SIAM/ASA J. Uncertainty Quantification
Vol. 10, No. 4, pp. 1410–1434

A General Framework of Rotational Sparse Approximation in Uncertainty Quantification*

Mengqi Hu†, Yifei Lou‡, and Xiu Yang§

Abstract. This paper proposes a general framework for estimating coefficients of generalized polynomial chaos (gPC) used in uncertainty quantification (UQ) via rotational sparse approximation. In particular, we aim to identify a rotation matrix such that the gPC expansion of a set of random variables after the rotation has a sparser representation. However, this rotational approach alters the underlying linear system to be solved, which makes finding the sparse coefficients more difficult than in the case without rotation. To solve this problem, we examine several popular nonconvex regularizations in compressive sensing (CS) that perform better than the classic ℓ_1 approach empirically. All these regularizations can be minimized by the alternating direction method of multipliers (ADMM). Numerical examples show superior performance of the proposed combination of rotation and nonconvex sparsity-promoting regularizations over those with and without rotation but using the convex ℓ_1 approach.

Key words. generalized polynomial chaos, uncertainty quantification, iterative rotations, compressive sensing, alternating direction, nonconvex regularization

MSC codes. 41A10, 65C30, 90C26

DOI. 10.1137/21M1391602

1. Introduction.

A surrogate model (also known as a "response surface") plays an important role in uncertainty quantification (UQ), as it can efficiently evaluate the quantity of interest (QoI) of a system given a set of inputs. Specifically, in parametric uncertainty studies, the input usually refers to a set of parameters in the system, while the QoI can be observables such as mass, density, pressure, and velocity, or even a trajectory of a dynamical system. The uncertainty in the system's parameters typically originates from the lack of physical knowledge, inaccurate measurements, etc. Therefore, it is common to treat these parameters as random variables, and statistics, e.g., mean, variance, and the probability density function (PDF) of the QoI with respect to such random parameters, are crucial in understanding the behavior of the system.

* Received by the editors January 13, 2021; accepted for publication (in revised form) June 6, 2022; published electronically October 27, 2022.
https://doi.org/10.1137/21M1391602
Funding: This work was partially supported by NSF CAREER 1846690. The work of the third author was supported by the U.S. Department of Energy (DOE), Office of Science, Office of Advanced Scientific Computing Research (ASCR) as part of Multifaceted Mathematics for Rare, Extreme Events in Complex Energy and Environment Systems (MACSER).

† Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA 18015 USA (meh621@lehigh.edu).
‡ Department of Mathematical Sciences, University of Texas at Dallas, Richardson, TX 75080 USA (yifei.lou@utdallas.edu).
§ Corresponding author. Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA 18015 USA (xiy518@lehigh.edu).

The generalized polynomial chaos (gPC) expansion [23, 59] is a widely used surrogate model in applied mathematics and engineering studies, which use orthogonal polynomials associated with measures of the aforementioned random variables. Under some conditions, the gPC expansion converges to the QoI in a Hilbert space as the number of polynomials increases [9, 21, 43, 59]. Both *intrusive* (e.g., stochastic Galerkin) and *nonintrusive* (e.g., probabilistic collocation) methods [4, 23, 53, 58, 59] have been developed for computing gPC coefficients. The latter are particularly desirable for studying a complex system, as it does not require modifying the computational models or simulation codes. For example, the gPC coefficients can be calculated based on input samples and corresponding output using least squared fitting, the probabilistic collocation method, etc.

However, in many practical problems, it is prohibitive to obtain a large amount of output samples in the nonintrusive methods, because it is costly to measure the QoI in experiments or to conduct simulations using a complicated model. Consequently, one should consider an underdetermined linear system (matrix), denoted as $\mathbf{\Psi}$, of size $M \times N$ with $M < N$ (or even $M \ll N$), where M is the size of available output samples and N is the number of basis functions used in the gPC expansion. When the solution to the underdetermined system is sparse, compressive sensing (CS) techniques [8, 10, 11, 19] are effective. Recent studies have shown some success in applying CS to UQ problems [2, 20, 33, 34, 45, 50, 61, 62, 65]. For example, sampling strategies [3, 27, 30, 47] can improve the property of $\mathbf{\Psi}$ to guarantee sparse recovery via the ℓ_1 minimization. Computationally, the weighted ℓ_1 minimization [1, 13, 44, 48, 65] assigns larger weights to smaller components (in magnitude) of the solution, and hence minimizing the weighted ℓ_1 norm leads to a sparser solution than the vanilla ℓ_1 minimization does. Additionally, adaptive basis selection [3, 6, 16, 28, 29] as well as dimension reduction techniques can be adopted to reduce the number of unknown variables [56, 68], which thus improves computational efficiency.

In this paper, we focus on a sparsity-enhancing approach, referred to as iterative rotation [34, 66, 67, 69], that intrinsically changes the structure of a surrogate model to make the gPC coefficients more sparse. However, this method tends to deteriorate properties of $\mathbf{\Psi}$ that are favored by CS algorithms, e.g., low coherence, which may counteract the benefit of the enhanced sparsity. Since the polynomials in the gPC expansion may not be orthogonal after the rotation of the random variables, the coherence of $\mathbf{\Psi}$ may not converges to zero asymptotically, leading to an amplified coherence after the rotation. To remedy this drawback, we innovatively combine the iterative rotation technique with a class of nonconvex regularizations to improve the efficiency of CS-based UQ methods. Specifically, our new approach uses rotations to increase the sparsity while taking advantage of the nonconvex formalism for dealing with a matrix $\mathbf{\Psi}$ that is highly coherent. In this way, we leverage the advantages of both methods to exploit information from limited samples of the QoI more efficiently and to construct gPC expansions more accurately.

The main contributions of this work are twofold. On the one hand, we propose a unified and flexible framework that combines iterative rotation and sparse recovery together with an efficient algorithm. On the other hand, we empirically validate the rule of thumb in CS that nonconvex regularizations often lead to better performance compared to the convex approach.

The rest of the paper is organized as follows. We briefly review gPC, CS, and rotational CS in section 2. We describe the combination of sparse signal recovery and rotation matrix

estimation in section 3. Section 4 is devoted to numerical examples, showing that the proposed approach significantly outperforms the stateoftheart. Finally, conclusions are given in section 5.

2. Prior works.

In this section, we briefly review gPC expansions, useful concepts in CS, and our previous work [66, 69] on the rotational CS for gPC methods.

2.1. Generalized polynomial chaos expansions.

We consider a QoI u that depends on location $\boldsymbol{x}$, time t, and a set of random variables $\boldsymbol{\xi}$ with the following gPC expansion:

$$u(\boldsymbol{x},t;\boldsymbol{\xi}) = \sum_{n=1}^{N} c_n(\boldsymbol{x},t)\psi_n(\boldsymbol{\xi}) + \varepsilon(\boldsymbol{x},t;\boldsymbol{\xi}), \tag{2.1}$$

where $c_n(\boldsymbol{x},t) = \mathbb{E}\{u(\boldsymbol{x},t;\boldsymbol{\xi})\psi_n(\boldsymbol{\xi})\}$ and ε denotes the truncation error. Here, $\{\psi_n\}_{n=1}^N$ are orthonormal with respect to the measure of $\boldsymbol{\xi}$, i.e.,

$$\int_{\mathbb{R}^d} \psi_i(\boldsymbol{\xi})\psi_j(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi} = \delta_{ij}, \tag{2.2}$$

where $\rho(\boldsymbol{\xi})$ is the PDF of $\boldsymbol{\xi}$, δ_{ij} is the Kronecker delta function, and we usually set $\psi_1(\boldsymbol{\xi}) \equiv 1$. We study systems relying on d-dimensional independent and identically distributed (i.i.d.) random variables $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_d)$, and the gPC basis functions are constructed by tensor products of univariate orthonormal polynomials associated with ξ_i. Specifically for a multi-index $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ with each $\alpha_i \in \mathbb{N} \cup \{0\}$, we set

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \psi_{\alpha_1}(\xi_1)\psi_{\alpha_2}(\xi_2)\cdots\psi_{\alpha_d}(\xi_d), \tag{2.3}$$

where ψ_{α_i} are univariate orthonormal polynomial (with respect to the PDF of ξ_i) of degree α_i. For two different multi-indices $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_d)$, we have

$$\int_{\mathbb{R}^d} \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\psi_{\boldsymbol{\beta}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi} = \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}} = \delta_{\alpha_1\beta_1}\delta_{\alpha_2\beta_2}\cdots\delta_{\alpha_d\beta_d}, \tag{2.4}$$

with $\rho(\boldsymbol{\xi}) = \rho_1(\xi_1)\rho_2(\xi_2)\cdots\rho_d(\xi_d)$, where $\rho_i(\xi_i)$ is the PDF of ξ_i. Specifically, in this work, since ξ_i are i.i.d., ρ_i are of the same form. Typically, a pth order gPC expansion involves all polynomials $\psi_{\boldsymbol{\alpha}}$ satisfying $|\boldsymbol{\alpha}| \le p$ for $|\boldsymbol{\alpha}| = \sum_{i=1}^d \alpha_i$, which indicates that a total number of $N = \binom{p+d}{d}$ polynomials are used in the expansion. For simplicity, we reorder $\boldsymbol{\alpha}$ in such a way that we index $\psi_{\boldsymbol{\alpha}}$ by ψ_n, which is consistent with (2.1).

In this paper, we focus on time independent problems in which the gPC expansion (2.1) at a fixed location $\boldsymbol{\xi}$ reduces to

$$u(\boldsymbol{\xi}) = \sum_{n=1}^{N} c_n\psi_n(\boldsymbol{\xi}) + \varepsilon(\boldsymbol{\xi}). \tag{2.5}$$

In practice, we collect M sample pairs $\{(\boldsymbol{\xi}^q, u(\boldsymbol{\xi}^q))\}_{q=1}^M$ (e.g., via running Monte Carlo simulations), and according to (2.5) we have

$$u(\boldsymbol{\xi}^q) = \sum_{n=1}^{N} c_n\psi_n(\boldsymbol{\xi}^q) + \varepsilon(\boldsymbol{\xi}^q), \quad q = 1, 2, \ldots, M.$$

Here, each $\boldsymbol{\xi}^q$ is a sample of $\boldsymbol{\xi}$, e.g., $\boldsymbol{\xi}^1 = (\xi_1^1, \ldots, \xi_d^1)$. We rewrite the above expansion in terms of the matrix-vector notation, i.e.,

$$\boldsymbol{\Psi}\boldsymbol{c} = \boldsymbol{u} - \boldsymbol{\varepsilon}, \tag{2.6}$$

where $\boldsymbol{u} = (u^1, \ldots, u^M)^\mathsf{T}$ is a vector of output samples, $\boldsymbol{c} = (c_1, \ldots, c_N)^\mathsf{T}$ is a vector of gPC coefficients, $\boldsymbol{\Psi}$ is an $M \times N$ matrix with $\Psi_{ij} = \psi_j(\boldsymbol{\xi}^i)$, and $\boldsymbol{\varepsilon} = (\varepsilon^1, \ldots, \varepsilon^M)^\mathsf{T}$ is a vector of errors with $\varepsilon^q = \varepsilon(\boldsymbol{\xi}^q)$. We are interested in identifying sparse coefficients $\mathbf{c} = (c_1, \ldots, c_N)^\mathsf{T}$ among the solutions of an underdetermined system with $M < N$ in (2.6), which is the focus of compressive sensing [12, 17, 22].

2.2. Compressive sensing. We review the concept of *sparsity*, which plays an important role in error estimation for solving the underdetermined system (2.6). The number of nonzero entries of a vector $\boldsymbol{c} = (c_1, \ldots, c_N)$ is denoted by $\|\boldsymbol{c}\|_0$. Note that $\|\cdot\|_0$ is named the "ℓ_0 norm" in [17], although it is neither a norm nor a seminorm. The vector $\boldsymbol{c}$ is called *s-sparse* if $\|\boldsymbol{c}\|_0 \le s$, and it is considered a sparse vector if $s \ll N$. Few practical systems have truly sparse gPC coefficients but rather *compressible* ones, i.e., only a few entries contributing significantly to its ℓ_1 norm. To this end, a vector $\boldsymbol{c}_s$ is defined as the best s-sparse approximation, which is obtained by setting all but the s-largest entries of $\boldsymbol{c}$ in magnitude to zero, and subsequently, $\boldsymbol{c}$ is regarded as sparse or compressible if $\|\boldsymbol{c} - \boldsymbol{c}_s\|_1$ is small for $s \ll N$.

