)}$, $\zeta^{(j)}$ *and the local solution* $u^{(j)}$
    - *Construct a new set of orthogonal basis functionals* $\psi_i^{(j)}$ *depending on* $\eta_i^{(j)}$ *and* ζ^j *w.r.t.* P *and* Q
    - *Generate new local initial conditions according to TD-gPC*
  Step 2d: *Store the calculated local mean* $\bar{u}^{(j)}$ *and variance* $\sigma^2(u^{(j)})$
Step 3: *Calculate the global mean and variance via the stored local quantities*
Step 4: *Postprocessing*

related problems, and the more challenging Kraichnan–Orszag three mode problem, which is representing the class of stochastic discontinuities. We focus on the illustration of the convergence properties and analyze the trade off between solving N independent local problems and employing an $(M+1)$–dimensional TD-gPC in each element.

5.1. **A simple one–dimensional ODE.** This problem has been studied in various work, e.g. [14, 4], having the advantage that its simplicity allows to calculate an analytical solution. The governing equations are given by

$$\frac{du}{dt} = -k(\zeta)u, \tag{5.1}$$

$$u(0) = 1, \tag{5.2}$$

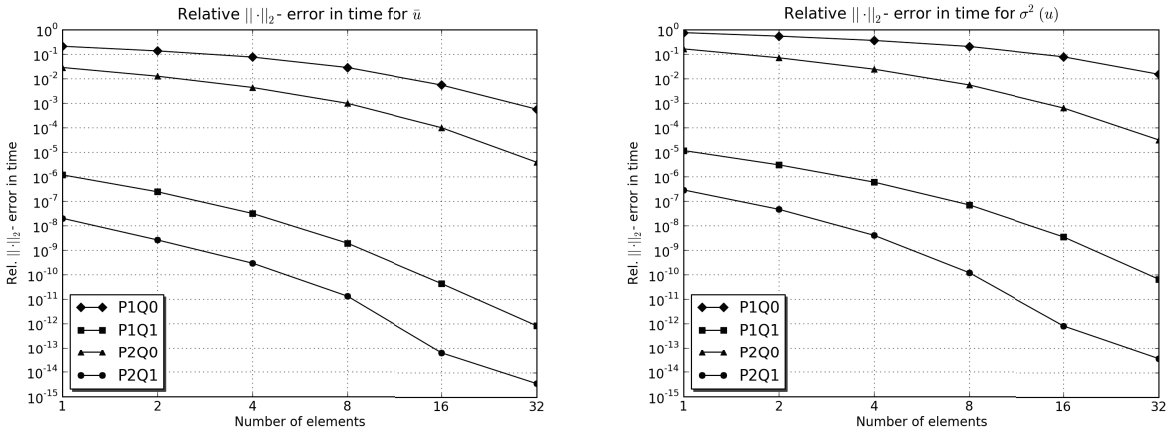
subject to a uniformly distributed random variable $\zeta \sim U(-1,1)$, where we assume further that $k(\zeta) = \frac{1}{2}(1+\zeta)$. Therefore, the analytical solution and its mean and variance are given by

$$u(t;\zeta) = \exp(-k(\zeta)t), \tag{5.3}$$

$$\bar{u}(t) = \frac{1-\exp(-t)}{t}, \tag{5.4}$$

$$\sigma^2(u)(t) = \frac{1}{2t}(1-\exp(-2t)) - \left(\frac{1-\exp(-t)}{t}\right)^2, \tag{5.5}$$

respectively. This clearly shows the increasing nonlinear dependency of u on ζ for increasing time t due to the exponential type of the solution, which leads to a poor convergence property when employing the classical gPC (see for example [4]). Here, we only focus on the convergence behavior of the local time–dependent gPC. For our numerical simulation we use a Runge–Kutta scheme of 4^{th} order with a time step of $\Delta t = 0.001$ to minimize the errors arising from the time discretization.



(A) Mean (B) Variance

FIGURE 5.1. $\|\cdot\|_2$–error in time with respect to various orders of chaos expansions with varying number of elements.

