

UNCECOMP 2015  
1<sup>st</sup> ECCOMAS Thematic Conference on  
Uncertainty Quantification in Computational Sciences and Engineering  
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)  
Crete Island, Greece, 25–27 May 2015

## STOCHASTIC COLLOCATION FOR CORRELATED INPUTS

Maria Navarro<sup>1</sup>, Jeroen Witteveen<sup>2</sup>, and Joke Blom<sup>2</sup>

<sup>1</sup>CEMSE Division, King Abdullah University of Science and Technology (KAUST),  
Thuwal, Kingdom of Saudi Arabia  
e-mail: mariaisabel.navarrojimenez@kaust.edu.sa

<sup>2</sup>Center for Mathematics and Computer Science (CWI),  
Science Park, 123, 1098 XG, Amsterdam (the Netherlands)  
e-mail: {jeroen.witteveen, joke.blom}@cwi.nl

**Keywords:** Gaussian quadrature, Correlated inputs, Uncertainty Quantification.

**Abstract.** *Stochastic Collocation (SC) has been studied and used in different disciplines for Uncertainty Quantification (UQ). The method consists of computing a set of appropriate points, called collocation points, and then using Lagrange interpolation to construct the probability density function (pdf) of the quantity of interest (QoI). The collocation points are usually chosen as Gauss quadrature points, i.e., the roots of orthogonal polynomials with respect to the pdf of the uncertain inputs. If the mathematical model has more than one stochastic parameter, the multidimensional set of points is usually build using the tensor product of the roots of the one-dimensional orthogonal polynomials. As a result of that, for multidimensional problems the same set of collocation points is used for both correlated and uncorrelated inputs. In this work, we propose to compute an alternative set of points for correlated inputs. The set will be derived using the orthogonal polynomials for correlated inputs that we developed in a previous work. As these polynomials are not unique, we will obtain multiple sets of collocations points for each input pdf. The aim of this paper is to study the differences between those sets of points and to find and optimal one.*

## 1 INTRODUCTION

Uncertainty quantification can be seen as the theory which tries to determine how likely certain outcomes of a model are. Therefore, the ability to quantify the uncertainty in systems and models with a strong “real-world” setting is fundamental to many inference problems in science and engineering. Proof of this is the large number of applications in different fields that can be found in the literature: combustion chemistry [22], neuroscience [21], fluid dynamics [14], ocean wave modeling [26], meteorology [16], to state just a few.

Several methods exist to quantify the uncertainties. The simplest approach is Monte Carlo simulation [1], where - given a probability density function (pdf) for the random inputs - the mean and other characteristics of the output distribution can be estimated by sampling repeatedly from the pdf and simulating the model for each sample. This method is often too expensive, since a lot of samples are required for a reasonable accuracy. Therefore, to decrease the computational effort, several modifications leading to new methods were introduced: Latin hypercube sampling (LHS) [9], the Quasi-Monte Carlo (QMC) method [5], the Markov Chain Monte Carlo method (MCMC) [12], etc. An alternative method is the Polynomial Chaos Expansion (PCE) method, which is based on Wiener’s homogeneous chaos theory [23]. In this method to find the polynomial coefficients either Galerkin projection or Spectral projection is used, which results in a coupled system of deterministic equations. Since the first version of this method for Gaussian random variables appeared, until now several generalizations and improvements of the PCE technique have been published: [2, 3, 20, 18], etc. So nowadays, due to increased relevance of UQ during the last years, a wide variety of techniques to compute the uncertainties in the model have been applied, for instance: Karhunen-Loève decomposition [11], gradient-based methods [24], sparse grids [7], perturbation methods [6] based on local Taylor series expansions, Bayesian methods [10] etc.

In this paper we consider the Stochastic Collocation method to compute the uncertainties. It is a nonintrusive method, which was developed by Mathelin and Hussaini [13]. The method consists of computing a set of appropriate points, called collocation points, and then to approximate the probability distribution function of the QoI by using Lagrange interpolation [25]. The standard collocation points are usually chosen as the roots of the orthogonal polynomials with respect to the marginals pdf’s of the uncertain inputs. Thus, the same points are used for both correlated and uncorrelated inputs. On account of this, the following question is raised: will there be a more suitable set of points for correlated inputs? In order to answer this question, we propose to compute an alternative set of points using the new orthogonal polynomials for correlated inputs that we developed in a previous study [15] and then to compare the obtained results with both sets of points. The new polynomials are not unique in the multidimensional case. As a result of that, we will have multiple sets of points for each random input. The aim of this paper is therefore, to perform a comparison study of different sets of points. Specifically, we will see that one of them can be seen as the optimal set of points, in the sense that in the limit of full and no correlation the collocation points are equal to the known optimal ones. We will also prove that all sets of points achieve the same integration degree of exactness. However, some differences are found in terms of integration error and interpolation error; the set with the smallest error of integration is the set with the highest error of interpolation and vice versa.

The outline of this paper is as follows: Section 2 reviews the Stochastic Collocation method. Section 3 introduces new sets of collocation points for the case of correlated inputs. In Section 4, we perform a comparison study between the new sets of collocation points in terms of integration degree of exactness, integration error, and interpolation error. The conclusions are

drawn in Section 5.

## 2 STOCHASTIC COLLOCATION

Consider the following stochastic equation in the probability space  $(\Omega, \mathcal{A}, P)$  where  $\Omega$  is the event space,  $\mathcal{A} \subseteq 2^\Omega$  its  $\sigma$ -algebra and  $P$  its probability measure

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\xi}(\boldsymbol{\omega}); \mathbf{u}) = \mathbf{f}(\mathbf{x}, \boldsymbol{\xi}(\boldsymbol{\omega})), \quad \mathbf{x} \in \mathbf{X} \text{ and } \boldsymbol{\omega} \in \Omega, \quad (1)$$

where  $\mathbf{u}(\mathbf{x}, \boldsymbol{\xi}(\boldsymbol{\omega}))$  is the stochastic solution vector,  $\mathcal{L}$  is an operator, and  $\mathbf{f}$  a source function;  $\mathbf{x}$  is the vector of *deterministic* input variables describing, e.g., time or space, and  $\boldsymbol{\xi}(\boldsymbol{\omega})$  is the  $n$ -dimensional vector of *random* input variables, with joint probability density function  $\rho(\boldsymbol{\xi}(\boldsymbol{\omega}))$ ;  $\boldsymbol{\xi}$  typically contains the uncertainties in model parameters or initial and boundary conditions. In order to make the notation less cumbersome, we denote the realization of a random vector  $\boldsymbol{\xi}(\boldsymbol{\omega})$ , for  $\boldsymbol{\omega} \in \Omega$ , by  $\boldsymbol{\xi} \in \Xi$ , with  $\Xi$  the support of the pdf.

In the Stochastic Collocation method the solution  $\mathbf{u}$  is approximated by the following expansion in case of one uncertain input parameter  $\xi$ :

$$\mathbf{u}(\mathbf{x}, \xi) = \sum_{i=1}^{p+1} \mathbf{u}_i(\mathbf{x}) h_i(\xi), \quad (2)$$

with  $\mathbf{u}_i(\mathbf{x})$  the solution values  $\mathbf{u}(\mathbf{x}, \xi)$  of Equation (1) at the one-dimensional collocation points  $\alpha_i$ . These unidimensional collocation points are computed as the zeros of the orthogonal polynomials to the pdf. The  $h_i(\xi)$  are the unidimensional Lagrange interpolating polynomials of degree  $p$  with  $h_i(\alpha_j) = \delta_{ij}$ . The Lagrange interpolating polynomial that passes through the collocation points is given by:

$$h_i(\xi) = \prod_{\substack{j=1 \\ j \neq i}}^{p+1} \frac{\xi - \alpha_j}{\alpha_i - \alpha_j}. \quad (3)$$

One-dimensional Stochastic Collocation can be extended to  $n$ -dimensional stochastic input vectors  $\boldsymbol{\xi} = \{\xi^1, \dots, \xi^n\}$  using a tensor product. This results in  $N$   $n$ -dimensional collocation points with  $N = (p+1)^n$ .