In order to find a sparse vector $\boldsymbol{c}$ from (2.6), one formulates the problem

$$\hat{\boldsymbol{c}}_0 = \arg\min_{\boldsymbol{c}} \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{c} - \boldsymbol{u}\|_2^2 + \lambda\|\boldsymbol{c}\|_0, \tag{2.7}$$

where λ is a positive parameter to be tuned such that $\|\boldsymbol{\Psi}\hat{\boldsymbol{c}}_0 - \boldsymbol{u}\|_2 \le \epsilon$. As the ℓ_0 minimization (2.7) is NP-hard to solve [42], one often uses the convex ℓ_1 norm to replace ℓ_0, i.e.,

$$\hat{\boldsymbol{c}}_1 = \arg\min_{\boldsymbol{c}} \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{c} - \boldsymbol{u}\|_2^2 + \lambda\|\boldsymbol{c}\|_1. \tag{2.8}$$

A sufficient condition of the ℓ_1 minimization for exactly recovering the sparse signal was proved based on the *restricted isometry property* (RIP) [11]. Unfortunately, RIP is numerically unverifiable for a given matrix [5, 54]. Instead, a computable condition for ℓ_1's exact recovery is *coherence*, which is defined as

$$\mu(\boldsymbol{\Psi}) = \max_{i \ne j} \frac{|\langle \boldsymbol{\psi}_i, \boldsymbol{\psi}_j \rangle|}{\|\boldsymbol{\psi}_i\|\|\boldsymbol{\psi}_j\|}, \quad \text{with } \boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_N]. \tag{2.9}$$

Donoho and Elad [18] and Gribonval and Nielsen [24] proved independently that if

$$\|\hat{\boldsymbol{c}}_1\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\boldsymbol{\Psi})}\right), \tag{2.10}$$

then $\hat{\boldsymbol{c}}_1$ is indeed the sparsest solution to (2.8). Although the inequality condition in (2.10) is not sharp, the coherence of a matrix $\boldsymbol{\Psi}$ is often used as an indicator to quantify how difficult it is to find a sparse vector from a linear system governed by $\boldsymbol{\Psi}$ in the sense that the larger the coherence, the more challenging the problem of finding the sparse vector $\boldsymbol{c}$ [37, 71]. Apparently

if the samples $\boldsymbol{\xi}^q$ are drawn independently according to the distribution of $\boldsymbol{\xi}$, $\mu(\boldsymbol{\Psi})$ for the gPC expansion converges to zero as $M \to \infty$ [20]. However, the rotation technique tends to increase the coherence of the matrix, leading to unsatisfactory performance of the subsequent ℓ_1 minimization as discussed in [69].

In this work, we promote the use of nonconvex regularizations to find a sparse vector when the coherence of $\boldsymbol{\Psi}$ is relatively large, referred to as a coherent linear system. There are many nonconvex alternatives to approximating the ℓ_0 norm that give superior results over the ℓ_1 norm, such as $\ell_{1/2}$ [14, 60, 32], capped ℓ_1 [74, 51, 38], transformed ℓ_1 [40, 72, 73, 25], ℓ_1-ℓ_2 [70, 37, 36], ℓ_1/ℓ_2 [46, 57], and error function (ERF) [26]. We formulate a general framework that works for any regularization whose proximal operator can be found efficiently. Recall that a proximal operator $\mathbf{prox}_J(\cdot)$ of a functional $J(\cdot)$ is defined by

$$\mathbf{prox}_J(\boldsymbol{c};\mu) \in \arg\min_{\boldsymbol{y}} \left(\mu J(\boldsymbol{y}) + \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{c}\|_2^2 \right), \tag{2.11}$$

where μ is a positive parameter. We provide the formula of the aforementioned regularizations, together with their proximal operators, as follows:

- The ℓ_1 norm of $\mathbf{y}$ is $\|\boldsymbol{y}\|_1$, with its proximal operator given by

$$\mathbf{prox}_{\ell_1}(\boldsymbol{y};\mu) = \operatorname{sign}(\boldsymbol{y}) \circ \max(|\boldsymbol{y}| - \mu, 0), \tag{2.12}$$

  where $\circ$ denotes the Hadamard operator for componentwise multiplication.
- The $\ell_{1/2}$ norm is defined as $\|\boldsymbol{y}\|_{1/2} = (\sum_j \sqrt{|y_j|})^2$. The proximal operator of the square-root of the $\ell_{1/2}$ norm has a closed-form solution [60],

$$\mathbf{prox}_{\ell_{1/2}}(\boldsymbol{y};\mu) = \frac{3\mathbf{y}}{4} \circ \left[\cos\left(\frac{\pi}{3} - \frac{\phi(\boldsymbol{y})}{3} \right) \right]^2 \circ \max\left(\mathbf{y} - \frac{3}{4}\mu^{2/3}, 0 \right), \tag{2.13}$$

  where $\phi(\boldsymbol{y}) = \arccos(\frac{\mu}{8}(\frac{3}{|\boldsymbol{y}|})^{3/2})$, and the square is also computed componentwise.
- Transformed ℓ_1 (TL1) is defined as $\sum_j \frac{(\gamma+1)|y_j|}{\gamma+|y_j|}$ for a positive parameter γ, and its proximal operator [72] is given by

$$\mathbf{prox}_{\mathrm{TL1}}(\boldsymbol{y};\mu) = \begin{cases} \left[\frac{2}{3}(\gamma + |\boldsymbol{y}|) \cos\frac{\phi(\boldsymbol{y})}{3} - \frac{2}{3}\gamma + \frac{|\boldsymbol{y}|}{3} \right] & \text{if } |\boldsymbol{y}| > \theta, \\ 0 & \text{if } |\boldsymbol{y}| \le \theta, \end{cases} \tag{2.14}$$

  with

$$\phi(\boldsymbol{y}) = \arccos\left(1 - \frac{27\mu\gamma(\gamma+1)}{2(\gamma+|\boldsymbol{y}|)^3} \right) \quad \text{and} \quad \theta = \begin{cases} \mu\frac{\gamma+1}{\gamma} & \text{if } \mu \le \frac{\gamma^2}{2(\gamma+1)}, \\ \sqrt{2\mu(\gamma+1)} - \frac{\gamma}{2} & \text{if } \mu > \frac{\gamma^2}{2(\gamma+1)}. \end{cases}$$

- The ℓ_1-ℓ_2 regularization is defined by $\|\boldsymbol{y}\|_1 - \|\mathbf{y}\|_2$, and its proximal operator [36] is given by the following cases:
  - If $\|\boldsymbol{y}\|_\infty > \mu$, one has $\mathbf{prox}_{\ell_{1-2}}(\boldsymbol{y};\mu) = \frac{\boldsymbol{z}(\|\boldsymbol{z}\|_2+\mu)}{\|\boldsymbol{z}\|_2}$, where $\boldsymbol{z} = \mathbf{prox}_{\ell_1}(\boldsymbol{y};\mu)$.

  - If $\|\boldsymbol{y}\|_\infty \le \mu$, $\boldsymbol{c}^* := \mathbf{prox}_{\ell_{1-2}}(\boldsymbol{y};\mu)$ is an optimal solution if and only if $c_i^* = 0$ for $|y_i| < \|\boldsymbol{y}\|_\infty$, $\|\boldsymbol{c}^*\|_2 = \|\boldsymbol{y}\|_\infty$, and $c_i^* y_i \ge 0$ for all i. The optimality condition implies infinitely many solutions of $\boldsymbol{c}^*$, among which we choose $c_i^* = \operatorname{sign}(y_i)\|y\|_\infty$ for the smallest i that satisfies $|y_i| = \|\boldsymbol{y}\|_\infty$, and the remaining coefficients are set to zero.

- The ERF [26] is defined by

$$J_\sigma^{\mathrm{ERF}}(\boldsymbol{y}) := \sum_{j=1}^{n} \int_0^{|y_j|} e^{-\tau^2/\sigma^2} d\tau \tag{2.15}$$

for $\sigma > 0$. Though there is no closed-form solution for this problem, one can find the solution via Newton's method. In particular, the optimality condition of (2.11) for the ERF reads as

$$\mathbf{v} \in \mu \partial J_\sigma^{\mathrm{ERF}}(\boldsymbol{y}) + \boldsymbol{y} = \mu \exp\left(-\frac{\boldsymbol{y}^2}{\sigma^2}\right) \circ \partial|\boldsymbol{y}| + \boldsymbol{y}.$$

When $|v_i| \le \mu$, we have $x_i = 0$. Otherwise, the optimality condition becomes

$$v_i = \mu \exp\left(-\frac{y_i^2}{\sigma^2}\right) \operatorname{sign}(v_i) + y_i,$$

which can be found by Newton's method.

2.3. Rotational compressive sensing. To further enhance the sparsity, we aim to find a linear map $\mathbf{A} : \mathbb{R}^d \mapsto \mathbb{R}^d$ such that a new set of random variables $\boldsymbol{\eta}$, given by

$$\boldsymbol{\eta} = \mathbf{A}\boldsymbol{\xi}, \quad \boldsymbol{\eta} = (\eta_1, \eta_2, \dots, \eta_d)^\mathsf{T}, \tag{2.16}$$

leads to a sparser polynomial expansion than $\boldsymbol{\xi}$ does. We consider $\mathbf{A}$ as an orthogonal matrix, i.e., $\mathbf{A}\mathbf{A}^\mathsf{T} = \mathbf{I}$ with the identity matrix $\mathbf{I}$, such that the linear map from $\boldsymbol{\xi}$ to $\boldsymbol{\eta}$ can be regarded as a rotation in $\mathbb{R}^d$. Therefore, different from the approximation $u(\boldsymbol{\xi}) \approx \sum_{n=1}^N c_n \psi_n(\boldsymbol{\xi})$ in (2.5), we have the new polynomial expansion for u,

$$u(\boldsymbol{\xi}) \approx u_g(\boldsymbol{\xi}) = \sum_{n=1}^{N} \tilde{c}_n \psi_n(\mathbf{A}\boldsymbol{\xi}) = \sum_{n=1}^{N} \tilde{c}_n \psi_n(\boldsymbol{\eta}) = v_g(\boldsymbol{\eta}), \tag{2.17}$$

where $\tilde{c}_n$ are expansion coefficients. Ideally, $\tilde{\boldsymbol{c}}$ is sparser than $\boldsymbol{c}$, i.e., we aim to obtain a sparser representation of u. In previous works [34, 66], it is assumed that $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, so $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. For general cases where $\{\xi_i\}_{i=1}^d$ are not i.i.d. Gaussian, $\{\eta_i\}_{i=1}^d$ are not necessarily independent. Moreover, $\{\psi_n\}_{n=1}^N$ are not necessarily orthogonal to each other with respect to $\rho_{\boldsymbol{\eta}}$. Therefore, $v_g(\boldsymbol{\eta})$ may not be a standard gPC expansion of $v(\boldsymbol{\eta})$ but rather a polynomial equivalent to $u_g(\boldsymbol{\xi})$, with potentially sparser coefficients [69].

We can identify $\mathbf{A}$ using the gradient information of u based on the framework of active subspace [15, 49]. In particular, we define

$$\mathbf{W} = \frac{1}{\sqrt{M}} [\nabla u(\boldsymbol{\xi}^1), \nabla u(\boldsymbol{\xi}^2), \dots, \nabla u(\boldsymbol{\xi}^M)]. \tag{2.18}$$

Note that $\mathbf{W}$ is a $d \times M$ matrix, and we consider $M \geq d$ in this work. The singular value decomposition (SVD) of $\mathbf{W}$ yields

$$\mathbf{W} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}}, \tag{2.19}$$

where $\mathbf{U}$ is a $d \times d$ orthogonal matrix, $\boldsymbol{\Sigma}$ is a $d \times M$ matrix whose diagonal consists of singular values $\sigma_1 \geq \cdots \geq \sigma_d \geq 0$, and $\mathbf{V}$ is an $M \times M$ orthogonal matrix. If u is known, we set the rotation matrix as $\mathbf{A} = \mathbf{U}^{\mathsf{T}}$ projecting $\boldsymbol{\xi}$ in the direction of the principle components of ∇u. Unfortunately, u is unknown, and thus the samples of ∇u are not available. In this case, we approximate (2.18) by a computed solution u_g that can be obtained by the ℓ_1 minimization [66]. In other words, we have

$$\mathbf{W} \approx \mathbf{W}_g = \frac{1}{\sqrt{M}}[\nabla u_g(\boldsymbol{\xi}^1), \nabla u_g(\boldsymbol{\xi}^2), \ldots, \nabla u_g(\boldsymbol{\xi}^M)], \tag{2.20}$$

and the rotation matrix is constructed based on the SVD of $\mathbf{W}_g$:

$$\mathbf{W}_g = \mathbf{U}_g\boldsymbol{\Sigma}_g\mathbf{V}_g^{\mathsf{T}}, \quad \mathbf{A} = \mathbf{U}_g^{\mathsf{T}}. \tag{2.21}$$

Defining $\boldsymbol{\eta} = \mathbf{A}\boldsymbol{\xi}$, we compute the corresponding input samples as $\boldsymbol{\eta}^q = \mathbf{A}\boldsymbol{\xi}^q$ and construct a new measurement matrix $\boldsymbol{\Psi}(\boldsymbol{\eta})$ as $(\boldsymbol{\Psi}(\boldsymbol{\eta}))_{ij} = \psi_j(\boldsymbol{\eta}^i)$. We then solve the minimization problem in (2.8) to obtain $\tilde{\boldsymbol{c}}$. If some singular values of $\mathbf{W}_g$ are much larger than others, we can expect to obtain a sparser representation of u with respect to $\boldsymbol{\eta}$, which is dominated by the eigenspace associated with these larger singular values. On the other hand, if all the singular values σ_i are of the same order, the rotation does not enhance the sparsity. In practice, this method can be designed as an iterative algorithm, in which $\mathbf{A}$ and $\tilde{\boldsymbol{c}}$ are updated separately following an alternating direction manner.

