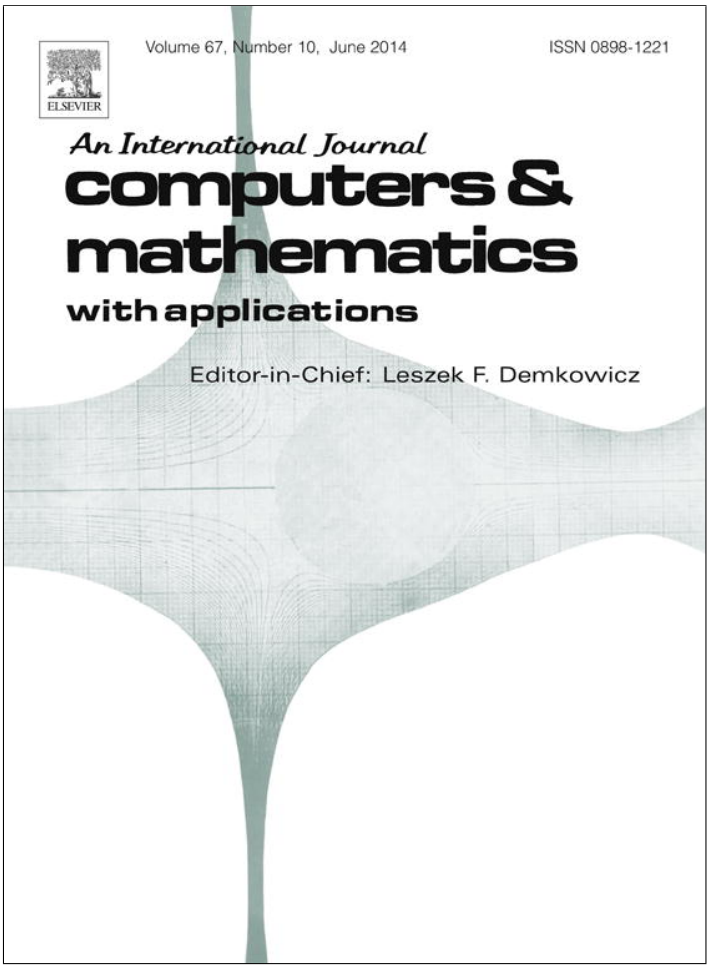


Computers and Mathematics with Applications 67 (2014) 1960–1977

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Numerical solution of high dimensional stationary Fokker–Planck equations via tensor decomposition and Chebyshev spectral differentiation

Yifei Sun [*,1], Mrinal Kumar [2]

Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250, USA

ARTICLE INFO

Article history:
Received 6 August 2013
Received in revised form 25 March 2014
Accepted 21 April 2014
Available online 9 May 2014

Keywords:
Numerical partial differential equation
Tensor decomposition
Chebyshev spectral method
Curse of dimensionality
Separation of variables
Fokker–Planck equation

ABSTRACT

This paper focuses on the curse of dimensionality in the numerical solution of the stationary Fokker–Planck equation for systems with state-independent excitation. A tensor decomposition approach is combined with Chebyshev spectral differentiation to drastically reduce the number of degrees of freedom required to maintain accuracy as dimensionality increases. Following the enforcement of the normality condition via a penalty method, the discretized system is solved using alternating least squares algorithm. Numerical results for a variety of systems, including separable/non-separable systems, linear/nonlinear systems and systems with/without closed-form stationary solutions up to ten dimensional state-space are presented to illustrate the effectiveness of the proposed method.

1. Introduction

The problem of uncertainty propagation through nonlinear dynamical systems subject to stochastic excitation and uncertain initial conditions has been extensively studied. Among all pertinent techniques for this problem, the Monte Carlo method is probably the simplest yet most robust. Its slow rate of convergence is however well documented, causing it to become computationally burdensome as the underlying dimensionality increases. On the other hand, truncation based methods such as equivalent linearization and/or higher order moment closure are suitable only for moderately nonlinear systems [1].

The exact description of the uncertainty propagation problem through nonlinear systems with white noise excitation is given by the Fokker–Planck equation (FPE) [2]. However, its closed-form analytical solution (especially for the transient FPE) exists only for a handful of systems and numerical techniques must be utilized. It has been widely observed that to maintain accuracy in traditional discretization based numerical methods such as finite element [3] or finite difference, the number of degrees of freedom (DOF) of the approximation, i.e. the number of unknowns grows exponentially as the dimensionality of the underlying state-space increases. This so-called curse of dimensionality [4] fundamentally limits the use of FPE for uncertainty quantification in high dimensional systems [5]. Recently, the meshless partition of unity finite element method (PUFEM) was employed to counter this problem with moderate success [5–7]. While the unstructured node-based approximation paradigm of PUFEM simplifies discretization of high dimensional spaces, the number of basis functions needed still

* Corresponding author.
E-mail addresses: yfsun@ufl.edu (Y. Sun), mrinalkumar@ufl.edu (M. Kumar).
1 Graduate Research Assistant.
2 Assistant Professor.

http://dx.doi.org/10.1016/j.camwa.2014.04.017
0898-1221/

increases rapidly with dimensionality. This can be attributed to the factorial growth of a complete set of polynomial basis functions.

In the related literature, the representation of data in a tensor product structure has been recognized as a key for managing the dimensionality issue [8]. Methods based on tensor decompositions e.g., CANDECOMP/PARAFAC decomposition (CPD) and Tucker decomposition have become well-established in a wide range of applications [9–11]. The essence of their application to numerical multidimensional partial differential equations (PDEs) [12–15] lies in the "decoupling" or "separation" of dimensions, following which expensive high dimensional operations are decomposed into a series of simple one-dimensional operations. As an outcome, the complexity scales linearly rather than exponentially with dimensionality. The price to pay is that it transforms a linear problem into a nonlinear problem (if the original PDE is linear), or exacerbates the nonlinearity of the problem (if the original PDE is nonlinear). It is important to understand the source of nonlinearity in this approach since both the operator of the PDE and the unknown function to be approximated must be expressed in tensor product form. Writing the operator in tensor product structure does not cause nonlinearity. On the other hand, using the CPD form for the unknown function does, but is also responsible for ameliorization of the curse of dimensionality.

Beylkin et al. [12] solved the Schrödinger equation (as well as certain other linear PDEs) with solution approximated in the CPD form and used the alternating least squares (ALS) algorithm for the discretized nonlinear system. This scheme was extended to stochastic PDEs in [16]. Ballani et al. [13] used a hierarchical Tucker decomposition to approximate the unknown function, which was then solved using a projection method. Khoromskij et al. proposed methods based on the quantics tensor train (QTT) format for solving the chemical master equation [17] and high dimensional numerical modeling [18]. The finite element approximation was used in their tensor methods and due to the QTT structure, logarithmic complexity was demonstrated. An ALS multigrid algorithm was presented in Ref. [19] to accelerate the convergence of tensor approximation for multi-dimensional problems. In Refs. [15,20], the so-called proper generalized decomposition (PGD) method was developed, in which the approximation was constructed in CPD form through a sequence of enrichment steps using the finite element method. It is important to note that at each enrichment step, only a single component rank-one tensor was obtained. Several variants of PGD exist, e.g. see Ref. [21], where a Tucker decomposition structure is adopted. In Ref. [15] and subsequent papers by the same authors, PGD was used to solve high dimensional FPEs for multi-bead-spring models of polymer chains encountered in the kinetic theory of complex fluids. Compared with simple discretization based methods such as finite element/finite difference for this problem, an improvement of several orders of magnitude was shown in terms of the degrees of freedom of the approximation. However, the application of PGD to problems outside the domain of kinetic theory of complex fluids is limited and in the experience of the current authors not very successful.

The current paper is concerned with Fokker–Planck equations commonly encountered in the field of nonlinear vibrations and is the extension of [22]. In particular, the stationary FPE is solved by combining the CPD structure with the Chebyshev spectral differentiation framework. This approach benefits from the superior performance of the Chebyshev spectral approach over other differentiation methods for smooth functions on domains of regular geometry [23,24]. The normality constraint is enforced via a penalty method, following which all component rank-one tensors are obtained simultaneously at each enrichment step using the alternating least squares method of [12]. It is shown through numerical examples that the total number of degrees of freedom scales favorably, which is a significant result for the FPE.

The remainder of this paper is organized as follows: Section 2 introduces the concepts relevant to the separation of spatial dimensions. The Chebyshev spectral method and its benefits are discussed in Section 3. Section 4 describes in detail the proposed method and numerical examples are provided in Section 5, including systems up to ten dimensional state-space. Finally, a summary and future research directions are provided in Section 6.

2. Separation of spatial dimensions

The origin of the classical separation of variables method dates back 200 years when Joseph Fourier studied the heat equation. Traditionally used for solving PDEs of certain simple forms, the basic idea of separating the dimensions is now being recognized as a key for breaking the curse of dimensionality [8]. In this section, we revisit some of the essential concepts of separated representation (here confined to the continuous case) and tensor decomposition (for the discrete case).

2.1. Separated representation

In the present context, separated representation refers to the expression of a multivariate function as a summation of separable functions. Given a smooth scalar function F defined on a compact subset of P-dimensional Euclidean space $\mathbb{R}^P$, one can approximate F as

$$F(x_1, x_2, \ldots, x_P) = \sum_{l=1}^{R_\epsilon} \prod_{d=1}^{P} f_d^l(x_d) + \mathcal{O}(\epsilon). \tag{1}$$

The question of existence of the above representation can be examined, for instance, by considering the multiple Fourier sine series of F: let $\mathbb{X} := [0, X_1] \times [0, X_2] \times \cdots \times [0, X_P]$ and $\mathbf{n} = (n_1, n_2, \ldots, n_P)$ be a vector of integers, then for a smooth

function $F : \mathbb{X} \to \mathbb{R}$,

$$F(x_1, x_2, \ldots, x_P) = \sum_{\mathbf{n}\in\mathbb{N}^P} B_{\mathbf{n}} \prod_{d=1}^{P} \sin\left(\frac{\pi n_d x_d}{X_d}\right), \tag{2}$$

where the multiple Fourier sine coefficients are given by

$$B_{\mathbf{n}} = \frac{2^P}{\prod_{d=1}^{P} X_d} \int_{\mathbb{X}} F(\mathbf{x}) \prod_{d=1}^{P} \sin\left(\frac{\pi n_d x_d}{X_d}\right) d\mathbf{x}. \tag{3}$$

It is evident that the computation of $B_{\mathbf{n}}$ requires high dimensional integration. The advantage of separated representation of Eq. (1) is that once it is determined, algebraic operations on F (such as Eq. (3)) can be decoupled into P single dimensional operations. Consequently, computational complexity grows linearly rather than exponentially with respect to dimensionality, as long as there is no dependence of R_ϵ (the number of summands) on dimensionality P.

