# Numerical methods for high-dimensional kinetic equations

Heyrim Cho, Daniele Venturi, and George Em Karniadakis

**Abstract** High-dimensionality is one of the major challenges in kinetic modeling and simulation of realistic physical systems. The most appropriate numerical scheme needs to balance accuracy and computational complexity, and it also needs to address issues such as multiple scales, lack of regularity, and long-term integration. In this chapter, we review state-of-the-art numerical techniques for high-dimensional kinetic equations, including low-rank tensor approximation, sparse grid collocation, and ANOVA decomposition.

## 1 Introduction

Kinetic equations are partial differential equations involving probability density functions (PDFs). They arise naturally in many different areas of mathematical physics. For example, they play an important role in modeling rarefied gas dynamics [12, 13], semiconductors [68], stochastic dynamical systems [18, 63, 74–76, 103, 114], structural dynamics [9, 61, 100], stochastic partial differential equations (PDEs) [19, 57, 66, 112, 113], turbulence [35, 71, 72, 90], system biology [30, 85, 123], etc. Perhaps, the most well-known kinetic equation is the Fokker-Planck equation [74, 96, 107], which describes the evolution of the probability density function of Langevin-type dynamical systems subject to Gaussian white noise. Another well-known example of kinetic equation is the Boltzmann equation [115] describing a thermodynamic system involving a large number of interacting particles [13]. Other examples that may not be widely known are the Dostupov-Pugachev

Heyrim Cho
University of Maryland, College Park (MD), e-mail: hcho1237@math.umd.edu

Daniele Venturi
University of California, Santa Cruz (CA), e-mail: venturi@ucsc.edu

George Em Karniadakis
Brown University, Providene (RI), e-mail: george_karniadakis@brown.edu

Fokker-Planck [74, 96]    $\dfrac{\partial p}{\partial t} + \sum_{i=1}^{n} \dfrac{\partial}{\partial z_i} (G_i p) = \dfrac{1}{2} \sum_{i,j=1}^{n} \dfrac{\partial^2}{\partial z_i \partial z_j} (b_{ij} p)$

Boltzmann [12, 67]    $\dfrac{\partial p}{\partial t} + \sum_{j=1}^{3} v_j \dfrac{\partial p}{\partial z_j} = H(p, p)$

Liouville [26, 57, 61, 103]    $\dfrac{\partial p}{\partial t} + \sum_{j=1}^{n} \dfrac{\partial}{\partial z_j} (G_j p) = 0$

Malakhov-Saichev [66, 112]    $\dfrac{\partial p}{\partial t} + \dfrac{\partial}{\partial z} \left( \sum_{j=1}^{3} G_j \int_{-\infty}^{z} \dfrac{\partial p}{\partial x_j} dz' \right) = -\dfrac{\partial (Hp)}{\partial z}$

Mori-Zwanzig [113, 127]    $\dfrac{\partial p_1}{\partial t} = PLp_1 + PLe^{tQL} p_2(0) + PL \int_0^t e^{(t-s)QL} QLp_1 ds$

**Table 1** Examples of kinetic equations in different areas of mathematical physics.


equations [26, 61, 103, 114], the reduced-order Nakajima-Zwanzig-Mori equations [24, 113, 127], and the Malakhov-Saichev PDF equations [66, 112] (see Table 1). Computing the numerical solution to a kinetic equation is a challenging task that needs to address issues such as:

1. *High-dimensionality:* Kinetic equations describing realistic physical systems usually involve many phase variables. For example, the Fokker-Planck equation of classical statistical mechanics is an evolution equation for a joint probability density function in $n$ phase variables, where $n$ is the dimension of the underlying stochastic dynamical system, plus time.
2. *Multiple scales:* Kinetic equations can involve multiple scales in space and time, which could be hardly accessible by conventional numerical methods. For example, the Liouville equation is a hyperbolic conservation law whose solution is purely advected (with no diffusion) by the underlying system's flow map. This can easily yield mixing, fractal attractors, and all sorts of complex dynamics.
3. *Lack of regularity:* The solution to a kinetic equation is, in general, a distribution [50]. For example, it could be a multivariate Dirac delta function, a function with shock-type discontinuities [19], or even a fractal object (see Figure 1 in [113]). From a numerical viewpoint, resolving such distributions is not trivial although in some cases it can be done by taking integral transformations or projections [122].
4. *Conservation properties:* There are several properties of the solution to a kinetic equation that must be conserved in time. The most obvious one is mass, i.e., the solution to a kinetic equation always integrates to one. Another property that must be preserved is the positivity of the joint PDF, and the fact that a partial marginalization still yields a PDF.
5. *Long-term integration:* The flow map defined by nonlinear dynamical systems can yield large deformations, stretching and folding of the phase space. As a consequence, numerical schemes for kinetic equations associated with such systems

will generally loose accuracy in time. This is known as long-term integration problem and it can be eventually mitigated by using adaptive methods.

Over the years, many different techniques have been proposed to address these issues, with the most efficient ones being problem-dependent. For example, a widely used method in statistical fluid mechanics is the particle/mesh method [77, 89–91], which is based directly on stochastic Lagrangian models. Other methods are based on stochastic fields [109] or direct quadrature of moments [33]. In the case of Boltzmann equation, there is a very rich literature. Both probabilistic approaches such as direct simulation Monte Carlo [8, 97], as well as deterministic methods, e.g., discontinuous Galerkin and spectral methods [15, 16, 31], have been proposed to perform simulations. However, classical techniques such as finite-volumes, finite-differences or spectral methods, are often prohibitive in terms of memory requirements and computational cost. Probabilistic methods such as direct Monte Carlo are extensively used instead because of their very low computational cost compared to the classical techniques. However, Monte Carlo usually yields poorly accurate and fluctuating solutions, which need to be post-processed appropriately, for example through variance reduction techniques. We refer to Di Marco and Pareschi [67] for a recent excellent review.

In this chapter, we review the state-of-the-art in numerical techniques to address the high-dimensionality challenge in both the *phase space* and the *space of parameters* of kinetic systems. In particular, we discuss the sparse grid method [84, 102], low-rank tensor approximation [5, 17, 29, 40, 59, 79, 80], and analysis of variance (ANOVA) decomposition [11, 36, 60, 125] including Bogoliubov-Born-Green-Kirkwood-Yvon (BBGKY) [73] closures. These methods have been established as new tools to address high-dimensional problems in scientific computing during the last years, and here we discuss those in the context of kinetic equations, particularly in the deterministic Eulerian approach. As we will see, most of these methods allow us to reduce the problem of computing high-dimensional PDF solutions to sequences of problems involving low-dimensional PDFs. The range of applicability of the numerical methods is sketched in Figure 1 as a function of the number of phase variables $n$ and the number of parameters $m$ appearing in the kinetic equation.

## 2 Numerical Methods

This chapter discuss three classes of algorithms to compute the numerical solution of high-dimensional kinetic equations. The first class is based on sparse grids, and we discuss its construction in both the phase space and the space of parameters. The second class is based on low-rank tensor approximation and alternating direction methods, such as alternating least squares (ALS). The third class is based on ANOVA decomposition and BBGKY closures.
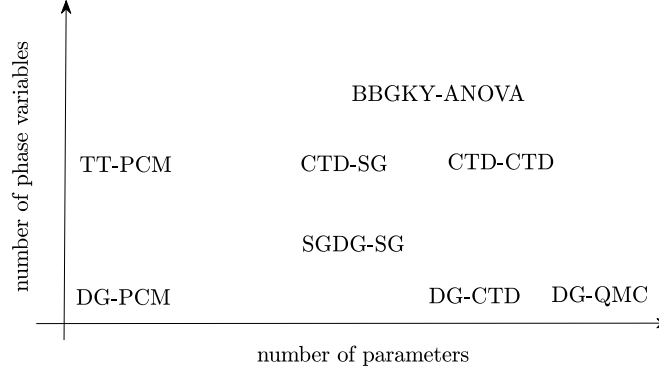
**Fig. 1** Range of applicability of different numerical methods for solving kinetic equations as a function of the number of phase variables and the number of parameters appearing in the equation. The first name refers to the numerical method we employ to discretize the phase variables, the second name we employ to discretize the space of parameters. For example, DG-PCM refers to an algorithm in which we discretize the phase variables with discontinuous Galerkin methods (DG), and random parameters with tensor product probabilistic collocation (PCM). Other methods listed are: canonical tensor decomposition (CTD), tensor train (TT), high-dimensional model representation (ANOVA), sparse grids (SG), and quasi Monte Carlo (QMC).

## 2.1 Sparse Grids

The sparse grid technique [10, 37] has been developed as a major tool to break the curse of dimensionality of grid-based approaches. The key idea relies on a tensor product hierarchical basis representation, which can reduce the degrees of freedom without losing much accuracy. Early work on sparse grid techniques can be traced back to Smolyak [102], in the context of high-dimensional numerical integration. The scheme is based on a proper balancing between the computational cost and the corresponding accuracy by seeking a proper truncation of the tensor product hierarchical bases, which can be formally derived by solving an optimization problem of cost/benefit ratios [41]. Sparse grid techniques have been incorporated in various numerical methods for high-dimensional PDEs, e.g., in finite element methods [10,99], finite difference methods [42], spectral methods [38, 101], and collocation methods for stochastic differential equations [64, 78, 117]. More recently, sparse grids have been proposed within the discontinuous Galerkin (DG) framework to simulate elliptic and hyperbolic systems using wavelet bases [43, 116].

The sparse grid formulation is based on a hierarchical set of basis functions in one-dimension. For instance, we can consider basis functions in a space $V_k$ of piecewise polynomials of degree at most $q$ on the $k$-th level grid that consists of $2^k$ uniform intervals, i.e.,

$$V_k \doteq \{v \,|\, v \in P^q(I_k^j), I_k^j = [2^{-k}j, 2^{-k}(j+1)], j = 0, \cdots, 2^k - 1\},$$

on $\Omega = [0, 1]$. Clearly, we have

$$V_0 \subset V_1 \subset V_2 \subset V_3 \subset \cdots.$$

These basis functions are suitable for the discontinuous Galerkin framework. Then, we define $W_k$ as the orthogonal complement of $V_{k-1}$ on $V_k$ with respect to the $L_2$ inner product on $\Omega$, that is,

$$V_{k-1} \oplus W_k = V_k, \quad V_{k-1} \perp W_k,$$

with $W_0 = V_0$. This yields the hierarchical representation of

$$V_k = \oplus_{0 \le j \le k} W_j.$$

Next, define the multidimensional increment space as defined as $\mathbf{W}_l = W_{l_1, z_1} \otimes W_{l_2, z_2} \otimes \cdots \otimes W_{l_N, z_N}$ with $l = (l_1, \cdots, l_N)$ as the multivariate mesh level. Accordingly, the standard tensor product space $\mathbf{V}_\ell$ can be represented as

$$\mathbf{V}_\ell = \bigcup_{|l|_\infty \le \ell} \mathbf{W}_l, \tag{1}$$

and the sparse grid approximation space as

$$\widetilde{\mathbf{V}}_\ell = \bigcup_{|l| \le \ell} \mathbf{W}_l, \tag{2}$$

where $|l|_\infty = \max_i l_i$ and $|l| = \sum_{i=1}^N l_i$. Then, $\widetilde{\mathbf{V}}_\ell \subset \mathbf{V}_\ell$. The number of degrees of freedom of $\widetilde{\mathbf{V}}_\ell$ is significantly smaller than the one of $\mathbf{V}_\ell$. This set of basis functions is also called multi-wavelet basis and it has been employed with the discontinuous Galerkin method to study the Vlasov and the Boltzmann equations [43, 101]. In particular, for sufficiently smooth solutions, it was shown in [101, 116] that a semi-discrete $L^2$ stability condition and an error estimate of the order $O\left((\log h)^N h^{q+1/2}\right)$ can be obtained. We emphasize that although the computational cost of the sparse grid formulation is significantly smaller than the full tensor product, the curse of dimensionality still remains as the sparse grid level $\ell$ increases. For this reason, [43, 101, 116] can handle problems with less than ten dimensions in the phase space.

