Journal of Scientific Computing (2021) 87:82
https://doi.org/10.1007/s10915-021-01498-9

Poly-Sinc Solution of Stochastic Elliptic Differential Equations

Maha Youssef[1] · **Roland Pulch**[1]

Received: 21 May 2019 / Revised: 18 April 2020 / Accepted: 16 April 2021 / Published online: 30 April 2021

Abstract
In this paper, we introduce a numerical solution of a stochastic partial differential equation (SPDE) of elliptic type using polynomial chaos along side with polynomial approximation at Sinc points. These Sinc points are defined by a conformal map and when mixed with the polynomial interpolation, it yields an accurate approximation. The first step to solve SPDE is to use stochastic Galerkin method in conjunction with polynomial chaos, which implies a system of deterministic partial differential equations to be solved. The main difficulty is the higher dimensionality of the resulting system of partial differential equations. The idea here is to solve this system using a small number of collocation points in space. This collocation technique is called Poly-Sinc and is used for the first time to solve high-dimensional systems of partial differential equations. Two examples are presented, mainly using Legendre polynomials for stochastic variables. These examples illustrate that we require to sample at few points to get a representation of a model that is sufficiently accurate.

Keywords Poly-Sinc methods · Collocation method · Galerkin method · Stochastic differential equations · Polynomial chaos · Legendre polynomials

Mathematics Subject Classification 65N35 · 65N12 · 65N30 · 65C20 · 35R60

1 Introduction

In many applications the values of the parameters of the problem are not exactly known. These uncertainties inherent in the model yield uncertainties in the results of numerical simulations. Stochastic methods are one way to model these uncertainties by using random fields [1]. If the physical system is described by a partial differential equation (PDE), then the combination with the stochastic model results in a stochastic partial differential equation (SPDE). The solution of the SPDE is again a random field, describing both the expected response and quantifying its uncertainty. SPDEs can be interpreted mathematically in several ways.

✉ Maha Youssef
maha.youssef@uni-greifswald.de

Roland Pulch
roland.pulch@uni-greifswald.de

[1] Institute of Mathematics and Computer Science, University of Greifswald, Walther-Rathenau-Straße 47, 17489 Greifswald, Germany

In the numerical framework, the stochastic regularity of the solution determines the convergence rate of numerical approximations, and a variational theory for this was earlier devised in [2]. The ultimate goal in the solution of SPDEs is usually the computation of response statistics, i.e. a functional of the solution. Monte Carlo (MC) methods can be used directly for this purpose, but they require a high computational effort [3,4]. Quasi Monte Carlo (QMC) and variance reduction techniques [3] may reduce the computational effort considerably without requiring much regularity. However, often we have high regularity in the stochastic variables, and this is not exploited by QMC methods.

Alternatives to MC methods have been developed, for example, perturbation methods [5], methods based on Neumann-series [6], or the spectral stochastic finite element method (SSFEM) [7,8]. Stochastic Galerkin methods have been applied to various linear problems, see [7,9,10]. Nonlinear problems with stochastic loads have been tackled in [11]. These Galerkin methods yield an explicit functional relationship between the independent random variables and the solution. In contrast with common MC methods, subsequent evaluations of functional statistics like the mean and covariance are very cheap.

In this paper, we do not consider (classical) stochastic differential equations driven by Wiener processes, cf. [12]. Alternatively, we investigate an elliptic PDE in space including a random field as material parameters. The polynomial chaos approach and the stochastic Galerkin method yield a deterministic system of PDEs in space [13]. We introduce a spatial collocation technique based on polynomial approximation by Lagrange interpolation. For the interpolation points we use a specific set of non-uniform points created by conformal maps, called Sinc points. Later, we use a small number of Sinc points as collocation points to compute a very accurate solution of the PDEs, see [14]. This technique is called Poly-Sinc collocation that was used to solve a single elliptic PDE in [15]. Recently it has been used to solve coupled system of parabolic equations [16]. In this paper, we use Poly-Sinc collocation to solve a high-dimensional system of PDEs resulting from a Galerkin projection, which shows that with a small degree of freedom we can get a highly accurate solution.

The paper is organized as follows: In Sect. 2, we introduce a model problem, the structure of its polynomial chaos model and the stochastic Galerkin solution. In Sect. 3, we illustrate the main theorem of Poly-Sinc approximation. In Sect. 4, we review a Poly-Sinc collocation technique with the main collocation theorem. Finally, in Sect. 5, we investigate numerical examples. We start with a simple example in one stochastic variable and then we discuss the model with multiple stochastic variables from Sect. 2.

2 Stochastic Model Problem

In this paper, we are interested to solve the following stochastic partial differential equations:

$$\begin{aligned} \mathcal{L}(u) \equiv -\nabla \cdot (a(x, y, \Theta)\nabla u(x, y, \Theta)) = f(x, y) \text{ in } Q \times \Omega \text{ and} \\ u = 0 \text{ on } \partial Q \times \Omega, \end{aligned} \tag{1}$$

where $\Theta = (\xi_1, \xi_2, \ldots, \xi_K)$ is a vector of stochastic parameters. These parameters are independent and uniformly distributed in $I = [-1, 1]$ and thus $\Theta : \Omega \longrightarrow [-1, 1]^K$ with an event space Ω. Moreover the domain of the spatial variables x and y is $Q = (0, 1)^2$ or $Q = (-1, 1)^2$. The function $a(x, y, \Theta)$ is defined as

$$a(x, y, \Theta) = a_0(x, y) + b_0 \sum_{k=1}^{K} \xi_k a_k(x, y), \tag{2}$$

where a_k's are functions in x and y only, b_0 is a constant and, ξ_k's are the random variables. Without loss of generality, we consider $a_0 = 1$ and $b_0 = 1/2$. We assume that $a(x, y, \Theta) \geq \alpha > 0$ for all $(x, y) \in Q$ and all $\Theta \in [-1, 1]^K$. Thus the differential operator in (1) is always uniformly elliptic. Equation (2) is a truncated form of a Karhunen-Loeve expansion where the spatial functions a_k can be eigenfunctions of a covariance operator and the random variables ξ_k are uncorrelated, see [17]. Alternatively, we will define these spatial functions ourselves. In addition, the random variables are independent in our case.

In the rest of the section, we introduce the main concepts used in the solution of (1) with (2). Basically, we discuss the polynomial chaos in one- and multidimensional cases and the stochastic Galerkin method.

2.1 Polynomial Chaos Expansion

Generalized Polynomial Chaos (gPC) is a particular set of polynomials in a given random variable, with which an approximation of a random variable with finite second moments is computed. This procedure is named Polynomial Chaos Expansion (PCE). This technique exploits orthogonal properties of polynomials involved, to detect a representation of random variables as series of functionals. Now, the function u can be expressed as an infinite series of orthogonal basis functions Φ_i with suitable coefficient functions u_i as

$$u(x, y, \Theta) = \sum_{i=0}^{\infty} u_i(x, y)\Phi_i(\Theta). \tag{3}$$

The expansion in (3) converges in the mean square of the probability space. The truncation form including $m + 1$ basis functions leads to

$$u(x, y, \Theta) \simeq \widetilde{u}(x, y, \Theta) = \sum_{i=0}^{m} u_i(x, y)\Phi_i(\Theta) \tag{4}$$

with coefficients functions

$$u_i(x, y) = \langle u(x, y, \Theta),\ \Phi_i(\Theta)\rangle\ ,\quad i = 0, 1, \ldots, m.$$