2.2. Tensor decomposition

Viewed as a generalization of a matrix in two dimensions, a tensor of order P and size $n_1 \times n_2 \times \cdots \times n_P$ is a P dimensional array denoted by $\mathcal{F} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_P}$. In numerous applications, multivariate functions are discretized using tensor grids, the storage alone of which is difficult owing to the exponential growth ($\prod_{d=1}^{P} n_d$) of its elements with respect to P. One remedy is to decompose the tensor into a summation of several rank-one tensors. A tensor $\mathcal{F} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_P}$ is of rank-one if

$$\mathcal{F} = f_1 \otimes f_2 \otimes \cdots \otimes f_P, \tag{4}$$

where f_d are $n_d \times 1$ vectors for $d = 1, 2, \ldots, P$ and "$\otimes$" is the standard tensor product:

$$\mathcal{F}_{i_1 i_2 \ldots i_P} = \prod_{d=1}^{P} f_d(i_d) \tag{5}$$

for all $1 \le i_d \le n_d$. The following important tensor decomposition of $\mathcal{F}$ was introduced by [25,26], named CANDECOMP/PARAFAC decomposition (CPD):

$$\mathcal{F} \approx \sum_{r=1}^{R} \bigotimes_{d=1}^{P} f_d^r. \tag{6}$$

Note that the above decomposition represents an approximation. The minimum R for which the "$\approx$" can be replaced with "$=$" is called the rank of tensor $\mathcal{F}$. We define factor matrices of the above decomposition as $F_d = \begin{bmatrix} f_d^1 & f_d^2 & \cdots & f_d^R \end{bmatrix}$, $d = 1, 2, \ldots, P$. Once the CPD is found, a tensor of order P can be reconstructed using its P factor matrices. For a prescribed accuracy, the total number of elements in the above approximation ($R \sum_{d=1}^{P} n_d$) grows linearly with P, provided that R does not exhibit explosive growth. Fortunately it is observed that in many applications, a tensor of high order can be very well approximated by CP decompositions of relatively low rank, although no theoretical results are available [27]. This makes the CPD structure a good candidate for numerically dealing with high dimensional PDEs.

The problem of determining the rank of a tensor is NP-hard [28]. Moreover, finding the *best* low-rank approximation is an ill-posed problem [9]. The selection of R for CPD is therefore a tradeoff between accuracy and computational efficiency.

2.2.1. Alternating least squares

Alternating least squares (ALS) is a commonly used method for computing the factor matrices of a CP decomposition. For a tensor $\mathcal{G} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_P}$, the problem of finding its CPD with prescribed R can be formulated as the following optimization problem:

$$\min_{\{f_d^r\}} \mathcal{R}, \qquad \mathcal{R} = \left\| \mathcal{G} - \sum_{r=1}^{R} \bigotimes_{d=1}^{P} f_d^r \right\|_F^2, \tag{7}$$

where Frobenius norm is used, and the unknowns are the vectors f_d^r, $d = 1, 2, \ldots, P$ and $r = 1, 2, \ldots, R$ or equivalently the factor matrices F_d. Although Eq. (7) appears to be a standard nonlinear least squares problem whose objective function is a $2P$th degree polynomial with $R \sum_{d=1}^{P} n_d$ variables, there exist no well established algorithms for its solution [27]. Among existing techniques, alternating least squares (ALS) is arguably the most popular, underlying which is the basic idea of holding all factor matrices except one constant, thus translating the original nonlinear least squares problem into a series of sequential linear least squares problems. The scheme of ALS for the problem in Eq. (7) is as follows:

- Initialize (for example, randomly) $F_d \in \mathbb{R}^{n_d\times R}$, $d = 1, 2, \ldots, P$.

- Cycle for $d = 1, 2, \ldots, P$ until $\sqrt{\frac{\mathcal{R}}{\prod_{d=1}^{P} n_d}} < \epsilon$ or the maximum number of iterations is reached:

$$F_d \leftarrow \mathcal{G}_{(d)}[(F_P \odot \cdots \odot F_{d+1} \odot F_{d-1} \odot \cdots \odot F_1)^T]^\dagger \tag{8}$$

  end of cycle.
- Return F_d for $d = 1, 2, \ldots, P$

where $(\cdot)_{(d)}$ is the mode-d matricization of a tensor, "$\odot$" is the Khatri-Rao product and $(\cdot)^\dagger$ denotes the Moore–Penrose pseudoinverse of a matrix. It should be noted that although a convergence proof of ALS (even to a local minimum) is not yet available, it can be shown that the objective function of Eq. (7) decreases monotonically [12]. The rate of convergence is often a concern especially as dimensionality increases, leading to problems such as "bottlenecks", "swamps" and "CP-degeneracy" [27] discussed further in the results section. For a comprehensive introduction to tensor decompositions, the reader is referred to the review paper [9].

3. Spectral differentiation

There are two main ingredients of the proposed approach: (i) discretization of the Fokker–Planck operator via spectral differentiation and, (ii) CPD representation of the unknown probability density function (pdf), obtained using a generalized version of the ALS algorithm described above. As previously mentioned, CPD representation of the unknown pdf necessarily requires a tensor product representation of the discretized FP operator. In general, the operator $\mathcal{L}(\cdot)$ of a linear PDE can be approximated in tensor product form through discretization in the following manner:

$$\mathcal{L} \approx \sum_{l=1}^{R_A} \bigotimes_{d=1}^{P} A_d^l, \tag{9}$$

where A_d^l are square matrices, which in order to be compatible with Eq. (6) must be of size $n_d \times n_d$. By virtue of the above separated form, high dimensional operations involving $\mathcal{L}(\cdot)$ can now be decoupled into simple one-dimensional operations on A_d^l. Numerical discretization techniques (most commonly the finite difference method) are used to determine the above form. It is most desirable if the linear operator can be approximated accurately by a small sized tensor, which is crucial for reducing the computational burden as dimensionality increases. In the current application of solving the stationary FPE admitting smooth solution on a hypercuboid, a much better choice for spatial differentiation is the Chebyshev spectral method [29].

3.1. Finite differences

The traditional finite difference method can be understood as using overlapping local polynomials of relatively low degree to approximate the unknown function. Consequently, its derivatives at a grid point are approximated by the derivatives of its local interpolating polynomial. Higher order accuracy can be obtained by choosing a stencil of a greater number of points. In an extreme scenario, all existing grid points may contribute to the derivatives at a point, which is equivalent to the use of a global polynomial interpolant of very high order. Of course, this approach is susceptible to the damaging Runge phenomenon [29]. However, the idea of "extremely high order finite difference" can provide excellent results if the error of polynomial interpolation is carefully controlled. Consider the Cauchy interpolation error theorem [30]:

Theorem 1. *Let $x_0, x_1, \ldots, x_n$ be distinct real numbers, and let f be a given real valued function with $n+1$ continuous derivatives on the interval $I_z = \mathcal{H}\{z, x_0, \ldots, x_n\}$ (the smallest interval containing all of the real numbers $z, x_0, \ldots, x_n$), where z is some given real number. Then there exists $\xi \in I_z$ such that:*

$$f(z) - \sum_{j=0}^{n} f(x_j) l_j(z) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{j=0}^{n} (z - x_j). \tag{10}$$

In Theorem 1, $\sum_{j=0}^{n} f(x_j) l_j(z)$ denotes the interpolating polynomial of f on $x_0, x_1, \ldots, x_n$. Note that there is no way to control the effect of $f^{(n+1)}(\xi)$ on the interpolation error since f is problem dependent. On the other hand, the user has complete control on the selection of the interpolation points $x_0, x_1, \ldots, x_n$. In this work, they are determined via the Chebyshev spectral method, which essentially minimizes the maximum of the monic polynomial $\prod_{j=0}^{n} (z - x_j)$ on I_z.

3.2. Chebyshev spectral method

The problem of finding the optimal interpolation points was addressed by the Chebyshev minimal amplitude theorem given below [24]:

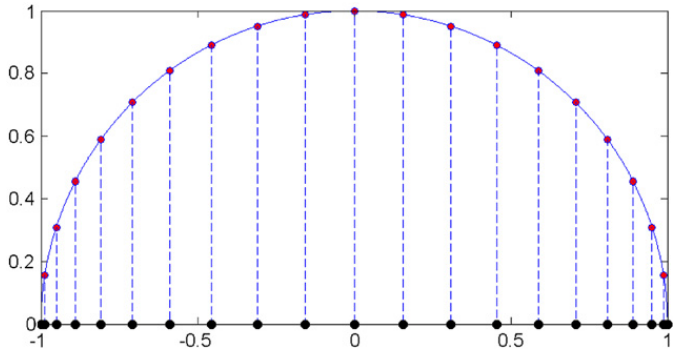


Fig. 1. Unevenly distributed Chebyshev extrema points (black dots).

Theorem 2. *Of all monic polynomials of degree N, the unique polynomial with the least maximum on* $[-1, 1]$ *is* $\frac{T_N(z)}{2^{N-1}}$*, the* N^{th} *Chebyshev polynomial divided by* 2^{N-1}*. In other words, all monic polynomials of the same degree satisfy the inequality:*

$$\max_{z\in[-1,1]} \left| \prod_{j=0}^{n} (z - x_j) \right| \geq \max_{z\in[-1,1]} \left| \frac{T_N(z)}{2^{N-1}} \right| = \frac{1}{2^{N-1}}. \tag{11}$$

Comparing Theorems 1 and 2, it is clear that the roots of the Chebyshev polynomial can be chosen as optimal interpolation points. Moreover, it can be proved that extrema of the Chebyshev polynomial provide the same interpolation error bound as its roots [24]. Here we simply use the Chebyshev extrema given by

$$x_j = \cos\left(\frac{j\pi}{N}\right), \quad j = 0, 1, \ldots, N. \tag{12}$$

The above points can be viewed as the projection of equally distributed nodes on the upper half of a unit circle onto the horizontal line passing through the center of this circle, as is illustrated by Fig. 1, where the black dots are the (unevenly distributed) Chebyshev extrema.

Derivatives can be approximated by differentiating the interpolating polynomial at the desired point. For instance, the first order differentiation matrix is given by the following theorem [23]:

Theorem 3. *For each* $N \geq 1$*, let the rows and columns of the* $(N + 1) \times (N + 1)$ *first order Chebyshev spectral differentiation matrix* D_N *be indexed from 0 to N. The entries of this matrix are*

$$(D_N)_{00} = \frac{2N^2 + 1}{6}, \qquad (D_N)_{NN} = -\frac{2N^2 + 1}{6},$$
$$(D_N)_{jj} = \frac{-x_j}{2(1 - x_j^2)}, \quad j = 1, \ldots, N - 1,$$
$$(D_N)_{ij} = \frac{c_i(-1)^{i+j}}{c_j(x_i - x_j)}, \quad i \neq j, i, j = 1, \ldots, N - 1,$$

where $c_i = \begin{cases} 2 & i = 0 \text{ or } N, \\ 1 & \text{otherwise.} \end{cases}$

The accuracy of Chebyshev spectral differentiation is given by [23]:

Theorem 4. *Suppose u is analytic on and inside the ellipse with foci* ± 1 *on which the Chebyshev potential takes the value* ϕ_f*, that is, the ellipse whose semi-major and semi-minor axis lengths sum to* $K = e^{\phi_f + \log 2}$*. Let* w *be the* vth *Chebyshev spectral derivative of* u $(v \geq 1)$*. Then*

$$|w_j - u^{(v)}(x_j)| = \mathcal{O}\left(e^{-N(\phi_f + \log 2)}\right) = \mathcal{O}(K^{-N}) \tag{13}$$

as $N \to \infty$*.*

In Theorem 4, the Chebyshev potential on complex plane is defined as

$$\phi(z) = \int_{-1}^{1} \frac{1}{\pi\sqrt{1 - x^2}} \log|z - x|\, dx. \tag{14}$$

If the assumption in Theorem 4 holds (as in the current application), the spectral derivative converges geometrically. In other words, a differentiation matrix of significantly smaller size will be sufficient to approximate the same one-dimensional

operator with equal accuracy as compared with the finite difference method. This is crucial for amelioration of the curse of dimensionality in the FPE as will be illustrated via examples in Section 5.

We note that as opposed to the attractive band-limited differentiation matrix of the finite difference method, the Chebyshev spectral differentiation matrix is dense. However, since only one-dimensional differentiation is required for the tensor decomposition based method used here, the dense structure fails to have an adverse impact.