### 2.1 Statistic

The mean and the variance of the stochastic solution can be determined using:

$$\boldsymbol{\mu}_u = \sum_{i=1}^N \mathbf{u}_i(\mathbf{x}) \omega_i, \quad (4)$$

$$\boldsymbol{\sigma}_u^2 = \sum_{i=1}^N (\mathbf{u}_i(\mathbf{x}))^2 \omega_i - \boldsymbol{\mu}_u^2, \quad (5)$$

where the  $\omega_i$  are the weights corresponding to the collocation points  $\alpha_i$ .

## 3 COLLOCATION POINTS FOR CORRELATED INPUTS

To find the collocation points the procedure below is followed. To avoid any misunderstanding between the sets of points, we will use the name  *$n$ -dimensional standard collocation points* for the ones that come from the tensor product approach; and  *$n$ -dimensional collocation points for correlated inputs* to name the points computed using the new orthogonal polynomials.

### 3.1 $n$ -dimensional standard collocation points

The computation of the  $n$ -dimensional standard collocation points is based on the computation of the unidimensional ones, which are the Gaussian quadrature points. These points are the roots of the orthogonal polynomials with respect to a given weight function. In our situation, the weight function will be given by the marginal pdf of the uncertain input. Once the unidimensional points are computed, the  $n$ -dimensional points are built based on the tensor product approach. For a simple example, we assume a model with two uncertain inputs,  $(\xi_1, \xi_2)$ , that follow a normal distribution where the mean  $\boldsymbol{\mu}$  and the covariance matrix  $\boldsymbol{\Sigma}$  are fixed as:

$$\boldsymbol{\mu} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.8 \\ 0.8 & 1 \end{pmatrix}. \quad (6)$$

It is also assumed that only four collocation points are necessary to approximate the solution. As the uncertain inputs are Gaussian distributed, we compute -for each uncertain input- the roots of the one dimensional Hermite orthogonal polynomials of degree 2:  $\Phi(\xi_1) = \xi_1^2 - 1$  and  $\Phi(\xi_2) = \xi_2^2 - 1$ . Finally the set of  $n$ -dimensional standard collocation points is built using the tensor product of the roots  $\pm 1$ . Figure 1 shows the final set of points.

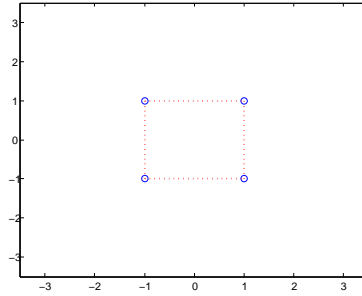


Figure 1:  $n$ -dimensional standard collocation points, which usually are used for both correlated and uncorrelated inputs.

### 3.2 $n$ -dimensional collocation points for correlated inputs

The  $n$ -dimensional collocation points for correlated inputs are calculated based on the multidimensional orthogonal polynomials for correlated inputs [15]. In this section we present both how the polynomials are built by using Gram-Schmidt orthogonalization and how the new points are derived from them.

#### 3.2.1 Computation of the orthogonal polynomials

For Gram-Schmidt the first step is the choice of a suitable set of linearly independent polynomials. It should be noticed that any linearly independent set of polynomials can be used in this method, but for simplicity we will use the set of monic polynomials  $\{e_j(\boldsymbol{\xi})\}_{j=0}^M$ . For example, if  $\dim(\boldsymbol{\xi}) = n = 2$  and the maximum degree  $m = 2$ , then  $M + 1 = \frac{(n+m)!}{n!m!} = 6$ , and the set of linearly independent polynomials  $\{e_j(\xi_1, \xi_2)\}_{j=0}^5$  equals  $\{1, \xi_1, \xi_2, \xi_1^2, \xi_1\xi_2, \xi_2^2\}$ . Next, the orthogonal polynomial basis  $\{\Phi_j(\boldsymbol{\xi})\}_{j=0}^M$  is constructed sequentially from  $\{e_j(\boldsymbol{\xi})\}_{j=0}^M$  using the

Gram Schmidt algorithm

$$\begin{aligned}\Phi_0(\boldsymbol{\xi}) &= 1, \\ \Phi_j(\boldsymbol{\xi}) &= e_j(\boldsymbol{\xi}) - \sum_{k=0}^{j-1} c_{jk} \Phi_k(\boldsymbol{\xi}) \quad \text{for } 1 \leq j \leq M,\end{aligned}\tag{7}$$

where the coefficients  $c_{jk}$  are given by

$$c_{jk} = \frac{\langle e_j(\boldsymbol{\xi}), \Phi_k(\boldsymbol{\xi}) \rangle}{\langle \Phi_k(\boldsymbol{\xi}), \Phi_k(\boldsymbol{\xi}) \rangle}\tag{8}$$

and the innerproduct is taken with respect to the pdf  $\rho(\boldsymbol{\xi})$ . Note, that in the multidimensional case the basis is not unique, it is dependent on the order of the set of polynomials  $\{e_j\}$  of a specific degree.

### 3.2.2 Computation of the new collocation points

The new set of collocation points is computed as the solution of the system of equations given by the new orthogonal polynomials of a specific degree  $m$ . Consider again the example in Section 3.1. We first generate the multidimensional orthogonal polynomials of degree 2 with respect to the normal distribution given by (6). To do that, the following set of linearly independent polynomials of degree 2 is used

$$\{\xi_1^2, \xi_1 \xi_2, \xi_2^2\}.\tag{9}$$

In general, the dimension of this sets  $\tilde{M}_m = \frac{(n+m-1)!}{(n-1)!m!}$ . For simplicity monic polynomials are used. Then the polynomials are build sequentially from (9) by applying Gram-Schmidt orthogonalization. In this case the following polynomials of degree 2 are obtained:

$$\begin{aligned}\Phi_1(\xi_1, \xi_2) &= \xi_1^2 - 1, \\ \Phi_2(\xi_1, \xi_2) &= \xi_1 \xi_2 - 0.8 \xi_1^2, \\ \Phi_3(\xi_1, \xi_2) &= \xi_2^2 + 0.64 \xi_1^2 - 1.6 \xi_1 \xi_2 - 0.36.\end{aligned}\tag{10}$$

If the order of (9) is changed, different polynomials will be obtained and therefore different sets of collocation points will also be derived. Specifically, we will have as many different sets of collocation points for correlated inputs as orders in (9). For the example  $6 = 3!$  different sets of points are obtained, that is, the total number of combinations in (9). Figure 2 shows all possible sets of points for this example. Appendix A shows all sets of points for polynomials of degree 3.

**Remark 3.1** Note that we are extending the tensor product approach, so, the points are computed by solving the system given by the polynomials that come from the monic polynomials of one variable,  $M_m$ . In this example the points are computed by solving the system given by  $\Phi_1$  and  $\Phi_3$  ( $M_2 = 2$ ). As we can see in Figure 2 some sets of nodes are equal, so from now on we will just consider the sets of points given by: order  $a$ , order  $b$ , and order  $e$ .

## 4 DIFFERENCES BETWEEN SET OF NODES

In this section, the differences between sets of points will be studied. In particular, we will try to find differences in: integration degree of exactness (DOE), integration error, and interpolation error.