It is worth noting that the idea of using a linear map to identify a possible low-dimensional structure is also used in sliced inverse regression (SIR) [35], the active subspace method [15, 49], basis adaptation [55], etc., but with different manners of computing the matrix. In contrast to these methods, the iterative rotation approach does not truncate the dimension in the sense that $\mathbf{A}$ is a square matrix. As an initial guess may not be sufficiently accurate, reducing the dimension before the iterations terminate may lead to suboptimal results. The dimension reduction was integrated with an iterative method in [63], while another iterative rotation method with SIR-based dimension reduction was proposed in [68]. We refer the interested reader to the respective literature.

3. The proposed approach. When applying the rotational CS techniques, the measurement matrix $\boldsymbol{\Psi}(\boldsymbol{\eta})$ may become more coherent compared with $\boldsymbol{\Psi}$. This is because popular polynomials ψ_i used in the gPC method, e.g., Legendre and Laguerre polynomials, are not orthogonal with respect to the measure of $\boldsymbol{\eta}$, so $\mu(\boldsymbol{\Psi}(\boldsymbol{\eta}))$ converges to a positive number instead of zero as $\mu(\boldsymbol{\Psi})$ does. Under such a coherent regime, we advocate the minimization of a nonconvex regularization to identify the sparse coefficients.

To start, we generate input samples $\{\boldsymbol{\xi}^q\}_{q=1}^M$ based on the distribution of $\boldsymbol{\xi}$ and select the gPC basis functions $\{\psi_j\}_{j=1}^N$ associated with $\boldsymbol{\xi}$ in order to generate the measurement matrix $\boldsymbol{\Psi}$ by setting $\Psi_{ij} = \psi_j(\boldsymbol{\xi}^i)$ as in (2.6), while initializing $\mathbf{A}^{(0)} = \mathbf{I}$ and $\boldsymbol{\eta}^{(0)} = \boldsymbol{\xi}$. Then we propose an alternating direction method (ADM) that combines the nonconvex minimization and

rotation matrix estimation. Specifically, given $\{\boldsymbol{\xi}^q\}_{q=1}^M$, $\{\psi_j\}_{j=1}^N$, and $\boldsymbol{u} := \{u^q = u(\boldsymbol{\xi}^q)\}_{q=1}^M$, we formulate the minimization problem

$$\arg\min_{\boldsymbol{c},\tilde{\mathbf{A}}} \lambda J(\boldsymbol{c}) + \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{c} - \boldsymbol{u}\|_2^2, \quad \mathbf{A}\mathbf{A}^\mathsf{T} = \mathbf{I} \text{ and } \boldsymbol{\Psi}_{ij} = \psi_j(\mathbf{A}\boldsymbol{\xi}^i), \tag{3.1}$$

where $J(\cdot)$ denotes a regularization functional and $\lambda > 0$ is a weighting parameter. For example, our early work [66] considered $J(\boldsymbol{c}) = \|\boldsymbol{c}\|_1$, i.e., the ℓ_1 approach. We can minimize (3.1) with respect to $\boldsymbol{c}$ and $\mathbf{A}$ in an alternating direction manner. When $\mathbf{A}$ is fixed, we minimize a sparsity-promoting functional $J(\cdot)$ to identify the sparse coefficients $\boldsymbol{c}$, as detailed in section 3.1. When $\boldsymbol{c}$ is fixed, optimizing $\mathbf{A}$ is computationally expensive, unless a dimension reduction technique is used, e.g., as in [56]. Instead, we use the rotation estimation introduced in section 2.3 to construct $\mathbf{A}$. Admittedly, this way of estimating $\mathbf{A}$ may not be optimal, but it can potentially promote the sparsity of $\boldsymbol{c}$ and thus improve the accuracy of the sparse approximation of the gPC expansion.

3.1. Finding sparse coefficient via ADMM. We focus on the $\boldsymbol{c}$-subproblem in (3.1), whose objective function is defined as

$$\tilde{\boldsymbol{c}} := \arg\min_{\boldsymbol{c}} \lambda J(\boldsymbol{c}) + \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{c} - \boldsymbol{u}\|_2^2. \tag{3.2}$$

We adopt the ADMM [7] to minimize (3.2). In particular, we introduce an auxiliary variable $\mathbf{y}$ and rewrite (3.2) as the equivalent problem

$$\min_{\boldsymbol{c},\boldsymbol{y}} \lambda J(\boldsymbol{c}) + \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{y} - \boldsymbol{u}\|_2^2 \quad \text{s.t.} \quad \boldsymbol{c} = \boldsymbol{y}. \tag{3.3}$$

This new formulation (3.3) makes the objective function separable with respect to two variables $\boldsymbol{c}$ and $\boldsymbol{y}$ to enable efficient computation. Specifically, the augmented Lagrangian corresponding to (3.3) can be expressed as

$$L_\rho(\boldsymbol{c}, \boldsymbol{y}; \boldsymbol{w}) = \lambda J(\boldsymbol{c}) + \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{y} - \boldsymbol{u}\|_2^2 + \langle \boldsymbol{w}, \boldsymbol{c} - \boldsymbol{y}\rangle + \frac{\rho}{2}\|\boldsymbol{c} - \boldsymbol{y}\|_2^2, \tag{3.4}$$

where $\boldsymbol{w}$ is an Lagrangian multiplier and ρ is a positive parameter. Then the ADMM iterations indexed by k consist of three steps,

$$\boldsymbol{c}_{k+1} = \arg\min_{\boldsymbol{c}} \lambda J(\boldsymbol{c}) + \frac{\rho}{2}\|\boldsymbol{c} - \boldsymbol{y}_k + \frac{\boldsymbol{w}_k}{\rho}\|_2^2 \tag{3.5}$$

$$\boldsymbol{y}_{k+1} = \arg\min_{\boldsymbol{y}} \frac{1}{2}\|\boldsymbol{\Psi}\boldsymbol{y} - \boldsymbol{u}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{c}_{k+1} - \boldsymbol{y} + \frac{\boldsymbol{w}_k}{\rho}\|_2^2 \tag{3.6}$$

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \rho(\boldsymbol{c}_{k+1} - \boldsymbol{y}_{k+1}). \tag{3.7}$$

Depending on the choice of $J(\cdot)$, the $\boldsymbol{c}$-update (3.5) can be given by its corresponding proximal operator, i.e.,

$$\boldsymbol{c}_{k+1} = \mathbf{prox}_J\left(\boldsymbol{y}_k - \frac{\boldsymbol{w}_k}{\rho}; \frac{\lambda}{\rho}\right). \tag{3.8}$$

Algorithm 3.1 The ADMM framework for solving a general sparse coding problem (3.2).

Input: measurement matrix $\mathbf{\Psi}$ and observed data $\boldsymbol{u}$.
Parameters: λ, ρ, $\epsilon \in \mathbb{R}^+$, and $\mathrm{k}_{\max} \in \mathbb{Z}^+$.
Initialize: $\boldsymbol{c}, \boldsymbol{y}, \boldsymbol{v}, \boldsymbol{w}$, and $k = 0$.
while $k \leq \mathrm{k}_{\max}$ or $\|\boldsymbol{c}_k - \boldsymbol{c}_{k-1}\| > \epsilon$ **do**
  $\boldsymbol{c}_{k+1} = \mathbf{prox}_J\left(\boldsymbol{y}_k - \frac{\boldsymbol{w}_k}{\rho}; \frac{\lambda}{\rho}\right)$
  $\boldsymbol{y}_{k+1} = (\mathbf{\Psi}^\mathsf{T}\mathbf{\Psi} + \rho\mathbf{I})^{-1}(\mathbf{\Psi}^\mathsf{T}\boldsymbol{u} + \rho\boldsymbol{c}_{k+1} + \boldsymbol{w}_k)$
  $\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \rho(\boldsymbol{c}_{k+1} - \boldsymbol{y}_{k+1})$
  $k = k + 1$
end while
Return $\tilde{\boldsymbol{c}} = \boldsymbol{c}_k$

Please refer to section 2.2 for detailed formula of proximal operators.

As for the $\boldsymbol{y}$-update, we take the gradient of (3.6) with respect to $\boldsymbol{y}$, thus leading to

$$\mathbf{\Psi}^T(\mathbf{\Psi}\boldsymbol{y} - \boldsymbol{u}) + \rho(\boldsymbol{y} - \boldsymbol{c}_{k+1} - \frac{\boldsymbol{w}_k}{\rho}) = \mathbf{0}. \tag{3.9}$$

Therefore, the update for $\mathbf{y}$ is given by

$$\boldsymbol{y}_{k+1} = (\mathbf{\Psi}^T\mathbf{\Psi} + \rho\mathbf{I})^{-1}\left(\mathbf{\Psi}^\mathsf{T}\boldsymbol{u} + \rho\boldsymbol{c}_{k+1} + \boldsymbol{w}_k\right). \tag{3.10}$$

Note that $\mathbf{\Psi}^\mathsf{T}\mathbf{\Psi} + \rho\mathbf{I}$ is a positive definite matrix and there are many efficient numeral algorithms for matrix inversion. Since $\mathbf{\Psi}$ has more columns than rows in our case, we further use the *Woodbury formula* to speed up,

$$\rho(\mathbf{\Psi}^\mathsf{T}\mathbf{\Psi} + \rho\mathbf{I})^{-1} = \mathbf{I} - \frac{1}{\rho}\mathbf{\Psi}^\mathsf{T}(\mathbf{\Psi}\mathbf{\Psi}^\mathsf{T} + \rho\mathbf{I})^{-1}, \tag{3.11}$$

as $\mathbf{\Psi}\mathbf{\Psi}^\mathsf{T}$ has a smaller dimension than $\mathbf{\Psi}^\mathsf{T}\mathbf{\Psi}$ to be inverted. In summary, the overall minimization algorithm based on ADMM is described in Algorithm 3.1.