4. A Chebyshev spectral tensor decomposition solution of the stationary FPE

In this section, we outline the use of Chebyshev spectral differentiation to discretize the Fokker–Planck operator. The CPD form is used for tensorization and the alternating least squares method [12] is used to solve the resulting discrete system while incorporating the normality constraint for a valid probability density function. It is shown through examples that the number of degrees of freedom of the approximation grows benignly with dimensionality, which is in sharp contrast with the exponential growth observed in traditional finite difference/element methods.

4.1. Fokker–Planck equation

Consider a nonlinear dynamical system perturbed by white noise excitation with initial condition uncertainty, modeled by the following Itô stochastic differential equation:

$$d\mathbf{x} = \mathbf{f}(t, \mathbf{x})dt + \mathbf{g}d\mathbf{B}(t), \quad \mathbf{x} \in \mathbb{R}^P, \tag{15}$$

where $\mathbf{B}(t)$ denotes an M-dimensional Brownian motion process with correlation function $\mathbf{Q}\delta(t_1 - t_2)$, $\mathbf{f}(t, \mathbf{x}) : [0, \infty) \times \mathbb{R}^P \to \mathbb{R}^P$ is a deterministic vector function and in the current paper, $\mathbf{g}$ is a constant noise influence matrix of size $(P \times M)$. The uncertainty in the state $\mathbf{x}(t)$ is quantified by its time varying probability density function (pdf) $\mathcal{W}(t, \mathbf{x})$. Initial condition uncertainty is prescribed as $\mathcal{W}(t_0, \mathbf{x}) = \mathcal{W}_0(\mathbf{x})$. The Fokker–Planck equation [2] corresponding to Eq. (15) is a PDE that captures the time evolution of the state pdf:

$$\frac{\partial}{\partial t}\mathcal{W}(t, \mathbf{x}) = \mathcal{L}_{\mathcal{FP}}[\mathcal{W}(t, \mathbf{x})], \tag{16}$$

where $\mathcal{L}_{\mathcal{FP}}$ is the Fokker–Planck operator given by

$$\mathcal{L}_{\mathcal{FP}} = \left[-\sum_{i=1}^{\mathrm{P}} \frac{\partial}{\partial x_i}\mathrm{D}_i^{(1)}(\cdot) + \sum_{i=1}^{\mathrm{P}}\sum_{j=1}^{\mathrm{P}} \frac{\partial^2}{\partial x_i \partial x_j}\mathrm{D}_{ij}^{(2)}(\cdot) \right], \tag{17}$$

$$\mathrm{D}^{(1)}(t, \mathbf{x}) = \mathbf{f}(t, \mathbf{x}), \qquad \mathrm{D}^{(2)} = \frac{1}{2}\mathbf{g}\mathrm{Q}\mathbf{g}^{\mathrm{T}}.$$

In Eq. (17), $\mathrm{D}^{(1)}$ and $\mathrm{D}^{(2)}$ are called drift coefficient vector and diffusion coefficient matrix respectively. Under certain conditions (in addition to the time invariance of $\mathbf{f}$ and $\mathbf{g}$: see [2]), there exists a unique global asymptotically stable steady state pdf $\mathcal{W}_s(\mathbf{x})$ that solves the following simplified form of Eq. (16):

$$\mathcal{L}_{\mathcal{FP}}[\mathcal{W}(\mathbf{x})] = 0. \tag{18}$$

The above equation, called the stationary Fokker–Planck equation, is the concern of the current paper.

4.2. Representation in tensor product form

To begin tensorization of the FP operator, we assume that $\mathbf{f}(\mathbf{x})$ is separable so that the drift vector can be expressed without error in the following separable form:

$$\mathrm{D}_i^{(1)}(t, \mathbf{x}) = \mathbf{f}_i(\mathbf{x}) = \sum_{j}^{R_{1i}} \prod_{d=1}^{P} f_d^j(x_d) \tag{19}$$

where R_{1i} is finite. The FPE is to be solved over a P-dimensional hypercuboid, of which the dth dimension is discretized by the vector $\mathbf{x}_d^v$ of nodes. Denote by D_{x_d} the Chebyshev spectral differentiation matrix of $\frac{\partial}{\partial x_d}$ on the solution domain. The drift coefficients can then be given in tensor form as follows:

$$\begin{aligned}\frac{\partial}{\partial x_i}\mathrm{D}_i^{(1)} \approx \sum_j &\left[\bigotimes_{d=1}^{i-1} \mathrm{diag}(f_d^j(\mathbf{x}_d^v))\right] \otimes \left[\mathrm{diag}(D_{x_i} f_i^j(\mathbf{x}_i^v))\right] \\ &\otimes \left[\bigotimes_{d=i+1}^{P} \mathrm{diag}(f_d^j(\mathbf{x}_d^v))\right] + \sum_j \left[\bigotimes_{d=1}^{i-1} \mathrm{diag}(f_d^j(\mathbf{x}_d^v))\right] \\ &\otimes \left[\mathrm{diag}(f_i^j(\mathbf{x}_i^v))D_{x_i}\right] \otimes \left[\bigotimes_{d=i+1}^{P} \mathrm{diag}(f_d^j(\mathbf{x}_d^v))\right].\end{aligned}$$

If $\mathbf{f}(\mathbf{x})$ is not separable, it can be approximated by separable functions either "continuously" via the multiple Fourier series described in Section 2.1 or "discretely" via CP decomposition discussed in Section 2.2. In such case, the second "=" of Eq. (19) must be replaced with "≈".

4.3. CPD approximation of the discretized FPE

Once all the terms in the FP operator are tensorized as shown in the above section, the stationary FPE is reduced to the following linear algebraic form:

$$\mathbb{A}\mathcal{U} = \mathcal{G}, \tag{20}$$

where, $\mathcal{U}$ is the approximation of the unknown stationary pdf $\mathcal{W}(\mathbf{x})$. Due to the homogeneous nature of the FPE, $\mathcal{G}$ (also in a CP decomposition structure) is identically zero. In order to handle the curse of dimensionality, the vector $\mathcal{U}$ is sought in a CP decomposition form as described in Section 2.1:

$$\mathcal{W}(\mathbf{x}) \approx \mathcal{U}(\mathbf{x}) = \sum_{l=1}^{R_U} \prod_{d=1}^{P} u_d^l(x_d). \tag{21}$$

Due to the special structure of the above decomposition, Eq. (20) is nonlinear in terms of the component functions $u_d^l(x_d)$ despite the fact that it is clearly linear in terms of the unknown tensor $\mathcal{U}$.

Solution schemes for Eq. (20) first appeared in [12], wherein two different types of methods were proposed. The first approach uses an iterative inversion scheme similar to the ones used for linear systems with a large sparse matrix, combined with simultaneous approximation of the intermediate solution by a tensor of lower rank to control the size growth of $\mathcal{U}$. The second method casts Eq. (20) as an equation error minimization problem and uses ALS to solve it. This can be understood as a generalized CP decomposition problem (Eq. (20) is the standard CP decomposition problem when $\mathbb{A} = \mathbb{I}$.). This paper uses a modified version of the second approach to suit the current application.

The developments below describe the solver for the general case of Eq. (20), which in addition to Eq. (21), has $\mathbb{A} = \sum_{i_A=1}^{R_A} \bigotimes_{d=1}^{P} A_d^{i_A}$ and $\mathcal{G} = \sum_{i_G=1}^{R_G} \bigotimes_{d=1}^{P} g_d^{i_G}$; where $A_d^{i_A}$ are $n_d \times n_d$ matrices while $u_d^{i_U}$ and $g_d^{i_G}$ are $n_d \times 1$ vectors. Transforming Eq. (20) into an optimization problem, we get

$$\min_{\{u_k^r\}} \| \mathbb{A}\mathcal{U} - \mathcal{G} \|_F^2, \tag{22}$$

and define $\mathcal{R} = \| \mathbb{A}\mathcal{U} - \mathcal{G} \|_F^2$, such that $\mathcal{R} = \langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle - 2\langle \mathbb{A}\mathcal{U}, \mathcal{G} \rangle + \langle \mathcal{G}, \mathcal{G} \rangle$, where,

$$\mathbb{A}\mathcal{U} = \sum_{i_A=1}^{R_A} \sum_{i_U=1}^{R_U} \bigotimes_{d=1}^{P} (A_d^{i_A} u_d^{i_U}),$$

$$\langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle = \sum_{i_A=1}^{R_A} \sum_{i_U=1}^{R_U} \sum_{j_A=1}^{R_A} \sum_{j_U=1}^{R_U} \prod_{d=1}^{P} \langle A_d^{i_A} u_d^{i_U}, A_d^{j_A} u_d^{j_U} \rangle, \quad \text{and}$$

$$\langle \mathbb{A}\mathcal{U}, \mathcal{G} \rangle = \sum_{i_A=1}^{R_A} \sum_{i_U=1}^{R_U} \sum_{i_G=1}^{R_G} \prod_{d=1}^{P} \langle A_d^{i_A} u_d^{i_U}, g_d^{i_G} \rangle.$$

The necessary condition for minimization of R is

$$\frac{\partial \langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle}{\partial u_k^r} - 2\frac{\partial \langle \mathbb{A}\mathcal{U}, \mathcal{G} \rangle}{\partial u_k^r} = 0, \tag{23}$$

for $k = 1, 2, \ldots, P$ and $r = 1, 2, \ldots, R_U$; where

$$\frac{\partial \langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle}{\partial u_k^r} = \sum_{i_A=1}^{R_A} \sum_{j_A=1}^{R_A} \sum_{j_U=1}^{R_U} (A_k^{j_A} u_k^{j_U})^T A_k^{i_A} \prod_{d \neq k} \langle A_d^{i_A} u_d^{i_U}, A_d^{j_A} u_d^{j_U} \rangle + \sum_{i_A=1}^{R_A} \sum_{i_U=1}^{R_U} \sum_{j_A=1}^{R_A} (A_k^{i_A} u_k^{i_U})^T A_k^{j_A} \prod_{d \neq k} \langle A_d^{i_A} u_d^{i_U}, A_d^{j_A} u_d^{j_U} \rangle,$$

$$\frac{\partial \langle \mathbb{A}\mathcal{U}, \mathcal{G} \rangle}{\partial u_k^r} = \sum_{i_A=1}^{R_A} \sum_{i_G=1}^{R_G} (g_k^{i_G})^T A_k^{i_A} \prod_{d \neq k} \langle A_d^{i_A} u_d^{i_U}, g_d^{i_G} \rangle.$$

Collecting terms in Eq. (23) for all r's and fixed dimension k, we have

$$\underbrace{\begin{pmatrix} \mathbf{M}_{1,1} & \cdots & \mathbf{M}_{1,R_U} \\ \vdots & \ddots & \vdots \\ \mathbf{M}_{R_U,1} & \cdots & \mathbf{M}_{R_U,R_U} \end{pmatrix}}_{=\mathbf{M}} \begin{pmatrix} u_k^1 \\ \vdots \\ u_k^{R_U} \end{pmatrix} = \underbrace{\begin{pmatrix} \mathbf{N}_1 \\ \vdots \\ \mathbf{N}_{R_U} \end{pmatrix}}_{=\mathbf{N}}, \tag{24}$$

where $\mathbf{M}_{i,j}$ and $\mathbf{N}_i$ are submatrices of the block matrices $\mathbf{M}$ and $\mathbf{N}$ respectively, given by

$$\mathbf{M}_{i,j} = \sum_{i_A=1}^{R_A} \sum_{j_A=1}^{R_A} (A_k^{j_A})^T A_k^{i_A} \prod_{d \neq k} \langle A_d^{i_A} u_d^j, A_d^{j_A} u_d^i \rangle, \tag{25}$$

$$\mathbf{N}_i = \sum_{i_A=1}^{R_A} \sum_{i_G=1}^{R_G} (A_k^{i_A})^T g_k^{i_G} \prod_{d \neq k} \langle A_d^{i_A} u_d^i, g_d^{i_G} \rangle. \tag{26}$$

Eqs. (25) and (26) first appeared (in a different form than above) in Ref. [12]. Clearly, the above equation is nonlinear in terms of the unknowns u_d^l. In the ALS framework, Eq. (24) is reduced to a linear system and solved sequentially for each dimension in an iterative manner. Therefore, the number of unknowns in a single iteration is $n_k R_U$, which is independent of the dimensionality P once the user prescribes R_U. Moreover, a significant amount of computation is saved by noting that only a small portion of the terms involved need to be recalculated for different k and R_U. In implementation, we begin with $R_U = 1$ and random initial values for u_d^l, and then increase R_U gradually until stopping criteria (magnitude of objective function and/or number of iterations) are met.

