Hybrid Uncertainty Quantification Methods

Akshay Mittal & Gianluca Iaccarino
Stanford University

Final Report

Executive Summary

Multiphysics processes modeled by a system of unsteady differential equations are naturally suited for partitioned (modular) solution strategies. We consider such a model where probabilistic uncertainties are present in each module of the system and represented as a set of random input parameters. A straightforward approach in quantifying uncertainties in the predicted solution would be to sample all the input parameters into a single set, and treat the full system as a black-box. Although this method is easily parallelizable and requires minimal modifications to deterministic solver, it is blind to the modular structure of the underlying multiphysical model. On the other hand, using spectral representations polynomial chaos expansions (PCE) can provide richer structural information regarding the dynamics of these uncertainties as they propagate from the inputs to the predicted output, but can be prohibitively expensive to implement in the high-dimensional global space of uncertain parameters. Therefore, we investigated hybrid methodologies wherein each module has the flexibility of using sampling or PCE based methods of capturing local uncertainties while maintaining accuracy in the global uncertainty analysis. For the latter case, we use a conditional PCE model which mitigates the curse of dimension associated with intrusive Galerkin or semi-intrusive Pseudospectral methods. After formalizing the theoretical framework, we demonstrate our proposed method using a numerical viscous flow simulation and benchmark the performance against a solely Monte-Carlo method and solely spectral method.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344, and was funded by the Department of Energy Office of Science Advanced Scientific Computing Research.

A flexible uncertainty propagation framework for stochastic multiphysics systems using modularly embedded projections

A. Mittal, X. Chen, G. Iaccarino, C. Tong

Abstract

Previous works by the corresponding author have proposed and successfully demonstrated a modular uncertainty propagation framework successfully in subsurface flow simulations for a stochastic reactive-transport model in homogenous [1] and heterogenous media [2]. In this article, we consider a general multiphysics system governed by a coupled system of field equations where each component of the system is prescribed with uncertainties represented as random input parameters. We also consider a partitioned numerical solution strategy which facilitates the reuse of legacy solvers for each uniphysics component of the system and a modularly flexible environment for simulation devlopment. However, due to the bidirectionally coupled nature of the system, each component must account for uncertainties that arise in other components in its respective solution field. Achieving this task along with the flexibility of reusing a component in other multiphysics problems can be a challenging task in practice. We therefore introduce a framework of reduced stochastic modeling using a conditional polynomial chaos representations of uncertainties in each component of the solver. Our framework facilitates the independent development, reuse and replacement of each component without affecting other components to achieve a modularly flexible stochastic simulation environment. We formalize the algorithmic framework and demonstrate our methodology for 1) a 1D fluid-thermal interaction problem and 2) a 1D fluid-thermal-particle interaction problem.

Keywords: Uncertainty Quantification, Stochastic Multiphysics, Hybrid UQ, Stochastic Galerkin method, Polynomial Chaos.

1. Motivations

Numerous fields of science and engineering are actively investigating coupled problems interacting across various physical domains, fields, scales or a combination thereof. While addressing all types of multiphysics problems is unreasonable, we will limit our concerns to coupled field equations which requires the simulateous solution of 2 or more nonlinear component models. An important concern that arises when we encounter such problems is whether to use monolithic or partitioned solution approaches. While this debate is far from

being settled due to each method's advatnages over the other, we will concern ourselves with partitioned solution strategies, which can enable the use of legacy solvers for each partitioned component of a coupled system. This is often considered as a practical advantage for computational engineers as flexibility and interdisciplinary expertise can trump the additional overheads entailed in setting up a partitioned solution method.

The rapid growth of uncertainty quantification (UQ) in the last decade has been primarily motivated by the demand for predictive simulations with accurate quantification of the credibility and confidence of the obtained numerical results. UQ has so far been mostly concerned with the analysis of simple uniphysics models and the extension of methodologies to multiphysics models is far from being straightforward. The simplest possible method of propagating uncertainties in the inputs of a computer simulation would be the use of statistical or random sampling such as Monte-Carlo (MC) methods [3, 4]. While this approach and improvements such as quasi-Monte-Carlo [5], Latin hypercube [6], Sobolev sequences, etc are easily adaptable to a deterministic solver code with attractive features such as parallelization and dimensional independence, it is essentially blind to the composition of the deterministic solver which may contain rich structures as with multiphysics models. Also, the typical convergence rate i.e. $O\left(N^{-1/2}\right)$ where N is the number of repeated simuation runs, is too slow for any practical implementation when each simulation run has a significant computational cost of its own. Nevertheless, MC methods are still the most robust for stochastic simulations and are often used in validation as well as for estimating statistics on a cheaper surrogate model of the output. On the other hand, non-statistical methods using stochastic expansions exhibit better convergence properties in the case when the simulation outputs are sufficiently smooth in the stochastic input space. Specifically, we are talking about the use polynomial chaos expansions [7, 8] for the representation of uncertainties in the inputs, model and outputs. The associated stochastic Galerkin projection method [9] reduces a stochastic model into a deterministic one with a single simulation run required to propagate uncertainties from the inputs to the outputs. The disadvantage however, is that the method is intrusive i.e. existing determinstic codes need to be rewritten and the cost of solving the stochastic Galerkin system (SGS) can be much higher than its determinstic counterpart. To deal with the former disadvantage, a non-intrusive spectral projection (NISP) method based on collocation in the stochastic domain has been proposed in various publications [10, 11] and its convergence rate with respect to the order of the polynomial has been demonstrated to be comparable with intrusive methods [12] (once again, in cases where the outputs are sufficiently smooth). Both methods fall under a class of methods known as spectral methods [13]. Several local variants of these methods have been proposed [14, 15] to deal with long term integration errors and to provide another aspect of adaptivity with the spectral representation besides the polynomial order. Another disadvantage that applies to intrusive and non-intrusive spectral methods alike is the curse of dimensionality. The number of degrees of freedom in the expansion and sample points for collocation grows exponentially with respect to the number of input

parameters. With coupled problems in mind, we can expect that the curse of dimension in implementing a spectral methods would be exacerbated with the multi-component structure. Even if each component contributes a moderate number of random variables, the coupled nature of the model can lead to a large number of uncertainties in the global sense.

Previous works by the corresponding author [1. 2] made use of a generelized polynomial chaos (gPC) approach to simulating unsteady reactive-transport phenomenon with stochastic velocity, dispersivity and first-order reaction rates which was partitioned into the transport and reaction components using a first order additive operator splitting method. The linear nature of the coupled transport and reactive operators allowed for an efficient polynomial order specific decomposition of the of the SGS for each component in the combined space of uncertainties. We wish to abandon this oversimplified assumption on the model and consider a general nonlinear coupled system of solution fields in this article. We will however assume that the uncertainties contributed by a given component is independent from the uncertainties contributed by other components. With this assumption, we can represent uncertainties in the conditional or reduced stochastic space of local uncertainties within each model while retaining the possibility of expressing the output in the global space of uncertainties for post-processing. Our proposed methodology can create significant degree of flexibility by facilitating the independent development, management and reuse of uniphysics components. Also, with this approach, the natural division of modeling expertise which is often a practical constraint in deterministic multiphysics simulation development can therefore be extended to stochastic multiphysics simulation development.

The remaining sections of this article are organized as follows. In section 2, we introduce our governing multiphysics model and briefly discuss partioned solution strategies used in deterministic simulations that are based on fixed point iterations. In section 3, we review gPC, the construction of the SGS for a partitioned component of the simulation and efficient solution methods using Krylov subspace methods. In section 4, we describe our proposed modularly embedded uncertainty propagation method using reduced gPC representations of uncertainties within components. In section 5, we demonstrate our method on a fluid-thermal interaction problem.

Convention	Symbol	Connotation
Lower case.	x	Deterministic scalar quantity
Lower case bold.	$\boldsymbol{x}$	Stochastic scalar quantity
Upper case.	X	Deterministic vector quantity
Upper case bold.	$\boldsymbol{X}$	Stochastic vector quantity
Bracket upper case.	$[X]$	Deterministic matrix quantity
Bracket upper case bold.	$[\boldsymbol{X}]$	Random matrix quantity

Table 1. List of symbols and their connotations in sections 2-5.

1.1. Remark on notations

Table 1 defines the notation used in the remainder of the article. Greek and calligraphic symbols (such as α and $\mathcal{A}$ resepectively) do not have a standard connotation in this article and will be defined as and when necessary.

2. Partitioned solution of coupled field systems.

2.1. Model problem

This article will consider the solution of the following stochastic coupled system.

$$\begin{array}{llll} \boldsymbol{F}\left(\boldsymbol{U};\boldsymbol{V},\boldsymbol{X}\right)=\boldsymbol{0}, & \boldsymbol{F},\boldsymbol{U}\in\mathbb{R}^{n_1}, & \boldsymbol{X}\in\mathcal{X}\subseteq\mathbb{R}^{m_1}, \\ \boldsymbol{G}\left(\boldsymbol{V};\boldsymbol{U},\boldsymbol{Y}\right)=\boldsymbol{0}, & \boldsymbol{G},\boldsymbol{V}\in\mathbb{R}^{n_2}, & \boldsymbol{Y}\in\mathcal{Y}\subseteq\mathbb{R}^{m_2}. \end{array} \tag{1}$$

The given coupled field system is assumed to be a either a spatial discretization of a steady-state equilibrium problem or at a single time step of an unsteady evolution problem. $\boldsymbol{U}$ and $\boldsymbol{V}$ could denote the same quantity in different spatial domains (multi-domain coupling) or in the same domain (multi-process coupling). Further, we assume that the system is well posed for all values of $\boldsymbol{X}$ and $\boldsymbol{Y}$ in (Ω,Σ,ρ) where Ω is the sample space, $\Sigma\subset 2^{\Omega}$ is a $\sigma-$algebra (non-empty collection of subsets in Ω that is closed under complementation and countable union of its members). and ρ is a probability measure that maps from Σ to $[0,1]$. Therefore, we need to find $\boldsymbol{U}$ and $\boldsymbol{V}$ in the probability triple (Ω,Σ,ρ) under the assumption of stbility and uniqueness. Discussions on obtaining the given form (1) from a stochastic partial differential equation (PDE) system are deferred until section 5. If we assume that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are fixed quantities, a monolithic solution method using Newton's method would proceed as follows. With an appropriate choice for $\boldsymbol{U}^{(0)}$ and $\boldsymbol{V}^{(0)}$, we iterate for $\ell=0,1,\ldots$

$$\begin{bmatrix} \nabla_{\boldsymbol{U}}\boldsymbol{F} & \nabla_{\boldsymbol{V}}\boldsymbol{F} \\ \nabla_{\boldsymbol{U}}\boldsymbol{G} & \nabla_{\boldsymbol{V}}\boldsymbol{G} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^{\ell+1}-\boldsymbol{U}^{\ell} \\ \boldsymbol{V}^{\ell+1}-\boldsymbol{V}^{\ell} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{F}\left(\boldsymbol{U}^{\ell};\boldsymbol{V}^{\ell},\boldsymbol{X}\right) \\ \boldsymbol{G}\left(\boldsymbol{V}^{\ell};\boldsymbol{U}^{\ell},\boldsymbol{Y}\right) \end{bmatrix}. \tag{2}$$

The monolithic method encapsulates both components, solution fields and uncertainties into a single vector. i.e., if $\boldsymbol{H}=(\boldsymbol{F},\boldsymbol{G})\in\mathbb{R}^{n_1+n_2}$, $\boldsymbol{W}=(\boldsymbol{U},\boldsymbol{V})\in\mathbb{R}^{n_1+n_2}$ and $\boldsymbol{Z}=(\boldsymbol{X},\boldsymbol{Y})\in\mathcal{Z}=\mathcal{X}\times\mathcal{Y}\subseteq\mathbb{R}^{m_1+m_2}$, we can rewrite (2) as follows.

$$\left[\nabla_{\boldsymbol{W}}\boldsymbol{H}\left(\boldsymbol{W}^{\ell-1};\boldsymbol{Z}\right)\right]\left(\boldsymbol{W}^{\ell}-\boldsymbol{W}^{\ell-1}\right)=-\boldsymbol{H}\left(\boldsymbol{W}^{\ell-1};\boldsymbol{Z}\right). \tag{3}$$

Developing solution strategies based on (2) can be difficult in practice if legacy solvers for the solution of each uniphysics components is already available. Another approach would be to eliminate one of the fields ($\boldsymbol{V}$ in this case) and represent $\boldsymbol{V}$ as an implicit function of $\boldsymbol{U}$ using the second component $\boldsymbol{G}\left(\boldsymbol{V};\boldsymbol{U},\boldsymbol{Y}\right)=0$. This would result in the first problem being dependent only on $\boldsymbol{U}$ as $\boldsymbol{F}\left(\boldsymbol{U},\boldsymbol{V}\left(\boldsymbol{U}\right);\boldsymbol{X}\right)=0$. This method is often limited in efficiency to

only linear problems using a Schur decomposition or in problems where some of the coupled variables are modeled as a handful of scalars. The latter case of weak or network coupling has been investigated recently [16]. We now introduce a partitioned solution method similar to (2).

2.2. Partitioned iterative solution

The simplest monolithic solution methods that can be modified into partitioned solution methods are relaxation approaches that employ nonlinear Gauss-Siedel iterations [17] as follows. With an appropriate choice for $\boldsymbol{U}^{(0)}$ and $\boldsymbol{V}^{(0)}$, we iterate for $\ell = 0, 1, \ldots$

$$\begin{aligned} \left(\boldsymbol{U}^{\ell+1} - \boldsymbol{U}^{\ell}\right) &= -\varsigma \boldsymbol{F}\left(\boldsymbol{U}^{\ell}; \boldsymbol{V}^{\ell}, \boldsymbol{X}\right), \\ \left(\boldsymbol{V}^{\ell+1} - \boldsymbol{V}^{\ell}\right) &= -\varsigma \boldsymbol{G}\left(\boldsymbol{V}^{\ell}; \boldsymbol{U}^{\ell+1}, \boldsymbol{Y}\right), \end{aligned} \tag{4}$$

where ς is the relaxation parameter which typically lies in the interval $(-1, 1)$. The slow linear convergence or a lack of convergence guaranteed by this method is disadvantageous in practice. Instead, we use a partitioned Newton method which solves (1) as follows. With an appropriate choice for $\boldsymbol{U}^{(0)}$ and $\boldsymbol{V}^{(0)}$, we iterate for $\ell = 0, 1, \ldots$

$$\begin{aligned} \left[\nabla_{\boldsymbol{U}} \boldsymbol{F}\left(\boldsymbol{U}^{\ell}; \boldsymbol{V}^{\ell}, \boldsymbol{X}\right)\right]\left(\boldsymbol{U}^{\ell+1} - \boldsymbol{U}^{\ell}\right) &= -\boldsymbol{F}\left(\boldsymbol{U}^{\ell}; \boldsymbol{V}^{\ell}, \boldsymbol{X}\right), \\ \left[\nabla_{\boldsymbol{V}} \boldsymbol{G}\left(\boldsymbol{V}^{\ell}; \boldsymbol{U}^{\ell+1}, \boldsymbol{Y}\right)\right]\left(\boldsymbol{V}^{\ell+1} - \boldsymbol{V}^{\ell}\right) &= -\boldsymbol{G}\left(\boldsymbol{V}^{\ell}; \boldsymbol{U}^{\ell+1}, \boldsymbol{Y}\right). \end{aligned} \tag{5}$$

Given that the solution strategy is partitioned for given deterministic coupled system with values of $\boldsymbol{X}$ and $\boldsymbol{Y}$, a straightforward MC method could be implemented in an outer loop in which we solve (5) for converged values of $\boldsymbol{U}$ and $\boldsymbol{V}$. This method would require minimum modification to each component's legacy solver and the computational model as a whole. However, as mentioned earlier, the slow convergence rate of the estimated statistics of the solution would require a large number of repeated simulations. Instead we wish to construct a cheaper surrogate model of the solutions with which exhaustive MC sampling can be implemented at a fraction of the total cost.

While many possibilities exist for the choice of the surrogate, we will limit our discussions in this article to the use of global polynomials with the assumption of sufficient smoothness of the stochastic solution fields in $\mathcal{Z}$. In the context of UQ, these global polynomial models are known as polynomial chaos (gPC) expansions. The next section is a review of constructing polynomial chaos representations of uncertainties and the projection of the coupled stochastic system (1) into the space of multivariate polynomials.

3. Uncertainty propagation using gPC

3.1. Preliminary definitions and review of gPC

The use of gPC methods has become ubiquitous for stochastic modeling in the last decade and has been successfully demonstrated in several domains

of computational science and engineering such as fluid mechanics [18], solid mechanics [19], etc. Originally, gPC was introduced as Weiner chaos [20] by Ghanem and Spanos; and subsequently extended to the Weiner-Askey family of orthogonal polynomials [8]. Given a probability triple (Ω, Σ, ρ) and a second order random variable $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{\omega}) : \Omega \to L_2(\Omega, \rho)$ the depends on a single random variable $\boldsymbol{\omega}$, we can represent $\boldsymbol{x}$ as the following polynomial expansion.

$$\boldsymbol{u}(\boldsymbol{\omega}) = \sum_{i \geq 0} u_i \hat{\boldsymbol{\psi}}_i(\boldsymbol{\omega}), \tag{6}$$

where the polynomials $\left\{\hat{\boldsymbol{\psi}}_i : i \geq 0\right\}$ are the $\rho-$orthogonormal polynomials with respective polynomial degree (order) i; i.e. given $i, j \geq 0$,

$$\int_{\mathcal{X}} \hat{\boldsymbol{\psi}}_i \hat{\boldsymbol{\psi}}_j d\rho = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}.$$

We remark here that the first basis polynomial $\hat{\boldsymbol{\psi}}_0$ would always equal 1 regardless of ρ, and polynomials of order > 1 represent $\rho-$uncorrelated random variables with zero mean and unit variance.