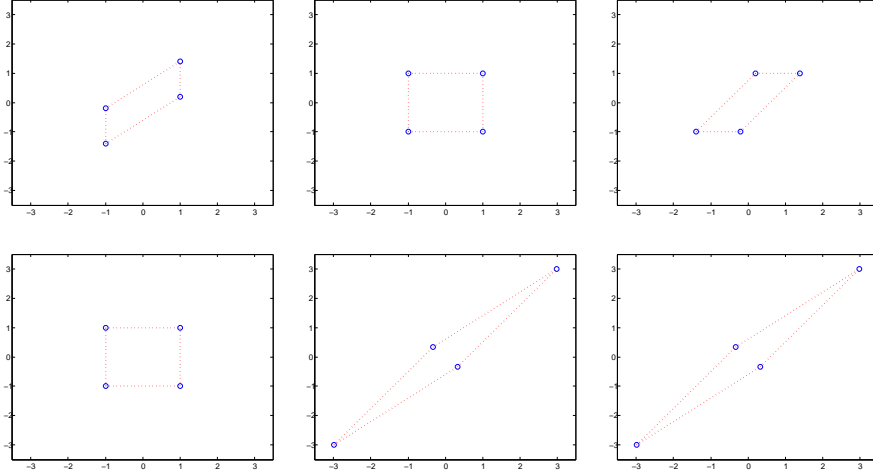


Figure 2: Plots of the nodes with order  $a$ :  $\{\xi_1^2, \xi_1\xi_2, \xi_2^2\}$ , order  $b$ :  $\{\xi_1^2, \xi_2^2, \xi_1\xi_2, \xi_1^2, \xi_2^2\}$ , order  $c$ :  $\{\xi_2^2, \xi_1\xi_2, \xi_1^2\}$ , order  $d$ :  $\{\xi_2^2, \xi_2^1, \xi_1\xi_2\}$  order  $e$ :  $\{\xi_1\xi_2, \xi_1^2, \xi_2^2\}$  and order  $f$ :  $\{\xi_1\xi_2, \xi_2^2, \xi_1^2\}$ .

#### 4.1 Integration degree of exactness

To check which set of points presents the highest degree of exactness, we will define first the concept of Gaussian quadrature rules based on the tensor product which is an extension of the unidimensional Gaussian quadrature rules. Also Theorem 4.2 and its proof follow the well-known proof of the Gaussian integration method (see, e.g., [19]).

**Definition 4.1** ( $n$ -dimensional tensor product quadrature rule). *Given  $[a_1, b_1] \times \dots \times [a_n, b_n] \subseteq \mathbb{R}^n$  such that  $a_i < b_i \forall i \in \{1, \dots, n\}$ , a weight function  $\rho : [a_1, b_1] \times \dots \times [a_n, b_n] \rightarrow \mathbb{R}_0^+$ , and  $N \in \mathbb{N}$ , let  $\alpha_1, \dots, \alpha_N \in [a_1, b_1] \times \dots \times [a_n, b_n]$  be the zeros of the system of equations given by the set of polynomials  $\Phi_m^1, \dots, \Phi_m^{M_m}$ , where  $\Phi_m^i$  is the  $i$ th orthogonal polynomial of degree  $m$  with respect to the inner product  $\langle \cdot, \cdot \rangle_\rho$  and let  $\mathbf{h}_1, \dots, \mathbf{h}_N$  be the corresponding multivariate Lagrange basis polynomials [17], then the quadrature rule*

$$\mathbf{I}_m^n : \mathcal{C}^n([a_1, b_1] \times \dots \times [a_n, b_n]) \rightarrow \mathbb{R}, \quad \mathbf{I}_m^n(\mathbf{f}) := \sum_{j=1}^N \omega_j \mathbf{f}(\alpha_j), \quad (11)$$

where

$$\omega_j := \langle \mathbf{h}_j, \mathbf{1} \rangle_\rho = \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{h}_j(\xi_1, \dots, \xi_n) \rho(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n \quad (12)$$

for each  $j \in \{1, \dots, N\}$  is called the  $m$ th order  $n$ -dimensional tensor product of the unidimensional Gaussian quadrature rule with respect to  $\rho$ .

**Theorem 4.2.** *Consider the situation of Definition 4.1. In particular, let  $\mathbf{I}_m^n$  be an  $m$ th order  $n$ -dimensional tensor product quadrature rule with respect to  $\rho$ , then  $\mathbf{I}_m^n$  is exact for each polynomial of degree  $d = 2m - 1$  or less, that means:*

$$\langle \mathbf{p}, \mathbf{1} \rangle_\rho = \sum_{j=1}^N \omega_j \mathbf{p}(\alpha_j) \quad \text{for each } \mathbf{p} \in \mathcal{P}_{2m-1}[\xi_1, \dots, \xi_n] \quad (13)$$

we call  $d$  the degree of exactness of the quadrature rule.

*Proof.* Let  $\mathbf{p}$  be a polynomial of degree  $2m - 1$  (or lower) in  $n$  variables, that means,  $\mathbf{p} \in \mathcal{P}_{2m-1}[\xi_1, \dots, \xi_n]$ . By construction (cf. Remark 4.1) the set of polynomials  $\{\Phi_m^i\}_{i=1}^n$  used to compute the collocation points are triangular polynomials (see Theorem 5.3, Appendix B). Applying the iterated pseudodivision, we can write  $\mathbf{p}$  as:

$$b_n^{\delta_n} \dots b_1^{\delta_1} \mathbf{p} = \mathbf{q}_1 \cdot \Phi_m^n + \dots + \mathbf{q}_n \cdot \Phi_m^1 + \mathbf{r} \quad (14)$$

Then the relation  $\Phi_m^i(\boldsymbol{\alpha}_j) = 0$ , with  $i = \{1, \dots, n\}$  and  $j = \{1, \dots, N\}$ , implies:

$$\mathbf{p}(\boldsymbol{\alpha}_j) = \mathbf{r}(\boldsymbol{\alpha}_j) \text{ for each } j \in \{1, \dots, N\} \quad (15)$$

Applying Proposition 5.5 (see Appendix B) gives  $N \geq Tnt(\mathbf{r})$ , so we have enough points to build the Lagrange multivariate interpolation polynomial of the final reminder  $\mathbf{r}$ :

$$\mathbf{r}(\xi_1, \dots, \xi_n) = \sum_{j=1}^N \mathbf{r}(\boldsymbol{\alpha}_j) \mathbf{r}_j(\xi_1, \dots, \xi_n). \quad (16)$$

This allows to compute the integral of  $\mathbf{p}$  as follows:

$$\begin{aligned} & \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{p}(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n = \\ &= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{q}_1(\xi_1, \dots, \xi_n) \Phi_m^n(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n + \dots + \\ &+ \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{q}_n(\xi_1, \dots, \xi_n) \Phi_m^1(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n + \dots + \\ &+ \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{r}(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n = \\ &= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{r}(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n \end{aligned}$$

because of the orthogonality property and

$$\begin{aligned} &= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{r}(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n = \\ &= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \sum_{j=1}^N \mathbf{r}(\boldsymbol{\alpha}_j) \mathbf{h}_j(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n = \\ &= \sum_{j=1}^N \mathbf{r}(\boldsymbol{\alpha}_j) \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \mathbf{h}_j(\xi_1, \dots, \xi_n) \boldsymbol{\rho}(\xi_1, \dots, \xi_n) d\xi_1 \dots d\xi_n = \\ &= \sum_{j=1}^N \omega_j \mathbf{r}(\boldsymbol{\alpha}_j) = \sum_{j=1}^N \omega_j \mathbf{p}(\boldsymbol{\alpha}_j). \end{aligned}$$