At this point, it is of interest to examine the growth of memory requirement in the above described CPD approximation. Without loss of generality, assume that n nodes are used for discretization per dimension. Then the total number of entries needed to represent $\mathbb{A}$, $\mathbb{G}$ and $\mathcal{U}$ are Pn^2R_A, PnR_G and PnR_U respectively, which grow only linearly with the dimensionality P for fixed values of R_A, R_G and R_U. In other words, the memory required for the terms appearing in Eqs. (25) and (26) grows rather benignly with dimensionality. For further reduction in memory requirement, these terms can be computed "dynamically", i.e. when needed during the execution of the code (of course, at the cost of increased computation time). See section 4.2 of Ref. [12] for a detailed analysis of the complexity of this ALS framework.

In the next section, we discuss the incorporation of the normality constraint, which is essential to ensure that the ALS scheme returns a nontrivial answer (note that for the stationary FPE, $\mathcal{G} \equiv 0$).

4.4. Constraints

A probability density function that solves the stationary FPE must satisfy the following constraints:

$$\lim_{\mathbf{x} \to \infty} \mathcal{W}(\mathbf{x}) = 0, \tag{27}$$

$$\int_{\Omega} \mathcal{W}(\mathbf{x}) d\mathbf{x} = 1. \tag{28}$$

Eqs. (27) and (28) are called the vanishing boundary condition and normality condition respectively. Any spatial discretization based method requires a compact domain for implementation, due to which the vanishing boundary condition is imposed on a conservatively chosen "large-enough" domain. Here, the boundary condition is enforced by simply removing the boundary rows and columns of $A_d^{i_A}$'s and $u_d^{i_U}$'s in Eq. (24). This is also typical in traditional finite difference/element methods.

Normality constraint

Adding the normality constraint to Eq. (18) precludes the trivial solution. By virtue of the CP structure of $\mathcal{U}$, the multi-dimensional integral of Eq. (28) can be decoupled into a product of one-dimensional integrals and therefore scales linearly with system dimensionality. Corresponding to the Chebyshev extrema points used for spatial discretization, the Clenshaw–Curtis quadrature are used for integration [23], resulting in the following tensorized structure of the normality condition:

$$\mathcal{B} = \bigotimes_{d=1}^{P} b_d, \qquad \langle \mathcal{B}, \mathcal{U} \rangle = 1, \tag{29}$$

where b_d is a $n_d \times 1$ vector, containing the Clenshaw–Curtis weights for dimension d. To impose the normality condition, a penalty term is added to the minimization problem of Eq. (22). With $\mathcal{G} = 0$, we have

$$\min_{\{u_k^r\}} \| \mathbb{A}\mathcal{U} \|_F^2 + \frac{\alpha}{2} \left(\langle \mathcal{B}, \mathcal{U} \rangle - 1 \right)^2, \tag{30}$$

where α is a positive penalty parameter. We thus redefine the cost function as

$$\begin{aligned} \mathcal{R} &= \| \mathbb{A}\mathcal{U} \|_F^2 + \frac{\alpha}{2} \left(\langle \mathcal{B}, \mathcal{U} \rangle - 1 \right)^2 \\ &= \langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle + \frac{\alpha}{2} \left(\langle \mathcal{B}, \mathcal{U} \rangle^2 - 2\langle \mathcal{B}, \mathcal{U} \rangle + 1 \right). \end{aligned} \tag{31}$$

The corresponding first order conditions for minimization are

$$\frac{\partial \langle \mathbb{A}\mathcal{U}, \mathbb{A}\mathcal{U} \rangle}{\partial u_k^r} + \alpha \left(\langle \mathcal{B}, \mathcal{U} \rangle \frac{\partial \langle \mathcal{B}, \mathcal{U} \rangle}{\partial u_k^r} - \frac{\partial \langle \mathcal{B}, \mathcal{U} \rangle}{\partial u_k^r} \right) = 0, \tag{32}$$

where, the terms not already given in Section 4.3 are

$$\langle \mathcal{B}, \mathcal{U} \rangle = \sum_{i_U=1}^{R_U} \prod_{d=1}^{P} \langle b_d, u_d^{i_U} \rangle, \qquad \frac{\partial \langle \mathcal{B}, \mathcal{U} \rangle}{\partial u_k^r} = (b_d)^T \prod_{d \neq k} \langle b_d, u_d^r \rangle.$$

For a fixed k, upon collecting terms for all r's we have,

$$\left[\mathbf{M} + \alpha \begin{pmatrix} \mathbf{B}_{1,1} & \cdots & \mathbf{B}_{1,R_U} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{R_U,1} & \cdots & \mathbf{B}_{R_U,R_U} \end{pmatrix} \right] \begin{pmatrix} u_k^1 \\ \vdots \\ u_k^{R_U} \end{pmatrix} = \alpha \begin{pmatrix} \mathbf{C}_1 \\ \vdots \\ \mathbf{C}_{R_U} \end{pmatrix}, \tag{33}$$

where $\mathbf{M}$ is the same as in Eq. (24) and

$$\mathbf{B}_{i,j} = b_k (b_k)^T \left(\prod_{d \neq k} (b_d)^T u_d^j \right) \left(\prod_{d \neq k} (b_d)^T u_d^i \right), \tag{34}$$

$$\mathbf{C}_i = b_k \prod_{d \neq k} (b_d)^T u_d^i. \tag{35}$$

The ALS scheme discussed in Section 4.3 can now be used directly to solve Eq. (33). A summary of the entire approach is provided below:

- Use Chebyshev extrema points for spatial discretization of each dimension and approximate the Fokker–Planck operator by Chebyshev differentiation matrices.
- Rewrite the FP operator in tensor product structure and incorporate the normality constraint to obtain Eq. (33).
- Cycle for $R_U = 1, 2, \ldots, R_{max}$ until stopping criterion $\mathbf{S}_1$ (see below) is met:
  - Initialize u_d^r for $d = 1, 2, \ldots, P$ and $r = 1, 2, \ldots, R_U$.
  - Repeat for $d = 1, 2, \ldots, P$ until stopping criterion $\mathbf{S}_2$ (see below) is met:
    - Solve Eq. (33) to obtain u_d^r for $r = 1, 2, \ldots, R_U$ while holding u_d^r's for other dimensions constant.
- Return u_d^r for $d = 1, 2, \ldots, P$ and $r = 1, 2, \ldots, R_U$.

Stopping criteria

The stopping criteria appearing above are:

- $\mathbf{S}_1$: $\sqrt{\frac{\mathcal{R}}{\prod_{d=1}^{P} n_d}} < \epsilon_1$ or $R_U = R_{max}$, where R_{max} is the maximum allowable number of component rank-one tensors. In other words, unless the prescribed computational limit is reached, this corresponds to reaching a satisfactorily low value of the objective function, such that the optimization problem of Eq. (30) is considered solved.
- $\mathbf{S}_2$: $\delta \doteq |\mathcal{R}_{\text{curr}} - \mathcal{R}_{\text{prev}}| < \epsilon_2$ or maximum number of iterations reached. This corresponds to checking for satisfactory reduction of the objective function over a single iteration with a fixed number (R_U) of component rank-one tensors. If the reduction is not significant, i.e. $\delta < \epsilon_2$, a new enrichment step is initiated by setting $R_U \leftarrow R_U + 1$, unless criterion $\mathbf{S}_1$ is met.

5. Numerical examples

In this section, we present several numerical examples involving relatively high dimensional systems. To facilitate illustration, R_U is referred to as the "number of enrichment steps" while $u_d^{i_U}$ for $i_U = 1, 2, \ldots, R_U$ are called basis functions for dimension d. All shown basis functions are scaled to have unit norm via the use of a scaling vector. Due to the random initialization of basis functions, the comparisons and conclusions are made in an average sense. It is shown through a progression of examples that the number of degrees of freedom in the proposed method scales favorably with the system dimensionality and presents a strong case for curbing the curse of dimensionality associated with FPE.

5.1. Example 1: 2-state system

Consider the following two-state nonlinear oscillator [31]:

$$\ddot{x} + b\dot{x} + x + a(x^2 + \dot{x}^2)\dot{x} = g'\xi(t), \tag{36}$$

where, $\xi(t)$ is white noise with intensity Q and $a = 0.125, b = -0.5, g' = 1, Q = 0.4$ are constants. FPE of the above system admits the following stationary solution:

$$\mathcal{W}_s(x, y) = C \exp\left\{ -\frac{1}{g'^2 Q} \left[\frac{a}{2}(x^2 + y^2)^2 + b(x^2 + y^2) \right] \right\}, \tag{37}$$

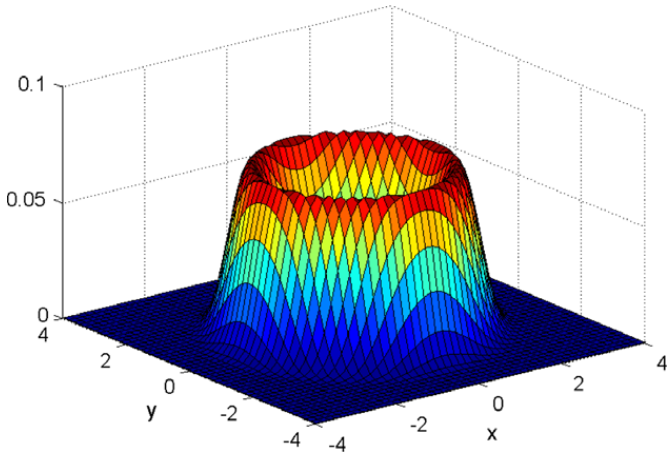
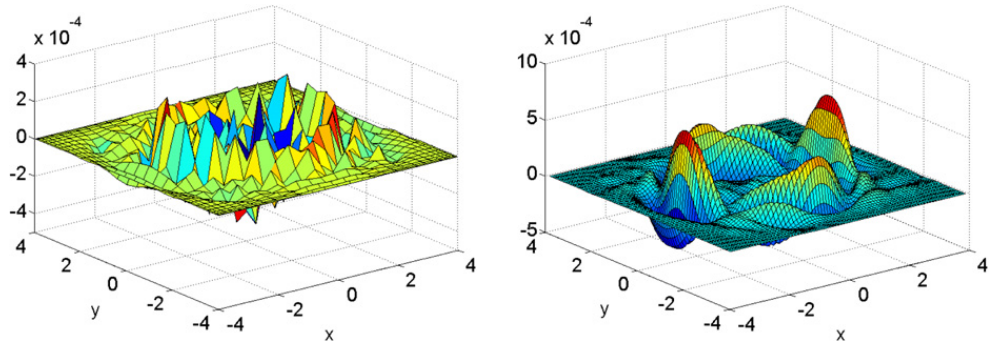


Fig. 2. Approximated stationary solution for the 2-state system.