The application of the sparse grid technique in the space of parameters differs from the one we just described only in regard of the choice of the basis functions. In fact, in this case, we are usually interested in computing multi-dimensional integrals in the form

$$p(z) = \int_{\mathbb{R}^m} p(z, \mathbf{b}) d\mathbf{b} \simeq \sum_{k=1}^q w^k p(z, \mathbf{b}^k), \tag{3}$$

where $\mathbf{b} = (b_1, ..., b_m)$. The collocation points $\mathbf{b}^k = (b_1^k, ..., b_m^k)$ and quadrature weights $w^k$ are obtained by suitable cubature rules with high polynomial exactness, e.g., Clenshaw-Curtis or Gauss abscissae [118]. More recent sparse collocation tech-

niques can increase the number of dimensions that can be handled in the space of parameters up to hundreds [119, 121].

## *2.2 Low-rank Tensor Approximation*

Low-rank tensor approximation has been established as a new tool to overcome the curse of dimensionality in representing high-dimensional functions and the solution to high-dimensional PDEs. The method has been recently applied to stochastic PDEs [25,29,56,69,79], approximation of high-dimensional Green's functions [44], the Boltzmann equation [48,55], and Fokker-Planck equation [2,22,49,53]. The key idea of low-rank tensor approximation [17,40,81] is to represent a multivariate function in terms of series involving products of low-dimensional functions. This allows us to reduce the problem of computing the solution from high-dimensional PDEs to a sequence of low-dimensional problems that can be solved recursively and in parallel, e.g., by alternating direction algorithms such as alternating least squares [20,25] and its parallel extension [52]. These algorithms are usually based on low-rank matrix techniques [39], and they have a convergence rate that depends on the type of kinetic equation and on its solution.

The most simplest tensor format is a rank one tensor of an $N$-dimensional function, $p(z_1, \cdots, z_N) = p_1(z_1) p_2(z_2) \cdots p_N(z_N)$, where $p_j(z_j)$ are one-dimensional functions. Upon discretization we can write $p$ in a tensor notation as

$$\mathbf{p} = \mathbf{p}_1 \otimes \cdots \otimes \mathbf{p}_N, \tag{4}$$

where $\mathbf{p}_j$ is a vector of length $q_z$ corresponding to the discretization of $p_j(z_j)$ with $q_z$ degrees of freedom[1]. More generally, we have a summation of rank-one tensors

$$p(z_1, \cdots, z_N) = \sum_{r=1}^{R} \alpha_r p_1^r(z_1) p_2^r(z_2) \cdots p_N^r(z_N), \tag{5}$$

and

$$\mathbf{p} = \sum_{r=1}^{R} \alpha_r \mathbf{p}_1^r \otimes \mathbf{p}_2^r \otimes \cdots \otimes \mathbf{p}_N^r, \tag{6}$$

where $R$ is the *tensor rank* or *separation rank*. This representation is also known as separated series expansion or canonical tensor decomposition. The main advantage of using a representation in the form (5)–(6) to solve a high-dimensional kinetic PDE relies on the fact that the algorithms to compute $\mathbf{p}_j^r$ and the normalization fac-

---

[1] For instance, if we represent $p_j(z_j)$ in terms of an interpolant

$$p_j(z_j) = \sum_{k=1}^{q_z} \mathrm{p}_{j,k} \phi_{j,k}(z_j),$$

then $\mathbf{p}_j = (\mathrm{p}_{j,1}, \cdots, \mathrm{p}_{j,q_z})$.

tors $\alpha_r$ involve operations with one-dimensional functions. In principle, the computational cost of such algorithms scales linearly with respect to the dimension $N$ of the phase space, thus potentially avoiding the curse of dimensionality. The representation can be generalized to any combination of low-dimensional separated functions. Canonical tensor decompositions have been employed to compute the solution to the Malakhov-Saichev kinetic equation [20], the Vlasov-Poisson equation [27], and functional differential equations [110].

More advanced tensor decomposition techniques involve Tucker decomposition, tensor train decomposition (TT), and hierarchical Tucker decomposition (HT). In particular, the tensor train decomposition is in the form of

$$p(z_1, \cdots, z_N) = Q_1(z_1)Q_2(z_2) \cdots Q_N(z_N), \quad Q_j(z_j) \in \mathbb{R}^{R_{j-1} \times R_j}, \qquad (7)$$

where the *tensor rank* becomes a tuple of $(R_1, \cdots, R_{N-1})$ with $R_0 = R_N = 1$. In each direction $j$, the index that runs over $\mathbb{R}^{R_{j-1}}$ and $\mathbb{R}^{R_j}$ takes care of the coupling to the $j-1$-th and the $j+1$-th dimension, respectively. A discretization of (7) with $q_z$ degrees of freedom in each dimensions yields

$$\mathbf{p} = \sum_{r_0=1}^{R_0} \cdots \sum_{r_N=1}^{R_N} \mathbf{Q}_1^{r_0,r_1} \otimes \mathbf{Q}_2^{r_1,r_2} \otimes \cdots \otimes \mathbf{Q}_N^{r_{N-1},r_N}, \qquad (8)$$

where $\mathbf{Q}_j^{r_{j-1},r_j}$ is a vector of length $q_z$. With a payoff of an additional tensor rank dimension, the problem of constructing a tensor train decomposition is closed and it can be solved to any given error tolerance or fixed rank [86]. The algorithm is based on a sequence of SVD applied to the matricizations of the tensor, i.e. the so-called high-order singular value decomposition (HOSVD) [39]. Methods for reducing the computational cost of tensor train are discussed in [82, 87, 126]. Applications to the Vlasov kinetic equation can be found in [23, 46, 58].

### 2.2.1 Temporal Dynamics

To include temporal dynamics in the low rank tensor representation of a field we can simply add additional time-dependent functions, i.e., represent $p(t, z_1, ..., z_N)$ as

$$p(t, z_1, \cdots, z_N) = \sum_{r=1}^{R} \alpha_r p_t^r(t) p_1^r(z_1) p_2^r(z_2) \cdots p_N^r(z_N). \qquad (9)$$

This approach has been considered by several authors, e.g., [2, 17], and it was shown to be effective for problems dominated by diffusion. However, for complex transient problems (e.g., hyperbolic dynamics), such approach is not practical as it requires a high resolution in the time domain. To address this issue, a discontinuous Galerkin method in time was proposed by Nouy in [79]. The key idea is to split the integration period into small intervals (finite elements in time) and then consider a space-time separated representation of the solution within each interval.

Alternatively, one can consider an explicit or implicit time-integration schemes [20,59]. In this case, the separated representation of the solution is computed at each time step. In such representations we look for expansions in the form

$$p(t,z_1,\cdots,z_N) = \sum_{r=1}^{R} \alpha_r(t) p_1^r(z_1,t) p_2^r(z_2,t) \cdots p_N^r(z_N,t). \tag{10}$$

Here, we demonstrate the procedure with reference to the simple Crank-Nicolson scheme. To this end, we consider the linear kinetic equation in the form

$$\frac{\partial p(\mathbf{z},t)}{\partial t} = L(\mathbf{z}) p(\mathbf{z},t), \tag{11}$$

where $\mathbf{z} = (z_1,...,z_N)$ is the vector of phase variables and $L(\mathbf{z})$ is a linear operator. For instance, in the case of the Fokker-Planck equation we have

$$L(\mathbf{z}) = -\sum_{j=1}^{N} \left( \frac{\partial G_j}{\partial z_j} + G_j \frac{\partial}{\partial z_j} \right) + \frac{1}{2} \sum_{i,j=1}^{N} \left( \frac{\partial^2 b_{ij}}{\partial z_i \partial z_j} + b_{ij} \frac{\partial^2}{\partial z_i \partial z_j} + 2 \frac{\partial b_{ij}}{\partial z_i} \frac{\partial}{\partial z_j} \right).$$

We discretize (11) in time by using the Crank-Nicolson scheme. This yields

$$\frac{p(\mathbf{z},t_{k+1}) - p(\mathbf{z},t_k)}{\Delta t} = \frac{1}{2} \left( L(\mathbf{z}) p(\mathbf{z},t_{k+1}) + L(\mathbf{z}) p(\mathbf{z},t_k) \right) + \tau_k(\mathbf{z}), \qquad \Delta t = t_{k+1} - t_k,$$

i.e.,

$$\left( I - \frac{1}{2} \Delta t L(\mathbf{z}) \right) p(\mathbf{z},t_{k+1}) = \left( I + \frac{1}{2} \Delta t L(\mathbf{z}) \right) p(\mathbf{z},t_k) + \tau_k(\mathbf{z}), \tag{12}$$

where $\tau_k(\mathbf{z})$ is the truncation error arising from the temporal discretization. Assuming that $p(\mathbf{z},t_k)$ is known, (12) is a linear equation for $p(\mathbf{z},t_{k+1})$ which can be written concisely (at each time step) as

$$A(\mathbf{z}) p(\mathbf{z}) = f(\mathbf{z}) + \tau(\mathbf{z}), \tag{13}$$

where

$$A(\mathbf{z}) \doteq \left( I - \frac{1}{2} \Delta t L(\mathbf{z}) \right), \qquad f(\mathbf{z}) \doteq \left( I + \frac{1}{2} \Delta t L(\mathbf{z}) \right) p(\mathbf{z},t_k).$$

Note that we dropped the time $t_{k+1}$ in $p(\mathbf{z},t_{k+1})$ with the understanding that the linear system (13) has to be solved at each time step. We emphasize that other multistep and time-splitting schemes [27,58] - including geometric integrators [45] - can be used instead of the Crank-Nicolson method.

### 2.2.2 Alternating Direction Algorithms

The low-rank tensor decomposition is particularly convenient when the system operator $A(\mathbf{z})$ and the right-hand-side $f(\mathbf{z})$ are separable with respect to $\mathbf{z}$, i.e.,

$$A(\mathbf{z}) = \sum_{k=1}^{n_A} A_1^k(z_1) \cdots A_N^k(z_N), \qquad f(\mathbf{z}) = \sum_{k=1}^{n_f} f_1^k(z_1) \cdots f_N^k(z_N). \tag{14}$$

Note that $A(\mathbf{z})$ is separable if $L(\mathbf{z})$ is separable. A simple example of a two-dimensional separable operator $L(\mathbf{z})$ with separation rank $n_L = 3$ is

$$L(z_1, z_2) = z_2 \frac{\partial^2}{\partial z_1 \partial z_2} + \sin(z_1) z_2 \frac{\partial^2}{\partial z_1^2} + e^{-z_1^2} \frac{\partial}{\partial z_2}. \tag{15}$$

Another example is the Liouville operator associated to nonlinear dynamical systems with polynomial nonlinearities. A substitution of the tensor representation (5) into (12) yields the residual[2]

$$W(\mathbf{z}) = A(\mathbf{z}) p(\mathbf{z}) - f(\mathbf{z}), \tag{16}$$

which depends on $\mathbf{z}$ and on all degrees of freedom associated with $p_j^r$. To determine such degrees of freedom we require that

$$\|W(\mathbf{z})\| = \|A(\mathbf{z}) p(\mathbf{z}) - f(\mathbf{z})\| \le \varepsilon, \tag{17}$$

in an appropriately chosen norm, and for a prescribed target accuracy $\varepsilon$. Ideally, the optimal tensor rank of can be defined as the minimal $R$ such that the solution has an exact tensor decomposition with $R$ terms, i.e., $\varepsilon = 0$. However, the storage requirements and the computational cost increase with $R$, which makes the tensor decomposition attractive for small $R$. Therefore, we look for a low-rank tensor approximation of the solution to (13), with a reasonable accuracy $\varepsilon$. Although there are at present no useful theorems on the size $R$ needed for a general class of functions, there are examples where tensor expansions are exponentially more efficient than one would expect a priori (see [5]).

Many existing algorithms to determine the best low-rank approximation of the solution to (13) are based on alternating direction methods. The key idea is to construct the tensor expansion (5) iteratively by determining $p_j^r(z_j)$ one at a time while freezing the degrees of freedom associated with all other functions. This yields a sequence of low-dimensional problems that can be solved efficiently [5, 6, 59, 79, 80, 83], eventually in parallel [52]. Perhaps, one of the first alternating direction algorithms to compute a low rank representation of the solution of a high-dimensional PDE was the one proposed in [2]. To clarify how the method works in simple terms, suppose we have constructed an approximated solution to the system

---

[2] The residual $W(\mathbf{z})$ incorporates both the truncation error arising from the time discretization as well as the error arising from the finite-dimensional expansion (5).

