The Numerical Approximation of Nonlinear Functionals and Functional Differential Equations

Daniele Venturi

Department of Applied Mathematics and Statistics
University of California, Santa Cruz

November 21, 2017

Abstract

The fundamental importance of functional differential equations has been recognized in many areas of mathematical physics, such as fluid dynamics (Hopf characteristic functional equation), quantum field theory (Schwinger-Dyson equations) and statistical physics (equations for generating functionals and effective Fokker-Planck equations). However, no effective numerical method has yet been developed to compute their solution. The purpose of this report is to fill this gap, and provide a new perspective on the problem of numerical approximation of nonlinear functionals and functional differential equations.

Contents

1 Introduction 3
2 Nonlinear Functionals 7
2.1 Functional Derivatives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
3 Approximation of Nonlinear Functionals 12
3.1 Functional Approximation in Finite-Dimensional Function Spaces . . . . . . . . . . . . . . 12
3.1.1 Functional Derivatives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
3.1.2 Distances between Function Spaces and Approximability of Functionals . . . . . . . 16
3.2 Functional Interpolation Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
3.2.1 Interpolation Nodes in Function Spaces . . . . . . . . . . . . . . . . . . . . . . . . . 19
3.2.2 Polynomial Interpolation of Nonlinear Functionals . . . . . . . . . . . . . . . . . . . 21
3.2.3 Porter Interpolants . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
3.2.4 Prenter Interpolants . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
3.2.5 Khlobystov Interpolants . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
3.3 Functional Approximation by Tensor Methods . . . . . . . . . . . . . . . . . . . . . . . . 34
3.3.1 Canonical Tensor Decomposition . . . . . . . . . . . . . . . . . . . . . . . . . . . 36
3.3.2 Tucker Decomposition . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41
3.3.3 Hierarchical Tucker Decomposition . . . . . . . . . . . . . . . . . . . . . . . . . . 43
3.3.4 Tensor Train Decomposition . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 44
3.3.5 Tensor Networks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 44
## 3.4 Generalized Lagrangian Interpolation

### 3.4.1 Optimal Interpolation Nodes

## 3.5 High-Dimensional Model Representation

## 3.6 Cluster Expansion

## 3.7 Functional Approximation Based on Random Processes

## 4 Functional Differential Equations

### 4.1 Variational Form of PDEs

### 4.2 Schwinger-Dyson Equations

### 4.3 Hopf Characteristic Functional Equations

### 4.4 Probability Density Functional Equations

### 4.5 Effective Fokker-Planck Systems

### 4.6 Conjugate Flow Action Functionals

### 4.7 Large Deviation Theory and Minimum Action Methods

## 5 Approximation of Functional Differential Equations

### 5.1 The Method of Weighted Functional Residuals

#### 5.1.1 Functional Collocation Methods

#### 5.1.2 Functional Least Squares

### 5.2 Temporal Discretization

#### 5.2.1 Second-order Adams-Bashforth (AB2) method

#### 5.2.2 Crank-Nicolson Method

### 5.3 Functional Approximation

### 5.4 CP-ALS Algorithm for FDEs with Implicit Time Stepping

#### 5.4.1 Collocation Setting

### 5.5 Tensor Formats for FDEs with Explicit Time Stepping

## 6 Numerical Results: Functionals

### 6.1 Linear functionals

#### 6.1.1 Polynomial Functional Interpolation

#### 6.1.2 Canonical Tensor Decomposition

### 6.2 Quadratic Functionals

#### 6.2.1 Polynomial Functional Interpolation

#### 6.2.2 Canonical Tensor Decomposition

#### 6.2.3 Hierarchical Tucker Expansion

### 6.3 Hopf Characteristic Functionals

#### 6.3.1 Effective Dimension

#### 6.3.2 Polynomial Functional Interpolation

#### 6.3.3 Canonical Tensor Decomposition

#### 6.3.4 Functional Derivatives

### 6.4 Sine Functional

#### 6.4.1 Functional Derivatives

## 7 Numerical Results: Functional Differential Equations

### 7.1 Advection-Reaction Functional Differential Equation

#### 7.1.1 Analytical Solution

### 7.2 Numerical Discretization

#### 7.2.1 HT and CP Algorithms with Explicit Time Stepping
1 Introduction

In this report we address a rather neglected but very important research area in computational mathematics, namely the numerical approximation of nonlinear functionals and functional differential equations (FDEs). FDEs are arise naturally in many different areas of mathematical physics. For example, in the context of fluid dynamics, the Hopf equation \[ \frac{\partial \Phi(\theta, t)}{\partial t} = \sum_{k=1}^{3} \int_V \theta_k(x) \left( i \sum_{j=1}^{3} \frac{\partial}{\partial x_j} \frac{\delta^2 \Phi(\theta, t)}{\delta \theta_j(\mathbf{x})} + \nu \nabla^2 \frac{\delta \Phi(\theta, t)}{\delta \theta_k(\mathbf{x})} \right) d\mathbf{x}, \tag{1} \]
was deemed by Monin and Yaglom ([145], Ch. 10) to be “the most compact formulation of the general turbulence problem”, which is the problem of determining the statistical properties of the velocity and the pressure fields of Navier-Stokes equations given statistical information on the initial state. In equation (1) \( V \subset \mathbb{R}^3 \) is a periodic box, \( \theta(x) = (\theta_1(x), \theta_2(x), \theta_3(x)) \) is a vector-valued test function in a suitable divergence-free space, and \( \Phi \) is a nonlinear complex-valued functional known as Hopf functional [89]. Remarkably, with such functional available it is possible to compute any statistical property of the velocity field that solves the Navier-Stokes equations (see [145]). This is of great conceptual importance: the solution to one single linear functional differential equation can describe all statistical features of turbulence and there is no need to refer back to the Navier-Stokes equations. From a mathematical viewpoint the Hopf functional is basically a time-dependent nonlinear operator in the space of test functions \( D(\Phi) \) (domain of the operator \( \Phi \)) with range in the complex plane (see Figure 1). The operator can be formally defined as a functional integral \[ \Phi(\theta, t) = \int_\Omega \exp \left[ i \int_V u(x, t; \omega) \cdot \theta(x) dx \right] P([u_0]) D[u_0] \] \[ \tag{2} \]
Stanišić [209] refers to the Hopf equation (1) as the “only exact formulation in the entire field of turbulence” (Ch. 12, p. 233).
Figure 1: Sketch of the mapping at the basis of the Hopf functional. The domain of the functional $\Phi$, denoted as $D(\Phi)$, is a suitable space of functions while the range of $\Phi$, denoted as $R(\Phi)$, is a subset of the complex plane $\mathbb{C}$. The approximation space $D_m$ is a subset of $D(\Phi)$, which is mapped into $R_m$.

