Stochastic Collocation Methods: A Survey

Dongbin Xiu

Contents

1 Introduction 2
2 Definition of Stochastic Collocation 3
3 Stochastic Collocation via Interpolation 4
3.1 Formulation 4
3.2 Interpolation SC on Structured Samples 6
3.3 Interpolation on Unstructured Samples 9
4 Stochastic Collocation via Regression 11
4.1 Over-sampled Case: Least Squares 11
4.2 Under-sampled Case: Sparse Approximations 12
5 Stochastic Collocation via Pseudo Projection 13
6 Summary 16
References 16

Abstract

Stochastic collocation (SC) has become one of the major computational tools for uncertainty quantification. Its primary advantage lies in its ease of implementation. To carry out SC, one needs only a reliable deterministic simulation code that can be run repetitively at different parameter values. And yet, the modern-day SC methods can retain the high-order accuracy properties enjoyed by most of other methods. This is accomplished by utilizing the large amount of literature in the classical approximation theory. Here we survey the major approaches in SC. In particular, we focus on a few well-established approaches: interpolation, regression, and pseudo projection. We present the basic formulations of these

D. Xiu (✉)
Department of Mathematics and Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT, USA
e-mail: dongbin.xiu@utah.edu

R. Ghanem et al. (eds.), *Handbook of Uncertainty Quantification*,
DOI 10.1007/978-3-319-11259-6_26-1

approaches and some of their major variations. Representative examples are also provided to illustrate their major properties.

Keywords
Compressed sensing • Interpolation • Least squares • Stochastic collocation

1 Introduction

Stochastic collocation (SC) is a sampling-based method. The term "collocation" originates from the deterministic numerical methods for differential equations, where one seeks to satisfy the governing continuous equations discretely at a set of *collocation points*. This is to the contrary of Galerkin method, where one seeks to satisfy the governing equation in a weak form. *Stochastic collocation* was first termed in [31], although the idea and its application have existed long before that.

To illustrate the idea, let us consider, for a spatial domain D and time domain $[0, T]$ with $T > 0$, the following partial differential equation (PDE) system:

$$\begin{cases} u_t(x, t, Z) = \mathcal{L}(u), & D \times (0, T] \times I_Z, \\ \mathcal{B}(u) = 0, & \partial D \times [0, T] \times I_Z, \\ u = u_0, & D \times \{t = 0\} \times I_Z, \end{cases} \tag{1}$$

where $I_Z \subset \mathbb{R}^d$, $d \geq 1$ is the support of the uncertain parameters $Z = (Z_1, \ldots, Z_d)$ and $\mathcal{B}$ is the boundary condition operator. The solution is then a mapping

$$u(x, t, Z) : \bar{D} \times [0, T] \times I_Z \to \mathbb{R},$$

where for the simplicity of exposition we consider a scalar equation. In UQ computations, we are primarily interested in the solution dependence in the parameter space, that is,

$$u(\cdot, Z) : I_Z \to \mathbb{R}, \tag{2}$$

where the dependence on the spatial and temporal variables (x, t) is suppressed. Hereafter, all statements are made for any fixed x and t.

In stochastic collocation, the system (1) is solved in a discrete manner. More specifically, we seek to enforce the equation at a discrete set of nodes – "collocation points." Let $\Theta_M = \{Z^{(j)}\}_{j=1}^M \subset I_Z$ be a set of (prescribed) nodes in the random space, where $M \geq 1$ is the number of nodes. Then in SC, we enforce (1) at the node $Z^{(j)}$, for all $j = 1, \ldots, M$, by solving

$$\begin{cases} u_t(x, t, Z^{(j)}) = \mathcal{L}(u), & D \times (0, T], \\ \mathcal{B}(u) = 0, & \partial D \times [0, T], \\ u = u_0, & D \times \{t = 0\}. \end{cases} \tag{3}$$

It is easy to see that for each j, (3) is a deterministic problem because the value of the random parameter Z is fixed. Therefore, solving the system poses no difficulty provided one has a well-established deterministic algorithm. Let $u^{(j)} = u(\cdot, Z^{(j)})$, $j = 1, \ldots, M$, be the solution of the above problem. The result of solving (3) is an ensemble of deterministic solutions $\{u^{(j)}\}_{j=1}^{M}$. And one can apply various post-processing operations to the ensemble to extract useful information about $u(Z)$.

From this point of view, all classical sampling methods belong to the class of collocation methods. For example, in *Monte Carlo sampling*, the nodal set Θ_M is generated randomly according to the distribution of Z, and the ensemble averages are used to estimate the solution statistics, e.g., mean and variance. In *deterministic sampling methods*, the nodal set is typically the nodes of a cubature rule (i.e., quadrature rule in multidimensional space) defined on I_Z such that one can use the integration rule defined by the cubature to estimate the solution statistics.

In SC, the goal is to construct an accurate approximation to the solution response function using the samples. This is a stronger goal than estimating the solution statistics and the major difference between SC and the classical sampling methods. Knowing the function response of the solution allows us to immediately derive all of the statistical information of the solution. It is not the case conversely, as knowing the solution statistics does not allow us to create the solution response. To this end, SC can be classified as *strong approximation* methods, whereas the traditional sampling methods are *weak approximation* methods. (More precise definitions of strong and weak approximations can be found in [30].)

