International Journal for Uncertainty Quantification, 4 (1): 37–61 (2014)

A HYBRID GENERALIZED POLYNOMIAL CHAOS METHOD FOR STOCHASTIC DYNAMICAL SYSTEMS

Vincent Heuveline & Michael Schick*

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

Original Manuscript Submitted: 01/24/2012; Final Draft Received: 08/08/2012

Generalized Polynomial Chaos (gPC) is known to exhibit a convergence breakdown for problems involving strong nonlinear dependencies on stochastic inputs, which especially arise in the context of long term integration or stochastic discontinuities. In the literature there are various attempts which address these difficulties, such as the time–dependent generalized Polynomial Chaos (TD-gPC) and the multielement generalized Polynomial Chaos (ME-gPC), both leading to higher accuracies but higher numerical costs in comparison to the standard gPC approach. A combination of these methods is introduced, which allows utilizing parallel computation to solve independent subproblems. 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.

KEY WORDS: *uncertainty quantification, stochastic ordinary differential equations, Polynomial Chaos, dynamical systems, representation of uncertainty, long term integration*

1. INTRODUCTION

Polynomial Chaos, as initially introduced by Wiener in 1938 [1], 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 nondeterministic parts. In 1947 Cameron and Martin [2] proved that this expansion converges in mean–square for square integrable random processes. Ghanem and Spanos [3] pioneered the application of Polynomial Chaos by a Galerkin projection in context of the finite–element method in the field of solid mechanics in 1991. In the following years the projection method became more popular, leading to a broader range of applications, such as CFD (e.g., [4, 5]). In their paper in 2002, Xiu and Karniadakis [6, 7] 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. [8] proved the convergence of generalized Polynomial Chaos (gPC) for certain probability distributions.

In 2006, Wan and Karniadakis [9–11] developed a multielement 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 solving independent local problems. ME-gPC proved to be efficient in reducing the degree of the nonlinear dependencies but still has some restriction with respect to long term integration. For the case of stable limit cycles Le Maître et al. [12, 13] proposed an asynchronous time integration, which was developed to allow gPC to capture the dynamics of the solution. In [14] dynamical orthogonal field equations are developed, which derive equations to compute an optimal

*Correspond to Michael Schick, E-mail: michael.schick@kit.edu, URL: http://www.emcl.kit.edu/

2152–5080/14/$35.00

basis in the deterministic and stochastic space by an orthogonality criteria. In their paper in 2010, Gerritsma et al. [15] introduced a discrete time–dependent approach (TD-gPC) for ordinary differential equations subject to uniformly distributed random inputs, which allows to gain a good accuracy with a possible overhead regarding the 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 [16, 17]. 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 a further development of the time–dependent approach regarding systems of stochastic ordinary differential equations. Thereby, the TD-gPC is extended by a modified version, which maintains an explicit dependency on the initially introduced random variables by maintaining an orthogonal basis. This allows for the application of TD-gPC to systems of ordinary differential equations involving uncertain parameters within the governing equations. However, the numerical cost is significantly increased, which is not only due to a blown-up system size but also to a necessarily large number of numerical quadrature points for maintaining accuracy over time. To be able to employ a parallel numerical computation and therefore distribute the numerical cost a hybrid approach is developed combining TD-gPC with a domain decomposition method.

This work is structured in the following way:

Section 2 gives a short review of gPC, an introduction to TD-gPC, and a modified TD-gPC approach. Section 3 investigates the application of the modified TD-gPC to a system of differential equations followed by an analysis of the numerical integration problems in Section 4. Section 5 recapitulates the multielement approach and introduces its combination with time–dependent basis functionals. Numerical results for the hybrid method are provided in Section 6 followed by conclusions drawn from this work in Section 7.

2. TIME–DEPENDENT GENERALIZED POLYNOMIAL CHAOS

First we recall the main results related to the standard gPC method and depict some issues related to nonlinear dependencies on the random input.

2.1 Generalized Polynomial Chaos

As developed by Xiu and Karniadakis [6], the generalized Polynomial Chaos method (gPC) represents an extension to the original Polynomial Chaos method, initially introduced by Wiener [1]. It is a stationary, i.e., time–independent spectral method for 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.,

$$\mathbb{E}(X^2) = \int_{\Omega} X^2 \, d\mathbb{P} < \infty. \tag{1}$$

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

$$X(\omega) = \sum_{i=0}^{\infty} x_i \psi_i[\zeta(\omega)]. \tag{2}$$

Here, ζ denotes a (possibly multidimensional) random variable subject to some probability distribution, 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_ζ of ζ, i.e.,

$$\langle \psi_i, \psi_j \rangle := \int_{\Omega} \psi_i(\zeta)\psi_j(\zeta) \, d\mathbb{P} = \int \psi_i(z)\psi_j(z) f_\zeta(z) \, dz = \langle \psi_i, \psi_i \rangle \delta_{ij}. \tag{3}$$

TABLE 1: Askey scheme for selecting polynomials corresponding to certain types of distributions

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	$\{0, 1, 2, \ldots, N\}$

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 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

$$X(t; \omega) = \sum_{i=0}^{\infty} x_i(t) \psi_i[\zeta(\omega)], \tag{4}$$

with time dependent modes $x_i = x_i(t)$. Of course, since infinite sums are numerically intractable, (4) needs to be truncated at some finite integer M, resulting in an approximation X^M of X defined by

$$X^M(t; \omega) := \sum_{i=0}^{M} x_i(t) \psi_i[\zeta(\omega)]. \tag{5}$$

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 multidimensional 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 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 whose distribution is known a priori. Both parameters combined result in the corresponding truncation parameter M, via

$$M + 1 = \frac{(P + L)!}{P!L!}. \tag{6}$$

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 has to be discretized further by an appropriate discretization method, which in the literature is often referred to as the "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 in 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 projection 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

$$\mathcal{L}(u,t;\omega) = f(t;\omega), \tag{7}$$

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

$$\mathcal{L}(u,t;\zeta) = f(t;\zeta), \tag{8}$$

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 (8) leading to

$$\mathcal{L}\left(\sum_{i=0}^{M} u_i \psi_i, t; \zeta\right) = f(t;\zeta). \tag{9}$$

There exist several possibilities of solving (9), e.g., a least–squares approach or collocation methods. In this work we focus on 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, (9) is multiplied by ψ_j for $j = 0, \ldots, M$ and integrated by means of the inner product $\langle\cdot\rangle$ on $\mathcal{V}$, resulting in

$$\left\langle \mathcal{L}\left(\sum_{i=0}^{M} u_i \psi_i, t; \zeta\right), \psi_j \right\rangle = \langle f, \psi_j \rangle, \quad j = 0, \ldots, M. \tag{10}$$

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

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 clearly be seen 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., [9, 11, 12, 15, 18, 19], 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. [15], called the time–dependent generalized Polynomial Chaos (TD-gPC).