where $u(x, t; \omega)$ is a stochastic solution to the Navier-Stokes equations and $P([u_0])$ is the probability functional of the random initial state (assuming it exists). Thus, computing the solution to the Hopf equation (1) is equivalent to compute a (complex-valued) time-dependent nonlinear operator $\Phi$ from an equation that involves classical partial derivatives with respect to space and time variables as well as derivatives with respect to functions, i.e., functional derivatives $\delta/\delta \theta_j(x)$ [220, 154].

Another example of functional differential equation is the Schwinger-Dyson equation of quantum field theory [49, 252]. Such equation describes the exact dynamics of the Green functions of a general field theory, and it allows us to propagate field interactions, either in a perturbation setting [165] (weak coupling regime) or in a strong coupling regime [212]. The Schwinger-Dyson formalism is also useful in computing the statistical properties of stochastic dynamical systems. For example, consider Langevin equation

$$\frac{d\psi(t)}{dt} = G(\psi(t), t) + f(t; \omega), \quad (3)$$

where $f(t; \omega)$ is random noise. Define the generating functional [174, 96]

$$Z([\xi, \eta]) = Z_0 \int D[\psi] D[\beta] A([\psi, \beta]) \exp \left[ \int_0^t d\tau \left( \xi(\tau) \cdot \psi(\tau) + \eta(\tau) \cdot \beta(\tau) \right) \right], \quad (4)$$

where $Z_0$ is a normalization constant and

$$A([\psi, \beta]) = C([\beta]) \exp \left[ -\frac{1}{2} \int_0^t d\tau \nabla \cdot G(\psi(\tau), \tau) - i \int_0^t d\tau \beta(\tau) \cdot \left( \frac{d\psi(\tau)}{dt} - G(\psi(\tau), \tau) \right) \right]. \quad (5)$$

The functional $C([\beta])$ in (5) denotes the (known) characteristic functional of the external random noise $f(t; \omega)$. Clearly, if we have available the stochastic solution to (3), then we can construct the functional $Z([\xi, \eta])$ and compute all statistical properties we are interested in. On the other hand, it is straightforward
to show that \( Z(\xi, \eta) \) satisfies the following system of linear FDEs\(^2\) (Schwinger-Dyson equations)

\[
\frac{1}{i} \frac{\partial}{\partial \tau} \frac{\delta Z}{\delta \xi_k(\tau)} = \eta_k(\tau) Z + G_k \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)}, \tau \right) Z - i D_k \left( \frac{1}{i} \frac{\delta}{\delta \eta(\tau)} \right) Z,
\]

(8)

\[
\frac{1}{i} \frac{\partial}{\partial \tau} \frac{\delta Z}{\delta \eta_k(\tau)} = -\xi_k(\tau) Z + i \sum_{j=1}^{n} \frac{\delta}{\delta \eta_j(\tau^+)} \frac{\partial G_k}{\partial \psi_j} \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)}, \tau \right) Z,
\]

(9)

where

\[
D_k(\beta, \tau) = \frac{\delta C(\beta)}{\delta \beta_k(\tau)}.
\]

(10)

The solution to the Schwinger-Dyson equations (8)-(9) is a nonlinear functional (i.e., a nonlinear operator) \( Z(\xi, \eta) \) which allows us to compute all statistical properties of the system without any knowledge of the stochastic process \( \psi(t; \omega) \) defined implicitly by the stochastic ODE (3). By generalizing (4), it is possible to derive a functional formalism for any classical field theory or stochastic system. This yields, in particular, Schwinger-Dyson-type equations for generating functionals associated with the solution to stochastic partial differential equations (SPDEs). If the SPDE admits an action functional, then the construction of the generating functional as well as the derivation of the corresponding Schwinger-Dyson equation are rather straightforward (see [96, 5, 109]).

The usage of functional differential equations grew very rapidly during the sixties, when it became clear that techniques developed for quantum field theory by Dyson, Feynman, and Schwinger could be applied, at least formally, to other branches of mathematical physics. The seminal work of Martin, Siggia, and Rose [136] became a landmark on this subject, since it revealed the possibility of applying (at least formally) quantum field theoretic methods, such as functional integrals and diagrammatic expansions [174, 96, 173, 97], to classical physics. Relevant applications of these techniques can be found in non-equilibrium statistical mechanics [96, 174, 173, 97, 58, 117, 217, 218], stochastic dynamics [87, 228, 112], and turbulence theory [67, 139, 59, 73, 46, 124, 144, 145, 193, 194, 197, 148, 90].

An open question that has persisted over the years is: How do we compute the solution to a functional differential equation? From the fifties to the eighties, researchers were of course investigating analytical methods, e.g., based on functional power series [193, 194, 231, 155], functional integrals [109, 127, 177, 51], transforms with respect to appropriate measures ([145], p. 802), and diagrammatic expansions. More recently, Waclawczyc and Oberlack [162, 232] proposed a Lie group analysis and applied it to the Hopf-Burgers equation, which represents a step forward toward developing new analytical solution methods. Specifically, invariant solutions of the Hopf-Burgers equation were found based on the analysis of the infinitesimal generator of suitable symmetry transformations. From a numerical viewpoint, recent advances in computational mathematics – in particular in numerical tensor methods [78] – open the possibility to solve functional differential equations on a computer. In this report, we will present state-of-the-art mathematical techniques and numerical algorithms to represent nonlinear functionals and compute the numerical solution to functional differential equations.

