MULTIVARIATE POLYNOMIAL CHAOS EXPANSIONS WITH DEPENDENT VARIABLES

JONATHAN FEINBERG* AND HANS PETTER LANGTANGEN†

Abstract. This paper describes with a new approach to handling dependent stochastic variables in polynomial chaos expansions for uncertainty quantification. The methodology is based on a decorrelation algorithm that only requires raw moments of multivariate random variables. When the mapping from input in probability space to the response is not smooth, polynomial chaos expansions may converge slowly. One remedy proposed in this paper is to introduce a transformation of the input parameters to create a smoother mapping in an alternative probability space. However, such a transformation quickly lead to dependent stochastic variables and hence a need for handling dependency.

We consider two cases to demonstrate how variable transformations and the new framework can significantly increase the convergence rate of polynomial chaos expansions. The first case involves an analytical, non-smooth mapping to exemplify serious convergence problems and the power of transforming the variables. The second case concerns diffusion in multi-material/multi-domain models with uncertain internal boundaries and uncertain material properties. We investigate in detail a simplified version of this physical problem where the performance of the method can be understood. Standard polynomial chaos expansions converge very slowly because of uncertain material boundaries, but with a variable transformation, leading to dependent variables, we are able to achieve significantly faster convergence compared with state-of-the-art methods.

Key words. uncertainty quantification, polynomial chaos expansions, dependent stochastic variables, variable transformations, heterogeneous media, uncertain material boundaries

1. Introduction. Polynomial chaos expansions have become a popular approach to uncertainty quantification in forward models, mainly because of the exponential convergence observed in many problems [28]. The number of samples of the underlying mathematical model can then be orders of magnitude smaller than what is required by Monte Carlo-based methods. This feature is of particular importance when the underlying model involves large-scale simulations solving partial differential equations where the computational cost is high. Unfortunately, we have experienced a significant slowdown in the speed of convergence when differential equations have uncertain discontinuous coefficients and uncertain internal boundaries (see Section 5 and [**?**, **?**]). Such coefficients and boundaries naturally arise in multi-material/multi-domain models. This paper presents a new approach, based on stochastic variable transformations, to improve the convergence rate for such problems. Our approach requires a method for handling dependent random variables in polynomial chaos expansions. We therefore need to derive a new framework for working with dependent variables. The paper describes the framework and how it can be applied to restore the attractive convergence properties of polynomial chaos expansions in multi-material diffusion problems.

The method of polynomial chaos expansions was originally defined for standard Gaussian independent random variables through the Hermite polynomial expansion [26]. The theory was extended by the Wiener-Askey scheme to a collection of common random variables [31]. Later, the method was further extended to include arbitrary univariate random variables and multivariate random variables with independent components, through tensor product operations. Some theory has been pro-

*Center of Biomedical Computing, Simula Research Laboratory, and Department of Mathematics, University of Oslo, Norway.

†Center of Biomedical Computing, Simula Research Laboratory, and Department of Informatics, University of Oslo, Norway.

posed to address multivariate random variables with dependent components through transformations [24, 23].

When the mapping from random input to the response is not sufficiently smooth, e.g., because of discontinuities in the mapping itself or its derivatives, a classical polynomial chaos expansion is less suited because of Gibb's phenomena; the result is an expansion with a lower convergence rate as the polynomial degree increases [18]. However, with the use of an appropriately chosen transformation of the random input variables, such problems can be reduced, and in some cases even removed, as will be demonstrated later. The goal of the variable transformation is to create a new mapping which enables the response to be well approximated by a low-order polynomial chaos expansion.

Although some mathematical models do feature stochastically dependent input parameters, such as ion channel densities in detailed neuroscience models [1], most physical models involve input parameters that are stochastically independent. But as mentioned, the convergence rate remedy involving variable transormations will in general introduce dependence among the stochastic variables, even if the physical parameters are independent. For this reason alone, there is a need to be able to deal with dependent stochastic variables in polynomial chaos expansions.

There are two distinct novel features of this paper. First, we derive a framework for multivariate polynomial chaos expansion for dependent variables. This framework is a new general methodology for constructing orthogonal polynomials with respect to any probability distribution, including if it is dependent upon other random variables and physical parameters like time and space. The implementation and application of the framework is straightforward and represent a simplification compared to existing methods for handling dependent variables (this follows from the fact that a Rosenblatt transformation is not required). Throughout the presentation of the framework, we review the literature and compare our method with previous work. Second, the paper proposes the use of auxiliary input variables to reformulate the response in a new probability space. These variables are generated using selected transformations, which are more general and easier to implement than Rosenblatt transformations. The auxiliary variables are mostly stochastically dependent, but this is handled by the aforementioned framework for implementation. Comparison with relevant literature is done as part of the presentation of the variable transformations.

The paper is organized as follows. Section 2 reviews the existing building blocks in non-intrusive polynomial chaos expansions. In Section 3, the new approach for constructing orthogonal polynomials with dependent variables are presented, and in Section 4 the new method for implementing polynomial chaos expansion with transformed variables is introduced. Section 5 applies the new methodology to a physical application with multiple domains, where uncertain domain boundaries pose serious convergence problems for existing polynomial chaos techniques. Finally, we present concluding remarks and remaining challenges.

2. Overview of Methods for Non-Intrusive Polynomial Chaos. The most common method for modeling the uncertainty in large scale problems is the Monte Carlo method. It is easy to apply, requires very few model assumptions to operate, and can treat the full forward problem solver as black box software. The output is on the form of random samples, as if drawn from the distribution of the response, and can be analyzed using a large collection of tools from classical statistics. A very attractive feature of Monte Carlo simulation is that it scales very well with the number of uncertain parameters.

On the other hand, Monte Carlo simulation is known for having a slow convergence rate. The variance estimate is typically $1/K$ times the variance of the model, where K is the number of model evaluations [10]. A forward model based on partial differential equations may easily have a high cost per run, with the consequence that the large number of samples required by Monte Carlo simulation makes the approach computationally infeasible. Polynomial chaos expansions provide an alternative methodology, which may reduce the number of samples of the forward model by orders of magnitude.

2.1. Polynomial Chaos Expansion. Consider a forward model described by a partial differential equation system, here written in the generic form

$$\mathcal{L}(\boldsymbol{x}, t, u, \boldsymbol{\xi}, \boldsymbol{\eta}) = 0, \tag{2.1}$$

where $\mathcal{L}$ is a possibly non-linear differential operator, $\boldsymbol{x} \in \Omega \subseteq \mathbb{R}^Q$ are spatial coordinates and $t \in \mathbb{R}$ is time. Together with $\mathcal{L}$ we associate appropriate boundary and initial conditions. Let $u(\boldsymbol{x}, t; \boldsymbol{\eta}, \boldsymbol{\xi})$ be the solution of the equation, where $\boldsymbol{\eta}$ and $\boldsymbol{\xi}$ denote model parameters. Furthermore let $\boldsymbol{\eta} = (\eta_0, \ldots, \eta_{R-1}) \in \mathbb{R}^R$ be all parameters known with certainty and let $\boldsymbol{\xi} = (\xi_0, \ldots, \xi_{D-1}) \in \mathbb{R}^D$ be the uncertain parameters. We assume that the uncertainty can be described through a joint probability density function $p_{\boldsymbol{\xi}}$. The goal is to compute statistics of a response quantity d derived from u. For simplicity, we take d as u itself in the following and aim to compute the spatial and temporal variation of the average and standard deviation of u. Nothing in the framework prevents d from being any functional of u.

Given that the response, here u, depends smoothly on $\boldsymbol{\xi}$, it can be approximated through a truncated polynomial chaos expansion [14] in the stochastic variables $\boldsymbol{\xi}$. In our case, we seek to compute the expansion

$$\hat{u}_M(\boldsymbol{x}, t; \boldsymbol{\eta}, \boldsymbol{\xi}) = \sum_{i=0}^{N} c_i(\boldsymbol{x}, t; \boldsymbol{\eta}) \boldsymbol{\Phi}_i(\boldsymbol{\xi}), \tag{2.2}$$

where $\{\boldsymbol{\Phi}_i(\boldsymbol{\xi})\}$ are prescribed orthogonal monic polynomials of increasing order and $\{c_i(\boldsymbol{x}, t; \boldsymbol{\eta})\}$ are unknown coefficients. The expansion is truncated at $N+1$ terms, and the value of $N+1$ is selected such that the expansion includes all polynomials of order M and lower. The resulting number of terms is

$$N + 1 = \frac{(M+D)!}{M!D!}. \tag{2.3}$$

To simplify the notation, let the variables $\boldsymbol{x}$, t and $\boldsymbol{\eta}$ be implicit such that $u(\boldsymbol{\xi})$ is a short form for $u(\boldsymbol{x}, t; \boldsymbol{\eta}, \boldsymbol{\xi})$ and c_i is a short form for $c_i(\boldsymbol{x}, t; \boldsymbol{\eta})$. For a given $\boldsymbol{\eta}$, u and c_i are (usually) evaluated as grid functions according to the numerical discretization used to solve the partial differential equation in (2.1).

Let each polynomial in the expansion $\{\boldsymbol{\Phi}_i\}$ be defined on a weighted function space $L_{\boldsymbol{\xi}}$ equipped with the inner product and norm

$$\langle q, r \rangle_{\boldsymbol{\xi}} = \mathbb{E}[q(\boldsymbol{\xi}) r(\boldsymbol{\xi})] \qquad \|q\|_{\boldsymbol{\xi}} = \sqrt{\langle q, q \rangle_{\boldsymbol{\xi}}}, \tag{2.4}$$

where $\mathbb{E}$ is the mathematical expected value operator with respect to $p_{\boldsymbol{\xi}}$. The polynomials are considered orthogonal with respect to $L_{\boldsymbol{\xi}}$ when satisfying

$$\langle \boldsymbol{\Phi}_i, \boldsymbol{\Phi}_j \rangle_{\boldsymbol{\xi}} = \|\boldsymbol{\Phi}_i\|_{\boldsymbol{\xi}}^2 \delta_{ij}, \tag{2.5}$$

where δ_{ij} is the Kronecker delta ($\delta_{ij} = 1$ when $i = j$ and $\delta_{ij} = 0$ otherwise).