3.2. Rotation matrix update. Suppose we obtain the gPC coefficients $\tilde{\boldsymbol{c}}^{(l)}$ at the lth iteration via Algorithm 3.1 with $l \geq 1$. Given $v_g^{(l)}(\boldsymbol{\eta})$ with input samples $\{(\boldsymbol{\eta}^{(l)})^q\}_{q=1}^M$ for $(\boldsymbol{\eta}^{(l)})^q = \mathbf{A}^{(l-1)}(\boldsymbol{\eta}^{(l-1)})^q$, we collect the gradient of $v_g^{(l)}$, denoted by

$$\mathbf{W}_g^{(l)} = \frac{1}{\sqrt{M}}\left[\nabla_{\boldsymbol{\xi}} v_g^{(l)}\left((\boldsymbol{\eta}^{(l)})^1\right), \ldots, \nabla_{\boldsymbol{\xi}} v_g^{(l)}\left((\boldsymbol{\eta}^{(l)})^M\right)\right], \tag{3.12}$$

where $\nabla_{\boldsymbol{\xi}}\cdot = (\partial\cdot/\partial\xi_1, \partial\cdot/\partial\xi_2, \cdots, \partial\cdot/\partial\xi_d)^\mathsf{T}$. It is straightforward to evaluate $\nabla\psi_n$ at $(\boldsymbol{\eta}^{(l)})^q$, as we construct ψ_n using the tensor product of univariate polynomials (2.3), and derivatives for widely used orthogonal polynomials, e.g., Hermite, Laguerre, Legendre, and Chebyshev, are well studied in the UQ literature. Here, we can analytically compute the gradient of $v_g^{(l)}$ with respect to $\boldsymbol{\xi}$ using the chain rule

$$\begin{aligned}\nabla_{\boldsymbol{\xi}} v_g^{(l)}\left((\boldsymbol{\eta}^{(l)})^q\right) &= \nabla_{\boldsymbol{\xi}} v_g^{(l)}\left(\mathbf{A}^{(l)}\boldsymbol{\xi}^q\right) = (\mathbf{A}^{(l)})^{\mathsf{T}}\nabla v_g^{(l)}(\boldsymbol{x})\Big|_{\boldsymbol{x}=\mathbf{A}^{(l)}\boldsymbol{\xi}^q} \\ &= (\mathbf{A}^{(l)})^{\mathsf{T}}\nabla\sum_{n=1}^{N}\tilde{c}_n^{(l)}\psi_n(\boldsymbol{x})\Big|_{\boldsymbol{x}=\mathbf{A}^{(l)}\boldsymbol{\xi}^q} = (\mathbf{A}^{(l)})^{\mathsf{T}}\sum_{n=1}^{N}\tilde{c}_n^{(l)}\nabla\psi_n(\boldsymbol{x})\Big|_{\boldsymbol{x}=\mathbf{A}^{(l)}\boldsymbol{\xi}^q}.\end{aligned} \tag{3.13}$$

Then, we have an update for $\mathbf{A}^{(l+1)} = (\mathbf{U}_g^{(l)})^{\mathsf{T}}$, where $\mathbf{U}_W^{(l)}$ is from the SVD of

$$\mathbf{W}_g^{(l)} = \mathbf{U}_g^{(l)}\boldsymbol{\Sigma}_g^{(l)}\left(\mathbf{V}_g^{(l)}\right)^{\mathsf{T}}. \tag{3.14}$$

Now we can define a new set of random variables as $\boldsymbol{\eta}^{(l+1)} = \mathbf{A}^{(l+1)}\boldsymbol{\xi}$ and compute their samples accordingly as $(\boldsymbol{\eta}^{(l+1)})^q = \mathbf{A}^{(l+1)}\boldsymbol{\xi}^q$. These samples are then used to construct a new measurement matrix $\boldsymbol{\Psi}^{(l+1)}$ as $\Psi_{ij}^{(l+1)} = \psi_j((\boldsymbol{\eta}^{(l+1)})^i)$ to feed into Algorithm 3.1 to obtain $\boldsymbol{c}^{(l+1)}$.

We summarize the entire procedure in Algorithm 3.2.

The stopping criteria we adopt are $l \le l_{\max}$ and the relative difference of coefficients between two consecutive iterations less than 10^{-3}. As the sparsity structure is problemdependent (see examples in section 4), more iterations do not grant significant improvements for many practical problems.

4. Numerical examples. In this section, we present four numerical examples to demonstrate the performance of the proposed method. Specifically, in section 4.1 we examine a function with low-dimensional structure, which has a truly sparse representation. Section 4.2 discusses an elliptic equation widely used in the UQ literature whose solution is not exactly sparse. The example discussed in section 4.3 has two dominant directions, even though the solution is not exactly sparse. Lastly, we present a high-dimensional example in section 4.4. These examples revisit some numerical tests in previous works [66, 69] and hence provide a direct comparison of the newly proposed approaches with the original one.

We compare the proposed framework to the rotation with the ℓ_1 approach (by setting $J(\mathbf{c}) = \|\mathbf{c}\|_1$ in (3.1)) and the one without rotation (by setting $l_{\max} = 1$ in Algorithm 3.2). The performance is evaluated in terms of relative error (RE), defined as $\frac{\|u-u_g\|_2}{\|u\|_2}$, where u is the exact solution, u_g is a reconstructed solution as a gPC approximation of u, and the integral in computing the norm $\|\cdot\|_2$ is approximated with a high-level sparse grid method based on one-dimensional Gaussian quadrature and the Smolyak structure [52].

We set $l_{\max} = 9$. To tune for the other two parameters (λ, ρ), we generate 10 independent random variables $\{\boldsymbol{\xi}^q\}_{q=1}^M$ and the corresponding function value $\boldsymbol{u} := \{u^q = u(\boldsymbol{\xi}^q)\}_{q=1}^M$ independent of the actual experiment to reconstruct the gPC expansion of the corresponding system. We begin by setting each component of (λ, ρ) to 10^{-4}, 10^{-3},...,10^1. For each combination of λ and ρ in this range, we apply Algorithm 3.2 for every regularization functional that is mentioned in section 2.2 and record all the relative errors. We find the two scalars $r_1, r_2 \in \{-4, -3, \ldots, 1\}$ such that $\lambda = 10^{r_1}$ and $\rho = 10^{r_2}$ give the smallest averaged RE over these 10 random sets. We then fine-tune the parameters by multiplying 10^{r_1} and $\rho = 10^{r_2}$ by 0.5, 1, ..., 9.5 individually. Denote $a_1, a_2 \in \{0.5, 1, \ldots, 9.5\}$ such that $\bar{\lambda} = a_1 10^{r_1}$ and $\bar{\rho} = a_2 10^{r_2}$ achieve the best result; these are regarded as the optimal parameters for

Algorithm 3.2 Alternating direction method of minimizing (3.1).

1: Generate input samples $\{\boldsymbol{\xi}^q\}_{q=1}^M$ based on the distribution of $\boldsymbol{\xi}$.
2: Generate corresponding output samples $\boldsymbol{u} := \{u^q = u(\boldsymbol{\xi}^q)\}_{q=1}^M$ by solving the complete model, e.g., running simulations, solvers, etc.
3: Select gPC basis functions $\{\psi_n\}_{n=1}^N$ associated with $\boldsymbol{\xi}$ and set counter $l = 0$. Set $\mathbf{A}^{(0)} = \mathbf{I}$, $\tilde{\boldsymbol{c}}^{(0)} = \mathbf{0}$, $e = 1$, and $\boldsymbol{\eta}^{(0)} = \boldsymbol{\xi}$.
4: Generate the measurement matrix $\boldsymbol{\Psi}^{(l)}$ by setting $\Psi_{ij}^{(l)} = \psi_j(\boldsymbol{\xi}^i)$.
5: **while** $l < l_{\max}$ and $e \geq 10^{-3}$ **do**
6: **if** $l > 0$ **then**
7: Construct $\mathbf{W}^{(l)}$ in (3.12) with $v_g^{(l)} = \sum_{n=1}^N \tilde{c}^{(l)} \psi_n(\boldsymbol{\xi}^{(l)})$.
8: Compute SVD of $\mathbf{W}_g^{(l)}$: $\mathbf{W}_g^{(l)} = \mathbf{U}_g^{(l)} \boldsymbol{\Sigma}_g^{(l)} \left(\mathbf{V}_g^{(l)}\right)^{\mathsf{T}}$.
9: Set $\mathbf{A}^{(l+1)} = \left(\mathbf{U}_g^{(l)}\right)^{\mathsf{T}}$ and $\boldsymbol{\eta}^{(l+1)} = \mathbf{A}^{(l+1)} \boldsymbol{\xi}$.
10: Construct the new measurement matrix $\boldsymbol{\Psi}^{(l+1)}$ with $\Psi_{ij}^{(l+1)} = \psi_j\left((\boldsymbol{\eta}^{(l+1)})^i\right)$.
11: **end if**
12: Solve the minimization problem via ADMM (Algorithm 3.1):

$$\tilde{\boldsymbol{c}}^{(l+1)} = \arg\min_{\mathbf{c}} \lambda J(\mathbf{c}) + \frac{1}{2} \|\boldsymbol{\Psi}^{(l+1)} \mathbf{c} - \boldsymbol{u}\|_2^2.$$

13: Calculate root square mean error of coefficients between two consecutive iterations

$$e = \frac{\|\tilde{\boldsymbol{c}}^{(l+1)} - \tilde{\boldsymbol{c}}^{(l)}\|_2}{\|\tilde{\boldsymbol{c}}^{(l+1)}\|_2}.$$

14: $l = l + 1$.
15: **end while**
16: Construct gPC expansion as $u(\boldsymbol{\xi}) \approx u_g(\boldsymbol{\xi}) = v_g^{(l)}(\boldsymbol{\eta}^{(l)}) = \sum_{n=1}^N \tilde{c}_n^{(l)} \psi_n(\mathbf{A}^{(l)} \boldsymbol{\xi})$.

subsequent experiments. We specify the parameter pair $(\bar{\lambda}, \bar{\rho})$ for each testing case in the corresponding section. After $(\bar{\lambda}, \bar{\rho})$ are identified, we conduct another 100 independent random trials (not including trials for the parameter tuning procedure) and report the average RE. In practice, these parameters can be determined by the k-fold cross-validation following the work of [20]. Empirically, we did not observe a significant difference between these two parameter tuning procedures for our numerical experiments, so we used a fixed set of (λ, ρ) for all 100 trials in each example.

In the first example, we do not terminate the iteration when $e > 10^{-3}$ in Algorithm 3.2 but finish *nine* iterations to compare the performance of different approaches when a truly sparse representation is available after a proper rotation. In the remaining examples, we follow the termination criterion (i.e., line 5 in Algorithm 3.2).

Table 4.1
Parameters $(\bar{\lambda}, \bar{\rho})$ for different regularization models in the case of ridge function.

	Legendre		Hermite		Laguerre	
UQ setting	$\bar{\lambda}$	$\bar{\rho}$	$\bar{\lambda}$	$\bar{\rho}$	$\bar{\lambda}$	$\bar{\rho}$
ℓ_1	6×10^{-4}	6×10^{-1}	1×10^{-2}	1×10^{-1}	5×10^{-2}	5×10^{0}
$l_{1/2}$	5×10^{-3}	1.6×10^{1}	1.8×10^{-2}	6×10^{-1}	6×10^{-2}	1.2×10^{0}
TL1	1×10^{-8}	1×10^{-7}	1×10^{-8}	1×10^{-7}	1×10^{-8}	1×10^{-7}
ERF	1.5×10^{-1}	1.6×10^{3}	1×10^{0}	1.2×10^{1}	1.1×10^{0}	1.5×10^{3}
$\ell_1 - \ell_2$	3×10^{-4}	5×10^{-1}	1×10^{-4}	1×10^{-2}	5×10^{-1}	1×10^{0}

4.1. Ridge function. Consider the following ridge function:

$$u(\xi) = \sum_{i=1}^{d} \xi_i + 0.25 \left(\sum_{i=1}^{d} \xi_i \right)^2 + 0.025 \left(\sum_{i=1}^{d} \xi_i \right)^3. \tag{4.1}$$

As $\{\xi_i\}_{i=1}^d$ are equally important in this example, adaptive methods [41, 64, 75] that build surrogate models hierarchically based on the importance of ξ_i may not be effective. We consider a rotation matrix in the form of

$$\mathbf{A} = \begin{pmatrix} d^{-\frac{1}{2}} & d^{-\frac{1}{2}} & \cdots & d^{-\frac{1}{2}} \\ & \tilde{\mathbf{A}} & & \end{pmatrix}, \tag{4.2}$$

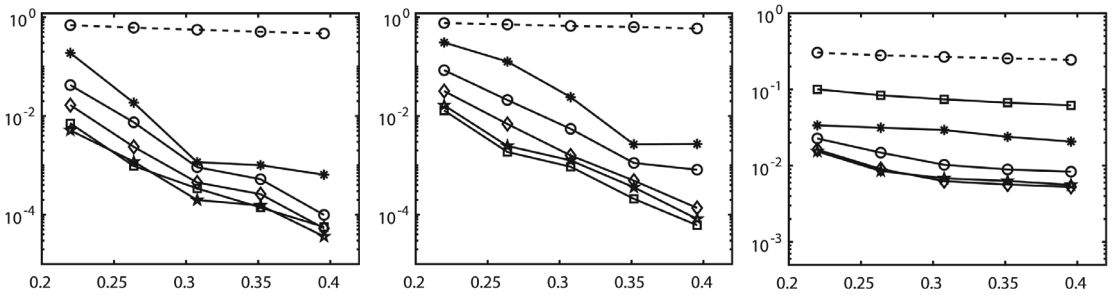
where $\tilde{\mathbf{A}}$ is a $d \times (d-1)$ matrix designed to guarantee the orthogonality of the matrix $\mathbf{A}$, e.g., $\mathbf{A}$ can be obtained by the Gram–Schmidt process. With this choice of $\mathbf{A}$, we have $\eta_1 = d^{-\frac{1}{2}} \sum_{i=1}^{d} \xi_i$, and u can be represented as

$$u(\boldsymbol{\xi}) = v(\boldsymbol{\eta}) = d^{\frac{1}{2}} \eta_1 + 0.25 d \eta_1^2 + 0.025 d^{\frac{3}{2}} \eta_1^3. \tag{4.3}$$

In the expression $u(\boldsymbol{\xi}) = \sum_{n=1}^{N} \tilde{c}_n \psi_n(\mathbf{A}\boldsymbol{\xi}) = \sum_{n=1}^{N} \tilde{c}_n \psi_n(\boldsymbol{\eta})$, all of the polynomials that are not related to η_1 make no contribution to the expansion, which guarantees the sparsity of $\tilde{\boldsymbol{c}} = (\tilde{c}_1, \ldots, \tilde{c}_N)$.