If FDEs are so important, why do they not have a prominent role in computational mathematics? There are several possible answers to this question. First of all, FDEs are infinite-dimensional equations, in the

\[^2\] The expression

\[
G_k \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)}, \tau \right)
\]

in equations (8) and (9) has to be interpreted in the sense of symbolic operators. For example, in one dimension, if \( G(z, t) = z + z^2 \) then

\[
G \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)}, \tau \right) Z = \frac{1}{i} \frac{\delta Z}{\delta \xi(\tau)} - \frac{\delta^2 Z}{\delta \xi(\tau)^2}.
\]
sense that they are, in principle, equivalent to an infinite-dimensional system of PDEs, or PDEs in an infinite number of variables. This may have understandably discouraged researchers in numerical analysis to even attempt a numerical discretization. Most schemes proposed so far are based on truncations of infinite hierarchies of PDEs obtained, e.g., from functional power series expansions [193, 194, 197, 196, 145, 2, 57], or Lundgren-Monin-Novikov hierarchies [233, 66, 130, 90, 198]. Other approaches are based on a direct discretization of the functional integral [109, 51, 177, 117] that defines the field theory (e.g., $Z$ in equation (4)), and its evaluation using Monte Carlo methods, or source Galerkin methods [120, 119]. Dealing with systems of infinitely many PDEs or very high-dimensional PDEs can indeed be discouraging, but nowadays it is quite common, for example when discretizing stochastic systems driven by colored random noise or stochastic partial differential equations (SPDEs) [250, 242, 234, 244, 230]. Another reason why FDEs have not yet been numerically studied extensively may be due to a lack of awareness of their existence within the computational mathematics community. Also, there is no universal agreement across scientific disciplines as to even the basic definition of an FDE. For example, most applied mathematics literature refers to FDEs as ordinary differential equations with memory or delay terms [221, 7, 80]. In the pure mathematics community, functional equations have been studied in the context of approximate homeomorphisms (the Ulam stability problem) [186, 187], or more generally within problems where the unknown is a function, e.g., Cauchy or d’Alembert functional equations [201]. The physics literature, on the other hand, clearly identifies FDEs as those equations whose unknown is a functional (i.e., a nonlinear operator) and that involve partial derivatives with respect to independent variables (e.g., space and time), as well as derivatives with respect to functions (functional derivatives). These kinds of equations are usually far more challenging than the functional equations studied by the pure mathematics community, and indeed there are very few general theorems on the existence and the uniqueness of their solution [80, 63].

In this report we take the physicist viewpoint and consider linear functional differential equations in the form

$$\frac{\partial F(\theta,t)}{\partial t} = L(\theta,t)F(\theta,t) + H(\theta,t), \quad F(\theta,0) = F_0(\theta),$$

(11)

where $F(\theta,t)$ is a real or complex-valued functional (time-dependent nonlinear operator in a space of functions), $F_0(\theta)$ is a given initial condition, $L(\theta,t)$ is a linear operator in the space of nonlinear functionals, and $H(\theta,t)$ is a known forcing functional. The linear operator $L(\theta,t)$ usually involves functional derivatives with respect to $\theta(x)$ as well as partial derivatives with respect to independent variables, e.g., space and time coordinates. For example, $L(\theta,t)$ could be the linear operator defining the right hand side of equation (1). We emphasize that the class of equations in the form (11) is very broad as it encompasses FDEs describing many physical systems, including statistical properties of nonlinear SODEs and SPDEs (e.g., Hopf characteristic functional equations [89, 148, 145, 112] or equations for probability density functionals [73, 15, 46]), functional equations arising in control theory [14], generalized principles of least actions [225], and functional equations of quantum field theory [252, 95, 174, 97, 96, 117].

To the best of our knowledge, no effective numerical methods have yet been developed to compute the solution to linear functional differential equations in the form (11), and little has been done for functional differential equations in general, despite their fundamental importance in many areas of mathematical physics. The purpose of this report is to fill this gap and present state-of-the-art mathematical techniques, including new classes of numerical algorithms, to approximate nonlinear functionals and the numerical solution to functional differential equations in the form (11). This report is organized in two parts:

1. **Approximation of Nonlinear Functionals** A nonlinear functional is a particular type of nonlinear operator from a space of functions into a vector space, e.g., $\mathbb{R}$ or $\mathbb{C}$. Therefore, the process of approximating a nonlinear functional is basically the same as approximating a nonlinear operator [92, 216, 14]. In this report we will present various techniques for nonlinear functional approximation, ranging from polynomial functional series expansions, to expansions based on stochastic processes, and functional
tensor methods. Within the context of polynomial functional series expansions we will discuss in particular Lagrange interpolation in Hilbert and Banach spaces [134, 183, 181, 17, 4, 28], where the interpolation “nodes” are functions in a suitable function space. We will also discuss series expansions based on functional tensor methods [78]. This class of methods relies on recent developments on multivariate function approximation such as canonical polyadic (CP) [188, 18, 113] and hierarchical Tucker (HT) [8, 77] series expansions.

2. Approximation of Functional Differential Equations

A functional differential equation is an equation whose solution is a nonlinear functional, i.e., a nonlinear operator. The equation usually involves functional derivatives of such functional and partial derivatives and integrals with respect to independent variables. The goal of approximation theory for functional differential equations is therefore to determine an approximation of such nonlinear functional, e.g., its time evolution given an initial state. In this report we will discuss new classes of methods that extend classical Galerkin, least-squares, and collocation techniques to functional differential equations. These methods are based on suitable representations of the solution functional, e.g., in terms of polynomial functionals or functional tensor networks.