Given the availability of an expansion as defined in (2.2), statistical information can be calculated analytically. For example, the model mean and variance is

$$\mathbb{E}[\hat{u}_M] = c_0 \qquad \text{Var}\,[\hat{u}_M] = \sum_{i=1}^{N} c_i^2 \|\mathbf{\Phi}_i\|_{\boldsymbol{\xi}}^2.$$

For an introduction to polynomial chaos expansions we recommend the book by Xiu [29].

2.2. Construction of Orthogonal Polynomials. Because of the dependency on the density function $p_{\boldsymbol{\xi}}$, orthogonal polynomials as in (2.5) must be chosen differently for each random variable $\boldsymbol{\xi}$. For densities in the so-called Askey-Wilson scheme[2], the corresponding orthogonal polynomial expansion is known analytically. Beyond these known cases, the expansion must usually be constructed such that the orthogonality property holds.

Though there are several options for calculating orthogonal polynomials numerically, most are numerically unstable [12]. The one exception is the discretized Stieltjes procedure [11], which allows for the construction of orthogonal polynomials from a univariate probability density function. To extend the method to the multivariate polynomial expansion $\{\mathbf{\Phi}_i\}$, a tensor product between univariate expansions can be used. In other words, each polynomial is naturally factorized as

$$\mathbf{\Phi}_i(\boldsymbol{\xi}) = \mathbf{\Phi}_{i_0,\ldots,i_{D-1}}(\boldsymbol{\xi}) = \Phi_{i_0 0}(\xi_0) \cdots \Phi_{i_{D-1} D-1}(\xi_{D-1}), \tag{2.6}$$

where $\Phi_{i_d d}$ represents a univariate polynomial of order i_d in dimension d. The index i is selected such that there is a one-to-one relationship between the single index and a multi-index

$$i \sim \boldsymbol{i} = (i_0, \ldots, i_{D-1}). \tag{2.7}$$

It will be assumed that the ordering of the polynomials is *graded reverse lexicographical* (grevlex) [5]. An example of grevlex in two dimensions can be seen in Figure 2.1. For example, the multi-index $\boldsymbol{i} = (1, 1)$ is equivalent to the integer index $i = 4$.

multi-index	single index
$\mathbf{\Phi}_{00}$	$\mathbf{\Phi}_0$
$\mathbf{\Phi}_{10}$ $\mathbf{\Phi}_{01}$	$\mathbf{\Phi}_1$ $\mathbf{\Phi}_2$
$\mathbf{\Phi}_{20}$ $\mathbf{\Phi}_{11}$ $\mathbf{\Phi}_{02}$	$\mathbf{\Phi}_3$ $\mathbf{\Phi}_4$ $\mathbf{\Phi}_5$
$\mathbf{\Phi}_{30}$ $\mathbf{\Phi}_{21}$ $\ldots$	$\mathbf{\Phi}_6$ $\mathbf{\Phi}_7$ $\ldots$

FIG. 2.1. *The mapping between single index and multi-index using graded reverse lexicographical sorting in two dimensions.*

One weakness with using the tensor product rule on orthogonal univariate polynomials is that it retains the orthogonality only if the random vector $\boldsymbol{\xi}$ consists of independent components. If that is not the case, *generalized polynomial chaos* (gPC) can be utilized [31]. This method addresses dependent variables by applying an appropriate mapping

$$T : \boldsymbol{\xi} \in \mathbb{R}^D \mapsto \boldsymbol{\zeta} \in \mathbb{R}^D$$

for creating stochastically independent variables $\boldsymbol{\zeta} = T(\boldsymbol{\xi})$. A polynomial chaos expansion can be performed on the transformed space $L_{\boldsymbol{\zeta}}$ with an expression for $\hat{u}_M$ on the form

$$\hat{u}_M(\boldsymbol{x}, t; \boldsymbol{\eta}, T^{-1}(\boldsymbol{\zeta})) = \sum_{i=0}^{N} \hat{c}_i \boldsymbol{\Phi}_i(\boldsymbol{\zeta}). \tag{2.8}$$

There are more than one mapping to make variables stochastically independent. If the full joint distribution of $\boldsymbol{\xi}$ is known, T is preferably a Rosenblatt transformation [24] defined as

$$T(\boldsymbol{\xi}) = \boldsymbol{F}_{\zeta}^{-1}(\boldsymbol{F}_{\xi}(\boldsymbol{\xi})), \tag{2.9}$$

where

$$\boldsymbol{F}_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = (F_{\xi_0}(\xi_0), F_{\xi_1|\xi_0}(\xi_1) \ldots, F_{\xi_{D-1}|\xi_0,\ldots,\xi_{D-2}}(\xi_{D-1}))$$

are cumulative distribution functions created from the components of a probability density decomposition

$$p_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = p_{\xi_0}(\xi_0) p_{\xi_1|\xi_0}(\xi_1) \cdots p_{\xi_{D-1}|\xi_0,\ldots,\xi_{D-2}}(\xi_{D-1}). \tag{2.10}$$

The inverse distribution functions $\boldsymbol{F}_{\boldsymbol{\zeta}}^{-1} = (F_{\zeta_0}^{-1}, \ldots, F_{\zeta_{D-1}}^{-1})$ are conversely created from $p_{\boldsymbol{\zeta}}$, where $\boldsymbol{\zeta} = (\zeta_0, \ldots, \zeta_{D-1})$ are stochastically independent.

One problem with the Rosenblatt transformation is that even for simple problems, the transformation can quickly become discontinuous and highly non-linear. Xiu and Karniadakis [30] showed that transformations between some standard univariate variables introduce Gibbs phenomena and deterioration of the convergence rate. Therefore, it seems to be no way of treating dependent stochastic variables in polynomial chaos expansions and retain convergence properties. But this is exactly what we aim at with the new method in Section 3.

We remark that if the joint distribution $p_{\boldsymbol{\xi}}$ is unavailable or unfeasible to obtain, approximate methods like the Box-Cox [3] and Nataf [23] transformations might be used. Under the correct conditions these methods can perform as well as the Rosenblatt transformation, but are simpler to set up. However, except for a small set of problems, a transformation from complicated dependencies to a stochastically independent vector might be challenging to construct.

2.3. Estimation of Coefficients. Given that the orthogonal polynomials $\{\boldsymbol{\Phi_i}\}$ are available, a polynomial chaos expansion can be created by calculating the Fourier coefficients $\{c_i\}$ as defined in (2.13). Exact integration is only possible in rare cases, so in general a numerical approximation $\hat{c}_i \approx c_i$ is necessary. This section reviews some of the various approaches.

The most common method for estimating the coefficients $\{c_i\}$ is to use point collocation and least squares minimization [21]. This approach, which we use here, results in a non-intrusive method where the forward solver can be treated as a black box. We select $J+1$ collocation nodes $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_J}$ from some appropriate sampling scheme. For our purpose we select the samples as suggested in [20]: $J+1 = 2(N+1)$ samples from the Hammersley sequence. For each collocation point, we solve the forward problem and evaluate the polynomials. The results are collected in

$$\boldsymbol{U} = \begin{bmatrix} u(\boldsymbol{\xi_0}) \\ \vdots \\ u(\boldsymbol{\xi_J}), \end{bmatrix} \qquad \boldsymbol{P} = \begin{bmatrix} \boldsymbol{\Phi_0}(\boldsymbol{\xi_0}) & \cdots & \boldsymbol{\Phi_N}(\boldsymbol{\xi_0}) \\ \vdots & & \vdots \\ \boldsymbol{\Phi_0}(\boldsymbol{\xi_J}) & \cdots & \boldsymbol{\Phi_N}(\boldsymbol{\xi_J}) \end{bmatrix}. \tag{2.11}$$

The coefficients $\boldsymbol{c}$ can now be estimated by a least squares procedure. Let $\boldsymbol{e} = \boldsymbol{U} - \boldsymbol{P}\boldsymbol{c}$ be the deviations between the observations $\boldsymbol{U}$ and the polynomial expansion $\boldsymbol{P}\boldsymbol{c}$. We find $\boldsymbol{c}$ such that square of the Euclidean norm of $\boldsymbol{e}$ is minimized. The result becomes

$$\hat{\boldsymbol{c}} = (\boldsymbol{P}^T\boldsymbol{P})^{-1}\boldsymbol{P}^T\boldsymbol{U}. \tag{2.12}$$

The described least squares procedure does not require the polynomials $\{\boldsymbol{\Phi_i}\}$ to be orthogonal to converge, but without orthogonality the procedure loses some of its properties [21].

Another approach for estimating the coefficients $\{c_i\}$ is to perform a pseudo-spectral projection [**?**], which can be viewed as a least squares minimization in the weighted function space $L_{\boldsymbol{\xi}}$, resulting in

$$c_i = \frac{\langle u, \boldsymbol{\Phi_i}\rangle_{\boldsymbol{\xi}}}{\|\boldsymbol{\Phi_i}\|^2_{\boldsymbol{\xi}}}. \tag{2.13}$$

In practice, the inner products and norms must be computed by numerical approximations:

$$c_i = \frac{\int u(\boldsymbol{\xi})\boldsymbol{\Phi_i}(\boldsymbol{\xi})p_{\boldsymbol{\xi}}(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi}}{\int \boldsymbol{\Phi_i}(\boldsymbol{\xi})^2 p_{\boldsymbol{\xi}}(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi}} \approx \frac{\sum_{j=0}^{J} w_j u(\boldsymbol{\xi_j})\boldsymbol{\Phi_i}(\boldsymbol{\xi_j})p_{\boldsymbol{\xi}}(\boldsymbol{\xi_j})}{\int \boldsymbol{\Phi_i}(\boldsymbol{\xi})^2 p_{\boldsymbol{\xi}}(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi}}, \tag{2.14}$$

where w_k and $\boldsymbol{\xi_k}$ are weights and nodes in a numerical integration method.