The above expansion $\boldsymbol{x}$ can be truncated upto a given polynomial order p. We therefore, define an approximation $\boldsymbol{x}^p$ as follows.

$$u^p = \sum_{i=0}^{p} u_i \hat{\boldsymbol{\psi}}_i(\boldsymbol{\omega}). \tag{7}$$

The Cameron-Martin theorem [21] states that an approximation $\boldsymbol{u}^p$ converges to $\boldsymbol{u}$ as $p \to \infty$ for any $\boldsymbol{u} \in L^2(\Omega, \rho)$. In practice, the coordinates $\{u_i : i \geq 0\}$ converge at an exponential rate if $\boldsymbol{u}$ is sufficiently smooth in Ω. For a random vector, we can extend this result for each component of the vector trivially. In the general case a random vector $\boldsymbol{U}$ depends on multiple independent random variables $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_m \end{pmatrix}$, we can construct the orthonormal polynomial bases using a tensor product of univariate bases. Therefore, for a multiindex $i \in \mathbb{N}^m$, we have

$$\boldsymbol{\psi}_\alpha(\boldsymbol{X}) = \prod_{i=1}^{m} \hat{\boldsymbol{\psi}}_{\alpha_i}^{(i)}(\boldsymbol{x}_i), \tag{8}$$

where $\left\{\hat{\boldsymbol{\psi}}_j^{(i)} : 1 \leq i \leq m, j \geq 0\right\}$ corresponds to $\rho^{(i)}-$orthogonormal polynomials with respective polynomial degree (order) j. It is assumed here that the joint probability density is a product of individual probability densities along each direction i.e. $d\rho = d\rho^{(1)} \times \cdots \times d\rho^{(m)}$. Therefore, $\boldsymbol{U}$ can be expanded as follows.

$$\boldsymbol{U}(\boldsymbol{X}) = \sum_{|\alpha| \geq 0} U_\alpha \boldsymbol{\psi}_\alpha(\boldsymbol{X}). \tag{9}$$

The truncation rule can in general be different along each direction but the popular choice of truncation uses a total order rule to define the set of possible

multi-indices $\mathcal{I}_m^p = \{\alpha \in \mathbb{N}^m : 0 \leq \alpha_1 + \cdots + \alpha_m \leq p\}$. Therefore, we have the approximation $\boldsymbol{U}^p$ as follows.

$$\boldsymbol{U}^p(\boldsymbol{X}) = \sum_{\alpha \in \mathcal{I}_m^p} U_\alpha \boldsymbol{\psi}_\alpha(\boldsymbol{X}) = \sum_{|\alpha|=0}^{p} U_\alpha \boldsymbol{\psi}_\alpha(\boldsymbol{X}), \tag{10}$$

The coordinates of the expansion can be obtained using the orthonormality of the basis functions as follows.

$$U_\alpha = \int_{\mathcal{X}} \boldsymbol{U}^p \boldsymbol{\psi}_\alpha d\rho. \tag{11}$$

We assume that the arrangement of multi-indices follows a total degree lexicographical order i.e.

$$\mathcal{I}_m^p = \left\{ \alpha^{(0)}, \ldots, \alpha^{(\nu_m^p)} \in \mathbb{N}^m : \left|\alpha^{(0)}\right| \leq \cdots \leq \left|\alpha^{(\nu_m^p)}\right|, \nu_m^p + 1 = \binom{m+p}{m} \right\}. \tag{12}$$

With this assumption, we would once again obtain $\boldsymbol{\psi}_\alpha = 1 : |\alpha| = 0$ and $\{\boldsymbol{\psi}_\alpha : |\alpha| \geq 1\}$ as a set of $\rho-$uncorrelated random variables with zero mean and unit variance. Also, the first and second order statistics of the solution can be readily approximated using the coordinates $(U_\alpha : 0 \leq |\alpha| \leq p)$ as follows.

$$\begin{array}{c} \mu_1(\boldsymbol{U}) \approx \int_\Omega \boldsymbol{U}^p d\rho = U_0, \\ \mu_2(\boldsymbol{U}) \approx \int_\Omega (\boldsymbol{U}^p - U_0)(\boldsymbol{U}^p - U_0)^T d\rho = \sum_{|i|=1}^{p} U_i U_i^T. \end{array} \tag{13}$$

The simplest method of obtaining the coordinates of the expansion (10) is by using a simple least squares minimization procedure on an ensemble of samples $\left\{ \boldsymbol{U}\left(\boldsymbol{X}^{(i)}\right) : 0 \leq i \leq q, q > \nu_m^p \right\}$. Such a method would be considered non-intrusive and the inputs samples could be obtained using MC. Another popular non-intrusive method is the pseudospectral approximation method [22]. where the input nodes are pre-selected instead of being randomly sampled, and the coordinates are obtained using a Gaussian-quadrature rule for evaluating the right hand side of (11). We will instead discuss an intrusive or embedded projection method for obtaining the PC coordinates when $\boldsymbol{U}$ is obtained using the solution of a nonlinear field equation.

3.2. Embedded Galerkin projection method

Now, we consider a scenario where $\boldsymbol{U}$ is implicitly defined using a function $\boldsymbol{F}(\boldsymbol{U}; \boldsymbol{X}) = 0$ and solved using a Newton method as follows.

$$\left[\nabla_{\boldsymbol{U}} \boldsymbol{F}\left(\boldsymbol{U}^\ell; \boldsymbol{X}\right)\right]\left(\boldsymbol{U}^{\ell+1} - \boldsymbol{U}^\ell\right) = -\boldsymbol{F}\left(\boldsymbol{U}^\ell; \boldsymbol{X}\right). \tag{14}$$

Defining $\boldsymbol{F}^{\ell,p}(\boldsymbol{X}) = \boldsymbol{F}\left(\boldsymbol{U}^{\ell,p}; \boldsymbol{X}\right)$ and $\left[\boldsymbol{J}^{\ell,p}(\boldsymbol{X})\right] = \left[\nabla_{\boldsymbol{U}} \boldsymbol{F}\left(\boldsymbol{U}^\ell; \boldsymbol{X}\right)\right]$, we have a truncated polynomial expansion of $\boldsymbol{F}^\ell$ and $\left[\boldsymbol{J}^\ell\right]$ given as follows.

$$\begin{array}{c} \boldsymbol{F}^{\ell,p}(\boldsymbol{X}) = \boldsymbol{F}^{\ell,p}\left(\boldsymbol{U}^{\ell,p}; \boldsymbol{X}\right) = \sum_{|\alpha|=0}^{p} F_\alpha^\ell \boldsymbol{\psi}_\alpha(\boldsymbol{X}), \\ \left[\boldsymbol{J}^{\ell,p}(\boldsymbol{X})\right] = \nabla_{\boldsymbol{U}} \boldsymbol{F}^{\ell,p}\left(\boldsymbol{U}^{\ell,p}; \boldsymbol{X}\right) = \sum_{|\alpha|=0}^{p} \left[J_\alpha^\ell\right] \boldsymbol{\psi}_\alpha(\boldsymbol{X}). \end{array} \tag{15}$$

We then formulate an approximate Newton step i.e.

$$\left[\boldsymbol{J}^{\ell,p}\left(\boldsymbol{X}\right)\right]\left(\boldsymbol{U}^{\ell+1,p}\left(\boldsymbol{X}\right)-\boldsymbol{U}^{\ell,p}\left(\boldsymbol{X}\right)\right)=-\boldsymbol{F}^{\ell,p}\left(\boldsymbol{X}\right), \tag{16}$$

which needs to be satisfied at every $\boldsymbol{X}\in\mathcal{X}$. Expanding (16), we get

$$\sum_{|\alpha|=0}^{p}\sum_{|\beta|=0}^{p}\left[J_{\alpha}^{\ell}\right]\left(U_{\beta}^{\ell+1}-U_{\beta}^{\ell}\right)\boldsymbol{\psi}_{\alpha}\left(\boldsymbol{X}\right)\boldsymbol{\psi}_{\beta}\left(\boldsymbol{X}\right)=-\sum_{|\alpha|=0}^{p}F_{\alpha}^{\ell}\boldsymbol{\psi}_{\alpha}\left(\boldsymbol{X}\right). \tag{17}$$

Finally, (17) is projected in the space spanned by the basis polynomials i.e. $\{\boldsymbol{\psi}_{\alpha}:0\leq|\alpha|\leq p\}$, and we obtain the embedded Galerkin projected form of (14) as follows.

$$\sum_{|\alpha|=0}^{p}\sum_{|\beta|=0}^{p}\epsilon_{\alpha\beta\gamma}\left[J_{\alpha}^{\ell}\right]\left(U_{\beta}^{\ell+1}-U_{\beta}^{\ell}\right)=-F_{\gamma}^{\ell},\ \forall\gamma:0\leq|\gamma|\leq p, \tag{18}$$

where $\epsilon_{\alpha\beta\gamma}=\int_{\mathcal{X}}\boldsymbol{\psi}_{\alpha}\boldsymbol{\psi}_{\beta}\boldsymbol{\psi}_{\gamma}d\rho$. Therefore, instead of solving a stochastic problem by repeated sampling, we obtain the coordinates of $\boldsymbol{U}^{\ell+1,p}$ by solving a single deterministic equation (18). This method is called intrusive or embedded for two reasons. Firstly, we require a significant overhaul of the deterministic solver which solves (14). Secondly, the stochastic information is embedded within the Galerkin system (18). We remark here that the deterministic linear system (18) is fully coupled and is larger than the linear system solved in (14). We now discuss an efficient method of solving (18) using commonly used iterative Krylov subspace solvers.

3.3. Krylov iterative solution method

To simplify the notational complexity in subsequent discussions in this section, we formulate a single-index version of (18) as follows.

$$\sum_{i=0}^{\nu_{m}^{p}}\sum_{j=0}^{\nu_{m}^{p}}\epsilon_{i,j,k}\left[J_{i}^{\ell}\right]\left(U_{j}^{\ell+1}-U_{j}^{\ell}\right)=-F_{k}^{\ell},\ \forall0\leq k\leq\nu_{m}^{p}, \tag{19}$$

which can be compactly written as $\left[\tilde{J}^{\ell}\right]\left(\tilde{U}^{\ell+1}-\tilde{U}^{\ell}\right)=-\tilde{F}^{\ell}$ such that

$$\left[\tilde{J}^{\ell}\right]=\sum_{i=0}^{\nu_{m}^{p}}\left[E_{i}\right]\otimes\left[J_{i}^{\ell}\right],\tilde{U}^{\ell}=\begin{bmatrix}U_{0}^{\ell}\\ \vdots\\ U_{\nu_{m}^{p}}^{\ell}\end{bmatrix},\tilde{F}^{\ell}=\begin{bmatrix}F_{0}^{\ell}\\ \vdots\\ F_{\nu_{m}^{p}}^{\ell}\end{bmatrix}, \tag{20}$$

where $\otimes$ denotes the kronecker product and $\left[E_{i}\right]_{jk}=\epsilon_{ijk}$. Using a generic Krylov subspace method on (19) results in a solution that is a linear combination of $\left\{\left[\tilde{J}^{\ell}\right]^{i-1}\tilde{F}^{\ell}:1\leq i\leq n\left(\nu_{m}^{p}+1\right)\right\}$ and the computations within the solver

would not need to store the matrix $\left[\tilde{J}^{\ell}\right]$ explicitly. Instead we would store the matrix indirectly by defining a routine to compute the linear transformation $M = M_{\tilde{J}^{\ell}} : M(Z) = \left[\tilde{J}^{\ell}\right] Z$ for an arbitrary input vector $Z \in \mathbb{R}^{n(\nu_m^p+1)}$. A method of evaluating M using an approximate pseudospectral factorization of $\left[\tilde{J}\right]$ has been proposed in [23] for linear field equations i.e. $\boldsymbol{F}(\boldsymbol{U};\boldsymbol{X}) = [\boldsymbol{A}(\boldsymbol{X})]\boldsymbol{U} - \boldsymbol{B}(\boldsymbol{X})$ where $[\boldsymbol{J}]$ corresponds to $[\boldsymbol{A}]$. We follow a similar approach that is based on the following property of $[\boldsymbol{J}]$. For a given positive $h \ll 1$, the first order approximation of a matrix-vector product evaluated using $[\boldsymbol{J}]$ and a vector $Z \in \mathbb{R}^n$ is

$$[\boldsymbol{J}(\boldsymbol{X})] Z = \left[\nabla_{\boldsymbol{U}} \boldsymbol{F}\left(\boldsymbol{U}^{\ell};\boldsymbol{X}\right)\right] Z \approx \frac{\boldsymbol{F}(\boldsymbol{U}+hZ;\boldsymbol{X}) - \boldsymbol{F}(\boldsymbol{U};\boldsymbol{X})}{h}, \tag{21}$$

and the second order approximation is

$$[\boldsymbol{J}(\boldsymbol{X})] Z = \left[\nabla_{\boldsymbol{U}} \boldsymbol{F}\left(\boldsymbol{U}^{\ell};\boldsymbol{X}\right)\right] Z \approx \frac{\boldsymbol{F}(\boldsymbol{U}+hZ;\boldsymbol{X}) - \boldsymbol{F}(\boldsymbol{U}-hZ;\boldsymbol{X})}{2h}. \tag{22}$$

The choice of h is usually defined using a fixed positive $\varepsilon \ll 1$ with $h = \varepsilon \|Z\|_{\infty}^{-1}$. In our implementations, we choose $\varepsilon = 10^{-3}$. This method of approximating matrix-vector products in a Krylov iterative solver is often called the Jacobian-Free-Newton Krylov method (JFNK) [24].

3.3.1. Factorization of the nonlinear SGS

We will now describe a factorization method for nonlinear field equations. Given a set of quadrature points and weights $\left\{\left(X^{(i)}, w^{(i)}\right) : 0 \le i \le q\right\}$, we define $[\Psi] \in \mathbb{R}^{(\nu_m^p+1)\times(q+1)} : [\Psi]_{ij} = \psi_i\left(X^{(j)}\right)$. Using these definitions, we propose the following 3−step procedure for evaluating M for a given vector Z.

1. Reshape $Z \in \mathbb{R}^{n(\nu_m^p+1)}$ into a matrix $\left[\tilde{Z}\right] \in \mathbb{R}^{n\times(\nu_m^p+1)}$. Evaluate $\left[\hat{Z}\right] = \left[\tilde{Z}\right][\Psi]$ with columns $\left[\begin{array}{ccc} \hat{Z}_0 & \cdots & \hat{Z}_q \end{array}\right]$.
2. For $i = 0, \ldots, q$, evaluate $\hat{M}_i = w^{(i)} \boldsymbol{J}\left(X^{(i)}\right) \hat{Z}_i$ using (21) or (22). Construct the matrix $\left[\hat{M}\right] = \left[\begin{array}{ccc} \hat{M}_0 & \cdots & \hat{M}_q \end{array}\right]$.
3. Evaluate $\left[\tilde{M}\right] = \left[\hat{M}\right][\Psi]^T$ and reshape $\left[\tilde{M}\right] \in \mathbb{R}^{n\times(\nu_m^p+1)}$ into a vector $M \in \mathbb{R}^{n(\nu_m^p+1)}$.

Steps 1 and 3 can be thought of as pre-processing and post-processing steps respectively and are relatively cheaper to process compared to step 2. Evaluating $\tilde{F}^{\ell}$ can also be done using the same set of quadrature points and weights. Suppose $\Psi_i : 0 \le i \le q$ is a column of Ψ, we use the values of $\boldsymbol{F}^{\ell,p}\left(X^{(i)}\right)$ to obtain $\tilde{F}^{\ell}$ as follows.

$$\tilde{F}^{\ell} = \sum_{i=0}^{q} w^{(i)} \Psi_i \otimes \boldsymbol{F}^{\ell,p}\left(X^{(i)}\right), \tag{23}$$

In practice, the convergence of the Krylov iterative solver can be significantly accelerated with the use of a preconditioner $\left[\tilde{P}^{\ell}\right]$, which results in the modified stochastic system

$$\left[\tilde{P}^{\ell}\right]^{-1}\left[\tilde{J}^{\ell}\right]\left(\tilde{U}^{\ell+1}-\tilde{U}^{\ell}\right)=-\left[\tilde{P}^{\ell}\right]^{-1}\tilde{F}^{\ell}. \tag{24}$$

We will now briefly discuss two types of commonly used preconditioning matrices.

3.3.2. Preconditioning

In [23] , the significant acceleration of convergence using block-diagonal preconditioners $\left[\tilde{P}^{\ell}\right]=[I]\otimes\left[P^{\ell}\right]$ is demonstrated on paramterized linear systems arising out of elliptic boundary values problems. The best choice for $\left[P^{\ell}\right]$ indicated by the experiments is the mean of $\left[\boldsymbol{J}^{\ell,p}\right]$ i.e.

$$\left[P^{\ell}\right]=\int_{\mathcal{X}}\left[\boldsymbol{J}^{\ell,p}\left(\boldsymbol{X}\right)\right]d\rho\approx\sum_{i=0}^{q}\left[\boldsymbol{J}^{\ell,p}\left(\boldsymbol{X}^{(i)}\right)\right]w^{(i)}. \tag{25}$$

This modified equation (24) with block-diagonal preconditioners can be also be factorized; we can implement a similar 3−step process of evaluating matrix-vector products as described earlier with a slight modification in step 2, where we instead solve $[P]\,\hat{M}_i=w^{(i)}\boldsymbol{J}\left(X^{(i)}\right)\hat{Z}_i:0\leq i\leq q$. Although this preconditioner fails to incorporate higher order stochastic information, it is fairly inexpensive to find its Cholesky decomposition for inversion. as a block-diagonal preconditioner for the solution of (19). A popular choice for non block-diagonal preconditioners is the kronecker product matrix [25] which is given as follows.

$$\left[\tilde{P}^{\ell}\right]=\sum_{i=0}^{\nu_m^p}\frac{\mathrm{tr}\left(\left[J_i^{\ell}\right]^T\left[J_0^{\ell}\right]\right)}{\mathrm{tr}\left(\left[J_0^{\ell}\right]^T\left[J_0^{\ell}\right]\right)}\left[E_i\right]\otimes\left[J_0^{\ell}\right]. \tag{26}$$

Although the kronecker product preconditioner incorporates higher order stochastic information, it is much more expensive to invert using standard decomposition methods. The overall computation time however, is reported to be comparable to the use of the mean block-diagonal preconditioner is used [24]. We will restrict our preconditioning for the test problems in section 5 to the mean block-diagonal form.

3.4. Reuse of components and flexibility

In sections 3.2 and 3.3, we described the construction and solution of the embedded Galerkin projection system (SGS) for a single stochastic field equation $\boldsymbol{F}\left(\boldsymbol{U},\boldsymbol{X}\right)=0$. We also described a non-intrusive method of implementing Krylov iterative solvers for the SGS However, for the problem at hand (1), where

2 field equations are coupled, the standard implementation of the SGS propagation method would entail a projection of each component of (5) using the global $(m_1 + m_2)$ −variate polynomial basis functions. This process can prohibit the reuse of the resultant stochastic solvers which could serve as legacy components in a future problem. Also, updating one component's stochastic space would require a significant update to all of the component solvers involved in the simulation. This would preclude the much desired flexibility in the development and reuse of legacy solver components for stochastic simulations. Instead of the standard SGS method, we propose a method of reduced or conditional projections which we call modularly embedded projections. The essential idea is to use a conditional stochastic expansion within each module and recondstruct the global stochastic expansion once the conditional stochastic expansion coordinates have been updated.