(a) Error surface: Chebyshev spectral differentiation. (b) Error surface: Traditional finite difference method.

Fig. 3. Comparison of error surface for the 2-state system.

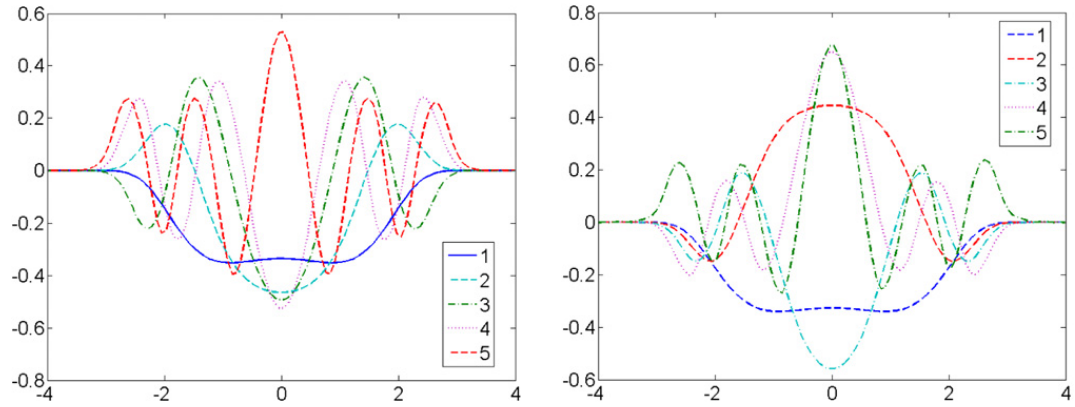
where, $y \doteq \dot{x}$ and C is the normalization constant. Note that the true solution is not a separable function. The current method was implemented using 29 Chebyshev extrema points per dimension on the finite domain $[-4, 4] \otimes [-4, 4]$. The approximation after five enrichment steps ($R_U = 5$) is shown in Fig. 2. The error between the approximated and in this case, known true solution in mean square sense (MSE) evaluated on the grid shown in Fig. 3(a) is 4.403×10^{-9}. The basis functions $\{u_d^{i_U}\}_{i_U=1}^{5}$ for $d = 1$ (dimension x) and $d = 2$ (dimension y) are shown in Fig. 4(a) and (b) respectively with scaling vector [0.6883, 0.4570, 0.1192, 0.0263, 0.0046]. In this and all subsequent examples, the basis functions have been ordered from most to least significant in terms of their contribution to the final approximation, as reflected by the relative magnitudes of the corresponding components of the scaling vector.

The total number of degrees of freedom of this approximation is (number of nodes per dimension) $\times$ (number of dimensions) $\times$ (number of enrichment steps) $= 29 \times 2 \times 5 = 290$. In comparison, if second order finite difference spatial discretization in tensor product form is used, 83 nodes per dimension provide an MSE of 1.242×10^{-8} after 8 enrichment steps (see error plot in Fig. 3(b)). The total DOFs is now $83 \times 2 \times 8 = 1328$, with a worse MSE. The advantage of Chebyshev spectral differentiation should be further evident for higher dimensional cases.

5.2. Example 2: 4-state system

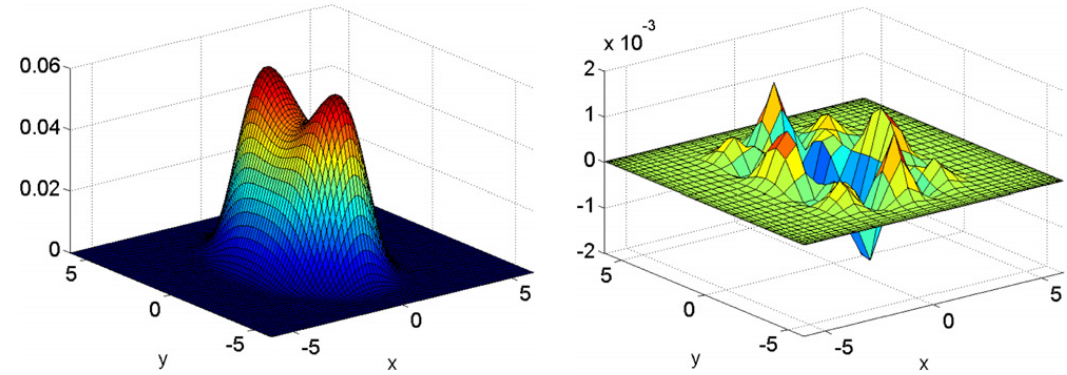
Next consider the 4-state nonlinear isolating suspension model given in [32],

$$\begin{aligned}
\dot{x_1} &= x_3, \\
\dot{x_2} &= x_4, \\
\dot{x_3} &= -ax_3 - \frac{1}{M}\frac{\partial V}{\partial x_1} + \xi_1, \\
\dot{x_4} &= -bx_4 - \frac{1}{I}\frac{\partial V}{\partial x_2} + \xi_2.
\end{aligned}$$



(a) Basis functions: x dimension. (b) Basis functions: y dimension.

Fig. 4. Basis functions for the 2-state system.



(a) Approximated $x_1 - x_2$ marginal pdf. (b) Error surface of the marginal pdf.

Fig. 5. Results for the 4-state system.

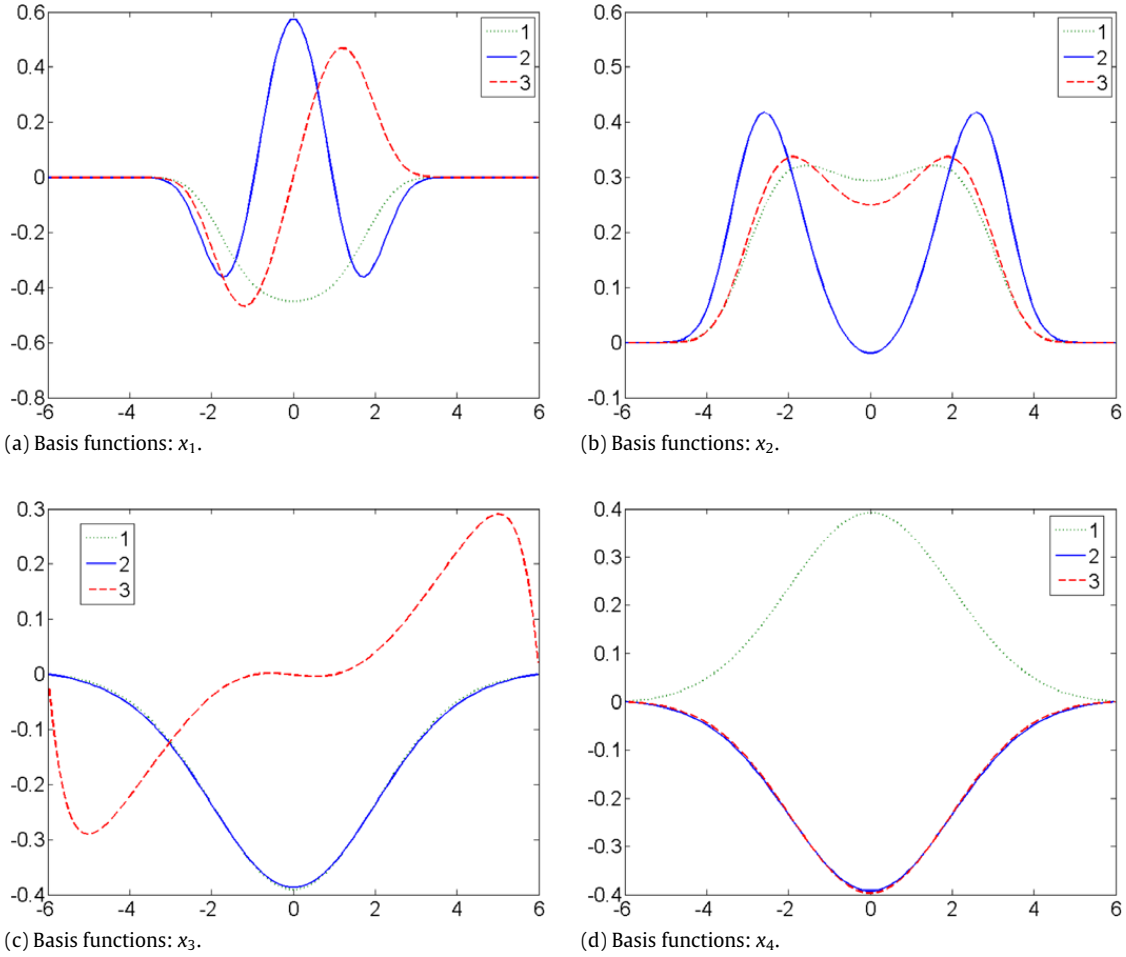
The above system belongs to a small class of nonlinear systems possessing a "Hamiltonian-like" [2] structure for which the true stationary distribution is known. In this example, the coupled potential function V is chosen as $V(x_1, x_2) = k_1 x_1^2 + k_2 x_2^2 + \epsilon(\lambda_1 x_1^4 + \lambda_2 x_2^4 + \mu x_1^2 x_2^2)$. System parameters are $a = 0.5$, $b = 1$, $k_1 = 0.5$, $k_2 = -0.5$, $\epsilon = 0.5$, $\lambda_1 = 0.25$, $\lambda_2 = 0.125$, $\mu = 0.375$. The noise process is two-dimensional with intensities $2D_1(\sim \xi_1) = 4$ and $2D_2(\sim \xi_2) = 8$. Parameters have been selected to ensure that $\frac{D_1 M}{a} = \frac{D_2 I}{b} = 2T$, because under this condition the analytical solution can be written as

$$\mathcal{W}_s(x_1, x_2, x_3, x_4) = C \exp\left[-\frac{1}{2T} V(x_1, x_2) - \frac{a}{2D_1} x_3^2 - \frac{b}{2D_2} x_4^2\right], \tag{38}$$

where C is the normalization constant as usual. Owing to the nature of the chosen potential function V, the true stationary solution is once again not separable.

The described method was implemented using 33 Chebyshev extrema points per dimension on the finite domain $\otimes^4[-6, 6]$. The $x_1 - x_2$ marginal pdf of the approximated solution is presented in Fig. 5(a), while the error plot is shown in Fig. 5(b). After three enrichment steps, the MSE between the approximated and true marginal is 5.118×10^{-8}. The time of execution of the entire algorithm (on a PC with 2.5 GHz CPU and 4 GB RAM) was less than 20 s. The basis functions for each dimension, i.e. $\{u_d^{i_U}\}_{i_U=1}^3$ are shown in Fig. 6 labeled in decreasing order of significance with scaling vector $[0.0930, 0.0154, 3.077 \times 10^{-4}]$, and the total DOF is $33 \times 4 \times 3 = 396$.

It is interesting to look at the nature of the various basis functions. In keeping with the true solution given in Eq. (38), all of them are even functions, except u_1^3 and u_3^3, which are odd. In the present example, such odd basis functions are unexpected, but while they do appear, they have a negligible effect on the solution because of their relatively insignificant magnitude (see scaling vector above). This is in fact a recurring trend in the present approach as seen below in higher dimensional examples and is a phenomenon worthy of further investigation. Once again, the second order finite difference method in its tensor-product form provides an MSE of 7.297×10^{-8} following six enrichment steps, requiring a total of 1608 DOFs (67 nodes per dimension).