(12) in the form (5), i.e., suppose we have available $p^R(\mathbf{z})$ with tensor rank $R$. Then we look for an *enriched* solution in the form

$$p^R(\mathbf{z}) + r_1(z_1) \cdots r_N(z_N),$$

where $\{r_1(z_1), ..., r_N(z_N)\}$ are $N$ unknown functions to be determined. In the alternating direction method, such functions are determined iteratively, one at a time. Typical algorithms to perform such iterations are based on alternating least squares (ALS),

$$\min_{r_j} \left\| \sum_{k=1}^{n_A} A_1^k \cdots A_N^k \left( p^R + r_1 \cdots r_N \right) - \sum_{k=1}^{n_f} f_1^k \cdots f_N^k \right\|^2, \tag{18}$$

or alternating Galerkin methods,

$$\left\langle q, \sum_{k=1}^{n_A} A_1^k \cdots A_N^k \left( p^R + r_1 \cdots r_N \right) \right\rangle = \left\langle q, \sum_{k=1}^{n_f} f_1^k \cdots f_N^k \right\rangle, \tag{19}$$

where $\langle \cdot \rangle$ is an inner product (multi-dimensional integral with respect to $\mathbf{z}$), and $q$ is a test function, typically chosen as $q(\mathbf{z}) = r_1(z_1) \cdots \phi_{j,k}(z_j) \cdots r_N(z_N)$ for $k = 1, ..., q_z$. In a finite-dimensional setting, the minimization problem (18) reduces to the problem of finding the minimum of a scalar function in as many variables as the number of unknowns we consider in each basis function $r_j(z_j)$, say $q_z$. Similarly, the alternating direction solution to (19) yields a sequence of low-dimensional linear systems of size $q_z \times q_z$. If $A(\mathbf{z})$ is a nonlinear operator, then we can still solve (18) or (19), e.g., by using Newton iterations. Once the functions $\{r_1(z_1), ..., r_N(z_N)\}$ are computed, they are normalized (yielding the normalization factor $\alpha_{R+1}$) and added to $p^R(\mathbf{z})$ to obtain $p^{R+1}(\mathbf{z})$. The tensor rank is increased until the norm of the residual (16) is smaller than the desired target accuracy $\varepsilon$ (see equation (17)). We would like to emphasize that it is possible to include additional constraints when solving the linear system (13) with alternating direction algorithms. For example, one can impose that the solution $p(\mathbf{z})$ is positive and it integrates to one [59], i.e., it is a probability density function.

The enrichment procedure just described has been criticized in the literature due to its slow convergence rate, in particular for equations dominated by advection [79]. Depending on the criterion used to construct the tensor decomposition, the enrichment procedure might not even converge. To overcome this problem, Doostan and Iaccarino [25] proposed an alternating least-square (ALS) algorithm with granted convergence properties. The algorithm simultaneously updates the entire rank of the basis set in the $j$-th direction. In this formulation, the least square approach (18) becomes

$$\min_{\left\{ p_j^1, ..., p_j^R \right\}} \left\| \sum_{k=1}^{n_A} A_1^k \cdots A_N^k \left( \sum_{r=1}^{R} \alpha_r p_1^r \cdots p_N^r \right) - \sum_{k=1}^{n_f} f_1^k \cdots f_N^k \right\|^2.$$

The computational cost of this method clearly increases compared to (18). In fact, in a finite dimensional setting, the simultaneous determination of $\{p_j^1,...,p_j^R\}$ requires the solution of a $Rq_z \times Rq_z$ linear system. However, this algorithm usually results in a separated solution with a lower tensor rank $R$ than the regular approach, which makes the algorithm more favorable to advection dominated kinetic systems. The basic idea of updating the entire rank of functions depending on a specific variable can be also applied to the alternating Galerkin formulation (19) (see [20]). In section 4 we provide a numerical example of such algorithm - see also Algorithm 1.

Further developments and applications of low-rank tensor approximation methods can be found in the excellent reviews papers [3, 40, 81]. Gradient-based and Newton-like methods modifying and improving the basic ALS algorithm are discussed in [1, 14, 28, 34, 54, 88, 93, 105, 106], Convergence of ALS and its parallel implementation has been studied in [21, 52, 70, 108].

## 2.3 ANOVA Decomposition and BBGKY Hierarchies

Another typical approach to model high-dimensional functions is based on the truncation of interactions. Hereafter we discuss two different methods to perform such approximation, namely, the ANOVA decomposition [11, 36, 60, 125] and the BBGKY (Bogoliubov-Born-Green-Kirkwood-Yvon) technique. Both these methods rely on a representation of multivariate functions in terms of series expansions involving functions with a smaller number of variables. For example, a second-order ANOVA approximation of a multivariate PDF in $N$ variables is a series expansion involving functions of at most two variables. As we will see, both the ANOVA decomposition and the BBGKY technique [73] yield a hierarchy of coupled PDF equations for each given stochastic dynamical system. These methods are especially appropriate for anisotropic problems where dimensional adaptivity can be pursued.

The ANOVA series expansion [11, 41, 120] involves a superimposition of functions with an increasing number of variables. Specifically, the ANOVA decomposition of an $N$-dimensional PDF takes the from

$$p(z_1,z_2,...,z_N) = p_0 + \sum_{i=1}^{N} p_i(z_i) + \sum_{i<j}^{N} p_{ij}(z_i,z_j) + \sum_{i<j<k}^{N} p_{ijk}(z_i,z_j,z_k) + \cdots . \quad (20)$$

The function $p_0$ is a constant. The functions $p_i(z_i)$, which we shall call first-order interaction terms, give us the overall effects of the variables $z_i$ in $p$ as if they were acting independently of the other variables. The functions $p_{ij}(z_i,z_j)$ represent the interaction effects of the variables $z_i$ and $z_j$, and therefore they will be called second-order interactions. Similarly, higher-order terms reflect the cooperative effects of an increasing number of variables, and the series is usually truncated at a certain interaction order. These terms can be computed in different ways [92,124], however, we point out the following procedure,

$$p_K(z_K) = \int p(z)d\mu(z_{K'}) - \sum_{S \subset K} p_T(z_S), \tag{21}$$

where $S \subset K \subset \{1, \cdots, N\}$, $K'$ is the complement of $K$ in $\{1, \cdots, N\}$, $p_K(z_K) = p_{j_1, \ldots, j_k}(z_{j_1}, \cdots, z_{j_k})$ for $K = \{j_1, \cdots, j_k\}$, and $\mu$ is the Lebesgue measure. Due to its construction, this procedure generates ANOVA terms that are orthogonal with respect to $\mu$, that is, $\int p_K(z_K)p_S(z_S)d\mu(z)$, for all $S \neq K$, which provides an effective criterion for dimensional adaptivity [65, 120].

The ANOVA expansion can be readily applied in the space of parameters of kinetic systems since the parameters do not depend on time and each terms computed at the initial time can be updated independently. To pursue a collocation approach similar to the sparse grid collocation method (3), we replace the Lebesgue measure with a Dirac measure $d\mu = \delta(z - \mathbf{c})$ at an appropriate anchor point $\mathbf{c}$, and consider the corresponding collocation scheme [118]. This method is called the anchored-ANOVA method (PCM-ANOVA) [7, 32, 36, 120]. The anchor points are often taken as the mean value of the random variable in each dimension [125]. Then, each PDF equations in Table 1 can be solved at the *PCM-ANOVA collocation* points in the space of parameters.

On the other hand, representing the dependence of the solution PDF on the phase variables through the ANOVA expansion yields a hierarchy of coupled PDF equations that resembles the BBGKY hierarchy of classical statistical mechanics. Let us briefly review the BBGKY technique type with reference to a nonlinear dynamical system in the form

$$\dot{\mathbf{z}}(t) = \mathbf{G}(\mathbf{z}, t), \qquad \mathbf{z}(0) = \mathbf{z}_0(\omega), \tag{22}$$

where $\mathbf{z}(t) \in \mathbb{R}^N$ is a multi-dimensional stochastic process including both phase and parametric variables, $\mathbf{G} : \mathbb{R}^{N+1} \to \mathbb{R}^N$ is a Lipschitz continuous (deterministic) function, and $\mathbf{z}_0 \in \mathbb{R}^N$ is a random initial state. The joint PDF of $\mathbf{z}(t)$ evolves according to the Liouville equation

$$\frac{\partial p(\mathbf{z}, t)}{\partial t} + \nabla \cdot [\mathbf{G}(\mathbf{z}, t)p(\mathbf{z}, t)] = 0, \qquad \mathbf{z} \in \mathbb{R}^N, \tag{23}$$

whose solution can be computed numerically with standard discretization methods only for relatively small $N$. This leads us to look for PDF equations involving only a reduced number of phase variables, for instance, the PDF of each component $z_i(t)$. Such equations can be formally obtained by marginalizing (23) with respect to different phase variables and discarding terms at infinity. This yields, for example,

$$\frac{\partial p_i(z_i, t)}{\partial t} = -\frac{\partial}{\partial z_i} \int [G_i(\mathbf{y}, t)\delta(z_i - y_i(t))p(\mathbf{y}, t)] \, d\mathbf{y}, \tag{24}$$

$$\frac{\partial p_{ij}(z_i, z_j, t)}{\partial t} = -\frac{\partial}{\partial z_i} \int [G_i(\mathbf{y}, t)\delta(z_i - y_i(t))\delta(z_j - y_j(t))p(\mathbf{y}, t)]\, d\mathbf{y}$$

$$-\frac{\partial}{\partial z_j} \int [G_j(\mathbf{y}, t)\delta(z_i - y_i(t))\delta(z_j - y_j(t))p(\mathbf{y}, t)]\, d\mathbf{y}. \quad (25)$$

Higher-order PDF equations can be derived similarly. The computation of the integrals in (24) and (25) requires the full joint PDF of $\mathbf{z}(t)$, which is available only if we solve the full Liouville equation (23). Alternatively, we can solve (24) or (25) directly, provided we need to introduce approximations. The most common one is to assume that the joint PDF $p(\mathbf{z}, t)$ can be written in terms of lower-order PDFs, e.g., as $p(\mathbf{z}, t) = p(z_1, t) \cdots p(z_N, t)$ (mean-field approximation). By using integration by parts, this assumption reduces the Liouville equation to a hierarchy of low-dimensional PDF equations (see, e.g., [20,113]). An example of such approximation will be presented later in this chapter with an application to Lorenz-96 model.

## 3 Computational Cost

Consider a kinetic partial differential equation with $n$ phase variables and $m$ parameters, i.e., a total number of $N = n + m$ variables. Suppose that we represent the solution by using $q_z$ degrees of freedom in each phase variable and $q_b$ degrees of freedom in each parameter. If we employ a tensor product discretization, the number of degrees of freedom becomes $q_z^n \cdot q_b^m$ and the computational cost grows exponentially as $O(q_z^{2n} \cdot q_b^m)$. Hereafter we compare the computational cost of the methods we discussed in the previous sections. Table 2 summarizes the main results.

### 3.1 Sparse Grids

The computational complexity of sparse grids grows logarithmically with the number of degrees of freedom in each dimension, i.e., $O(q_z |\log_2(q_z)|^{n-1})$. If we employ the multi-wavelet basis we mentioned before in the context of the discontinuous Galerkin framework, then it can be shown that the computational complexity is $O((q_z + 1)^n 2^\ell \ell^{n-1})$, where $\ell$ is the element level and $q_z$ is the polynomial order in each element (see [43]). In the space of parameters, the sparse grid collocation method yields $2^l (m + l)!/(m!l!)$ points, where $l$ is the sparse grid level and $m$ is the number of parameters. Thus, if we consider sparse grid in both phase and parametric space, the total computational cost can be estimated as $O(q_z^2 |\log_2(q_z)|^{2n-2}) \cdot \sum_{l=0}^\ell 2^l (m + l)!/(m!l!)$.