By setting $d = 12$ (hence, the number of gPC basis functions is $N = 455$ for $p = 3$), we compare the accuracy of computing gPC expansions by minimizing different regularization functionals with and without rotations. We consider gPC expansion using a Legendre polynomial (assuming ξ_i are i.i.d. uniform random variables), Hermite polynomial expansion (assuming ξ_i are i.i.d. Gaussian random variables), and a Laguerre polynomial (assuming ξ_i are i.i.d. exponential random variables), respectively. The number of samples M ranges from 100 to 180, with each repeated 100 times to compute the average RE. The parameters $\bar{\lambda}, \bar{\rho}$ for all the regularization models are listed in Table 4.1.

Figure 4.1 plots relative errors corresponding to the ratios of M/N, showing that nonconvex regularizations outperform the convex ℓ_1 approach in most cases. In particular, the best regularizations for different polynomials are different, but ℓ_1-ℓ_2 and ERF generally perform very well (within top two). TL1 and $\ell_{1/2}$ work well in the cases of Legendre and Hermite polynomials, but they are less accurate for the Laguerre polynomial case. Please refer to



(a) Legendre (b) Hermite (c) Laguerre

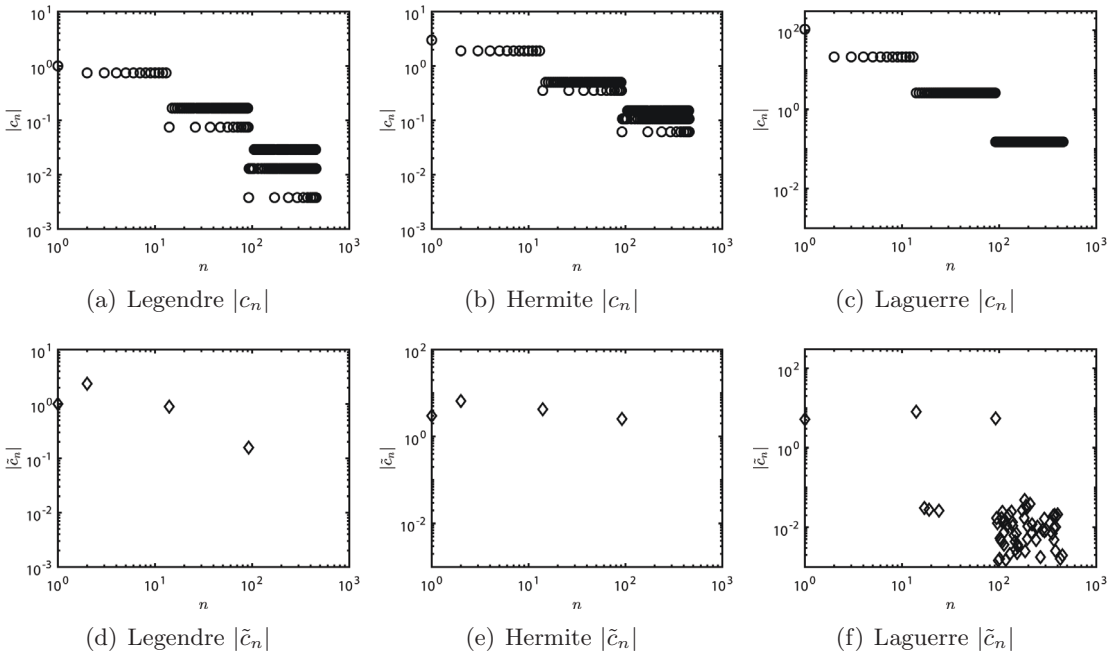
Figure 4.1. *Relative errors of Legendre, Hermite, and Laguerre polynomial expansions for ridge function against ratio M/N. Solid lines represent experiments with rotations applied, whereas the dashed line references the ℓ_1 test result without rotation. "$\circ$" is the marker for ℓ_1 minimization, "$*$" is the marker for $\ell_{1/2}$, "$\square$" is the marker for TL1, "$\star$" is the marker for ERF, and "$\diamond$" is the marker for $\ell_1 - \ell_2$.*

section 4.5 for in-depth discussions on the performance of these regularizations with respect to the matrix coherence.

Figure 4.1 also indicates that the standard ℓ_1 minimization without rotation is not effective, as its relative error is close to 50% even when M/N approaches 0.4. Our iterative rotation methods with any regularization yield much higher accuracy, especially for a larger M value. The reason can be partially explained by Figure 4.2. Specifically, the plots (a), (b), and (c) of Figure 4.2 are about the absolute values of exact coefficients $|c_n|$ of Legendre, Hermite, and Laguerre polynomials, while the plots (d), (e), and (f) show corresponding coefficients $|\tilde{c}_n|$ after nine iterations with the ℓ_1-ℓ_2 minimization using 120 samples randomly chosen from the 100 independent experiments; we exclude $\tilde{c}_n$, whose absolute value is smaller than 10^{-3}, since they are sufficiently small (more than two magnitudes smaller than the dominating ones). As demonstrated in Figure 4.2, the iterative update on the rotation matrix significantly sparsifies the representation of u, and, as a result, the efficiency of CS methods is substantially enhanced. Moreover, $\tilde{c}_n$ for the Laguerre polynomial is not as sparse as the ones for the Legendre and Hermite polynomials. Consequently, the Laguerre polynomial shows less accurate results in Figure 4.1(c) compared with the other two polynomials in Figure 4.1(a),(b). The phenomenon can be partially explained by the coherence of $\mathbf{\Psi}$ for different polynomials, i.e., the coherence in the Laguerre case is much larger (> 0.94 before rotation and > 0.99 after rotation) than in the other two, which thus makes any sparse regression algorithms less effective. For more detailed discussions on coherence, please refer to Table 4.5 and section 4.5.

4.2. Elliptic equation. Next we consider a one-dimensional elliptic differential equation with a random coefficient [20, 66],

$$-\frac{d}{dx}\left(a(x;\boldsymbol{\xi})\frac{du(x;\boldsymbol{\xi})}{dx}\right) = 1, \quad x \in (0,1), \tag{4.4}$$
$$u(0) = u(1) = 0,$$



(a) Legendre $|c_n|$ (b) Hermite $|c_n|$ (c) Laguerre $|c_n|$

(d) Legendre $|\tilde{c}_n|$ (e) Hermite $|\tilde{c}_n|$ (f) Laguerre $|\tilde{c}_n|$

Figure 4.2. *(Ridge function.) Absolute values of exact coefficients c_n (first row) and coefficients $\tilde{c}_n$ after* 9 *rotations with the ℓ_1-ℓ_2 minimization (second row) using* 120 *samples.*

where $a(x;\boldsymbol{\xi})$ is a log-normal random field based on Karhunen–Loève (KL) expansion,

$$a(x;\boldsymbol{\xi}) = a_0(x) + \exp\left(\sigma \sum_{i=1}^{d} \sqrt{\lambda_i}\phi_i(x)\xi_i\right), \tag{4.5}$$

$\{\xi_i\}$ are i.i.d. random variables, and $\{\lambda_i, \phi_i(t)\}_{i=1}^{d}$ are eigenvalues/eigenfunctions (in descending order in λ_i) of an exponential covariance kernel,

$$C(x,x') = \exp\left(\frac{|x-x'|}{l_c}\right). \tag{4.6}$$

The value of λ_i and the analytical expressions for ϕ_i are given in [31]. We set $a_0(x) \equiv 0.1, \sigma = 0.5, l_c = 0.2$, and $d = 15$ such that $\sum_{i=1}^{d}\lambda_i > 0.93\sum_{i=1}^{\infty}\lambda_i$. For each input sample $\boldsymbol{\xi}^q$, the solution of the deterministic elliptic equation can be obtained by [65] as follows:

$$u(x) = u(0) + \int_0^x \frac{a(0)u(0)' - y}{a(y)}\mathrm{d}y. \tag{4.7}$$

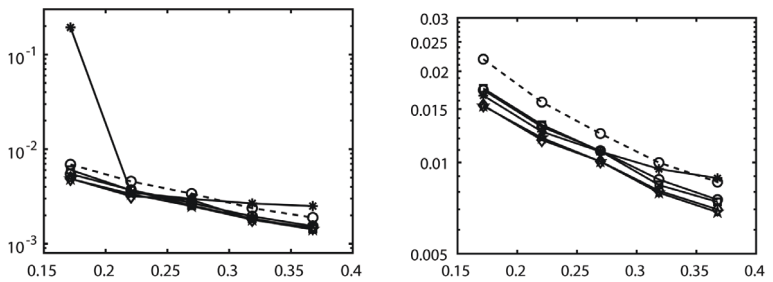
By imposing the boundary condition $u(0) = u(1) = 0$, we can compute $a(0)u(0)'$ as

$$a(0)u(0)' = \left(\int_0^1 \frac{y}{a(y)}\mathrm{d}y\right) \Big/ \left(\int_0^1 \frac{1}{a(y)}\mathrm{d}y\right). \tag{4.8}$$

The integrals in (4.7) and (4.8) are obtained by highly accurate numerical integration. For this example, we choose the quantity of interest to be $u(x;\boldsymbol{\xi})$ at $x = 0.35$. We aim to build a

Table 4.2
Parameters $(\bar{\lambda}, \bar{\rho})$ for different regularization models in the case of the elliptic equation.

	Legendre		Hermite	
UQ setting	$\bar{\lambda}$	$\bar{\rho}$	$\bar{\lambda}$	$\bar{\rho}$
ℓ_1	6×10^{-4}	1×10^{-1}	1.3×10^{-3}	3×10^{0}
$l_{1/2}$	8×10^{-5}	1×10^{2}	3×10^{-4}	1.2×10^{2}
TL1	1×10^{-8}	1×10^{-7}	1×10^{-8}	1×10^{-7}
ERF	2×10^{-2}	2×10^{4}	9×10^{-3}	1.3×10^{3}
$\ell_1 - \ell_2$	1×10^{-3}	6×10^{-2}	1.9×10^{-3}	2.3×10^{-2}



(a) Legendre (b) Hermite

Figure 4.3. *Relative error of Legendre and Hermite polynomial expansions for an elliptic equation against ratio M/N. Solid lines represent experiments with rotations applied, whereas the dashed line references the ℓ_1 test result without rotation. "$\circ$" is the marker for ℓ_1 minimization, "$*$" is the marker for $\ell_{1/2}$, "$\square$" is the marker for TL1, "$\star$" is the marker for ERF, and "$\diamond$" is the marker for $\ell_1 - \ell_2$.*

third-order Legendre (or Hermite) polynomial expansion, which includes $N = 816$ basis functions. The relative error is approximated by a level-6 sparse grid method. The parameters $(\bar{\lambda}, \bar{\rho})$ are given in Table 4.2.

Relative errors of the Legendre and Hermite polynomial expansions are presented in Figure 4.3. All the methods with rotation perform almost the same, except that $\ell_{1/2}$ may yield unstable results, especially when sample size is small as shown in Figure 4.3(a). This phenomenon is consistent with previous works [37, 71]. In this elliptic equation case, we do not observe a significant improvement in the nonconvex regularizations over the convex ℓ_1 model, as opposed to Figure 4.1. ℓ_1-ℓ_2 and ERF perform the best for both Legendre and Hermite polynomials. In this case, the solution does not have an underlying low-dimensional structure under rotation as in the previous example, and the truncation error exists, which is common in practical problems. This is why the improvement by the rotational method is minor.