## 4.2 Integration error

We have proved in the previous section that there is no difference between the sets in terms of integration DOE. Let us study now the integration error, EI, the error in approximating the integral by the quadrature rule. In particular, we will study  $EI_d$ , that is, the error associated to the first non exact integral whose degree of exactness is  $d$ , it is given by:

$$EI_d(\mathbf{f}) = \int_{\Omega} \mathbf{f}(\boldsymbol{\xi})\boldsymbol{\rho}(\boldsymbol{\xi})d\boldsymbol{\xi} - \mathbf{I}_d^n(\mathbf{f}). \quad (17)$$

### 4.2.1 2 dimensions integration error comparison

In order to compare  $EI_d$  between different sets of nodes, we follow this procedure. First, the integral of a general polynomial is computed; second, that integral is approximated by quadrature rules using different sets of nodes; and finally the error of integration is computed by Equation (17). Let us assume we are under the hypothesis of the example in Section 3, therefore we have computed the set collocation points for degree of exactness  $d = 3$  ( $d = 2m - 1$  with  $m = 2$ ). Consider the following polynomial of degree 4 (the next order after exactness).

$$\begin{aligned} f(x, y) = & a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2 + a_7x^3 + a_8x^2y + a_9xy^2 + a_{10}y^3 + \\ & + a_{11}x^4 + a_{12}x^3y + a_{13}x^2y^2 + a_{14}xy^3 + a_{15}y^4. \end{aligned}$$

After computing the errors with Equation (17), we obtain these results:

$$\begin{aligned} EI_3^a(f) &= 2a_{11} + 1.6a_{12} + 1.28a_{13} + 1.024a_{14} + 1.0784a_{15}, \\ EI_3^b(f) &= 2a_{11} + 1.6a_{12} + 1.28a_{13} + 1.6a_{14} + 2a_{15}, \\ EI_3^e(f) &= -5.10128a_{11} - 5.68814a_{12} - 5.83043a_{13} - 5.68828a_{14} - 5.11058a_{15} \end{aligned}$$

where the upper indices  $a, b, e$  account for the order in the set of points (see Figure 2). It turns out that the only important coefficients to measure the error are the ones which correspond to the highest degree in the polynomial. Finally, without a deeper analysis, we can only say that when  $a_{11}, a_{12}, a_{13}, a_{14}, a_{15}$  are all of equal sign the errors satisfy:

$$|EI_3^a(f)| \leq |EI_3^b(f)| \leq |EI_3^e(f)| \quad (18)$$

## 4.3 Interpolation error

As we have different sets of points, we also have different Lagrange interpolating polynomials as approximations of the function of interest. Throughout this section we will focus on the study of the interpolation error, that we defined as follows:

$$E_{L_2}(\mathbf{f}) = \int_{\Omega} (\mathbf{f}(\boldsymbol{\xi}) - \mathbf{p}(\boldsymbol{\xi}))^2 \boldsymbol{\rho}(\boldsymbol{\xi}) d\boldsymbol{\xi} \quad (19)$$

where  $\mathbf{p}$  is the Lagrange interpolating polynomial



#### 4.4 2 dimensions interpolation error comparison

We compute the interpolation error for the set of points in Figure 2 orders  $a$ ,  $b$ ,  $e$  and as weight function the normal distribution given in Equation (6). We consider a general polynomial of degree 2:

$$f(x, y) = ax^2 + by^2 + cxy + dx + ey + f \quad (20)$$

The Lagrange interpolating polynomials for the three set of points are:

$$\begin{aligned} p_a(x, y) &= a - 0.28b + f + dx + ey + (1.6b + c)xy, \\ p_b(x, y) &= a + b + f + dx + ey + cxy, \\ p_e(x, y) &= 0.219489a + 0.219509b + f + dx + ey + (0.975595a + 0.975613b + 0.999996c)xy, \end{aligned}$$

Applying Equation (19) gives the  $L_2$  interpolating error:

$$\begin{aligned} E_{L_2}^a(a, b) &= 2a^2 - 2.56ab + 1.0784b^2, \\ E_{L_2}^b(a, b) &= 2a^2 + 2.56ab + 2b^2, \\ E_{L_2}^e(a, b) &= 0.439024a^2 - 0.561951ab + 0.439024b^2, \end{aligned}$$

The minimum of all the errors is (0,0) (computed analytically). Figure 3 shows the behavior of the error. As we can see, order  $e$ , which was the order with highest integration error, is the order with the smallest interpolation error. On the other hand, order  $a$  and order  $b$  are similar, but it is possible to appreciate a slight difference between them in favor of order  $a$ .

#### 4.5 Optimal set of points

In this section we will check the convergence of the sets of points when the correlation is increased. We know in advance the optimal points for the extreme cases. For uncorrelated inputs the optimal set of points is the set that comes from the tensor product approach and for fully correlated inputs the one that comes from the unidimensional Gaussian quadrature points. Thus, it is reasonable to think that optimal points for correlation  $\rho \in [0, 1]$  should converge to these limiting cases. Figure 4 give us an idea about how the behavior of the points is when the correlation is increased. The plot shows that the set of points computed with order  $a$  is the optimal set of points. Optimal in the sense that when the problem is fully correlated the four points are on the fully correlated line, and when the problem is uncorrelated the four points are equal to the tensor product approach points.

## 5 CONCLUSIONS

In this paper we have shown how to compute new sets of points for Stochastic Collocation for correlated inputs. As the points were not unique in the multidimensional case, we have also performed an analytical comparison between sets of points in terms of integration degree of accuracy, integration error, interpolation error and convergence. We can conclude that: i) analytically it is possible to prove that all sets have the same integration degree of exactness;