## 3.2 Low-rank Tensor Approximation

The total number of degrees of freedom in a low-rank tensor decomposition grows *linearly* with both $n$ and $m$. For instance, we have $R(nq_z + mq_b)$ in the canonical tensor decomposition (6), and $R^2(nq_z + mq_b)$ in the tensor train approach (8). If the tensor rank $R$ turns out to be relatively small, then the tensor approximation is far more efficient than full tensor product, sparse grid, or ANOVA approaches, in terms of memory requirements as well as the computational cost. The classical alternating direction algorithm at the basis of the canonical tensor decomposition can be divided into two steps, i.e., the enrichment and the projection steps (see Algorithm 1). The computational cost of the projection step can be neglected with respect to the one of the enrichment step, as it reduces to solving a linear system of rather small size $(r \times r)$. The enrichment step at tensor rank $r$ requires $O((rq_z)^2 + (rq_z)^2)$ operations - provided we employ appropriate iterative linear solvers. If we assume that the average number of iterations is $n_{itr}$, and sum up the cost for $r = 1, ..., R$, the overall computational cost of canonical tensor decomposition can be estimated as $O\left(R^3\left(nq_z^2 + mq_b^2\right)\right) \cdot n_{itr}$. In the tensor train approach, the cost also depends on the matrix rank $S$ that comes from the procedure of HOSVD, and it becomes $O\left(R^2 S^2 nq_z^2 + R^3 S^3 nq_z\right)$ [58].

## 3.3 ANOVA Decomposition

If we consider the ANOVA expansion or the BBGKY hierarchy, the computational complexity has a factorial dependency on the dimensionality $n + m$ and the interaction orders of the variables [32]. In particular, the total number of degrees of freedom for a fixed interaction order $\ell$ and assuming $q_b = q_z$ is

$$\sum_{l=0}^{\ell} C(n+m, l, q_z) \qquad \text{where} \qquad C(N, l, q_z) = q_z^l \frac{N!}{(N-l)! l!}. \qquad (26)$$

The computational cost of matrix-vector operations involving discretized variables in each level is $O\left(C(n+m, \ell, q_z^{2\ell})\right)$. It is possible to combine the BBGKY technique with the PCM-ANOVA approach to improve the accuracy, since the interaction order of the phase variables and the parameters, denoted as $\ell$ and $\ell'$, can be controlled separately. In this case, the total number of degrees of freedom and the corresponding computational cost become, $\left(\sum_{l=0}^{\ell} C(n, l, q_z)\right) \cdot \left(\sum_{l=0}^{\ell'} C(m, l, q_b)\right)$ and $O\left(C(n, \ell, q_z^{2\ell}) \cdot \left(\sum_{l=0}^{\ell'} C(m, l, q_b)\right)\right)$, respectively.

|  | Degrees of freedom | Computational Cost |
|---|---|---|
| Sparse grids | $O(q_z|\log(q_z)|^{n-1})$ | $O\left(q_z^2|\log(q_z)|^{2n-2}\right)$ |
| ANOVA or BBGKY | $\sum_{s=0}^{\ell} q_z^s \dfrac{n!}{(n-s)!s!}$ | $O\left(q_z^{2\ell} \dfrac{n!}{(n-\ell)!\ell!}\right)$ |
| Canonical tensor decomposition | $q_z R n$ | $O\left(q_z^2 R^3 n\right)$ |

**Table 2** Number of degrees of freedom and computational cost of solving kinetic equations by using different methods. Shown are results for sparse grid, ANOVA decomposition, and low-rank canonical tensor decomposition. In the Table, $n$ is the phase space dimension in the kinetic equation assuming that the PDF solution is discretized with $q_z$ degrees of freedom in each phase variable. Also, $R$ is the tensor rank and $\ell$ is the interaction orders of the ANOVA expansion or the BBGKY closure.

# 4 Applications

In this section, we present numerical examples to illustrate the performance and accuracy of the algorithms we discussed in this chapter. Specifically, we study the alternating Galerkin formulation (canonical tensor decomposition) of a kinetic model describing stochastic advection of a scalar field. We also study the BBGKY hierarchy of the Lorentz-96 model evolving from a random initial state.

## 4.1 Stochastic Advection of Scalar Fields

Let us consider the following stochastic advection equations

$$\frac{\partial u}{\partial t} + \left(1 + \sum_{k=1}^{m} \frac{1}{2k} \sin(kt)\xi_k(\omega)\right)\frac{\partial u}{\partial x} = 0, \tag{27}$$

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \sin(t)\sum_{k=1}^{m} \frac{1}{5(k+1)} \sin((k+1)x)\xi_k(\omega), \tag{28}$$

where $x \in [0, 2\pi]$ and $\{\xi_1, ..., \xi_m\}$ are i.i.d. uniform random variables in $[-1, 1]$. The kinetic equations governing the joint probability density function of $\{\xi_1, ..., \xi_m\}$ and the solution to (27) or (28) are, respectively,

$$\frac{\partial p}{\partial t} + \left(1 + \sum_{k=1}^{m} \frac{1}{2k} \sin(kt)b_k\right)\frac{\partial p}{\partial x} = 0, \tag{29}$$

$$\frac{\partial p}{\partial t} + \frac{\partial p}{\partial x} = -\left(\sin(t)\sum_{k=1}^{m} \frac{1}{5(k+1)} \sin((k+1)x)b_k\right)\frac{\partial p}{\partial a}, \tag{30}$$

where $p = p(x,t,a,\mathbf{b})$, $\mathbf{b} = \{b_1, ..., b_m\}$ (see [112] for a derivation). Note that this PDF depends on $x$, $t$, one phase variable $a$ (corresponding to $u(x,t)$), and $m$ parameters $\mathbf{b}$ (corresponding to $\{\xi_1, ..., \xi_m\}$). The analytical solutions to Eqs. (29) and (30) can be obtained by using the method of characteristics [95]. They are both in the form

$$p(x,t,a,\mathbf{b}) = p_0(x - X(t,\mathbf{b}), a - A(x,t,\mathbf{b}), \mathbf{b}) \tag{31}$$

where

$$X(t,\mathbf{b}) = t - \sum_{k=1}^{m} \frac{(\cos(kt) - 1)b_k}{2k^2}, \qquad A(x,t,\mathbf{b}) = 0 \tag{32}$$

in the case of equation (29) and

$$X(t,\mathbf{b}) = t, \quad A(x,t,\mathbf{b}) = \sum_{k=2}^{m+1} \frac{b_{k-1}}{10k} \left( \frac{\sin(kx-t)}{k-1} - \frac{\sin(kx+t)}{k+1} - \frac{2\sin(k(x-t))}{(k-1)(k+1)} \right) \tag{33}$$

in the case of equation (30). Also, $p_0(x,a,\mathbf{b})$ is the joint PDF of $u(x,t_0)$ and $\{\xi_1, ..., \xi_m\}$. In our simulations we take

$$p_0(x,a,\mathbf{b}) = \frac{1}{2} \left( \frac{\sin^2(x)}{2\pi\sigma_1} \exp\left[ -\frac{(a-\mu_1)^2}{2\sigma_1} \right] + \frac{\cos^2(x)}{2\pi\sigma_2} \exp\left[ -\frac{(a-\mu_2)^2}{2\sigma_2} \right] \right),$$

which has tensor rank $R = 2$. Non-separable initial conditions can be approximated in the tensor format (5). Also, we consider different number of parameters in equations (29) and (30), i.e., $m = 3, 13, 24, 54, 84, 114$.

### 4.1.1 Finite-Dimensional Representations

Let us represent the joint probability density function (5) in terms of polynomial basis functions as

$$p_n^r(z_n) = \sum_{k=1}^{q_z} \mathrm{p}_{n,k}^r \phi_{n,k}(z_n), \tag{34}$$

where $q_z$ is the number of degrees of freedom in each variable. In particular, for (29) and (30), we consider a spectral collocation method in which $\{\phi_{1,j}\}$ and $\{\phi_{2,j}\}$ are trigonometric polynomials, while $\{\phi_{n,j}\}_{n=3}^{N}$ (basis elements for the space of parameters) are Lagrange interpolants at Gauss-Legendre-Lobatto points. The finite-dimensional representation of the joint PDF admits the following canonical tensor form

$$\mathbf{p} = \sum_{r=1}^{R} \alpha_d \mathbf{p}_1^r \otimes \cdots \otimes \mathbf{p}_N^r,$$

where the vector

$$\mathbf{p}_n^r = \left[ \mathrm{p}_{n,1}^r, \cdots, \mathrm{p}_{n,q_z}^r \right],$$

collects the (normalized) values of the solution at the collocation points. The fully discrete Galerkin formulation of our kinetic equations takes the form

$$\mathbf{Ap} = \mathbf{f}, \tag{35}$$

where

$$\mathbf{A} = \sum_{k=1}^{n_A} \mathbf{A}_1^k \otimes \cdots \otimes \mathbf{A}_N^k, \qquad \mathbf{f} = \sum_{k=1}^{n_f} \mathbf{f}_1^k \otimes \cdots \otimes \mathbf{f}_N^k, \tag{36}$$

$$\mathbf{A}_n^k[i,j] = \int \phi_{n,i}(z_n) A_n^k(z_n) \phi_{n,j}(z_n)\, dz_n, \qquad \mathbf{f}_n^k[i] = \int f_n^k(z_n) \phi_{n,i}(z_n)\, dz_n. \tag{37}$$

By using a Gauss quadrature rule to evaluate the integrals, we obtain system matrices $\mathbf{A}_n^k$ that are either diagonal or coincide with the classical differentiation matrices of spectral collocation methods [47]. For example, in the case of equation (29) we have

$$\mathbf{A}_1^1[i,j] = \mathbf{w}_x[i]\delta_{ij}, \quad \mathbf{A}_1^k[i,j] = \frac{\Delta t}{2} \mathbf{w}_x[i] \mathscr{D}_x[i,j],\ k = 2,...,n_A,$$

$$\mathbf{A}_2^1[i,j] = \mathbf{A}_2^2[i,j] = \mathbf{w}_z[i]\delta_{ij}, \quad \mathbf{A}_2^{k+2}[i,j] = \frac{\sin(kt_{n+1})}{2k} \mathbf{w}_z[i]\delta_{ij},\ k = 1,...,m,$$

$$\mathbf{A}_3^k[i,j] = \mathbf{w}_b[i]\delta_{ij},\ k \neq 3, \quad \mathbf{A}_3^3[i,j] = \mathbf{w}_b[i]\mathbf{q}_b[i]\delta_{ij}, \quad \cdots$$

where $\mathbf{q}_b$ denotes the vector of collocation points, $\mathbf{w}_x$, $\mathbf{w}_z$, and $\mathbf{w}_b$ are collocation weights, $\mathscr{D}_x$ is the differentiation matrix, and $\delta_{ij}$ is the Kronecker delta function. In an alternating direction setting, we aim at solving the system (35) in a greedy way, by freezing all degrees of freedom except those representing the dimension $n$. This yields a sequence of linear systems

$$\mathbf{B}_n \mathbf{p}_n^R = \mathbf{g}_n, \tag{38}$$

where $\mathbf{B}_n$ is a block matrix with $R \times R$ blocks of size $q_z \times q_z$, and $\mathbf{g}_n$ is multi-component vector. Specifically, the $hv$-th block of $\mathbf{B}_n$ and the $h$-th component of $\mathbf{g}_n$ are obtained as

$$\mathbf{B}_n^{hv} = \sum_{k=1}^{n_A} \left( \prod_{\substack{i=1 \\ i \neq n}}^{N} \left[ \mathbf{p}_i^h \right]^T \mathbf{A}_i^k \mathbf{p}_i^v \right) \mathbf{A}_n^k, \qquad \mathbf{g}_n^h = \sum_{k=1}^{n_f} \left( \prod_{\substack{i=1 \\ i \neq n}}^{N} \left[ \mathbf{p}_i^h \right]^T \mathbf{f}_i^k \right) \mathbf{f}_n^k.$$

The solution vector

$$\mathbf{p}_n^R = \left[ \mathbf{p}_n^1, ..., \mathbf{p}_n^R \right]^T$$

is normalized as $\mathbf{p}_n^r / \|\mathbf{p}_n^r\|$ for all $r = 1,..,R$ and $n = 1,...,N$. This operation yields the coefficients $\boldsymbol{\alpha} = (\alpha_1,...,\alpha_R)$ as a solution to the linear systems

$$\mathbf{D}\boldsymbol{\alpha} = \mathbf{d}, \tag{39}$$

where the entries of the matrix $\mathbf{D}$ and the vector $\mathbf{d}$ are, respectively

$$\mathbf{D}^{hv} = \sum_{k=1}^{n_A} \prod_{i=1}^{N} \left[ \mathbf{p}_i^h \right]^T \mathbf{A}_i^k \mathbf{p}_i^v, \qquad \mathbf{d}^h = \sum_{k=1}^{n_f} \prod_{i=1}^{N} \left[ \mathbf{p}_i^h \right]^T \mathbf{f}_i^k.$$

---

**Algorithm 1** Alternating least squares with canonical tensor decomposition.

Compute the tensor representation of the initial condition $\mathbf{p}(t_0)$
**for** $t_1 \leq t_k \leq t_{n_T}$ **do**
    Compute $\mathbf{f}$ by using $\mathbf{p}(t_{k-1})$
    Set $R = 1$
    **while** $\|\mathbf{A}\mathbf{p}^R(t_k) - \mathbf{f}\| > \varepsilon$ **do**
        Initialize $\{\mathbf{p}_1^R(t_k),...,\mathbf{p}_N^R(t_k)\}$ at random
        **while** $\|\mathbf{A}\mathbf{p}^R(t_k) - \mathbf{f}\|$ does not decrease **do**
            Solve $\mathbf{B}_n\mathbf{p}_n^R = \mathbf{g}_n$ (38) in each direction for $1 \leq n \leq N$
        **end while**
        Normalize the basis set and solve $\mathbf{D}\boldsymbol{\alpha} = \mathbf{d}$ (39) to compute the coefficients $\boldsymbol{\alpha}$
        Set $R = R + 1$
    **end while**
**end for**

---

The main steps of the computational scheme are summarized in Algorithm 1. We also refer the reader to [21, 70, 108] for a convergence analysis of the alternating direction algorithm.