A fundamental property of the basis functions is the orthogonality,

$$\left\langle \Phi_i(\Theta),\ \Phi_j(\Theta)\right\rangle = \int_{I^K} \Phi_i(\Theta)\ \Phi_j(\Theta)\ W(\Theta)d\Theta = c_i\ \delta_{ij},\ \text{ for all } i, j, \tag{5}$$

where c_i are real positive numbers and δ_{ij} is the Kronecker-delta. In general, the inner product in (5) can be defined for different types of weighting function W; however, it is possible to prove that the optimal convergence rate of a gPC model can be achieved when the weighting function W agrees to the joint probability density function (PDF) of the random variables considered in a standard form [9,18]. In this framework, an optimal convergence rate means that a small number of basis functions is sufficient to obtain an accurate PC model (4). Hence, the choice of the basis functions depends only on the probability distribution of the random variables Θ, and it is not influenced by the type of system under study. In particular, if the random variables Θ are independent, their joint PDF corresponds to the product of the PDFs of each random variable: in this case, the corresponding basis functions Φ_i can be calculated as product combinations (tensor product) of the orthogonal polynomials corresponding to

each individual random variable [13,17,19,20]:

$$\Phi_i(\Theta) = \Phi_{\mathbf{i}}(\Theta) := \prod_{r=1}^{K} \Phi_{i_r}^{(r)}(\xi_r), \ \mathbf{i} = (i_1, \ldots, i_K), \tag{6}$$

where $\Phi_{i_r}^{(r)}$ represents the univariate basis polynomial of degree i_r associated to the rth random variable and with one-to-one correspondence between the integers i and the multi-indices $\mathbf{i}$. We assume degree$(\Phi_i) \leq$ degree(Φ_{i+1}) for each i. Now let

$$\mathcal{R}_P = \left\{ \Phi_i(\Theta) : \sum_{r=1}^{K} i_r \leq P \right\}, \tag{7}$$

be the set of all multivariate polynomials up to total degree P. Furthermore, for random variables with specific PDFs, the optimal basis functions are known and are formed by the polynomials of the Wiener-Askey scheme [9]. For example, in the uniform probability distribution, the basis functions are the Legendre polynomials.

Using (6) and (7), it is possible to show that the total number of basis functions $m + 1$ in (4) is expressed as

$$m + 1 = \frac{(K + P)!}{K!P!}. \tag{8}$$

The total degree of the PC (the maximum degree) P can be chosen relatively small to achieve the desired accuracy in the solution.

In the case of the orthogonal polynomials, we can see that $\Phi_0(\Theta) = 1$ and for orthonormal polynomials

$$\langle \Phi_i(\Theta), \ \Phi_i(\Theta) \rangle = 1. \tag{9}$$

Once a PC model in the form of (4) is obtained, stochastic moments like the mean $E(u)$ and the variance $V(u)$ can be analytically calculated by the PC expansion coefficients, see [17]. The expected value reads as

$$E\left(u(x, y, \Theta)\right) = u_0(x, y).$$

The variance can be approximated by

$$V(u(x, y, \Theta)) \approx \sum_{i=1}^{m} u_i^2(x, y).$$

It is clear now that, in order to obtain a PC model in (4) and the stochastic moments, the coefficients functions $u_i(x, y)$ must be computed. The PC coefficient estimation depends on the type of the resulting system from the chaos expansion, not only the PC truncation.

2.2 Stochastic Galerkin Method

To solve the problem in (1) and (2), a Galerkin method is used along side the PC. The main idea is to assume that the solution of (1) and (2) is written as expansion in (4) and then use the PC theory introduced in the previous section. This process transform the SPDE (1) and (2) into a deterministic system of PDEs.

To recover the coefficient functions $u_i(x, y)$ we apply the inner product of (1) with the basis polynomial $\Phi_j(\Theta)$

$$\left\langle \mathcal{L}(\widetilde{u}) - f(x, y), \Phi_j(\Theta) \right\rangle = 0 \quad \text{for } j = 0, 1, \ldots, m. \tag{10}$$

Substituting (4) in (1) we obtain

$$\mathcal{L}(\widetilde{u}) = -\nabla \cdot \left(\nabla \sum_{i=0}^{m} u_i(x, y)\Phi_i(\Theta) \right) - \frac{1}{2} \sum_{k=1}^{K} \xi_k \nabla \cdot \left(a_k \nabla \sum_{i=0}^{m} u_i(x, y)\Phi_i(\Theta) \right).$$

Now applying the inner residual product in (10) and use the orthogonality property of the multivariate basis Φ_i's to get

$$-\nabla^2 u_j - \frac{1}{2} \sum_{k=1}^{K} \sum_{i=0}^{m} \left\langle \xi_k \Phi_i(\Theta), \Phi_j(\Theta) \right\rangle \nabla \cdot (a_k \nabla u_i) = F_j \tag{11}$$

where $F_j(x, y) = \left\langle f(x, y), \Phi_j(\Theta) \right\rangle$ forms an $(m + 1)$ vector and the array $\left\langle \xi_k \Phi_i, \Phi_j \right\rangle$ is a triple tensor of dimension $K \times (m + 1) \times (m + 1)$. (11) is a system of elliptic PDEs with unknown variables $u_i(x, y)$, $i = 0, 1, 2, \ldots, m$. With large number of random variables K (say $K \geq 5$) the size of the system in (11) becomes huge due to (8). One of our targets in the solution of the system in (11) is to use a collocation method to achieve a high accuracy with small numbers of collocation points. The proposed method in this report is to use Sinc points in a Lagrange interpolation.

2.3 Quadrature

The inner product (5) is defined by an integral. For the integration of polynomials analytic methods are used. Alternatively, we can use highly accurate quadrature techniques to evaluate the integrals exactly except for round-off errors. We omit the details of these techniques, since they can be easily found in several textbooks. For example, descriptions of Gaussian quadrature can be found in most texts on numerical analysis [21], while [22] contains descriptions of Sinc quadratures over finite, semi-infinite, infinite intervals and contours.

3 Poly-Sinc Approximation

A novel family of polynomial-like approximations that interpolate given Sinc data of the form $\{(x_j, u_j)\}_{j=-M}^{N}$ where the x_j are Sinc points was derived in [22] and extended in [23]. The interpolation to this data is of course accurate, provided that the function u with $u_j = u(x_j)$ belongs to a suitable space of functions. We also desire approximations of the derivative of the function u. On the one hand, this approximation can be obtained straightforward by a differentiation of the interpolant. On the other hand, this type of approximation of the derivative may not be very accurate, as it can happen in the case of Chebyshev polynomial, or Sinc approximation. The main purpose of this interpolation was to be able to get a more accurate method of obtaining an approximation for the derivative of the function u and to improve a wavelet or other method of approximation by a very simple procedure, see [23]. In [14,24,25], a complete theory of this approximation has been introduced where the error analysis for the function approximation, quadrature and stability have been studied. Moreover, this polynomial-like approximation (Poly-Sinc) has been used to solve singular boundary value problems based on ordinary differential equations (ODEs) and PDEs [14,15,26,27].