4. Modularly embedded projection method

4.1. Conditional gPC approximation

Considering the model problem (1), we define a truncated and separated gPC approximation $\boldsymbol{U}^p \approx \boldsymbol{U}$ (or $\boldsymbol{V}^p \approx \boldsymbol{V}$) that are dependent on independent random vectors $\boldsymbol{X} \in \mathcal{X} \subseteq \mathbb{R}^{m_1}$ and $\boldsymbol{Y} \in \mathcal{Y} \subseteq \mathbb{R}^{m_2}$ as follows.

$$\begin{aligned}\boldsymbol{U}^p(\boldsymbol{X},\boldsymbol{Y}) &= \sum_{\alpha \in \mathcal{I}^p_{m_1}} \sum_{\beta \in \mathcal{I}^{p-|\alpha|}_{m_2}} U_{\alpha\beta}\boldsymbol{\psi}_\alpha(\boldsymbol{X})\boldsymbol{\phi}_\beta(\boldsymbol{Y}) \\ &= \sum_{|\alpha|+|\beta|=0}^{p} U_{\alpha\beta}\boldsymbol{\psi}_\alpha(\boldsymbol{X})\boldsymbol{\phi}_\beta(\boldsymbol{Y}),\end{aligned} \tag{27}$$

where the polynomial basis functions $\{\psi_\alpha : 0 \le |\alpha| \le p\}$ and $\{\phi_\alpha : 0 \le |\alpha| \le p\}$ are defined with respect to $\rho_{\boldsymbol{X}}$ and $\rho_{\boldsymbol{Y}}$ respectively. We can construct the tensor product form of global polynomial basis functions based on the assumption that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are independent i.e. $d\rho(\boldsymbol{X},\boldsymbol{Y}) = d\rho_{\boldsymbol{X}}(\boldsymbol{X}) \times d\rho_{\boldsymbol{Y}}(\boldsymbol{Y})$. The constituent densities $\rho_{\boldsymbol{X}}$ and $\rho_{\boldsymbol{Y}}$ are also assumed to be in their product form so that their corresponding polynomial basis functions also assume a tensor product form. Using the global gPC form (27), we can construct a conditional or reduced [25] gPC decomposition of $\boldsymbol{U}^p$ and $\boldsymbol{V}^p$ as follows.

$$\begin{aligned}\boldsymbol{U}^p(\boldsymbol{X},\boldsymbol{Y}) &= \textstyle\sum_{|\alpha|+|\beta|=0}^{p} U_{\alpha\beta}\boldsymbol{\psi}_\alpha(\boldsymbol{X})\boldsymbol{\phi}_\beta(\boldsymbol{Y}) = \sum_{|\alpha|=0}^{p} \hat{\boldsymbol{U}}_\alpha(\boldsymbol{Y})\boldsymbol{\psi}_\alpha(\boldsymbol{X}), \\ \boldsymbol{V}^p(\boldsymbol{X},\boldsymbol{Y}) &= \textstyle\sum_{|\alpha|+|\beta|=0}^{p} V_{\alpha\beta}\boldsymbol{\psi}_\alpha(\boldsymbol{X})\boldsymbol{\phi}_\beta(\boldsymbol{Y}) = \sum_{|\beta|=0}^{p} \hat{\boldsymbol{V}}_\beta(\boldsymbol{X})\boldsymbol{\phi}_\beta(\boldsymbol{Y}),\end{aligned} \tag{28}$$

where

$$\begin{aligned}\hat{\boldsymbol{U}}_\alpha(\boldsymbol{Y}) &= \textstyle\int_{\mathcal{X}} \boldsymbol{U}^p(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{\psi}_\alpha(\boldsymbol{X})\, d\rho_{\boldsymbol{X}}(\boldsymbol{X}) = \sum_{|\beta|=0}^{p-|\alpha|} U_{\alpha\beta}\boldsymbol{\phi}_\beta(\boldsymbol{Y}), \\ \hat{\boldsymbol{V}}_\beta(\boldsymbol{X}) &= \textstyle\int_{\mathcal{Y}} \boldsymbol{V}^p(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{\phi}_\beta(\boldsymbol{Y})\, d\rho_{\boldsymbol{Y}}(\boldsymbol{Y}) = \sum_{|\alpha|=0}^{p-|\beta|} V_{\alpha\beta}\boldsymbol{\psi}_\alpha(\boldsymbol{X}).\end{aligned} \tag{29}$$

The decompositions given by (28) result in a conditional expansions of the $\boldsymbol{U}^p(\boldsymbol{X},\boldsymbol{Y}) = \hat{\boldsymbol{U}}^p(\boldsymbol{X};\boldsymbol{Y})$ and $\boldsymbol{V}^p(\boldsymbol{X},\boldsymbol{Y}) = \hat{\boldsymbol{V}}^p(\boldsymbol{Y};\boldsymbol{X})$, where the coordinates

are now random variables themselves and can be obtained from the global expansion coordinates using (29). We have thus reduced the number of degrees of freedom of $\boldsymbol{U}^p$ and $\boldsymbol{V}^p$ from $\nu^p_{m_1+m_2}+1=\binom{p+m_1+m_2}{p}$ to $\nu^p_{m_1}+1=\binom{p+m_1}{p}$ and $\nu^p_{m_2}+1=\binom{p+m_2}{p}$ respectively. With respect to the model problem (1), we will now introduce a reduced or conditional Galerkin projection method where only the local uncertainties are embedded by the transformation and the external random variables are handled non-intrusively by an interfacing wrapper.

4.2. Conditional Galerkin projection method

In section 3.2, we reviewed the embedded solution strategy using SGS for a stochastic single component model. Following a local Galerkin projection protocol on the partitioned solution method (5), where we project the first Newton step on each polynomial basis function of $\boldsymbol{X}$ and project the second Newton step on each polynomial basis function of $\boldsymbol{Y}$, we obtain the following stochastic linear systems.

$$\begin{aligned}\left[\tilde{\boldsymbol{J}}^\ell_{\boldsymbol{F}}(\boldsymbol{Y})\right]\left(\tilde{\boldsymbol{U}}^{\ell+1}(\boldsymbol{Y})-\tilde{\boldsymbol{U}}^{\ell}(\boldsymbol{Y})\right)&=-\tilde{\boldsymbol{F}}^{\ell}(\boldsymbol{Y}),\\ \left[\tilde{\boldsymbol{J}}^\ell_{\boldsymbol{G}}(\boldsymbol{X})\right]\left(\tilde{\boldsymbol{V}}^{\ell+1}(\boldsymbol{X})-\tilde{\boldsymbol{V}}^{\ell}(\boldsymbol{X})\right)&=-\tilde{\boldsymbol{G}}^{\ell}(\boldsymbol{X}),\end{aligned} \tag{30}$$

where

$$\begin{aligned}&\tilde{\boldsymbol{J}}^\ell_{\boldsymbol{F}}(\boldsymbol{Y})=\textstyle\sum_{i=0}^{\nu^p_{m_1}}[E_{\mathcal{X},i}]\otimes\left[\hat{\boldsymbol{J}}^\ell_{\boldsymbol{F},i}(\boldsymbol{Y})\right],\tilde{\boldsymbol{J}}^\ell_{\boldsymbol{G}}(\boldsymbol{X})=\textstyle\sum_{i=0}^{\nu^p_{m_2}}[E_{\mathcal{Y},i}]\otimes\left[\hat{\boldsymbol{J}}^\ell_{\boldsymbol{G},i}(\boldsymbol{X})\right],\\ &\tilde{\boldsymbol{U}}^{\ell}(\boldsymbol{Y})=\begin{pmatrix}\hat{\boldsymbol{U}}^\ell_0(\boldsymbol{Y})\\ \vdots\\ \hat{\boldsymbol{U}}^\ell_{\nu^p_{m_1}}(\boldsymbol{Y})\end{pmatrix},\tilde{\boldsymbol{V}}^{\ell}(\boldsymbol{X})=\begin{pmatrix}\hat{\boldsymbol{V}}^\ell_0(\boldsymbol{X})\\ \vdots\\ \hat{\boldsymbol{V}}^\ell_{\nu^p_{m_2}}(\boldsymbol{X})\end{pmatrix},\\ &\tilde{\boldsymbol{F}}^{\ell}(\boldsymbol{Y})=\begin{pmatrix}\hat{\boldsymbol{F}}^\ell_0(\boldsymbol{Y})\\ \vdots\\ \hat{\boldsymbol{F}}^\ell_{\nu^p_{m_1}}(\boldsymbol{Y})\end{pmatrix},\tilde{\boldsymbol{G}}^{\ell}(\boldsymbol{X})=\begin{pmatrix}\hat{\boldsymbol{G}}^\ell_0(\boldsymbol{X})\\ \vdots\\ \hat{\boldsymbol{G}}^\ell_{\nu^p_{m_2}}(\boldsymbol{X})\end{pmatrix}.\end{aligned} \tag{31}$$

$[E_{\mathcal{X},i}]_{jk}=\int_{\mathcal{X}}\boldsymbol{\psi}_i(\boldsymbol{X})\boldsymbol{\psi}_j(\boldsymbol{X})\boldsymbol{\psi}_k(\boldsymbol{X})\,d\rho_{\boldsymbol{X}}$ for $0\le i,j,k\le\nu^p_{m_1}$ and $[E_{\mathcal{Y},i}]_{jk}=\int_{\mathcal{X}}\boldsymbol{\phi}_i(\boldsymbol{Y})\boldsymbol{\phi}_j(\boldsymbol{Y})\boldsymbol{\phi}_k(\boldsymbol{Y})\,d\rho_{\boldsymbol{Y}}$ for $0\le i,j,k\le\nu^p_{m_2}$. The residual coordinates are $\hat{\boldsymbol{F}}_i(\boldsymbol{Y})=\int_{\mathcal{X}}\boldsymbol{F}\left(\hat{\boldsymbol{U}}^{\ell,p};\hat{\boldsymbol{V}}^{\ell,p}\right)\boldsymbol{\psi}_i d\rho_{\boldsymbol{X}}$ and $\hat{\boldsymbol{G}}_i(\boldsymbol{X})=\int_{\mathcal{Y}}\boldsymbol{G}\left(\hat{\boldsymbol{V}}^{\ell,p};\hat{\boldsymbol{U}}^{\ell+1,p}\right)\boldsymbol{\phi}_i d\rho_{\boldsymbol{Y}}$; the Jacobian coordinates are $\left[\hat{\boldsymbol{J}}_{\boldsymbol{F},i}(\boldsymbol{Y})\right]=\int_{\mathcal{X}}\nabla_{\boldsymbol{U}}\boldsymbol{F}\left(\hat{\boldsymbol{U}}^{\ell,p};\hat{\boldsymbol{V}}^{\ell,p}\right)\boldsymbol{\psi}_i d\rho_{\boldsymbol{X}}$ and $\left[\hat{\boldsymbol{J}}_{\boldsymbol{G},i}(\boldsymbol{X})\right]=\int_{\mathcal{Y}}\nabla_{\boldsymbol{V}}\boldsymbol{G}\left(\hat{\boldsymbol{V}}^{\ell,p};\hat{\boldsymbol{U}}^{\ell+1,p}\right)\boldsymbol{\phi}_i d\rho_{\boldsymbol{Y}}$. Therefore, by using this reduced Galerkin projection method, each component solver or 'module' can embed its local uncertainties in the corresponding nonlinear Newton step. We therefore refer to the projections as the modularly embedded projections and refer to the first and second components of (5) as Module 1 and Module 2 respectively.

4.3. Interfacing wrappers

The conditional and global coordinates of the solution iterates needed are transformed between one another with the help of the respective restriction (scatter) and prolongation (gather) procedures. We make use of interfacing wrappers that apply these transformations as the iterates pass between the modules.

The transformation from the global coordinates to the conditional coordinates (restriction) has been formulated in (29). Applying the reverse transformation from the conditional coordinates to the global coordinates (prolongation) however, would require a numerical approximation due to the following property of the global coordinates.

$$U_{\alpha\beta} = \int_{\mathcal{Y}} \hat{\boldsymbol{U}}_\alpha(\boldsymbol{Y})\,\boldsymbol{\phi}_\beta(\boldsymbol{Y})\,d\rho_{\boldsymbol{Y}}(\boldsymbol{Y}),\ V_{\alpha\beta} = \int_{\mathcal{Y}} \hat{\boldsymbol{V}}_\beta(\boldsymbol{X})\,\boldsymbol{\psi}_\alpha(\boldsymbol{X})\,d\rho_{\boldsymbol{X}}(\boldsymbol{X}). \quad (32)$$

We could approximate (32) using a numerical integration rule on the respective external stochastic spaces. Instead we propose a simpler least squares method that allows for an arbitrary choice of points to be sampled. Before we further discuss this method, we define a matrix $[U] \in \mathbb{R}^{n_1 \times \left(\nu^p_{m_1+m_2}+1\right)}$ matrix with the global gPC coordinates of $\boldsymbol{U}^p \approx \boldsymbol{U}$ forming its columns. Similary, we define a matrix $\left[\hat{\boldsymbol{U}}(\boldsymbol{Y})\right] : \mathcal{Y} \to \mathbb{R}^{n_1 \times \left(\nu^p_{m_1}+1\right)}$ with the conditional gPC coordinates forming its columns. Instead of a total degree lexicographical arrangement, we let the columns of $[U]$ follow a graded lexicographical order i.e.

$$\mathcal{I}^p_{m_1+m_2} = \left\{\alpha_0\beta_0, \ldots, \alpha_{\nu^p_{m_1+m_2}}\beta_{\nu^p_{m_1+m_2}} \in \mathbb{N}^{m_1+m_2} : |\alpha|_0 \leq \cdots \leq |\alpha|_{\nu^p_{m_1+m_2}}\right\}. \quad (33)$$

We therefore construct a sparse linear transformation between $[U]$ and $\left[\hat{\boldsymbol{U}}\right]$ as follows.

$$\left[\hat{\boldsymbol{U}}(\boldsymbol{Y})\right] = [U]\,[\boldsymbol{\Pi}_p(\boldsymbol{Y})], \quad (34)$$

where $[\boldsymbol{\Pi}_p(\boldsymbol{Y})] \in \mathbb{R}^{\left(\nu^p_{m_1+m_2}+1\right)\times\left(\nu^p_{m_1}+1\right)}$ is a sparse matrix such that each row has at most one non-zero element. If we define $\boldsymbol{\Phi}^q_{m_2} = \begin{bmatrix} \boldsymbol{\phi}_0 & \cdots & \boldsymbol{\phi}_{\nu^q_{m_2}} \end{bmatrix}^T$ as the vector of polynomial basis functions upto total degree q and $\mathcal{I}^p_{m_1} = \left\{\alpha_0, \ldots, \alpha_{\nu^p_{m_1}} \in \mathbb{N}^{m_1} : |\alpha_0| \leq \cdots \leq \left|\alpha_{\nu^p_{m_1}}\right|\right\}$ as the index set of the conditional coordinates, then $[\boldsymbol{\Pi}_p]$ would have the following structure.

$$[\boldsymbol{\Pi}_p(\boldsymbol{Y})] = \begin{bmatrix} \boldsymbol{\Phi}^{p-|\alpha_0|}_{m_2}(\boldsymbol{Y}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \boldsymbol{\Phi}^{p-\left|\alpha_{\nu^p_{m_1}}\right|}_{m_2}(\boldsymbol{Y}) \end{bmatrix}. \quad (35)$$

Therefore, the linear transformation (34) is equivalent to (29). We can similarly define transformations for the field $\boldsymbol{V}^p \approx \boldsymbol{V}$ but the details have been omitted

here for the sake of brevity. Also, the first equation in (32) can be compactly written as follows.

$$[U] = \int_{\mathcal{Y}} \left[\hat{\boldsymbol{U}}\left(\boldsymbol{Y}\right)\right] \left[\boldsymbol{\Pi}_p\left(\boldsymbol{Y}\right)\right]^T d\rho_{\boldsymbol{Y}}\left(\boldsymbol{Y}\right). \tag{36}$$

4.3.1. Least squares prolongation

Given a sufficiently large set of points $\left\{Y^{(i)} \in \mathcal{Y} : 0 \leq i \leq q\right\}$, we construct the transformation matrices $\left\{\left[\boldsymbol{\Pi}_p\left(Y^{(i)}\right)\right] : 0 \leq i \leq q\right\}$ obtain samples of the conditional coordinate matrices $\left\{\left[\hat{\boldsymbol{U}}\left(Y^{(i)}\right)\right] : 0 \leq i \leq q\right\}$. The least squares method of approximating $[U]$ is therefore based on the following overdetermined system of equations.

$$[U]\left[\begin{array}{ccc}\left[\boldsymbol{\Pi}_p\left(Y^{(0)}\right)\right] & \cdots & \left[\boldsymbol{\Pi}_p\left(Y^{(q)}\right)\right]\end{array}\right] = \left[\begin{array}{ccc}\left[\hat{\boldsymbol{U}}\left(Y^{(0)}\right)\right] & \cdots & \left[\hat{\boldsymbol{U}}\left(Y^{(q)}\right)\right]\end{array}\right]. \tag{37}$$

Therefore, we get

$$[U] = \left[\sum_{i=0}^{q}\left[\hat{\boldsymbol{U}}\left(Y^{(i)}\right)\right]\left[\boldsymbol{\Pi}_p\left(Y^{(i)}\right)\right]^T\right]\left[\sum_{i=0}^{q}\left[\boldsymbol{\Pi}_p\left(Y^{(i)}\right)\right]\left[\boldsymbol{\Pi}_p\left(Y^{(i)}\right)\right]^T\right]^{-1}. \tag{38}$$

Based on the structure of $[\boldsymbol{\Pi}_p]$ given by (35), we see that $[\boldsymbol{\Pi}_p]\,[\boldsymbol{\Pi}_p]^T$ is block diagonal and therefore, the inverse matrix on the right hand side of (38) is also block diagonal. In general, we choose $q \geq 2\nu^p_{m_2}$ to ensure that the problem is not ill-conditioned.

Alternatively, we could use a weighted least-squares approach. We remark here that if the sampling points and weights are chosen according to a quadrature rule, the inverse matrix would be equal to the identity matrix.