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

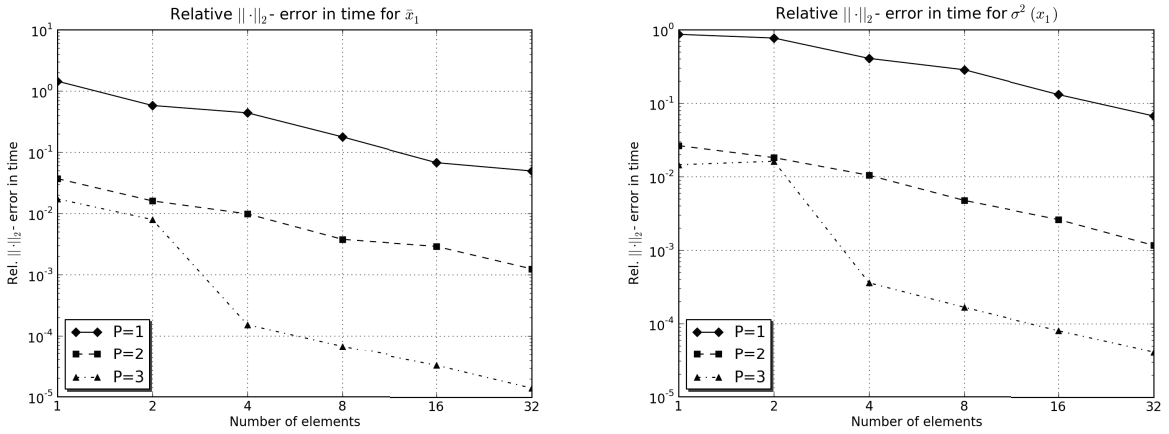
$$\|\bar{u} - \bar{u}_{exact}\|_2 = \sqrt{\left(\sum_n \left(\bar{u}(t_n) - \bar{u}_{exact}(t_n)\right)^2\right)}, \tag{5.6}$$

$$\|\sigma^2(u) - \sigma^2(u_{exact})\|_2 = \sqrt{\left(\sum_n \left(\sigma^2(u)(t_n) - \sigma^2(u_{exact})(t_n)\right)^2\right)}, \tag{5.7}$$

for all discrete time steps $t_n = n\Delta t$ within the simulation interval $[0, 100]$. Figure 5.1 plots the 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 [14]. 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 independency 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.

5.2. **The Kraichnan–Orszag three mode problem.**

5.2.1. *Problem definition.* The Kraichnan–Orszag three mode problem [12] is known to fail in a short time when employing gPC. It therefore represents a challenging benchmark problem, which has been studied in various contexts, such as adaptive multi–element gPC in [14, 16] and TD-gPC



(A) Mean (B) Variance

FIGURE 5.2. $\|\cdot\|_2$–error in time with respect to various orders of chaos expansions with varying number of elements.

in [4]. It is a nonlinear three–dimensional system of ordinary differential equations:

$$\frac{dx_1}{dt} = x_2 x_3, \tag{5.8}$$

$$\frac{dx_2}{dt} = x_3 x_1, \tag{5.9}$$

$$\frac{dx_3}{dt} = -2x_1 x_2, \tag{5.10}$$

with $x_1(t=0) = \alpha + 0.01\zeta$, $x_2(t=0) = 1.0$ and $x_3(t=0) = 1.0$, subject to a uniformly distributed random variable $\zeta \sim U(-1,1)$. It is known [4, 14] that the critical range of α for which there is a strong dependency on the initial conditions is given by $(0.9, 1)$. Our analysis is therefore focusing on $\alpha = 0.995$.