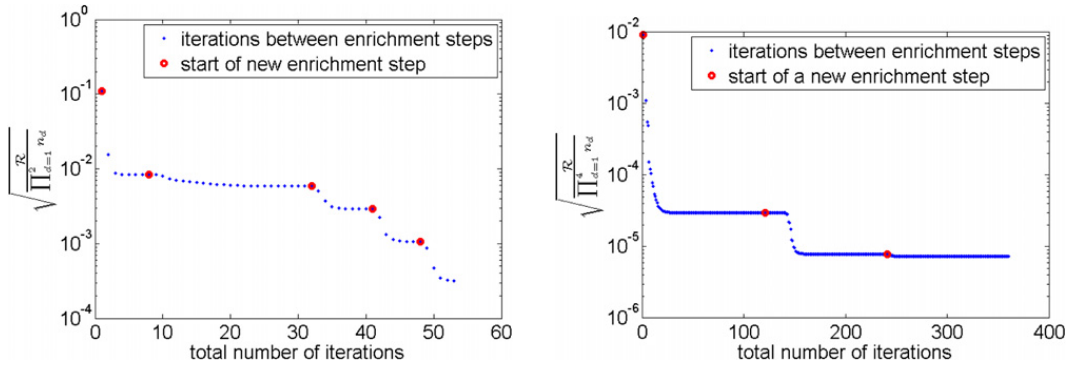
(a) Basis functions: x_1.

(b) Basis functions: x_2.

(c) Basis functions: x_3.

(d) Basis functions: x_4.

Fig. 6. Basis functions for the 4-state system.

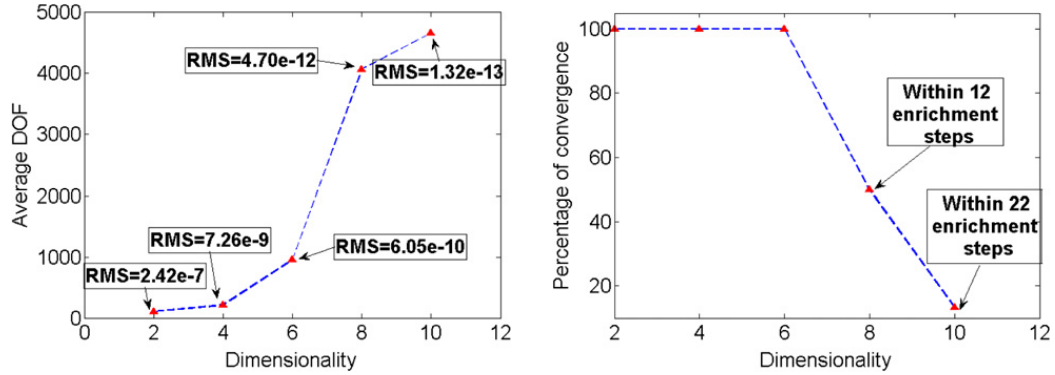


(a) 2-state system.

(b) 4-state system.

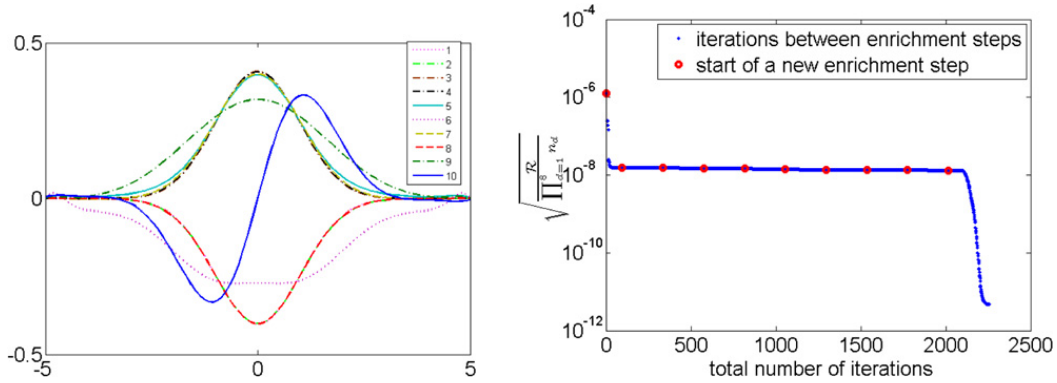
Fig. 7. Evolution of equation error: Examples 1 and 2.

Fig. 7(a) illustrates the convergence characteristics of ALS for examples 1 and 2. Clearly, the two-state system exhibits relatively "gradual" convergence whereas the four-state system goes through extended periods of sluggish performance in between abrupt improvements in the approximation. Similar slow convergence of ALS has been noted widely in the literature (e.g. see [27,33,34]) and attributed to "degeneracies" in the tensor. For example, the so-called "bottleneck" sluggish behavior is observed if two or more basis functions of a single dimension are collinear. If linear dependency is observed in all dimensions, the phenomenon is sometimes referred to as a "swamp" [27]. It is clear from Fig. 4 that the two-state system suffers from no such collinearity whereas the four-state system has linear dependency between the bases $\{u_3^{i_U}\}$ and $\{u_4^{i_U}\}$ (Fig. 6(c) and (d)), leading to a bottleneck scenario and the concomitant sluggish convergence (Fig. 7(b)). In addition, we note that there is another factor in play in the current application, namely the dimensionality of the underlying system which can potentially cause sluggish convergence. This is discussed in greater detail in the next example.



(a) Average DOF growth. (b) Success rates.

Fig. 8. Performance trends versus dimensionality.



(a) Basis functions for x_1, i.e. $\{u_1^{i_U}\}_{i_U=1}^{10}$. (b) ALS convergence characteristics.

Fig. 9. A bottleneck situation: 8-state system.

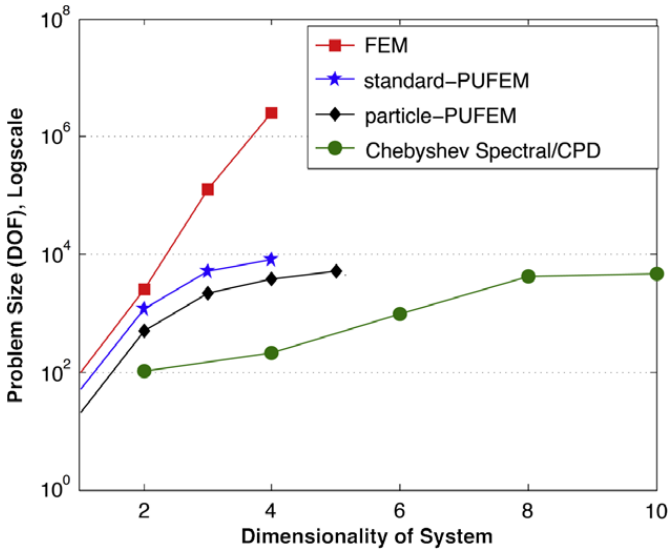


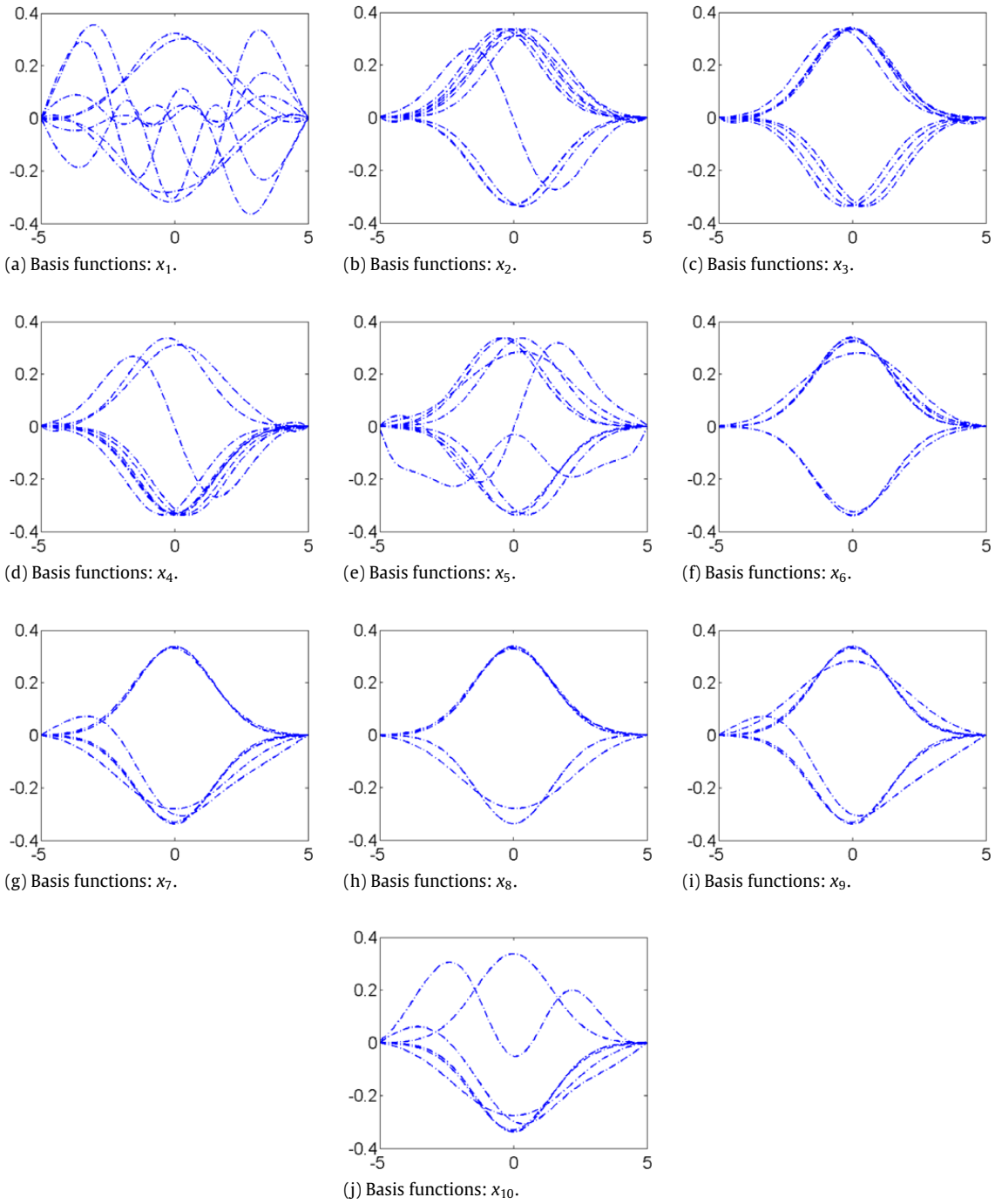
Fig. 10. A comparison of methods: FEM, s/p-PUFEM and the current approach.

5.3. Example 3: higher order systems

To demonstrate the scalability of the proposed method, we extend the previous example to solve the FPE for a generalized high dimensional nonlinear oscillator given by the following template:

$$\dot{x}_i = y_i,$$

$$\dot{y}_i = -a_i y_i - b_i \frac{\partial V}{\partial x_i} + \xi_i, \quad i = 1, 2, \ldots, P$$



(a) Basis functions: x_1. (b) Basis functions: x_2. (c) Basis functions: x_3.

(d) Basis functions: x_4. (e) Basis functions: x_5. (f) Basis functions: x_6.

(g) Basis functions: x_7. (h) Basis functions: x_8. (i) Basis functions: x_9.