Recall the gPC approximation u^M of u given by

$$u^M(t;\zeta) = \sum_{i=0}^{M} u_i(t)\psi_i(\zeta). \tag{11}$$

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

$$\eta := u^M(t^*;\zeta) = \sum_{i=0}^{M} u_i(t^*)\psi_i(\zeta). \tag{12}$$

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 almost linear for short times $t \geq t^*$, which is leading to an efficient representation of u via gPC in terms of η. 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 (orthogonal) basis functionals the Chaos Polynomials are recomputed for every change of variables, such that

$$\int \psi_i^{(new)}(\eta)\psi_j^{(new)}(\eta) f_\eta(\eta)\, d\eta = \langle \psi_i^{(new)}, \psi_i^{(new)} \rangle \delta_{ij}, \tag{13}$$

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. Therefore, the integration in (13) is transformed to the original random variable ζ in the following way:

$$\int \psi_i^{(new)}(\eta)\psi_j^{(new)}(\eta) f_\eta(\eta)\, d\eta = \int \psi_i^{(new)}[u^M(t^*;\zeta)]\psi_j^{(new)}[u^M(t^*;\zeta)] f_\zeta(\zeta)\, d\zeta, \tag{14}$$

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

$$u_j^{(new)}(t^*) = \sum_{i=0}^{M} u_i^{(old)}(t^*) \frac{\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle}{\langle \psi_j^{(new)}, \psi_j^{(new)} \rangle}, \quad j = 0, \ldots, M, \tag{15}$$

whereas

$$\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle = \int \psi_i^{(old)}(\zeta)\psi_j^{(new)}[u^M(t^*;\zeta)] f_\zeta(\zeta)\, d\zeta. \tag{16}$$

The projection step (15) can be simplified further via the relations

$$u_0^{(new)}(t^*) = u_0^{(old)}(t^*), \tag{17}$$

$$u_1^{(new)}(t^*) = 1, \tag{18}$$

$$u_j^{(new)}(t^*) = 0, \quad \text{for } j = 2, \ldots, M, \tag{19}$$

since we are dealing with a first-order expansion at time $t = t^*$, whereas we use the convention that the initial polynomial of degree 0 is defined by $\psi_0^{(new)} := 1$ and the polynomial $\psi_1^{(new)}$ is defined with leading coefficient equal to 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

$$\bar{u}(t) = u_0(t), \tag{20}$$

$$\sigma^2(u)(t) = \sum_{i=1}^{M} u_i(t)^2 \langle \psi_i^{(new)}, \psi_i^{(new)} \rangle, \tag{21}$$

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

2.4 Modified TD-gPC

When dealing with system equations $\mathcal{L}(u) = f$, which involve an explicit dependence of the right-hand side f on the random input ζ, the basis as constructed in the previous section might not be sufficient. This issue is illustrated by the following example:

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

$$\frac{d^i}{dt^i} f(u;\zeta) = (-1)^{i+1}\zeta^{i+1}u, \quad i \geq 0. \tag{22}$$

Now, if we consider the Taylor expansion of u we arrive at

$$u(t^* + \Delta t;\eta) = \eta - \Delta t\zeta u(t^*;\eta) + \frac{(\Delta t)^2}{2}\zeta^2 u(t^*;\eta) + \dots. \tag{23}$$

From this it can be seen that a basis in terms of η cannot approximate the time evolution of u in an exact way due to the explicit dependence of f on ζ. 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 ζ via a tensor product, such that u can be expressed by the expansion

$$u(t;\omega) = \sum_{i=0}^{P}\sum_{j=0}^{Q} u_{ij}(t)\psi_i(\eta)\phi_j(\zeta), \tag{24}$$

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 w.r.t. a system of differential equations and give numerical results, which display the improved convergence behavior.

3. SYSTEM OF DIFFERENTIAL EQUATIONS—A LINEAR OSCILLATOR

To the knowledge of the authors the approach described in (24) has not been considered for a system of differential equations. Our goal in this section is to adapt this approach in that context. This is exemplified considering a linear oscillator. For this problem it is well known that the standard gPC expansion fails to capture the dynamics of the solution after some certain time [12]. To overcome this issue Le Maître et al. [12] 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 modified accordingly.

3.1 Model Equations

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

$$\frac{d}{dt}x_1(t) = x_2(t), \tag{25}$$

$$\frac{d}{dt}x_2(t) = -qx_1(t), \tag{26}$$

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

$$x_1(t;\zeta) = \cos\left[\sqrt{q(\zeta)}t\right], \tag{27}$$

$$x_2(t;\zeta) = -\sqrt{q(\zeta)}\sin\left[\sqrt{q(\zeta)}t\right]. \tag{28}$$

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 standard gPC discretization of x_1 and x_2 given by

$$x_1(t;\zeta) = \sum_{i=0}^{P} x_i^{(1)}(t) L_i(\zeta), \tag{29}$$

$$x_2(t;\zeta) = \sum_{i=0}^{P} x_i^{(2)}(t) L_i(\zeta), \tag{30}$$

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 is equal 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

$$\eta^{(1)}(\zeta) := \sum_{i=0}^{P} x_i^{(1)}(t^*) L_i(\zeta), \tag{31}$$

$$\eta^{(2)}(\zeta) := \sum_{i=0}^{P} x_i^{(2)}(t^*) L_i(\zeta). \tag{32}$$

From this point on, we are dealing with a multidimensional 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 outlined in [15] to maintain orthogonality of the multidimensional 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

$$\int \psi_s^{(i)}(\eta)\psi_r^{(i)}(\eta) f_{\eta^{(i)}}(\eta)\, d\eta = \langle \psi_s^{(i)}, \psi_s^{(i)} \rangle \delta_{sr}, \quad s, r = 0, \ldots, P,\ i = 1, 2. \tag{33}$$

This can be achieved, for example, by employing a Gram–Schmidt orthogonalization method. Note that computing the integral in (33) 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

$$x_k(t;\eta^{(1)};\eta^{(2)}) = \sum_{0 \le i+j \le P} x_{k,ij}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)}), \quad t \ge t^*, k = 1, 2, \tag{34}$$

which alternatively can be expressed by

$$x_k(t;\eta^{(1)};\eta^{(2)}) = \sum_{j=0}^{M} x_j^{(k)}(t)\phi_j(\eta^{(1)}, \eta^{(2)}), \quad t \ge t^*, k = 1, 2, \tag{35}$$

by a one–to–one correspondence between the basis functionals and coefficients, whereas for the number $M + 1$ of terms in (35) it holds

$$M + 1 = \frac{(P+2)!}{P!2!} = \frac{(P+1)(P+2)}{2}. \tag{36}$$

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

$$\psi_0 := 1, \tag{37}$$

$$\psi_i(\eta^{(1)}, \eta^{(2)}) := \phi_i(\eta^{(1)}, \eta^{(2)}) - \sum_{j=0}^{i-1} \frac{\langle \phi_i, \psi_j \rangle}{\langle \psi_j, \psi_j \rangle} \psi_j(\eta^{(1)}, \eta^{(2)}), \quad i = 1, \ldots, M. \tag{38}$$

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

$$x_j^{(new)}(t^*) = \sum_{i=0}^{M} x_i^{(old)}(t^*) \frac{\langle \psi_i^{(old)}, \psi_j^{(new)} \rangle}{\langle \psi_j^{(new)}, \psi_j^{(new)} \rangle}, \quad \text{for } j = 0, \ldots, M. \tag{39}$$

Since the Gram–Schmitdt orthogonalization exhibits difficulties regarding numerical stability, we suggest using the "twice is enough" modification, as stated in [20].

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. However, it is also possible to start with a lower expansion and add the new required terms initialized with 0 when needed. 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 by the classical gPC via

$$\bar{x}_i(t) = x_0^{(i)}(t), \tag{40}$$

$$\sigma^2(x_i)(t) = \sum_{j=1}^{M} \left(x_j^{(i)} \right)^2 \langle \psi_j, \psi_j \rangle, \tag{41}$$

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

3.3 Application of the 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:

$$\frac{d}{dt} x_2 = -q(\zeta) x_1. \tag{42}$$

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

$$x_2(t + \Delta t; \zeta) = g[x_1(t; \zeta), x_1(t + \Delta t; \zeta), t, \zeta], \tag{43}$$

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

$$x_1(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{0\le i+j\le P}^{P} \sum_{k=0}^{Q} x_{ijk}^{(1)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\zeta), \tag{44}$$

$$x_2(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{0\le i+j\le P}^{P} \sum_{k=0}^{Q} x_{ijk}^{(2)}(t)\psi_i^{(1)}(\eta^{(1)})\psi_j^{(2)}(\eta^{(2)})L_k(\zeta), \tag{45}$$

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

$$x_i(t;\eta^{(1)},\eta^{(2)},\zeta) = \sum_{j=0}^{M} x_j^{(i)}(t)\psi_j(\eta^{(1)},\eta^{(2)},\zeta), \quad t \ge t^*, i = 1,2, \tag{46}$$

with the number of terms given by

$$M+1 = \frac{(P+1)(P+2)(Q+1)}{2}. \tag{47}$$

The calculation of the initial values at time $t = t^*$ is carried out using the projection described in (39). 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

$$q(\zeta) := 4\pi^2(1 + 0.2\zeta). \tag{48}$$

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 $[0, 75]$. To reduce the errors arising from the numerical integration we employ a Gauss–Legendre quadrature rule with 100 quadrature points. 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 Figs. 1 and 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 standard 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 standard 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 the context of TD-gPC. If one compares the results for $P1Q1$ and $P2Q1$ there are no significant error 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 standard gPC and therefore lead to unfeasible results. This emphasizes the importance of an optimal basis both for the solution itself as well as for its time derivative.