2 Definition of Stochastic Collocation

The goal of SC is to construct a numerical approximation to the solution response (2) in the parameter space I_Z, using the deterministic solution ensemble $\{u(\cdot, Z^{(j)}\}$, $j = 1, \ldots, M$, of (3). Following [30], we give the following formal definition of SC:

Definition 1 (Stochastic collocation). Let $\Theta_M = \{Z^{(j)}\}_{j=1}^{M} \subset I_Z$ be a set of (prescribed) nodes in the random space I_Z, where $M \geq 1$ is the number of nodes, and $\{u^{(j)}\}_{j=1}^{M}$ be the solution of the governing equation (3). Then find $w(Z) \approx u(Z)$ such that it is an approximation to the true solution $u(Z)$ in the sense that $\|w(Z) - u(Z)\|$ is sufficiently small in a strong norm defined on I_Z.

In this general definition, the norm is left unspecified. In practice, different choices of the norm lead to different SC methods. Typically, we employ L^p-norm ($p \geq 1$), with the L^2-norm used the most in practice and leading to "mean-square" approximation.

The numerical approximation $w(Z)$ shall be chosen from a class of functions. Mathematically speaking, this implies that $w \in V$, where V is a linear space from which the approximation is sought. In SC, the most widely used choice is polynomial space, which leads to strong ties of SC methods to generalized polynomial chaos (gPC) approximation ([16,32]). Here, the space V is

$$\mathbb{P}_n^d = \text{span}\{z^\alpha : |\alpha| = (\alpha_1 + \cdots + \alpha_d) \le n\}, \tag{4}$$

where $\alpha = (\alpha_1, \dots, \alpha_d)$ is multi-index. This is the space of polynomials of degree up to n, whose cardinality is $\dim \mathbb{P}_n^d = \binom{n+d}{n}$. Other spaces of polynomials, or other classes of functions, can certainly be chosen.

The construction and properties of SC methods then critically depend on the approximation properties of w and the choice of the collocation nodal set Θ_M. Broadly speaking, the current SC methods fall into the following three categories: interpolation type, regression type, and pseudo projection type.

3 Stochastic Collocation via Interpolation

3.1 Formulation

In interpolation approach, we seek to match the numerical approximation w with the true solution u exactly at the nodal set Θ_M. More specifically, let $w \in V_N$ be constructed from a linear space V_N with cardinality $\dim V_N = N$. Let $(b_1, \dots, b_N)$ be a basis for V_N. Then, we can express w as

$$w(Z) = \sum_{i=1}^{N} c_i b_i(Z), \tag{5}$$

where c_i are the coefficients to be determined. We then enforce the interpolation condition

$$w(Z^{(j)}) = u(Z^{(j)}), \qquad \text{for all } j = 1, \dots, M. \tag{6}$$

This immediately leads to a linear system of equations for the unknown coefficients

$$\mathbf{A}\mathbf{c} = \mathbf{f}, \tag{7}$$

where

$$\mathbf{A} = (a_{ij})_{1 \le i \le M, 1 \le j \le N}, \qquad a_{ij} = b_j(Z^{(i)}), \tag{8}$$

and

$$\mathbf{c} = (c_1, \dots, c_N)^T, \qquad \mathbf{f} = (u(Z^{(1)}, \dots, u(Z^{(M)}))^T \tag{9}$$

are the coefficient vector and solution sample vector, respectively. For example, if one adopts the gPC expansion, then V_N is the polynomial space $\mathbb{P}_n^d$ from (4), and the matrix $\mathbf{A}$ becomes the Vandermonde-like matrix with entries

$$a_{ij} = \Phi_j(Z^{(i)}), \tag{10}$$

where $\Phi_j(Z)$ are the gPC orthogonal polynomials chosen based on the probability distribution of Z and satisfy

$$\int_{I_Z} \Phi_i(z)\Phi_j(z)\rho(z)dz = \delta_{ij}. \tag{11}$$

Here, δ_{ij} is the Kronecker delta function and the polynomials are normalized.

When the number of the collocation points is the same as the number of the basis functions, i.e., $M = N$, the matrix $\mathbf{A}$ is square and can be inverted when it is nonsingular. One then immediately obtains

$$\mathbf{c} = \mathbf{A}^{-1}\mathbf{f}, \tag{12}$$

and can construct the approximation $w(Z)$ using (5). Although very flexible, this approach is not used widely in practice. The reason is that interpolation is often not very robust and leads to wild behavior in $w(Z)$. This is especially true in multidimensional spaces ($d > 1$). The accuracy of the interpolation is also difficult to assess and control. Even though the interpolation $w(Z)$ has no error at the nodal points, it can incur large errors between the nodes. Rigorous mathematical analysis is also lacking on this front, particularly in high dimensions. Some of these facts are well documented in texts such as [7, 23, 27].

Another approach to accomplish interpolation is to employ the *Lagrange interpolation approach*. That is, we seek

$$w(Z) = \sum_{j=1}^{M} u(Z^{(j)}) L_j(Z), \tag{13}$$

where

$$L_j(Z^{(i)}) = \delta_{ij}, \quad 1 \le i, j \le M, \tag{14}$$

are the Lagrange interpolating polynomials. By construction, the polynomial $w(Z)$ automatically satisfies the interpolation conditions. We then need to explicitly construct the Lagrange interpolation polynomials $L_j(Z)$. This can be easily done in one dimension $d = 1$, i.e.,