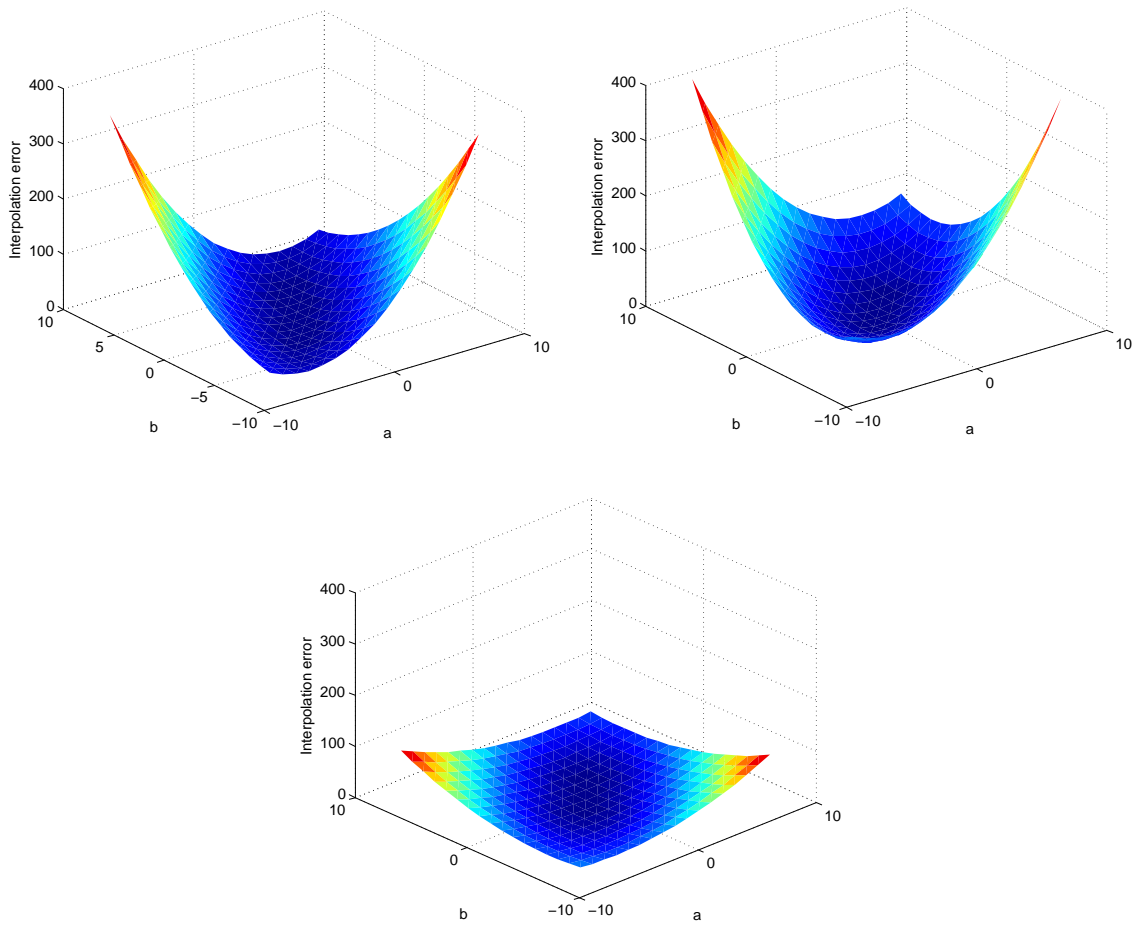


Figure 3: Interpolation error with order  $a$ , order  $b$ , and order  $e$ .

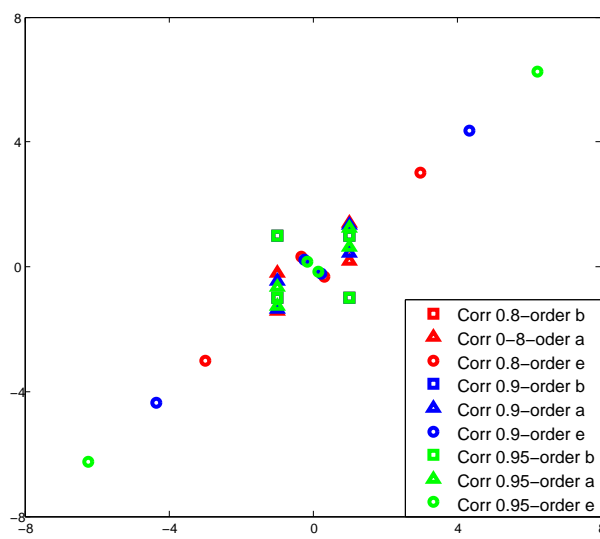


Figure 4: Convergence of the set of points when the correlation is increased.

ii) the sets differ in terms of integration and interpolation errors, specifically the set with the smallest integration error presents the highest interpolation error and vice versa; iii) one of the set of points is optimal in the sense that in the limit to full and no correlation its points are equal to the known optimal ones.

**APPENDIX A**

In this Appendix we plot the sets of collocation points for a polynomial of degree 3. To generate the polynomials of degree 3, we use this set of linearly independent monic polynomials:  $\{\xi_1^3, \xi_2^3, \xi_1^2\xi_2, \xi_1\xi_2^2\}$ . The number of combinations is  $4!$ , meaning, 24 different sets of collocation points. The results are presented in four groups, where Figure 5 is the group whose first element in each order is always  $\xi_1^3$ , Figure 6 is the one with  $\xi_2^3$ , and Figure 7 and Figure 8 with  $\xi_1^2\xi_2$  and  $\xi_1\xi_2^2$  respectively.

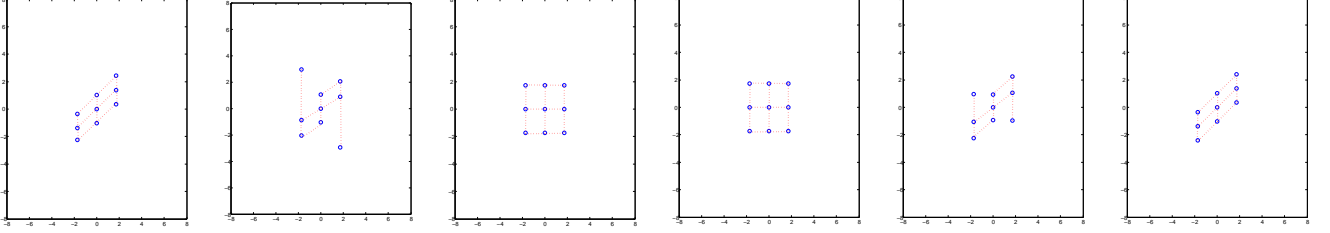


Figure 5: order a:  $\{\xi_1^3, \xi_1^2\xi_2, \xi_1\xi_2^2, \xi_2^3\}$ , order b:  $\{\xi_1^3, \xi_1^2\xi_2, \xi_2^3, \xi_1\xi_2^2\}$ , order c:  $\{\xi_1^3, \xi_2^3, \xi_1^2\xi_2, \xi_1\xi_2^2\}$ , order d:  $\{\xi_1^3, \xi_2^3, \xi_1\xi_2^2, \xi_1^2\xi_2\}$ , order e:  $\{\xi_1^3, \xi_1\xi_2^2, \xi_2^3, \xi_1^2\xi_2\}$  and order f:  $\{\xi_1^3, \xi_1^2\xi_2, \xi_1\xi_2^2, \xi_2^3\}$ .

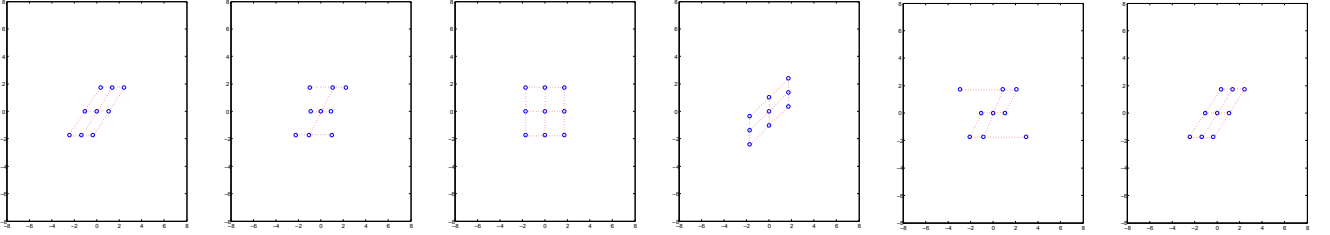


Figure 6: order g:  $\{\xi_2^3, \xi_1^2\xi_2, \xi_1\xi_2^2, \xi_1^3\}$ , order h:  $\{\xi_2^3, \xi_1^2\xi_2, \xi_1^3, \xi_1\xi_2^2\}$ , order i:  $\{\xi_2^3, \xi_1^3, \xi_1^2\xi_2, \xi_1\xi_2^2\}$ , order j:  $\{\xi_2^3, \xi_1^3, \xi_1\xi_2^2, \xi_1^2\xi_2\}$ , order k:  $\{\xi_2^3, \xi_1\xi_2^2, \xi_1^3, \xi_1^2\xi_2\}$  and order l:  $\{\xi_2^3, \xi_1^2\xi_2, \xi_1^2\xi_2, \xi_1^3\}$ .