This paper is organized as follows: In Section 2 we briefly review what nonlinear functionals, functional derivatives and provide useful examples of nonlinear functionals in physics. In Section 3 we address the approximation of nonlinear functionals and functional derivatives. In particular, we discuss Lagrange interpolation in spaces of infinite dimensions (function spaces), series expansions in terms of polynomial functionals, and functional tensor methods such as canonical polyadic and hierarchical Tucker expansions. In Section 4 we discuss functional differential equations. We begin by presenting several examples of FDEs and show how they arise in the context of well-known physical theories. In Section 5 we address the problem of computing the numerical solution to an FDE. Specifically, we introduce infinite-dimensional extensions of least squares, Galerkin, and collocation methods. Finally, in Section 6 and Section 7 we present numerical results on nonlinear functional approximation and also compute the numerical solution to a prototype functional advection-reaction problem.

2 Nonlinear Functionals

Let $X$ be a Banach space of functions. A nonlinear functional on $X$ is a nonlinear operator $F$ that takes in an element $\theta$ of $X$ (i.e., a function), and returns a real or a complex number. The functional $F$ usually does not operate on the entire linear space $X$ but rather on a subset set of $X$, which we denote as $D(F) \subseteq X$ (domain of the functional). Let us first provide simple examples of nonlinear functionals.

**Example 1:** Consider

$$F([\theta]) = \int_0^1 x^3 e^{-\theta(x)} \, dx, \quad \theta \in D(F) = C^{(0)}([0, 1]),$$

where $C^{(0)}([0, 1])$ is the space of continuous functions in $[0, 1]$.

**Example 2** (homogeneous polynomial functional of order $n$): Consider

$$P_n([\theta]) = \int_0^1 \cdots \int_0^1 K_n(x_1, \ldots, x_n) \theta(x_1) \cdots \theta(x_n) \, dx_1 \cdots dx_n,$$

$n$ times
where $K_n$ are given kernel functions.

Example 3 (Ginzburg-Landau energy functional): The Ginzburg-Landau theory describes phase transitions and critical phenomena in a great variety of statistical systems ranging from magnetic systems, to diluted polymers and superconductors [110, 5]. At the basis of the theory is the energy functional

$$E(\phi) = \int_{\mathbb{R}^3} \left[ \frac{1}{2} \left| \nabla \phi(x) \right|^2 + m^2 \phi^2(x) \right] \, dx + \lambda \frac{\phi^4(x)}{4!} \right] \, dx$$

(14)

where $m$ is the “mass” of the field $\phi$ and $\lambda$ is a coupling constant. In this case, the domain of the functional $E$ can be chosen as $D(E) = C(2)(\mathbb{R}^3)$.

Example 4 (Hopf characteristic functional of a Gaussian random field): Consider a scalar Gaussian random field $u(x; \omega)$ defined on a domain $V \subseteq \mathbb{R}^3$ it can be shown (see, e.g., [112]) that the Hopf characteristic functional of such random field is

$$\Phi(\theta) = \exp \left[ i \int_V \mu(x) \theta(x) \, dx - \frac{1}{2} \int_V \int_V C(x, y) \theta(x) \theta(y) \, dx \, dy \right],$$

(15)

where $\mu(x)$ and $C(x, y)$ denote, respectively, the mean and the covariance function of the field $u(x; \omega)$.

Analysis of nonlinear functionals in Banach spaces is a well-developed subject [220, 154, 203]. In particular, the classical definition of continuity and differentiability at a point that holds for real-valued functions can extended in a more or less straightforward way to nonlinear functionals. For instance, we say that a functional $F([\theta])$ is continuous at a point $\theta(x) \in D(F)$ if for any sequence of functions $\{\theta_1(x), \theta_2(x), \ldots\}$ in $D(F)$ converging to $\theta$ we have that the sequence $\{F([\theta_n])\}$ converges to $\{F([\theta])\}$, i.e.,

$$\lim_{n \to \infty} |\theta_n(x) - \theta(x)| \to 0 \quad \Rightarrow \quad \lim_{n \to \infty} |F([\theta_n]) - F([\theta])| \to 0.$$

(16)

The functionals we discussed in Examples 1-4 are all continuous. From the continuity definition (16) it follows, in particular, that if $F([\theta])$ is continuous on a compact function space $D(F)$ then $F$ is bounded.

Example 4: Another example of a continuous functional in $D(F) = C(\infty)([-1, 1])$ is

$$F([\theta]) = \theta(0)^2 + \int_{-1}^{1} \sin(\theta(x)) \, dx.$$

(17)

In fact, consider any sequence of functions $\{\theta_n\}$ in $D(F)$. Also choose an integrable function $g(x)$ such that $|\theta_n(x)| \leq g(x)$ for all $n \in \mathbb{N}$. In these conditions, the Lebesgue dominated convergence theorem applies and we have

$$\lim_{n \to \infty} \int_{-1}^{1} \sin(\theta_n(x)) \, dx = \int_{-1}^{1} \sin(\theta(x)) \, dx,$$

(18)

i.e.,

$$\theta_n \to \theta \quad \Rightarrow \quad F([\theta_n]) \to F([\theta]).$$

(19)
2.1 Functional Derivatives

Consider a real or a complex valued functional \( F \) defined on the function space \( D(F) \) (domain of the functional). For simplicity, let us assume that \( D(F) \) is a space of real valued functions \( \theta(x) \) on the real line. We say that the functional \( F \) is differentiable at \( \theta(x) \) if the limit

\[
\lim_{\epsilon \to 0} \frac{F([\theta(x) + \epsilon \eta(x)]) - F([\theta(x)])}{\epsilon}
\]

exists and it is finite. The quantity (20) is known as \( \text{Gâteaux differential} \) of \( F \) in the direction of \( \eta(x) \) (see, e.g., [220, 223]). Under rather general assumptions such derivative can be represented as a linear operator \([154, 220, 225]\) acting on \( \eta(x) \). For small \( \epsilon \) we have

\[
F([\theta(x) + \epsilon \eta(x)]) = F([\theta(x)]) + \epsilon L([\theta(x)]) \eta(x) + R_1(\epsilon \eta(x); \theta(x)),
\]