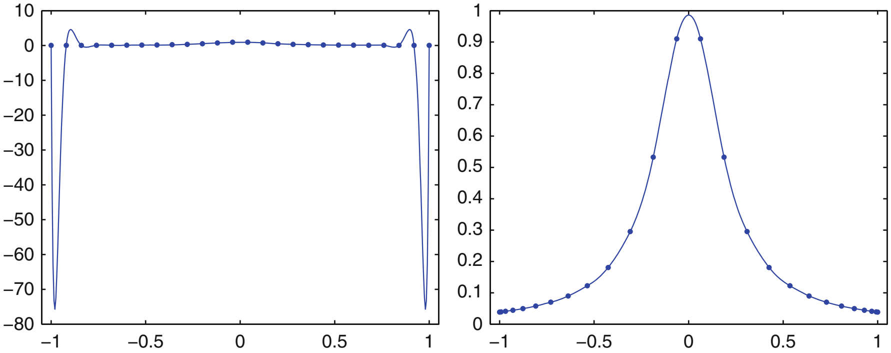

Fig. 1 Polynomial interpolation of $f(x) = 1/(1+25x^2)$ in $[-1, 1]$, the rescaled Runge function. *Left*: interpolation on uniformly distributed nodes; *Right*: interpolation on nonuniform nodes (the zeros of Chebyshev polynomials)

$$L_j(Z) = \prod_{i=1, i\neq j}^{M} \frac{Z - Z^{(i)}}{Z^{(j)} - Z^{(i)}}, \qquad j = 1, \ldots, M. \tag{15}$$

Much is known about polynomial interpolation in one dimension. It is widely acknowledged that interpolation on equidistance grids is unstable at higher degree polynomials. To construct robust and accurate interpolations, one should employ grids that are clustered toward the boundaries of the interval. The well-known example of interpolating the Runge function clearly illustrates this property. The results are shown in Fig. 1. Even though both interpolations can faithfully interpolate the function data, the result by the equidistance nodes admits wild oscillations between the nodes, whereas the result by the Chebyshev nodes is well behaved and accurate.

Interpolation in multiple dimensions ($d > 1$) is usually carried out in two different approaches. The first approach is to extend the well- studied one-dimensional interpolation methods to multiple dimensions via a certain tensor product rule. This naturally results in sampling sets that are structured. An immediate consequence is that the growth of the number of samples in high dimensions can be prohibitively fast – courtesy of the “curse of dimensionality.” The second approach is to directly construct interpolations on a set of unstructured nodes. This, however, is a mathematically challenging task and leaves many open issues to study.

3.2 Interpolation SC on Structured Samples

The major difficulty in the interpolation SC is the construction of interpolation polynomials in multiple dimensions. Traditionally, this is carried out by extending the one-dimensional interpolation techniques (15) to higher dimensions.

3.2.1 Tensor Nodes

Since univariate interpolation is a well-studied topic, it is straightforward to employ a univariate interpolation and then fill up the multidimensional parameter space dimension by dimension. By doing so, the properties and error estimates of univariate interpolation can be retained as much as possible.

Let

$$Q_{m_i}[f] = \sum_{j=1}^{m_i} f(Z_i^{(j)}) L_j(Z_i), \qquad i = 1, \dots, d, \tag{16}$$

be the one-dimensional Lagrange interpolation in the i-th dimension, where L_j are defined in (15) and the number of samples is m_i. Let $\Theta_1^{m_i}$ be the interpolation nodal set in this direction. To extend this into the entire d-dimensional space I_Z, we can use tensor product approach and define the multidimensional interpolation operator as

$$Q_M = Q_{m_1} \otimes \cdots \otimes Q_{m_d}, \tag{17}$$

and the nodal set is

$$\Theta_M = \Theta_1^{m_1} \times \cdots \times \Theta_1^{m_d}, \tag{18}$$

where the total number of nodes is $M = m_1 \times \cdots \times m_d$.

The advantage of this approach is that all the properties of the underlying one-dimensional interpolation scheme can be retained. For example, if one employs the Gauss points as $\Theta_1^{m_i}$, then the interpolation can be highly accurate and robust. The drawback is that the total number of points (18) grows too fast in high dimensions. And the desirable properties of the one-dimensional interpolation will be severely offset by this. For example, let us assume one uses the same number of samples in every dimensions, i.e., $m_1 = \cdots = m_d = m$. Then, the total number of points is $M = m^d$. Let us further assume that the one-dimensional interpolation error in each dimension $1 \le i \le d$ follows

$$(I - Q_{m_i})[f] \propto m^{-\alpha},$$

where the constant $\alpha > 0$ depends on the smoothness of the function f. Then, the overall interpolation error also follows the same convergence rate

$$(I - Q_M)[f] \propto m^{-\alpha}.$$

However, if we measure the convergence in terms of the total number of points, $M = m^d$ in this case, then

$$(I - Q_M)[f] \propto M^{-\alpha/d}, \quad d \ge 1.$$

For large dimensions $d \gg 1$, the rate of convergence deteriorates drastically and we observe very slow convergence, if there is any, in terms of the total number of collocation points. This is the well known *curse of dimensionality*. For this reason, the tensor product construction is mostly used for low-dimensional problems with d typically less than 5. A detailed theoretical analysis for the tensor interpolation SC for stochastic diffusion equations can be found in [2].

3.2.2 Sparse Grids

An alternative approach is Smolyak sparse grids interpolation. This is based on the original work by Smolyak in [25]. It has been studied extensively in various deterministic settings (cf. the reviews in [3, 4] and the references therein) and was first used in UQ computations in [31]. The Smolyak sparse interpolation also relies on the one-dimensional interpolation (16). Instead of taking the full tensor product (17), the Smolyak interpolation takes a subset of the full tensor construction in the following manner (cf. [28]),

$$Q_\ell = \sum_{\ell-d+1\le|\mathbf{i}|\le\ell} (-1)^{\ell-|\mathbf{i}|} \cdot \binom{d-1}{\ell-|\mathbf{i}|} \cdot \left(Q_{i_1} \otimes \cdots \otimes Q_{i_d}\right), \tag{19}$$

where $\ell \ge d$ is an integer denoting the *level* of the construction. Though the expression is rather complex, (19) is nevertheless a combination of the subsets of the full tensor construction. The nodal set, the *sparse grids*, is

$$\Theta_M = \bigcup_{\ell-d+1\le|\mathbf{i}|\le\ell} (\Theta_1^{i_1} \times \cdots \times \Theta_1^{i_d}). \tag{20}$$

Again it is clear that this is the union of a collection of subsets of the full tensor grids. Unfortunately there is usually no explicit formula to determine the total number of nodes M in terms of d and ℓ.