5. Implementation on a stochastic multiphysics model

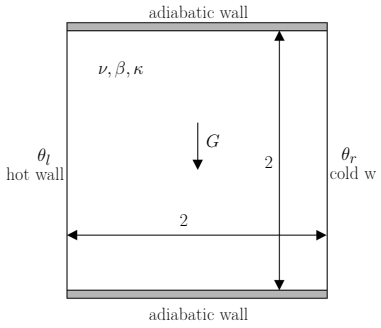

Fig 1: Geometry of spatial domain for modeling thermally driven cavity flow.

5.1. 2D Fluid-Thermal interactions in a thermally driven cavity

We consider a 2D square cavity $\Omega_S = [-1,1]^2$ in which an incompressible fluid is driven by the effect of temperature differences between the left and right walls (Fig 1). The top and botton walls are assumed to be adiabatic. We wish to model the interactions between the velocity field $\boldsymbol{U} = \begin{bmatrix} \boldsymbol{u} & \boldsymbol{v} \end{bmatrix}^T : [-1,1]^2 \times \Omega \to \mathbb{R}^2$ and internal energy field $\boldsymbol{e} : [-1,1]^2 \times \Omega \to \mathbb{R}$ of the fluid using the following coupled steady state PDE system

$$\begin{aligned} (\boldsymbol{U}\cdot\nabla)\boldsymbol{U} + (\gamma-1)\nabla\boldsymbol{e} - \boldsymbol{\nu}(\boldsymbol{x}_1)(\nabla\cdot\nabla)\boldsymbol{U} - G(1+\boldsymbol{\beta}(\boldsymbol{x}_2)\boldsymbol{e}) = 0, \\ (\boldsymbol{U}\cdot\nabla)\boldsymbol{e} - \kappa(\nabla\cdot\nabla)\boldsymbol{e} = 0, \end{aligned} \tag{39}$$

in $]-1,1[^2 \times \Omega$ with boundary conditions

$$\begin{aligned} \boldsymbol{U}|_{s_1=-1} = \boldsymbol{U}|_{s_1=1} = \boldsymbol{U}|_{s_2=-1}\boldsymbol{U}|_{s_2=1} = 0, \\ \boldsymbol{e}|_{s_1=-1} = c_p\boldsymbol{\theta}_l(\boldsymbol{y}_1),\ \boldsymbol{e}|_{s_1=1} = c_p\boldsymbol{\theta}_r(\boldsymbol{y}_2),\ \hat{n}_1\cdot\nabla\boldsymbol{e}|_{s_2=-1} = \hat{n}_1\cdot\nabla\boldsymbol{e}|_{s_2=1} = 0. \end{aligned} \tag{40}$$

The uncertain parameters are prescribed as follows.

$$\begin{aligned} \boldsymbol{\nu}(\boldsymbol{x}_1) &= \bar{\nu} + \sqrt{3}\nu'\boldsymbol{x}_1, \\ \boldsymbol{\beta}(\boldsymbol{x}_2) &= \bar{\beta} + \sqrt{3}\beta'\boldsymbol{x}_2, \\ \boldsymbol{\theta}_l(\boldsymbol{y}_1) &= \bar{\theta}_l + \sqrt{3}\theta_l'\boldsymbol{y}_1, \\ \boldsymbol{\theta}_r(\boldsymbol{y}_2) &= \bar{\theta}_r + \sqrt{3}\theta_r'\boldsymbol{y}_2, \end{aligned} \tag{41}$$

where $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are i.i.d uniform random variables in $[-1,1]$. For physical validity of our model, the mean and standard deviations of the uncertain parameters are chosen to ensure positive values of $\boldsymbol{\nu}$, $\boldsymbol{\beta}$, $\boldsymbol{\theta}_l$, $\boldsymbol{\theta}_r$ and $(\boldsymbol{\theta}_l - \boldsymbol{\theta}_r)$. Table 2 lists out the values prescribed for the deterministic parameters.

Parameter	Symbol	Value
Mean of fluid viscosity	$\bar{\nu}$	0.2
Mean of thermal expansion coefficient	$\bar{\beta}$	0.02
Mean of left wall temperature	$\bar{\theta}_l$	5
Mean of right wall temperature	$\bar{\theta}_r$	1
StDv of fluid viscosity	ν'	0.04
StDv of thermal expansion coefficient	β'	0.004
StDv of left wall temperature	θ_l'	1
StDv of right wall temperature	θ_r'	0.2
Heat capacity ratio	γ	1.4
Gravity vector	G	$\begin{bmatrix} 0 & -10 \end{bmatrix}^T$
Thermal diffusivity	κ	10
Specfic heat	c_p	2

Table 2: List of deterministic parameters and their values.

The spatial discretization of (39) and (40) is formulated on an $\mathcal{N}\times\mathcal{N}-$element uniform quadrilateral mesh, using bilinear shape functions and upwinding stabilization (SUPG) [26]. Therefore, we obtain a coupled system of field equations as per the model problem (1) where Module 1 updates the velocity field while Module 2 updates the internal energy field. We have $n_1 = 2\left(\mathcal{N}-1\right)^2$ solution variables in Module 1 and $n_2 = \left(\mathcal{N}-1\right)^2 + 2\left(\mathcal{N}-1\right)$ solution variables in Module 2.

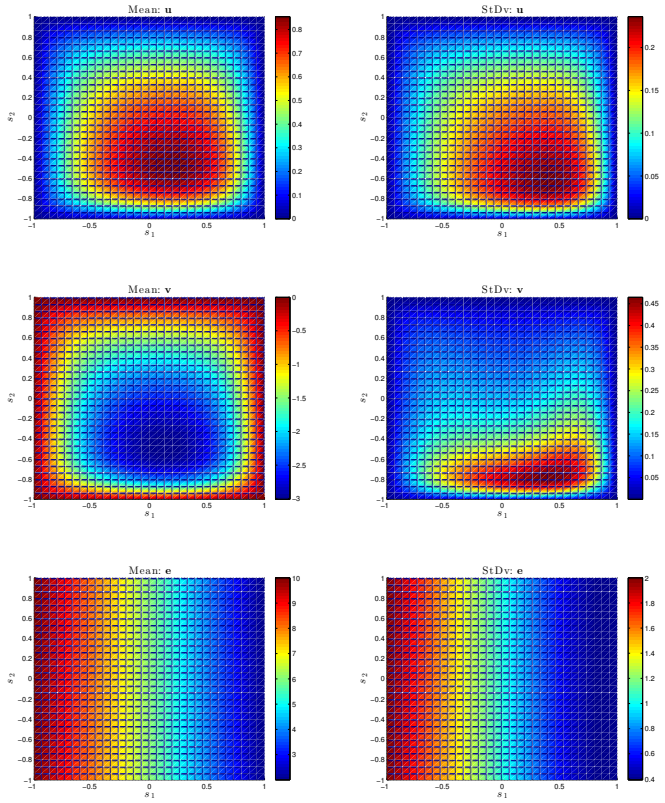

Fig 2: Spatial distribution of mean and standard deviations of velocity fields $\boldsymbol{u}$, $\boldsymbol{v}$, and internal energy field $\boldsymbol{e}$.

5.1.1. Mean and variance approximation

Following the modularly embedded projection method (30), we obtained the mean and standard deviation of the converged velocity and internal energy fields for $p = 3$ and $\mathcal{N} = 30$ (Fig 2). Retaining the spatial resolution of the solution, Fig 3 shows the convergence of the spatial average of the mean and standard deviation of $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{e}$ as p increases. The error is computed as the (absolute value) of the difference between an statistics computed at truncation order p and the statistics computed at truncation order $p-1$.

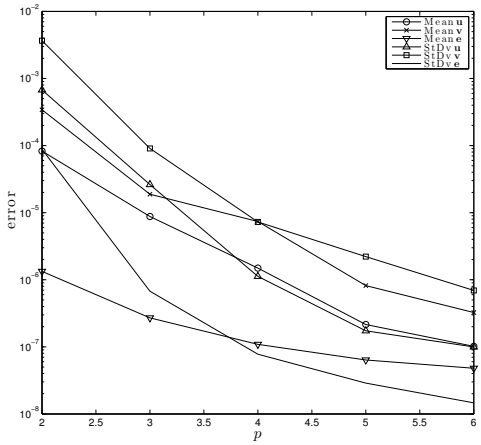

Fig 3: Convergence of ther spatial average of the mean and standard deviation approximations of $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{e}$ obtained using the global gPC coordinates.

5.1.2. Computational performance

For the implemented modularly embedded projection method, the total computational was obtained for various values of p and $\mathcal{N}$. We also implemented the monolithic Newton method (3) with a globally embedded Galerkin projection method and obtained the respective computational cost for various values of p and $\mathcal{N}$. The results are compared in Fig 4. We observe that for higher values of p, the modularly embedded projection method was computationally less expensive due to two main reasons. Firstly, the total number of iterations needed for both methods to converge did not increase significantly as p was successively increased (Fig 5). Secondly, when compared to the globally embedded projection method, the effective cost of solving a single Newton step with the modularly embedded projection method was significantly lower even with the sampling overheads from the wrappers (Fig 6). All experiments were performed in `MATLAB` on a single core Intel 3.1Ghz i5 CPU with 4GB RAM, and the inhouse BiCGStab solver was used solve the expensive Newton steps with a tolerance value of 10^{-10}.

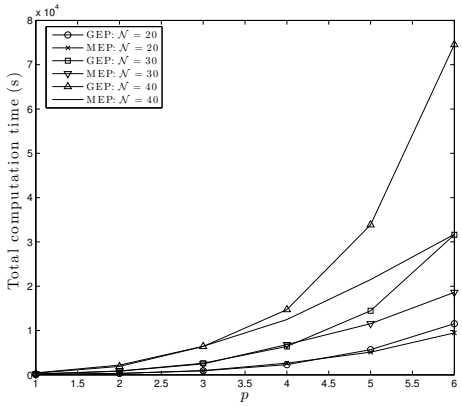

Fig 4: Total computation time of the modularly embedded projection (MEP) method compared to the monolithic globally embedded projection (GEP) method.

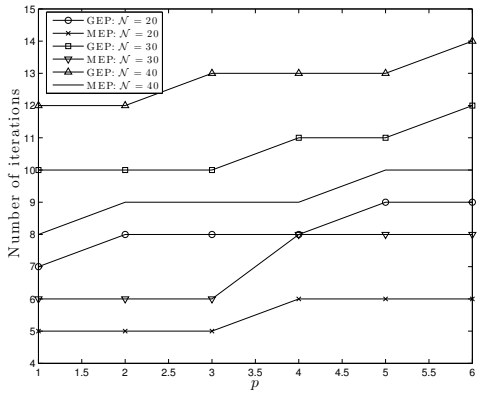

Fig 5: Comparison of the total number of iterations needed to converge.

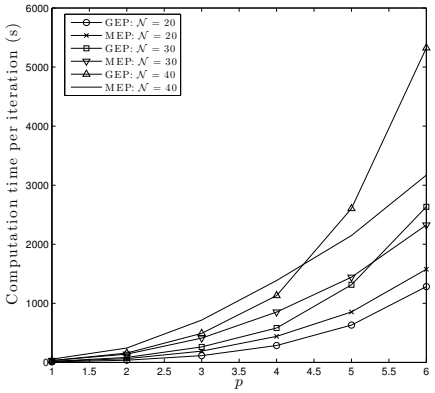

Fig 6: Comparison of the (average) computation time per iteration.

6. Conclusions and future work

We have demonstrated an efficient multicomponent approach for uncertainty propagation in stochastic multiphysics systems such as fluid-thermal interactions. The modularly embedded strategy of developing components presented can be easily integrated into a multicomponent solver using interfacing wrappers. Therefore, our method facilitates a modularly flexible framework for developing stochastic simulations of complex multiphysics models. Although we have only considered intrusive propagation methods within modules, our method can be easily adapted to non-intrusive sampling based propagation as well. Thus, our approach can in general be considered as a hybrid UQ methodology, which is a hot topic of research within the community.

However, we have yet not addressed key bottlenecks associated with the gPC representation of uncertainty such as the curse-of-dimension and long term integration. In the context of high-dimensional stochastic models, we are currently investigating methods of dimension-reduction in the inter-module sense, which is based on the contention that information exchanged between modules of a partitioned solution approach often resides in a lower dimensional space. Based on initial experimentation, we are confident about our claims and in the near future, we are condfident of incorporating an efficient dimension reduction methodology and extending our framework to handle very high-dimensional stochastic multiphysics models and move a significant step closer to mitigating the curse of dimension.

Acknowledgements

This research was funder by the U.S. Department of Energy - Office of Advanced Computing Research and Applied Mathematics Program and partially funded by the U.S. Department NNSA ASC Program. This collaborative research was carried out at Stanford University under Contract No. DE-AC52-07NA27344 under the auspices of the U.S. Deparment of Energy and Lawrence Livermore National Laboratory.

References

[1] (2013) X. Chen, Y. Sun, B. Ng and C. Tong. A flexible uncertainty quantification method for linearly coupled multiphysics systems. Journal of Computational Physics 248. pp 383-401.

[2] (2013) X. Chen, Y. Sun, B. Ng and C. Tong. A computational method for simulating subsurface flow and reactive transport in heteregenous porous media embedded with flexible UQ. Water Resources Research 49-9. pp 5740-5755.

[3] (1949) N. Metropolis and S. Ulam. The Monte Carlo Method. J. Amer. Stat. Assoc. 44, pp. 335-341.

[4] (2010) C. Roberts and G. Casella. Monte Carlo Statistical Methods. Springer, New York.

[5] (1985) W. Morokoff and R. Caflisch. Quasi-Monte Carlo integration. J. Computational Physics 122. pp. 218-230.

[6] (1979) M.C. McKay, R. Beckman and W. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21. pp 239-245.

[7] (2004) C. Soize and R. Ghanem. Physical systems with random uncertainties: chaos representations on arbitrary probability measures. SIAM J. on Scientific Computing 26. pp. 395-410.

[8] (2002) D. Xiu and G. Karniadakis. The Weiner-Askey Polynomial Chaos for stochastic differential equations. SIAM J. on Scientific Computing

[9] (2004) I. Babuska, R. Tempone and G. Zouraris. Galerkin Finite Element Approximations of Stochastic Elliptic PDEs. SIAM J. on Numerical Analysis 42. pp 800-825.

[10] (2007) D. Xiu. Efficient collocational approach for parametric uncertainty analysis. Comm. Computational Physics 2. pp. 293-309.

[11] (2005)D. Xiu and J. Hesthaven. High-order collocation methods for differential equations with random inputs. SIAM J. on Scientific Computing 27. pp. 1118-1139.

[12] (2009) M. Eldred and J. Burkhardt. Comparison of Polynomial Chaos and Collocation methods for uncertainty quantification. AIAA Paper 2009-0976 (0).

[13] (2010) P. Constantine, D. Gleich and G. Iaccarino. Spectral Methods for parameterized matrix equations. SIAM J. on Matrix Analysis and Applications.

[14] (2006) X. Wan and G. Karniadakis. *Multi-Element Generalized Polynomial Chaos for Arbitrary Probability Measures*. SIAM Journal on Scientific Computing Vol 28-3.

[15] (2010) M. Gerritsma, J. van der Steen, P. Vos and G. Karniadakis. *Time-dependent generalized polynomial chaos*. Journal of Computational Physics vol 229-22.

[16] (2012 pre-print) P. Constantine, E. Phipps and T. Wildey. Efficient uncertainty propagation for network multiphysics systems. Intl. J. for Numerical Methods in Engineering.

[17] (1969) T. Porsching. Jacobi and Gausssiedel methods for nonlinear network problems. SIAM J. on Numerical Analysis 6. 437-449

[18] (2009) H.N. Najm. *Uncertainty Quantification and Polynomial Chaos techniques in computational fluid dynamics*. Annual Review of Fluid Mechanics 41, 35-52.

[19] (1999) R. Ghanem and J.R. Red-Horse. *Propagation of probabilistic undertainty in complex physics systems using a stochastic finite element approach*. Physica vol 133-1.

[20] (2003) R. Ghanem and P. Spanos. Stochastic Finite Elements: A Spectral approach. Dover, Mineola, New York.

[21] (1944) R.H. Cameron and W.T. Martin. *Transformations of Weiner Integrals under translations*. Annals of Mathematics vol 48-2.

[22] (2012) P. Constantine, E. Phipps and M. Eldred. Sparse Pseudospectral Approximation Method. Comp. Methods in Applied Mechanics and Engineering.

[23] (2011) P. Constantine, D. Gleich and G. Iaccarino. A factorization of the spectral Galerkin system for parameterized matrix equations. SIAM J. on Scientific Computing

[24] (2009) D. Xiu and J. Shen. Stochastic Galerkin methods for Random Diffusion Equations. J of Computational Physics 228. pp. 266-281.

[25] (2009) C. Soize and R. Ghanem. Reduced Chaos Decomposition with Random Coefficients of Vector-Valued Random Variables and Random fields. Comp. Methods in Applied Mechanics and Engineering 98. pp 21-26.

[26] (2006) J. Shadid, A. Salinger and A. Pawlowski. Large-scale stabilized FE computational analysis of steady-state transport/reaction systems. Comp. Methods in Applied Mechanics and Engineering 95. pp 1846-1971

A review of dimension and model reduction methodologies in multiphysics systems with random inputs

A. Mittal, G. Iaccarino

Abstract

In this article, we review state-of-the-art methodologies proposed for mitigating the curse-of-dimensionality in tackling multiphysics systems that are parameterized using high-dimensional random inputs. We consider an extension of partitioned simulation strategies for strongly and weakly coupled problems wherein, parametric uncertainties are represented and propagated using polynomial chaos (gPC) expansions. In general, the number of independent random inputs in coupled problems can be large and render any standard implementation of gPC to be computationally intractable. However, exploiting the coupling structure can possibly lead to reduced dimension and reduced order stochastic models of the exchanged information, and thus lead to significant computational savings over the standard gPC approach. We explore this possibility and test the reduction methodologies on a thermally driven cavity flow problem.

Keywords: Uncertainty Quantification, Multiphysics systems, Dimension reduction, Model reduction, Polynomial chaos, Thermally driven cavity flow.

1. Introduction

With the aim of validating predictions from computer simulations against real-world phenomenon, it becomes necessary to include naturally random quantities i.e. uncertainties within the associated governing model. Uncertainties mainly exist due to two main reasons. Firstly, models are often idealized approximations of their target scenarios. Secondly, limitations of experimental techniques and measurements can lead to additional parametric uncertainties. Uncertainty quantification (UQ) has therefore, become a key requirement for achieving realistic predictive simulations.