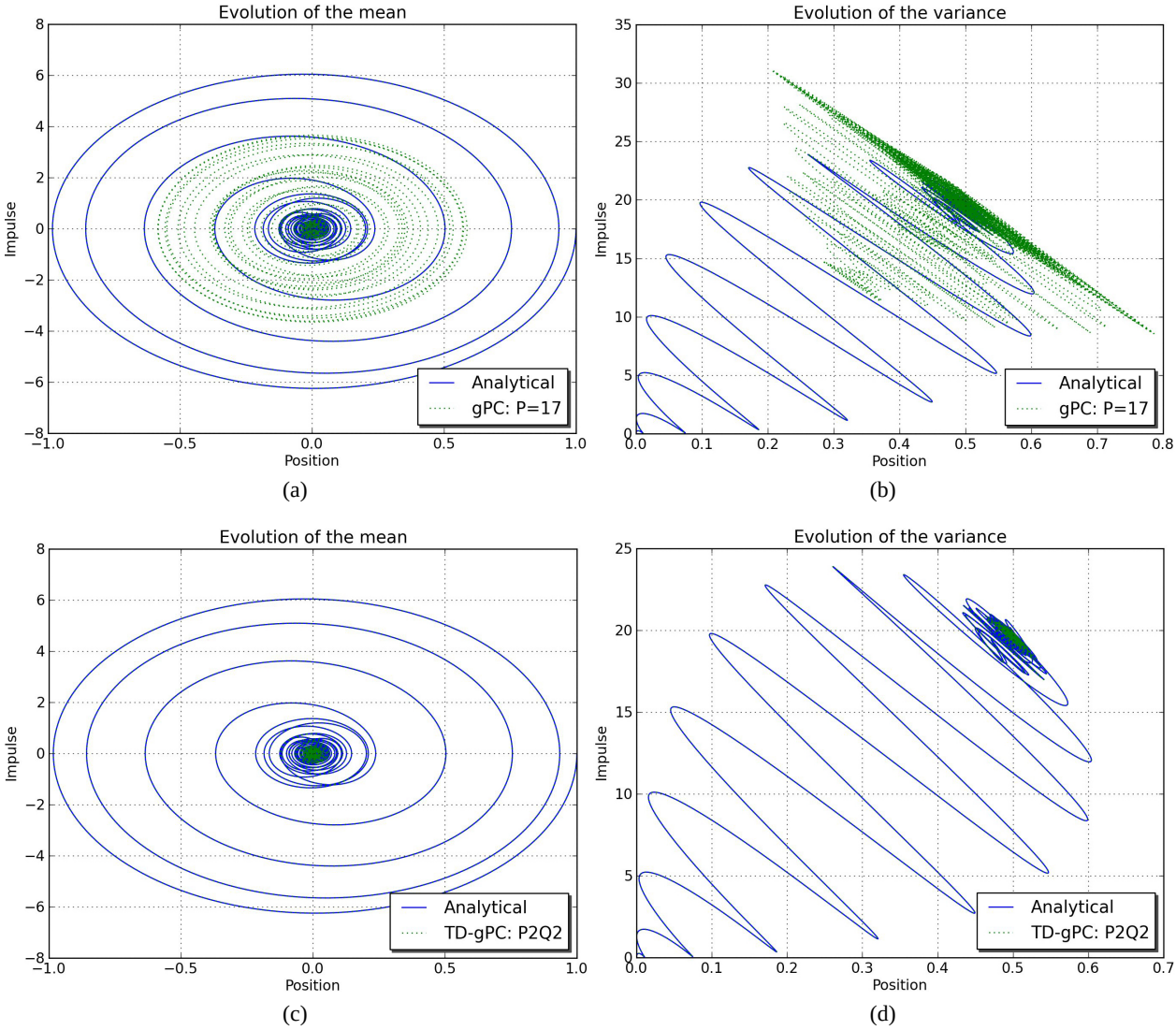


FIG. 1: Evolution of the trajectories corresponding to the mean and the variance for the standard gPC employing $P = 17$ modes (total 18) and the time–dependent variant employing $P = 2, Q = 2$. (a,b) Standard legendre chaos $P = 17$ and (c,d) time–dependent variant $P = 2, Q = 2$.

4. ANALYSIS OF THE RESET STEPS

4.1 Numerical Quadrature

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, we elaborate our analysis on the first reset step at time $t = t^*$ and consider the nonmodified TD-gPC version. Our goal is to evaluate the inner product

$$\langle \psi_i \psi_j, \psi_k \rangle = \int \psi_i(\eta)\psi_j(\eta)\psi_k(\eta) f_\eta(\eta)\, d\eta, \tag{49}$$