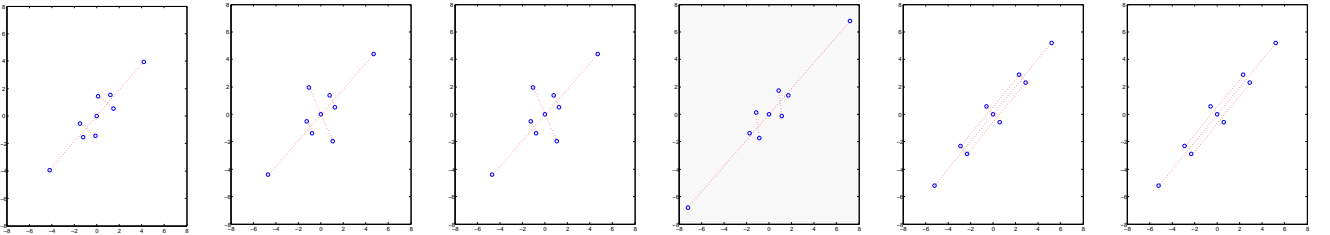


Figure 7: order m:  $\{\xi_1^2\xi_2, \xi_2^3, \xi_1\xi_2^2, \xi_1^3\}$ , order n:  $\{\xi_1^2\xi_2, \xi_2^3, \xi_1^3, \xi_1\xi_2^2\}$ , order o:  $\{\xi_1^2\xi_2, \xi_1^3, \xi_1\xi_2^2, \xi_2^3\}$ , order p:  $\{\xi_1^2\xi_2, \xi_1^3, \xi_2^2\xi_1, \xi_2^2\}$ , order q:  $\{\xi_1^2\xi_2, \xi_1\xi_2^2, \xi_1^3, \xi_2^3\}$  and order r:  $\{\xi_1^2\xi_2, \xi_1\xi_2^2, \xi_2^3, \xi_1^3\}$ .

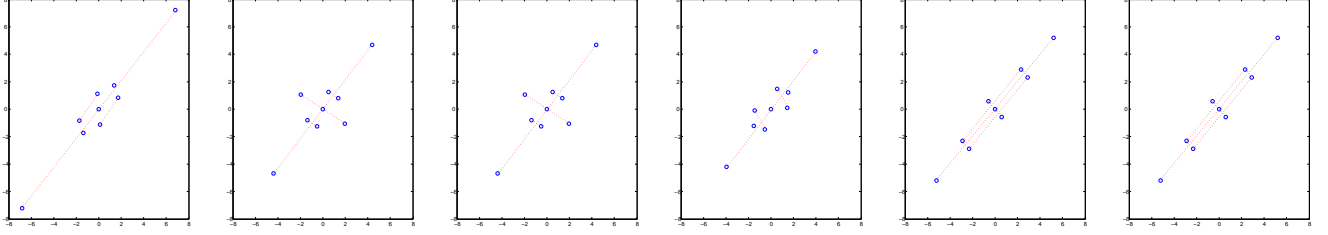


Figure 8: order s:  $\{\xi_1\xi_2^2, \xi_2^3, \xi_1^2\xi_2, \xi_1^3\}$ , order t:  $\{\xi_1\xi_2^2, \xi_2^3, \xi_1^3, \xi_1^2\xi_2\}$ , order u:  $\{\xi_1\xi_2^2, \xi_1^3, \xi_1^2\xi_2, \xi_2^3\}$ , order v:  $\{\xi_1\xi_2^2, \xi_1^3, \xi_2\xi_1^2, \xi_2^2\}$ , order w:  $\{\xi_1\xi_2^2, \xi_1^2\xi_2, \xi_1^3, \xi_2^3\}$  and order x:  $\{\xi_1\xi_2^2, \xi_1^2\xi_2, \xi_2^3, \xi_1^3\}$ .

## APPENDIX B

In this section we define and prove some technicalities that are necessary to prove Theorem 4.2. We will start introducing the concept of pseudodivision. Note that proof of Theorem 5.1 and Theorem 5.2 are based on references [24, 4].

**Theorem 5.1.** *Let  $p(x)$  and  $\Phi(x) \neq 0$  be two polynomials in  $\mathcal{P}[x]$  of respective degrees  $n$  and  $m$ :*

$$\begin{aligned} p(x) &= b_mx^m + \dots + b_0, \\ \Phi(x) &= a_nx^n + \dots + a_0. \end{aligned}$$

Let  $\delta = \max(m - n + 1, 0)$ . Then there exist polynomials  $q(x)$  and  $r(x)$  in  $\mathcal{P}[x]$  such that

$$a_n^\delta p(x) = q(x)\Phi(x) + r(x) \quad \text{and} \quad \deg(r) < \deg(\Phi). \quad (21)$$

Moreover, if  $a_n$  is not a zero divisor, then  $q(x)$  and  $r(x)$  are unique.

*Proof.*

We can show the existence of the polynomials  $q(x)$  and  $r(x)$  by induction on  $m$ :

- $m < n$

Take  $q(x) = 0$  and  $r(x) = \Phi(x)$ .

- $m \geq n$

The polynomial

$$\widehat{p}(x) = a_n \cdot p(x) - b_mx^{m-n} \cdot \Phi(x)$$

has degree at most  $(m - 1)$ .

If we pseudodivide  $\widehat{p}(x)$  by  $\Phi(x)$ , then by the inductive hypothesis, there exist polynomials  $\widehat{q}(x)$  and  $\widehat{r}(x)$  such that:

$$a_n^{m-n} \left( a_n \cdot p(x) - b_mx^{m-n} \cdot \Phi(x) \right) = \widehat{q}(x) \cdot \Phi(x) + \widehat{r}(x) \quad \text{and} \quad \deg(\widehat{r}) < \deg(\Phi).$$

If we leave just  $p(x)$  in the lhs:

$$a_n^{m-n} \cdot a_n \cdot p(x) = a_n^{m-n} b_mx^{m-n} \cdot \Phi(x) + \widehat{q}(x) \cdot \Phi(x) + \widehat{r}(x)$$

Taking  $q(x) = a_n^{m-n}b_mx^{m-n} + \widehat{q}(x)$  and  $r(x) = \widehat{r}(x)$ ,

$$a_n^\delta p(x) = q(x)\Phi(x) + r(x) \quad \text{and} \quad \deg(r) < \deg(\Phi).$$

**Proof of Uniqueness:**

Suppose that there are  $q(x)$ ,  $\widehat{q}(x)$ ,  $r(x)$  and  $\widehat{r}(x)$  verifying  $q(x) - \widehat{q}(x) \neq 0$ ,  $r(x) - \widehat{r}(x) \neq 0$  and such that:

$$\begin{aligned} a_n^\delta p(x) &= q(x) \cdot \Phi(x) + r(x) \quad \text{and} \quad \deg(r) < \deg(\Phi) \\ a_n^\delta p(x) &= \widehat{q}(x) \cdot \Phi(x) + \widehat{r}(x) \quad \text{and} \quad \deg(\widehat{r}) < \deg(\Phi) \end{aligned}$$

then  $0 = (q(x) - \widehat{q}(x))\Phi + (r(x) - \widehat{r}(x))$ , since  $a_n \neq 0$ , then  $[q(x) - \widehat{q}(x)]\Phi(x) \neq 0$  and has degree at least  $n$  and degree  $\deg(r - \widehat{r}) < \deg(\Phi) = n$ . However, this is impossible since

$$(q(x) - \widehat{q}(x))\Phi(x) = \widehat{r}(x) - r(x). \quad (22)$$

Thus  $q(x) - \widehat{q}(x) = 0 = \widehat{r}(x) - r(x)$ .

Theoreme 5.1 allows us to define the concept of pseudodivision.

**Definition 5.2** (Pseudodivision). *For any two polynomials  $p(x)$  and  $\Phi(x) \neq 0$  in  $\mathcal{P}[x]$ , we will call polynomials  $q(x)$  and  $r(x)$  in  $\mathcal{P}[x]$  the pseudoquotient and the pseudoremainder, respectively, of  $p(x)$  with respect to  $\Phi(x)$ , denoted  $pquo(p, \Phi)$  and  $prem(p, \Phi)$  if*

$$a_n^\delta p(x) = q(x)\Phi(x) + r(x) \quad \text{and} \quad \deg(r) < n \quad (23)$$

where  $m = \deg(p)$ ,  $n = \deg(\Phi)$ ,  $b_n = \text{lcoef}(\Phi)$  the leading coefficient of  $\Phi(x)$  and  $\delta = \max(m - n + 1, 0)$ . Also, if  $p = \text{prem}(p, \Phi)$ , then  $p(x)$  is said to be reduced with respect to  $\Phi(x)$ .