Probability theory provides a unified mathematical framework for propagating modeling errors as well as parametric uncertainties. The first step in a probabilistic framework is to characterize input uncertainties as random variables or fields, based on methods borrowed from mathematical statistics [1] and inference analysis [2]. The second step is to propagate uncertainties from inputs to predictions by mapping their respective probabilistic representations. This process, known as uncertainty propagation, can be achieved using Monte Carlo

(MC) sampling methods [3] or stochastic spectral methods [4]. In the latter, we are faced with the objective of constructing an accurate polynomial surrogate response of the quantities of interest predicted by the simulations, which, in the context of UQ, is known as polynomial chaos (gPC) [5]. gPC methods have received special attention in the UQ community due to their advantages over traditional MC methods, and several approaches such as embedded projection [6], non-intrusive projection [7] and collocation [8] have been proposed for propagating the respective gPC coordinates.

Many of the current engineering and scientific problems that are being tackled involve a complex physical model covered by many distinct disciplines in physics and associated mathematics. A multiphysics system can, in general, be mathematically modeled as an algebraically coupled system of equations based on the interactions of two or more physical fields, domains, scales or a combination thereof. In addition, computational requirements such as algorithmic customization, independent modeling, reuse of legacy software and modularity favor a partitioned solution strategy over the simultaneous (monolithic) approach [9,10].

In extending this partitioned approach to gPC based stochastic simulations of coupled problems, we are often faced with the well known curse-of-dimensionality due to a large number of independent sources of uncertainties. This can lead to intractable computational costs in propagating uncertainties using monolithic (global) gPC approaches. While recent works have addressed this issue for special cases such as unidirectionally coupled problems [11] and linear multiphysics systems [12], we focus our discussions on gPC propagation in nonlinear and bidirectionally coupled problems. We assert that for strongly and weakly coupled models alike, the information that is communicated across various uniphysics components and iterations can be accurately represented using fewer stochastic degrees of freedom than the global gPC approach. This is particularly true when the solution field is smoothed between iterations using a forward operator, or when a fine-scale quantity is projected onto a coarse-scale representation in multiscale problems.

The remainder of this article is organized as follows. In Section 2, we briefly review the partitioned simulation strategy for strongly and weakly coupled deterministic multiphysics systems, as well as the non-intrusive projection method of globally propagating the gPC coordinates. In Section 3, we outline a dimension reduction method for strongly coupled systems that maintains the separability of the random inputs, as well as dimension reduction for weakly coupled systems using a composite gPC representation. In Section 4, we outline methods of model reduction that are based on a reduction of polynomial order and number of quadrature points. In Section 5, we implement and benchmark these methods on a thermally driven cavity flow problem.

Algorithm 1 Partitioned Newton method for strongly coupled problems

input $\boldsymbol{x}_1$, $\boldsymbol{x}_2$.
set $\boldsymbol{u}_1^0$, $\boldsymbol{u}_2^0$, $\boldsymbol{v}_1^0$ and $\boldsymbol{v}_2^0$.
set $\ell \leftarrow 0$.
do

$$\textbf{set } \boldsymbol{u}_1^{\ell+1} \leftarrow \boldsymbol{u}_1^{\ell} - \left[\frac{\partial \boldsymbol{f}_1\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_2^{\ell}, \boldsymbol{x}_1\right)}{\partial \boldsymbol{u}_1}\right]^{-1} \boldsymbol{f}_1\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_2^{\ell}, \boldsymbol{x}_1\right),$$

$$\textbf{set } \boldsymbol{v}_1^{\ell+1} \leftarrow \boldsymbol{g}_1\left(\boldsymbol{u}_1^{\ell+1}, \boldsymbol{x}_1\right).$$

$$\textbf{set } \boldsymbol{u}_2^{\ell+1} \leftarrow \boldsymbol{u}_2^{\ell} - \left[\frac{\partial \boldsymbol{f}_2\left(\boldsymbol{u}_2^{\ell}, \boldsymbol{v}_1^{\ell+1}, \boldsymbol{x}_2\right)}{\partial \boldsymbol{u}_2}\right]^{-1} \boldsymbol{f}_2\left(\boldsymbol{u}_2^{\ell}, \boldsymbol{v}_1^{\ell}, \boldsymbol{x}_2\right),$$

$$\textbf{set } \boldsymbol{v}_2^{\ell+1} \leftarrow \boldsymbol{g}_2\left(\boldsymbol{u}_2^{\ell+1}, \boldsymbol{x}_2\right).$$

$$\textbf{set } \ell \leftarrow \ell + 1.$$

while $\max\left(\|\boldsymbol{u}_1^{\ell} - \boldsymbol{u}_1^{\ell-1}\|, \|\boldsymbol{v}_1^{\ell} - \boldsymbol{v}_1^{\ell-1}\|, \|\boldsymbol{u}_2^{\ell} - \boldsymbol{u}_2^{\ell-1}\|, \|\boldsymbol{v}_2^{\ell} - \boldsymbol{v}_2^{\ell-1}\|\right) > \epsilon$

2. Multiphysics systems with random inputs

2.1. Model problem

In this article, we will devote our attention to obtaining the solution statistics and global sensitivites of the following non-linear and bidirectionally coupled system of equations.

$$\begin{array}{l} \boldsymbol{f}_1\left(\boldsymbol{u}_1, \boldsymbol{v}_2, \boldsymbol{x}_1\right) = 0;\ \boldsymbol{f}_1, \boldsymbol{u}_1 \in \mathbb{R}^{n_1}, \quad \boldsymbol{v}_1 = \boldsymbol{g}_1\left(\boldsymbol{u}_1, \boldsymbol{x}_1\right);\ \boldsymbol{g}_1, \boldsymbol{v}_1 \in \mathbb{R}^{m_1}, \\ \boldsymbol{f}_2\left(\boldsymbol{u}_2, \boldsymbol{v}_1, \boldsymbol{x}_2\right) = 0;\ \boldsymbol{f}_2, \boldsymbol{u}_2 \in \mathbb{R}^{n_2}, \quad \boldsymbol{v}_2 = \boldsymbol{g}_2\left(\boldsymbol{u}_2, \boldsymbol{x}_2\right);\ \boldsymbol{g}_2, \boldsymbol{v}_2 \in \mathbb{R}^{m_2}. \end{array} \tag{2.1}$$

We define m_1 and m_2 as the coupling dimensions, and s_1 and s_2 as the stochastic dimensions: $\boldsymbol{x}_1 \in \mathbb{R}^{s_1}$ with probability density function (PDF) ρ_1, and $\boldsymbol{x}_2 \in \mathbb{R}^{s_2}$ with PDF ρ_2. We assume that $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are independent. To avoid the numerical issues with gPC and long time integration, we will assume that (2.1) denotes a spatially discretized model of a steady state system of partial differential equations (PDE's).

2.2. Strong versus weak coupling

From the point of view of stochastic dimension reduction, we can classify the coupled problem (2.1), based on the coupling and stochastic dimensions, as either *strongly* coupled with $m_1 + m_2 \gg s_1 + s_2$ or *weakly* coupled with $m_1 + m_2 < s_1 + s_2$. In the latter case, the dimension of the stochastic space is reduced by reformulating the solution variables $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ as functions the coupling variables $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. We will discuss these further in subsequent sections after outlining the deterministic and stochastic simulation methods.

Algorithm 2 Nonlinear elimination method for weakly coupled problems

input $\boldsymbol{x}_1$, $\boldsymbol{x}_2$.
set $\boldsymbol{u}_1^0$, $\boldsymbol{u}_2^0$, $\boldsymbol{v}_1^0$ and $\boldsymbol{v}_2^0$.
set $\ell \leftarrow 0$.
do

$$\textbf{set } \boldsymbol{u}_1^{\ell+1} \leftarrow \boldsymbol{u}_1^{\ell} - \left[\frac{\partial \boldsymbol{f}_1\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_2^{\ell}, \boldsymbol{x}_1\right)}{\partial \boldsymbol{u}_1}\right]^{-1} \boldsymbol{f}_1\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_2^{\ell}, \boldsymbol{x}_1\right),$$

$$\textbf{set } [C_1] \leftarrow -\frac{\partial \boldsymbol{g}_1}{\partial \boldsymbol{u}_1}\left[\frac{\partial \boldsymbol{f}_1\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_2^{\ell}, \boldsymbol{x}_1\right)}{\partial \boldsymbol{u}_1}\right]^{-1} \frac{\partial \boldsymbol{f}_1}{\partial \boldsymbol{v}_2}.$$

$$\textbf{set } \boldsymbol{u}_2^{\ell+1} \leftarrow \boldsymbol{u}_2^{\ell} - \left[\frac{\partial \boldsymbol{f}_2\left(\boldsymbol{u}_2^{\ell}, \boldsymbol{v}_1^{\ell}, \boldsymbol{x}_2\right)}{\partial \boldsymbol{u}_2}\right]^{-1} \boldsymbol{f}_2\left(\boldsymbol{u}_2^{\ell}, \boldsymbol{v}_1^{\ell}, \boldsymbol{x}_2\right),$$

$$\textbf{set } [C_2] \leftarrow -\frac{\partial \boldsymbol{g}_2}{\partial \boldsymbol{u}_2}\left[\frac{\partial \boldsymbol{f}_2\left(\boldsymbol{u}_2^{\ell}, \boldsymbol{v}_1^{\ell}, \boldsymbol{x}_2\right)}{\partial \boldsymbol{u}_2}\right]^{-1} \frac{\partial \boldsymbol{f}_2}{\partial \boldsymbol{v}_1}.$$

$$\textbf{set } \begin{bmatrix} \boldsymbol{v}_1^{\ell+1} \\ \boldsymbol{v}_2^{\ell+1} \end{bmatrix} \leftarrow \begin{bmatrix} \boldsymbol{v}_1^{\ell} \\ \boldsymbol{v}_2^{\ell} \end{bmatrix} - \begin{bmatrix} [I_{m_1}] & [C_1] \\ [C_2] & [I_{m_2}] \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{v}_1^{\ell} - \boldsymbol{g}_1\left(\boldsymbol{u}_1^{\ell+1}, \boldsymbol{x}_1\right) \\ \boldsymbol{v}_2^{\ell} - \boldsymbol{g}_2\left(\boldsymbol{u}_2^{\ell+1}, \boldsymbol{x}_2\right) \end{bmatrix}.$$

set $\ell \leftarrow \ell + 1$.

while $\max\left(\|\boldsymbol{u}_1^{\ell} - \boldsymbol{u}_1^{\ell-1}\|, \|\boldsymbol{v}_1^{\ell} - \boldsymbol{v}_1^{\ell-1}\|, \|\boldsymbol{u}_2^{\ell} - \boldsymbol{u}_2^{\ell-1}\|, \|\boldsymbol{v}_2^{\ell} - \boldsymbol{v}_2^{\ell-1}\|\right) > \epsilon$

2.3. Partitioned Newton's method

For fixed values of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, i.e. the deterministic case, we consider a partitioned iterative solution method which reuses legacy solvers for each component $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$. In the strongly coupled case, we could implement partial Newton iterative procedure outlined in Algorithm 1. As stated by the Banach contraction mapping theorem, a sufficient condition for the partitioned Newton's method to converge is that the mapping from $\left(\boldsymbol{u}_1^{\ell}, \boldsymbol{v}_1^{\ell}, \boldsymbol{u}_2^{\ell}, \boldsymbol{v}_2^{\ell}\right)$ to $\left(\boldsymbol{u}_1^{\ell+1}, \boldsymbol{v}_1^{\ell+1}, \boldsymbol{u}_2^{\ell+1}, \boldsymbol{v}_2^{\ell+1}\right)$ is a contraction mapping.

For weakly coupled models, we could implement the nonlinear elimination based iterative procedure outlined in Algorithm 2, which has better convergence properties than the standard Newton's method outlined in Algorithm 1 [13].

2.4. Non-intrusive gPC propagation

We assume that the cost of executing each component solver is significant and prohibits the use of any exhaustive sampling approach. Therefore, we seek a polynomial surrogate of the solution and coupling variables using truncated gPC expansions. We assume here that they each satisfy the finite second-moment

[14]. At each iteration step ℓ, we define $\boldsymbol{y}_1(\boldsymbol{x}_1,\boldsymbol{x}_2) = \left[\boldsymbol{u}_1^{\ell}(\boldsymbol{x}_1,\boldsymbol{x}_2);\boldsymbol{v}_2^{\ell}(\boldsymbol{x}_1,\boldsymbol{x}_2)\right]$ as the inputs to the first component and $\boldsymbol{y}_2(\boldsymbol{x}_1,\boldsymbol{x}_2) = \left[\boldsymbol{u}_2^{\ell}(\boldsymbol{x}_1,\boldsymbol{x}_2);\boldsymbol{v}_1^{\ell}(\boldsymbol{x}_1,\boldsymbol{x}_2)\right]$ as the inputs to the second component.

Assuming that the PDF's of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are well known (e.g. uniform, Gaussian), we can construct ρ_1-orthonormal polynomials $\{\psi_1^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathbb{N}_0^{s_1}\}$ and ρ_2-orthonormal polynomials $\{\psi_2^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathbb{N}_0^{s_2}\}$ from univariate orthonormal polynomials. Therefore, the *global* gPC expansions are as follows.

$$\begin{aligned}\boldsymbol{y}_1(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|\geq 0} \boldsymbol{y}_1^{\boldsymbol{\alpha}\boldsymbol{\beta}} \psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2),\\ \boldsymbol{y}_2(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|\geq 0} \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} \psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2),\end{aligned} \tag{2.2}$$

Therefore, we can define the respective $p-$order truncated approximations as follows.

$$\begin{aligned}\hat{\boldsymbol{y}}_1^{p}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \boldsymbol{y}_1^{\boldsymbol{\alpha}\boldsymbol{\beta}} \psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2),\\ \hat{\boldsymbol{y}}_2^{p}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} \psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2).\end{aligned} \tag{2.3}$$

The gPC coordinates in (2.3) can be obtained using the non-intrusive spectral projection (NISP) [7] method by defining the respective quadrature rules $\left\{\boldsymbol{x}_1^{(j)}, w_1^{(j)}\right\}_{j=1}^{Q_1}$ and $\left\{\boldsymbol{x}_2^{(j)}, w_2^{(j)}\right\}_{j=1}^{Q_2}$ in each stochastic domain. We therefore approximate the gPC coordinates as follows.

$$\begin{aligned}\boldsymbol{y}_1^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \int_{\mathbb{R}^{s_2}}\int_{\mathbb{R}^{s_1}} \boldsymbol{y}_1(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2)\,\rho_1(\boldsymbol{x}_1)\,\rho_2(\boldsymbol{x}_2)\,d\boldsymbol{x}_1 d\boldsymbol{x}_2\\ &\approx \sum_{j=1}^{Q_1}\sum_{k=1}^{Q_2} \boldsymbol{y}_1\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1^{(j)}\right)\psi_2^{\boldsymbol{\beta}}\left(\boldsymbol{x}_2^{(k)}\right) w_1^{(j)} w_2^{(k)},\\ \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \int_{\mathbb{R}^{s_2}}\int_{\mathbb{R}^{s_1}} \boldsymbol{y}_2(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2)\,\rho_1(\boldsymbol{x}_1)\,\rho_2(\boldsymbol{x}_2)\,d\boldsymbol{x}_1 d\boldsymbol{x}_2\\ &\approx \sum_{j=1}^{Q_1}\sum_{k=1}^{Q_2} \boldsymbol{y}_2\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1^{(j)}\right)\psi_2^{\boldsymbol{\beta}}\left(\boldsymbol{x}_2^{(k)}\right) w_1^{(j)} w_2^{(k)}.\end{aligned} \tag{2.4}$$

Separate quadrature rules in each stochastic domain are required in order to approximate the coordinates of the following reduced [15] or *modular* gPC rep-

resentations of $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$.

$$\begin{aligned}\hat{\boldsymbol{y}}_1^p(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|=0}^{p} \tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1),\\ \hat{\boldsymbol{y}}_2^p(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|=0}^{p} \tilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\alpha}}(\boldsymbol{x}_2),\end{aligned} \tag{2.5}$$

where

$$\begin{aligned}\tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_2) &= \textstyle\int_{\mathbb{R}^{s_1}} \boldsymbol{y}_1(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\rho_1(\boldsymbol{x}_1)\,d\boldsymbol{x}_1\\ &\approx \sum_{j=1}^{Q_1} \boldsymbol{y}_1\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2\right)\psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1^{(j)}\right)w_1^{(j)},\\ \tilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}(\boldsymbol{x}_1) &= \textstyle\int_{\mathbb{R}^{s_2}} \boldsymbol{y}_2(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\psi_2^{\boldsymbol{\alpha}}(\boldsymbol{x}_2)\,\rho_2(\boldsymbol{x}_2)\,d\boldsymbol{x}_2\\ &\approx \sum_{j=1}^{Q_2} \boldsymbol{y}_2\left(\boldsymbol{x}_1,\boldsymbol{x}_2^{(j)}\right)\psi_2^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_2^{(j)}\right)w_2^{(j)}.\end{aligned} \tag{2.6}$$

In order to compute the global mean, covariance and sensitivity indices, we would require the global gPC coordinates, which can be obtained from the modular gPC coordinates as follows.

$$\begin{aligned}\boldsymbol{y}_i^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \textstyle\int_{\mathbb{R}^{s_2}} \tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_2)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2)\,\rho_2(\boldsymbol{x}_2)\,d\boldsymbol{x}_2\\ &\approx \sum_{j=0}^{Q_2} \tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_2)\,\psi_2^{\boldsymbol{\beta}}\left(\boldsymbol{x}_2^{(j)}\right)w_2^{(j)},\\ \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \textstyle\int_{\mathbb{R}^{s_1}} \tilde{\boldsymbol{y}}_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\rho_1(\boldsymbol{x}_1)\,d\boldsymbol{x}_1\\ &\approx \sum_{j=0}^{Q_1} \tilde{\boldsymbol{y}}_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2)\,\psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1^{(j)}\right)w_1^{(j)}.\end{aligned} \tag{2.7}$$

Although we have outlined the NISP method, the modular gPC coordinates could be obtained using embedded projection or regression methods instead. For strongly coupled models, samples of these coordinates are required as intermediate quantities in the dimension reduction procedure described in the next section.