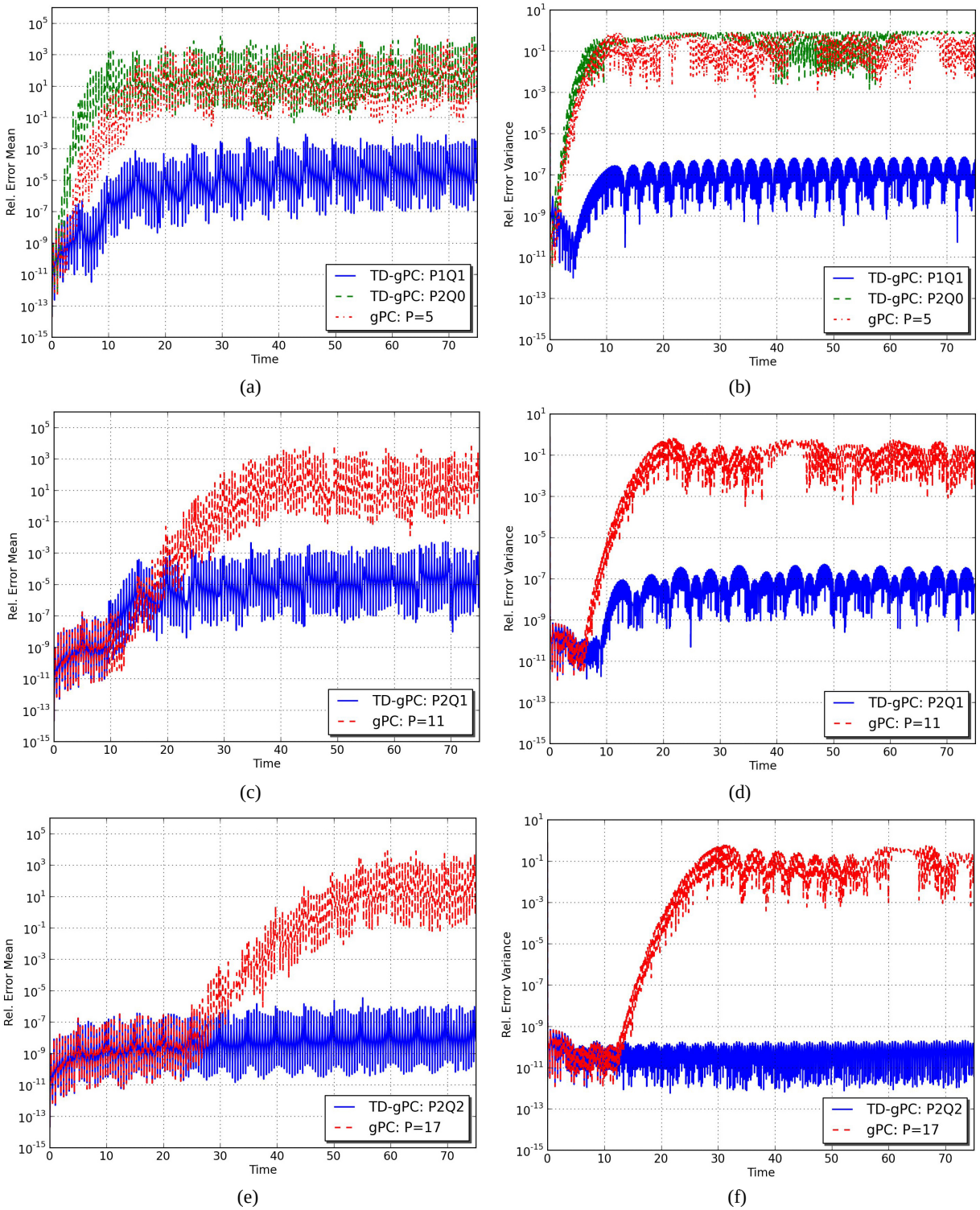


FIG. 2: Relative errors of mean and variance of x_1 corresponding to various discretization parameters. Relative error mean with a total of (a) 6 modes each, (c) 12 modes each, and (e) 18 modes each and Relative error variance with a total of (b) 6 modes each, (d) 12 modes each, and (f) 18 modes.

for some $i, j, k \in \{0, \ldots, M\}$. As described in the previous section, we transform the integral in (49) to the original random variable ζ via the relation

$$\langle \psi_i \psi_j, \psi_k \rangle = \int \psi_i[u(t^*;\zeta)]\psi_j[u(t^*;\zeta)]\psi_k[u(t^*;\zeta)] f_\zeta(\zeta)\, d\zeta, \tag{50}$$

whereas u denotes the solution of the underlying system of ODEs, u being expressed via the relation

$$u(t;\zeta) = \sum_{i=0}^{M} u_i(t)\tilde{\psi}_i(\zeta), \tag{51}$$

for $t < t^*$, where $\tilde{\psi}_i$ denotes the Chaos Polynomials before the reset. Here, without loss of generality, we assume the same expansion order M before and after the reset step at time $t = t^*$ (note that unnecessary modes can be set equal to 0). Estimating (50) is carried out by utilizing an appropriate Gaussian quadrature rule according to the probability distribution represented by f_ζ of ζ. Therefore, we arrive at

$$\langle \psi_i \psi_j, \psi_k \rangle \approx \sum_{n=1}^{N_q} w_n (\psi_i \circ u)(\zeta_n)(\psi_j \circ u)(\zeta_n)(\psi_k \circ u)(\zeta_n), \tag{52}$$

where N_q denotes the number of quadrature points, w_n the corresponding weights, and ζ_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 (50) for all $i, j, k \in \{0, \ldots, M\}$ the well established result for Gaussian quadrature rules applies, leading to the condition

$$N_q = \lceil \frac{3M^2 + 1}{2} \rceil. \tag{53}$$

Applying this analysis to further reset times, we arrive at

$$N_q = \lceil \frac{3M^{N_r} + 1}{2} \rceil, \tag{54}$$

where 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 resetting 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 (50) do 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.

4.2 Automatic Reset Criteria

As described in the previous section, the numerical integration of the random quantities within the initial probability space can become increasingly demanding on the accuracy of the employed quadrature scheme. However, increasing the number of quadrature points employed will have a significant impact on the numerical cost involved when computing a new basis within a reset step. One way to address this problem is to minimize the needed amount of reset steps within one solution of a stochastic dynamical system. This can be achieved by defining a variance based automatic reset criteria in the following form:

Given a threshold parameter $\theta \in (0, 1)$ and a solution $u(t;\eta,\zeta) = \sum_{i=0}^{P}\sum_{j=0}^{Q} u_{ij}(t)\psi_i(\eta)\phi_j(\zeta)$ at time $t > 0$, perform a reset if the following condition holds:

$$\frac{\sum_{i\in\mathcal{I}} u_{i0}(t)\|\psi_i\|^2_{L^2(\Omega)}}{\sigma(t)^2} > \theta, \tag{55}$$

whereas $\sigma^2(t)$ denotes the variance of the solution at time $t > 0$ and $\mathcal{I} := \{i \in [0, P] \cap \mathbb{N} : \psi_i \text{ is nonlinear}\}$. The idea is to monitor the dominance of the nonlinear part of the solution's dependency on the reset variable η with respect to the variance. In our numerical computations this criteria was able to reduce the total amount of reset steps significantly. An alternative to this criteria can be found in [15], which directly measures the magnitude of the nonlinear modes with respect to the linear modes. However, we still choose to apply a reset at every discrete time step without using an automatic criteria. The reason is that for highly dynamical processes too few reset steps can result in significantly increasing numerical errors, such that a solution cannot be captured accurately anymore.

Therefore a new approach is needed to reduce the number of quadrature points to keep the computational costs as low as possible. In this work, this is achieved by a hybrid combination of a domain decomposition in probability space and the TD-gPC. This has the benefit, that due to smaller stochastic variations in subelements of the probability space, a smaller number of quadrature points suffices if the number of elements in the domain decomposition is large enough. But since a solution for every element can be computed independently from each other, the computational costs can be computed by an embarrassingly parallel way. The following section will address this hybrid formulation and first demonstrate its convergence behavior for one-dimensional probability spaces and afterwards it is applied to more complex higher–dimensional problems for which the reduction in the number of quadrature points is demonstrated.

5. HYBRID GENERALIZED POLYNOMIAL CHAOS

A major drawback of the (modified) TD-gPC is the fast–growing number of modes resulting from even low expansion orders P and Q and the accompanying increase in the number of quadrature points to ensure an accurate numerical integration. Therefore, 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, we want to introduce a local approach to TD-gPC motivated by the multielement generalized Polynomial Chaos introduced by Wan and Karniadakis [9, 10], 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 subproblem compared to solving the global problem.

5.1 Domain Decomposition

Following the procedure described in [9, 10] 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

$$B = \bigcup_{j=1}^{N} B_j, \quad B_{j_1} \cap B_{j_2} = \emptyset \quad \text{for } j_1 \neq j_2, \tag{56}$$

$$B_j := [a_1^j, b_1^j) \times [a_2^j, b_2^j) \times \cdots \times [a_d^j, b_d^j]. \tag{57}$$

Therefore, B_j defines a multidimensional interval of dimension d for every $j = 1, \ldots, N$. Note that if $\pm\infty \in B_j$ for some j, as 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

$$I_j = \begin{cases} 1 & \text{if } \zeta \in B_j, \\ 0 & \text{otherwise.} \end{cases} \tag{58}$$

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

$$f_j(\zeta^j | I_j = 1) = \frac{f(\zeta^j)}{\mathbb{P}(I_j = 1)}, \tag{59}$$

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

$$\zeta_k^j = \frac{b_k^j - a_k^j}{2} Y_k^j + \frac{b_k^j + a_k^j}{2}, \tag{60}$$

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

$$\bar{f}_j(y_j) = \det \left| \frac{\partial \zeta^j}{\partial y_j} \right| f_j[\zeta^j(y_j) \,|\, I_j = 1] = \frac{f[\zeta^j(y_j)]}{\mathbb{P}(I_j = 1)} \prod_{k=1}^{d} \frac{b_k^j - a_k^j}{2}. \tag{61}$$

5.2 Problem Structure

The strength in decomposing the probability space lies in the independence of the resulting local problems [9, 10]. Therefore, 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

$$\eta_k^j := \sum_{i=0}^{M} u_i^{(j)}(t_k) \psi_i(\eta_{k-1}^j, \zeta^j), \tag{62}$$

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.