(21)

In this series \( L([\theta(x)]) \) is a linear operator that sends the function \( \eta(x) \) to a real or a complex number, while \( R_1 \) represents a reminder term. It is clear that \( L([\theta(x)]) \) involves integration with respect to \( x \), since \( L([\theta]) \eta(x) \) is a real or complex number. Thus, we look for a representation of \( L([\theta]) \eta(x) \) in the form

\[
L([\theta(x)]) \eta(x) = \int \frac{\delta F([\theta])}{\delta \theta(x)} \eta(x) dx,
\]

(22)

where the kernel function \( \delta F([\theta])/\delta \theta(x) \) is a functional of \( \theta \) and a function of \( x \). At this point it is convenient to establish a parallel between functionals and functions in \( m \) variables. Recall that the differential of a scalar field \( f(a_1, \ldots, a_m) \) in the direction \( \vec{n} = (n_1, \ldots, n_m) \) is the scalar product of the gradient \( \nabla f \) and \( \vec{n} \), i.e.,

\[
df_{\vec{n}} = \nabla f \cdot \vec{n}.
\]

(23)

By analogy, \( \delta F([\theta])/\delta \theta(x) \) in equation (22) can be considered as an infinite-dimensional gradient, known as \textit{first-order functional derivative} of \( F \) with respect to \( \theta(x) \).

The next question is: how do we compute such functional derivative? A possible way is to use the definition (20) and a compactly supported class of test functions, e.g., functions that are nonzero only in a small neighborhood \( I_x(r) \) of radius \( r \) centered at \( x \). Such functions could be compactly supported elements of a Dirac delta sequence (see Figure 2), or even a delta function itself. This allows us to write

\[
\frac{\delta F([\theta(x)])}{\delta \theta(x)} = \lim_{\epsilon, r \to 0} \frac{F([\theta(x) + \epsilon \alpha(x)]) - F([\theta(x)])}{\epsilon \int_{I_x(r)} \alpha(y) dy}
\]

(24)

In particular, if we set \( \alpha(y) = \delta(x - y) \) then the denominator in (24) simply reduces to \( \epsilon \), yielding the formula

\[
\frac{\delta F([\theta])}{\delta \theta(x)} = \lim_{\epsilon \to 0} \frac{F([\theta(y) + \epsilon \delta(x - y)]) - F([\theta(y)])}{\epsilon}.
\]

(25)

Functional derivatives of higher order can be defined in a similar manner. For example, the second order functional derivative of \( F \) is

\[
\frac{\delta^2 F([\theta])}{\delta \theta(x) \delta \theta(y)} = \frac{d}{d\epsilon} \left[ \frac{\delta F([\theta(z) + \epsilon \delta(z - y)])}{\delta \theta(x)} \right]_{\epsilon=0}.
\]

(26)

Note that (26) is a function and \( x \) and \( y \) and a functional of \( \theta(x) \).
Figure 2: A possible function for the calculation of the kernel (24) at \(x_k\). If we set \(\mathcal{I}_{x_k}(r) = \{x \in \mathbb{R} : |x - x_k| \leq r\}\), then we have \(\int_{\mathcal{I}_{x_k}(r)} \alpha(x) dx = 1\), independently on \(r\).

Example 1: The first-order functional derivative of the nonlinear functional (12) can be obtained as follows. We first compute the Gâteaux differential

\[
\frac{d}{d\epsilon} F([\theta + \epsilon \eta]) \bigg|_{\epsilon=0} = - \int_0^1 x^3 e^{-\theta(x)} \eta(x) dx = \int_0^1 \frac{\delta F([\theta])}{\delta \theta(x)} \eta(x) dx.
\]

Therefore,

\[
\frac{\delta F([\theta])}{\delta \theta(x)} = -x^3 e^{-\theta(x)}.
\]

Note that the functional derivative is a function of \(x\) and a (local) functional of \(\theta(x)\).

Example 2 (Functional Derivatives of the Hopf Functional): Consider the Hopf characteristic functional of a random function \(u(x; \omega)\) defined in \([0, 1]\)

\[
\Phi([\theta]) = \left\langle \exp \left[ i \int_0^1 u(x; \omega) \theta(x) dx \right] \right\rangle.
\]

The average operator \(\langle \cdot \rangle\) here denotes a functional integral over the probability functional of \(u(x, \omega)\). The Gâteaux differential of \(\Phi([\theta])\) along \(\eta\) is

\[
\frac{d}{d\epsilon} \Phi([\theta(x) + \epsilon \eta(x)]) \bigg|_{\epsilon=0} = i \left\langle \exp \left[ i \int_0^1 u(x; \omega) \theta(x) dx \right] \int_0^1 u(x; \omega) \eta(x) dx \right\rangle
\]

\[
= \int_0^1 i \left\langle \exp \left[ i \int_0^1 u(x; \omega) \theta(x) dx \right] u(x; \omega) \right\rangle \eta(x) dx
\]

\[
= \int_0^1 \frac{\delta \Phi([\theta])}{\delta \theta(x)} \eta(x) dx.
\]

This implies that

\[
\frac{\delta \Phi([\theta])}{\delta \theta(x)} = i \left\langle u(x; \omega) \exp \left[ i \int_a^b u(x; \omega) \theta(x) dx \right] \right\rangle
\]