3. Stochastic dimension reduction

3.1. Strong coupling case

Karhunen-Loeve expansion (KLE) is widely used as a reduced dimensional representation of random fields [16,17]. In the context of the coupled problem (2.1), the standard KLE representation of the exchanged random quantities $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ would be as follows.

$$\begin{aligned}\hat{\boldsymbol{y}}_1^{p,d}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \boldsymbol{\varphi}_{1,0} + \sum_{j=1}^{d} \varsigma_{1,j}\boldsymbol{\varphi}_{1,j}\vartheta_{1,j}(\boldsymbol{x}_1,\boldsymbol{x}_2),\\ \hat{\boldsymbol{y}}_2^{p,d}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \boldsymbol{\varphi}_{2,0} + \sum_{j=1}^{d} \varsigma_{2,j}\boldsymbol{\varphi}_{2,j}\vartheta_{2,j}(\boldsymbol{x}_1,\boldsymbol{x}_2),\end{aligned} \tag{3.1}$$

where

$$\begin{aligned}\vartheta_{1,j}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \vartheta_{1,j}^{\boldsymbol{\alpha}\boldsymbol{\beta}}\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2),\\ \vartheta_{2,j}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \vartheta_{2,j}^{\boldsymbol{\alpha}\boldsymbol{\beta}}\psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2).\end{aligned} \tag{3.2}$$

Since, $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are discrete approximations of spatially varying random fields, the orthogonality of the basis vectors in the KLE representation (3.1) would be satisfied in the functional or Hilbert space of basis vectors [18]. Therefore, we define the corresponding Gram matrices $[G_1]$ and $[G_2]$ as the positive definite weighting matrices for the following orthogonality conditions.

$$\boldsymbol{\varphi}_{1,j}^T[G_1]\,\boldsymbol{\varphi}_{1,k} = \delta_{jk},\ \boldsymbol{\varphi}_{2,j}^T[G_2]\,\boldsymbol{\varphi}_{2,k} = \delta_{jk}. \tag{3.3}$$

Similar to the modular gPC representation, we instead seek a *modular* KLE representation that maintains the segregation of the random inputs $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. We therefore define alternative reduced dimensional representations of $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ as follows.

$$\begin{aligned}\hat{\boldsymbol{y}}_1^{p,d_1}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \boldsymbol{\phi}_{1,0}(\boldsymbol{x}_1) + \sum_{j=1}^{d_1} \sigma_{1,j}\boldsymbol{\phi}_{1,j}(\boldsymbol{x}_1)\,\theta_{1,j}(\boldsymbol{x}_2),\\ \hat{\boldsymbol{y}}_1^{p,d_2}(\boldsymbol{x}_1,\boldsymbol{x}_2) &= \boldsymbol{\phi}_{2,0}(\boldsymbol{x}_2) + \sum_{j=1}^{d_2} \sigma_{2,j}\boldsymbol{\phi}_{2,j}(\boldsymbol{x}_2)\,\theta_{2,j}(\boldsymbol{x}_1),\end{aligned} \tag{3.4}$$

where

$$\begin{aligned}\theta_{1,j}\left(\boldsymbol{x}_2\right) &= \sum_{|\boldsymbol{\alpha}|=0}^{p} \theta_{1,j}^{\boldsymbol{\alpha}} \psi_2^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_2\right), \ \boldsymbol{\phi}_{1,j}\left(\boldsymbol{x}_1\right) = \sum_{|\boldsymbol{\alpha}|=0}^{p} \boldsymbol{\phi}_{1,j}^{\boldsymbol{\alpha}} \psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1\right), \\ \theta_{2,j}\left(\boldsymbol{x}_1\right) &= \sum_{|\boldsymbol{\alpha}|=0}^{p} \theta_{2,j}^{\boldsymbol{\alpha}} \psi_1^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_1\right), \ \boldsymbol{\phi}_{2,j}\left(\boldsymbol{x}_2\right) = \sum_{|\boldsymbol{\alpha}|=0}^{p} \boldsymbol{\phi}_{2,j}^{\boldsymbol{\alpha}} \psi_2^{\boldsymbol{\alpha}}\left(\boldsymbol{x}_2\right).\end{aligned} \tag{3.5}$$

The gPC coordinates in (3.5) can be obtained using samples of the modular gPC coordinates $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ as follows.

3.1.1. Reduced SVD using modular gPC coordinates

If we define $\left\{\tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}\left(x_2^{(j)}\right) : 0 \leq |\boldsymbol{\alpha}| \leq p\right\}_{j=1}^{Q_2}$ and $\left\{\tilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}\left(x_1^{(j)}\right) : 0 \leq |\boldsymbol{\alpha}| \leq p\right\}_{j=1}^{Q_1}$ as the corresponding sets of modular gPC coordinates that are sampled at the respective quadrature points, we have

$$\boldsymbol{\phi}_{1,0}^{\boldsymbol{\alpha}} = \sum_{j=1}^{Q_2} \tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}\left(x_2^{(j)}\right) w_2^{(j)}, \ \boldsymbol{\phi}_{2,0}^{\boldsymbol{\alpha}} = \sum_{j=1}^{Q_1} \tilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}\left(x_1^{(j)}\right) w_1^{(j)}. \tag{3.6}$$

Once the coordinates of the mean are computed, we define the normalized and zero−mean samples of the modular gPC coordinates as follows.

$$\begin{aligned}\boldsymbol{z}_1^{\boldsymbol{\alpha}}\left(x_2^{(j)}\right) &= \sqrt{w_2^{(j)}\left[G_1\right]}\left(\tilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}\left(x_2^{(j)}\right) - \boldsymbol{\phi}_{1,0}^{\boldsymbol{\alpha}}\right), \\ \boldsymbol{z}_2^{\boldsymbol{\alpha}}\left(x_1^{(j)}\right) &= \sqrt{w_1^{(j)}\left[G_2\right]}\left(\tilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}\left(x_1^{(j)}\right) - \boldsymbol{\phi}_{2,0}^{\boldsymbol{\alpha}}\right).\end{aligned} \tag{3.7}$$

Once these samples have been computed, we perform a reduced singular value decomposition (SVD) of the following data matrices $[Z_1]$ and $[Z_2]$.

$$\begin{aligned}[Z_1] &= \begin{bmatrix} \boldsymbol{z}_1^{\boldsymbol{\alpha}_1}\left(x_2^{(1)}\right) & \cdots & \boldsymbol{z}_1^{\boldsymbol{\alpha}_1}\left(x_2^{(Q_2)}\right) \\ \vdots & & \vdots \\ \boldsymbol{z}_1^{\boldsymbol{\alpha}_{P_1}}\left(x_2^{(1)}\right) & \cdots & \boldsymbol{z}_1^{\boldsymbol{\alpha}_{P_1}}\left(x_2^{(Q_2)}\right) \end{bmatrix}, \\ [Z_2] &= \begin{bmatrix} \boldsymbol{z}_2^{\boldsymbol{\alpha}_1}\left(x_1^{(1)}\right) & \cdots & \boldsymbol{z}_2^{\boldsymbol{\alpha}_1}\left(x_1^{(Q_1)}\right) \\ \vdots & & \vdots \\ \boldsymbol{z}_2^{\boldsymbol{\alpha}_{P_2}}\left(x_1^{(1)}\right) & \cdots & \boldsymbol{z}_1^{\boldsymbol{\alpha}_{P_2}}\left(x_1^{(Q_1)}\right) \end{bmatrix},\end{aligned} \tag{3.8}$$

where $P_1 = \binom{s_1+p}{p}$ and $P_2 = \binom{s_2+p}{p}$ denote the size of the respective sets of ρ_1-orthonormal and ρ_2-orthonormal polynomials of degree less than or equal

to p. The reduced SVD factorizes the data matrices as follows.

$$[Z_1] = \underbrace{\begin{bmatrix} \hat{\boldsymbol{\phi}}_{1,1}^{\boldsymbol{\alpha}_1} & \cdots & \hat{\boldsymbol{\phi}}_{1,d_1}^{\boldsymbol{\alpha}_1} \\ \vdots & & \vdots \\ \hat{\boldsymbol{\phi}}_{1,1}^{\boldsymbol{\alpha}_{P_1}} & \cdots & \hat{\boldsymbol{\phi}}_{1,d_1}^{\boldsymbol{\alpha}_{P_1}} \end{bmatrix}}_{[U_1]} \underbrace{\begin{bmatrix} \sigma_{1,1} & & \\ & \ddots & \\ & & \sigma_{1,d_1} \end{bmatrix}}_{[\Sigma_1]} \underbrace{\begin{bmatrix} \hat{\theta}_{1,1}^{(1)} & \cdots & \hat{\theta}_{1,d_1}^{(1)} \\ \vdots & & \vdots \\ \hat{\theta}_{1,1}^{(Q_2)} & \cdots & \hat{\theta}_{1,d_1}^{(Q_2)} \end{bmatrix}^T}_{[V_1]^T},$$

$$[Z_2] = \underbrace{\begin{bmatrix} \hat{\boldsymbol{\phi}}_{2,1}^{\boldsymbol{\alpha}_1} & \cdots & \hat{\boldsymbol{\phi}}_{2,d_2}^{\boldsymbol{\alpha}_1} \\ \vdots & & \vdots \\ \hat{\boldsymbol{\phi}}_{2,1}^{\boldsymbol{\alpha}_{P_2}} & \cdots & \hat{\boldsymbol{\phi}}_{2,d_2}^{\boldsymbol{\alpha}_{P_2}} \end{bmatrix}}_{[U_2]} \underbrace{\begin{bmatrix} \sigma_{2,1} & & \\ & \ddots & \\ & & \sigma_{2,d_2} \end{bmatrix}}_{[\Sigma_2]} \underbrace{\begin{bmatrix} \hat{\theta}_{2,1}^{(1)} & \cdots & \hat{\theta}_{2,d_2}^{(1)} \\ \vdots & & \vdots \\ \hat{\theta}_{2,1}^{(Q_1)} & \cdots & \hat{\theta}_{2,d_2}^{(Q_1)} \end{bmatrix}^T}_{[V_2]^T}. \tag{3.9}$$

Finally, we can obtain the coordinates in (3.5) as follows.

$$\begin{aligned} \theta_{1,j}^{\boldsymbol{\alpha}} &= \sum_{k=1}^{Q_2} \hat{\theta}_{1,j}^{(k)} \psi_2^{\boldsymbol{\alpha}} \left(\boldsymbol{x}_2^{(k)} \right) \sqrt{w_2^{(k)}}, \ \boldsymbol{\phi}_{1,j}^{\boldsymbol{\alpha}} = \sqrt{[G_1]^{-1}} \hat{\boldsymbol{\phi}}_{1,j}^{\boldsymbol{\alpha}}, \\ \theta_{2,j}^{\boldsymbol{\alpha}} &= \sum_{k=1}^{Q_1} \hat{\theta}_{2,j}^{(k)} \psi_1^{\boldsymbol{\alpha}} \left(\boldsymbol{x}_1^{(k)} \right) \sqrt{w_1^{(k)}}, \ \boldsymbol{\phi}_{2,j}^{\boldsymbol{\alpha}} = \sqrt{[G_2]^{-1}} \hat{\boldsymbol{\phi}}_{2,j}^{\boldsymbol{\alpha}}. \end{aligned} \tag{3.10}$$

The dimensions d_1 and d_2 are selected based on a tolerance ε that indicates the fraction of the total variance that is ignored by the reduced SVD. Therefore, we first compute the s_2-rank SVD of $[Z_1]$ and s_1-rank SVD of $[Z_2]$ and select the dimensions as follows.

$$d_1 = \min_{j \in \mathbb{N}} \left\{ \sum_{k=j+1}^{s_2} \sigma_{2,k} \leq \varepsilon \sum_{k=1}^{s_2} \sigma_{1,k} \right\}, \ d_2 = \min_{j \in \mathbb{N}} \left\{ \sum_{k=j+1}^{s_1} \sigma_{2,k} \leq \varepsilon \sum_{k=1}^{s_1} \sigma_{2,k} \right\}.$$

3.1.2. gPC approximation in reduced dimensions

We define $\boldsymbol{\theta}_i = [\theta_{i,1}; \ldots; \theta_{i,d_i}] : i \in \{1, 2\}$ as the reduced dimensional random vectors. Also, ϱ_1 and ϱ_2 are defined as the PDF's of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ respectively. The modular gPC approximations using the reduced random variables can be formulated as follows.

$$\begin{aligned} \hat{\boldsymbol{y}}_1^{p,d_1} (\boldsymbol{x}_1, \boldsymbol{x}_2) = \widehat{\boldsymbol{y}}_1^p (\boldsymbol{x}_1, \boldsymbol{\theta}_1) = \sum_{|\boldsymbol{\alpha}|=0}^{p} \widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}} (\boldsymbol{\theta}_1) \psi_1^{\boldsymbol{\alpha}} (\boldsymbol{x}_1), \\ \hat{\boldsymbol{y}}_2^{p,d_2} (\boldsymbol{x}_1, \boldsymbol{x}_2) = \widehat{\boldsymbol{y}}_2^p (\boldsymbol{x}_2, \boldsymbol{\theta}_2) = \sum_{|\boldsymbol{\alpha}|=0}^{p} \widetilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}} (\boldsymbol{\theta}_2) \psi_2^{\boldsymbol{\alpha}} (\boldsymbol{x}_2). \end{aligned} \tag{3.11}$$

where $\widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_1)$ and $\widetilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_2)$ are approximated using (2.4). Similarly, the global gPC approximation can be defined as follows.

$$\begin{aligned}\widehat{\boldsymbol{y}}_1^p(\boldsymbol{x}_1,\boldsymbol{\theta}_1) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \boldsymbol{y}_1^{\boldsymbol{\alpha}\boldsymbol{\beta}} \psi_1^{\boldsymbol{\alpha}}(\boldsymbol{x}_1)\,\pi_1^{\boldsymbol{\beta}}(\boldsymbol{\theta}_1),\\ \widehat{\boldsymbol{y}}_2^p(\boldsymbol{x}_2,\boldsymbol{\theta}_2) &= \sum_{|\boldsymbol{\alpha}|+|\boldsymbol{\beta}|=0}^{p} \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} \pi_2^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_2)\,\psi_2^{\boldsymbol{\beta}}(\boldsymbol{x}_2).\end{aligned} \tag{3.12}$$

The ϱ_1- orthonormal polynomials $\left\{\pi_1^{\boldsymbol{\alpha}} : |\boldsymbol{\alpha}| \in \mathbb{N}_0^{d_1}\right\}$ and ϱ_2-orthonormal polynomials $\left\{\pi_2^{\boldsymbol{\alpha}} : |\boldsymbol{\alpha}| \in \mathbb{N}_0^{d_2}\right\}$ in (3.12) cannot be constructed using univariate polynomials and a Cholesky factorization method, outlined in Section 4, is used instead. We remark here that the same notation used for the global gPC coordinates in (2.2) has been used in (3.12). Similar to (2.7), the transformations from the modular to global gPC coordinates would be as follows.

$$\begin{aligned}\boldsymbol{y}_1^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \textstyle\int_{\mathbb{R}^{d_1}} \widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_1)\,\pi_1^{\boldsymbol{\beta}}(\boldsymbol{\theta}_1)\,\varrho_1(\boldsymbol{\theta}_1)\,d\boldsymbol{\theta}_1\\ &= \textstyle\int_{\mathbb{R}^{s_2}} \widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_1(\boldsymbol{x}_2))\,\pi_1^{\boldsymbol{\beta}}(\boldsymbol{\theta}_1(\boldsymbol{x}_2))\,\rho_2(\boldsymbol{x}_2)\,d\boldsymbol{x}_2\\ &\approx \sum_{j=1}^{Q_2} \widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}}\left(\boldsymbol{\theta}_1\left(\boldsymbol{x}_2^{(j)}\right)\right)\pi_1^{\boldsymbol{\beta}}\left(\boldsymbol{\theta}_1\left(\boldsymbol{x}_2^{(j)}\right)\right) w_2^{(j)}.\\ \boldsymbol{y}_2^{\boldsymbol{\alpha}\boldsymbol{\beta}} &= \textstyle\int_{\mathbb{R}^{d_2}} \widetilde{\boldsymbol{y}}_2^{\boldsymbol{\beta}}(\boldsymbol{\theta}_2)\,\pi_2^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_2)\,\varrho_2(\boldsymbol{\theta}_2)\,d\boldsymbol{\theta}_2\\ &= \textstyle\int_{\mathbb{R}^{s_1}} \widetilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_2(\boldsymbol{x}_1))\,\pi_2^{\boldsymbol{\alpha}}(\boldsymbol{\theta}_2(\boldsymbol{x}_1))\,\rho_1(\boldsymbol{x}_1)\,d\boldsymbol{x}_1\\ &\approx \sum_{j=1}^{Q_1} \widetilde{\boldsymbol{y}}_2^{\boldsymbol{\beta}}\left(\boldsymbol{\theta}_2\left(\boldsymbol{x}_1^{(j)}\right)\right)\pi_2^{\boldsymbol{\alpha}}\left(\boldsymbol{\theta}_2\left(\boldsymbol{x}_1^{(j)}\right)\right) w_1^{(j)}.\end{aligned} \tag{3.13}$$

This indicates that respective quadrature rules $\left\{\boldsymbol{\theta}_1^{(j)} = \boldsymbol{\theta}_1\left(\boldsymbol{x}_2^{(j)}\right), w_2^{(j)}\right\}_{j=1}^{Q_2}$ and $\left\{\boldsymbol{\theta}_2^{(j)} = \boldsymbol{\theta}_2\left(\boldsymbol{x}_1^{(j)}\right), w_1^{(j)}\right\}_{j=1}^{Q_1}$ can be defined. Since these rules contain the same number of points, and therefore the same number of solver executions, we have still not reduced the computational effort. Therefore, we would need to construct a reduced set of quadrature points and weights, which is outlined in Section 4.

3.1.3. Limitations and Effectiveness

Because the dimension reduction procedure described would initially require the modular gPC coordinates, it can only be implemented in the unidirectional

sense. In the context of the model problem (2.1), this means that any computational gains achieved in reducing the number of quadrature points would be limited to a single component. The same limitations would arise if the standard KLE representation (3.1) is implemented [19]. Therefore, the effectiveness of the method would depend on the difference between the stochastic dimensions s_1 and s_2. If the difference is high, then the computational gains in the component that has a smaller stochastic dimension would be high. Extending the method to a suitable bidirectional implementation is still an area of active investigation.