5.3 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 $\sigma^2(u^{(j)})$, $j = 1, \ldots, N$. Next, according to Bayes' theorem and the law of total probability [21], the global stochastic moments of order m, denoted by μ_m, can be calculated via

$$\mu_m(u)(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1) \mu_m(u^{(j)})(t). \tag{63}$$

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

$$\bar{u}(t) \approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1) \bar{u}^{(j)}(t). \tag{64}$$

For the variance it holds

$$\begin{aligned}\sigma^2(u)(t) &= \mu_2(u)(t) - \mu_1(u)(t)^2 \\ &\approx \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\mu_2(u^{(j)})(t) - \sum_{j=1}^{N}\sum_{k=1}^{N} \mathbb{P}(I_j = 1)\mathbb{P}(I_k = 1)\mu_1(u^{(j)})(t)\mu_1(u^{(k)})(t) \\ &= \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\left[\sigma^2(u^{(j)})(t) + (\bar{u}^{(j)})^2 - \bar{u}^{(j)}(t)\sum_{k=1}^{N}\mathbb{P}(I_k = 1)\bar{u}^{(k)}(t)\right] \\ &= \sum_{j=1}^{N} \mathbb{P}(I_j = 1)\left\{\sigma^2(u^{(j)})(t) + \bar{u}^{(j)}(t)[\bar{u}^{(j)}(t) - \bar{u}(t)]\right\}. \end{aligned} \tag{65}$$

5.4 Implementation Aspects

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 in a straightforward manner 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 [15] 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 at every time step. A summary of the local TD-gPC algorithm is given in Fig. 3.

6. NUMERICAL RESULTS FOR THE HYBRID APPROACH

In this section we demonstrate the effect of employing the local TD-gPC in the context of some benchmark problems. First we consider 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, which represents the class of

---

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* (60) *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 $\zeta^{(j)}$ 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*

FIG. 3: Local TD-gPC algorithm.

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. Afterwards we consider a multidimensional random input for the Kraichnan–Orszag three-mode problem and a chemical system exhibiting an oscillatory solution to analyze the effect of the hybrid approach on the number of employed quadrature points.

6.1 A Simple One–Dimensional ODE

This problem has been studied in various works, e.g., [9, 15], having the advantage that its simplicity allows one to calculate an analytical solution. The governing equations are given by

$$\frac{du}{dt} = -k(\zeta)u, \tag{66}$$

$$u(0) = 1, \tag{67}$$

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

$$u(t;\zeta) = \exp(-k(\zeta)t), \tag{68}$$

$$\bar{u}(t) = \frac{1-\exp(-t)}{t}, \tag{69}$$

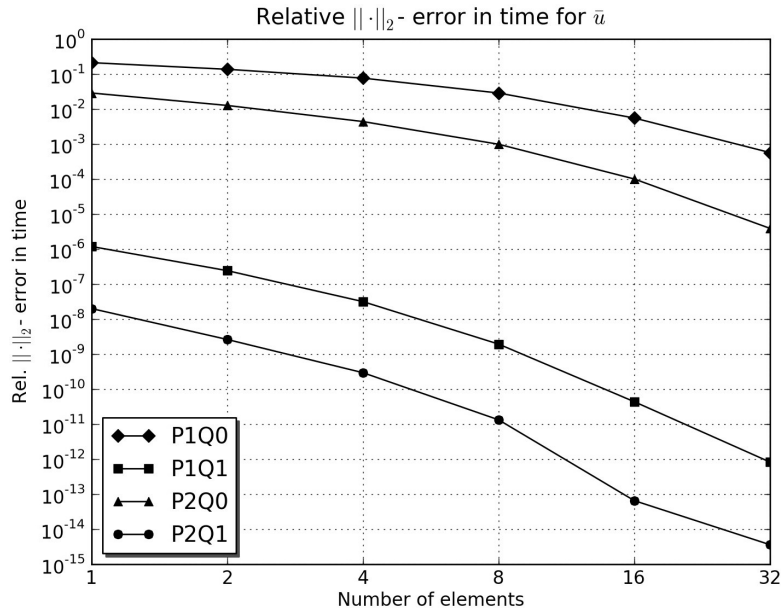
$$\sigma^2(u)(t) = \frac{1}{2t}[1-\exp(-2t)] - \left[\frac{1-\exp(-t)}{t}\right]^2, \tag{70}$$

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, [15]). 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 fourth order with a time step of $\Delta t = 0.001$ to minimize the errors arising from the time discretization. The errors are measured in the discrete euclidean norm $\|\cdot\|_2$, i.e.,

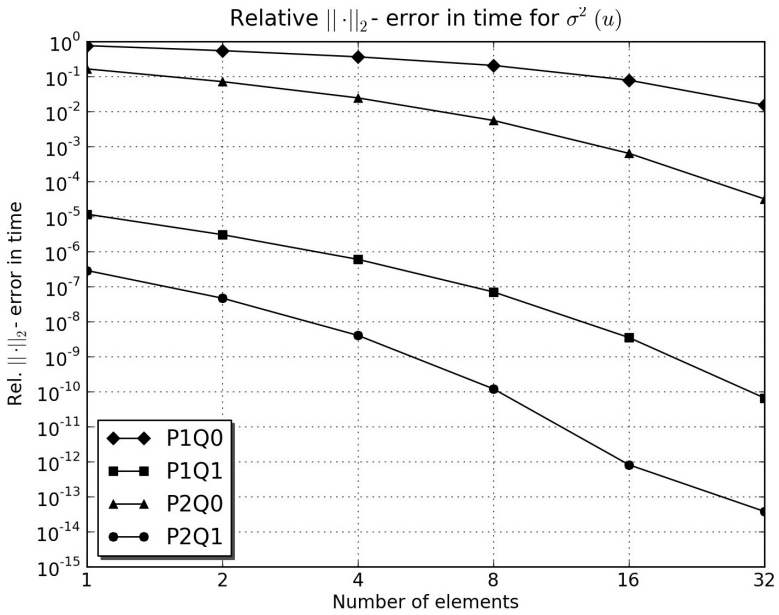
$$\|\bar{u} - \bar{u}_{exact}\|_2 = \sqrt{\left\{\sum_n \left[\bar{u}(t_n) - \bar{u}_{exact}(t_n)\right]^2\right\}}, \tag{71}$$

$$\|\sigma^2(u) - \sigma^2(u_{exact})\|_2 = \sqrt{\left\{\sum_n \left[\sigma^2(u)(t_n) - \sigma^2(u_{exact})(t_n)\right]^2\right\}}, \tag{72}$$

for all discrete time steps $t_n = n\Delta t$ within the simulation interval $[0, 100]$. Figure 4 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 multielement gPC in [9]. 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.



(a)



(b)

FIG. 4: $\|\cdot\|_2$–error in time with respect to various orders of chaos expansions with varying number of elements. (a) Mean and (b) variance.

6.2 The Kraichnan–Orszag three-mode problem

6.2.1 Problem Definition

The Kraichnan–Orszag three-mode problem [22] 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 multielement gPC in [9, 10] and TD-gPC in [15]. It is a nonlinear three–dimensional system of ordinary differential equations:

$$\frac{dx_1}{dt} = x_2 x_3, \tag{73}$$

$$\frac{dx_2}{dt} = x_3 x_1, \tag{74}$$

$$\frac{dx_3}{dt} = -2 x_1 x_2, \tag{75}$$

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 [9, 15] 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 focused on $\alpha = 0.995$ and the computation is carried out within the time interval $[0, 40]$.

6.2.2 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 is given by

$$M + 1 = \frac{(P+3)!}{P!3!} = \frac{(P+1)(P+2)(P+3)}{6} = O(P^3). \tag{76}$$

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 use a Runge–Kutta solver of fourth 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 (71),(72). 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 $P = 3$.

Figure 5 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 6.1. From this it follows that when utilizing parallel computation of the subproblems on each element (which is possible in a trivial way due to the independence of the subproblems) 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 $M + 1$ depends 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 subproblems even further for larger systems of differential equations.