One popular choice of sparse grids is based on Clenshaw-Curtis nodes, which are the extrema of the Chebyshev polynomials and are defined as, for any $1 \le i \le d$,

$$Z_i^{(j)} = -\cos\frac{\pi(j-1)}{m_i^k - 1}, \qquad j = 1, \ldots, m_i^k, \tag{21}$$

where an additional index k is introduced to indicate the *level* of the Clenshaw-Curtis nodes. The number of points doubles with the increasing index $k > 1$, $m_i^k = 2^{k-1} + 1$, where we define $m_i^1 = 1$ and $Z_i^{(1)} = 0$. By doing so, the Clenshaw-Curtis nodes are nested, a property strongly preferred in the Smolyak construction. (For a more detailed discussion on the Clenshaw-Curtis nodes, see [14].) The total number of points satisfies the following estimate

$$M = \#\Theta_k \sim 2^k d^k / k!, \qquad d \gg 1. \tag{22}$$

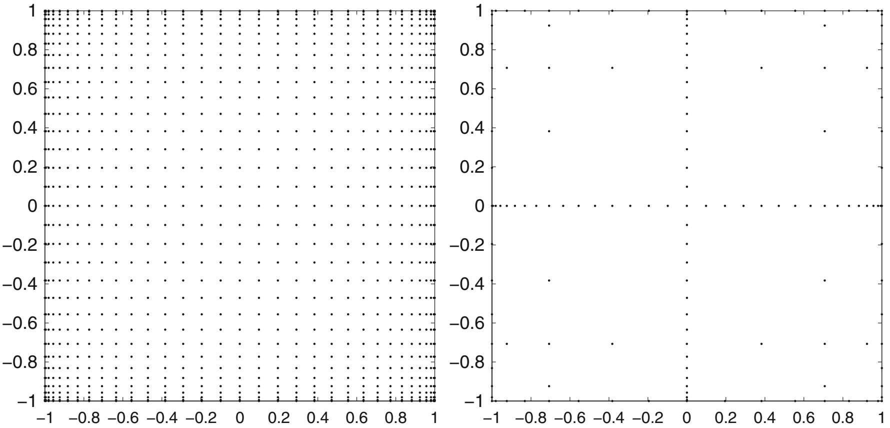

Fig. 2 Two-dimensional ($d = 2$) nodes based on the same one-dimensional extrema of Chebyshev polynomials at level $k = 5$. *Left*: Tensor grids. The total number of points is $1,089$. *Right*: Smolyak sparse grids. The total number of nodes is 145

It has been shown ([3]) that the interpolation through the Clenshaw-Curtis sparse grid interpolation is exact if the function is in $\mathbb{P}_k^d$. (In fact, the polynomial space for which the interpolation is exact is slightly bigger than $\mathbb{P}_k^d$.) For large dimensions $d \gg 1$, $\dim \mathbb{P}_k^d = \binom{d+k}{d} \sim d^k/k!$. Therefore, the number of points from (22) is about 2^k more and the factor is *independent* of the dimension d. For this reason, the Clenshaw-Curtis-based sparse grid construction is sometimes regarded as optimal in high dimensions.

An example of the two-dimensional tensor grids and sparse grids is in Fig. 2, where we observe significant reduction of the number of nodes in sparse grids. The reduction becomes more obvious in Table 1, where the number of samples of the sparse grids and tensor grids is listed for various dimensions d and polynomial degree k. We clearly see the drastic reduction in the number of samples for sparse grids, compared to that of the tensor grids. We observe the 2^k factor between the number of samples in sparse grids and the cardinality of the polynomial space $\mathbb{P}_k^d$ in high dimensions $d \gg 1$.

Even though the sparse grids enjoy great reduction in the number of sample points, one should be aware that the total number of points can still be exceedingly large at high dimensions. The estimate for Clenshaw-Curtis (22) is almost the best, as all other types of sparse grid constructions have (much) larger number of points. To this end, sparse grid interpolation has been used for moderately high dimensions, say, $d \sim O(10)$.

3.3 Interpolation on Unstructured Samples

From a practical point of view, it is highly desirable to conduct multidimensional interpolation on an arbitrary set of nodes Θ_M. To this end, however, it becomes much less clear what the best approach shall be. Arguably the only existing approach