Let us first establish some mathematical notation which we shall require. Let $\mathbb{Z}$ denote the set of all integers, $\mathbb{R}$ the real line, and $\mathbb{C}$ the complex plane $\{\chi + \mathrm{i}\,\tau : \chi \in \mathbb{R}, \tau \in \mathbb{R}\}$. Given a number $d > 0$, we define the strip $\mathcal{D}_d$ as

$$\mathcal{D}_d = \{z \in \mathbb{C} : |\mathcal{I}z| < d\}.$$

Let $\mathcal{D} \subset \mathbb{C}$ be a simply connected domain having a boundary $\partial\mathcal{D}$, and let a and b denote two distinct points of $\partial\mathcal{D}$. Let $\phi : \mathcal{D} \longrightarrow \mathcal{D}_d$ denote a conformal map such that $\lim_{z\to a} \phi(z) = -\infty$ and $\lim_{z\to b} \phi(z) = \infty$, and let us define the inverse conformal map by $\psi = \phi^{-1}$, $\psi : \mathcal{D}_d \longrightarrow \mathcal{D}$. In addition, let Γ be an arc defined by

$$\Gamma = \{z \in [a,\, b] : z = \psi(x),\ x \in \mathbb{R}\}.$$

For real numbers a and $\Gamma \subseteq \mathbb{R}$, a one-dimensional Poly-Sinc approximation for a function u defined on an arc Γ using $n = M + N + 1$ Sinc points can be obtained by applying the following Lagrange interpolation formula:

$$u(x) \approx P_n(x) = \sum_{k=-M}^{N} b_k(x)\, u(x_k),\ x \in \Gamma \tag{12}$$

where $x_k = \psi(kh)$ are Sinc points on Γ and $b_k(x)$ are Lagrange basis polynomials. These polynomial basis are defined as follows

$$b_k(x) = \frac{g(x)}{(x - x_k)\, g'(x_k)},\quad g(x) = \prod_{j=-M}^{N} \big(x - x_j\big).$$

Using the interpolation (12) in calculations generates an accurate approximation with an exponentially decaying error rate which for $h = (\pi d/N)^{1/2}$ and $M = N$, by

$$\|u(x) - P_n(x)\| \le A\frac{\sqrt{N}}{B^{2N}}\ \exp\left(\frac{-\pi^2 N^{\frac{1}{2}}}{2}\right), \tag{13}$$

and

$$\left\|u'(x) - P_n'(x)\right\| \le C\frac{\sqrt{N}}{B^{2N}}\ \exp\left(\frac{-\pi^2 N^{\frac{1}{2}}}{2}\right), \tag{14}$$

where $\|.\|$ denotes the supremum norm on Γ and $A > 0$, $C > 0$, $B > 1$ are three constants, independent of N. For the proof of (13) and (14), see [23].

If a and b are finite real numbers then $\phi(x) = \log\left((x - a)/(b - x)\right)$ and the Sinc points are $x_j = (be^{jh} + a)/(1 + e^{jh})$. For the full list of conformal maps and Sinc points, see [22,23].

Another criterion to discuss the convergence and stability of the Poly-Sinc approximation is the Lebesgue constant. In [25] an estimate of the Lebesgue constant for Lagrange approximation at Sinc points has been derived as

$$\Lambda_n \approx \frac{1}{\pi}\log(n + 1) + 1.07618 \tag{15}$$

where $n = M+N+1$ is the number of Sinc points in (12). The upper bound of Lebesgue constant in (15) is better than the bounds using Chebyshev and Legendre points [25]. Besides the advantage of exponential decaying of the error from using Sinc points as interpolation points in Lagrange interpolation formula, especially in a finite interval, we can see that Lagrange

approximation at Sinc points delivers approximation results closer to the conjectured optimal approximation than using Chebyshev or Legendre points [25].

An extention of the one-dimensional Poly-Sinc approximation to the multi-dimensional case has been introduced in [24]. Let $X = (x_1, \ldots, x_l)$ be a point in an l-dimensional domain Q, then Poly-Sinc approximation of a function $u(X)$ can be defined by a nested operator as

$$u(X) \approx (P_n u)(X) = \sum_{k_1=-M_1}^{N_1} \sum_{k_2=-M_2}^{N_2} \cdots \sum_{k_l=-M_l}^{N_l} u(X_{\boldsymbol{k}})\, b_{k_1}(x_1) b_{k_2}(x_2) \cdots b_{k_l}(x_l), \quad (16)$$

where $X_{\boldsymbol{k}} = (x_{1,k_1}, \ldots, x_{l,k_l})$ with $k_i = -M_i, \ldots, N_i$ and x_{i,k_i} are the Sinc points in one-dimension.

Next, we assume $M_i = N_i = N$, $i = 1, \ldots, l$ and $n = 2N + 1$ is the number of Sinc points in each dimension $i = 1, 2, \ldots, l$. The convergence and stability of the approximation (16) are discussed in [24] and [25]. For the upper bound of the error E_n, we have

$$E_n = \|u(X) - (P_n u)(X)\| \leq \sum_{i=0}^{l-1} \left(C_i \log^i N \right) \frac{\sqrt{N}}{\gamma_i^{2N}} \exp\left(\frac{-\pi^2 N^{\frac{1}{2}}}{2} \right), \quad (17)$$

where $C_i > 0$, $\gamma_i > 1$, $i = 0, \ldots, l-1$ are two sets of constants, independent of N.

The notation $\Lambda_{n,l}$ is used to denote the Lebesgue constant using n interpolation points in each dimension $i = 1, 2, \ldots, l$, i.e. n^l Sinc points in total. If $P_n(X)$ is defined as in (16), then:

$$\Lambda_{n,l} \leq \left(\frac{1}{\pi} \log(n+1) + 1.07618 \right)^l. \quad (18)$$

We can write the interpolation scheme in (12) and (16) in simple operator form, as

$$u(x) \simeq \boldsymbol{B}(x)\, U \quad (19)$$

$$u(X) \simeq \bigodot_{i=1}^{l} \boldsymbol{B}_i(X)\, U, \quad (20)$$

where U is vector/matrix including data of the function u calculated at Sinc points.

4 Poly-Sinc Collocation Method

In [15], a collocation method based on the use of bivariate Poly-Sinc interpolation defined in (20) is introduced to solve elliptic equations defined on rectangular domains. In [26], Poly-Sinc collocation domain decomposition method for elliptic boundary value problems is investigated on complicated domains. In [27], a collocation method is introduced to solve certain type of singular differential equations. The idea of the collocation method is to reduce the boundary value problem to a system of algebraic equations which have to be solved subsequently. To start let us introduce the following collocation theorem.

Theorem 1 *Let $u : \overline{Q} \to \mathbb{R}$ be an analytic bounded function on the compact domain $\overline{Q}$. Let $U = \{u(x_j, y_k)\}_{j,k=-N}^{N}$ be a vector, where x_j and y_k are the Sinc points. If $\widetilde{U} = \{\widetilde{u}_{jk}\}_{j,k=-N}^{N}$ is a vector satisfying*

$$\left\| U - \widetilde{U} \right\|_\infty = \max_{j,k} \left| u_{jk} - \widetilde{u}_{jk} \right| < \delta,$$

then

$$\left\| u(x,y) - \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y)\,\widetilde{U} \right\| < E_n + \delta\,\Lambda_{n,2}, \tag{21}$$

where $n = 2N+1$, E_n *from* (17), *and* $\Lambda_{n,2}$ *from* (18).

Proof We apply triangle inequality

$$\begin{aligned}\left\| u(x,y) - \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y)\,\widetilde{U} \right\| &\le \left\| u(x,y) - \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y)\,U \right\| \\ &\quad + \left\| \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y)\,U - \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y)\,\widetilde{U} \right\| \\ &\le E_n + \delta \left\| \bigodot_{i=1}^{2} \boldsymbol{B}_i(x,y) \right\| \\ &\le E_n + \delta\,\Lambda_{n,2},\end{aligned}$$

which is the statement of the theorem. □

This theorem guarantees an accurate final approximation of u on its domain of definition provided that we know a good approximation to u at the Sinc points.

To set up the collocation scheme, let us consider the following partial differential operator,

$$\begin{aligned}\mathcal{L}u \equiv u_{xx} + u_{yy} = f(x,y),\ \ (x,y) \in Q, \\ u(x,y) = u_{ex}(x,y),\ \ (x,y) \in \partial Q,\end{aligned} \tag{22}$$

where $Q = \{a < x < b,\ c < y < d\}$ and $u_{xx} = \frac{\partial^2 u}{\partial x^2}$, $u_{yy} = \frac{\partial^2 u}{\partial y^2}$.