6.2.3 A Three–Dimensional Random Input

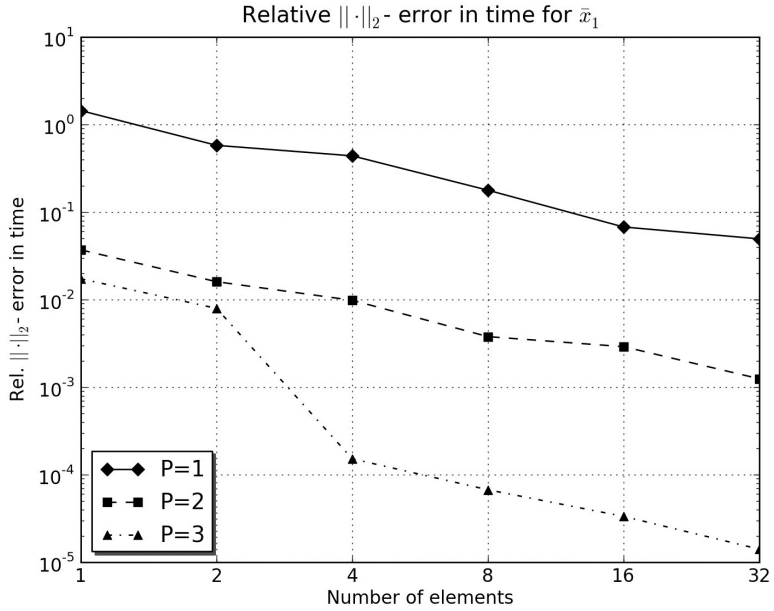
Up to now we demonstrated the successful application of the local TD-gPC, which preserves the exponential convergence rate of the domain decomposition approach and significantly increases the accuracy compared to the standard gPC approach. Here, we want to study the effect of the hybrid formulation on the needed amount of quadrature points for numerical integration.

For this purpose the initial conditions are defined to be random in every variable, i.e.,

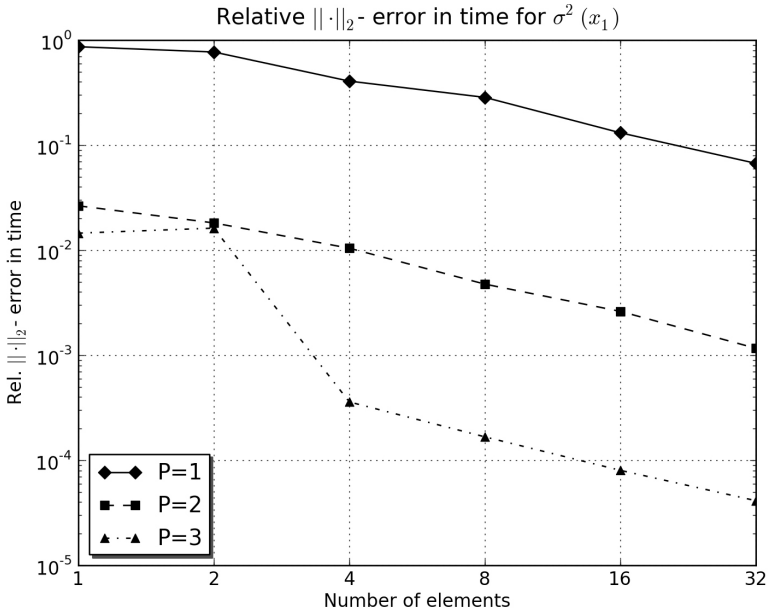
$$x_1(t=0) = \alpha + 0.01\zeta_1, \quad x_2(t=0) = \beta + 0.01\zeta_2, \quad x_3(t=0) = \gamma + 0.01\zeta_3, \tag{77}$$

whereas $\alpha = 0.99,\ \beta = \gamma = 1.0$. The ζ_i are chosen to be identically distributed and independent random variables with uniform distribution within $(-1, 1)$ each. We employ an explicit Runge–Kutta method of fourth order with a time step size $\Delta t = 0.001$ within the time interval $[0, 40]$ and a Gaussian quadrature rule in every random dimension, whereas a reset step is performed at every discrete time step.

As can be seen in Fig. 6 the number of employed quadrature points has a significant impact on the accuracy of the solution compared to a Monte Carlo computation employing 100,000 samples. The results were computed by a



(a)



(b)

FIG. 5: $\|\cdot\|_2$–error in time with respect to various orders of chaos expansions with varying number of elements. (a) Mean and (b) variance.

Gaussian quadrature rule employing 4 quadrature points and 20 quadrature points in each dimension. The 4-point rule looses accuracy after a short simulation time, whereas the 20-point rule exhibits a significantly higher accuracy over the whole time integration interval. However, the computational cost involved for the 20-point rule is significantly higher than for the 4-point rule, since due to a three–dimensional random space, every reset step needs to evaluate multiple integrals with either $20^3 = 8000$ or $4^3 = 256$ total quadrature points. Figure 7 depicts the effect of the

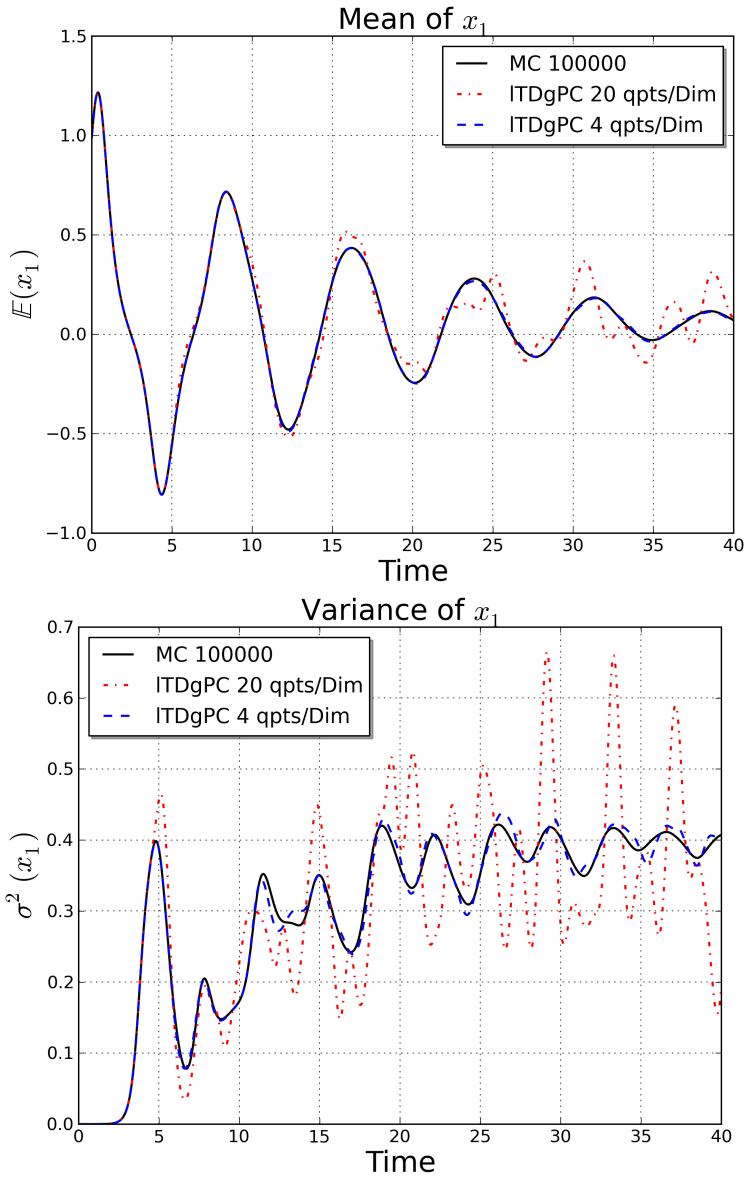
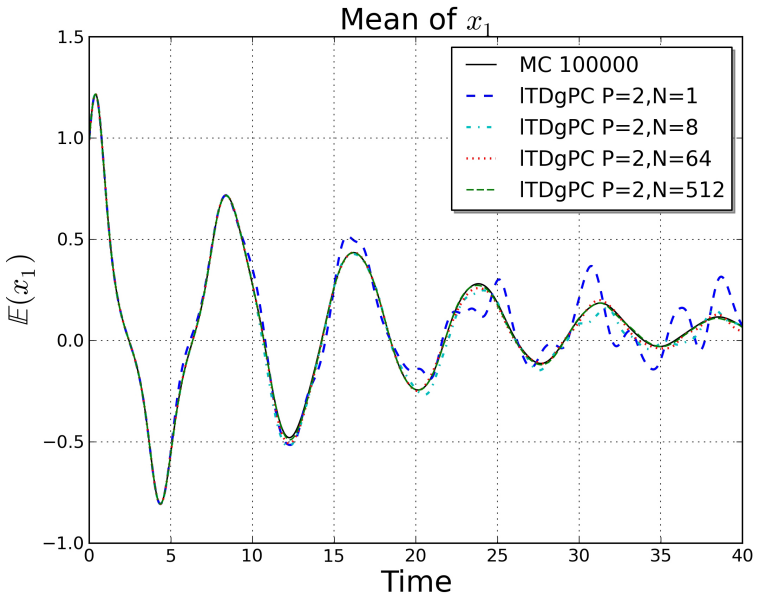