(j) Basis functions: x_{10}.

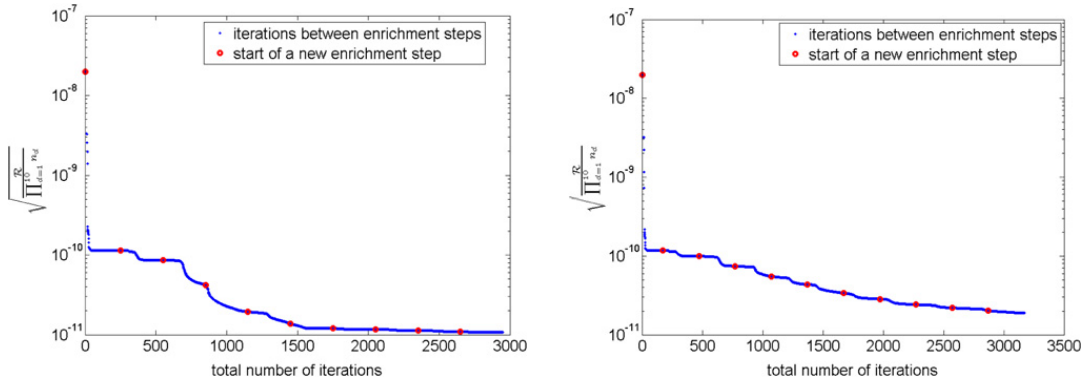
Fig. 11. Basis functions for the 10-state system in Case II.

where ξ_i's are uncorrelated white noise processes, i.e., $D_{ij} = 0$ for $i \neq j$. When $\frac{D_i}{a_i b_i} = 2T$, the stationary solution can be given by

$$w_s(\mathbf{x}, \mathbf{y}) = C \exp\left[-\frac{1}{2T} V(x_1, x_2, \ldots, x_P) - \sum_{i=1}^{P} \frac{a_i}{2D_i} y_i^2\right], \tag{39}$$

where intensity of noise ξ_i is $2D_i$. Three types of configurations are considered:

- **Case I:** For simplicity, set $V(x_1, \ldots, x_P) = \frac{1}{2}\sum_{i=1}^{P} x_i^2$ and $\frac{D_i}{a_i b_i} = 2T$ for $i = 1, 2, \ldots, P$. In this configuration, the normalization constant is $C = (8T\pi^2 D_i/a_i)^{-P/2}$ and Eq. (39) can be separated into a product of P identical one-dimensional functions. Results for various P are presented in Table 1 of Section 5.3.1.
- **Case II:** A non-separable linear system is considered by selecting $V(x_1, \ldots, x_P) = \frac{1}{2}\sum_{i=1}^{P} x_i^2 + \frac{1}{4}\sum_{i=1}^{P-1} x_1 x_{i+1}$. We let $2P = 10$, and in order to ensure the existence of a closed-form solution given by Eq. (39), set $\frac{D_i}{a_i b_i} = 2T$ for $i = 1, 2, \ldots, P$.



(a) Case II. (b) Case III.

Fig. 12. Evolution of equation error for the 10-state system in Case II and III.

Table 1
Results for higher order systems.

Dimensionality ($2P$)	Results of a typical run				Avg. number of enrichment steps in successful runs	Average DOF
	Number of enrichment steps (R_U)	DOF	$\sqrt{\frac{\mathcal{R}}{\prod_{d=1}^{2P} n_d}}$	Time of execution		
2	1	106	2.42×10^{-7}	0.5 s	1	106
4	1	212	7.26×10^{-9}	1.5 s	1	212
6	3	954	6.05×10^{-10}	60 s	3	954
8	10	4240	4.70×10^{-12}	1.8 h	9.6	4070
10	6	3180	1.32×10^{-13}	0.9 h	8.8	4664

The corresponding results are provided in Section 5.3.2 and this case demonstrates the effectiveness of the proposed method for non-separable systems.

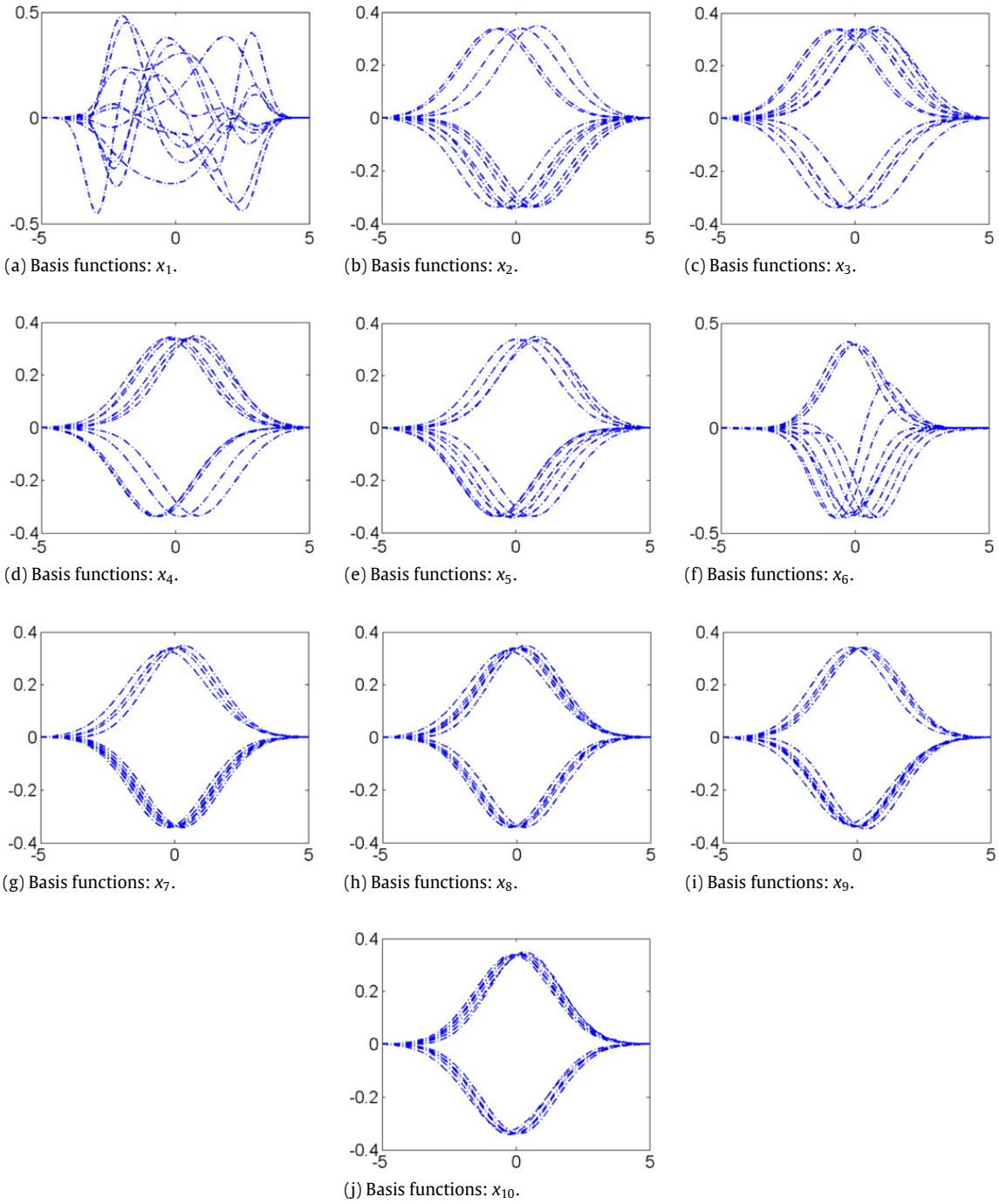
- **Case III:** To demonstrate the versatility of the current method, a non-separable nonlinear system is tested by setting $V(x_1, \ldots, x_P) = \frac{1}{2}(\frac{1}{16}x_1^4 + \sum_{i=2}^{P} x_i^2) + \frac{1}{4}\sum_{i=1}^{P-1} x_1 x_{i+1}$ with $2P = 10$. This system admits a stationary solution, however, the parameters D_i, a_i and b_i are chosen such that $\frac{D_i}{a_i b_i} = 2T$ no longer holds for all i. Consequently, a closed-form solution does not exist. Results are given in Section 5.3.3 to support the strength of the proposed method.

In all cases, 53 Chebyshev extrema points were used per dimension on the domain $\otimes^{2P}[-5, 5]$.

5.3.1. Case I with separable analytical stationary solution

For the case with separable stationary solution, let $a_i = b_i = D_i = 1$, thus $2T = 1$. For each P, Table 1 shows the results of a typical run of the described procedure, as well as its average performance over multiple runs, each starting from a different random initialization of the basis functions. The following observations can be made:

- *Computation Time:* As expected, the average computation time grows with dimensionality, although this growth may not manifest in an individual run of the algorithm (compare typical runs for $2P = 8$, 10).
- *Enrichment Steps/DOF Growth:* Due to the fact that Eq. (39) is perfectly separable, a single enrichment step should ideally be sufficient to obtain the solution. In reality, the number of enrichment steps (R_U) generally increases with P, although not exponentially (see average R_U). Consequently the DOF count ($=2PR_U n_d$) does not grow explosively with dimensionality. This is a key result, and is illustrated further in Fig. 8(a). It is clearly visible that although not linear (since $R_U = R_U(P)$), the average DOF count of the current method for the chosen problem shows a rather benign growth rate for obtaining highly accurate approximations.
- *Bottlenecks and Dimensionality:* As mentioned in the previous example, ALS is susceptible to bottleneck/swamp behavior (i.e. periods of slow convergence) resulting from linear dependency of basis functions. It is observed that as dimensionality increases, this problem becomes increasingly inescapable. Fig. 9(a) shows a bottleneck situation in the x_1-basis functions for the 8-state system (see for example the collinearity of u_1^2 through u_1^8). The resulting slow convergence characteristics of ALS is apparent in Fig. 9(b).
  We note here that in addition to collinearity type degenracies, high dimensionality also plays a role in the observed sluggish behavior of ALS in the current application. The objective function of Eq. (30) is a $2P$th degree polynomial, due to which ALS must overcome an increasingly large number of local minima as the dimensionality grows. Even without tensor degenracies, ALS can therefore be expected to meander through various local minima until it receives just the "right" random perturbation that guides it towards the global minimum. Each new enrichment step adds a randomly generated rank-one tensor to the existing solution tensor, which may help ALS meet the $\sqrt{\frac{\mathcal{R}}{\prod_{d=1}^{2P} n_d}} < \epsilon_1$ stopping



(a) Basis functions: x_1. (b) Basis functions: x_2. (c) Basis functions: x_3.

(d) Basis functions: x_4. (e) Basis functions: x_5. (f) Basis functions: x_6.

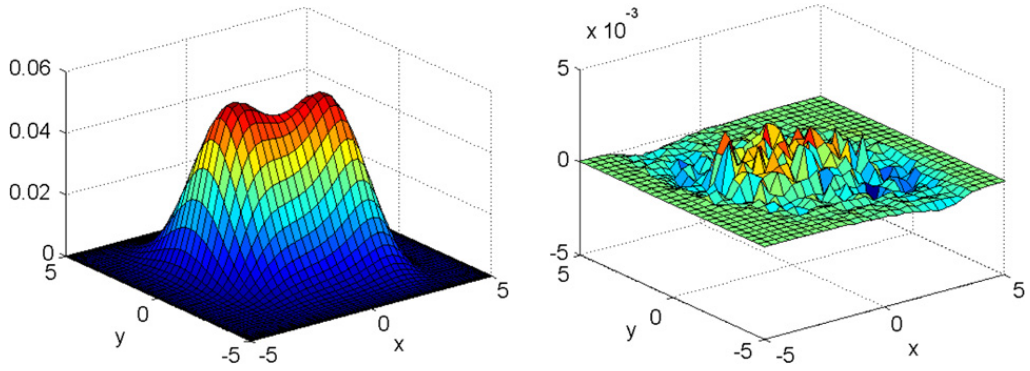
(g) Basis functions: x_7. (h) Basis functions: x_8. (i) Basis functions: x_9.