Table 1 The number of samples of the Smolyak sparse grids using Clenshaw-Curtis nodes, the cardinality of the polynomial space $\mathbb{P}_k^d$, and the number of samples of the full tensor grids $(k+1)^d$, for various dimensions d and polynomial order k (This is a reproduction of the Table 3.1 from [31])

d	n	Sparse grids	$\dim(\mathbb{P}_n^d)$	Tensor grids
2	1	5	3	4
	2	13	6	9
	3	29	10	16
	4	65	15	25
10	1	21	11	1,024
	2	221	66	59,049
	3	1,581	286	1,048,576
20	1	41	21	1,048,576
	2	841	231	$\approx 3.5 \times 10^9$
50	1	101	51	$\approx 1.1 \times 10^{16}$
	2	5,101	1,326	$\approx 7.2 \times 10^{23}$

is least interpolation. This was first developed in [10, 11] and later extended to general orthogonal polynomials in [21]. The mathematical theory of this approach is rather involved and technical, although its implementation is straightforward using only numerical linear algebra. The major advantage of this approach is that one can then conduct SC interpolation on nested samples and, more importantly, samples that are arbitrarily distributed. This is especially useful from a practical point of view, for in many applications the samples are collected by practitioners at locations not following certain mathematical theory. Not to mention that in many cases there are restrictions on where one can or cannot collect the samples. Robust interpolation using this approach can be achieved by carefully designed sample sets [22]. An example of its effectiveness can be seen from simple interpolations of the following one-dimensional functions with increasing smoothness:

$$f_0(x) = \begin{cases} -1, & x < \frac{1}{2}, \\ 1, & x \geq \frac{1}{2}, \end{cases} \qquad f_s(x) = \int_{-1}^{x} f_{s-1}(t)dx, \qquad s = 1, 2, 3. \tag{23}$$

The results are shown in Fig. 3. We observe almost the same convergence rate and errors in both the least interpolation and the standard interpolation using Gauss-Legendre nodes. The fundamental difference between the two methods is that the least interpolation is conducted on completely nested sample nodes, allowing one to progressively add sample points for improved accuracy. On the other hand, the standard Gauss node interpolation is not nested – increasing the accuracy of the interpolation implies sampling at a new set of points.

Again, the theory of the least interpolation is quite involved. We refer the interested readers to [10, 11, 21, 22] for more details.

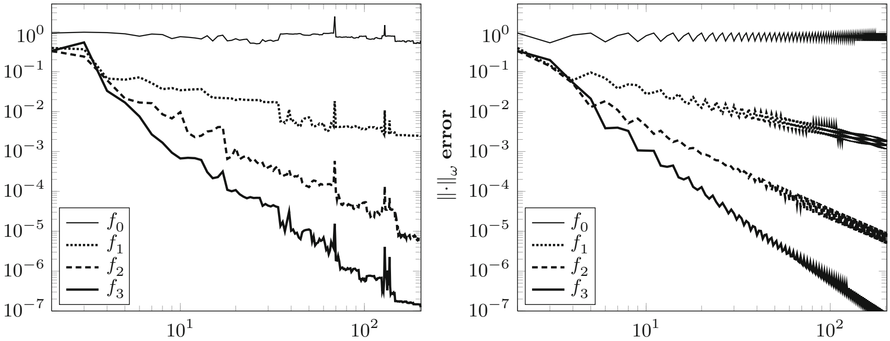

Fig. 3 Interpolation accuracy for functions f_0, f_1, f_2, and f_3 in (23). *Left*: errors by least interpolation; *Right*: errors by interpolation on Legendre-Gauss nodes (Reproduction of Fig. 5.2 in [22])

4 Stochastic Collocation via Regression

In regression type SC, one does not require the approximation $w(Z)$ to precisely match the solution $u(Z)$ at the collocation nodes Θ_M. Instead, one resorts to minimize the error difference

$$\|w(Z) - u(Z)\|_\Theta, \tag{24}$$

where the norm $\|\cdot\|_\Theta$ is a discrete norm defined over the nodal set Θ. By doing so, the numerical errors are more evenly distributed in the entire parameter space I_Z, assuming that the set Θ fills up the space I_Z in a reasonable manner. Thus, the non-robustness of interpolation can be alleviated. The regression approach also becomes a natural choice when the solution samples $u(Z^{(j)})$ are of low accuracy or contaminated by noise, in which case interpolation of the solution samples becomes an unnecessarily strong requirement.

4.1 Over-sampled Case: Least Squares

When the number of samples is larger than the cardinality of the linear space V_N, we have an over-determined system of equations (7) with $M > N$. Consequently, the equality cannot hold true in general. The natural approach is to use the least squares method. By doing so, the norm in (24) is the vector 2-norm, and we have the well-known least squares solution:

$$\mathbf{c} = \mathbf{A}^\dagger \mathbf{f} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{f}, \tag{25}$$

where $\mathbf{A}^\dagger$ denotes the pseudo-inverse of $\mathbf{A}$.

The least squares method is an orthogonal projection onto the range of the matrix $\mathbf{A}$. Consequently, it is the optimal approximation (in vector 2-norm) over the Hilbert space defined by the vector inner product. Over-sampling is the key to the accuracy and efficiency of the least squares method. A rule of thumb for practical problems is that one should over-sample the system by a linear factor, i.e., $M \approx \alpha N$, where $\alpha \sim O(1)$ and is often chosen to be $\alpha = 1.5 \sim 3$. There are a variety of choices for the nodal set Θ_M. The most commonly used sets include Monte Carlo points (random sampling), quasi-Monte Carlo points, etc. It is also worthwhile to strategically choose the points to achieve better accuracy. This is the topic of *experimental design*. Interested readers can refer to, for example, [1, 15], and the references therein.

Despite the large amount of literature on least squares methods, a series of more recent studies from the computational mathematics perspective show that the linear over-sampling of Monte Carlo and quasi-Monte Carlo points for polynomial approximation can be asymptotically unstable (cf. [8, 18–20]). Care must be taken if one intends to conduct very high-order polynomial approximations using the least squares method.