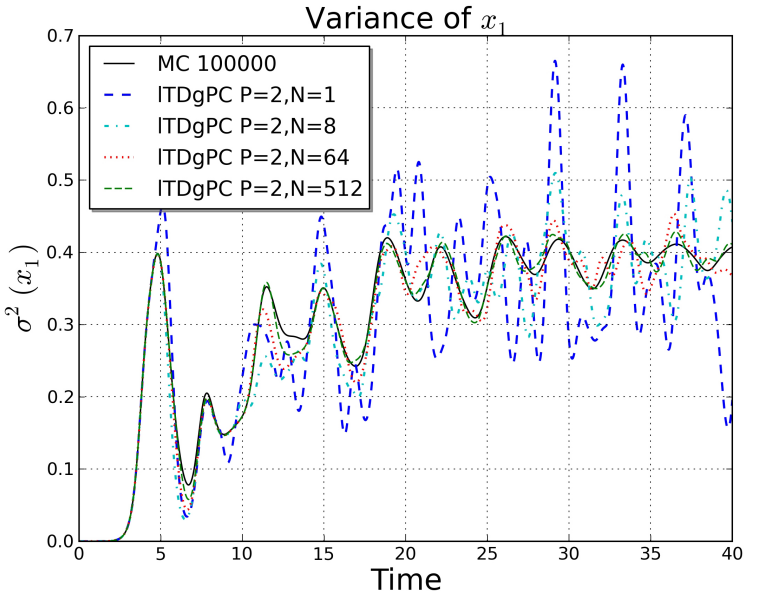
FIG. 6: Computed mean and variance of x_1 with respect to varying number of quadrature points for a second order (non-modified) TD-gPC expansion.

application of the domain decomposition. Here it can be observed that the errors resulting from a poor approximation quality of the quadrature rule is reduced quickly by increasing the number of elements employed. For 512 elements the mean is approximated accurately over the whole time integration interval, whereas for the variance more elements are needed to achieve convergence to accurate results. However, due to the parallel computation of the independent subproblems in each element, the total computational time is significantly reduced.

Remark 1. *This hybrid method surely would benefit from an adaptive domain decomposition based on variance estimate criteria. However, the combination of an adaptive procedure is quite complex due to the necessity of refining and coarsening the domain associated with the time-dependent random variables. This complex issue will be addressed in future publications.*



(a)



(b)

FIG. 7: Computed mean and variance of x_1 with respect to varying number of elements and 4 quadrature points in each random dimension for a second order (non-modified) TD-gPC expansion. (a) Mean and (b) variance.

6.3 A Three–Dimensional Chemical System

Here we want to extend the hybrid approach to a system of so-called Oregonator chemical reactions exhibiting an oscillatory dynamical behavior. This was also studied in various works; see, for example, [12, 23]. It is a three–dimensional system of ordinary differential equations with five parameters, of which two are modeled as random variables. The governing equations read

$$\frac{dx_1}{dt} = k_1 x_2 - k_2 x_1 x_2 + k_3 x_1 - k_4 x_1^2, \tag{78}$$

$$\frac{dx_2}{dt} = -k_1 x_2 - k_2 x_1 x_2 + k_5 x_3, \tag{79}$$

$$\frac{dx_3}{dt} = k_3 x_1 - k_5 x_3, \tag{80}$$

with deterministic initial conditions $x_1(t=0) = x_2(t=0) = x_3(t=0) = 6000$. The parameters are defined as

$$k_1 = 2, \quad k_2 = 0.1, \quad k_3 = 104, \tag{81}$$

and

$$k_4(\zeta_1) = 0.008(1.0 + 0.1\zeta_1), \tag{82}$$

$$k_5(\zeta_2) = 26(1.0 + 0.1\zeta_2), \tag{83}$$

whereas ζ_1 and ζ_2 are identically distributed and independent random variables each subject to a uniform distribution within the interval $(-1, 1)$.

In contrast to the Kraichnan–Orszag three-mode problem, the uncertainty is directly introduced within the governing equations, which necessitates the application of the modified TD-gPC version. The numerical computations are carried out within the time interval [0,20] and a 10-point Gaussian quadrature rule is used in each random dimension. The reset is carried out at every discrete time step subject to the time step size $\Delta t = 0.001$ solved by an explicit Runge–Kutta method of fourth order. Furthermore, a second-order TD-gPC expansion is considered for the reset random variable and a first-order expansion for the initially introduced random variables ζ_1 and ζ_2 (notation $P2Q1$).

Figure 8 depicts the numerical results for the mean and the variance of every solution component on the time interval [10,20]. The standard gPC approach quickly fails in approximating the stochastic moments after a short time (around $t = 1.5$), which is not explicitly depicted in the plots. However, the poor approximation quality of the standard gPC approach can be observed within the time interval [10,20].

A convergence of the modified TD-gPC to the Monte Carlo results can be observed both for the mean and the variance when increasing the number of employed elements from 1 to 64. Since the number of quadrature points in each dimension is quite low and the numerical computations were carried out in parallel, the total numerical cost was distributed effectively. This displays the efficiency of the hybrid approach also for the modified TD-gPC version.

7. CONCLUSIONS

The possible convergence breakdown in cases involving strong nonlinear dependencies on the random input for gPC as introduced in [6] necessitates the development of improved (w.r.t. computational cost and accuracy) Polynomial Chaos methods. This work, based on [15, 16], provides new extensions and modifications of two established variants of gPC, namely the time–dependent generalized Polynomial Chaos (TD-gPC [15]) and the multielement generalized Polynomial Chaos (ME-gPC [9]) in the context of a hybrid approach. These are based on maintaining an orthogonal basis representation in time by coupling newly defined random variables in terms of the solution itself and the initially introduced random variables in the context of systems of ordinary differential equations.

It was demonstrated that TD-gPC, along with the extensions introduced in this work, is capable of approximating the solution of stochastic dynamical systems very accurately; however, it introduces significant additional numerical cost, which is mainly due to a increased system size along with a high requirement on the number of employed quadrature points for numerical integration of the random quantities. A reduction of the numerical cost can be achieved by a hybrid combination of ME-gPC and TD-gPC, especially when employing an embarrassingly parallel computation of the independent local subproblems, which can significantly reduce the number of modes and along the amount of quadrature points needed to solve each subproblem accurately.