3.2. Weak coupling case

When the overall stochastic dimension $s_1 + s_2$ is larger than the coupling dimension $m_1 + m_2$, we can implement a composite gPC representation [20] in the coupling variables $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ to obtain a reduced dimensional model of the of the exchanged information $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$. We define ϱ_1 and ϱ_2 as the PDF's of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ respectively with $\boldsymbol{v} = [\boldsymbol{v}_1; \boldsymbol{v}_2]$. Therefore, we can define the truncated gPC approximations as follows.

$$\begin{aligned}
\boldsymbol{y}_1\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right) &\approx \widehat{\boldsymbol{y}}_1^{p'}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right) = \sum_{|\boldsymbol{\alpha}|=0}^{p'} \widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}} \pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right), \\
\boldsymbol{y}_2\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right) &\approx \widehat{\boldsymbol{y}}_2^{p'}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right) = \sum_{|\boldsymbol{\alpha}|=0}^{p'} \widetilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}} \pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right),
\end{aligned} \tag{3.14}$$

where

$$\begin{aligned}
\widetilde{\boldsymbol{y}}_1^{\boldsymbol{\alpha}} &= \int_{\mathbb{R}^{m_2}}\int_{\mathbb{R}^{m_1}} \boldsymbol{y}_1\left(\boldsymbol{v}_1,\boldsymbol{v}_2\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}_1,\boldsymbol{v}_2\right)\varrho_1\left(\boldsymbol{v}_1\right)\varrho_2\left(\boldsymbol{v}_2\right)d\boldsymbol{v}_1 d\boldsymbol{v}_2 \\
&= \int_{\mathbb{R}^{s_2}}\int_{\mathbb{R}^{s_1}} \boldsymbol{y}_1\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right)\rho_1\left(\boldsymbol{x}_1\right)\rho_2\left(\boldsymbol{x}_2\right)d\boldsymbol{x}_1 d\boldsymbol{x}_2 \\
&\approx \sum_{j=1}^{Q_1}\sum_{k=1}^{Q_2} \boldsymbol{y}_1\left(\boldsymbol{v}\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\right) w_1^{(j)} w_2^{(k)}, \\
\widetilde{\boldsymbol{y}}_2^{\boldsymbol{\alpha}} &= \int_{\mathbb{R}^{m_2}}\int_{\mathbb{R}^{m_1}} \boldsymbol{y}_2\left(\boldsymbol{v}_1,\boldsymbol{v}_2\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}_1,\boldsymbol{v}_2\right)\varrho_1\left(\boldsymbol{v}_1\right)\varrho_2\left(\boldsymbol{v}_2\right)d\boldsymbol{v}_1 d\boldsymbol{v}_2 \\
&= \int_{\mathbb{R}^{s_2}}\int_{\mathbb{R}^{s_1}} \boldsymbol{y}_2\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1,\boldsymbol{x}_2\right)\right)\rho_1\left(\boldsymbol{x}_1\right)\rho_2\left(\boldsymbol{x}_2\right)d\boldsymbol{x}_1 d\boldsymbol{x}_2 \\
&\approx \sum_{j=1}^{Q_1}\sum_{k=1}^{Q_2} \boldsymbol{y}_2\left(\boldsymbol{v}\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\right)\pi^{\boldsymbol{\alpha}}\left(\boldsymbol{v}\left(\boldsymbol{x}_1^{(j)},\boldsymbol{x}_2^{(k)}\right)\right) w_1^{(j)} w_2^{(k)},
\end{aligned} \tag{3.15}$$

In general, we assume that $p' \leq p$. Therefore, the cardinality of the set of orthonormal polynomials in $\boldsymbol{v}$ would be $P' = \binom{p'+m_1+m_2}{p'}$ and significantly less than the cardinality of the original set of orthonormal polynomials $P = \binom{p+s_1+s_2}{p}$. However, since the components of the solver need to be executed

at various values of the random inputs $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, computational gains can be achieved only by reducing the number of quadrature points that are required to compute the composite gPC coordinates in (3.15). The computational construction of the gPC basis and quadarature rules is described in the next section.

4. Stochastic model reduction

In the previous section, a reduced dimensional gPC approximation of the exchanged information was formulated for both strongly and weakly coupled models. We will now describe the computational construction of orthonormal polynomials and optimally sparse quadrature rules for a reduced dimensional random vector. Before we proceed with describing the methods, we define $\boldsymbol{x} \in \mathbb{R}^s$ as the original random vector with PDF ρ and $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}) \in \mathbb{R}^m$ as the reduced dimensional random vector with PDF ϱ ($m < s$). Also, we let $\left\{\boldsymbol{x}^{(j)}, w^{(j)}\right\}_{j=1}^{Q}$ and $\left\{\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right), w^{(j)}\right\}_{j=1}^{Q}$ be the respective quadrature rules in $\boldsymbol{x}$ and $\boldsymbol{v}$ with $\boldsymbol{w} = \left[w^{(1)}; \cdots; w^{(Q)}\right]$.

4.1. Reduced dimensional gPC basis

We define $P' = \binom{p'+m}{p'}$ as the cardinality of the set of orthonormal polynomials $\{\pi^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathbb{N}_0^m, 0 \leq |\boldsymbol{\alpha}| \leq p'\}$ that need to be computed. We define $(\boldsymbol{v})^{\boldsymbol{\alpha}}$ as the multivariate monomial in $\boldsymbol{v} = [v_1; \cdots; v_m]$ with exponent $\boldsymbol{\alpha} = [\alpha_1; \cdots; \alpha_m]$, which has the following product form.

$$(\boldsymbol{v})^{\boldsymbol{\alpha}} = \prod_{j=1}^{m} v_j^{\alpha_j}. \tag{4.1}$$

We then define a Hankel matrix $[H]$ as follows.

$$\begin{aligned}
[H] &= \int_{\mathbb{R}^m} \begin{bmatrix} (\boldsymbol{v})^{\boldsymbol{\alpha}_1} \\ \vdots \\ (\boldsymbol{v})^{\boldsymbol{\alpha}_{P'}} \end{bmatrix} \begin{bmatrix} (\boldsymbol{v})^{\boldsymbol{\alpha}_1} \\ \vdots \\ (\boldsymbol{v})^{\boldsymbol{\alpha}_{P'}} \end{bmatrix}^T \varrho(\boldsymbol{v})\, d\boldsymbol{v} \\
&= \int_{\mathbb{R}^s} \begin{bmatrix} (\boldsymbol{v}(\boldsymbol{x}))^{\boldsymbol{\alpha}_1} \\ \vdots \\ (\boldsymbol{v}(\boldsymbol{x}))^{\boldsymbol{\alpha}_{P'}} \end{bmatrix} \begin{bmatrix} (\boldsymbol{v}(\boldsymbol{x}))^{\boldsymbol{\alpha}_1} \\ \vdots \\ (\boldsymbol{v}(\boldsymbol{x}))^{\boldsymbol{\alpha}_{P'}} \end{bmatrix}^T \rho(\boldsymbol{x})\, d\boldsymbol{x} \\
&\approx \sum_{j=1}^{Q} \begin{bmatrix} \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_1} \\ \vdots \\ \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_{P'}} \end{bmatrix} \begin{bmatrix} \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_1} \\ \vdots \\ \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_{P'}} \end{bmatrix}^T w^{(j)},
\end{aligned} \tag{4.2}$$

where $|\boldsymbol{\alpha}_1| \leq \cdots \leq |\boldsymbol{\alpha}_{P'}|$. The Cholesky factorization of $[H]$ would yield a lower triangular matrix $[L]$ such that $[H] = [L]\,[L]^T$. From the vector of monomials

used in (4.2), we can obtain the vector of orthonormal polynomials as follows.

$$\begin{bmatrix} \pi^{\boldsymbol{\alpha}_1}\left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right) \\ \vdots \\ \pi^{\boldsymbol{\alpha}_{P'}}\left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right) \end{bmatrix} = [L]^{-1} \begin{bmatrix} \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_1} \\ \vdots \\ \left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right)^{\boldsymbol{\alpha}_{P'}} \end{bmatrix}. \tag{4.3}$$

We remark here that the left hand side quantity is needed in the formulations (3.13) and (3.15).

4.2. Optimally sparse quadrature rules

In the previous section, we concluded that a reduction of the number of quadrature points needed in approximating the gPC coordinates was essential from the point of view of computational efficiency. If the weight corresponding to a particular point is zero, then that point can effectively be excluded from the quadrature rule. Therefore, we seek an optimally sparse vector of weights $\hat{\boldsymbol{w}} = \left[\hat{w}^{(1)}; \cdots; \hat{w}^{(Q)}\right]$ using the following $\mathcal{L}^0-$minimization problem.

$$\hat{\boldsymbol{w}} = \arg\min_{\boldsymbol{\nu}\in\mathbb{R}^Q} \|\boldsymbol{\nu}\|_{\mathcal{L}^0} : \ [M]\,\boldsymbol{w} = [M]\,\boldsymbol{\nu}, \tag{4.4}$$

where

$$[M] = \begin{bmatrix} \left(\boldsymbol{v}\left(\boldsymbol{x}^{(1)}\right)\right)^{\boldsymbol{\alpha}_1} & \cdots & \left(\boldsymbol{v}\left(\boldsymbol{x}^{(Q)}\right)\right)^{\boldsymbol{\alpha}_1} \\ \vdots & & \vdots \\ \left(\boldsymbol{v}\left(\boldsymbol{x}^{(1)}\right)\right)^{\boldsymbol{\alpha}_{N'}} & \cdots & \left(\boldsymbol{v}\left(\boldsymbol{x}^{(Q)}\right)\right)^{\boldsymbol{\alpha}_{N'}} \end{bmatrix} \tag{4.5}$$

is the monomial matrix and $N' = \binom{2p'+m-1}{m}$. We convert the $\mathcal{L}^0-$minimization problem into an $\mathcal{L}^1-$minimization problem as follows.

$$\hat{\boldsymbol{w}} = \arg\min_{\boldsymbol{\nu}\in\mathbb{R}^Q} \|\boldsymbol{\nu}\|_{\mathcal{L}^1} : \ [M]\,\boldsymbol{w} = [M]\,\boldsymbol{\nu}. \tag{4.6}$$

Following the approach in [21], we reformulate (4.6) as a linear minimization as follows.

$$[\hat{\boldsymbol{z}}; \hat{\boldsymbol{w}}] = \arg \min_{\substack{\boldsymbol{\nu}\,\in\,\mathbb{R}^Q, \\ \boldsymbol{\zeta}\,\in\,\mathbb{R}^Q}} \boldsymbol{e}^T\boldsymbol{\zeta} : \ [M]\,\boldsymbol{w} = [M]\,\boldsymbol{\nu}, \ \boldsymbol{\zeta} \succeq \boldsymbol{0}, \ -\boldsymbol{\zeta} \preceq \boldsymbol{\nu} \preceq \boldsymbol{\zeta}, \tag{4.7}$$

where $\boldsymbol{e} = [1; \cdots; 1]$ is the vector of ones. The linear programming (LP) approach can yield an optimal solution which may not be optimally sparse. The Tchakaloff's theorem [22] guarantees a unique solution that has the minimum number of non-zeros. However, we can implement Algorithm 3 to obtain a degenerate LP optimal solution that is optimally sparse.

Algorithm 3 Extracting a sparse optimal solution from the LP optimal solution

input LP optimal solution $\hat{\boldsymbol{w}}$.

do

$$\textbf{set } \mathcal{A} \leftarrow \left\{1 \leq j \leq Q : \hat{w}^{(j)} \neq 0\right\}.$$
$$\textbf{set } [M_{\mathcal{A}}] \leftarrow \mathcal{A}^{th} \textbf{ rows of } [M].$$
$$\textbf{find } \boldsymbol{p} = [p_1; \cdots; p_Q] \in \ker [M_{\mathcal{A}}].$$
$$\textbf{set } \delta \leftarrow \min_{j \in \mathcal{A}} \left\{ \left| \frac{p_j}{w^{(j)}} \right| : p_j w^{(j)} < 0 \right\}.$$
$$\textbf{set } \hat{\boldsymbol{w}} \leftarrow \hat{\boldsymbol{w}} + \delta \boldsymbol{p}.$$

while $\delta > 0$

After extracting the sparse optimal solution to (4.7), we can exclude points from the quadrature rule that correspond to a zero weight. If Q' is defined as the number of points in the optimally sparse quadrature rule, we can approximate an expectation of a quantity $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{v})$ as follows.

$$\int_{\mathbb{R}^m} \boldsymbol{y}(\boldsymbol{v})\, \varrho(\boldsymbol{v})\, d\boldsymbol{v} = \int_{\mathbb{R}^s} \boldsymbol{y}(\boldsymbol{v}(\boldsymbol{x}))\, \rho(\boldsymbol{x})\, d\boldsymbol{x} \approx \sum_{j=1}^{Q'} \boldsymbol{y}\left(\boldsymbol{v}\left(\boldsymbol{x}^{(j)}\right)\right) \hat{w}^{(j)}. \tag{4.8}$$

5. Realization for a multiphysics problem

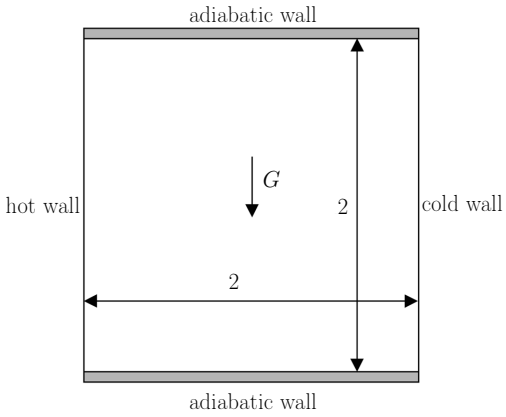

Figure 1: Schematic representation of 2D Thermally driven cavity flow.

With an eye on practical applications such as solar cavity modeling and building insulation, we consider the steady state interactions of temperature and laminar flow of an incompressible fluid in a 2D thermally driven square cavity $\boldsymbol{r} = [r_1; r_2] \in [-1, 1]^2$. We define $\mathcal{U} = \mathcal{U}(\boldsymbol{r})$, $\mathcal{V} = \mathcal{V}(\boldsymbol{r})$, $\mathcal{P} = \mathcal{P}(\boldsymbol{r})$ and $\mathcal{T} = \mathcal{T}(\boldsymbol{r})$ as the non-dimensionalized horizontal velocity, vertical velocity, pressure and temperature fields repsectively. As illustrated in Figure 1, the left wall is maintained at a higher temperature than the right wall, while the top and bottom walls are adiabatic. In this example, we model the conservation of mass and momentum in the first component and the conservation of energy in the second component.

5.1. Sources of uncertainty

We assume that the coefficient of thermal expansion is a random field with an exponential covariance kernal. Therefore, we have

$$\beta(\boldsymbol{r}, \boldsymbol{x}_1) = \beta_0 \left(1 + \sqrt{3}\delta_\beta \sum_{j=1}^{s_1} \gamma(\boldsymbol{r})\, \xi_j\right) : \; C_\beta(\boldsymbol{r}, \boldsymbol{r}') \propto \exp|\boldsymbol{r} - \boldsymbol{r}'|, \tag{5.1}$$

where $\boldsymbol{x}_1 = [\xi_1; \cdots; \xi_{s_1}]$ is a random vector in $[-1, 1]^{s_1}$. We use $s_1 = 6$ in this example.

Also, the left wall temperature is assumed to vary uniformly as follows.

$$\mathcal{T}(-1, r_2, \boldsymbol{x}_2) = \mathcal{T}_0 \left(1 + \sqrt{3}\delta_\mathcal{T} \eta\right), \tag{5.2}$$

where $\boldsymbol{x}_2 = [\eta]$ is a random scalar in $[-1, 1]$. Therefore, $s_2 = 1$. In this example, we set the coefficients of variation to $\delta_\beta = \delta_\mathcal{T} = 0.2$.

5.2. Strongly coupled model

Following the non-dimensionalization in [23], the conservation equations for the mass/momentum component are as follows.

$$\begin{aligned}
&\frac{\partial \mathcal{U}}{\partial r_1} + \frac{\partial \mathcal{V}}{\partial r_2} = 0, \\
&\frac{\partial (\mathcal{U}^2)}{\partial r_1} + \frac{\partial (\mathcal{V}\mathcal{U})}{\partial r_2} + \frac{\partial \mathcal{P}}{\partial r_1} - Pr\left(\frac{\partial^2 \mathcal{U}}{\partial r_1^2} + \frac{\partial^2 \mathcal{U}}{\partial r_2^2}\right) = 0, \\
&\frac{\partial (\mathcal{U}\mathcal{V})}{\partial r_1} + \frac{\partial (\mathcal{V}^2)}{\partial r_2} + \frac{\partial \mathcal{P}}{\partial r_2} - Pr\left(\frac{\partial^2 \mathcal{V}}{\partial r_1^2} + \frac{\partial^2 \mathcal{V}}{\partial r_2^2}\right) - PrRa\mathcal{T} = 0.
\end{aligned} \tag{5.3}$$

The conservation equation for energy component is as follows.

$$\frac{\partial (\mathcal{U}\mathcal{T})}{\partial r_1} + \frac{\partial (\mathcal{V}\mathcal{T})}{\partial r_2} - \left(\frac{\partial^2 \mathcal{T}}{\partial r_1^2} + \frac{\partial^2 \mathcal{T}}{\partial r_2^2}\right) = 0. \tag{5.4}$$

The non-dimensional quantities Pr and Ra are the Prandtl and Rayleigh numbers repsectively. The Rayleigh number is proportional to β and therefore, is

modeled as a random field with the same formulation as (5.1). We set $Pr = 0.7$ and $Ra_0 = 100$ in this example. The continuity equation for mass conservation in the first component is transformed into a pressure Poisson equation as follows.

$$\frac{\partial^2 \mathcal{P}}{\partial r_1^2} + \frac{\partial^2 \mathcal{P}}{\partial r_2^2} + \frac{\partial^2 \left(\mathcal{U}^2\right)}{\partial r_1^2} + 2\frac{\partial^2 \left(\mathcal{U}\mathcal{V}\right)}{\partial r_1 r_2} + \frac{\partial^2 \left(\mathcal{V}^2\right)}{\partial r_2^2} - Pr\frac{\partial \left(Ra\mathcal{T}\right)}{\partial r_2} = 0. \tag{5.5}$$

5.3. Weakly coupled model

The weakly coupled model is based on approximating the coupled temperature field in the first component and the coupled velocity field in the second component using surrogate polynomial functions. Therefore, we have the following equation for conservation of vertical momentum.

$$\frac{\partial \left(\mathcal{U}\mathcal{V}\right)}{\partial r_1} + \frac{\partial \left(\mathcal{V}^2\right)}{\partial r_2} + \frac{\partial \mathcal{P}}{\partial r_2} - Pr\left(\frac{\partial^2 \mathcal{V}}{\partial r_1^2} + \frac{\partial^2 \mathcal{V}}{\partial r_2^2}\right) - PrRa\tilde{\mathcal{T}} = 0, \tag{5.6}$$

where $\tilde{\mathcal{T}} = \frac{1}{2}\left(1 - r_1\right)\left(\left(1 - r_2\right)\mathcal{T}\left(0, -1\right) + \left(1 + r_2\right)\mathcal{T}\left(0, 1\right)\right)$ is a bilinear approximation of the coupled temperature field. Similarly, we have the following equation for conservation of energy.

$$\frac{\partial \left(\hat{\mathcal{U}}\mathcal{T}\right)}{\partial r_1} + \frac{\partial \left(\hat{\mathcal{V}}\mathcal{T}\right)}{\partial r_2} - \left(\frac{\partial^2 \mathcal{T}}{\partial r_1^2} + \frac{\partial^2 \mathcal{T}}{\partial r_2^2}\right) = 0, \tag{5.7}$$

where $\hat{\mathcal{U}} = \left(1 - r_1^2\right)\left(r_2 - r_2^3\right)\frac{\partial \mathcal{U}}{\partial r_2}\left(0, 0\right)$ and $\hat{\mathcal{V}} = \left(r_1 - r_1^3\right)\left(1 - r_2^2\right)\frac{\partial \mathcal{V}}{\partial r_1}\left(0, 0\right)$. Therefore, we have $m_1 = m_2 = 2$.