4.2 Under-sampled Case: Sparse Approximations

When the number of samples is smaller than the cardinality of the linear space V_N, we have an under-determined system of equation (7) with $M < N$. Equation (7) then admits an infinite number of solutions. In this case, one can resort to the idea of compressive sensing (CS) and seek a sparse solution:

$$\min \|\mathbf{c}\|_0 \quad \text{subject to } \mathbf{Ac} = \mathbf{f}, \tag{26}$$

where the $\|\mathbf{c}\|_0 = \{\#c_i : c_i \neq 0, i = 1, \dots, N\}$ is the number of nonzero entries in the vector $\mathbf{c}$. The solution of this constrained optimization problem leads to a sparse solution, in the sense that the number of nonzero entries in the solution is minimized. Unfortunately, this optimization is an NP-hard problem and cannot be easily solved. As a compromise, the ℓ_1 norm is often used, leading to the well-known compressive sensing formulation (cf. [5, 6, 12]):

$$\min \|\mathbf{c}\|_1 \quad \text{subject to } \mathbf{Ac} = \mathbf{f}, \tag{27}$$

where $\|\mathbf{c}\|_1 = |c_1| + \cdots + |c_N|$. The use of the ℓ_1 norm also promotes sparsity. But the optimization problem can be cast into a linear programming and solved easily. The constraint $\mathbf{Ac} = \mathbf{f}$ effectively enforces interpolation. This does not need to be the case, especially when the samples $\mathbf{f}$ contain errors or noises. In this case, the de-noising version of CS [5, 6, 12] can be used.

$$\min \|\mathbf{c}\|_1 \quad \text{subject to } \|\mathbf{Ac} - \mathbf{f}\| \leq \tau, \tag{28}$$

where $\tau > 0$ is a real number associated with the noise level in the sample data $\mathbf{f}$.

The idea of CS was first used in UQ computations in [13], where the gPC-type orthogonal approximations using Legendre polynomials were used. The advantage of this approach lies in the fact that it allows one to construct reliable sparse gPC approximations when the underlying system is (severely) under sampled. Most of the existing studies focus on the use of Legendre polynomials [17, 24, 33]. For example, it was proved in [24] that the Chebyshev-weighted ℓ_1 minimization algorithm using Chebyshev samples has notably higher rate of recovery and should be preferred in practice. That is, instead of (27), one solves

$$\min \|\mathbf{c}\|_1 \quad \text{subject to } \mathbf{WAc} = \mathbf{Wf}, \tag{29}$$

where $\mathbf{W}$ is a diagonal matrix with entries $w_{j,j} = (\pi/2)^{d/2} \prod_{i=1}^{d}(1 - (z_i^{(j)})^2)^{1/4}$, $j = 1, \ldots, M$. Note that this is the tensor product of the one-dimensional Chebyshev weights. The corresponding de-noising version for (28) takes a similar form. On the other hand, it was also proved that in high dimensions $d \gg 1$, the standard non-weighted version (27) using uniformly distributed random samples is in fact better than the weighted Chebyshev version (29).

The performance of the ℓ_1 minimization methods is typically measured by the recovery probability of the underlying sparse functions. For high rate of recovery, the required number of samples typically follows:

$$M \propto s \log^3(s) \log(N),$$

where s is the sparsity of the underlying function, i.e., s is the number of nonzero terms in the underlying function. A representative result can be seen in Fig. 4, where the ℓ_1 minimization is used to recover a $d = 3$ dimensional polynomial function with sparsity $s = 10$. Although the different implementations result in variations in the results, we can still see that the methods can recover, with very high probability, the underlying function with $M \sim 100$ samples. This is notably lower than the cardinality of the polynomial space $\dim \mathbb{P}_k^d = 286$. This clearly demonstrates the effectiveness of CS methods. We shall remark that, despite these few works mentioned here, the application of CS in UQ is still in its early stage. There exist many open issues to study.

5 Stochastic Collocation via Pseudo Projection

In the pseudo projection approach (first formally defined in [29]), one seeks to approximate the continuous orthogonal projection using an integration rule. Since the orthogonal projection is the "best approximation," based on a properly chosen norm, the pseudo projection method allows one to obtain a "near best" approximation whenever the chosen integration is sufficiently accurate. Again, let V_N be a properly chosen linear space, from which an approximation is sought. Then, the orthogonal projection of the solution is

$$u_N := \mathcal{P}_{V_N} u, \tag{30}$$

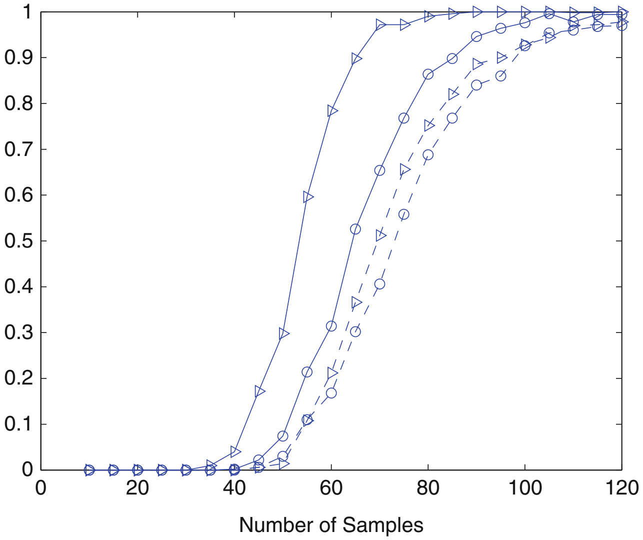

Fig. 4 Probability of successful function recovery *vs.* number of samples. ($d = 3$ and $s = 10$). The degree of polynomial space is $k = 10$ with $\dim \mathbb{P}_k^d = 286$. Line pattern: *dotted-circle*, uniform samples; *dotted-triangle*: Chebyshev samples; *solid-circle*: weighted uniform samples; *solid-triangle*: weighted Chebyshev samples (Reproduction of Fig. 1 in [33])

where $\mathcal{P}$ denotes the orthogonal projection operator. The projection operator is often defined via integrals. In the pseudo projection approach, one then approximates the integrals using a quadrature/cubature rule.