The notion of pseudodivision can be generalized to polynomials in more than one variable. Given two nonzero polynomials  $\Phi(x_1) \in \mathcal{P}[x_1]$  and  $\mathbf{p}(x_1, \dots, x_n) \in \mathcal{P}[x_1, \dots, x_n]$ , it is possible to find two polynomials  $\mathbf{q}(x_1, \dots, x_n)$  pseudoquotient and  $\mathbf{r}(x_1, \dots, x_n)$  pseudoremainder such that

$$b_n^\delta \mathbf{p}(x_1, \dots, x_n) = \mathbf{q}(x_1, \dots, x_n) \cdot \Phi(x_1) + \mathbf{r}(x_1, \dots, x_n), \quad \text{and} \quad \deg_{x_1}(\mathbf{r}) < \deg(\Phi),$$

where  $b_n = \text{lcoef}(\Phi)$  and  $\delta = \max(\deg_{x_1}(\mathbf{p}) - \deg(\Phi) + 1, 0)$ . To make clear that the pseudodivision is performed with respect to the variable  $x_1$ , we will write

$$\mathbf{q}(x_1, \dots, x_n) = pquo(\mathbf{p}, \Phi, x_1) \quad \text{and} \quad \mathbf{r}(x_1, \dots, x_n) = \text{prem}(\mathbf{p}, \Phi, x_1).$$

**Theorem 5.3** (Iterated Pseudodivision). *Consider the following succession of triangular polynomials:*

$$\begin{aligned} &\Phi_1(u_1, \dots, u_d, x_1) \\ &\Phi_2(u_1, \dots, u_d, x_1, x_2) \\ &\quad \vdots \\ &\Phi_n(u_1, \dots, u_d, x_1, \dots, x_n) \end{aligned} \quad (24)$$

and polynomial  $\mathbf{p}(u_1, \dots, u_d, x_1, \dots, x_n)$  all in the ring  $\mathcal{P}[u_1, \dots, u_d, x_1, \dots, x_n]$ . Let the following sequence of polynomials be obtained by iterated pseudodivisions:

$$\begin{aligned}
 \mathbf{r}_n &= \mathbf{p} \\
 \mathbf{r}_{n-1} &= \text{prem}(\mathbf{r}_n, \Phi_n, x_n) \\
 \mathbf{r}_{n-2} &= \text{prem}(\mathbf{r}_{n-1}, \Phi_{n-1}, x_{n-1}) \\
 &\vdots \\
 \mathbf{r}_0 &= \text{prem}(\mathbf{r}_1, \Phi_1, x_1).
 \end{aligned}$$

The polynomial  $\mathbf{r}_0 \in \mathcal{P}[u_1, \dots, u_d, x_1, \dots, x_n]$  is said to be the generalized pseudoremainder of  $\mathbf{p}$  with respect to  $\Phi_1, \dots, \Phi_n$  and denoted

$$\mathbf{r}_0 = \text{prem}(\mathbf{p}, \{\Phi_n, \dots, \Phi_1\}). \quad (25)$$

We also say  $\mathbf{p}$  is reduced with respect to  $\Phi_1, \dots, \Phi_n$  if

$$\mathbf{p} = \text{prem}(\mathbf{p}, \{\Phi_n, \dots, \Phi_1\}). \quad (26)$$

Furthermore, there are non-negative integers  $\delta_1, \dots, \delta_n$  and polynomials  $\mathbf{q}_1, \dots, \mathbf{q}_n$  such that

$$i) \quad b_n^{\delta_n} \cdots b_1^{\delta_1} \mathbf{p} = \mathbf{q}_1 \cdot \Phi_n + \cdots + \mathbf{q}_n \cdot \Phi_1 + \mathbf{r}_0$$

$$\begin{aligned}
 \text{where } b_1 &= \text{lcoef}(\Phi_1) \in \mathcal{P}[u_1, \dots, u_d], \\
 &\vdots \\
 b_n &= \text{lcoef}(\Phi_n) \in \mathcal{P}[u_1, \dots, u_d, x_1, \dots, x_{n-1}].
 \end{aligned}$$

ii)  $\deg_{x_i}(\mathbf{r}_0) < \deg_{x_i}(\Phi_i)$ , for  $i = 1, \dots, n$ .

*Proof.* The proof is by induction on  $r$  and by repeated applications of the pseudodivision theorem given in the beginning of the section.

**Lemma 5.4.** Let  $\mathbf{p}$  be a complete polynomial of degree  $m$  in  $n$  variables,  $\mathbf{p} \in \mathcal{P}_m[x_1, \dots, x_n]$ , then:

i) The number of terms in the polynomial  $\mathbf{p}$  with degree  $s \in \mathbb{N}$ ,  $0 \leq s \leq m$ ,  $nt_s(\mathbf{p})$ , is given by:

$$nt_s(\mathbf{p}) = \binom{s+n-1}{n-1} \quad (27)$$

ii) The total number of terms in the polynomial  $\mathbf{p}$ ,  $Tnt(\mathbf{p})$ , is given by:

$$Tnt(\mathbf{p}) = \binom{m+n}{n} \quad (28)$$

iii) The number of terms in the polynomial  $\mathbf{p}$  where a specific variable,  $x_i$  has an specific degree  $s \in \mathbb{N}$  with  $0 \leq s \leq m$ ,  $nt_s^{x_i}(\mathbf{p})$ , is given by:

$$nt_s^{x_i}(\mathbf{p}) = \binom{(m-s) + (n-1)}{n-1} \quad (29)$$

In other words,  $nt_s^{x_i}(\mathbf{p})$  is the number of times that the variable  $x_i^s$  appears in  $\mathbf{p}$

*Proof.*

i) A general term of degree  $s$  of the polynomial  $\mathbf{p}$  can be expressed as

$$\prod_{i=1}^n x_i^{e_i} \text{ such that } \sum_{i=1}^n e_i = s. \quad (30)$$

To find the number of terms that verifies Equation 30 is equal to find the number of solutions of the equation  $e_1 + \dots + e_n = s$ , where  $e_i$  is a non-negative integers for each  $i \in \{1, \dots, n\}$ . To find the solution of that equation we apply a technique popularly known as “stars and bars” [ref]. The idea behind this method is, essentially, if there are  $s$  indistinguishable objects to be divided into  $n$  distinguishable groups, then there are  $C_{s+n-1, n-1}$  ways of distributing the objects. We assume the objects are “stars” in our problem the exponents and the groups are the “bars” in our problem the variables. Since 1 bar divides into 2 groups, 2 bars into 3 groups, and so on, the number of bars will be  $n - 1$  and therefore the number of the solution for the equation is  $C_{s+n-1, n-1} = \binom{s+n-1}{n-1}$ .

ii) As we know the number of terms of a specific degree we compute the sum of all them:

$$Tnt_{\mathbf{p}} = \binom{0+n-1}{n-1} + \binom{1+n-1}{n-1} + \dots + \binom{m+n-1}{n-1} = \sum_{j=n-1}^{m+n-1} \binom{j}{n-1}$$