4.3. Korteweg–de Vries equation. As an example of a more complicated and nonlinear differential equation, we consider the Korteweg–de Vries (KdV) equation with time-dependent additive noise,

$$\begin{aligned} &u_t(x,t;\boldsymbol{\xi}) - 6u(x,t;\boldsymbol{\xi})u_x(x,t;\boldsymbol{\xi}) + u_{xxx}(x,t;\boldsymbol{\xi}) = f(t;\boldsymbol{\xi}), \quad x \in (-\infty,\infty), \\ &u(x,0;\boldsymbol{\xi}) = -2\,\mathrm{sech}^2(x). \end{aligned} \tag{4.9}$$

Here $f(t;\xi)$ is modeled as a random field represented by the Karhunen–Loève expansion,

$$f(t;\boldsymbol{\xi}) = \sigma \sum_{i=1}^{d} \sqrt{\lambda_i}\phi_i(t)\xi_i, \tag{4.10}$$

where σ is a constant and $\{\lambda_i, \phi_i(t)\}_{i=1}^{d}$ are eigenvalues/eigenfunctions (in descending order) of the exponential covariance kernel equation (4.6). We set $l_c = 0.25$ and $d = 10$ in (4.6) such that $\sum_{i=1}^{d} \lambda_i > 0.96 \sum_{i=1}^{\infty} \lambda_i$. Under this setting, we have an analytical solution given by

$$u(x,t;\boldsymbol{\xi}) = \sigma \sum_{i=1}^{d} \sqrt{\lambda_i}\xi_i \int_0^t \phi_i(y)\mathrm{d}y - 2\,\mathrm{sech}^2\left(x - 4t + 6\sigma \sum_{i=1}^{d} \sqrt{\lambda_i}\xi_i \int_0^t \int_0^z \phi_i(y)\mathrm{d}y\mathrm{d}z\right). \tag{4.11}$$

We choose the QoI to be $u(x,t;\xi)$ at $x = 6, t = 1$, and $\sigma = 0.4$. Thanks to analytical expressions of $\phi_i(x)$, we can compute the integrals in (4.11) with high accuracy. Denote

$$A_i = \sqrt{\lambda_i} \int_0^1 \phi_i(y)\mathrm{d}y \quad \text{and} \quad B_i = \sqrt{\lambda_i} \int_0^1 \int_0^z \phi_i(y)\mathrm{d}y\mathrm{d}z, \quad i = 1, 2, \ldots, d; \tag{4.12}$$

the analytical solution can be written as

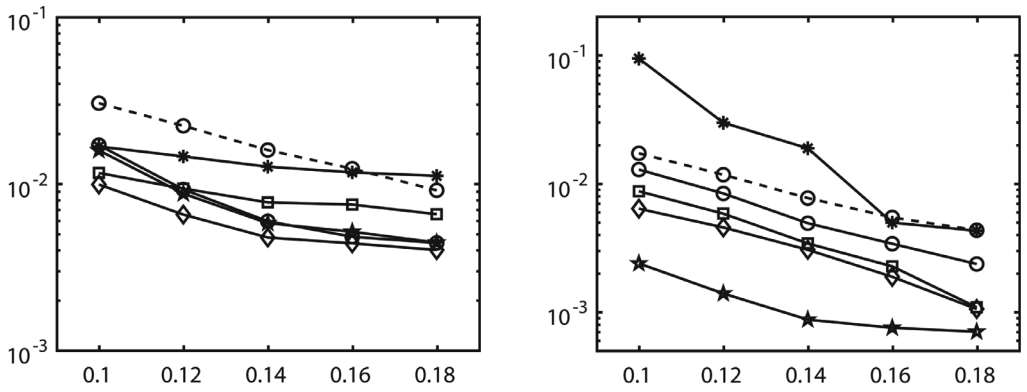
$$u(x,t;\boldsymbol{\xi})|_{x=6,t=1} = \sigma \sum_{i=1}^{d} A_i\xi_i - 2\,\mathrm{sech}^2\left(2 + 6\sigma \sum_{i=1}^{d} B_i\xi_i\right). \tag{4.13}$$

We use a fourth-order gPC expansion to approximate the solution, i.e., $p = 4$, and the number of gPC basis functions is $N = 1001$. The experiment is repeated 50 times to compute the average relative error for each gPC expansion. Parameters chosen for different regularizations are given in Table 4.3. Relative errors of the Legendre and Hermite polynomial expansions are presented in Figure 4.4, which illustrates the combined method of iterative rotation, and nonconvex minimization outperforms the simple ℓ_1 approaches. The coherence μ of $\boldsymbol{\Psi}$ for the Legendre polynomial is around 0.6/0.85 before/after the rotation, while it increases to over 0.92 for the Hermite polynomial. In such a highly coherent regime, $\ell_{1/2}$ does not work very well, and other nonconvex regularizations, i.e., ERF, TL1, and ℓ_1-ℓ_2, perform better than ℓ_1 in the Hermite polynomial case (especially ERF).

4.4. High-dimensional function. We illustrate the potential capability of the proposed approach for dealing with higher-dimensional problems, referred to as the HD function. Specifically, we select a function similar to the one in section 4.1 but with a much higher dimension,

$$u(\boldsymbol{\xi}) = \sum_{i=1}^{d} \xi_i + 0.25 \left(\sum_{i=1}^{d} \xi_i/\sqrt{i}\right)^2, \quad d = 100. \tag{4.14}$$

The total number of basis functions for this example is $N = 5151$. The experiment is repeated 20 times to compute the average relative errors for each polynomial. Parameters for this set of experiments are given in Table 4.4. The results are presented in Figure 4.5, showing that



(a) Legendre (b) Hermite

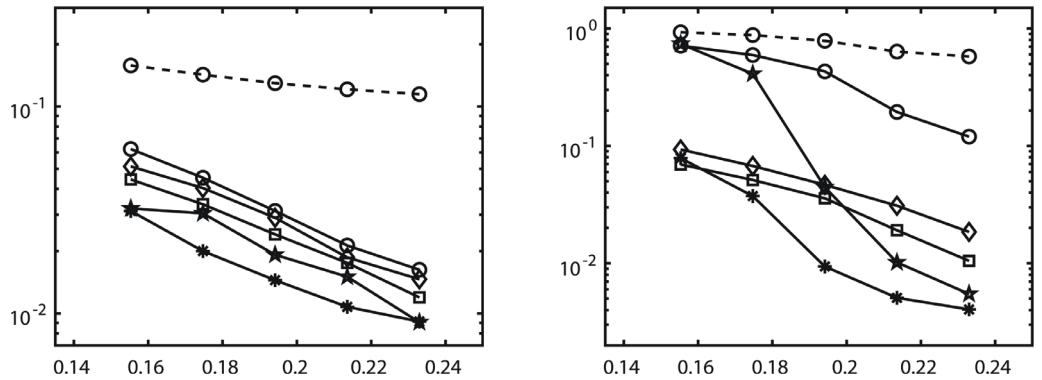
Figure 4.4. *(KdV equation.) Relative error of Legendre and Hermite polynomial expansions for the KdV equation against ratio M/N. Solid lines represent experiments with rotations applied, whereas the dashed line references the ℓ_1 test result without rotation. "$\circ$" is the marker for ℓ_1 minimization, "$*$" is the marker for $\ell_{1/2}$, "$\square$" is the marker for TL1, "$\star$" is the marker for ERF, and "$\diamond$" is the marker for $\ell_1 - \ell_2$.*

Table 4.3
Parameters $(\bar{\lambda}, \bar{\rho})$ for different regularization models in the case of the KdV function.

	Legendre		Hermite	
Method	$\bar{\lambda}$	$\bar{\rho}$	$\bar{\lambda}$	$\bar{\rho}$
ℓ_1	1×10^{-4}	1×10^{-2}	1×10^{-2}	4×10^{-4}
$l_{1/2}$	5×10^{-3}	1.6×10^{1}	1.8×10^{-2}	6×10^{-1}
Transformed ℓ_1	1×10^{-8}	1×10^{-7}	1×10^{-8}	1×10^{-7}
ERF	1.5×10^{-1}	1.6×10^{3}	1×10^{0}	1.2×10^{1}
$\ell_1 - \ell_2$	1×10^{-4}	1×10^{-2}	1×10^{-4}	1×10^{-2}

all nonconvex methods with rotation outperform the ℓ_1 approach except for ERF under the Hermite basis when sample size is small. Different from previous examples (ridge, elliptic, and KdV), $\ell_{1/2}$ achieves the best result, as the corresponding coherence is relatively small, being around 0.2/0.3 before/after rotation for the Legendre polynomial and 0.3 for the Hermite polynomial. In addition, Figure 4.5(b) suggests that the ERF method is stable with respect to sampling ratios for the Hermite basis.

4.5. Discussion. We intend to discuss the effects of coherence and the number of rotations on the performance of the ℓ_1 and other nonconvex approaches. As reported in [26, 37, 71], the ℓ_1-ℓ_2 and ERF methods perform particularly well for coherent matrices (i.e., large μ), and ℓ_p performs well for incoherenct matrices (i.e., small μ), which motivates us to compute the coherence values and report them in Tables 4.5 and 4.6 using the ℓ_1-ℓ_2 method for ridge function and elliptic equation/KdV equation/HD function, respectively. Here, we use μ of each iteration by ℓ_1-ℓ_2 as an example, and its values for other regularizations (including ℓ_1) are similar. Both tables confirm that applying rotation increases the coherence level of the sensing matrix $\mathbf{\Psi}$ except for the Hermite basis. As we show in numerical examples, when the coherence is large (e.g., around 0.9 or even larger in the Hermite polynomial for KdV) ℓ_1-ℓ_2



(a) Legendre (b) Hermite

Figure 4.5. *(High-dimensional function.) Relative error of Legendre and Hermite polynomial expansions for the high-dimensional function against sampling ratio M/N. Solid lines represent experiments with rotations applied, whereas the dashed line references the ℓ_1 test result without rotation. "$\circ$" is the marker for ℓ_1 minimization, "$*$" is the marker for $\ell_{1/2}$, "$\square$" is the marker for TL1, "$\star$" is the marker for ERF, and "$\diamond$" is the marker for $\ell_1 - \ell_2$. All results with rotations are plotted with a threshold of 10^{-3}.*

Table 4.4
Parameters $(\bar{\lambda}, \bar{\rho})$ for different regularization models in the case of HD function.

	Legendre		Hermite	
UQ setting	$\bar{\lambda}$	$\bar{\rho}$	$\bar{\lambda}$	$\bar{\rho}$
ℓ_1	1.2×10^{0}	5×10^{-2}	1×10^{-2}	1×10^{-3}
$l_{1/2}$	1×10^{-1}	1×10^{2}	5×10^{-2}	1×10^{2}
TL1	1×10^{-6}	1×10^{-7}	4×10^{-6}	1×10^{-7}
ERF	1.5×10^{-1}	1.6×10^{3}	5×10^{-2}	1×10^{2}
$\ell_1 - \ell_2$	5×10^{-2}	1×10^{2}	1×10^{-3}	9×10^{-2}

and ERF perform better than the convex ℓ_1 method. When the coherence is small (e.g., $\lesssim 0.3$ in the high-dimensional case), $\ell_{1/2}$ gives the best results among all the competing methods. One the other hand, $\ell_{1/2}$ may lead to unstable and unsatisfactory results, sometimes even worse than those of the convex ℓ_1 method, when the coherence of the sensing matrix is large. In the extreme case when the coherence is close to 1 (e.g., Laguerre in the ridge function), the best result is only slightly better than ℓ_1, which seems difficult for any sparse recovery algorithms to achieve. This series of observations coincide with the empirical performance in CS; i.e., $\ell_{1/2}$ works the best for incoherent matrices, while ℓ_1-ℓ_2 and ERF work better for coherent cases.

We then examine the effect of the number of rotations in Figure 4.6. Here we use the ridge function and the KdV equation as examples and show the results by ℓ_1-ℓ_2. For the ridge function, more rotations lead to better accuracy, while it stagnates at 3–5 rotations for the KdV equation. This is because the ridge function has a very good low-dimensional structure, i.e., the dimension can be reduced to one using a linear transformation $\boldsymbol{\eta} = \mathbf{A}\boldsymbol{\xi}$, while the KdV equation does not have this property. Also, there is no truncation error $\varepsilon(\boldsymbol{\xi})$ when using

Table 4.5
Average coherence of matrix $\mathbf{\Psi}$ (size 160×455) for the ridge function.

	Ridge function		
Rotations	Legendre	Hermite	Laguerre
0	0.4692	0.7622	0.9448
3	0.6833	0.7527	0.9910
6	0.6822	0.7519	0.9911
9	0.6762	0.7657	0.9911

Table 4.6
Average coherence of matrix $\mathbf{\Psi}$ for elliptic equation (matrix size 160×816), KdV equation (matrix size 160×1001), and HD function (matrix size 1000×5151).

	Elliptic equation		KdV equation		HD function	
Rotations	Legendre	Hermite	Legendre	Hermite	Legendre	Hermite
0	0.5014	0.7770	0.6079	0.9121	0.2117	0.2852
1	0.6958	0.7830	0.8825	0.9223	0.2580	0.3075
2	0.6943	0.7719	0.8487	0.9167	0.2664	0.2956
3	0.6896	0.7710	0.8677	0.9214	0.2522	0.3003

the gPC expansion to represent the ridge function, as we use a third order expansion, while $\varepsilon(\boldsymbol{\xi})$ exists for the KdV equation. In most practical problems, the truncation error exists, and the linear transform may not yield the optimal low-dimensional structure so that it has sparse coefficients of the gPC expansion. Therefore, we empirically set a maximum number of rotations $l_{\max}$ to terminate iterations in the algorithm.