The iterative procedure at each time step is terminated when the norm of the residual is smaller than a tolerance, i.e., when $\|\mathbf{A}\mathbf{p}^R - \mathbf{f}\| \leq \varepsilon$. This usually involves the computation of an $N$-dimensional tensor norm, which can be expensive and compromise the computational efficiency of the whole algorithm. To avoid this problem, we replace the condition $\|\mathbf{A}\mathbf{p}^R - \mathbf{f}\| \leq \varepsilon$ with the simpler convergence criterion

$$\max\left\{\frac{\|\widetilde{\mathbf{p}}_1^R - \mathbf{p}_1^R\|}{\|\mathbf{p}_1^R\|}, ..., \frac{\|\widetilde{\mathbf{p}}_N^R - \mathbf{p}_N^R\|}{\|\mathbf{p}_N^R\|}\right\} \leq \varepsilon_1, \tag{40}$$

where $\{\widetilde{\mathbf{p}}_1^R, ..., \widetilde{\mathbf{p}}_N^R\}$ denotes the solution at the previous iteration. This criterion involves the computation of $N$ vector norms instead of one $N$-dimensional tensor norm.

### 4.1.2 Numerical Results: Low-rank Tensor Approximation

We compute the solution to the kinetic equations (29) and (30) by using Algorithm 1. The PDF solution is represented in the canonical tensor format as

$$p(x,t,a,\mathbf{b}) \simeq \sum_{r=1}^{R} \alpha_r(t)p_x^r(x,t)p_a^r(a,t)P_1^r(b_1,t)\cdots P_m^r(b_m,t). \tag{41}$$

We chose the degrees of freedom of the expansion to carefully balance the error between the space and time discretization, as well as the truncation error due to the finite rank $R$. In particular, $x$ and $a$ are discretized in terms of an interpolant with collocation points $q_z = 50$ in each variable, while the parametric dependence on $b_j$ ($j = 1, .., m$) is represented with Legendre polynomials of order $q_b = 7$.
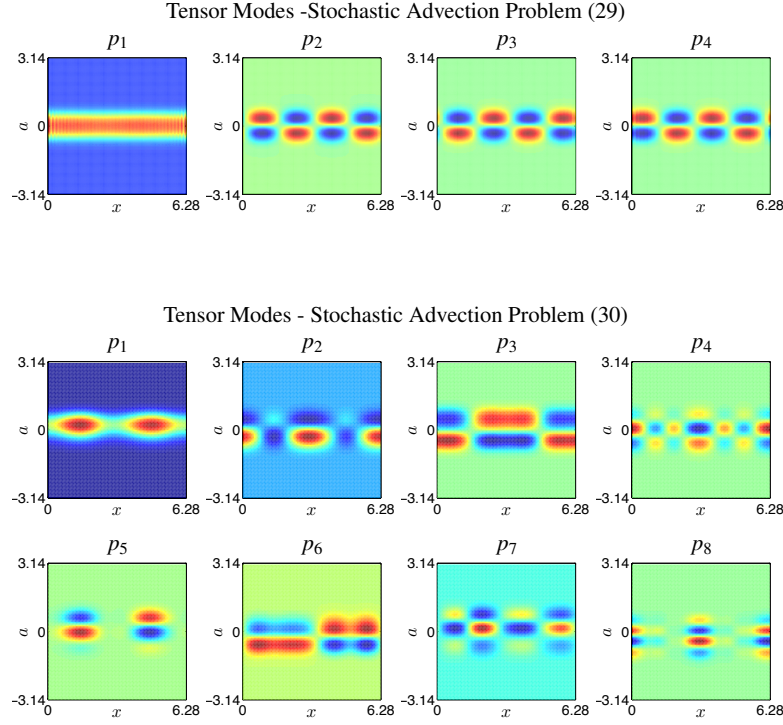
Tensor Modes -Stochastic Advection Problem (29)



Tensor Modes - Stochastic Advection Problem (30)



**Fig. 2** Tensor modes of the kinetic solution to the stochastic advection equations at $t = 2$ [20].
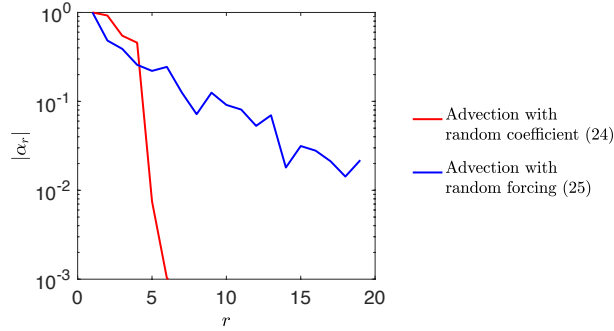


**Fig. 3** Spectra of the canonical tensor decomposition of the stochastic advection problem at $t = 2$.

In Figure 2 we plot the first few tensor modes $p_r(x, a, t) \doteq p_x^r(x, t) p_a^r(a, t)$ of the solution to Eqs. (29) and (30) at time $t = 2$. Specifically, we considered $m = 54$ in (29) and $m = 3$ in (30). Note that the tensor modes we obtain from equation (29), $p_r$, are very similar to each other for $r \geq 2$, while in the case of equation (30) the modes are quite distinct, suggesting the presence of modal interactions and the
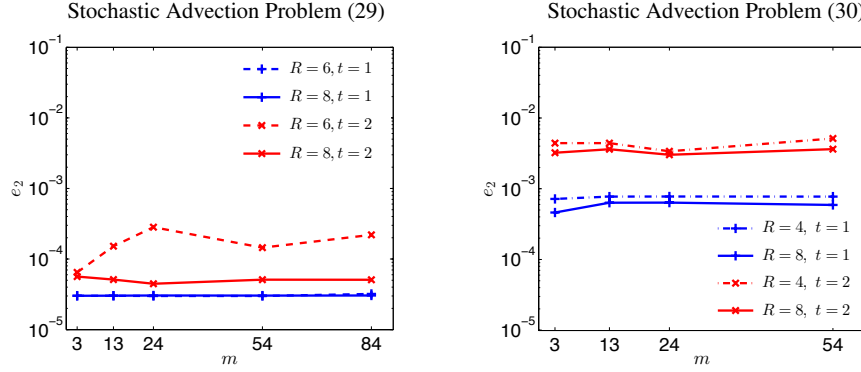
**Fig. 4** Relative $L_2$ errors of the low-rank tensor approximation of the solution with respect to the analytical solution (31). Shown are results for different number of random variables $m$ in (27)-(28) and different tensor ranks $R$. It is seen that the accuracy of the tensor method mainly depends on the actual tensor rank rather than on the dimensionality [20].

need of a larger tensor rank to achieve a certain accuracy. This is also observed in Figure 3, where we plot the normalization coefficients $\{\alpha_1, ..., \alpha_R\}$, which can be interpreted as the spectrum of the tensor solution. The stochastic advection problem with random forcing yields a stronger coupling between the tensor modes, i.e., a slower spectral decay than the problem of random coefficient.

In Figure 4 we plot the error of the low-rank tensor approximation of the solution versus the number of parameters $m$ for different tensor rank $R$. As it is predicted from the spectra shown in Figure 3, the overall relative error of the solution in the random forcing case is larger than in the random coefficient case (see also Figure 5 for the convergence with respect to $R$). This is due to the presence of the time-dependent forcing term in Eq. (28), which injects additional energy in the system and activates new modes. This yields a higher tensor rank for a prescribed level of accuracy. In addition, the plots suggest that the accuracy of the low-rank tensor approximation method depends primarily on the tensor rank rather than on the number of parameters of the problem. The choice of the tensor format that yields the smallest possible tensor rank for a specific problem is an open question. Recent studies suggest that the answer is usually problem-dependent. For instance, Kormann [58] has recently shown that a semi-Lagrangian solver for the Vlasov equation in tensor train format achieves best performances if the phase variables are sorted as $(v_1, x_1, x_2, v_2, x_3, v_3)$.

### 4.1.3 Comparison between Tensor Approximation and ANOVA

In this section we compare the accuracy and the computational cost of the low-rank alternating Galerkin method with the ANOVA expansion technique to compute the solution to Eqs. (29) and (30). The PCM-ANOVA representation of the solution is
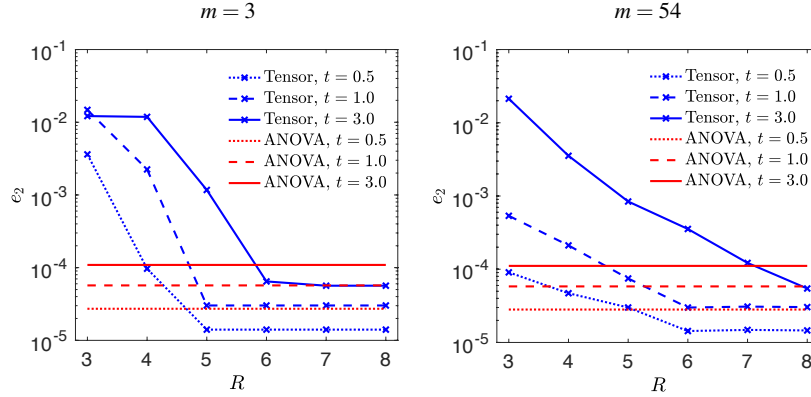
**Fig. 5** Relative $L_2$ errors of the tensor solution and the ANOVA solution of level $\ell = 2$ with respect to the analytical solution (31). Shown are results of the kinetic equation (29) at $t = 0.5$, $t = 1$ and $t = 3$ for different tensor ranks $R$ and dimensionality $m$.

$$p(x,t,a,\mathbf{b}) \simeq \sum_{|K| \le \ell} p_K(t,x,a) P_K(b_K). \tag{42}$$

For $\ell = 2$ (level 2) and $m$ parameters, the expansion (42) has $1 + m + m(m-1)/2$ terms.

In Figure 5 we compare the accuracy of the low-rank tensor approximation and the PCM-ANOVA expansion in computing the solution to the kinetic equation (29). In particular, the convergence of the tensor solution with respect to $R$ is demonstrated. Note that the tensor solution attains the same level of accuracy as the ANOVA decomposition with just five modes for $t \le 1$. Therefore the low-rank tensor approximation is preferable over ANOVA especially when $m \ge 54$. However, this is not true in the case of equation (30) due to its relatively large tensor ranks. To overcome this problem, we developed an adaptive algorithm that sets the separation rank of the solution based on a prescribed target accuracy on the residual of the kinetic equation, or other quantities related to it.