The first step in the collocation algorithm is to replace $u(x,y)$ in Eq. (22) by the Poly-Sinc approximation defined in (16). Next, we collocate the equation by replacing x and y by Sinc points

$$x_i = \frac{a + b\,\mathrm{e}^{ih}}{1 + \mathrm{e}^{ih}},\ i = -M, \ldots, N$$

and

$$y_q = \frac{c + d\,\mathrm{e}^{qh}}{1 + \mathrm{e}^{qh}},\ q = -M, \ldots, N.$$

In this case, we have,

$$u_{xx}(x_i, y_q) \approx \sum_{k=-M}^{N} \sum_{j=-M}^{N} u_{jk}\, B''(j,h)(x_i) B(k,h)(y_q),$$

where

$$B(j,h)(x_i) = \delta_{ji} = \begin{cases} 0 & j \ne i. \\ 1 & j = i, \end{cases}$$

and $B^{''}(j,h)(x_i)$ defines an $n \times n$ matrix, with $n = M + N + 1$

$$B^{''}(j,h)(x_i) = [b_{ji}] = \begin{cases} \frac{-2g'(x_i)}{(x_i - x_j)^2 g'(x_j)} + \frac{g''(x_i)}{(x_i - x_j)g'(x_j)} & \text{if } j \neq i \\ \sum_{s=-M}^{N} \sum_{\substack{l=-M \\ l,s \neq i}}^{N} \frac{1}{(x_i - x_l)(x_i - x_s)} & \text{if } j = i. \end{cases}$$

So,

$$\mathcal{U}_{xx} = \left(u_{xx}(x_i, y_q)\right)_{i,q=-M,\ldots,N} = \mathcal{M}_1 \mathcal{U},$$

where $\mathcal{M}_1$ is a $n^2 \times n^2$ matrix defined as,

$$\mathcal{M}_1 = \begin{cases} b_{ji} & k = q \wedge i,\ j,\ k,\ q = -M,\ \ldots,\ N \\ 0 & k \neq q \wedge i,\ j,\ k,\ q = -M,\ \ldots,\ N, \end{cases}$$

and where $\mathcal{U}_{xx}$ is collected in a vector of of length n^2. Likewise, it holds that

$$\mathcal{U}_{yy} = \left(u_{yy}(x_i, y_q)\right)_{i,q=-M,\ldots,N} = \mathcal{M}_2 \mathcal{U},$$

where $\mathcal{M}_2$ is defined in the same way as $\mathcal{M}_1$.

The differential equation (22) has been transformed to a system of n^2 algebraic equations,

$$\mathcal{A}\mathcal{U} = \mathcal{F},$$

where $\mathcal{U}$ is the vector of length n^2 including the unknowns $u_{iq} = u(x_i, y_q)$ and

$$\mathcal{A} = \mathcal{M}_1 + \mathcal{M}_2.$$

The right hand side $\mathcal{F}$ is a vector of Length n^2 and defined as

$$\mathcal{F} = f(x_i,\ y_q),\ i,\ q = -M, \ldots, N.$$

Now the PDE (22) was transformed to a system of n^2 algebraic equations in n^2 unknowns. The boundary conditions are collocated separately to yield $4n$ algebraic equations. More precisely,

$$\begin{aligned} u(a, y_j) &= u_{ex}(a, y_j) \\ u(b, y_j) &= u_{ex}(b, y_j) \\ u(x_i, c) &= u_{ex}(x_i, c) \\ u(x_i, d) &= u_{ex}(x_i, d), \end{aligned}$$

where x_i and y_j are the Sinc points defined on (a, b) and (c, d), respectively. Adding these $4n$ equations to the $n^2 \times n^2$ algebraic system, produced from the collocation of the PDE, yields a system of linear equations given by a rectangular matrix. Finally, solving this least squares problem produces the desired numerical solution.

Notes:

- In our calculations, we multiplied the algebraic equations associated to the boundary conditions by a factor $\tau = 10^3$. This scaling emphasizes the boundary values and improve the error behavior at the boundaries.
- The Poly-Sinc collocation technique is based on the collocation of the spatial variables using Sinc points. This means that it is valid also for PDEs with space-dependent coefficients. Moreover, it can be generalized to solve a system of PDEs.

5 Numerical Results

In this section, we present the computational results. Mainly, we discuss two examples. The first simple example includes one stochastic parameter. In the second example we solve the model problem introduced in Sect. 2.

5.1 One Stochastic Variable

Consider the Poisson equation in two spatial dimensions with one random parameter. This problem is described by the following SPDE

$$\begin{aligned} a(\xi)\left(u_{xx}(x,y,\xi)+u_{yy}(x,y,\xi)\right) &= f(x,y) \ \text{ on } Q\times\Omega \\ u(x,y,\xi) &= 0 \ \text{ on } \partial Q\times\Omega, \end{aligned} \tag{23}$$

where $Q=(-1,1)^2$ is the spatial domain and Ω is an event space and $\xi:\Omega\to[-1,1]$ is a random variable. The function $a(\xi)=\xi+2$ is a linear function of a uniformly distributed random variable ξ and $f(x,y)=1$ for all $(x,y)\in Q$.

Now, we use the PC representation in (4) with $m=3$ to have

$$u(x,y,\xi)=\sum_{i=0}^{3} u_i(x,y)\,\Phi_i(\xi), \tag{24}$$

where Φ_i's are the univariate orthonormal Legendre polynomials defined on $[-1,1]$. Substitution of (24) in the SPDE (23) yields the residual

$$R=(\xi+2)\sum_{i=0}^{3}\left((u_i)_{xx}+(u_i)_{yy}\right)\,\Phi_i(\xi)-1.$$

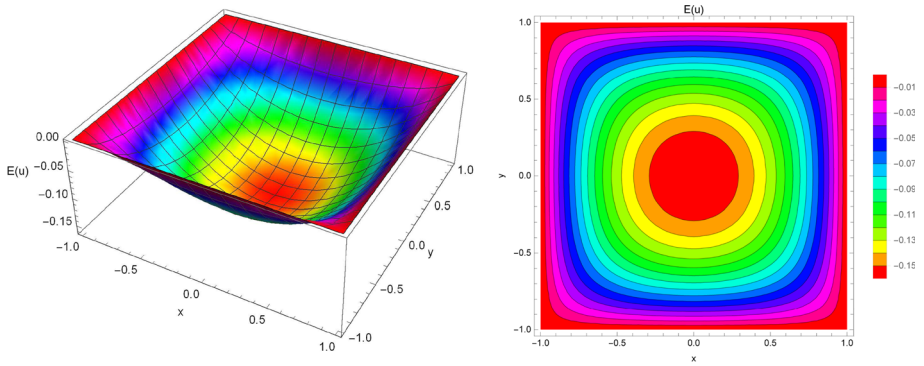
We then perform a Galerkin projection and use the orthogonality of Legendre polynomials, which yields the system of elliptic PDEs

$$\begin{aligned} \sum_{i=0}^{3}\langle(\xi+2)\,\Phi_i,\ \Phi_k\rangle\mathcal{L}u_i &= \langle 1,\Phi_k\rangle \ \text{ for } k=0,1,2,3, \ \text{ on } Q \\ u_i &= 0 \text{ for } i=0,1,2,3 \text{ on } \partial Q, \end{aligned} \tag{25}$$

where $\mathcal{L}u_i=(u_i)_{xx}+(u_i)_{yy}$. It holds that $\langle 1,\Phi_k\rangle=\delta_{1k}$.
The computational results of this example are given in the following experiments.