A popular integration rule is optimal Gaussian quadrature [12]. This method has nodes and weights optimized for a known weighting function, which in our case can be selected to be the probability density function $p_{\boldsymbol{\xi}}(\boldsymbol{\xi})$. The approach results in the slightly simplified integration formula

$$c_i \approx \frac{\sum_{j=0}^{J} w_j u(\boldsymbol{\xi_j})\boldsymbol{\Phi_i}(\boldsymbol{\xi_j})}{\int \boldsymbol{\Phi_i}(\boldsymbol{\xi})^2 p_{\boldsymbol{\xi}}(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi}}.$$

The Golub-Welsch algorithm can be used to calculate the weights and nodes in this scheme [16]. Alternatively, one can use more generic schemes like Clenshaw-Curtis quadrature or Gauss-Patterson quadrature [13].

Multivariate quadrature are constructed as tensor products of one-dimensional integration rules, but using a full tensor products will result in a curse of dimensionality [**?**]. This problem can be somewhat mitigated by using Smolyak's sparse grid rule [25, **?**]. In either case, for the optimal Gaussian quadrature to use $p_{\boldsymbol{\xi}}$ as a weighting function, the method requires that the components of $\boldsymbol{\xi}$ are stochastically independent.

Originally, the coefficients $\{c_i\}$ were calculated using the *intrusive Galerkin method* [14]. The method relies on a Galerkin formulation of the equations in the forward model and replacing the exact solution by the polynomial chaos expansion. The result is a system of algebraic equations for the coefficients. Essentially, the methodology requires developing a new PDE solver in high dimension (D).

3. Dependent Orthogonal Polynomials. As noted in the last section, the method for addressing input variable with stochastically dependent compoents is to apply generalized polynomial chaos expansions. However, as will be discussed in Section **??**, this approach does not address the problem of discontinueties in the probability space. To be able to address such discontinueties, we need a new method for constructing multivariate polynomial chaos expansions with dependent variables, which is the topic of the present section.

3.1. Decorrelation Algorithm for Constructing Orthogonal Polynomials. First we define the initial polynomial $\mathbf{\Phi}_0 = 1$. For any polynomial orthogonal to $\mathbf{\Phi}_0$ we have that

$$\mathbb{E}[\mathbf{\Phi}_i] = \mathbb{E}[\mathbf{\Phi}_i \mathbf{\Phi}_0] = \langle \mathbf{\Phi}_i, \mathbf{\Phi}_0 \rangle_{\boldsymbol{\xi}} = 0, \quad i \neq 0.$$

This implies that

$$\mathrm{Cov}(\mathbf{\Phi}_i, \mathbf{\Phi}_j) = \mathbb{E}[\mathbf{\Phi}_i \mathbf{\Phi}_j] - \mathbb{E}[\mathbf{\Phi}_i]\,\mathbb{E}[\mathbf{\Phi}_j] = \mathbb{E}[\mathbf{\Phi}_i \mathbf{\Phi}_j] = \langle \mathbf{\Phi}_i, \mathbf{\Phi}_j \rangle_{\boldsymbol{\xi}} \quad i \neq 0, j \neq 0$$

In other words, the orthogonality definition in (2.5) is equivalent with *uncorrelated* polynomials (when $\mathbf{\Phi}_0$ is not considered). This key observation will now be used to construct polynomials in the multivariate case with dependent variables.

Let $\{v_i\}$ be the sequence of simple monic polynomials,

$$\{v_i(\xi)\} = 1, \xi, \xi^2, \ldots$$

A sequence $\{\boldsymbol{v}_i\}$ of multivariate polynomials can be created by a tensor product

$$\{\boldsymbol{v}_i(\boldsymbol{\xi})\} = \bigotimes_{d=0}^{D-1} \{v_i(\xi_d)\}\,. \tag{3.1}$$

We introduce for convenience a one-to-one mapping between the single index i and the multi-index $\boldsymbol{i} = (i_0, \ldots, i_{D-1})$ and let the mapping be sorted using grevlex. Also, let the expansion be truncated at $N+1$ terms according to (2.3).

Let $\boldsymbol{v}$ be the vector $(\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N)$ and let $\boldsymbol{v} \setminus \boldsymbol{v}_0$ denote $\boldsymbol{v}$ with the first component $\boldsymbol{v}_0 = 1$ removed. The vector $\boldsymbol{v}$ is a vector of dependent, and hence in most cases, correlated stochastic variables. Since any set of uncorrelated polynomials will be orthogonal, we can apply a decorrelation procedure to compute a vector of uncorrelated polynomials $\mathbf{\Phi} \setminus \mathbf{\Phi}_0$:

$$\mathbf{\Phi} \setminus \mathbf{\Phi}_0 = L^{-1}(\boldsymbol{v} \setminus \boldsymbol{v}_0 - \mathbb{E}[\boldsymbol{v} \setminus \boldsymbol{v}_0]) \tag{3.2}$$

where L is defined as the Cholesky factor of the covariance matrix:

$$\mathrm{Cov}(\boldsymbol{v} \setminus \boldsymbol{v}_0) = \Sigma_{\boldsymbol{v} \setminus \boldsymbol{v}_0} = LL^T.$$

It is easy to verify that the polynomials $\mathbf{\Phi} \setminus \mathbf{\Phi}_0$ are uncorrelated with zero mean:

$$\begin{aligned} \mathbb{E}[\mathbf{\Phi} \setminus \mathbf{\Phi}_0] &= L^{-1}\left(\mathbb{E}[\boldsymbol{v} \setminus \boldsymbol{v}_0] - \mathbb{E}[\boldsymbol{v} \setminus \boldsymbol{v}_0]\right) = \mathbf{0} \\ \mathrm{Cov}(\mathbf{\Phi} \setminus \mathbf{\Phi}_0) &= L^{-1}\mathrm{Cov}(\boldsymbol{v} \setminus \boldsymbol{v}_0)\, L^{-T} = L^{-1}LL^TL^{-T} = I. \end{aligned}$$

The polynomial collection $\mathbf{\Phi}$ without $\mathbf{\Phi}_0$ is therefore orthogonal. Since $\mathbf{\Phi}_0$ is uncorrelated with the other polynomials, we can just append $\mathbf{\Phi}_0 = 1$ to get the complete set of orthogonal polynomials.

Despite the fact that $\mathrm{Cov}(\boldsymbol{v} \setminus \boldsymbol{v}_0)$ is by definition symmetric and positive definite, rounding errors may accumulate such that the standard Cholesky decomposition algorithm applied to $\mathrm{Cov}(\boldsymbol{v} \setminus \boldsymbol{v}_0)$ breaks down. To mitigate this, we apply a modified Cholesky decomposition where L is set as $\Sigma_{\boldsymbol{v} \setminus \boldsymbol{v}_0} + E$, E being a diagonal stabilization matrix. The modified decomposition is implemented using the algorithm developed by Gill and King in [15].

3.2. Exemplifying the Polynomial Construction Algorithm. To ensure all details of the procedure described above are clear, consider a specific example where $\boldsymbol{\xi}$ has a bivariate normal distribution with

$$\mathbb{E}[\boldsymbol{\xi}] = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \mathrm{Cov}(\boldsymbol{\xi}) = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix} \tag{3.3}$$

To construct orthogonal polynomials of order at most $M = 2$, we need the following basis, sorted using the grevlex ordering defined in Section 2.2:

$$\{\boldsymbol{v}_i\} = 1, \xi_0, \xi_1, \xi_0^2, \xi_0\xi_1, \xi_1^2,$$

with the following first and second order statistics:

$$\mathbb{E}[\boldsymbol{v} \setminus \boldsymbol{v}_0] = \begin{bmatrix} 0 \\ 0 \\ 1 \\ \frac{1}{2} \\ 1 \end{bmatrix} \qquad \mathrm{Cov}(\boldsymbol{v} \setminus \boldsymbol{v}_0) = \begin{bmatrix} 1 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & \frac{1}{2} \\ 0 & 0 & 1 & \frac{5}{4} & 1 \\ 0 & 0 & \frac{1}{2} & 1 & 2 \end{bmatrix}$$

We can now compute the modified Cholesky decomposition and obtain (to three decimals)

$$L = \begin{bmatrix} 1.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.500 & 0.866 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 1.414 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.707 & 0.866 & 0.000 \\ 0.000 & 0.000 & 0.353 & 0.866 & 1.060 \end{bmatrix}$$

$$L^{-1} = \begin{bmatrix} 1.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ -0.577 & 1.154 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.707 & 0.000 & 0.000 \\ 0.000 & 0.000 & -0.577 & 1.154 & 0.000 \\ 0.000 & 0.000 & 0.235 & -0.942 & 0.942 \end{bmatrix}.$$

Using equation (3.2) we get the following polynomials:

$$\begin{aligned} \boldsymbol{\Phi}_0 &= 1.000 & \boldsymbol{\Phi}_3 &= 0.707\xi_0^2 - 0.707 \\ \boldsymbol{\Phi}_1 &= 1.000\xi_0 & \boldsymbol{\Phi}_4 &= -0.577\xi_0^2 + 1.154\xi_0\xi_1 \\ \boldsymbol{\Phi}_2 &= 1.154\xi_1 - 0.577\xi_0 & \boldsymbol{\Phi}_5 &= 0.942\xi_1^2 - 0.942\xi_0\xi_1 + 0.235\xi_0^2 - 0.707 \end{aligned}$$

Orthogonality can be verified for each polynomial pair. For example for $\boldsymbol{\Phi}_1$ and $\boldsymbol{\Phi}_2$ we have that

$$\begin{aligned} \mathbb{E}[\boldsymbol{\Phi}_1\boldsymbol{\Phi}_2] &= \mathbb{E}[1.000\xi_0(1.154\xi_1 - 0.577\xi_0)] \\ &= \mathbb{E}\big[1.154\xi_0\xi_1 - 0.577\xi_0^2\big] = 0.577 - 0.577 = 0.000. \end{aligned}$$

3.3. Investigating Possible Ill-Conditioning. In Section 2.2, it was mentioned that constructing orthogonal polynomials from moments, as we do here, is an ill-conditioned problem [12]. We therefore employ the mentioned modified Cholesky decomposition to stabilize the construction. This subsection investigates the effectiveness of this stabilization. First, we look at how well the new method performs when

the polynomial order grows and the number of uncertain parameters is held constant. Second, we look at the stability of the method as the number of uncertain parameters grows while the polynomial order is fixed.