(j) Basis functions: x_{10}.

Fig. 13. Basis functions for the 10-state system in Case III.

criterion ($\mathbf{S}_1$). Such meandering can clearly become longer as dimensionality increases (compare Figs. 7(a), (b) and 9). This argument also explains the growth of R_U with P described in the previous bullet.

- *Success Rates:* Owing to bottleneck/dimensionality issues described above, not all runs of the described algorithm end successfully (i.e. some runs terminate after hitting the upper limit on the number of iterations without achieving the desirable accuracy. If more iterations were executed, ALS would probably have obtained better results.). The success rate declines with increasing dimensionality (see Fig. 8(b)) and puts in perspective the benign growth rate of DOF versus P. While this is a problem, it can be handled by numerous initializations of ALS in parallel, thus improving the chance of quick termination of the procedure.
- *Curse of Dimensionality and Comparative Performance:* At this point, it is interesting to compare the DOF growth of the current method with some existing numerical techniques for solving FPE. Fig. 10 depicts the curse of dimensionality faced by direct discretization based methods such as finite element [35,36], which become quickly infeasible as dimensionality increases. Partition of unity based meshless methods (such as standard-PUFEM and particle-PUFEM) were able to achieve much more favorable scaling and polynomial-like DOF growth [5,1]. In these methods, even though the number of nodes



(a) $x_1 - x_2$ marginal pdf by the proposed tensor method. (b) Error surface of the marginal pdf with respect to MCS.

Fig. 14. Results for the marginal pdf of Case III.

needed does not grow exponentially, the number of basis functions needed per node suffers from factorial growth, thus resulting in excessive computational burden. In comparison, the present approach scales very well and is easily able to solve high dimensional systems in a reasonable amount of time. Data for FEM, sPUFEM and pPUFEM are not available for dimensions greater than 4, 4 and 5 respectively because of infeasiblity on the computational platforms they were implemented. It must be noted that Fig. 10 must be interpreted with caution because of variability in the type of problems solved in various dimensions and different methods. This figure is only meant to provide a gross order-of-magnitude comparison. Moreover, the comparison is not exhaustive, i.e. does not consider the potential performance of other advanced tensor approaches, e.g. QTT approximation of [17,18].

5.3.2. Case II with non-separable analytical stationary solution

In this case, let $a_i = b_i = 1$ and $D_i = 2$, thus $2T = 2$. Through 10 enrichment steps, the root mean square (RMS) error of the approximated solution with respect to the analytical solution is 4.24×10^{-11}. The basis functions along all the 10 dimensions are given by Fig. 11 and the convergence characteristic for this example is provided in Fig. 12(a). It is demonstrated via this non-separable 10D case that the proposed method works effectively for solving high dimensional stationary Fokker–Planck equations.

5.3.3. Case III with no closed-form stationary solution

We set $D_1 = 1$, $D_i = 2$ for $i = 2, \ldots, P$, and $a_i = b_i = 1$ for $i = 1, \ldots, P$. Following 11 enrichment steps, the convergence characteristics and spatial basis functions for each dimension are provided in Figs. 12(b) and 13 respectively. For visualization, the marginal pdf of the first two states is provided in Fig. 14(a). Since a closed-form stationary solution is not available for this example, Monte Carlo simulation (Milstein scheme [37]) was employed to estimate the accuracy of the obtained approximation. Fig. 14(b) shows the error surface constructed by comparing the marginal pdf of Fig. 14(a) with the histogram obtained from Milstein integration of an ensemble of 10^6 initial conditions (a final time of 5 s is sufficient for the system to reach near stationary behavior). The mean squared error (MSE) corresponding to Fig. 14(b) is 2.13×10^{-7}, and reduces to 1.48×10^{-7} if 10^7 samples are used.

6. Conclusions and further research

In this paper, the stationary FPE was solved by incorporating Chebyshev spectral differentiation into the framework of solving linear PDEs by CP decomposition. The FP operator was expressed in a tensor product structure and the normality constraint was enforced via a penalty method. Using the alternating least squares algorithm to solve the resulting discretized system, numerical results for non-separable systems up to 10 dimensional state-space were provided. It was shown that the DOF count of the proposed approach scales favorably, making it an attractive approach for nontrivial uncertainty quantification problems. Also the computational time involved is reasonable, potentially allowing the present approach for use in nonlinear state estimation. Reduction of the incidence of bottlenecks is a topic of current investigation which will further increase the applicability of the proposed approach to real life high dimensional problems. Further research is also needed to extend the proposed approach to the transient Fokker–Planck equation (see [38] for some initial results).

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. ECCS-1254244 and the NASA Marshall Space Flight Center Grant No. NNM13AA07G. The authors would like to thank Dr. Amine Ammar for his insights into the method of proper generalized decomposition.

References

[1] M. Kumar, S. Chakravorty, J.L. Junkins, A semianalytic meshless approach to the transient Fokker–Planck equation, Probab. Eng. Mech. 25 (2010) 323–331.
[2] A. Fuller, Analysis of nonlinear stochastic systems by means of the Fokker–Planck equation, Internat. J. Control 9 (1969) 603–655.
[3] R.S. Langley, A finite element method for the statistics of random nonlinear vibration, J. Sound Vib. 101 (1985) 41–54.
[4] R.E. Bellman, Dynamic Programming, Princeton University Press, Princeton, New Jersey, 1957.
[5] M. Kumar, S. Chakravorty, P. Singla, J.L. Junkins, The partition of unity finite element approach with hp-refinement for the stationary Fokker–Planck equation, J. Sound Vib. 327 (2009) 144–162.
[6] M. Kumar, S. Chakravorty, J.L. Junkins, On the curse of dimensionality in the Fokker–Planck equation, in: AAS Astrodynamics Specialist Conference, Pittsburgh, PA, USA, Aug 9–13, 2009.
[7] Y. Sun, M. Kumar, A meshless p-pufem Fokker–Planck equation solver with automatic boundary condition enforcement, in: American Control Conference, ACC, 2012, IEEE, 2012, pp. 74–79.
[8] B.N. Khoromskij, Tensors-structured numerical methods in scientific computing: survey on recent advances, Chemometr. Intell. Lab. Syst. 110 (2012) 1–19.
[9] T. Kolda, B. Bader, Tensor decompositions and applications, SIAM Rev. 51 (2009) 455–500.
[10] L. Grasedyck, D. Kressner, C. Tobler, A literature survey of low-rank tensor approximation techniques, 2013, arXiv preprint arXiv:1302.7121.
[11] W. Hackbusch, B.N. Khoromskij, E.E. Tyrtyshnikov, Hierarchical kronecker tensor-product approximations, J. Numer. Math. 13 (2005) 119–156.
[12] G. Beylkin, M. Mohlenkamp, Algorithms for numerical analysis in high dimensions, SIAM J. Sci. Comput. 26 (2005) 2133–2159.
[13] J. Ballani, L. Grasedyck, A projection method to solve linear systems in tensor format, Numer. Linear Algebra Appl. 20 (2013) 27–43.
[14] D. Kressner, C. Tobler, Krylov subspace methods for linear systems with tensor product structure, SIAM J. Matrix Anal. Appl. 31 (2010) 1688–1714.
[15] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids, J. Non-Newton. Fluid Mech. 139 (2006) 153–176.
[16] A. Doostan, G. Iaccarino, A least-squares approximation of partial differential equations with high-dimensional random inputs, J. Comput. Phys. 228 (2009) 4332–4345.
[17] S. Dolgov, B. Khoromskij, Simultaneous state-time approximation of the chemical master equation using tensor product formats, 2013, arXiv preprint arXiv:1311.3143, submitted for publication.
[18] B.N. Khoromskij, O (dlog n)-quantics approximation of nd tensors in high-dimensional numerical modeling, Constr. Approx. 34 (2011) 257–289.
[19] B.N. Khoromskij, V. Khoromskaia, Multigrid accelerated tensor approximation of function related multidimensional arrays, SIAM J. Sci. Comput. 31 (2009) 3002–3026.
[20] A. Ammar, F. Chinesta, E. Cueto, M. Doblaré, Proper generalized decomposition of time-multiscale models, Internat. J. Numer. Methods Engrg. 90 (2012) 569–596.
[21] L. Giraldi, A. Nouy, G. Legrain, P. Cartraud, Tensor-based methods for numerical homogenization from high-resolution images, Comput. Methods Appl. Mech. Engrg. (2012).
[22] Y. Sun, M. Kumar, A tensor decomposition approach to high dimensional stationary Fokker–Planck equations, in: American Control Conference, ACC, 2014, Portland, OR, June 4–6, 2014.
[23] L. Trefethen, Spectral Methods in MATLAB, in: EngineeringPro Collection, SIAM, 2000.
[24] J.P. Boyd, Chebyshev and Fourier Spectral Methods, Courier Dover Publications, 2001.
[25] J.D. Carroll, J.-J. Chang, Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition, Psychometrika 35 (1970) 283–319.
[26] R.A. Harshman, Foundations of the parafac procedure: models and conditions for an "explanatory" multimodal factor analysis, UCLA Working Papers in Phonetics, 1970.
[27] P. Comon, X. Luciani, A.L.F. de Almeida, Tensor decompositions, alternating least squares and other tales, J. Chemometrics 23 (2009) 393–405.
[28] J. Håstad, Tensor rank is np-complete, J. Algorithms 11 (1990) 644–654.
[29] R. LeVeque, Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems, SIAM (2007).
[30] K.E. Atkinson, An Introduction to Numerical Analysis, John Wiley & Sons, 2008.
[31] G.R.G. Muscolino, M. Vasta, Stationary and non-stationary probability density function for nonlinear oscillators, Int. J. Nonlinear Mech. 32 (1997) 1051–1064.
[32] S.T. Ariaratnam, Random vibrations of non-linear suspensions, J. Mech. Eng. Sci. 2 (1960) 195–201.
[33] M. Rajih, P. Comon, R.A. Harshman, Enhanced line search: a novel method to accelerate parafac, SIAM J. Matrix Anal. Appl. 30 (2008) 1128–1147.
[34] P. Paatero, Construction and analysis of degenerate parafac models, J. Chemometrics 14 (2000) 285–299.
[35] S.F. Wojtkiewicz, L. Bergman, B.F. Spencer Jr., Numerical solution of some three state random vibration problems, in: L.A. Bergman, B.F. Spencer Jr. (Eds.), Vibration of Nonlinear, Random and Time-Varying Systems, Boston, MA, USA, 1995.
[36] S.F. Wojtkiewicz, L.A. Bergman, Numerical solution of high dimensional fokker planck equations, in: 8th ASCE Specialty Conference on Probablistic Mechanics and Structural Reliability, Notre Dame, IN, USA, 2000.
[37] G.N. Milstein, Numerical Integration of Stochastic Differential Equations, Springer, 1995.
[38] Y. Sun, M. Kumar, A tensor decomposition approach to high dimensional transient Fokker–Planck equations, in: 53rd IEEE Conference on Decision and Control, CDC, IEEE, 2014, in press.