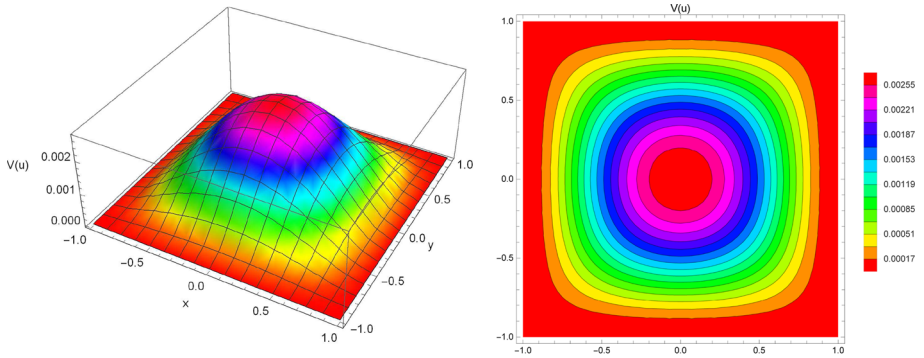
Experiment 1 E(u) and V(u) *In this experiment, we use Poly-Sinc collocation from Sect. 4 to solve the system of PDEs in (25). In our computation, we use $N=5$, i.e. 11×11 of 2D grid of Sinc points defined on the domain Q. As a result of the Poly-Sinc solution, the coefficient functions $u_i(x,y)$ are obtained. In Fig. 1, the expectation $E(u)=u_0(x,y)$ and its contour plot are represented while in Fig. 2, the variance calculations are presented.*

Experiment 2 Coefficients functions *As we mentioned above, to get an accurate result, just a small number of orthogonal polynomials, Φ_i, is needed. In our computations, we used $m=3$, i.e. The four coefficient orthonormal Legendre polynomials. The 4 coefficients functions, $u_i(x,y)$, $i=0,\ldots,3$, are given in Fig. 3. In addition, we verify that this number is sufficient by showing that the coefficient functions u_i tend to zero as m increases. The*



(a) $E(u)$. **(b)** Contour plot of $E(u)$

Fig. 1 The expectation, $E(u)$, using $m = 3$ and Poly-Sinc with $N = 5$



(a) $V(u)$. **(b)** Contour plot of $V(u)$

Fig. 2 The variance, $V(u)$, using $m = 3$ and Poly-Sinc with $N = 5$

results are given in Fig. 4. In Fig. 4, the dots represent the maximum of the coefficient functions $u_i(x, y)$ on the spatial domain. We then use these maximum values in a least square estimation to find the coefficients of the decaying rate function $\alpha \exp(-\beta i)$, where α and β are constants. In Fig. 4, the solid line represents the best fitting function with $\alpha = 0.135$ and $\beta = 1.2$. This means that the coefficient functions $u_i(x, y)$ follow an exponentially decay relation.

Experiment 3 Error *To discuss the convergence of Poly-Sinc solution, we need a reference (nearly exact) solution. For that, we create a discrete list of PDEs of the equation (23) at a finite set of instances of $\xi \in [-1, 1]$. We choose 100 points of Gauss-Legendre nodes as values of $\xi \in [-1, 1]$ and create corresponding 100 PDEs. To solve each one of these 100 equations we use Mathematica Package NDSolve. NDSolve uses a combination of highly accurate numeric schemes to solve initial and boundary value PDEs [1]. We then calculate the expectation and variance of the solutions of our set of boundary value problems of PDEs. In Fig. 5, the errors in the calculations of $E(u)$ and $V(u)$ using $m = 3$ (with orthonormal*

[1] For more information about NDSolve, see Wolfram documentation center at https://reference.wolfram.com/language/ref/NDSolve.html

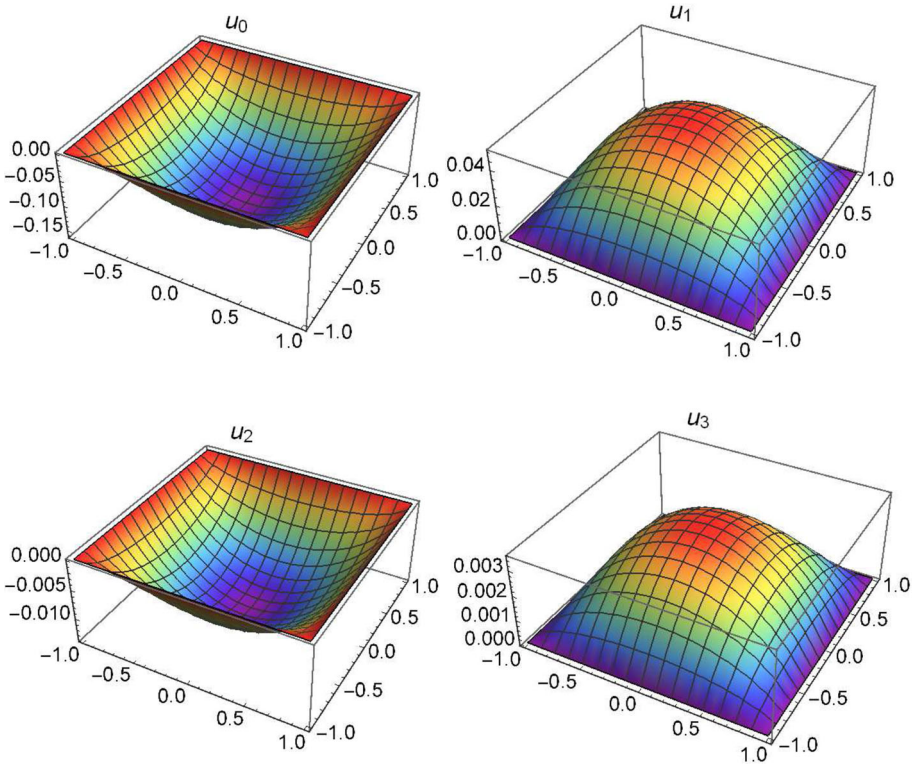


Fig. 3 Coefficients functions, $u_i(x, y),\ i = 0, \ldots, 3$

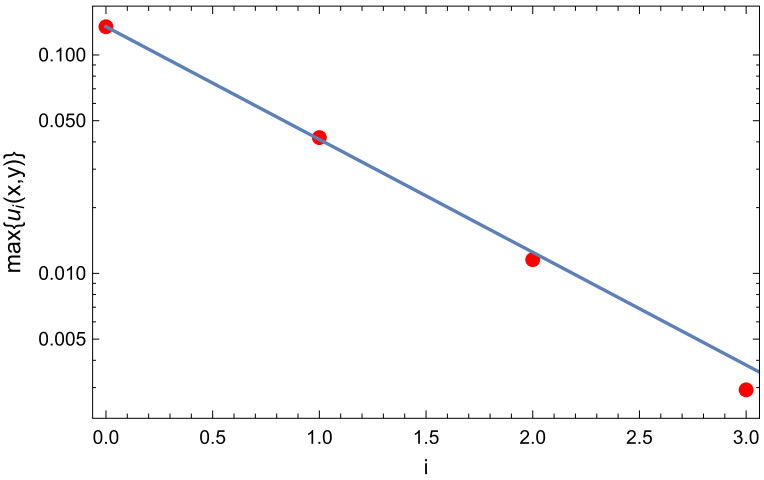
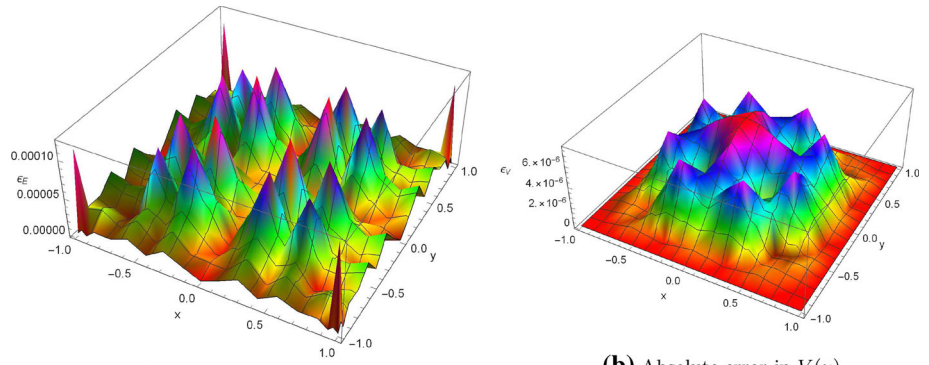