In Figure 6 (left) we plot the temporal dynamics of the tensor rank $R(t)$ obtained by setting a threshold on the spectral condition number defined as the ratio between the smallest and the largest $\alpha_i$. Specifically, we increase $R$ by one at $t = t^*$ whenever the following condition is verified $\alpha_R(t^*)/\alpha_1(t^*) > \theta$. For a small threshold $\theta$, we notice that $R$ can increase to 20 and more at later times. This result reveals two key aspects of efficient tensor algorithms in practical applications. It is essential to develop a robust adaptive procedure that can identify the proper tensor rank *on-the-fly* and an effective compression technique that can reduce the tensor rank in time. This is critical especially when computing long term behavior of kinetic systems.

In Figure 6 (right) we plot the error of the adaptive tensor method and the level 2 ANOVA method versus time. It is seen that error in the tensor method is almost independent of $m$, while the error of ANOVA increases with $m$. The accuracy can be improved either by increasing the tensor rank (canonical tensor decomposition)
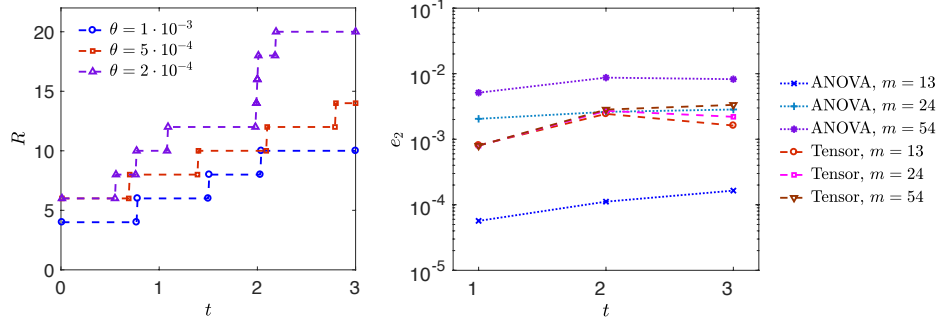
**Fig. 6** Comparison between the relative $L_2$ errors of the adaptive tensor method and the ANOVA method of level $\ell = 2$. Results are for the kinetic equation (30) with threshold $\theta = 5 \cdot 10^{-4}$. It is seen that the error of the tensor solution is slightly independent of $m$, while the error of ANOVA level 2 increases as we increase $m$.
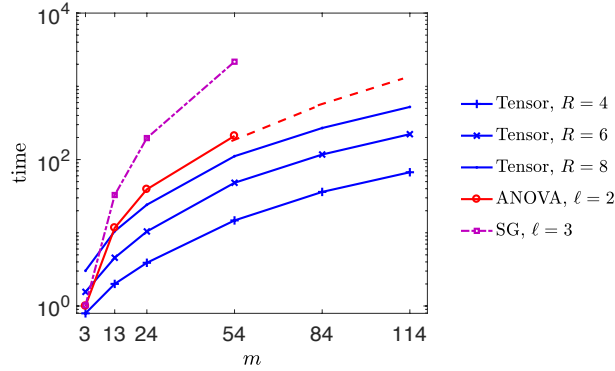


**Fig. 7** Computational time of the tensor decomposition, ANOVA level 2, and sparse grid (SG) level 3 with respect to the dimensionality $m$ and the tensor rank $R$. The results are normalized with respect to the computing time of ANOVA when $m = 3$.

or increasing the interaction order (ANOVA method). Before doing so, however, one should carefully examine the additional computational cost incurred by each method. For example, increasing the interaction order from two to three in the PCM-ANOVA expansion would increase the number of collocation points from 70498 to 8578270 (case $m = 54$). In Figure 7 we compare the computational cost of canonical tensor decomposition with different ranks, ANOVA of level two, and sparse grid of level three in computing the solution to equation (30). It is seen that the tensor method is the most efficient one, in particular for high dimensions and low tensor rank, e.g., $m \geq 24$ and $R \leq 8$.

## 4.2 The Lorenz-96 Model

The Lorenz-96 model is a continuous in time and discrete in space model often used in atmospheric sciences to study fundamental issues related to forecasting and data assimilation [51, 62]. The basic equations are

$$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \qquad i = 1, ..., n. \tag{43}$$

Here we consider $n = 40$, $F = 1$, and assume that the initial state $[x_1(0), ..., x_{40}(0)]$ is jointly Gaussian with PDF

$$p_0(z_1, ..., z_{40}) = \left(\frac{25}{2\pi}\right)^{20} \prod_{i=1}^{40} \exp\left[-\frac{25}{2}\left(z_i - \frac{i}{40}\right)^2\right]. \tag{44}$$

Without an additional parametric space, the dimensionality of this system is $n = 40$. The kinetic equation governing the joint PDF of the phase variables $[x_1(t), ..., x_{40}(t)]$ is

$$\frac{\partial p(\mathbf{z}, t)}{\partial t} = -\sum_{i=1}^{40} \frac{\partial}{\partial z_i}\left[((z_{i+1} - z_{i-2})z_{i-1} - z_i + F)\, p(\mathbf{z}, t)\right], \quad \mathbf{z} \in \mathbb{R}^{40}. \tag{45}$$

Such hyperbolic conservation law cannot be obviously solved in a classical tensor product representation because of high-dimensionality and possible lack of regularity (for $F > 10$) related to the fractal structure of the attractor [51]. Thus, we are led to look for reduced-order PDF equations.

### 4.2.1 Truncation of the BBGKY Hierarchy

In this section we illustrate how to compute low order probability density function equations by truncations of the BBGKY hierarchy. To this end, consider the dynamical system

$$\frac{dy_i}{dt} = G_i(\mathbf{y}, t),$$

where

$$G_i(\mathbf{y}, t) = g_{ii}(y_i, t) + \sum_{\substack{k=1 \\ k \neq i}}^{N} g_{ik}(y_i, y_k, t).$$

With such velocity field $G_i(\mathbf{y}, t)$ we can calculate the integrals at the right hand side of the one-point PDF equation (24) exactly as

$$\frac{\partial p_i}{\partial t} = -\frac{\partial}{\partial z_i}\left[g_{ii}(z_i, t)p_i + \sum_{k \neq i}^{N} \int g_{ik}(z_i, z_k, t)p_{ik}dz_k\right], \tag{46}$$

where $p_i = p(z_i, t)$ and $p_{ik} = p(z_i, z_k, t)$. Similarly, the two-point PDF equations (25) can be approximated as

$$
\frac{\partial p_{ij}}{\partial t} = -\frac{\partial}{\partial z_i} \left[ \left( g_{ii}(z_i, t) + g_{ij}(z_i, z_j, t) \right) p_{ij} + \left( \sum_{k \neq i, j}^{N} \int g_{ik}(z_i, z_k, t) p_{ik} dz_k \right) p_j \right]
$$
$$
- \frac{\partial}{\partial z_j} \left[ \left( g_{jj}(z_j, t) + g_{ji}(z_j, z_i, t) \right) p_{ij} + \left( \sum_{k \neq i, j}^{N} \int g_{jk}(z_j, z_k, t) p_{jk} dz_k \right) p_i \right],
$$
(47)

where we discarded all contributions from the three-point PDFs and the two-point PDFs except the ones interacting with the $i$-th variable. A variance-based sensitivity analysis in terms of Sobol indices [98, 104, 111] can be performed to identify the system variables with strong correlations. This allows us to determine whether it is necessary to add the other two-points correlations or the three-points PDF equations for a certain triple $\{x_k(t), x_i(t), x_j(t)\}$, and to further determine the equation for a general form of $G_i$.

In the specific case of the Lorenz-96 system, we can write equation (46) as

$$
\frac{\partial p_i}{\partial t} = -\frac{\partial}{\partial z_i} \left[ \left( \langle z_{i+1} \rangle - \langle z_{i-2} \rangle \right) \langle z_{i-1} \rangle_{i-1|i} - (z_i - F) p_i \right],
$$
(48)

where $\langle f(\mathbf{z}) \rangle_{i|j} \doteq \int f(\mathbf{z}) p_{ij}(z_i, z_j, t) dz_i$. In order to close such a system within the level of one-point PDFs, $\langle z_{i-1} \rangle_{i-1|i}$ could be replaced, e.g., by $\langle z_{i-1} \rangle p_i(z_i, t)$. Similarly, equation (47) can be written for the two adjacent nodes as

$$
\frac{\partial p_{ii+1}}{\partial t} = -\frac{\partial}{\partial z_i} \left[ z_{i+1} \langle z_{i-1} \rangle_{i-1|i} p_{i+1} - \langle z_{i-2} \rangle \langle z_{i-1} \rangle_{i-1|i} p_{i+1} - (z_i - F) p_{ii+1} \right]
$$
$$
- \frac{\partial}{\partial z_{i+1}} \left[ \langle z_{i+2} \rangle_{i+2|i+1} z_i p_i - \langle z_{i-1} \rangle z_i p_{ii+1} - (z_{i+1} - F) p_{ii+1} \right].
$$
(49)

By adding the two-points closure of one node apart, i.e., $p_{i-1 i+1}(z_{i-1}, z_{i+1}, t)$, the quantity $\langle z_{i-2} \rangle \langle z_{i-1} \rangle_{i-1|i} p_{i+1}$ in the first row and $\langle z_{i-1} \rangle z_i p_{ii+1}$ in the second row can be substituted by $\langle z_{i-2} \rangle_{i-2|i} \langle z_{i-1} \rangle_{i-1|i+1}$ and $\langle z_{i-1} \rangle_{i-1|i+1} z_i p_i$, respectively. In Figure 8, we compare the mean and the standard deviation of the solution to (43) as computed by the one- and two-points BBGKY closures (Eqs. (48) and (49), respectively) and a Monte Carlo simulation with 50000 solution samples. It is seen that the mean of both the one-point and the two-points BBGKY closures basically coincide with the Monte Carlo results. On the other hand, the error in standard deviation is slightly different, and it can be improved in the two-points BBGKY closure (Figure 9).
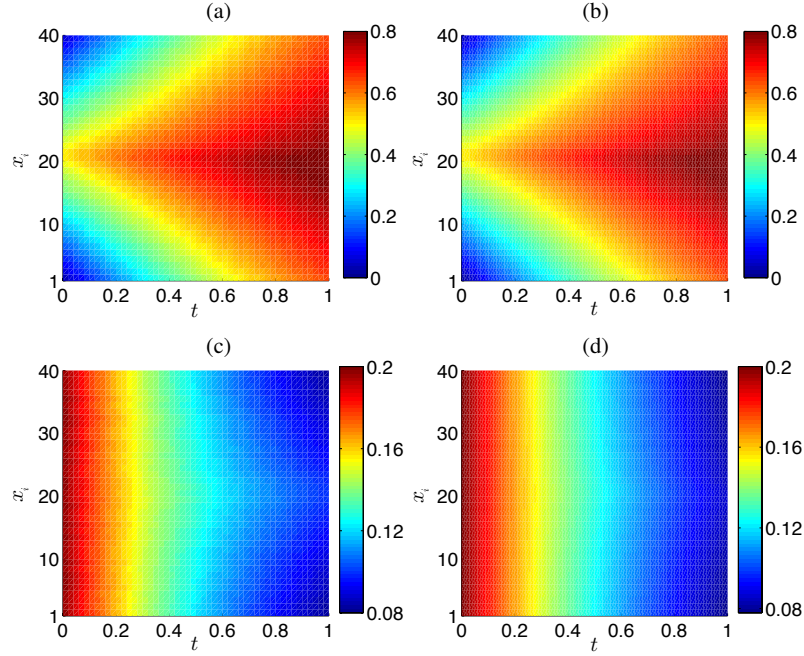
**Fig. 8** Mean (a, b) and standard deviation (c, d) of the Lorenz-96 system computed by the one-point (a) and two-points (c) BBGKY closure compared to the Monte-Carlo simulation (b, d) [20].