We compare the polynomials generated by the new method with polynomials generated from the three terms recursion method (TTR), where the coefficients are calculated analytically [16]. Since the orthogonality property in polynomials is invariant up to multiplication of a scalar, we normalize the polynomials. For TTR, it is possible to calculate the norm of the polynomial in a stable manner using the product rule

$$\|\Phi_i\|_\xi = \sqrt{\prod_{n=0}^{i} B_n},$$

where B_n is the second recursion coefficient in the method. Polynomials calculated by our new method are automatically normalized. To measure the divergence from orthogonality, we look at the maximum absolute value of the inner product of each polynomial pair:

$$\varepsilon = \max_{i \neq j} \left| \langle \mathbf{\Phi}_i, \mathbf{\Phi}_j \rangle_{\boldsymbol{\xi}} \right|.$$

All raw statistical moments used, in our method and TTR, are calculated analytically.

First, we explore the error ε as a function of polynomial order while keeping the number of dimensions at one. We have conducted tests on the following distributions: the standard normal distribution, the standard gamma distribution, a Wigner distribution with unit radius, a uniform distribution on $[-1, 1]$, a beta distribution with parameters $a = 2$ and $b = 4$, and a triangle distribution with bounds at -1 and 1, and top at 0.

Figure 3.1 shows the error on a logarithmic scale for the various distributions. Polynomials generated from the method with a modified Cholesky decomposition are marked as Chol. All ε curves are left truncated since at low enough polynomial orders the error identifies as zero exactly. As the order grows, the errors in orthogonality all start around 10^{-16}, before they grow with exponential rate. The trend for TTR is a steady increasing error with little variation. On the other hand, for Chol the growth is more unstable in a more piecewise linear behavior. For the Wigner, uniform, beta, and triangle distributions, the new method is sometimes as good as TTR, and sometimes better. For the Gamma and Normal distributions, the error ε for Chol is larger than TTR for lower orders. The former method reaches a plateau where the error stagnates. Since the TTR error continues to grow, the Chol error will be lower for higher orders here as well.

The second part of the analysis is to investigate how the error ε is affected by the number of dimensions (D). The polynomial order is now fixed at 4. In Figure 3.2, the orthogonality error ε is plotted as a function of the number of dimensions. The gamma and the Wigner distributions are not displayed, because both of them have error that identifies as zero exact at all dimensions for both TTR and Chol. For Chol there is a growth, but it is small: from 1 to 10 dimensions, the error only grows from 10^{-12} to 10^{-10} for the triangle distribution. For the beta distribution the range is from 10^{-12} to 10^{-11}. The change is much more negligible for the rest of the distributions.

It is worth remarking that the number of polynomial terms N grows quickly with D, according to (2.3) with $M = 4$. For example, at dimension $D = 10$, the number

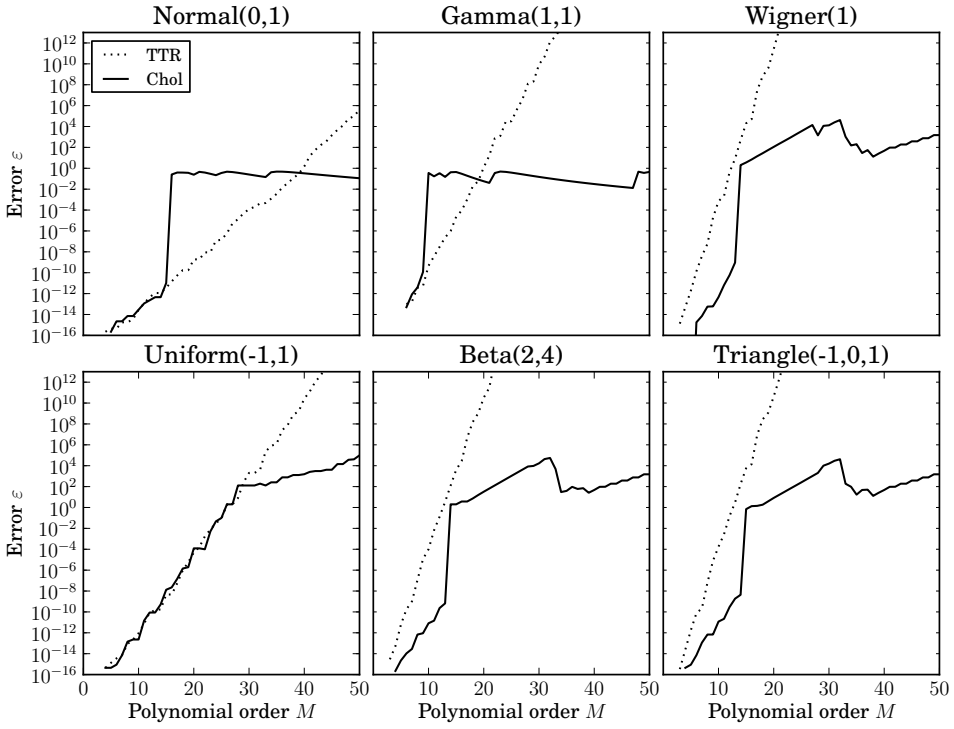

FIG. 3.1. *The absolute error ε in orthogonality arising from three terms recursion method (TTR) and modified Cholesky decomposition method (Chol) for various distributions.*

of polynomial terms is $N + 1 = 1001$, which means that the Cholesky decomposition needs to handle a 1000×1000 matrix. In comparison, if $D = 1$ and $M = 30$, we get a small matrix ($N + 1 = 31$). As illustrated in Figure 3.2, the error in orthogonality is much larger in the latter case, despite involving a much smaller matrix. It is therefore reasonable to assume that even though D dramatically increases the number of polynomial terms and the size the problem, it has little effect on the stability of the modified Cholesky method. Since most physical applications involve low M and large D where the error is small, it seems from the tests (and also our further experience with the new method) that stability is not a significant concern.

4. Using Forward Transforms to Improve Convergence. We shall now describe a problem class where the previously described framework for handling dependent random variables in polynomial chaos expansions will be particularly useful. One major assumption in polynomial chaos theory is that the response is a smooth function of the random variables. In the presence of, e.g., discontinuities in this function, the polynomial chaos expansion will converge with a reduced speed compared to the smooth case. We shall propose a way to improve the convergence by using certain types of variable transformations, which introduce dependent random variables even if the physical input parameters are stochastically independent. With the previously suggested algorithm for constructing multivariate orthogonal polynomials with dependent variables at hand, we can easily explore variable transformations to improve the convergence in problems where there is lack of smoothness in probability

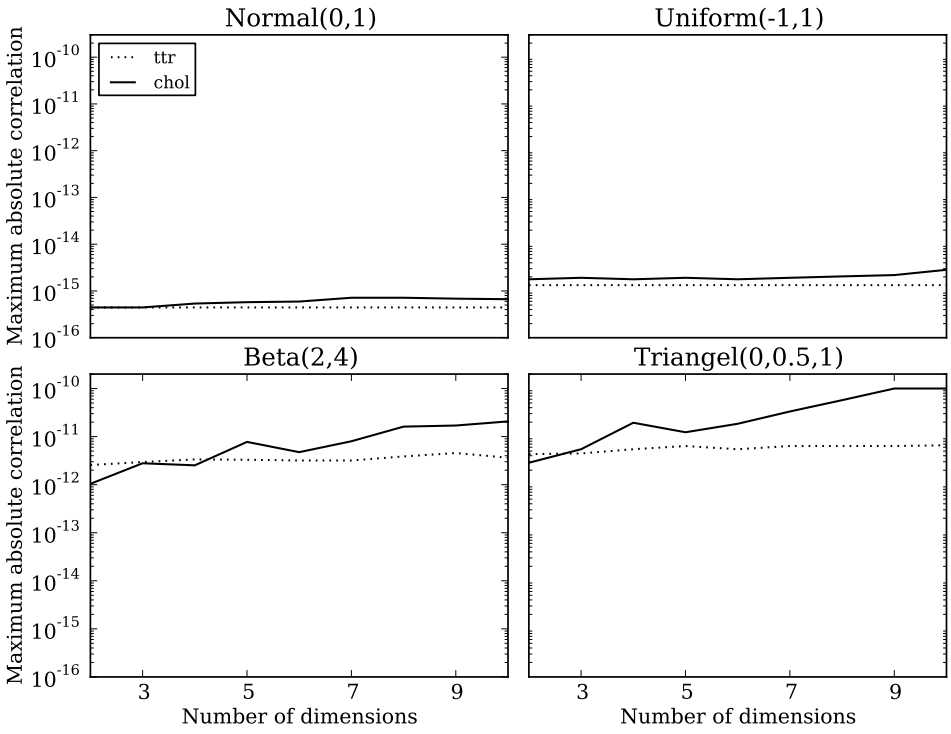

FIG. 3.2. *The absolute error ε in orthogonality arising from Three Terms Recursion (TTR) and modified Cholesky decomposition (Chol) at order 4 for various distributions as a function of the number of dimensions.*

space.

Finding the right effective variable transformation for improving the convergence is highly problem dependent, so a theoretical analysis of the convergence properties seems difficult. We shall therefore proceed with specific examples to illustrate the effectiveness of transformations.

4.1. Variable Transformation. Suppose the response quantity in our model, here u, is not a sufficiently smooth function of $\boldsymbol{\xi}$ for fast convergence of polynomial chaos expansions. A possible remedy is then to introduce a map

$$T : (\boldsymbol{x}, t, \boldsymbol{\xi}, \boldsymbol{\eta}) \in \mathbb{R}^Q \times \mathbb{R} \times \mathbb{R}^D \times \mathbb{R}^R \mapsto \boldsymbol{\zeta} \in \mathbb{R}^C, \tag{4.1}$$

to create an auxiliary random variable $\boldsymbol{\zeta} = (\zeta_0, \ldots, \zeta_{C-1})$. By performing a variable substitution, we can reformulate $u(\boldsymbol{x}, t, \boldsymbol{\xi}, \boldsymbol{\eta})$ to be $u^*(\boldsymbol{x}, t, \boldsymbol{\zeta}, \boldsymbol{\eta})$, i.e., a function in $\boldsymbol{\zeta}$ instead of $\boldsymbol{\xi}$. The hope is that u^* becomes a smooth function in $\boldsymbol{\zeta}$. We can then create a polynomial chaos approximation $\hat{u}^*_M$ with polynomials orthogonal in $L_{\boldsymbol{\zeta}}$ instead of $L_{\boldsymbol{\xi}}$. For convenience we denote $T(\boldsymbol{x}, t, \boldsymbol{\xi}, \boldsymbol{\eta})$ as $T(\boldsymbol{\xi})$.