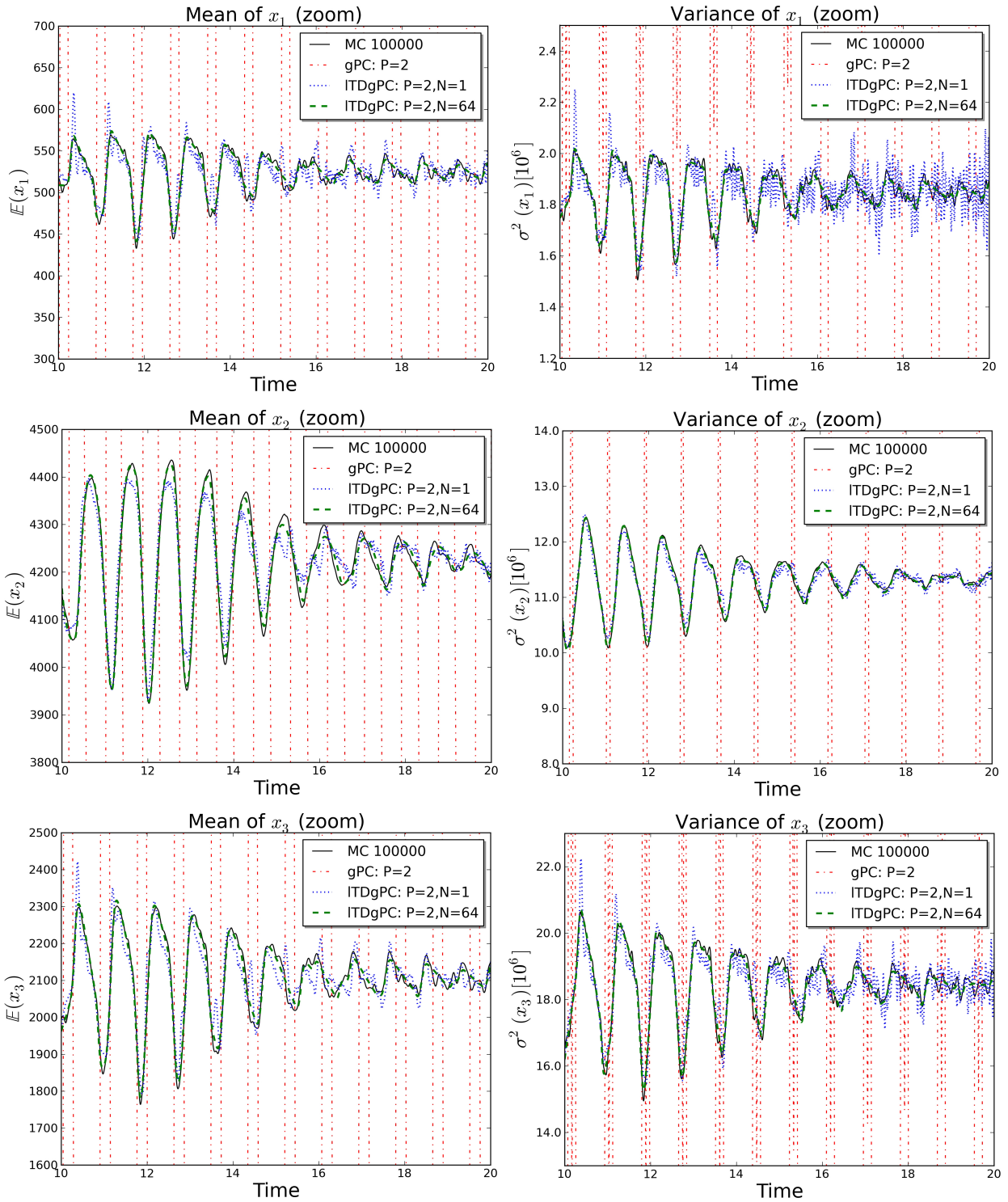


FIG. 8: Results for the mean and variance of the three components x_1, x_2 and x_3. Zoom on time interval [10,20]. Comparison of Monte Carlo 100,000 samples, standard gPC second order expansion and the hybrid modified TD-gPC $P2Q1$ for 1 and 64 elements.

Current research is focusing on improving the accuracies in the numerical integration and an 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.

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

ACKNOWLEDGMENTS

This research was supported by the Engineering Mathematics and Computing Lab (EMCL) at the Karlsruhe Institute of Technology (KIT), Germany.

REFERENCES

1. Wiener, N., The homogeneous chaos, *Am. J. Math.*, 60(4):897–936, 1938.
2. Cameron, R. and Martin, W., The orthogonal development of nonlinear functionals in series of Fourier-Hermite functionals, *Ann. Math.*, 48(2):385–392, 1947.
3. Ghanem, R. G. and Spanos, P. D., *Stochastic Finite Elements: A Spectral Approach*, New York: Springer, 1991.
4. Le Maître, O. P., Knio, O. M., Najm, H. N., and Ghanem, R. G., A stochastic projection method for fluid flow I. Basic formulation, *J. Computat. Phys.*, 173(2):481–511, 2001.
5. Le Maître, O. P., Reagan, M. T., Najm, H. N., Ghanem, R. G., and Knio, O. M., A stochastic projection method for fluid flow II. random process, *J. Comput. Phys.*, 181(1):9–44, 2002.
6. Xiu, D. and Karniadakis, G., The Wiener-Askey polynomial chaos for stochastic differential equations, *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.
7. Xiu, D. and Karniadakis, G., Modeling uncertainty in flow simulations via generalized polynomial chaos, *J. Comput. Phys.*, 187:137–167, 2003.
8. Ernst, O. G., Mugler, A., Starkloff, H.-J., and Ullmann, E., On the convergence of generalized polynomial chaos expansions, Preprint 60, DFG-SPP 1324, 2010.
9. Wan, X. and Karniadakis, G., An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, *J. Comput. Phys.*, 209:617–642, 2005.
10. Wan, X. and Karniadakis, G., Multi-element generalized polynomial chaos for arbitrary probability measures, *SIAM J. Sci. Comput.*, 28(3):901–928, 2006.
11. Wan, X. and Karniadakis, G., Long-term behavior of polynomial chaos in stochastic flow simulations, *Comput. Meth. Appl. Mech. Eng.*, 195(41-43):5582–5596, 2006.
12. Le Maître, O. P. and Mathelin, L., Asynchronous time integration for polynomial chaos expansion of uncertain periodic dynamics, *Discrete Continuous Dynam. Syst.*, 28(1):199–226, 2010.
13. Le Maître, O. P. and Knio, O. M., *Spectral Methods for Uncertainty Quantification*, Dordrecht: Springer, 2010.
14. Sapsis, T. P. and Lermusiaux, P. F., Dynamically orthogonal field equations for continuous stochastic dynamical systems, *Phys. D*, 238:2347–2360, 2009.
15. Gerritsma, M., van der Steen, J.-B., Vos, P., and Karniadakis, G., Time-dependent generalized polynomial chaos, *J. Comput. Phys.*, 229:8333–8363, 2010.
16. Heuveline, V. and Schick, M., Towards a hybrid numerical method using generalized Polynomial Chaos for stochastic differential equations, EMCL Preprint Series, no. 2011-03, 2011.
17. Schick, M., Uncertainty quantification for stochastic dynamical systems: Spectral methods using generalized polynomial chaos, Ph.D. Thesis, 2012.
18. Doi, M. and Imamura, T., The Wiener-Hermite expansion with time-dependent ideal random function, *Prog. Theor. Phys.*, 41(2):358–366, 1969.

19. Tanaka, S. and Imamura, T., The Wiener-Hermite expansion with time-dependent ideal random function. Part II. The three-mode model problem, *Prog. Theor. Phys.*, 45(4):1098–1105, 1971.

20. Giraud, L., Langou, J., and Rozložní, M., On the round-off error analysis of the Gram-Schmidt algorithm with reorthogonalization, CERFACS Technical Report no. TR/PA/02/33, 2011.

21. Loève, M., *Probability Theory*, 4th ed., Springer: New York, 1978.

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

23. Epstein, I. and Pojman, J., An Introduction to Nonlinear Chemical Systems: Oscillations, Waves, Patterns, and Chaos, New York: Oxford University Press, 1998.