5.2.2. *Numerical results.* When employing TD-gPC in each element, we set the discretization parameter $Q = 0$, 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

$$M + 1 = \frac{(P+3)!}{P!3!} = \frac{(P+1)(P+2)(P+3)}{6} = O(P^3). \tag{5.11}$$

This illustrates the necessity of keeping the order P of the expansion low, such that the resulting number of modes $M+1$ remains low. For our computations we used a Runge–Kutta solver of 4^{th} order to solve the deterministic part with a time step $\Delta t = 0.001$. Furthermore, we employ an equidistant refinement of the interval $(-1, 1)$ resulting in $N = 2^i$ elements for each refinement level i. Relative errors are measured in the L^2–norm defined in (5.6),(5.7). 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. 5.2 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 5.1. 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.

6. Conclusions

The possible convergence breakdown in cases involving strong nonlinear dependencies on the random input for gPC as introduced in [18], necessitates the development of improved (w.r.t. computational cost and accuracy) Polynomial Chaos methods. This work, based on [6], provides new extensions and modifications of two established variants of gPC, namely the time–dependent generalized Polynomial Chaos (TD-gPC [4]) and the multi–element generalized Polynomial Chaos (ME-gPC [14]). It was demonstrated that TD-gPC along with new modifications is capable of approximating the solution of stochastic dynamical systems very accurately, however, introducing additional numerical cost, which is due to a significantly increased system size. A reduction of the numerical cost can be achieved by a hybrid combination of ME-gPC and TD-gPC, especially when employing a trivial parallel computation of the independent local sub–problems, which can significantly reduce the number of modes needed to solve each sub–problem accurately. Therefore this combination provides a powerful solving method capable of dealing with strong nonlinear dependencies on the random input, which efficiency is expected to increase for large problem sizes of the corresponding deterministic systems.

Current research is focusing on the extension of (local) TD-gPC to the class of partial differential equations. Due to the additional dependencies of the solution on the space variable, the procedure becomes significantly more complex. However, it is expected that the local approach will lead to an even more effective solver, since the solution of a system of partial differential equations is considerably more expensive than in the case of ordinary differential equations.

Future research will address adaptive local approaches reducing the number of elements needed for achieving a given accuracy.

References

[1] R. Cameron, W. Martin, The orthogonal development of nonlinear functionals in series of Fourier-Hermite functionals, Ann. Math. 48 (2) (1947) 385–392.

[2] M. Doi, T. Imamura, The Wiener-Hermite expansion with time-dependent ideal random function, Progress of Theoretical Physics 41 (2) (1969) 358–366.

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

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

[5] R. G. Ghanem, P. D. Spanos, Stochastic Finite Elements: A Spectral Approach, Springer, New York, 1991.

[6] V. Heuveline, M. Schick, Towards a hybrid numerical method using Generalized Polynomial Chaos for Stochastic Differential Equations, EMCL Preprint Series (2011).
URL `http://www.emcl.kit.edu/preprints/emcl-preprint-2011-03.pdf`

[7] O. P. Le Maître, O. M. Knio, Spectral Methods for Uncertainty Quantification, Springer, 2010.

[8] O. P. Le Maître, O. M. Knio, H. N. Najm, R. G. Ghanem, A stochastic projection method for fluid flow I. Basic formulation, Journal of Computational Physics 173 (2) (2001) 481–511.

[9] O. P. Le Maître, L. Mathelin, Asynchronous time integration for polynomial chaos expansion of uncertain periodic dynamics, Discrete and Continuous Dynamical Systems 28 (1) (2010) 199–226.
[10] O. P. Le Maître, M. T. Reagan, H. N. Najm, R. G. Ghanem, O. M. Knio, A stochastic projection method for fluid flow II. random process, Journal of Computational Physics 181 (1) (2002) 9–44.
[11] M. Loève, Probability Theory, 4th ed., New York.
[12] S. A. Orszag, L. R. Bissonnette, Dynamical properties of truncated Wiener-Hermite expansions, Physics of Fluids 10 (12) (1967) 2603–2613.
[13] S. Tanaka, T. Imamura, The Wiener-Hermite expansion with time-dependent ideal random function. II - The three-mode model problem, Progress of Theoretical Physics 45 (4) (1971) 1098–1105.
[14] X. Wan, G. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, Journal of Computational Physics 209 (2005) 617–642.
[15] X. Wan, G. Karniadakis, Long-term behavior of polynomial chaos in stochastic flow simulations, Computer Methods in Applied Mechanics and Engineering 195 (41-43) (2006) 5582–5596.
[16] X. Wan, G. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, SIAM Journal on Scientific Computing 28 (3) (2006) 901–928.
[17] N. Wiener, The homogeneous chaos, Am. J. Math. 60 (4) (1938) 897–936.
[18] D. Xiu, G. Karniadakis, The Wiener-Askey polynomial chaos for stochastic differential equations, SIAM Journal on Scientific Computing 24 (2) (2002) 619–644.
[19] D. Xiu, G. Karniadakis, Modeling uncertainty in flow simulations via generalized polynomial chaos, Journal of Computational Physics 187 (2003) 137–167.