## 5 Summary

In this chapter we reviewed state-of-the-art algorithms to compute the numerical solution of high-dimensional kinetic equations. The algorithms are based on low-rank tensor approximation, sparse grids, and ANOVA decomposition. A common feature of these methods is that they allow us to reduce the problem of computing the solution to a high-dimensional PDE to a sequence of low-dimensional problems. The range of applicability of the algorithms is sketched in Figure 1 as a function of the number of phase variables and the number of parameters appearing in the kinetic equation. The computational complexity ranges from logarithmic (sparse grids) to linear (canonical tensor decomposition) with respect to the dimension of the system. Further extensions of the proposed algorithms can be addressed along different directions. For example, adaptive procedures capable of resolving different phase variables with different accuracy may allow applications to kinetic systems with non-smooth solutions and scaling to extremely high-dimensions. In the context of low-rank tensor approximation methods [20, 27, 58], a fundamental question is the development of effective techniques for rank reduction [4, 94]. This is especially challenging for hyperbolic PDEs, since such equations can yield a slow convergence rate when solved with canonical tensor decompositions [20, 79]. Future work should address the development of adaptive algorithms for the construction of controlled
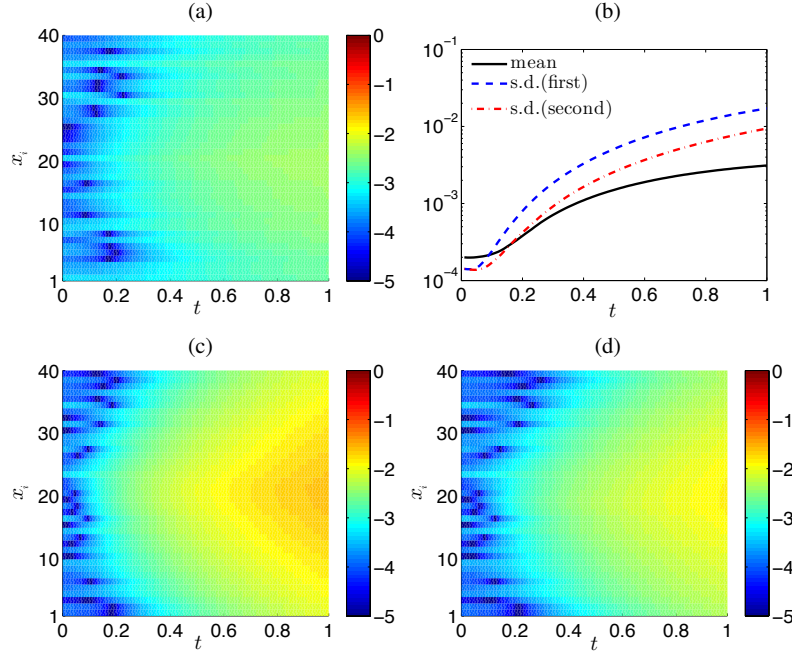
**Fig. 9** The absolute error of the mean (a) and standard deviation (c, d) of the Lorenz-96 system by using the BBGKY closure compared to the Monte-Carlo simulation in log-scale. In (c) and (d), the results are computed by the one- and two-points BBGKY closure, respectively, and the $L_1$ error is shown in (b).

low-rank approximations and an adaptive selection of separation ranks and tensor formats.

# References

1. Acar, E., Dunlavy, D.M., Kolda, T.G.: A scalable optimization approach for fitting canonical tensor decompositions. J. Chemometrics **25**, 67–86 (2011)
2. Ammar, A., Mokdad, B., Chinesta, F., Keunings, R.: A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids: Part II: Transient simulation using space-time separated representations. Journal of Non-Newtonian Fluid Mechanics **144**(2), 98–121 (2007)
3. Bachmayr, M., Schneider, R., Uschmajew, A.: Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations. Foundations of Computational Mathematics **16**, 1423–1472 (2016)

4. Battaglino, C., Ballard, G., Kolda, T.G.: A practical randomized CP tensor decomposition. ArXiv p. 1701.06600 (2017)
5. Beylkin, G., Garcke, J., Mohlenkamp, M.J.: Multivariate regression and machine learning with sums of separable functions. SIAM J. Sci. Comput. **31**(3), 1840–1857 (2009)
6. Beylkin, G., Mohlenkamp, M.J.: Algorithms for numerical analysis in high dimensions. SIAM J. Sci. Comput. **26**(6), 2133–2159 (2005)
7. Bieri, M., Schwab, C.: Sparse high order fem for elliptic SPDEs. Comput. Methods Appl. Mech. Eng. **198**, 1149–1170 (2009)
8. Bird, G.A.: Molecular gas dynamics and direct numerical simulation of gas flows. Clarendon Press (1994)
9. Bolotin, V.V.: Statistical methods in structural mechanics. Holden-Day, San Francisco (1969)
10. Bungartz, H.J., Griebel, M.: Sparse grids. Acta numer. **13**, 147–269 (2004)
11. Cao, Y., Chen, Z., Gunzbuger, M.: ANOVA expansions and efficient sampling methods for parameter dependent nonlinear PDEs. Int. J. Numer. Anal. Model. **6**, 256–273 (2009)
12. Cercignani, C.: The Boltzmann equation and its applications. Springer (1988)
13. Cercignani, C., Gerasimenko, U.I., Petrina, D.Y.: Many particle dynamics and kinetic equations, first edn. Kluwer Academic Publishers (1997)
14. Chen, Y., Han, D., Qi, L.: New ALS methods with extrapolating search directions and optimal step size for complex-valued tensor decompositions. IEEE Transactions on Signal Processing **59**, 5888–5898 (2011)
15. Cheng, Y., Gamba, I.M., Majorana, A., Shu, C.W.: A discontinuous Galerkin solver for Boltzmann-Poisson systems in nano devices. Computer Methods in Applied Mechanics and Engineering **198**, 3130–3150 (2009)
16. Cheng, Y., Gamba, I.M., Majorana, A., Shu, C.W.: A brief survey of the discontinuous Galerkin method for the Boltzmann-Poisson equations. SEMA J. **54**, 47–64 (2011)
17. Chinesta, F., Ammar, A., Cueto, E.: Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. Comput. Methods. Appl. Mech. Engrg. **17**(4), 327–350 (2010)
18. Cho, H., Venturi, D., Karniadakis, G.E.: Adaptive discontinuous Galerkin method for response-excitation PDF equations. SIAM J. Sci. Comput. **35**(4), B890–B911 (2013)
19. Cho, H., Venturi, D., Karniadakis, G.E.: Statistical analysis and simulation of random shocks in Burgers equation. Proc. R. Soc. A **260**, 20140,080(1–21) (2014)
20. Cho, H., Venturi, D., Karniadakis, G.E.: Numerical methods for high-dimensional probability density function equations. J. Comput. Phys. **305**, 817–837 (2016)
21. Comon, P., Luciani, X., de Almeida, A.L.F.: Tensor decompositions, alternating least squares and other tales. J. Chemometrics **23**, 393–405 (2009)
22. Dolgov, S.V., Khoromskij, B.N., Oseledets, I.V.: Fast solution of parabolic problems in the tensor train/quantized tensor train format with initial application to the Fokker-Planck equation. SIAM J. Sci. Comput. **34**(6), A3016–A3038 (2012)
23. Dolgov, S.V., Smirnov, A.P., Tyrtyshnikov, E.E.: Low-rank approximation in the numerical modeling of the Farley-Buneman instability in ionospheric plasma. J. Comp. Phys. **263**, 268–282 (2014)
24. Dominy, J., Venturi, D.: Duality and conditional expectations in the Nakajima-Mori-Zwanzig formulation. J. Math. Phys. p. In Press (2017)
25. Doostan, A., Iaccarino, G.: A least-squares approximation of partial differential equations with high-dimensional random inputs. J. Comp. Phys. **228**(12), 4332–4345 (2009)
26. Dostupov, B.G., Pugachev, V.S.: The equation for the integral of a system of ordinary differential equations containing random parameters. Automatika i Telemekhanica (in Russian) **18**, 620–630 (1957)
27. Ehrlacher, V., Lombardi, D.: A dynamical adaptive tensor method for the Vlasov-Poisson system. J. Comp. Phys. **339**, 285–306 (2017)
28. Espig, M., Hackbusch, W.: A regularized Newton method for the efficient approximation of tensors represented in the canonical tensor format. Numer. Math. **122**, 489–525 (2012)

29. Espig, M., Hackbusch, W., Litvinenko, A., Matthies, H.G., Wähnert, P.: Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats. Comput. Math. Appl. **67**(4), 818–829 (2014)

30. Fiasconaro, A., Spagnolo, B., Ochab-Marcinek, A., Gudowska-Nowak, E.: Co-occurrence of resonant activation and noise-enhanced stability in a model of cancer growth in the presence of immune response. Phys. Rev. E **74**(4), 041,904 (2006)

31. Filbet, F., Russo, G.: High-order numerical methods for the space non-homogeneous Boltzmann equations. J. Comput. Phys. **186**, 457–480 (2003)

32. Foo, J., Karniadakis, G.E.: Multi-element probabilistic collocation method in high dimensions. J. Comp. Phys. **229**, 1536–1557 (2010)

33. Fox, R.O.: Computational Models for Turbulent Reactive Flows. Cambridge University Press (2003)

34. Friedland, S., Mehrmann, V., Pajarola, R., Suter, S.K.: On best rank one approximation of tensors. Numer. Linear Algebra Appl. **20**, 942–955 (2013)

35. Frisch, U.: Turbulence: the legacy of A. N. Kolmogorov. Cambridge University Press (1995)

36. Gao, Z., Hesthaven, J.S.: On ANOVA expansions and strategies for choosing the anchor point. Appl. Math. Comput. **217**(7), 3274–3285 (2010)

37. Garcke, J., Griebel, M.: Sparse grids and applications. Springer (2013)

38. Gradinaru, V.: Fourier transform on sparse grids: code design and the time dependent Schrödinger equation. Computing **80**(1), 1–22 (2007)

39. Grasedyck, L.: Hierarchical singular value decomposition of tensors. SIAM J. Matrix Anal. Appl. **31**, 2029–2054 (2010)

40. Grasedyck, L., Kressner, D., Tobler, C.: A literature survey of low-rank tensor approximation techniques. GAMM Mitteilungen **36**(1), 53–78 (2013)

41. Griebel, M.: Sparse grids and related approximation schemes for higher dimensional problems. In: L.M. Pardo, A. Pinkus, E. Süli, M.J. Todd (eds.) Foundations of Computational Mathematics, Santander 2005, 331, pp. 106–161. Cambridge University Press (2006)

42. Griebel, M., Zumbusch, G.: Adaptive sparse grids for hyperbolic conservation laws. In: Hyperbolic Problems: Theory, Numerics, Applications, pp. 411–422. Springer (1999)

43. Guo, W., Cheng, Y.: An adaptive multiresoluton discontinuous Galerkin method for time-dependent transport equations in multi-dimensions. SIAM J. Sci. Comput. **38**(6), 1–29 (2016)

44. Hackbusch, W., Khoromskij, B.N.: Tensor-product approximation to multidimensional integral operators and Green's functions. SIAM J. Matrix Anal. Appl. **30**(3), 1233–1253 (2008)

45. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration illustrated by the Störmer-Verlet method. Acta Numer. **12**, 399–450 (2003)

46. Hatch, D.R., del Castillo-Negrete, D., Terry, P.W.: Analysis and compression of six-dimensional gyrokinetic datasets using higher order singular value decomposition. J. Comput. Phys. **22**, 4234–4256 (2012)

47. Hesthaven, J.S., Gottlieb, S., Gottlieb, D.: Spectral methods for time-dependent problems. Cambridge Univ. Press (2007)

48. Ibragimov, I., Rjasanow, S.: Three way decomposition for the Boltzmann equation. J. Comput. Math. **27**, 184–195 (2009)

49. Jahnke, T., Huisinga, W.: A dynamical low-rank approach to the chemical master equation. Bull. Math. Bio. **70**, 2283–2302 (2008)

50. Kanwal, R.P.: Generalized functions: theory and technique, second edn. Birkhäuser Boston (1998)

51. Karimi, A., Paul, M.R.: Extensive chaos in the Lorenz-96 model. Chaos **20**(4), 043,105(1–11) (2010)

52. Karlsson, L., Kressner, D., Uschmajew, A.: Parallel algorithms for tensor completion in the CP format. Parallel Compting **57**, 222–234 (2016)

53. Kazeev, V., Khammash, M., Nip, M., Schwab, C.: Direct solution of the chemical master equation using quantized tensor trains. Seminar for Applied Mathematics **2013-04**, 2283–2302 (2013)