We present the computation time in Table 4.7. All the experiments were performed on an AMD Ryzen 5 3600, 16 GB RAM machine running Windows 10, versions 1904 and 2004, with MATLAB 2018b. The major computation comes from two components: one is the ℓ_1-ℓ_2 minimization, and the other is finding the rotation matrix A. The computation complexity for every iteration of the ℓ_1-ℓ_2 algorithm is $O(M^3 + M^2N)$, which reduces to $O(M^2N)$ as we assume $M \ll N$. In practice, we choose the maximum outer/inner numbers in Algorithm 3.1 as $n_{max} = 10, k_{max} = 2N$, respectively, and hence the complexity for the ℓ_1-ℓ_2 algorithm is $O(M^2N^2)$. To find the rotation matrix A, one has to construct a matrix W using (3.12) with a complexity of $O(M^3N)$, followed by an SVD with a complexity of $O(M^3N + M^2N^2)$. Therefore, the total complexity of our approach is $O(M^2N^2)$ per rotation. We divide the time of the Legendre polynomial reported in Table 4.7 by $l_{\max}(MN)^2$, getting $1.39e^{-10}, 1.95e^{-10}$, and $0.25e^{-10}$. As the ratios are of the same order, the empirical results are consistent with the complexity analysis.

Furthermore, since the weighted minimization is an effective approach in the sparse-regression-based UQ study (e.g., [65, 44]), we compare the ADMM approach to reweighted algorithms to minimize a log-sum regularization [13] and the ℓ_p norm [39]. The reweighted framework requires solving a weighted ℓ_1 subproblem iteratively, which is thus computationally more expensive than ADMM. In addition, the reweighted algorithms are more sensitive to parameters than ADMM and hence require more elaborate parameter tuning. We compare the performance of ADMM and weighted minimization using the ridge function and

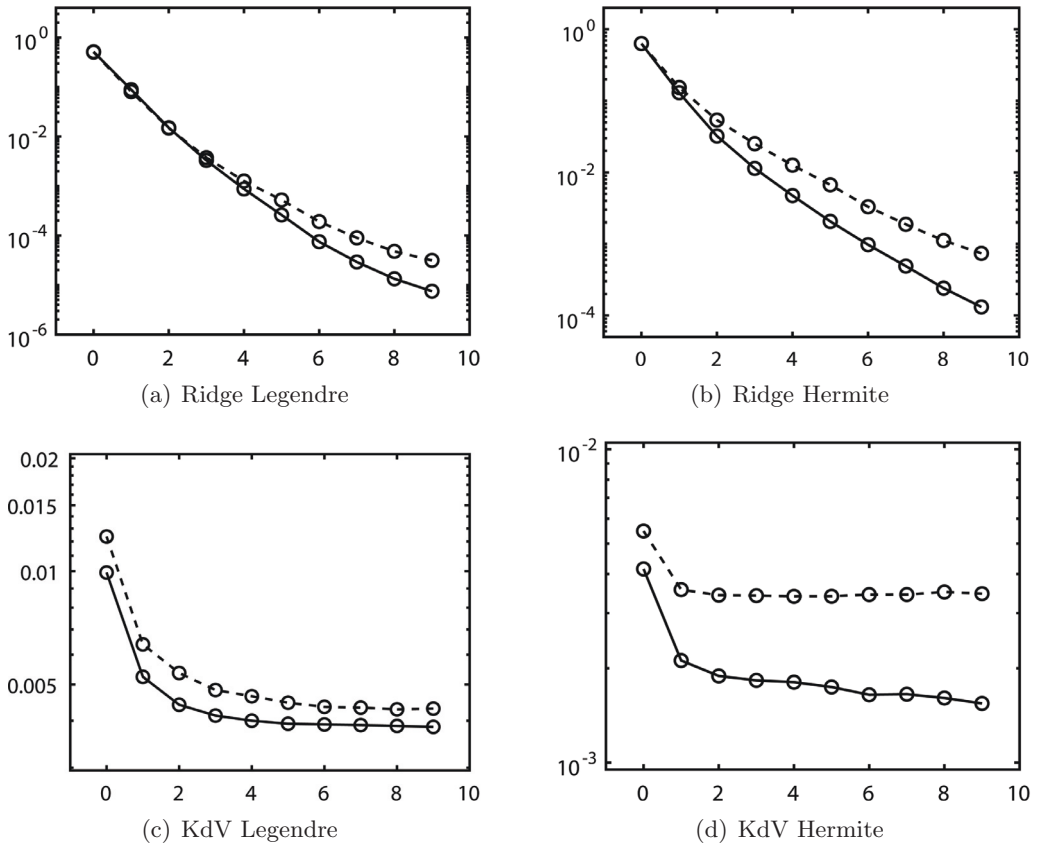


Figure 4.6. *Relative error vs. rotation. The size of* $\mathbf{\Psi}$ *is* 160×455 *in the ridge problem and* 160×1001 *in the KdV problem.*

Table 4.7

Computation time per each random realization, averaged over 20 *trials. (N/A means a certain case is not available.)*

Time (*sec.*)	Dimension	Rotations	Legendre	Hermite	Laguerre
Ridge function	160×455	9	6.53	4.31	16.19
Elliptic equation	220×816	3	5.26	11.96	N/A
KdV equation	160×1001	3	15.03	14.33	N/A
HD function	1000×5151	3	2041.82	2102.04	N/A

elliptic equation cases. Here, the ridge function has a low-dimensional structure under rotation which leads to a very sparse representation without truncation error, and we can investigate the efficiency of different algorithms at identifying this low-dimensional structure for different polynomials. On the other hand, the elliptic equation case does not have a very good low-dimensional structure under rotation, and the truncation error ε exists. Both factors imply that the elliptic equation is a more difficult problem for the rotation method. The comparison results shown in Figure 4.7 illustrate that both the ADMM and reweighted approaches provide similar results for the Legendre and Hermite polynomial expansions. The

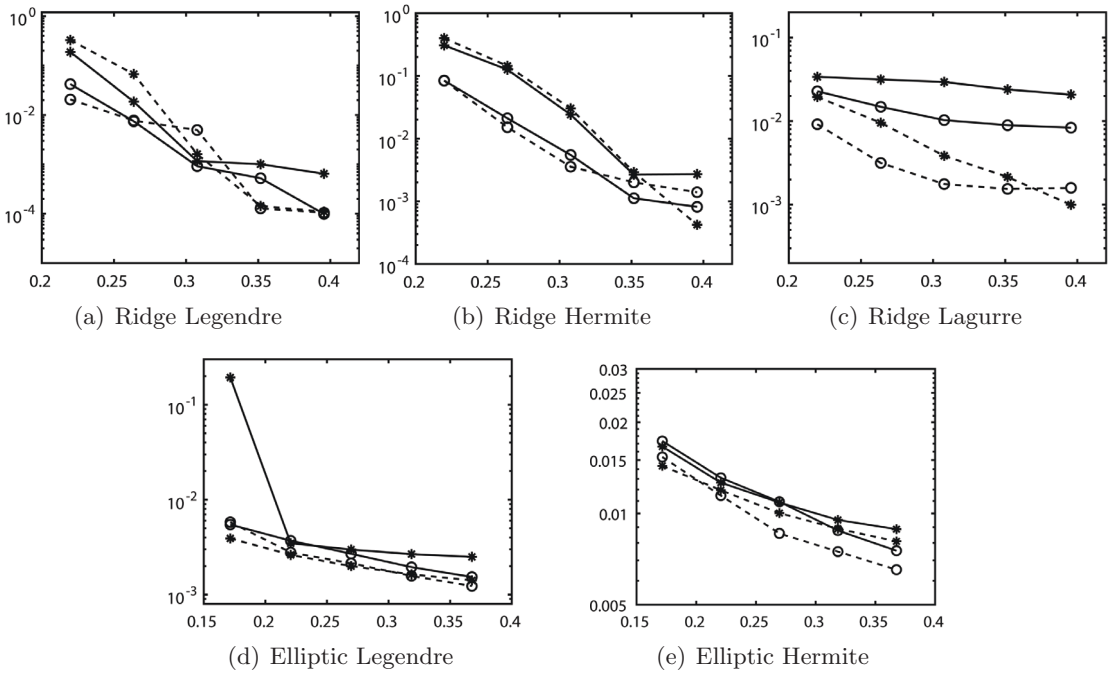


Figure 4.7. *Comparison between ADMM and reweighted frameworks. Relative error of Legendre, Hermite, and Lagurre polynomial expansions for ridge function and elliptic equation against ratio M/N. The dashed lines are generated by reweighted algorithms, and the solid lines are results from nonreweighted algorithms. "$\circ$" is the marker for ℓ_1 minimization, and "$*$" is the marker for $\ell_{1/2}$.*

improvement by the reweighted approaches is more significant for the Laguerre polynomial probably because its extremely high coherence (almost 1) leads to different behaviors of the minimization algorithms. Of note, the Legendre and Hermite polynomials are associated with the uniform and normal random variables, which are the most widely used random variables in scientific and engineering problems. For these two polynomials, the reweighted algorithms are only marginally better than ADMM despite high computational costs and the burden on parameter tuning. Therefore, we advocate the use of ADMM for solving the sparse coding problem.

5. Conclusions.

In this work, we proposed an alternating direction method to identify a rotation matrix iteratively in order to enhance the sparsity of gPC expansion. We followed this with several nonconvex minimization schemes to efficiently identify the sparse coefficients. We used a general framework to incorporate any regularization whose proximal operator can be found efficiently (including ℓ_1) into the rotational method. As such, our method improves the accuracy of the compressive sensing method in constructing the gPC expansions from a small amount of data. In particular, the rotation is determined by seeking the directions of maximum variation for the QoI through the SVD of the gradients at different points in the parameter space. The linear system after rotations becomes ill-conditioned—specifically, more coherent—which motivated us to choose the nonconvex method instead of the convex ℓ_1 approach for sparse recovery. We conducted extensive simulations under various scenarios,

including a ridge function, an elliptic equation, the KdV equation, and an HD function with Legendre, Hermite, and Laguerre polynomials, all of which are widely used in practice. Our experimental results demonstrated that the proposed combination of rotation estimation and nonconvex methods significantly outperforms the standard ℓ_1 minimization (without rotation). In different coherence scenarios, there are different nonconvex regularizations (combined with rotations) that outperform the rotational CS with the ℓ_1 approach. Specifically, ℓ_1-ℓ_2 and ERF work well for coherent systems, while $\ell_{1/2}$ excels in incoherent ones. These conclusions align with the observations in CS studies.

REFERENCES

[1] B. Adcock, *Infinite-dimensional compressed sensing and function interpolation*, Found. Comput. Math., 18 (2017), pp. 1–41.

[2] B. Adcock, S. Brugiapaglia, and C. G. Webster, *Compressed sensing approaches for polynomial approximation of high-dimensional functions*, in Compressed Sensing and Its Applications, 2017, Springer, pp. 93–124.

[3] N. Alemazkoor and H. Meidani, *Divide and conquer: An incremental sparsity promoting compressive sampling approach for polynomial chaos expansions*, Comput. Methods Appl. Mech. Eng., 318 (2017), pp. 937–956.

[4] I. Babuška, F. Nobile, and R. Tempone, *A stochastic collocation method for elliptic partial differential equations with random input data*, SIAM Rev., 52 (2010), pp. 317–355, https://doi.org/10.1137/100786356.

[5] A. S. Bandeira, E. Dobriban, D. G. Mixon, and W. F. Sawin, *Certifying the restricted isometry property is hard*, IEEE Trans. Inf. Theory, 59 (2013), pp. 3448–3450.

[6] G. Blatman and B. Sudret, *Adaptive sparse polynomial chaos expansion based on least angle regression*, J. Comput. Phys., 230 (2011), pp. 2345–2367.

[7] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Mach. Learn., 3 (2011), pp. 1–122.

[8] A. M. Bruckstein, D. L. Donoho, and M. Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Rev., 51 (2009), pp. 34–81, https://doi.org/10.1137/060657704.

[9] R. H. Cameron and W. T. Martin, *The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals*, Ann. Math., 48 (1947), pp. 385–392.

[10] E. J. Candès, *The restricted isometry property and its implications for compressed sensing*, C. R. Math. Acad. Sci. Paris, 346 (2008), pp. 589–592.

[11] E. J. Candès, J. K. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Commun. Pure Appl. Math., 59 (2006), pp. 1207–1223.

[12] E. J. Candès and M. B. Wakin, *An introduction to compressive sampling*, IEEE Signal Process. Mag., 25 (2008), pp. 21–30.