By applying the Hockeystick Identity:  $\sum_{j=n}^m \binom{j}{n} = \binom{m+1}{n+1}$ , we obtain:

$$Tnt(\mathbf{p}) = \sum_{j=n-1}^{m+n-1} \binom{j}{n-1} = \binom{m+n}{n}$$

iii) We use again the “stars and bars” technique, where the exponents are the “stars” and the variables the “bars”. As we know that one variable has fixed degree equals  $s$ , then we have  $m - s$  “stars” and  $n - 2$  “bars” and the number  $C_{(m-s)+(n-2), n-2}$  gives us the number of terms of degree  $m$  where the fixed variable has degree  $s$ . To compute all terms that contains that fixed variable we have to sum everything

$$nt_s^{x_i}(\mathbf{p}) = \binom{(m-s)+n-2}{n-2} + \binom{(m-s)-1+n-2}{n-2} + \dots + \binom{0+n-2}{n-2} = \sum_{j=n-2}^{(m-s)+(n-2)} \binom{j}{n-2}$$

By applying the Hockeystick Identity, we obtain:

$$nt_s^{x_i}(\mathbf{p}) = \sum_{j=n-2}^{(m-s)+(n-2)} \binom{j}{n-2} = \binom{(m-s)+(n-1)}{n-1}$$

**Proposition 5.5.** *Let  $\mathbf{p}$  be a polynomial in the ring  $\mathcal{P}_{2m-1}[x_1, \dots, x_n]$  with  $m \in \mathbb{N}$  and  $\Phi_1, \dots, \Phi_m$ , such that:  $\Phi_1(x_1, \dots, x_n) = \Phi_1^m(x_1)$ ,  $\Phi_2(x_1, \dots, x_n) = \Phi_2^m(x_1, x_2)$ ,  $\dots$ ,  $\Phi_m(x_1, \dots, x_n) = \Phi_m^m(x_1, \dots, x_n)$ , a succession of triangular polynomials in the ring  $\mathcal{P}_m[x_1, \dots, x_n]$ , where each triangular polynomial has only one terms of degree  $m$ , then:*



- i) the generalized pseudoremainder of  $\mathbf{p}$  with respect to  $\Phi_1^m, \dots, \Phi_m^m, \mathbf{r}_0 = \text{prem}(\mathbf{p}, \{\Phi_m, \dots, \Phi_1\})$ , has at least  $\binom{(2m-1)+n}{n} - n \binom{n+m-1}{n-1}$  number of terms.
- ii) The non-linear system of equation formed by the set of orthogonal triangular polynomials has at least  $N = m^n$  real and different solutions.
- iii)  $N \geq \text{Cnt}(\mathbf{r}_0)$ .

*Proof.*

i) Analogously to the proof of Lemma 5.4.

ii) Because of the orthogonality of the polynomials. It is known that unidimensional orthogonal polynomials of degree  $d$  always have  $d$  real roots. Given that, the system of linear equations is triangular, the first equation always will have all its roots because it will correspond to a unidimensional orthogonal polynomial. If we replace the roots into the second equation, we will be able to compute all the roots. After substituting the polynomial is a unidimensional orthogonal polynomial again.

iii) Analogously to the proof of Lemma 5.4.

### Acknowledgment

The authors would like to thank Professor Jose Luis Bueso Montero, University of Granada, for assisting with preparation of proofs in Appendix B. JB also acknowledges support by European Commission's 7th Framework Program, project BioPreDyn grant number 289434.

### REFERENCES

- [1] H.L. Anderson, Metropolis, Monte Carlo and the MANIAC, Los Alamos Science, 14 (1986), pp. 96-108.
- [2] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, *Journal of Computational Physics*, **230(6)**, 20 March 2011, 2345-2367, 2011.
- [3] B.M. Chen-Charpentier, J.C. Cortes, J.A. Licea, J.V. Romero, M.D. Rosello, F.J. Santonja, R.J. Villanueva, Constructing adaptive generalized polynomial chaos method to measure the uncertainty in continuous models: A computational approach, *Mathematics and Computers in Simulation*, **109**, 113-129, 2015.
- [4] S.-S. Chou, Mechanical Geometry Theorem Proving, D. Reidel Publishing Company, 1988.
- [5] B. Fox, Strategies for Quasi-Monte Carlo, Kluwer academic Pub., 1999.
- [6] M. Kleiber, T.D. Hien, The stochastic finite element method: basic perturbation technique and computer implementation, Wiley, 1992.
- [7] G. Lin, A.M. Tartakovsky, D.M. Tartakovsky, Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids, *Journal of Computational Physics*, **229(19)**, 6995-7012, 2010.

- [8] B. Lockwood, D. Mavriplis, Gradient-based methods for uncertainty quantification in hypersonic flows, *Computers & Fluids*, **85**, 27-38, 2013.
- [9] W. Loh, On Latin hypercube sampling, *Annals of Statistics*, 24 (1996), pp. 2058-2080.
- [10] D. Lu, M. Ye, M.C. Hill, E.P. Poeter, G.P. Curtis, A computer program for uncertainty analysis integrating regression and Bayesian methods, *Environmental Modelling & Software*, **60**, 45-56, 2014.
- [11] Z. Lu, D. Zhang, A Comparative Study on Uncertainty Quantification for Flow in Randomly Heterogeneous Media Using Monte Carlo Simulations and Conventional and KL-Based Moment-Equation Approaches, *SIAM Journal on Scientific Computing*, **26(2)**, 558-577, 2004.
- [12] N. Madras, Lectures on Monte Carlo methods, American Mathematical Society, Providence, RI, 2002.
- [13] L. Mathelin, M.Y. Hussaini, A Stochastic Collocation algorithm for uncertainty analysis, Tech. Report NASA/CR-2003-212153, NASA Langley Research Center, 2003.
- [14] H.N. Najm, Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics, *Annual Review of Fluid Mechanics*, **41**, 35-52, 2009.
- [15] M. Navarro, J. Witteveen, J. Blom, Polynomial Chaos Expansion for general multivariate distributions with correlated variables, arXiv:1406.5483.
- [16] A. Rossa, K. Liechti, M. Zappa, M. Bruen, U. Germann, G. Haase, C. Keil, P. Krahe, The COST 731 Action: A review on uncertainty propagation in advanced hydro-meteorological forecast systems, *Atmospheric Research*, **100(23)**, 150-167, 2011.
- [17] K. Sanjeev, A simple expression for multivariate Lagrange interpolation, SIAM, 2007.
- [18] C. Soize, Polynomial Chaos Expansion of a Multimodal Random Vector, *SIAM/ASA Journal on Uncertainty Quantification*, **3(1)**, 34-60, 2015.
- [19] J. Stoer, R. Bulirsch, Introduction to numerical analysis, third edition, Springer, New York, 2002.
- [20] G. Tang, G. Iaccarino, Subsampled Gauss Quadrature Nodes for Estimating Polynomial Chaos Expansions, *SIAM/ASA Journal on Uncertainty Quantification*, 2(1), 423-443, 2014.
- [21] A. Torres Valderrama, J. Witteveen, M. Navarro, J. Blom, Uncertainty Propagation in Nerve Impulses Through the Action Potential Mechanism, *The Journal of Mathematical Neuroscience*, **5(3)**, 2015.
- [22] H. Wang, D.A. Sheen, Combustion kinetic model uncertainty quantification, propagation and minimization, *Progress in Energy and Combustion Science*, **47(0)**, 1-31, 2015.
- [23] N. Wiener, The homogeneous chaos, *American Journal of Mathematics*, **60**, 897-936, 1938.

- [24] W.W. Adams, P. Loustau, An Introduction to Grobner Bases, American Mathematical Society, 1991.
- [25] D. Xiu, J.S. Hesthaven, High-order collocation methods for differential equations with random inputs, *SIAM Journal on Scientific Computing*, 27(3), 1118-1139, 2005.
- [26] B. Yildirim, G.E. Karniadakis, Stochastic simulations of ocean waves: An uncertainty quantification study, *Ocean Modelling*, **86**, 15-35, 2015.