54. Kazeev, V.A., Tyrtyshnikov, E.E.: Structure of the Hessian matrix and an economical implementation of Newtons method in the problem of canonical approximation of tensors. Comput. Math. Math. Phys. **50**, 927–945 (2010)
55. Khoromskij, B.N.: Structured data-sparse approximation to high order tensors arising from the deterministic Boltzmann equation. Math. Comp. **76**(259), 1291–1315 (2007)
56. Khoromskij, B.N., Oseledets, I.V.: Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs. Comput. Meth. Appl. Math. **10**(4), 376–394 (2010)
57. Klyatskin, V.I.: Dynamics of stochastic systems. Elsevier Publishing Company (2005)
58. Kormann, K.: A semi-lagrangian Vlasov solver in tersor train format. SIAM J. Sci. Comput. **37**(4), B613–B632 (2015)
59. Leonenko, G., Phillips, T.: On the solution of the Fokker-Planck equation using a high-order reduced basis approximation. Comput. Methods Appl. Mech. Engrg. **199**(1-4), 158–168 (2009)
60. Li, G., Wang, S.W., Rabitz, H., Wang, S., Jaffé, P.: Global uncertainty assessments by high dimensional model representations (HDMR). Chemical Engineering Science **57**(21), 4445–4460 (2002)
61. Li, J., Chen, J.B.: Stochastic dynamics of structures. Wiley (2009)
62. Lorenz, E.N.: Predictability - A problem partly solved. In: ECMWF seminar on predictability: Volume 1, pp. 1–18. Reading (1996)
63. Lucor, D., Su, C.H., Karniadakis, G.E.: Generalized polynomial chaos and random oscillators. Int. J. Numer. Meth. Engng. **60**(3), 571–596 (2004)
64. Ma, X., Zabaras, N.: An adaptive hierarchical sparse grid collocation method for the solution of stochastic differential equations. J. Comp. Phys. **228**, 3084–3113 (2009)
65. Ma, X., Zabaras, N.: An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations. J. Comp. Phys. **229**, 3884–3915 (2010)
66. Malakhov, A.N., Saichev, A.I.: Kinetic equations in the theory of random waves. Radiophysics and Quantum Electronics **17**(5), 526–534 (1974)
67. Marco, G.D., Pareschi, L.: Numerical methods for kinetic equations. Acta Numerica **23**, 369–520 (2014)
68. Markovich, P., Ringhofer, C., Schmeiser, C.: Semiconductor equations. Springer (1989)
69. Matthies, H.G., Zander, E.: Solving stochastic systems with low-rank tensor compression. Linear Algebra Appl. **436**(10), 3819–3838 (2012)
70. Mohlenkamp, M.J.: Musings on multilinear fitting. Linear Algebra Appl. **438**, 834–852 (2013)
71. Monin, A.S., Yaglom, A.M.: Statistical fluid mechanics, volume I. Dover (2007)
72. Monin, A.S., Yaglom, A.M.: Statistical fluid mechanics, volume II. Dover (2007)
73. Montgomery, D.: A BBGKY framework for fluid turbulence. Phys. Fluids **19**(6), 802–810 (1976)
74. Moss, F., McClintock, P.V.E. (eds.): Noise in nonlinear dynamical systems. Volume 1: theory of continuous Fokker-Planck systems. Cambridge Univ. Press (1995)
75. Moss, F., McClintock, P.V.E. (eds.): Noise in nonlinear dynamical systems. Volume 2: theory of noise induced processes in special applications. Cambridge Univ. Press (1995)
76. Moss, F., McClintock, P.V.E. (eds.): Noise in nonlinear dynamical systems. Volume 3: experiments and simulations. Cambridge Univ. Press (1995)
77. Muradoglu, M., Jenny, P., Pope, S.B., Caughey, D.A.: A consistent hybrid finite-volume/particle method for the PDF equations of turbulent reactive flows. J. Comput. Phys. **154**, 342–371 (1999)
78. Nobile, F., Tempone, R., Webster, C.: A sparse grid stochastic collocation method for partial differential equations with random input data. SIAM J. Numer. Analysis **46**(5), 2309–2345 (2008)
79. Nouy, A.: A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. Comput. Methods Appl. Mech. Eng. **199**(23-24), 1603–1626 (2010)

80. Nouy, A.: Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. Comput. Methods Appl. Mech. Eng. **17**, 403–434 (2010)
81. Nouy, A.: Handbook of Uncertainty Quantification, chap. Low-rank tensor methods for model order reduction, pp. 1–26. Springer International Publishing (2016)
82. Nouy, A.: Higher-order principal component analysis for the approximation of tensors in tree-based low rank formats pp. 1–43 (2017)
83. Nouy, A., Maître, O.P.L.: Generalized spectral decomposition for stochastic nonlinear problems. J. Comp. Phys. **228**, 202–235 (2009)
84. Novak, E., Ritter, K.: Simple cubature formulas with high polynomial exactness. Constructive Approximation **15**, 499–522 (1999)
85. Nozaki, D., Mar, D.J., Grigg, P., Collins, J.J.: Effects of colored noise on stochastic resonance in sensory neurons. Phys. Rev. Lett. **82**(11), 2402–2405 (1999)
86. Oseledets, I.V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**(5), 2295–2317 (2011)
87. Phan, A.H., Tichavský, P., Cichocki, A.: CANDECOMP/PARAFAC decomposition of high-order tensors through tensor reshaping. IEEE Transactions on Signal Processing **61**, 4847–4860 (2013)
88. Phan, A.H., Tichavský, P., Cichocki, A.: Low complexity damped gaussnewton algorithms for CANDECOMP/ PARAFAC. SIAM Journal on Matrix Analysis and Applications **34**, 126–147 (2013)
89. Pope, S.B.: A Monte Carlo method for the PDF equations of turbulent reactive flow. Combust. Sci. Technol. **25**, 159–174 (1981)
90. Pope, S.B.: Lagrangian PDF methods for turbulent flows. Annu. Rev. Fluid Mech. **26**, 23–63 (1994)
91. Pope, S.B.: Simple models of turbulent flows. Phys. Fluids **23**(1), 011,301(1–20) (2011)
92. Rabitz, H., Aliş, Ö.F., Shorter, J., Shim, K.: Efficient input–output model representations. Computer Physics Communications **117**(1-2), 11–20 (1999)
93. Rajih, M., Comon, P., Harshman, R.A.: Enhanced line search: a novel method to accelerate PARAFAC. SIAM J. Matrix Anal. Appl. **30**, 1128–1147 (2008)
94. Reynolds, M., Beylkin, G., Doostan, A.: Optimization via separated representations and the canonical tensor decomposition pp. 1–13 (2016)
95. Rhee, H.K., Aris, R., Amundson, N.R.: First-order partial differential equations, volume 1: Theory and applications of single equations. Dover (2001)
96. Risken, H.: The Fokker-Planck equation: methods of solution and applications, second edn. Springer-Verlag (1989). Mathematics in science and engineering, vol. 60
97. Rjasanow, S., Wagner, W.: Stochastic numerics for the Boltzmann equation. Springer (2004)
98. Saltelli, A., Chan, K., Scott, M.: Sensitivity Analysis. John Wiley (2000)
99. Schwab, C., Suli, E., Todor, R.A.: Sparse finite element approximation of high-dimensional transport-dominated diffusion problems. ESAIM: Mathematical Modelling and Numerical Analysis **42**, 777–819 (2008)
100. Shlesinger, M.F., Swean, T.: Stochastically excited nonlinear ocean structures. World Scientific (1998)
101. Shu, R., Hu, J., Jin, S.: A stochastic Galerkin method for the Boltzmann equation with multi-dimensional random inputs using sparse wavelet bases. Numer. Math. Theor. Meth. Appl. **10**(2), 465–488 (2017)
102. Smolyak, S.: Quadrature and interpolation formulas for tensor products of certain classes of functions. Soviet Mathematics, Doklady **4**, 240–243 (1963)
103. Sobczyk, K.: Stochastic differential equations: with applications to physics and engineering. Springer (2001)
104. Sobol, I.M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math. Comput. Simul. **55**, 271–280 (2001)
105. Sterck, H.D.: A nonlinear GMRES optimization algorithm for canonical tensor decomposition. SIAM J. Sci. Comput. **34**, A1351–A1379 (2012)
106. Sterck, H.D., Miller, K.: An adaptive algebraic multigrid algorithm for low-rank canonical tensor decomposition. SIAM J. Sci. Comput. **35**, B1–B24 (2012)

107. Stratonovich, R.L.: Some Markov methods in the theory of stochastic processes in nonlinear dynamical systems. In: F. Moss, P.V.E. McClintock (eds.) Noise in nonlinear dynamical systems (Vol. 1), pp. 16–68. Cambridge Univ. Press (1989)

108. Uschmajew, A.: Local convergence of the alternating least squares algorithm for canonical tensor approximation. SIAM J. Matrix Anal. Appl. **33**, 639–652 (2012)

109. Valino, L.: A field Monte Carlo formulation for calculating the probability density function of a single scalar in a turbulent flow. Flow, Turbul. Combust. **60**(2), 157–172 (1998)

110. Venturi, D.: The numerical approximation of functional differential equations. ArXiv (1604.05250), 1–113 (2016)

111. Venturi, D., Choi, M., Karniadakis, G.E.: Supercritical quasi-conduction states in stochastic Rayleigh-Bénard convection. Int. J. Heat and Mass Transfer **55**(13-14), 3732–3743 (2012)

112. Venturi, D., Karniadakis, G.E.: New evolution equations for the joint response-excitation probability density function of stochastic solutions to first-order nonlinear PDEs. J. Comp. Phys. **231**(21), 7450–7474 (2012)

113. Venturi, D., Karniadakis, G.E.: Convolutionless Nakajima-Zwanzig equations for stochastic analysis in nonlinear dynamical systems. Proc. R. Soc. A **470**(2166), 1–20 (2014)

114. Venturi, D., Sapsis, T.P., Cho, H., Karniadakis, G.E.: A computable evolution equation for the probability density function of stochastic dynamical systems. Proc. Roy. Soc. A **468**, 759–783 (2012)

115. Villani, C.: A review of mathematical topics in collisional kinetic theory. In: S. Friedlander, D. Serre (eds.) Handbook of mathematical fluid mechanics, vol. 1, pp. 71–305. North-Holland (2002)

116. Wang, Z., , Tang, Q., Guo, W., Cheng, Y.: Sparse grid discontinuous galerkin methods for high-dimensional elliptic equations. J. Comp. Phys. **314**, 244–263 (2016)

117. Xiu, D.: Efficient collocatioal approach for parametric uncertainty analysis. Communications in Computational Physics **2**(2), 293–309 (2007)

118. Xiu, D., Hesthaven, J.: High-order collocation methods for differential equations with random inputs. SIAM J. Sci. Comput. **27**(3), 1118–1139 (2005)

119. Yan, L., Guo, L., Xiu, D.: Stochastic collocation algorithms using $\ell_1$-minimization. International Journal for Uncertainty Quantification **2**, 279–293 (2012)

120. Yang, X., Choi, M., Karniadakis, G.E.: Adaptive ANOVA decomposition of stochastic incompressible and compressible fluid flows. J. Comp. Phys. **231**, 1587–1614 (2012)

121. Yang, X., Lei, H., Baker, N.A., Lin, G.: Enhancing sparsity of Hermite polynomial expansions by iterative rotations. J. Comput. Phys. **307**, 94–109 (2016)

122. Yang, Y., Shu, C.W.: Discontinuous Galerkin method for hyperbolic equations involving $\delta$-singularities: negarive-order norma error estimate and applications. Numerische Mathematik **124**, 753–781 (2013)

123. Zeng, C., Wang, H.: Colored noise enhanced stability in a tumor cell growth system under immune response. J. Stat. Phys. **141**(5), 889–908 (2010)

124. Zhang, Z., Choi, M., Karniadakis, G.E.: Anchor points matter in ANOVA decomposition. Proceedings of ICOSAHOM'09, Springer, eds. E. Ronquist and J. Hesthaven (2010)

125. Zhang, Z., Choi, M., Karniadakis, G.E.: Error estimates for the ANOVA method with polynomial chaos interpolation: Tensor product functions. SIAM J. Sci. Comp. **34**(2), 1165–1186 (2012)

126. Zhou, G., Cichocki, A., Xie, S.: Accelerated canonical polyadic decomposition by using mode reduction. IEEE Trans Neural Netw Learn Syst. **24**, 2051–62 (2013)

127. Zwanzig, R.: Memory effects in irreversible thermodynamics. Phys. Rev. **124**, 983–992 (1961)