is the first-order functional derivative of \(\Phi([\theta])\) at \(\theta(x)\). Note that (31) is itself a functional of \(\theta(x)\), which depends also on \(x\). As a result, \(\delta \Phi([\theta]) / \delta \theta(x)\) has two types of derivatives: an ordinary one with respect
to $x$, and a functional one with respect to $\theta(x)$. The latter is the the second-order functional derivative of
$\Phi([\theta(x)])$. A simple calculation shows that
\[
\frac{\delta^2 \Phi([\theta])}{\delta \theta(x) \delta \theta(y)} = i^2 \left< u(y;\omega) u(x;\omega) \exp \left[ i \int_a^b u(x;\omega) \theta(x) dx \right] \right>.
\] (32)
Proceeding similarly, we can obtain the expression of higher-order functional derivatives. For instance, the
third-order one is explicitly given as
\[
\frac{\delta^3 \Phi([\theta])}{\delta \theta(x) \delta \theta(y) \delta \theta(z)} = i^3 \left< \exp \left[ i \int_a^b u(x;\omega) \theta(x) dx \right] u(y;\omega) u(x;\omega) u(z;\omega) \right>.
\] (33)
Now, suppose we have available $\delta \Phi([\theta]) / \delta \theta(x)$. Based on the definition (31), we see that
\[
\left< u(x;\omega) \right> = \frac{1}{i} \frac{\delta \Phi([0])}{\delta \theta(x)}. \tag{34}
\]
Similarly, higher order moments and cumulants of the random function $u(x;\omega)$ can be obtained by computing
higher order functional derivatives of $\Phi([\theta(x)])$ and $\ln \Phi([\theta(x)])$, respectively, and evaluating them at
$\theta(x) = 0$. In particular, the second- and third-order correlation functions are, respectively
\[
\left< u(x;\omega) u(y;\omega) \right> = \frac{1}{i^2} \frac{\delta^2 \Phi([0])}{\delta \theta(x) \delta \theta(y)}, \quad \left< u(x;\omega) u(y;\omega) u(z;\omega) \right> = \frac{1}{i^3} \frac{\delta^3 \Phi([0])}{\delta \theta(x) \delta \theta(y) \delta \theta(z)}. \tag{35}
\]
Example 3: The Gateaux differential of the nonlinear functional $F([\theta + \epsilon \eta])$ in the direction $\eta(x)$ is
\[
\frac{d}{d\epsilon} F([\theta + \epsilon \eta]) \bigg|_{\epsilon=0} = \frac{d}{d\epsilon} \left( \theta(0) + \epsilon \eta(0) + \int_{-1}^1 \sin(\theta(x) + \epsilon \eta(x)) dx \right) \bigg|_{\epsilon=0}
\]
\[
= \int_{-1}^1 (\delta(x) + \cos(\theta(x))) \eta(x) dx \tag{36}
\]
Therefore the first-order functional derivative is
\[
\frac{\delta F([\theta])}{\delta \theta(x)} = \delta(x) + \cos(\theta(x)), \tag{37}
\]
where $\delta(x)$ at the right hand side is the Dirac delta function [98].

**Regularity of Functional Derivatives** The last example clearly shows that functional derivatives of nonlinear
functionals can easily be distributions [93], e.g., Dirac delta functions. For example, let $h \in C^{(\infty)}(\mathbb{R})$.
Then any functional in the form $F([\theta]) = h((\theta, \theta))$, where $(,)$ is an inner product in $C^{(\infty)}(\mathbb{R})$, has a singular
second-order functional derivative. In fact,
\[
\frac{\delta F([\theta])}{\delta \theta(x)} = 2 \left. \frac{\partial h}{\partial a} \right|_{a=(\theta, \theta)} \theta(x) \tag{38}
\]
\[
\frac{\delta^2 F([\theta])}{\delta \theta(x) \delta \theta(y)} = 2\delta(x - y) \left. \frac{\partial h}{\partial a} \right|_{a=(\theta, \theta)} + 4 \left. \frac{\partial^2 h}{\partial a^2} \right|_{a=(\theta, \theta)} \theta(x)\theta(y). \tag{39}
\]
The characteristic functional of zero-mean Gaussian white noise
\[
\Phi([\theta]) = e^{-\langle \theta, \theta \rangle / 2} \tag{40}
\]
belongs to this class, i.e., it has a “singular” second-order functional derivative. On the other hand, the functional
\[ G(\theta) = h((K, \theta)), \]
where \( K(x) \) is a given smooth kernel, has smooth functional derivatives
\[ \frac{\delta G(\theta)}{\delta \theta(x)} = \frac{\partial h}{\partial a}\bigg|_{a=(K,\theta)} K(x), \quad \frac{\delta^2 G(\theta)}{\delta \theta(x) \delta \theta(y)} = \frac{\partial^2 h}{\partial a^2}\bigg|_{a=(K,\theta)} K(x)K(y). \]

3 Approximation of Nonlinear Functionals

Approximation theory for nonlinear functionals is strongly related to approximation theory of nonlinear operators \([92, 216, 14]\). A nonlinear functional \( F \) is in fact a particular type of nonlinear operator from a space of functions \( D(F) \) (the domain of the functional \( F \)) into a vector space, e.g., \( \mathbb{R} \) or \( \mathbb{C} \). Thus, the problem of approximating nonlinear functionals is basically the same as approximating nonlinear operators. This topic has been studied extensively by different scientific communities (see, e.g., \([216, 199, 155, 205, 70, 183, 17, 134, 104]\)) for obvious reasons. What does it mean to approximate a nonlinear functional? Consider, as an example, the functional (12), hereafter rewritten for convenience
\[ F(\theta) = \int_0^1 x^3 e^{-\theta(x)} dx, \quad D(F) = C(0)([0,1]). \]

Approximating \( F(\theta) \) in this case means that we are aiming at constructing a nonlinear operator \( \hat{F}(\theta) \) that allows us to compute an approximation of all possible integrals in the form (43), for arbitrary continuous functions \( \theta \in C(0)([0,1]) \). This challenging problem includes cases in which \( F(\theta) \) admits an analytical solution, e.g., \( F(x) \) or \( F(\sin(x)) \), as well as cases where no analytical solution is available, e.g., \( F(x^2) \).

Perhaps, the most classical and widely used approach to represent nonlinear functionals relies on functional power series\(^3\). The method was originally developed by Volterra \([231]\), and it represents the counterpart of power series expansions in the theory of functions. In practice, the functional of interest is represented in terms of a series of integral operators involving increasing powers of the test function and kernels that need to be determined. The canonical form of the power series expansion is
\[ F(\theta) = \sum_{k=0}^{\infty} P_k(\theta), \quad \text{where} \quad P_k(\theta) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} K_k(x_1, \ldots, x_k) \theta(x_1) \cdots \theta(x_k) dx_1 \cdots dx_k. \]

**Remark:** Functional power series are known to have bad approximation properties and other issues. For example, they often do not preserve important properties of the functional, e.g., positive definiteness or normalization in the case of Hopf functionals.