Many will associate T with a Rosenblatt transformation, but (4.1) is much more general. The transformation can depend upon time t and space $\boldsymbol{x}$, making $\boldsymbol{\zeta}$ a stochastic process. The number of dimensions in the output, C, does not need to match the number of dimensions in the input, D, and the new transformation does not need to be injective or surjective. In comparison, non of these features are shared by the Rosenblatt transformation.

For almost any choice of T, $\boldsymbol{\zeta}$ is going to have stochastically dependent components, irrespectively of any dependencies found in $\boldsymbol{\xi}$. This difficulty of dependent variables can be solved with the new proposed framework for constructing orthogonal polynomials.

With great flexibility in choosing T, calculating the probability density $p_{\boldsymbol{\zeta}}$ from $p_{\boldsymbol{\xi}}$ is most often difficult and infeasible. Even simple arithmetics of random variables often end up in calculations involving complicated convolutions [27]. However, since we only need the raw statistical moments, we do not need to find $p_{\boldsymbol{\zeta}}$, because the raw statistical moments of $\boldsymbol{\zeta}$ can be calculated directly in $L_{\boldsymbol{\xi}}$ space:

$$\mathbb{E}\left[\prod_{d=0}^{D-1} \zeta_d^{k_d}\right] = \int p_{\boldsymbol{\zeta}}(\boldsymbol{\zeta}) \prod_{d=0}^{D-1} \zeta_d^{k_d} \,\mathrm{d}\boldsymbol{\zeta} = \int p_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \prod_{d=0}^{D-1} T_d(\boldsymbol{\xi})^{k_d} \,\mathrm{d}\boldsymbol{\xi}. \tag{4.2}$$

Given the raw moments of $\boldsymbol{\zeta}$, an orthogonal polynomial expansion $\{\boldsymbol{\Psi}_i\}$ can be constructed using the method described in Section 3.2 for each space-time degree of freedom in the response. The corresponding coefficients $\boldsymbol{c}^*$ can thereafter be estimated using a least squares procedure based on point collocation, as in (2.14). The $\boldsymbol{U}$ and $\boldsymbol{P}$ quantities are now

$$\boldsymbol{U} = \begin{bmatrix} u(\boldsymbol{\xi}_0) \\ \vdots \\ u(\boldsymbol{\xi}_J) \end{bmatrix} \qquad \boldsymbol{P} = \begin{bmatrix} \boldsymbol{\Psi}_0(T(\boldsymbol{\xi}_0)) & \cdots & \boldsymbol{\Psi}_N(T(\boldsymbol{\xi}_0)) \\ \vdots & & \vdots \\ \boldsymbol{\Psi}_0(T(\boldsymbol{\xi}_J)) & \cdots & \boldsymbol{\Psi}_N(T(\boldsymbol{\xi}_J)) \end{bmatrix}. \tag{4.3}$$

Comparing this new method to the point collocation method described in Section 2.3, there is an important difference in the computational cost between (2.11) and (4.3). In the former, $\boldsymbol{P}$ and its inverse are evaluated only once, while in the latter, $\boldsymbol{P}$ has to be evaluated and inverted for every time point t and spatial location x. If the space-time resolution in the model response is high, this can result in a much higher computational cost than in gPC. In addition, because of the repeated calculations of variations of the same problem, the possibility of $\boldsymbol{P}$ being ill-posed is greater. Since $\boldsymbol{P}$ needs to be inverted in (2.12), this might introduce numerical instabilities. To alleviate this problem, the inversion is done with a first-order Tikhonov regularization [19]. The size of the regularization parameter is determined using robust generalized cross-validation with $\gamma = 0.1$ as proposed by Lukas [22].

It is worth noting that even though the transformed samples

$$(\boldsymbol{\zeta}_0, \ldots, \boldsymbol{\zeta}_J) = (T(\boldsymbol{\xi}_0), \ldots, T(\boldsymbol{\xi}_J))$$

are used to evaluate $\boldsymbol{P}$, $\boldsymbol{U}$ is evaluated given non-transformed samples $(\boldsymbol{\xi}_0, \ldots, \boldsymbol{\xi}_J)$. In other words, irrespectively of what the transformation is, we only use one set of samples. And because running the forward model is often very time consuming, this is a favorable feature since it allows us to experiment with many variations of T without additional costly evaluations of the forward model.

The first and second order statistics from the alternative expansion $\{\boldsymbol{\Psi}_i\}$ can be calculated from the moments in (4.2) as:

$$\begin{aligned} \mathbb{E}[u] \approx \mathbb{E}[\hat{u}_M^*] &= \mathbb{E}\left[\sum_{i=0}^{N} \hat{c}_i^* \boldsymbol{\Psi}_i\right] = \hat{c}_0^* \\ \operatorname{Var}[u] \approx \operatorname{Var}[\hat{u}^*] &= \operatorname{Var}\left[\sum_{i=0}^{N} \hat{c}_i^* \boldsymbol{\Psi}_i\right] = \sum_{i=1}^{N} (\hat{c}_i^*)^2 \|\boldsymbol{\Psi}_i\|_{\boldsymbol{\xi}}^2. \end{aligned} \tag{4.4}$$

Remark. To some extent it is possible to approximate the response u^* using the tools available in gPC expansions. Nevertheless, problems will arise, because generalized polynomial chaos expansions require the introduction of a secondary transformation S, such that we can create a new stochastically independent vector

$$\boldsymbol{\upsilon} = S(\boldsymbol{\zeta}) = S(T(\boldsymbol{\xi})).$$

Even if the density $p_{\boldsymbol{\zeta}}$ is known, which most likely it is not, finding a transform S such that $\boldsymbol{\upsilon}$ consists of independent components is not trivial. Furthermore, let us assume that T is selected to be a Rosenblatt transformation. One obvious solution would then be to select $S = T^{-1}$. The resulting vector $\boldsymbol{\upsilon}$ will then be stochastically independent since $\boldsymbol{\upsilon} = T^{-1}(T(\boldsymbol{\xi})) = \boldsymbol{\xi}$, but this approach retains the non-smooth properties intact. So, rather than constructing two transformations in gPC expansions, we just construct one and instead deal with dependent variables.

4.2. An Illustrating Example. With the previously suggested algorithm for constructing multivariate orthogonal polynomials with dependent variables at hand, we can explore variable transformations to improve the convergence in problems where there is lack of smoothness in probability space. As a simple example, consider the following non-smooth mapping from two stochastic input variables to the response u:

$$u(t, \xi_0, \xi_1) = |\xi_0| e^{-t\xi_0\xi_1}.$$

Here, ξ_0 and ξ_1 are independent uniform random variables on $[-1, 1]$. Because of the discontinuity in the derivative of the mapping from $\boldsymbol{\xi}$ to u, generalized polynomial chaos expansions in the $\boldsymbol{\xi}$ variables will result in reduced convergence speed. For example, the uppermost curves in Figure 4.1 show that on a log scale, the pseudo-spectral approach where the coefficients are estimated by Gaussian quadrature integration is only marginally faster than Monte Carlo sampling.

The remedy to the convergence problem is to introduce new stochastic variables by choosing a suitable transformation as in (4.1). Preferably, the transformation should result in a smooth mapping from the auxiliary variables $\boldsymbol{\zeta}$ to the response u. One possible transformation is $\zeta_0 = |\xi_0|$ and $\zeta_1 = t\xi_0\xi_1$, because this leads to the mapping

$$u^*(\zeta_0, \zeta_1) = \zeta_0 e^{-\zeta_1},$$

which is smooth. However, the transformation results in ζ_0 and ζ_1 being dependent stochastic variables.

With our new framework we can construct orthogonal polynomials in the new variables $\boldsymbol{\zeta} = (\zeta_0, \zeta_1)$. The raw moments of $\boldsymbol{\zeta}$ can in this case be calculated analytically:

$$\begin{aligned}\mathbb{E}\Big[\zeta_0^i \zeta_1^j\Big] &= \mathbb{E}\big[|\xi_0|^i (t\xi_0\xi_1)^j\big] = \int_{-1}^{1}\int_{-1}^{1} 0.25|\xi_1|^i (t\xi_0\xi_1)^j \,\mathrm{d}\xi_0\,\mathrm{d}\xi_1 \\ &= \frac{t^j}{4}\int_{-1}^{1} |\xi_0|^i \xi_0^j \,\mathrm{d}\xi_0 \int_{-1}^{1} \xi_1^j \,\mathrm{d}\xi_1 = \frac{t^j}{4}\frac{1+(-1)^j}{i+j+1}\frac{1+(-1)^j}{j+1} = \frac{t^j}{2}\frac{1+(-1)^j}{(i+j+1)(j+1)}\end{aligned}$$

A numerical integration scheme can be used when analytical moments are not feasible to calculate.