5.4. Spatial discretization

We implemented a Finite-Volume method [24] on a uniform 20×20 mesh. Therefore, we have $n_1 = 1200$ and $n_2 = 400$. The gradient, divergence and Laplacian operators were all discretized using linear second order schemes. In our implementation, we used a convergence tolerance of $\epsilon = 10^{-8}$ with which the iterations converged within a maximum of 5 iterations.

5.5. Verification of NISP implementation

We first implemented the NISP method in `MATLAB` on a single core machine (3.1 GHz Intel i5 CPU, 4GB DDR3 RAM) without dimension reduction for successively increasing total polynomial degree p. Table 1 indicates the size of the gPC basis and quadrature rules that were required. In this example, latter corresponded to the Clenshaw-Curtis quadrature rules. A benchmark set of converged solutions was generated at N random points $\left\{\hat{\boldsymbol{x}}_1^{(j)}, \hat{\boldsymbol{x}}_2^{(j)}\right\}_{j=1}^{N}$ using Monte-Carlo sampling. As per the notation in the model problem (2.1), we define a pointwise error indicator as follows.

$$\mathcal{E}_i^p = \max_{1 \le j \le N} \| \boldsymbol{u}_i\left(\hat{\boldsymbol{x}}_1{}^{(j)}, \hat{\boldsymbol{x}}_2{}^{(j)}\right) - \hat{\boldsymbol{u}}_i^p\left(\hat{\boldsymbol{x}}_1{}^{(j)}, \hat{\boldsymbol{x}}_2{}^{(j)}\right) \|_{\mathcal{L}^\infty} : i \in \{1, 2\}.$$

The convergence plots are shown in Figure 2.

p	P	Q
0	1	1
1	8	39
2	36	425
3	120	3501

Table 1: Cardinality of gPC basis and quadrature rules for NISP implementation.

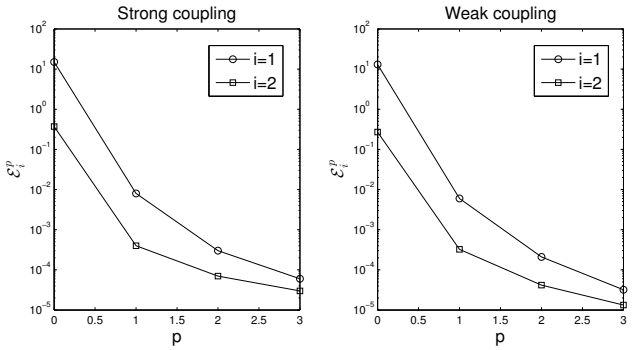

Figure 2: Pointwise error of solutions $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ versus polynomial degree p.

5.6. Mean , standard deviation and sensitivities

For the strong coupling case, the mean and standard deviation of the quantities are shown in Figure 3 and Figure 4. For the weak coupling case, the mean and standard deviation of the quantities are shown in Figure 3 and Figure 4. The sensitivity indices are computed using the ANOVA method [25] and reported in Table 2. A total polynomial degree of $p = 3$ is used here.

Quantity	Strong			Weak		
	S_1 (%)	S_{12} (%)	S_2 (%)	S_1 (%)	S_{12} (%)	S_2 (%)
$\mathcal{U}$	56.06	1.73	42.21	52.35	2.10	45.55
$\mathcal{V}$	58.04	1.72	40.24	56.07	2.21	41.72
$\mathcal{P}$	34.26	1.42	64.32	32.28	1.34	66.38
$\mathcal{T}$	1.26	0.14	98.60	0.11	0.02	99.87

Table 2: ANOVA based sensitivity indices. $S_1$and S_2 correspond to the main effects of $\boldsymbol{x}_1$and $\boldsymbol{x}_2$. S_{12} corresponds to the interactions.

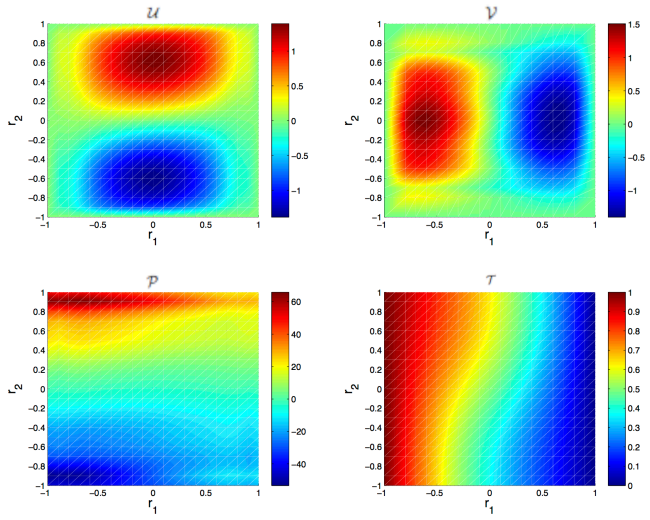

Figure 3: Mean of quantities (strong coupling)

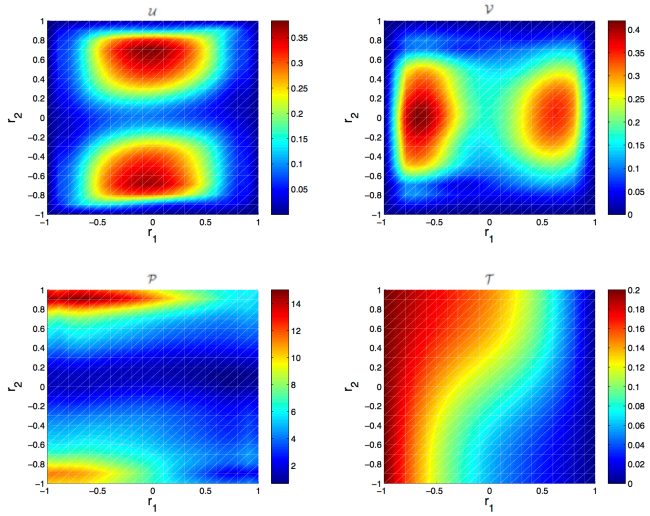

Figure 4: Standard deviation of quantities (strong coupling)

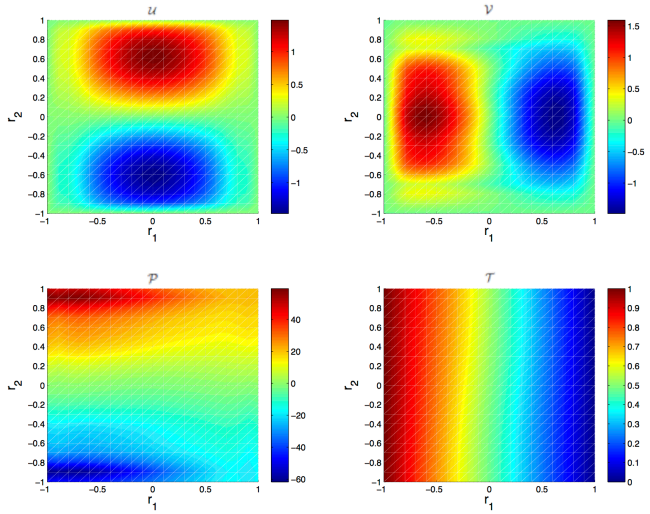

Figure 5: Mean of quantities (weak coupling)

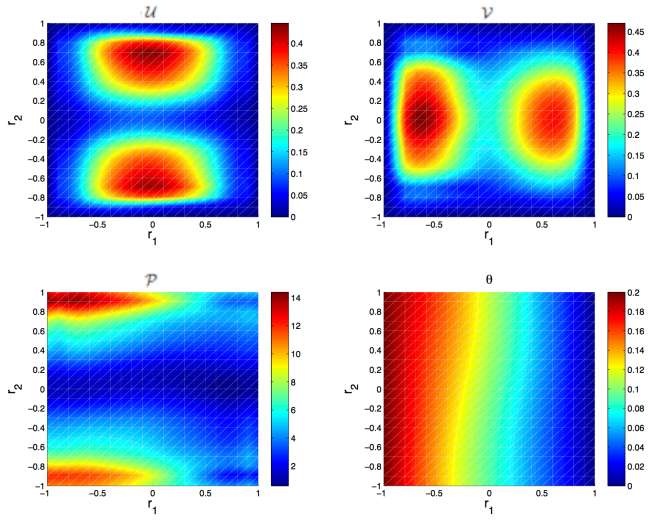

Figure 6: Standard deviation of quantities (weak coupling)

5.7. NISP implementation with dimension reduction

We then implemented the NISP method with dimension reduction in the strongly coupled model using $p' = p = 3$ and various values of the tolerance ε. The reduction was limited to the energy component of the solver. To observe the convergence as ε decreases, the $\mathcal{L}^\infty$ distance between the mean and variance of the converged solutions with and without dimension reduction were compared and are reported in Table 3.

ε	d_2	$\|\mathbb{E}\left(\hat{\boldsymbol{y}}_2^{3,d_2}\right) - \mathbb{E}\left(\hat{\boldsymbol{y}}_2^{3}\right)\|_{\mathcal{L}^\infty}$	$\|\mathrm{Var}\left(\hat{\boldsymbol{y}}_2^{3,d_2}\right) - \mathrm{Var}\left(\hat{\boldsymbol{y}}_2^{3}\right)\|_{\mathcal{L}^\infty}$
0.004	1	3.4×10^{-4}	6.6×10^{-3}
0.002	2	8.1×10^{-5}	2.3×10^{-3}
0.001	3	4.7×10^{-5}	7.3×10^{-4}

Table 3: $\mathcal{L}^\infty$ distance between mean and variance approximations with and without dimension reduction in energy component.

In the weakly coupled model with $p' = p = 3$, we compared the mean and variance approximations obtained using the global and reduced dimensional composite gPC representations of the solutions. We observed that the $\mathcal{L}^\infty$ distances between the approximated mean was 7.2×10^{-4} for $\boldsymbol{y}_1$ and 8.7×10^{-5} for $\boldsymbol{y}_2$. The same distances were computed for the variance approximations and observed to be 2.6×10^{-2} for $\boldsymbol{y}_1$ and 3.8×10^{-3} for $\boldsymbol{y}_2$.

5.7.1. Computational gains

In the strongly coupled model, the expected gain (limited to the energy component) in computational cost can be estimated as Q_1/Q_1'. However, due to the required SVD and LP computations, we observed a lower value for the gain $t_{\mathrm{CPU}}/t'_{\mathrm{CPU}}$ based on the actual run time. The results are reported in Table 4.

ε	d_2	P'	Q_1'	Expected gain (Q_1/Q_1')	Actual gain ($t_{\mathrm{CPU}}/t'_{\mathrm{CPU}}$)
0.004	1	10	9	43.2	26.5
0.002	2	20	28	13.9	7.1
0.001	3	35	68	5.7	2.9

Table 4: Computational gains observed in the energy component with $p' = p = 3$. The original values of P and Q_1 are 120 and 389 respectively.

In the weakly coupled model, the number of quadrature points needed was $Q' = 587$ and therefore, the expected gain is $Q_1 Q_2/Q' = 6.0$. However, based on the CPU run time, we observed a gain of 3.1.

6. Summary and Conclusions

We reviewed state-of-the-art methods of dimension and model reduction for efficient gPC based uncertainty propagation in multiphysics systems. The methods described facilitated an efficient construction of reduced dimensional gPC basis and quadrature rules using simple and widely used linear algebra tools. By demonstrating their implementations on a 2D thermally driven cavity flow example, we observed a significant reduction in dimension for the strongly coupled model, as the effects of 7 random variables could be captured (within 0.5% tolerance) using just 2 random variables in the energy component. However, the dimension reduction is limited in scope, and we are actively exploring ways of extending it to be suitable for bidirectional implementation. A promising approach for reducing the curse of dimensionality has been proposed for coupled domain problems in [26].

In the weakly coupled paradigm, the composite gPC approach also provided a reasonable amount of computational savings. Although it falls outside the context of dimension and model reduction, a method of transformatng a bidirectionally coupled system into a network of unidirectionally coupled components, which can overcome the need for costly iterations, has been proposed in [27]. Extensions and improvements of such methods are also an important topic of our ongoing research.

In the context of gPC based uncertainty propagation, we note that several undeveloped and unexplored areas of improving its computational performance still remain. These include methods of incorporating derivatives, known as active subspace methods [28], and reducing the errors associated with long time integration in unsteady models [29]. Investigating these topics is on our agenda for the near future.

[1] Kullback, S. (2012). *Information theory and statistics.* Courier Dover Publications.

[2] Aitkin, M. A. (2010). *Statistical inference: An integrated Bayesian/likelihood approach* (p. 236). Chapman & Hall/CRC.

[3] Robert, C. P., & Casella, G. (2004). *Monte Carlo statistical methods* (Vol. 319). New York: Springer.

[4] Le Maître, O. P., & Knio, O. M. (2010). *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics.* Springer.

[5] Xiu, D., & Karniadakis, G. E. (2003). *Modeling uncertainty in flow simulations via generalized polynomial chaos.* Journal of Computational Physics, 187(1), 137-167.

[6] Babuska, I., Tempone, R., & Zouraris, G. E. (2004). *Galerkin finite element approximations of stochastic elliptic partial differential equations.* SIAM Journal on Numerical Analysis, 42(2), 800-825.

[7] Reagana, M. T., Najm, H. N., Ghanem, R. G., & Knio, O. M. (2003). *Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection.* Combustion and Flame, 132(3), 545-555.

[8] Babuška, I., Nobile, F., & Tempone, R. (2010). *A stochastic collocation method for elliptic partial differential equations with random input data.* SIAM review, 52(2), 317-355.

[9] Felippa, C. A., Park, K. C., & Farhat, C. (2001). *Partitioned analysis of coupled mechanical systems.* Computer methods in applied mechanics and engineering, 190(24), 3247-3270.

[10] Matthies, H. G., Niekamp, R., & Steindorf, J. (2006). *Algorithms for strong coupling procedures.* Computer Methods in Applied Mechanics and Engineering, 195(17), 2028-2049.

[11] Constantine, P. G., Doostan, A., & Iaccarino, G. (2009). *A hybrid collocation/Galerkin scheme for convective heat transfer problems with stochastic boundary conditions.* International journal for numerical methods in engineering, 80(6-7), 868-880.

[12] Chen, X., Ng, B., Sun, Y., & Tong, C. (2013). *A flexible uncertainty quantification method for linearly coupled multi-physics systems.* Journal of Computational Physics, 248, 383-401.

[13] Lanzkron, P. J., Rose, D. J., & Wilkes, J. T. (1996). *An analysis of approximate nonlinear elimination.* SIAM Journal on Scientific Computing, 17(2), 538-559.

[14] Ernst, O. G., Mugler, A., Starkloff, H. J., & Ullmann, E. (2012). *On the convergence of generalized polynomial chaos expansions.* ESAIM Math. Model. Numer. Anal, 46(2), 317-339.

[15] Soize, C., & Ghanem, R. G. (2009). *Reduced chaos decomposition with random coefficients of vector-valued random variables and random fields.* Computer Methods in Applied Mechanics and Engineering, 198(21), 1926-1934.

[16] Oja, E., & Karhunen, J. (1985). *On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix.* Journal of mathematical analysis and applications, 106(1), 69-84.

[17] Zhang, D., & Lu, Z. (2004). *An efficient, high-order perturbation approach for flow in random porous media via Karhunen–Loeve and polynomial expansions.* Journal of Computational Physics, 194(2), 773-794.

[18] Courant, R., & Hilbert, D. (1966). *Methods of mathematical physics* (Vol. 1). CUP Archive.

[19] Arnst, M., Ghanem, R., Phipps, E., & Red-Horse, J. (2012). *Dimension reduction in stochastic modeling of coupled problems*. International Journal for Numerical Methods in Engineering, 92(11), 940-968.

[20] Constantine, P. G., & Phipps, E. T. (2012). *A Lanczos method for approximating composite functions*. Applied Mathematics and Computation, 218(24), 11751-11762.

[21] Wright, S. J., & Nocedal, J. (1999). *Numerical optimization* (Vol. 2). New York: Springer.

[22] Putinar, M. (1997). *A note on Tchakaloff's theorem*. Proceedings of the American Mathematical Society, 125(8), 2409-2414.

[23] Rubio, O., Bravo, E., & Claeyssen, J. R. (2002). *Thermally driven cavity flow with Neumann condition for the pressure*. Applied numerical mathematics, 40(1), 327-336.

[24] LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems* (Vol. 31). Cambridge university press.

[25] Saltelli, A., Chan, K., & Scott, E. M. (Eds.). (2000). *Sensitivity analysis* (Vol. 134). New York: Wiley.

[26] Hadigol, M., Doostan, A., Matthies, H. G., & Niekamp, R. (2013). *Partitioned treatment of uncertainty in coupled domain problems: A separated representation approach*. arXiv preprint arXiv:1305.6818.

[27] Sankararaman, S., & Mahadevan, S. (2012). *Likelihood-based Approach for Uncertainty Propagation in Multidisciplinary Analysis*. Analysis 1(g2), u12.

[28] Constantine, P. G., Dow, E., & Wang, Q. (2013). *Active subspace methods in theory and practice: applications to kriging surfaces*. arXiv preprint arXiv:1304.2070.

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