3.1 Functional Approximation in Finite-Dimensional Function Spaces

The simplest ways to establish a closed functional representation is to restrict the domain of the functional to a finite-dimensional function space spanned by the basis \( \{\varphi_1(x), \ldots, \varphi_m(x)\} \), i.e.,
\[ D_m = \text{span}\{\varphi_1(x), \ldots, \varphi_m(x)\}. \]

\(^3\)Functional power series have been widely used in the turbulence theory to obtain moment and cumulant expansions (see \([67, 143, 194]\)).
In this way, any element in $D_m$ can be represented as

$$\theta_m(x) = \sum_{k=1}^{m} a_k \varphi_k(x). \quad (47)$$

Possible choices of $\varphi_k(x)$ are:

1. **Lagrange Characteristic Polynomials.** Given a set of $m$ distinct interpolation nodes $\{x_j\}$ in the interval $[a, b]$, for example Gauss-Chebyshev-Lobatto nodes, we set

$$\varphi_j(x) = \prod_{i=1, i \neq j}^{m} \frac{x - x_i}{x_i - x_j}. \quad (48)$$

2. **Jacobi Polynomials.** The function space $D_m \subseteq D(F)$ can be also represented by a finite set of Jacobi polynomials $J_j^{(\alpha, \beta)}(x)$ (see [72, 84]), i.e.

$$\varphi_j(x) = J_j^{(\alpha, \beta)}(x). \quad (49)$$

As is well known, Jacobi polynomials include many other families of widely used polynomials such as Gegenbauer, Legendre and Chebyshev.

3. **Trigonometric Polynomials.** If $D(F)$ is the space of periodic functions in $[0, 2\pi]$, then a convenient choice for $\varphi_k(x)$ may be the set of (nodal) trigonometric polynomials [84]

$$\varphi_j(x) = \frac{1}{m} \sin \left( \frac{m}{2} \frac{x - x_j}{2} \right) \cot \left( \frac{x - x_j}{2} \right), \quad x_j = \frac{2\pi}{m}, j = 0, \ldots, m \quad (50)$$

or, equivalently, classical Fourier modes

$$\varphi_0(x) = 1, \quad \varphi_k(x) = \sin(kx), \quad \varphi_{m+k}(x) = \cos(kx), \quad k = 1, \ldots, m. \quad (51)$$

For each specific choice of the basis set $\{\varphi_1(x), \ldots, \varphi_m(x)\}$, the test function [47] lies on a parametric manifold of dimension $m$, i.e., a hyperplane. Any discretization of the function space $D(F)$ in terms of a finite-dimensional basis, reduces the functional $F$ into a multivariate function with domain $D_m$ and range $F([D_m])$. Such function depends on as many variables as the number of degrees of freedom we consider in the finite-dimensional approximation of $D(F)$.

**Example 1:** A substitution of (47) into the Hopf functional (29) yields the complex-valued multivariate function

$$\phi(a_1, \ldots, a_m) = \left\langle \exp \left( i \sum_{k=1}^{m} a_j U_j(\omega) \right) \right\rangle, \quad U_j(\omega) = \int_{0}^{1} u(x; \omega) \varphi_j(x) dx, \quad (52)$$

---

4 If $D(F)$ is a space of multivariate functions defined on some subset of $V \subseteq \mathbb{R}^d$ then (47) takes the form

$$\theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x), \quad x \in V. \quad (46)$$

More generally, $\theta(x)$ can be represented by series expansions based on tensor products, or more advanced expansions that rely on HDMR [185, 125] or tensor methods [18, 113] (see also Section 3.5 and Section 3.3). The latter techniques are recommended when operating on test function spaces defined on high-dimensional domains $V$. 

13
i.e., the joint characteristic function of the Fourier coefficients \( U_j(\omega) \). Note that \( \phi \) depends on \( m \) real variables \((a_1, \ldots, a_m)\). Such multivariate function can be seen as a \( m \)-dimensional parametrization of the mapping \( \Phi \) shown in Figure 1, i.e., a parametrization of the nonlinear transformation \( D_m \rightarrow \Phi(D_m) \). In this setting, approximation of nonlinear functionals is equivalent to approximation of a real- or complex-valued multivariate functions. The question of whether a functional can be approximated by evaluating it in a space spanned by a finite-dimensional basis is different, and will be addressed in Section 3.1.2.

**Example 2:** Consider the nonlinear functional (12) and let \( D_m \subset D(F) \) be the function space spanned by a suitable set of orthogonal polynomials in \([0, 1]\). Evaluating \( F(\theta) \) in \( D_m \), i.e., considering test functions \( \theta_m \) in the form (47), yields the multivariate function

\[
f(a_1, \ldots, a_m) = \int_0^1 x^3 \prod_{k=1}^m e^{-a_k \varphi_k(x)} dx.
\]

(53)

This is the exact form of the functional \( F \), evaluate in \( D_m \), i.e.,

\[
F(\theta_m) = f(a_1, \ldots, a_m).
\]

(54)

### 3.1.1 Functional Derivatives

Evaluating a nonlinear functional \( F(\theta) \) in a finite-dimensional function space \( D_m \) allows for a simple and effective representation of functional derivatives. In particular, it can be shown that

\[
\frac{\delta F(\theta)}{\delta \theta(x)} \bigg|_{\theta \in D_m} = \sum_{k=1}^m \varphi_k(x) \frac{\partial f}{\partial a_k},
\]

(55)

\[
\frac{\delta^2 F(\theta)}{\delta \theta(x) \delta \theta(y)} \bigg|_{\theta \in D_m} = \sum_{j,k=1}^m \varphi_k(x) \varphi_j(y) \frac{\partial f}{\partial a_j \partial a_k}.
\]

(56)

Here, \( f(a_1, \ldots, a_m) = F(\theta_m) \) is the function we obtain by evaluating the functional \( F \) in the finite dimensional function space \( D_m \). The meaning of (55) and (56) is the following: if we evaluate the functional derivatives of \( F \) in the finite dimensional space \( D_m \) (recall that the functional derivatives are themselves nonlinear functionals) then we can represent them in terms of classical partial derivatives of \( f(a_1, \ldots, a_m) \).