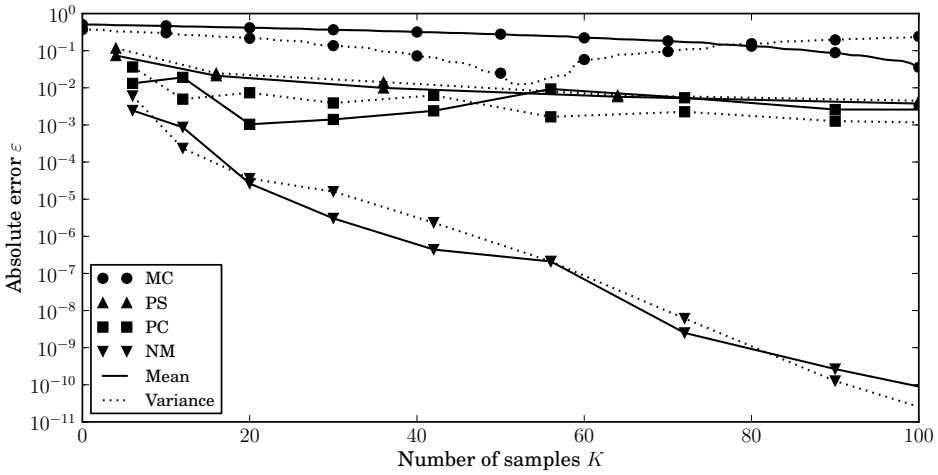

FIG. 4.1. *The L^1 norm in time of the difference between polynomial approximations and the exact model.*

With the raw moments available, an orthogonal polynomial expansion can be created using the method described in Section 3. The orthogonal polynomials up to second order, at $t = 1$ for example, become

$$\begin{aligned}
\mathbf{\Phi}_0 &= 1.000 & \mathbf{\Phi}_3 &= -\ 2.457\xi_0 + \ 0.409 \quad\ + 7.373\xi_1^2 \\
\mathbf{\Phi}_1 &= 3.000\xi_1 & \mathbf{\Phi}_4 &= -11.618\xi_1 + 15.491\xi_0\xi_1 \\
\mathbf{\Phi}_2 &= 3.464\xi_0 - 1.732 & \mathbf{\Phi}_5 &= \ \ 13.647\xi_0^2 - 13.189\xi_0 \quad + 2.198 - 1.373\xi_1^2.
\end{aligned}$$

To measure the approximation errors of the various methods, the L^1 norm integrated over time for the mean and variance is plotted against the number of samples. More precisely, Figure 4.1 shows

$$\varepsilon_E = \int_0^1 |\mathbb{E}[u] - \mathbb{E}[\tilde{u}]\,|\,\mathrm{d}t \qquad \varepsilon_V = \int_0^1 |\mathrm{Var}\,[u] - \mathrm{Var}\,[\tilde{u}]\,|\,\mathrm{d}t, \tag{4.5}$$

where $\tilde{u}$ represent the approximative model by polynomial chaos expansion. In our example the true mean and variance are calculated analytically as

$$\mathbb{E}[u] = \frac{\cosh(t) - 1}{t} \qquad \mathrm{Var}\,[u] = \frac{\cosh(2t)}{t} - \frac{\sinh(2t)}{2t^3} - \frac{\cosh(2t)}{2t^4} + \frac{2\cosh(t)}{t^4}.$$

Figure 4.1 compares four approaches: a quasi-Monte Carlo scheme using Hammersley samples [17], labeled MC; two polynomial chaos expansions with pseudo-spectral projection and point collocation for estimation of the coefficients, labeled PS and PC respectively; and the new proposed method with the suggested variable transformation, labeled NM. The L^1 error was computed from 1000 evenly distributed sampling points over $t \in [0, 1]$. The figure shows that all methods except the new proposed method has an error larger than 10^{-3} within the first 100 samples. On the other hand, the new proposed method reaches as low as 10^{-10} in the same span.

The model problem in this section is quite simple. This is partially to exemplify the steps in the algorithms and partially to demonstrate the impact of smooth and non-smooth mappings in probability space. The next section deals with a physical application.

5. Application to Diffusion in Layered Media. We address single-phase fluid flow in porous, geological formations. The governing partial differential equation

$$\nabla \cdot (K \nabla \phi) = 0, \tag{5.1}$$

where ϕ is the pressure head and K is the permeability tensor, both of which are functions of spatial position. In geological applications, one often views the medium as composed of layers, arising from depositions of different types of sediments through different geological periods. The layers may therefore feature very different physical compositions and values of K. A reasonable assumption is to let K be constant in each layer, and that it is sufficient to distinguish between the permeability of flow parallel to the layer and perpendicular to the layer (which means that the K tensor is diagonal in a coordinate system aligned with the layer). The values of the elements in K between two neighboring layers can differ by many orders of magnitude. We remark that the same mathematical model arises in other physical contexts as well. For example, ϕ may be the temperature in a multi-layered solid material, and K the heat conduction coefficient, which is taken to be constant in each layer.

To better understand why layered media pose difficulties for polynomial chaos expansions in diffusion problems, we simplify and let ϕ depend on the vertical coordinate only. With the z axis pointing upward and the x, y axes pointing in the horizontal directions, we consider a medium consisting of $m+1$ horizontal, parallel layers. We further assume that ϕ is prescribed at the bottom $z = B$ and at the top $z = H$. The flow problem can then be reduced to the model

$$\frac{d}{dz}\left(K(z)\frac{d\phi}{dz}\right) = 0, \quad \phi(B) = \phi_B,\ \phi(H) = \phi_H. \tag{5.2}$$

We scale the problem for convenience by introducing the dimensionless variables

$$x = \frac{z - B}{H - B}, \quad u = \frac{\phi - \phi_B}{\phi_H - \phi_B}, \quad a = \frac{K}{\max_{z\in[B,H} K(z)}.$$

Clearly, x, u, and a vary between 0 and 1. The corresponding dimensionless boundary-value problem reads

$$\frac{d}{dx}\left(a(x)\frac{du}{dx}\right) = 0, \quad u(0) = 0,\ u(1) = 1. \tag{5.3}$$

The exact solution of (5.3) is found to be

$$u(x) = \frac{\int_0^x [a(\tau)]^{-1} d\tau}{\int_0^1 [a(\tau)]^{-1} d\tau}. \tag{5.4}$$

A model problem with this analytical insight is helpful to find remedies to decreased convergence speed of polynomial chaos expansions.

For the described layered medium, $a(x)$ is a piecewise constant function with value a_i in layer i, recognized by $x \in [\ell_i, \ell_{i+1}]$. We can express the $a(x)$ function as

$$a(x) = \begin{cases} a_0 & 0 = \ell_0 \le x < \ell_1 \\ a_1 & \ell_1 \le x < \ell_2 \\ \vdots & \vdots \\ a_m & l_m \le x \le l_{m+1} = 1. \end{cases} \tag{5.5}$$

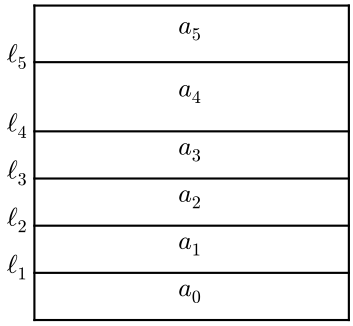

FIG. 5.1. *Sample medium with horizontal layers.*

Figure 5.1 depicts a medium where $m = 5$. The analytical solution corresponding to (5.5) becomes

$$u(x) = \frac{\sum_{j=0}^{i-1}(\ell_{j+1} - \ell_j)/a_j + (x - \ell_i)/a_i}{\sum_{j=0}^{m}(\ell_{j+1} - \ell_j)/a_j} \qquad x \in [\ell_i, \ell_{i+1}). \tag{5.6}$$

5.1. Two-Layered Medium. The most simplified medium where polynomial chaos expansions lead to a decreased convergence rate consists of two layers with individual uncertain permeabilities a_0 and a_1, and an uncertain interval boundary location l_0. This problem setting involves three random variables: $\boldsymbol{\xi} = (\xi_0, \xi_1, \xi_2)$ with $\xi_0 = a_0$, $\xi_1 = a_1$, and $\xi_2 = \ell_1$, as illustrated in Figure 5.2. We assume that the permeability and thicknesses of the layers are completely independent, implying that ξ_0, ξ_1, and ξ_2 are stochastically independent.

If one applies standard polynomial chaos methods to this problem, the accuracy in computed statistics versus the number of deterministic solutions of the forward model will be reduced by an order of magnitude compared with what is to be expected. (Quantitative results appear toward the end of the paper in Figure 5.3.) To repair the performance of the original polynomial chaos method, we shall introduce a variable transformation, but first we need to understand the details of the convergence problem.

The original mapping in probability space from ξ_0, ξ_1, and ξ_2 to the response u becomes

$$u(\xi_0, \xi_1, \xi_2; x) = \left(\frac{\xi_2}{\xi_0} + \frac{1 - \xi_2}{\xi_1}\right)^{-1} \begin{cases} x/\xi_0 & x \in [0, \xi_2) \\ \xi_2/\xi_0 + (x - \xi_2)/\xi_1 & x \in [\xi_2, 1] \end{cases} \tag{5.7}$$

For fixed ξ_2 and x, only one of the formulas describes the entire mapping of ξ_0 and ξ_1 to u, depending on $x < \xi_2$ or $x \geq \xi_2$, and the mapping is a smooth rational function. On the other hand, for fixed x, ξ_0, and ξ_1, we see that the mapping from ξ_2 to u is piecewisely defined and continuous, but with a discontinuity in $\partial u/\partial \xi_2$ at $\xi_2 = x$. The heart of the problem is this lack of smoothness, arising when the internal boundary $\xi_2 = \ell_1$ is uncertain, while uncertain material parameters with fixed layer geometries pose no problems.

Since the flux $-au'$ has to be continuous in this problem, the jump in a is compensated by a jump in u'. The moving discontinuity of u' implies a complicated mapping in probability space, and hence a polynomial is a bad candidate for approximating this mapping. We shall bring evidence to this assertion later in Figure 5.3.

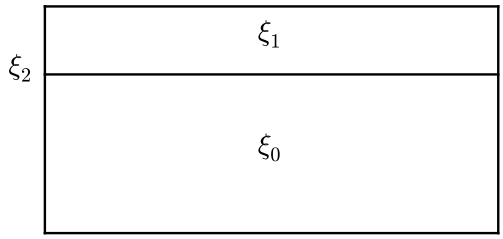

FIG. 5.2. *The two-layer medium used for numerical investigations.*

The uncertain model parameters $\boldsymbol{\xi} = (\xi_0, \xi_1, \xi_2)$ follow simple distributions that are popular in engineering:

(5.8)
$$\xi_0 \sim \text{Triangle}(0.10, 0.15, 0.20) \quad \xi_1 \sim \text{Triangle}(0.8, 0.9, 1.0) \quad \xi_2 \sim \text{Uniform}(0.1, 0.9).$$

The triangle distribution is defined as

$$p_{\tau|a,b,c}(\tau \mid a,b,c) = \frac{2}{(b-a)(c-a)} \begin{cases} \tau - a & \tau \in [a,b) \\ c - \tau & \tau \in [b,c) \\ 0 & \tau \notin [a,c) \end{cases},$$

where (a, b, c) are distribution parameters representing the lower bound, the peak, and the upper bound of the distribution.