To better illustrate the method, let us again use the gPC-based approximation. In this case, $V_N = \mathbb{P}_n^d$, and we seek an approximation

$$w(Z) = \sum_{i=1}^{N} c_i \Phi_i(Z), \qquad N = \dim \mathbb{P}_n^d = \binom{n+d}{n}. \tag{31}$$

The orthogonal projection, which is the best approximation in L^2_ρ norm, takes the following form,

$$u_N(Z) := \sum_{i=1}^{N} \hat{u}_i \Phi_i(Z), \qquad \hat{u}_i = \int u(z)\Phi_i(z)\rho(z)dz. \tag{32}$$

In the pseudo projection approach, we then use a cubature rule to approximate the coefficients $\hat{u}_i$. That is, in (31), we seek

$$c_i = \sum_{j=1}^{M} u(Z^{(j)})\Phi_i(Z^{(j)})w_j \approx \hat{u}_i, \qquad i = 1, \ldots, N. \tag{33}$$

This means the collocation nodal set Θ needs to be a quadrature such that integrals over the I_Z can be approximated by a weighted sum. That is,

$$\int_{I_Z} f(z)\rho(z)dz \approx \sum_{j=1}^{M} f(Z^{(j)})w_j, \tag{34}$$

where w_j are the weights.

The pseudo projection method turns out to be remarkably robust and accurate, provided one finds an efficient and accurate quadrature rule. Unlike the other approaches for SC, e.g., interpolation SC and regression SC, where the goal is to approximate the underlying multidimensional function u directly, in the pseudo projection approach, the challenge becomes the approximation of multidimensional integrals. The nodal set Θ should now be a good cubature rule. Depending on the problem at hand and the accuracy requirement, one can choose different cubature sets. The field of multidimensional integration is by itself a big and evolving field, with a large amount of literature. Interested readers should consult the literature on cubature rules (cf. [9,26]).

It should also be remarked that an "easier" way to construct cubature rules is to extend the one-dimensional quadrature rule to multiple dimensions via tensor products. A construction is very much similar to the tensor product multidimensional interpolation SC, as discussed in Sect. 3.2. Quadrature rules in one dimension are well studied and understood. It is widely accepted that Gauss quadrature, particularly Chebyshev quadrature, is near optimal. One can then extend it into multiple dimensions either via full tensor products or the Smolyak construction (which is a subset of full tensor products). Upon doing so, one obtains tensor cubature or sparse grid cubature, respectively. An example of this is seen in Fig. 2.

When pseudo projection is used, one should pay attention to the accuracy of the cubature rule. This is because the integrals to be approximated in (32) become progressively more complex, when a higher degree gPC expansion is used. As a general rule of thumb, one should employ a cubature that is accurate with order at least $2n$, where n is the degree of the gPC expansion. This ensures that if the underlying unknown function is an n-degree polynomial (which is almost never the case), then the n-degree gPC expansion can be accurately constructed by the pseudo projection method. When the cubature rule is of low accuracy, then a higher order gPC expansion is pointless. This can be clearly seen in the example in Fig. 5. This is an example of approximating a three- dimensional ($d = 3$) nonlinear function, the example 5.1 from [29]. Here, the full tensor product cubature rule based on Gauss-Legendre quadrature is used. The number of Gauss quadrature points in each dimension is q_1. We observe that when q_1 is small, the gPC approximation error deteriorates at higher order and the expansion fails to converge. It is only when the cubature is of sufficiently high accuracy, $q_1 = 6$ in this case, that the gPC approximation exhibits the expected exponential error convergence for up to order $n = 6$.

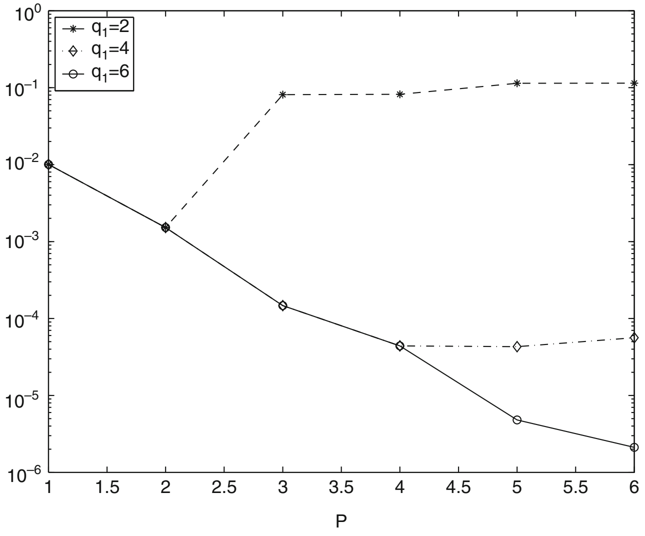

Fig. 5 Error convergence of pseudo projection method with different choices of cubature rule. Here the cubature rule is the full tensor quadrature rule with q_1 number of points in each dimension

6 Summary

Here, we briefly reviewed several major numerical approaches for stochastic collocation (SC). Our focus is on the fundamental approximation properties, basic formulations, and major properties of the methods. We reviewed the interpolation type SC, the regression type SC, and the pseudo projection type SC. Most, if not all, of the mainstream SC methods fall into these categories. There exist, however, a large variety of modifications and improvements over the core methods reviewed here. For example, many efforts have been devoted to the development of adaptive SC methods, particularly in conjunction of the sparse grid collocation. There also exist a large amount of works on the improvements of least squares methods. On the other hand, for under-sampled systems, the use of compressive sensing SC is still in its early stage, with many open issues to study. Overall, stochastic collocation has been under rapid development in the last decade, with many new variations emerging. Interested readers should consult the more recent literature for the latest developments.