Fig. 4 Logarithmic plot of maximum of coefficient functions $u_i,\ i = 0, \ldots, 3$. The dots are the calculated maximum and the solid line represent the exponential fitting function $0.135\, e^{-1.2\, i}$



(a) Absolute error in $E(u)$. **(b)** Absolute error in $V(u)$

Fig. 5 Absolute error between the Poly-Sinc calculation and the calculations obtained from 100 solutions

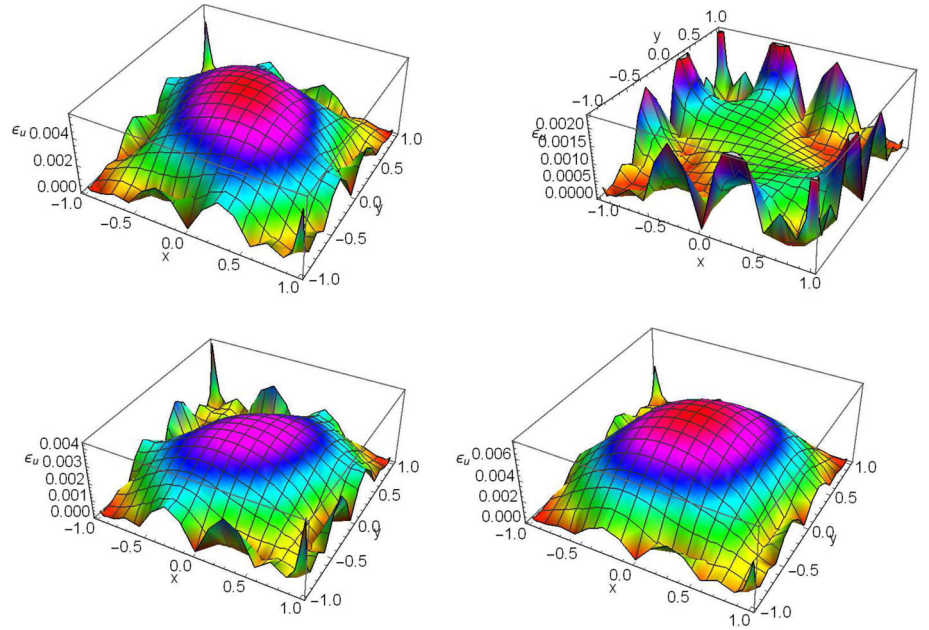
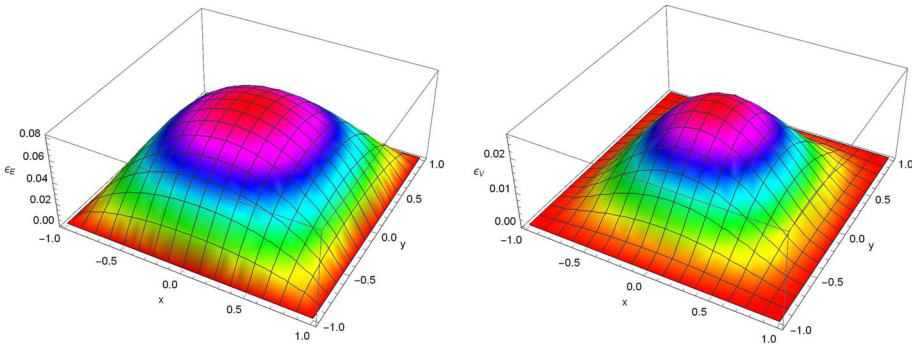


Fig. 6 Absolute error in u for some discrete $\xi \in \{-0.757, 0, 0.757, 0.989\}$

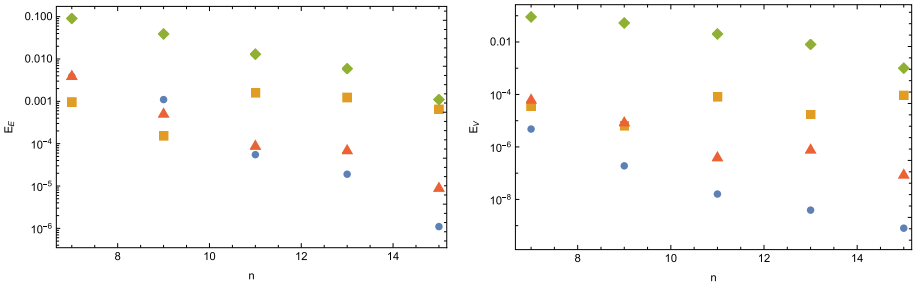
Legendre polynomials) and Poly-Sinc and the references from the 100 *PDEs are presented. Using the spatial L_2-norm error, calculating the error in both $E(u)$ and $V(u)$ delivers error of order $\mathcal{O}(10^{-4})$ and $\mathcal{O}(10^{-6})$, respectively. In Fig. 6, the error between the solution of the SPDE in (23), using the method in this paper, and the reference solution is presented. We choose four instances of ξ.*

Experiment 4 Comparison *In this experiment we compare the Poly-Sinc solution with the classical finite difference (FD) solution. In 5-point-star FD method [28], we use an* 11×11 *meshing with constant step size for the spatial variables x and y, which is the same number of Sinc points used in the Poly-Sinc solution. The error between finite difference solution and*



(a) Absolute error in $E(u)$. **(b)** Absolute error in $V(u)$

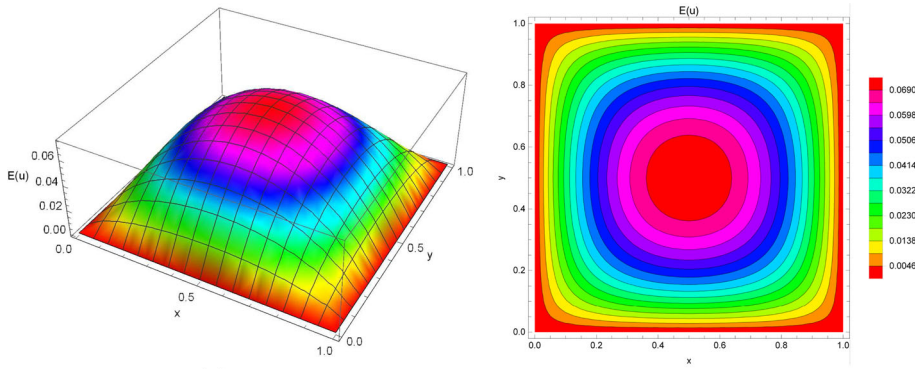
Fig. 7 Absolute error between the FD calculation and the calculations obtained from 100 solutions



(a) L_2 error in $E(u)$. **(b)** L_2 error in $V(u)$

Fig. 8 Results for Poly-Sinc method (dots), FD method with uniform meshes (rhombus), Poly-Chebyshev (squares) and, Poly-Legendre (triangles)

the reference exact solution is given in Fig. 7. Using the spatial L_2-norm error, calculating the error in both $E(u)$ and $V(u)$ delivering error of order $\mathcal{O}(10^{-2})$. These calculations shows that for the same number of points, Poly-Sinc delivers better approximation for the solution of the SPDE. In Fig. 8 we run the calculations for different numbers of Sinc points $n = 2N + 1$ and use the same number of points in the FD method. In addition, we use Chebyshev and Legengdre points as interpolation points in Lagrange polynomials (Poly-Chebyshev and Poly-Legendre) and as collocation points to solve the system of the PDEs. We then calculate the L_2-norm errors. Figure 8 shows the decaying rates of the error, in both mean and variance, using Poly-Sinc, FD method, Poly-Chebyshev, and Poly-Legendre. We can see that the decaying rate of Poly-Sinc is better than the other methods. The comparison between Sinc points and Chebyshev points confirms the results in [23] and [25] that while increasing the number of interpolation points, Sinc points are creating less error and more stable approximation than Chebyshev points. Moreover, in [29] it has been shown that the entries of the Chebyshev differentiation matrix can be computed by explicit formulas. Some of these formulas are numerically unstable (from practical point of view) as the number of Chebyshev points increases. This might cause ill conditioned matrix or non-uniform error at the boundaries. This error affecting the overall solution of the system of the PDEs.