The challenge is to construct a transformation $\boldsymbol{\zeta} = T(\boldsymbol{\xi})$ such that the reformulation of $u(\boldsymbol{\xi})$ in terms of $u^*(\boldsymbol{\zeta})$ creates a smooth mapping in probability space. The discontinuity is caused by the integral from 0 to x of the discontinuous function $a(x)$. If the integral itself were a stochastic variable, it would be a linear dependence between the integrand and u without any discontinuity. One idea is to decompose the analytical form of the solution in (5.7). For example it is possible to split it in two:

$$\zeta_0 = \frac{\xi_2}{\xi_0} + \frac{1-\xi_2}{\xi_1} \qquad \zeta_1 = \begin{cases} \frac{x}{\xi_0} & x \in [0, \xi_2) \\ \frac{\xi_2}{\xi_0} + \frac{x-\xi_2}{\xi_1} & x \in [\xi_2, 1) \end{cases} \tag{5.9}$$

leading to the simple form

$$u^*(\boldsymbol{\zeta}) = \frac{\zeta_1}{\zeta_0}, \tag{5.10}$$

which is a smooth function. Since both variables ζ_0 and ζ_1 can be expressed as a function of $\boldsymbol{\xi}$, they can be considered stochastically dependent. Our described framework for multivariate polynomial chaos expansions with dependent variables can easily treat this case. But, unlike the simpler problem described in Section 4.2, we here choose to estimate the raw statistical moments numerically. We employ a 100th order Clenshaw-Curtis quadrature [4] using a full tensor product rule to calculate the Fourier coefficients $\{c_i\}$. The domain is split into smaller domains which are integrated separately using a composite rule. Specifically, the domains are split at the discontinuities in the derivative for both the triangle distributions and at the location $x = \xi_0$.

For more complicated problems, an analytical form of the solution u as in (5.7) is unavailable. Finding the correct transformation T can therefore be more challenging.

However, since the new methodology allows for flexible construction of transformations, it opens up for experimenting with various choices of T. For example, in our problem formulation we know the location of the discontinuity in the derivative of the solution u. This knowledge suggests trying a function that has the discontinuity in the derivative at the same location. As an example, we propose the use of the integral from 0 to x of a as an auxiliary variable:

$$A(x, \boldsymbol{\xi}) = \int_0^x a(\tau, \boldsymbol{\xi}) \, \mathrm{d}\tau.$$

It is likely that u as a function of only A is under-parameterized. Therefore, the proposed transformation is combined with the original variables defined in equation (5.8). However, it is possible to imagine that a parameterization does not require all variables to avoid under-parameterization. To investigate if this possibility is the case, we explore these alternative transformations:

$$\zeta_0 = \xi_0 \qquad \zeta_1 = \xi_1 \qquad \zeta_2 = \xi_2 \qquad \zeta_3 = \int_0^x a(\tau, \boldsymbol{\xi}) \, \mathrm{d}\tau \tag{5.11}$$

$$\zeta_0 = \xi_0 \qquad \zeta_1 = \xi_1 \qquad \zeta_2 = \int_0^x a(\tau, \boldsymbol{\xi}) \, \mathrm{d}\tau \tag{5.12}$$

$$\zeta_0 = \xi_2 \qquad \zeta_1 = \int_0^x a(\tau, \boldsymbol{\xi}) \, \mathrm{d}\tau \tag{5.13}$$

Observe that the last variable in each transformation depends on the former, making the variables stochastically dependent. An interesting question is whether these transformations are inferior to (5.9), which was based on analytical insight that is normally not available. The answer to this question lies in Figure 5.3.

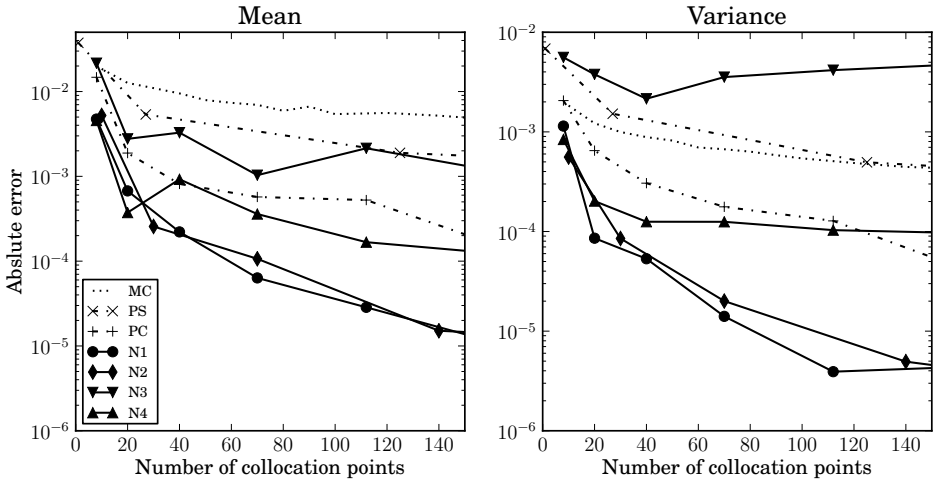

FIG. 5.3. *The absolute error plotted against the number of forward model samples in a diffusion problem with two layers.*

Figure 5.3 compares the performance of the four variable transformations with popular and state-of-the-art methods. Seven curves are listed. The Monte Carlo method is denoted MC. Because of the fluctuations in the method, the solution is

averaged over 100 repeated simulations using bootstrapping [6]. Classical point collocation and pseudo-spectral projection in polynomial chaos expansions are denoted PC and PS, respectively. The former uses samples from the Hammersley sequence, while the latter uses optimal Gaussian quadrature points where $p_{\boldsymbol{\xi}}$ are weighing functions. The various proposed transformations listed in (5.9), (5.11), (5.12) and (5.13) are denoted N1, N2, N3 and N4, respectively. All of these four methods use the same samples as in the classical point collocation scheme. The figure shows the error in mean and variance as defined in (4.5). The reference solution for the mean and variance is estimated using a classical Monte Carlo scheme with 10^9 samples. The approximate mean and variance in the polynomial chaos methods are calculated from (4.4).

Figure 5.3 also shows that all methods converge as the number of samples increases. For the pseudo-spectral method, the results are on par with Monte Carlo, but not much better. The point collocation method, on the other hand, performs much better, with an error that goes down to 10^{-3}. The accuracy of our new methodology varies somewhat. The choice N3, which is based of (5.13), fails to converge in variance. It illustrates that selecting the wrong transformation may result in a bad approximation, likely due to the model being under-parameterized. A better selection is N2, but in that case it does as well as PC. For N1 and N2, the results are about the same. Both cases are much better than the other alternatives and converges with an exponential rate.

5.2. Three-Layered Medium. To illustrate that the method also works for multiple layers, we have increased the problem to three layers. Here we designate the variables as follows:

$$\boldsymbol{\xi} = (\xi_0, \xi_1, \xi_2, \xi_3, \xi_4) = (a_0, a_1, a_2, \ell_1, \ell_2),$$

with the distributions

$$\begin{aligned}
\xi_0 &\sim \text{Triangle}(0.10, 0.15, 0.20) \\
\xi_1 &\sim \text{Triangle}(0.8, 0.9, 1.0) \\
\xi_2 &\sim \text{Triangle}(0.10, 0.15, 0.20) \\
\xi_3 &\sim \text{Uniform}(0.1, 0.45) \\
\xi_4 &\sim \text{Uniform}(0.55, 0.9).
\end{aligned} \tag{5.14}$$

The mapping in probability space from $\boldsymbol{\xi}$ to the response u becomes

(5.15)

$$u(\xi_0, \xi_1, \xi_2, \xi_3, \xi_4) = \left(\frac{\xi_3}{\xi_0} + \frac{\xi_4 - \xi_3}{\xi_1} + \frac{1 - \xi_4}{\xi_2} \right)^{-1} \begin{cases} \frac{x}{\xi_0} & x \in [0, \xi_3) \\ \frac{\xi_3}{\xi_0} + \frac{x - \xi_3}{\xi_1} & x \in [\xi_3, \xi_4) \\ \frac{\xi_3}{\xi_0} + \frac{\xi_4 - \xi_3}{\xi_1} + \frac{x - \xi_4}{\xi_2} & x \in [\xi_4, 1) \end{cases}$$

An analog to "N1" in three layers would be to use the two auxiliary variables

$$\zeta_0 = \frac{\xi_3}{\xi_0} + \frac{\xi_4 - \xi_3}{\xi_1} + \frac{1 - \xi_4}{\xi_2} \qquad \zeta_1 = \begin{cases} \frac{x}{\xi_0} & x \in [0, \xi_3) \\ \frac{\xi_3}{\xi_0} + \frac{x - \xi_3}{\xi_1} & x \in [\xi_3, \xi_4) \\ \frac{\xi_3}{\xi_0} + \frac{\xi_4 - \xi_3}{\xi_1} + \frac{x - \xi_4}{\xi_2} & x \in [\xi_4, 1) \end{cases}$$

Equivalently, we define N2, N3 and N4 as the following transformations:

$$\zeta_0 = \xi_0 \qquad \zeta_1 = \xi_1 \qquad \zeta_2 = \xi_2 \qquad \zeta_3 = \xi_3 \qquad \zeta_4 = \xi_4 \qquad \zeta_5 = \int_0^x a(\tau, \boldsymbol{\xi})\,\mathrm{d}\tau \tag{5.16}$$

$$\zeta_0 = \xi_0 \qquad \zeta_1 = \xi_1 \qquad \zeta_2 = \xi_2 \qquad \zeta_3 = \int_0^x a(\tau, \boldsymbol{\xi})\,\mathrm{d}\tau \tag{5.17}$$

$$\zeta_0 = \xi_3 \qquad \zeta_1 = \xi_4 \qquad \zeta_2 = \int_0^x a(\tau, \boldsymbol{\xi})\,\mathrm{d}\tau \tag{5.18}$$

The rest of the problem is the same as before.