References

1. Atkinson, A.C., Donev, A.N., Tobias, R.D.: Optimum Experimental Designs, with SAS. Oxford University Press, Oxford (2007)
2. Babuska, I., Nobile, F., Tempone, R.: A stochastic collocation method for elliptic partial differential equations with random input data. SIAM J. Numer. Anal. **45**(3), 1005–1034 (2007)

3. Barthelmann, V., Novak, E., Ritter, K.: High dimensional polynomial interpolation on sparse grid. Adv. Comput. Math. **12**, 273–288 (1999)
4. Bungartz, H.-J., Griebel, M.: Sparse grids. Acta Numer. **13**, 1–123 (2004)
5. Candès, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
6. Candès, E.J., Tao, T.: Near-optimal signal recovery from random projections: universal encoding strategies? IEEE Trans. Inform. Theory **52**(12), 5406–5425 (2006)
7. Cheney, W., Light, W.: A Course in Approximation Theory. Brooks/Cole Publishing Company, Pacific Grove (2000)
8. Cohen, A., Davenport, M.A., Leviatan, D.: On the stability and accuracy of least squares approximations. Found. Comput. Math. **13**(5), 819–834 (2013)
9. Cools, R.: Advances in multidimensional integration. J. Comput. Appl. Math. **149**, 1–12 (2002)
10. De Boor, C., Ron, A.: On multivariate polynomial interpolation. Constr. Approx. **6**, 287–302 (1990)
11. De Boor, C., Ron, A.: Computational aspects of polynomial interpolation in several variables. Math. Comput. **58**, 705–727 (1992)
12. Donoho, D.L.: Compressed sensing. IEEE Trans. Inform. Theory **52**(4), 1289–1306 (2006)
13. Doostan, A., Owhadi, H.: A non-adapted sparse approximation of PDEs with stochastic inputs. J. Comput. Phys. **230**(8), 3015–3034 (2011)
14. Engels, H.: Numerical Quadrature and Cubature. Academic, London/New York (1980)
15. Fedorov, V.V., Leonov, S.L.: Optimal Design for Nonlinear Response Models. CRC Press, Boca Raton (2014)
16. Ghanem, R.G., Spanos, P.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991)
17. Mathelin, L., Gallivan K.A.: A compressed sensing approach for partial differential equations with random input data. Commun. Comput. Phys. **12**, 919–954 (2012)
18. Migliorati, G., Nobile, F.: Analysis of discrete least squares on multivariate polynomial spaces with evaluations at low-discrepancy point sets. J. Complex. **31**(4), 517–542 (2015)
19. Migliorati, G., Nobile, F., E. von Schwerin, Tempone, R.: Approximation of quantities of interest in stochastic PDEs by the random discrete L^2 projection on polynomial spaces. SIAM J. Sci. Comput. **35**(3), A1440–A1460 (2013)
20. Migliorati, G., Nobile, F., von Schwerin, E., Tempone, R.: Analysis of the discrete L^2 projection on polynomial spaces with random evaluations. Found. Comput. Math. **14**(3), 419–456 (2014)
21. Narayan, A., Xiu, D.: Stochastic collocation methods on unstructured grids in high dimensions via interpolation. SIAM J. Sci. Comput. **34**(3), A1729–A1752 (2012)
22. Narayan, A., Xiu, D.: Constructing nested nodal sets for multivariate polynomial interpolation. SIAM J. Sci. Comput. **35**(5), A2293–A2315 (2013)
23. Powell, M.J.D.: Approximation Theory and Methods. Cambridge University Press, Cambridge (1981)
24. Rauhut, H., Ward, R.: Sparse Legendre expansions via ℓ_1-minimization. J. Approx. Theory **164**, 517–533 (2012)
25. Smolyak, S.A.: Quadrature and interpolation formulas for tensor products of certain classes of functions. Soviet Math. Dokl. **4**, 240–243 (1963)
26. Stroud, A.H.: Approximate Calculation of Multiple Integrals. Prentic-Hall, Englewood Cliffs (1971)
27. Trefethen, L.N.: Approximation Theory and Approximation Practice. SIAM, Philadelphia (2013)
28. Wasilkowski, G.W., Woźniakowski, H.: Explicit cost bounds of algorithms for multivariate tensor product problems. J. Complex. **11**, 1–56 (1995)
29. Xiu, D.: Efficient collocational approach for parametric uncertainty analysis. Commun. Comput. Phys. **2**(2), 293–309 (2007)
30. Xiu, D.: Numerical Methods for Stochastic Computations. Princeton University Press, Princeton (2010)

31. Xiu, D., Hesthaven, J.S.: High-order collocation methods for differential equations with random inputs. SIAM J. Sci. Comput. **27**(3), 1118–1139 (2005)
32. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput. **24**(2), 619–644 (2002)
33. Yan, L., Guo, L., Xiu, D.: Stochastic collocation algorithms using ℓ_1-minimization. Int. J. UQ **2**(3), 279–293 (2012)