Engineering Mathematics and Computing Lab (EMCL), Institute for Applied and Numerical Mathematics 4, Karlsruhe Institute of Technology, Fritz-Erler-Str. 23, 76133 Karlsruhe, Germany

Preprint Series of the Engineering Mathematics and Computing Lab

recent issues

No. 2011-03 Vincent Heuveline, Michael Schick: Towards a hybrid numerical method using Generalized Polynomial Chaos for Stochastic Differential Equations

No. 2011-02 Panagiotis Adamidis, Vincent Heuveline, Florian Wilhelm: A High-Efficient Scalable Solver for the Global Ocean/Sea-Ice Model MPIOM

No. 2011-01 Hartwig Anzt, Maribel Castillo, Juan C. Fernández, Vincent Heuveline, Rafael Mayo, Enrique S. Quintana-Ortí, Björn Rocker: Power Consumption of Mixed Precision in the Iterative Solution of Sparse Linear Systems

No. 2010-07 Werner Augustin, Vincent Heuveline, Jan-Philipp Weiss: Convey HC-1 Hybrid Core Computer – The Potential of FPGAs in Numerical Simulation

No. 2010-06 Hartwig Anzt, Werner Augustin, Martin Baumann, Hendryk Bockelmann, Thomas Gengenbach, Tobias Hahn, Vincent Heuveline, Eva Ketelaer, Dimitar Lukarski, Andrea Otzen, Sebastian Ritterbusch, Björn Rocker, Staffan Ronnås, Michael Schick, Chandramowli Subramanian, Jan-Philipp Weiss, Florian Wilhelm: HiFlow^3 – A Flexible and Hardware-Aware Parallel Finite Element Package

No. 2010-05 Martin Baumann, Vincent Heuveline: Evaluation of Different Strategies for Goal Oriented Adaptivity in CFD – Part I: The Stationary Case

No. 2010-04 Hartwig Anzt, Tobias Hahn, Vincent Heuveline, Björn Rocker: GPU Accelerated Scientific Computing: Evaluation of the NVIDIA Fermi Architecture; Elementary Kernels and Linear Solvers

No. 2010-03 Hartwig Anzt, Vincent Heuveline, Björn Rocker: Energy Efficiency of Mixed Precision Iterative Refinement Methods using Hybrid Hardware Platforms: An Evaluation of different Solver and Hardware Configurations

No. 2010-02 Hartwig Anzt, Vincent Heuveline, Björn Rocker: Mixed Precision Error Correction Methods for Linear Systems: Convergence Analysis based on Krylov Subspace Methods

No. 2010-01 Hartwig Anzt, Vincent Heuveline, Björn Rocker: An Error Correction Solver for Linear Systems: Evaluation of Mixed Precision Implementations

No. 2009-02 Rainer Buchty, Vincent Heuveline, Wolfgang Karl, Jan-Philipp Weiß: A Survey on Hardware-aware and Heterogeneous Computing on Multicore Processors and Accelerators

No. 2009-01 Vincent Heuveline, Björn Rocker, Staffan Ronnas: Numerical Simulation on the SiCortex Supercomputer Platform: a Preliminary Evaluation