[13] E. J. Candès, M. B. Wakin, and S. P. Boyd, *Enhancing sparsity by reweighted l_1 minimization*, J. Fourier Anal. Appl., 14 (2008), pp. 877–905.

[14] R. Chartrand, *Exact reconstruction of sparse signals via nonconvex minimization*, IEEE Signal Process. Lett., 14 (2007), pp. 707–710.

[15] P. G. Constantine, E. Dow, and Q. Wang, *Active subspace methods in theory and practice: Applications to kriging surfaces*, SIAM J. Sci. Comput., 36 (2014), pp. A1500–A1524, https://doi.org/10.1137/130916138.

[16] W. Dai and O. Milenkovic, *Subspace Pursuit for Compressive Sensing: Closing the Gap between Performance and Complexity*, Tech. report, Illinois University at Urbana-Champaign, 2008.

[17] D. L. Donoho, *Compressed sensing*, IEEE Trans. Inf. Theory, 52 (2006), pp. 1289–1306.

[18] D. L. Donoho and M. Elad, *Optimally sparse representation in general (nonorthogonal) dictionaries via l^1 minimization*, Proc Natl. Acad. Sci. USA, 100 (2003), pp. 2197–2202.

[19] D. L. Donoho, M. Elad, and V. N. Temlyakov, *Stable recovery of sparse overcomplete representations in the presence of noise*, IEEE Trans. Inf. Theory, 52 (2006), pp. 6–18.
[20] A. Doostan and H. Owhadi, *A non-adapted sparse approximation of PDEs with stochastic inputs*, J. Comput. Phys., 230 (2011), pp. 3015–3034.
[21] O. G. Ernst, A. Mugler, H.-J. Starkloff, and E. Ullmann, *On the convergence of generalized polynomial chaos expansions*, ESAIM Math. Model. Numer. Anal., 46 (2012), pp. 317–339.
[22] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Birkhäuser Basel, 2013.
[23] R. G. Ghanem and P. D. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Springer-Verlag, 1991.
[24] R. Gribonval and M. Nielsen, *Sparse representations in unions of bases*, IEEE Trans. Inf. Theory, 49 (2003), pp. 3320–3325.
[25] L. Guo, J. Li, and Y. Liu, *Stochastic collocation methods via minimisation of the transformed l_1-penalty*, East Asian J. Appl. Math., 8 (2018), pp. 566–585.
[26] W. Guo, Y. Lou, J. Qin, and M. Yan, *A Novel Regularization Based on the Error Function for Sparse Recovery*, preprint, https://arxiv.org/abs/2007.02784, 2020.
[27] J. Hampton and A. Doostan, *Compressive sampling of polynomial chaos expansions: Convergence analysis and sampling strategies*, J. Comput. Phys., 280 (2015), pp. 363–386.
[28] J. Hampton and A. Doostan, *Basis adaptive sample efficient polynomial chaos (base-pc)*, J. Comput. Phys., 371 (2018), pp. 20–49.
[29] J. D. Jakeman, M. S. Eldred, and K. Sargsyan, *Enhancing ℓ_1-minimization estimates of polynomial chaos expansions using basis selection*, J. Comput. Phys., 289 (2015), pp. 18–34.
[30] J. D. Jakeman, A. Narayan, and T. Zhou, *A generalized sampling and preconditioning scheme for sparse approximation of polynomial chaos expansions*, SIAM J. Sci. Comput., 39 (2017), pp. A1114–A1144, https://doi.org/10.1137/16M1063885.
[31] M. Jardak, C.-H. Su, and G. E. Karniadakis, *Spectral polynomial chaos solutions of the stochastic advection equation*, J. Sci. Comput., 17 (2002), pp. 319–338.
[32] M.-J. Lai, Y. Xu, and W. Yin, *Improved iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization*, SIAM J. Numer. Anal., 51 (2013), pp. 927–957, https://doi.org/10.1137/110840364.
[33] H. Lei, X. Yang, Z. Li, and G. E. Karniadakis, *Systematic parameter inference in stochastic mesoscopic modeling*, J. Comput. Phys., 330 (2017), pp. 571–593.
[34] H. Lei, X. Yang, B. Zheng, G. Lin, and N. A. Baker, *Constructing surrogate models of complex systems with enhanced sparsity: Quantifying the influence of conformational uncertainty in biomolecular solvation*, Multiscale Model. Simul., 13 (2015), pp. 1327–1353, https://doi.org/10.1137/140981587.
[35] K.-C. Li, *Sliced inverse regression for dimension reduction*, J. Am. Stat. Assoc., 86 (1991), pp. 316–327.
[36] Y. Lou, and M. Yan, *Fast L_1–L_2 minimization via a proximal operator*, J. Sci. Comput., 74 (2018), pp. 767–785.
[37] Y. Lou, P. Yin, Q. He, and J. Xin, *Computing sparse representation in a highly coherent dictionary based on difference of L_1 and L_2*, J. Sci. Comput., 64 (2015), pp. 178–196.
[38] Y. Lou, P. Yin, and J. Xin, *Point source super-resolution via non-convex L_1 based methods*, J. Sci. Comput., 68 (2016), pp. 1082–1100.
[39] Z. Lu, *Iterative reweighted minimization methods for ℓ_p regularized unconstrained nonlinear programming*, Math. Program., 147 (2014), pp. 277–307.
[40] J. Lv and Y. Fan, *A unified approach to model selection and sparse recovery using regularized least squares*, Ann. Statist., 37 (2009), pp. 3498–3528.
[41] X. Ma and N. Zabaras, *An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations*, J. Comput. Phys., 229 (2010), pp. 3884–3915.
[42] B. K. Natarajan, *Sparse approximate solutions to linear systems*, SIAM J. Comput., 24 (1995), pp. 227–234, https://doi.org/10.1137/S0097539792240406.
[43] H. Ogura, *Orthogonal functionals of the Poisson process*, IEEE Trans. Inf. Theory, 18 (1972), pp. 473–481.
[44] J. Peng, J. Hampton, and A. Doostan, *A weighted ℓ_1-minimization approach for sparse polynomial chaos expansions*, J. Comput. Phys., 267 (2014), pp. 92–111.

[45] J. Peng, J. Hampton, and A. Doostan, *On polynomial chaos expansion via gradient-enhanced ℓ_1-minimization*, J. Comput. Phys., 310 (2016), pp. 440–458.
[46] Y. Rahimi, C. Wang, H. Dong, and Y. Lou, *A scale invariant approach for sparse signal recovery*, SIAM J. Sci. Comput., 41 (2019), pp. A3649–A3672, https://doi.org/10.1137/18M123147X.
[47] H. Rauhut and R. Ward, *Sparse legendre expansions via l_1-minimization*, J. Approx. Theory, 164 (2012), pp. 517–533.
[48] H. Rauhut and R. Ward, *Interpolation via weighted ℓ_1 minimization*, Appl. Comput. Harmon. Anal., 40 (2016), pp. 321–351.
[49] T. M. Russi, *Uncertainty Quantification with Experimental Data and Complex System Models*, Ph.D. thesis, UC Berkeley, 2010.
[50] K. Sargsyan, C. Safta, H. N. Najm, B. J. Debusschere, D. Ricciuto, and P. Thornton, *Dimensionality reduction for complex models via Bayesian compressive sensing*, Int. J. Uncertain. Quantif., 4 (2014), pp. 63–93.
[51] X. Shen, W. Pan, and Y. Zhu, *Likelihood-based selection and sharp parameter estimation*, J. Amer. Statist. Assoc., 107 (2012), pp. 223–232.
[52] S. Smolyak, *Quadrature and interpolation formulas for tensor products of certain classes of functions*, Soviet Math. Dokl., 4 (1963), pp. 240–243.
[53] M. A. Tatang, W. Pan, R. G. Prinn, and G. J. McRae, *An efficient method for parametric uncertainty analysis of numerical geophysical models*, J. Geophys. Res-Atmos., 102 (1997), pp. 21925–21932.
[54] A. M. Tillmann, and M. E. Pfetsch, *The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing*, IEEE Trans. Inf. Theory, 60 (2014), pp. 1248–1259.
[55] R. Tipireddy and R. Ghanem, *Basis adaptation in homogeneous chaos spaces*, J. Comput. Phys., 259 (2014), pp. 304–317.
[56] P. Tsilifis, X. Huan, C. Safta, K. Sargsyan, G. Lacaze, J. C. Oefelein, H. N. Najm, and R. G. Ghanem, *Compressive sensing adaptation for polynomial chaos expansions*, J. Comput. Phys., 380 (2019), pp. 29–47.
[57] C. Wang, M. Yan, Y. Rahimi, and Y. Lou, *Accelerated schemes for the L_1/L_2 minimization*, IEEE Trans. Signal Process., 68 (2020), pp. 2660–2669.
[58] D. Xiu and J. S. Hesthaven, *High-order collocation methods for differential equations with random inputs*, SIAM J. Sci. Comput., 27 (2005), pp. 1118–1139, https://doi.org/10.1137/040615201.
[59] D. Xiu and G. E. Karniadakis, *The Wiener–Askey polynomial chaos for stochastic differential equations*, SIAM J. Sci. Comput., 24 (2002), pp. 619–644, https://doi.org/10.1137/S1064827501387826.
[60] Z. Xu, X. Chang, F. Xu, and H. Zhang, *$L_{1/2}$ regularization: A thresholding representation theory and a fast solver*, IEEE Trans. Neural Netw. Learn. Syst., 23 (2012), pp. 1013–1027.
[61] Z. Xu and T. Zhou, *On sparse interpolation and the design of deterministic interpolation points*, SIAM J. Sci. Comput., 36 (2014), pp. A1752–A1769, https://doi.org/10.1137/13094596X.
[62] L. Yan, L. Guo, and D. Xiu, *Stochastic collocation algorithms using l_1-minimization*, Int J. Uncertain. Quantif., 2 (2012), pp. 279–293.
[63] X. Yang, D. A. Barajas-Solano, W. S. Rosenthal, and A. M. Tartakovsky, *PDF Estimation for Power Grid Systems via Sparse Regression*, preprint, https://arxiv.org/abs/1708.08378, 2017.
[64] X. Yang, M. Choi, G. Lin, and G. E. Karniadakis, *Adaptive ANOVA decomposition of stochastic incompressible and compressible flows*, J. Comput. Phys., 231 (2012), pp. 1587–1614.
[65] X. Yang and G. E. Karniadakis, *Reweighted ℓ_1 minimization method for stochastic elliptic differential equations*, J. Comput. Phys., 248 (2013), pp. 87–108.
[66] X. Yang, H. Lei, N. Baker, and G. Lin, *Enhancing sparsity of Hermite polynomial expansions by iterative rotations*, J. Comput. Phys., 307 (2016), pp. 94–109.
[67] X. Yang, H. Lei, P. Gao, D. G. Thomas, D. L. Mobley, and N. A. Baker, *Atomic radius and charge parameter uncertainty in biomolecular solvation energy calculations*, J. Chem. Theory Comput., 14 (2018), pp. 759–767.
[68] X. Yang, W. Li, and A. Tartakovsky, *Sliced-inverse-regression-aided rotated compressive sensing method for uncertainty quantification*, SIAM/ASA J. Uncertain. Quantif., 6 (2018), pp. 1532–1554, https://doi.org/10.1137/17M1148955.

[69] X. Yang, X. Wan, L. Lin, and H. Lei, *A general framework for enhancing sparsity of generalized polynomial chaos expansions*, Int. J. Uncertain. Quantif., 9 (2019), pp. 221–243.
[70] P. Yin, E. Esser, and J. Xin, *Ratio and difference of l_1 and l_2 norms and sparse representation with coherent dictionaries*, Commun. Inf. Syst., 14 (2014), pp. 87–109.
[71] P. Yin, Y. Lou, Q. He, and J. Xin, *Minimization of ℓ_{1-2} for compressed sensing*, SIAM J. Sci. Comput., 37 (2015), pp. A536–A563, https://doi.org/10.1137/140952363.
[72] S. Zhang and J. Xin, *Minimization of transformed L_1 penalty: Closed form representation and iterative thresholding algorithms*, Commun. Math. Sci., 15 (2017), pp. 511–537.
[73] S. Zhang and J. Xin, *Minimization of transformed L_1 penalty: Theory, difference of convex function algorithm, and robust application in compressed sensing*, Math. Program., 169 (2018), pp. 307–336.
[74] T. Zhang, *Multi-stage convex relaxation for learning with sparse regularization*, in Advances in Neural Information Processing Systems (NIPS), 2009, pp. 1929–1936.
[75] Z. Zhang, X. Yang, I. V. Oseledets, G. E. Karniadakis, and L. Daniel, *Enabling high-dimensional hierarchical uncertainty quantification by ANOVA and tensor-train decomposition*, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., 34 (2015), pp. 63–76.