(a) $E(u)$. **(b)** Contour plot of $E(u)$

Fig. 9 The expectation, $E(u)$, using $K = 5,\ P = 3$ and Poly-Sinc with $N = 5$

5.2 Multiple Stochastic Variables

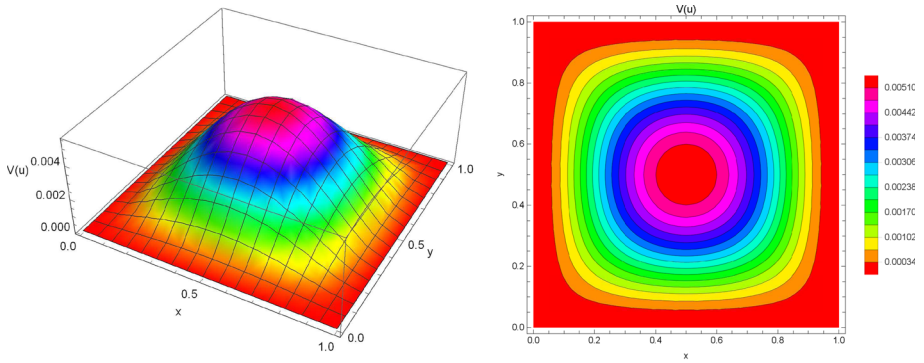
We solve the model problem defined in Sect. 2 for five stochastic variables, cf. [30]. Consider the SPDE defined in (1) with $K = 5$ in (2) and where,

$$
\begin{aligned}
a_1(x, y) &= \frac{1}{4}\cos(2\pi x)\\
a_2(x, y) &= \frac{1}{4}\cos(2\pi y)\\
a_3(x, y) &= \frac{1}{16}\cos(4\pi x)\\
a_4(x, y) &= \frac{1}{16}\cos(4\pi y)\\
a_5(x, y) &= \frac{1}{8}\cos(2\pi x)\cos(2\pi y).
\end{aligned}
$$

$\Theta = \{\xi_k\}_{k=1}^{5}$ is a set of independent random variables uniformly distributed in $[-1, 1]$ and $Q = (0, 1)$. For this SPDE we run four experiments.

Experiment 5 *E(u) and V(u) In this experiment, we perform the Galerkin method along side the multivariate PC. For the PC parameters, we choose* $K = 5$ *and* $P = 3$. *Due to (8), the number of multivariate Legendre polynomials is* $m + 1 = 56$. *As a result the three-dimensional array* $\langle \xi_k \Phi_i(\Theta), \Phi_j(\Theta) \rangle$ *is of dimension* $5 \times 56 \times 56$. *For the Poly-Sinc solution of the resulting system of PDEs, we use* $N = 5$, *i.e.* $n = 11$ *Sinc points. In Figs. 9 and 10 the expectation and variance plots are presented.*

Experiment 6 Coefficients Functions *Similar to the second experiment in Example 1, we would like to study the accuracy of the polynomial expansion. In other words, study the decaying rate, to zero, of these functions. In Fig. 11, the first six coefficients functions of the Poly-Sinc solution are given. These six coefficient functions are associated to the basis polynomials of degree zero and one. In Fig. 12, the logarithmic plot of the maximum of the absolute value of the coefficient functions* $u_{i-1}(x, y),\ i = 1, \ldots, 56$ *on the spatial domain is presented. We can see the fast decaying rate to zero.*



(a) $V(u)$. **(b)** Contour plot of $V(u)$

Fig. 10 The variance, $V(u)$, using $K = 5$, $P = 3$ and Poly-Sinc with $N = 5$

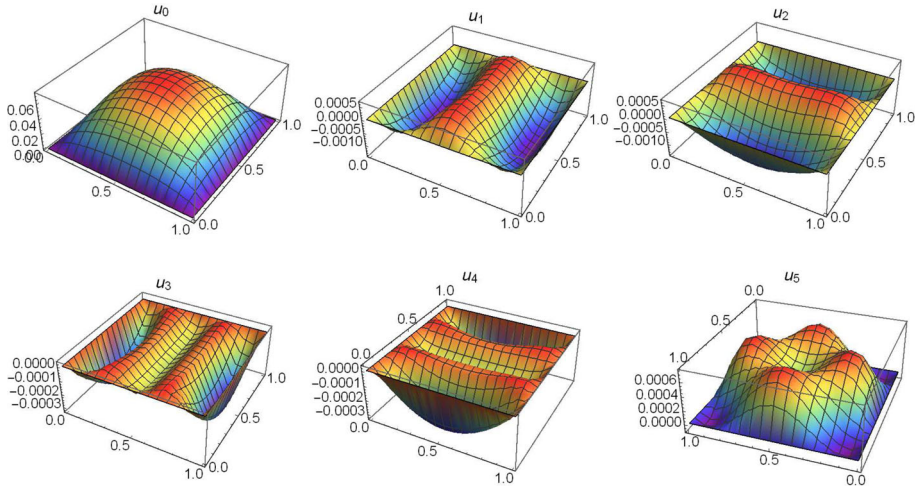


Fig. 11 Coefficients functions $u_i(x, y)$, $i = 0, 1, \ldots, 5$

Experiment 7 Error *The idea of creating a set of (exact) instance solutions we used in the previous example is not applicable here as we have a set of* 5 *random variables. For that we need to find a different reference to check the accuracy of our solution. We use the Finite Element (FE) solution with cell meshing* 10^{-3} *to solve the stochastic Galerkin system of PDEs. The FE element method is a part of the package NDSolve"FEM" in Mathematica 11 that uses the rectangular meshing of the domain and Dirichlet boundary conditions* [2]*. In Fig. 13, the error for the expectation and variance is presented. Using the* L_2*-norm error, calculating the errors in both* $E(u)$ *and* $V(u)$ *delivers errors of order* $\mathcal{O}(10^{-4})$ *and* $\mathcal{O}(10^{-8})$*, respectively.*

Experiment 8 Comparison *In this experiment we compare the Poly-Sinc solution with the 5-point-star FD method. The reference solution is the Finite Element (FE) solution with*

[2] For more information about NDSolve "FEM", see Wolfram documentation center at https://reference.wolfram.com/language/FEMDocumentation/guide/FiniteElementMethodGuide.html