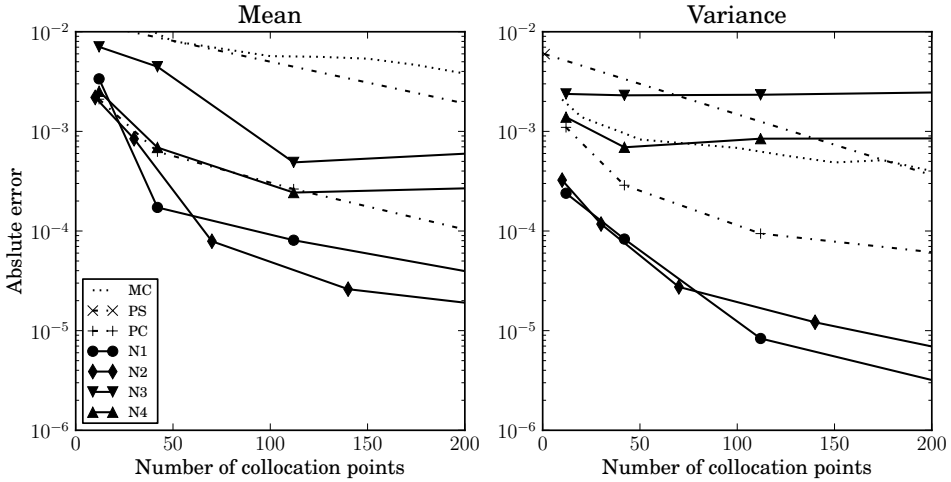

FIG. 5.4. *The absolute error plotted against the number of forward model samples in a diffusion problem with three layers.*

In Figure 5.4 the error is plotted for the three-layered problem. The results are very similar to the two-layered problem. Here neither N3 nor N4 converge in variance. The methods all converge in mean, but with error in the same order as PC. The PS method converges, but more slowly than N1 and N2. As before, the results from N1 and N2 are on the same order of magnitude, with the consistently lowest error estimate among the tested methods. Both examples show that the convergence rate in a non-smooth problem can be recovered to an exponential rate. The reason for this difference in convergence rate between N3 and N4, and N1 and N2 is likely due to under-parameterization of the form two.

6. Conclusions. This paper has presented a new framework for constructing polynomial chaos expansions for stochastically dependent variables. Besides being able to handle problems where input variables are dependent, the framework may have even more important applications when variable transformations are used to restore the fast exponential convergence of polynomial chaos expansions. The convergence rate can significantly detoriate when the mapping from uncertain input parameters to a response contains discontinuities. One difficult case, pointed out in this paper, is when the location of the discontinuities varies with the uncertain parameters. The purpose of the variable transformation is to obtain a smooth mapping in the new probability space. Very often, variable transformations involve auxiliary dependent

stochastic variables. Finding the right variable transformation can be a challenge in a new problem, but for the diffusion problem in multi-layered media considered herein, the improvement in convergence was not significantly sensitive to the specific variant of the transformation. However, it is necessary to utilize mathematical and physical insight in the problem at hand to identify the non-smoothness and construct effective variable transformations. Fortunately, the proposed framework allows to experiment with multiple transformations without any further evaluations of the forward model.

The new, simple methodology for constructing orthogonal polynomials has accuracy and stability on par with the most accurate method available: the three-term recursion formula (TTR). The error in our method flattens out at a certain threshold, while the error in TTR continues to grow. This event happens at such a high polynomial order M that the problem is not of much concern in most applications. When it comes to the error's sensitivity to the number of dimensions D, TTR is a little more resistant, but insignificant compared to the error introduced from increasing the polynomial order.

The new methodology has been tested on a one-dimensional diffusion problem with multiple domains with uncertainties in material properties and domain boundaries. Using the structure of the analytical solution in the variable transformation results in much faster convergence. Other types of transformations where the discontinuity is integrated work almost equally well, provided we keep some of the original parameters as components in the transformed variables. The main conclusion is that the type of variable transformation is not critical as long as the mapping from $\boldsymbol{\zeta}$ to u^* in the new probability space is smooth.

The framework for constructing orthogonal expansions in dependent variables works also for 2D and 3D problems in physical space. However, effective variable transformations must be further investigated through selected cases. In this paper, we have worked with error measures based on the true error. Finding appropriate error measures is much more difficult in 2D and 3D problems with discontinuities.

All the methodologies described in this paper are available in the open source package Chaospy [7, 8], which was used for all computations.

7. Acknowledgement. This work was supported by funding from Statoil ASA through the Simula School of Research and Innovation and by a Center of Excellence grant from the Research Council of Norway through the Center for Biomedical Computing.

The idea of using the Cholesky decomposition for constructing polynomials was suggested by Professor Arne Bang Huseby at the Department of Mathematics, University of Oslo. For the modified Cholesky decomposition used in this paper, we thank Dr. M. M. Forbes for giving us access to his Python version of Dr. M. L. Overton's Matlab implementation of the algorithm by Gill and King [15]. Also, thanks goes out to Margrethe Gustavsen, Stuart Clark, Karoline Hagane, and Rory Williams for their contributions.

REFERENCES

[1] P. Achard and E. De Schutter, *Complex parameter landscape for a complex neuron model*, PLoS Computational Biology, 2 (2006).

[2] R. Askey and J. A. Wilson, *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*, Amer Mathematical Society, 1985.

[3] G. E. P. Box and D. R. Cox, *An analysis of transformations*, Journal of the Royal Statistical Society, (1964), pp. 211–252.

[4] C. W. Clenshaw and A. R. Curtis, *A method for numerical integration on an automatic computer*, Numerische Mathematik, 2 (1960), pp. 197–205.
[5] D. A. Cox, J. Little, and D. O'Shea, *Ideals, varieties, and algorithms: An introduction to computational algebraic geometry and commutative algebra*, Springer, 3rd ed., Feb. 2007.
[6] B. Efron, *Bootstrap methods: another look at the jackknife*, Annals of Statistics, (1979), pp. 1–26.
[7] J. Feinberg and H. P. Langtangen, *Chaospy Software Package for Uncertainty Quantification*, 2014. *https://github.com/hplgit/chaospy*.
[8] ———, *Chaospy: An Open Source Tool for Designing Methods of Uncertainty Quantification*, (2015). URL: *https://github.com/hplgit/chaospy/blob/master/doc/chaospy_software.pdf*.
[9] R. A. Fisher, *On the mathematical foundations of theoretical statistics*, Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, 222 (1922), pp. 309–368.
[10] G. Fishman, *Monte Carlo: concepts, algorithms, and applications*, Springer, 1996.
[11] G. E. Forsythe, *Generation and use of orthogonal polynomials for data-fitting with a digital computer*, Journal of the Society for Industrial and Applied Mathematics, 5 (1957), p. 74.
[12] W. Gautschi, *Construction of Gauss-Christoffel quadrature formulas*, Mathematics of Computation, 22 (1968), p. 251.
[13] T. Gerstner and M. Griebel, *Numerical integration using sparse grids*, Numerical Algorithms, 18 (1998), pp. 209–232.
[14] R. Ghanem and P. D. Spanos, *Stochastic finite elements: a spectral approach*, Courier Dover Publications, Aug. 2003.
[15] J. Gill and G. King, *What to Do When Your Hessian is Not Invertible – Alternatives to Model Respecification in Nonlinear Estimation*, Sociological Methods & Research, 33 (2004), pp. 54–87.
[16] G. H. Golub and J. H. Welsch, *Calculation of Gauss quadrature rules*, vol. 23, Mathematics of Computation, 1967.
[17] J. M. Hammersley, *Monte Carlo methods for solving multivariable problems*, Annals of the New York Academy of Sciences, 86 (1960), pp. 844–874.
[18] E. Hewitt and R. E. Hewitt, *The Gibbs-Wilbraham phenomenon: an episode in Fourier analysis*, Archive for History of Exact Sciences, 21 (1979), pp. 129–160.
[19] A. E. Hoerl, *Application of ridge analysis to regression problems*, Chemical Engineering Progress, 58 (1962), pp. 54–59.
[20] S. Hosder, R. W. Walters, and M. Balch, *Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables*, in Proceedings of the 48th Structures, Structural Dynamics, and Materials Conference, vol. 125, Honolulu, HI, 2007.
[21] S. S. Isukapalli, *Uncertainty analysis of transport-transformation models*, PhD thesis, Rutgers, The State University of New Jersey, 1999.
[22] M. A. Lukas, *Robust generalized cross-validation for choosing the regularization parameter*, Inverse Problems, 22 (2006), p. 1883.
[23] A. Nataf, *Dtermination des distributions de probabilites dont les marges sont donnes*, Comptes rendus de lacademie des sciences, 225 (1962), pp. 42–43.
[24] M. Rosenblatt, *Remarks on a Multivariate Transformation*, Annals of Mathematical Statistics, 23 (1952), pp. 470–472.
[25] S. A. Smolyak, *Quadrature and interpolation formulas for tensor products of certain classes of functions*, in Doklady Akademii Nauk SSSR, vol. 4, 1963, p. 123.
[26] N. Wiener, *The homogeneous chaos*, American Journal of Mathematics, 60 (1938), pp. 897–936.
[27] R. C. Williamson and T. Downs, *Probabilistic arithmetic. I. Numerical methods for calculating convolutions and dependency bounds*, International Journal of Approximate Reasoning, 4 (1990), pp. 89–158.
[28] D. Xiu, *Fast numerical methods for stochastic computations: a review*, Communications in computational physics, 5 (2009), pp. 242–272.
[29] ———, *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press, July 2010.
[30] D. Xiu and G. E. Karniadakis, *Modeling uncertainty in flow simulations via generalized polynomial chaos*, Journal of Computational Physics, 187 (2003), pp. 137–167.
[31] ———, *The Wiener-Askey polynomial chaos for stochastic differential equations*, SIAM Journal on Scientific Computing, 24 (2003), pp. 619–644.