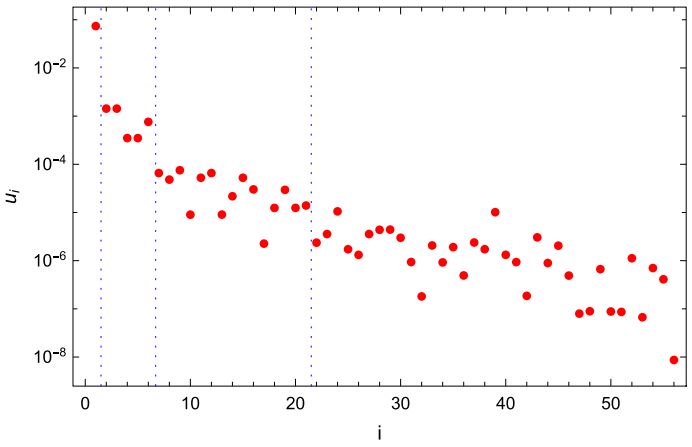
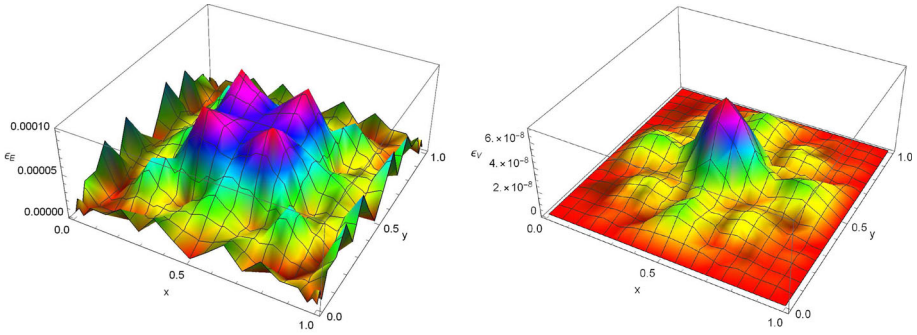
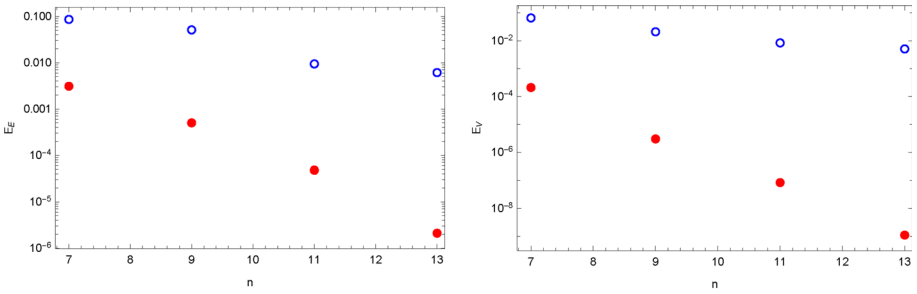


Fig. 12 Logarithmic plot of maximum of coefficient functions $u_{i-1}(x, y)$ for $i = 1, \ldots, 56$. The dotted lines separate the degrees of basis polynomials



(a) Absolute error in $E(u)$. **(b)** Absolute error in $V(u)$

Fig. 13 Absolute error between the Poly-Sinc calculation and the FE



(a) L_2 error in $E(u)$. **(b)** L_2 error in $V(u)$

Fig. 14 Spatial L_2-error. The red dots for Poly-Sinc calculations and the blue circles for FD method (Color figure online)

cell meshing 10^{-3}. *In Fig. 14 we run the calculations for different numbers of Sinc points* $n = 2N + 1$ *and use the same number of points in FD. We then calculate the* L_2*-norm error. These calculations show that the decaying rate of the error, in both mean and variance, is better in Poly-Sinc than the FD method. Moreover, the Poly-Sinc decaying rates of errors are following qualitatively the exponential decaying rate in (21).*

6 Conclusion

In this work we have formulated an efficient and accurate collocation scheme for solving a system of elliptic PDEs resulting from an SPDE. The idea of the scheme is to use a small number of collocation points to solve a large system of PDEs. We introduced the collocation theorem based on the error rate and the Lebesgue constant of the 2D Poly-Sinc approximation. As applications, we discussed two examples, the first example with one random variable while the other with five random variables. For each case the expectation, variance, and error are discussed. The experiments show that using Poly-Sinc approximation to solve the system of PDEs is an efficient method. The number of Sinc points needed to get this accuracy is small and the error decays faster than in the classical techniques, as the finite difference method.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The data in this paper are results of numerical computations and not experimental measurements. So, data sharing not applicable.

References

1. Adler, R.J.: The Geometry of Random Fields. Wiley, Chichester (1981)
2. Theting, T.G.: Solving Wick-stochastic boundary value problems using a finite element method. Stoch. Stoch. Rep. **70**(3–4), 241–270 (2000)
3. Caflisch, R.E.: Monte Carlo and quasi-Monte Carlo methods. Acta Numer. **7**, 1–49 (1998)
4. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. SIAM, Philadelphia (1992)
5. Kleiber, M., Hien, T.D.: The Stochastic Finite Element Method. Basic Perturbation Technique and Computer Implementation. Wiley, Chichester (1992)
6. Babuska, I., Chatzipantelidis, P.: On solving elliptic stochastic partial differential equations. Comput. Methods Appl. Mech. Eng. **191**, 4093–4122 (2002)
7. Ghanem, R., Spanos, P.D.: Stochastic Finite Elements: A Spectral Approach. Springer, Berlin (1991)
8. Ghanem, R.: Ingredients for a general purpose stochastic finite elements implementation. Comput. Methods Appl. Mech. Eng. **168**, 19–34 (1999)
9. Xiu, D., Karniadakis, G.E.: The Wiener–Askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput. **24**, 619–644 (2002)
10. Pulch, R.: Stochastic collocation and stochastic Galerkin methods for linear differential algebraic equations. J. Comput. Appl. Math. **262**, 281–291 (2014)

11. Xiu, D., Lucor, D., Su, C.-H., Karniadakis, G.E.: Stochastic modeling of flow-structure interactions using generalized polynomial chaos. ASME J. Fluid Eng. **124**, 51–69 (2002)
12. Øksendal, Bernt: Stochastic Differential Equations, 6th edn. Springer, Berlin (2003)
13. Pulch, R.: Polynomial chaos for boundary value problems of dynamical systems. Appl. Numer. Math. **62**(10), 1477–1490 (2012)
14. Youssef, M.: Poly-Sinc Approximation Methods. Ph.D. thesis, Mathematics Department German University in Cairo (2017)
15. Youssef, M., Baumann, G.: Collocation method to solve elliptic equations, bivariate poly-sinc approximation. J Prog. Res. Math. (JPRM) **7**(3), 1079–1091 (2016)
16. Youssef, M.: Poly-Sinc collocation method for solving coupled burgers' equations with a large reynolds number. In: Baumann, G (Eds) New Methods of Numerical Analysis. Springer (2019)
17. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press, Princeton (2010)
18. Witteveen, J.A.S., Bijl, H.: Modeling arbitrary uncertainties using Gram–Schmidt Polynomial Chaos. In: Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit, Number AIAA-2006-0896, Reno, pp. 9–12. NV, USA (2006)
19. Eldred, M.S.: Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design. In:: Proceedings of the 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, CA, USA, 4–7 May 2009
20. Pulch, R., Xiu, D.: Generalised polynomial chaos for a class of linear conservation laws. J. Sci. Comput. **51**(2), 293–312 (2012)
21. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis. Springer, New York (2002)
22. Stenger, F.: Handbook of Sinc Methods. CRC Press, Boca Raton (2011)
23. Stenger, F., Youssef, M., Niebsch, J.: Improved Approximation via Use of Transformations. In: Shen, X., Zayed, A.I. (Eds), Multiscale Signal Analysis and Modeling. NewYork: Springer, pp. 25–49 (2013)
24. Youssef, M., El-Sharkawy, H.A., Baumann, G.: Multivariate poly-sinc approximation, error estimation and lebesgue constant. J. Math. Res. (2016). https://doi.org/10.5539/jmr.v8n4p118
25. Youssef, M., El-Sharkawy, H.A., Baumann, G.: Lebesgue constant using sinc points. Adv. Numer. Anal. Article ID 6758283, 10 pages (2016). https://doi.org/10.1155/2016/6758283
26. Youssef, M., Baumann, G.: On bivariate Poly-Sinc collocation applied to patching domain decomposition. Appl. Math. Sci. **11**(5), 209–226 (2017)
27. Youssef, M., Baumann, G.: Troesch's problem solved by Sinc methods. Math. Comput. Simul. **162**, 31–44 (2019). https://doi.org/10.1016/j.matcom.2019.01.003
28. Grossmann, Ch., Roos, H.-G., Stynes, M.: Numerical Treatment of Partial Differential Equations. Springer, Berlin (2007)
29. Trefethen, L.N.: Spectral Methods in MATLAB. SIAM, Philadelphia (2006)
30. Gittelson, C.J.: An adaptive stochastic Galerkin method for random elliptic operators. Math. Comput. **82**(283), 1515–1541 (2013)