Note that the basis function \( \varphi_j \) spanning \( D_m \) also appear in (55)-(56), suggesting that the accuracy of the functional derivatives depend on the choice of such basis functions. Rather than proving (55) and (56) in a general setting, let we provide two constructive examples that yield expressions in the form (55) and (56).

**Example 1:** Consider the Hopf functional (29). By evaluating the analytical expression of the first- and second-order functional derivatives (31)-(32) in the finite-dimensional function space (45) we obtain

\[
\delta \Phi(\theta) \bigg|_{\theta \in D_m} = i \left\langle u(x) e^{i(a_1 U_1(\omega) + \cdots + a_m U_m(\omega))} \right\rangle,
\]

(57)

\[
\frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} \bigg|_{\theta \in D_m} = - \left\langle u(x) u(y) e^{i(a_1 U_1(\omega) + \cdots + a_m U_m(\omega))} \right\rangle,
\]

(58)

---

Given any positive measure in a one-dimensional interval, it is always possible to construct a set of polynomials that is orthogonal with respect to such measure [71, 72].
where $U_j(\omega)$ are random variables defined in (52). By using the definition of the characteristic function (52) we have that

$$\frac{\partial \phi}{\partial a_k} = \int_0^1 \varphi_k(x) \frac{\delta \Phi(\theta)}{\delta \theta(x)} \bigg|_{\theta \in D_m} dx,$$

(59)

$$\frac{\partial^2 \phi}{\partial a_k \partial a_j} = \int_0^1 \int_0^1 \varphi_k(x) \varphi_j(y) \frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} \bigg|_{\theta \in D_m} dxdy.$$

(60)

This means that the partial derivatives of the characteristic function are nothing but the projection of the Hopf functional derivatives onto the space $D_m$. Clearly, if the random function $u(x; \omega)$ is $D_m$ then the following inverse formulas hold

$$\frac{\delta \Phi(\theta)}{\delta \theta(x)} \bigg|_{\theta, u \in D_m} = \sum_{k=1}^m \varphi_k(x) \frac{\partial \phi}{\partial a_k},$$

(61)

$$\frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} \bigg|_{\theta, u \in D_m} = \sum_{j,k=1}^m \varphi_k(x) \varphi_j(y) \frac{\partial \phi}{\partial a_j \partial a_k}.$$

(62)

**Example 2:** Consider the sine functional

$$F(\theta) = \sin \left( \int_a^b K(x) \theta(x) dx \right)$$

(63)

where $K(x)$ is a given kernel function. Evaluating $F$ in $D_m$ yields the multivariate function

$$f(a_1, ..., a_m) = \sin \left( \sum_{i=1}^m a_i \int_a^b K(x) \varphi_i(x) dx \right).$$

(64)

Similarly, evaluating the functional derivative of $F$ in $D_m$ yields

$$\frac{\delta F(\theta)}{\delta \theta(x)} \bigg|_{\theta \in D_m} = \cos \left( \sum_{i=1}^m a_i \int_a^b K(x) \varphi_i(x) dx \right) K(x).$$

(65)

A comparison between (64) and (65) immediately yields

$$\frac{\partial f}{\partial a_k} = \int_a^b \frac{\delta F(\theta)}{\delta \theta(x)} \bigg|_{\theta \in D_m} \varphi_j(x) dx,$$

(66)

i.e., the gradient of $f$ is the projection of the functional derivative of $F(\theta)$ (evaluated in $D_m$) onto $D_m$. On the other hand, if $K(x)$ is a function in $D_m$ then

$$\frac{\delta F(\theta)}{\delta \theta(x)} \bigg|_{\theta \in D_m} = \sum_{k=1}^m \frac{\partial f}{\partial a_k} \varphi_k(x).$$

(67)

This clarifies the meaning of the functional derivative in both finite- and infinite-dimensional ($m \to \infty$) cases.
3.1.2 Distances between Function Spaces and Approximability of Functionals

A key concept when approximating a nonlinear functional $F(\theta)$ by restricting its domain $D(F)$ to a finite-dimensional space functions $D_m$ is the distance between $D_m$ and $D(F)$. Such distance can be quantified in different ways (see, e.g., [175]). For example we can define the deviation of $D_m$ from $D(F)$ as

$$E(D_m, D(F)) = \sup_{\theta \in D(F)} \inf_{\theta_m \in D_m} \| \theta - \theta_m \|$$  \hspace{1cm} (68)

The number $E$ measure the extent to which the worst element of $D(F)$ can be approximated from $D_m$. One may also ask how well we can approximate $D(F)$ with $m$-dimensional subspaces of $D(F)$ which are allowed to vary within $D(F)$. A measure of such approximation is given by the Kolmogorov $m$-width

$$d_m(D_m, D(F)) = \inf_{D_m} \sup_{\theta \in D(F)} \inf_{\theta_m \in D_m} \| \theta - \theta_m \|_{D(F)}$$  \hspace{1cm} (69)

which quantifies the error of the best approximation to the elements of $D(F)$ by elements in a vector subspace $D_m$ of dimension at most $m$. The Kolmogorov $m$-width can be rigorously defined, e.g., for nonlinear functionals in Hilbert spaces ([175], Ch. 4). In simpler terms we can define the notion of approximability of a nonlinear functional as follows. Let $F(\theta)$ be a continuous nonlinear functional with domain $D(F)$, and consider a finite-dimensional subspace $D_m \subseteq D(F)$, for example $D_m = \text{span}\{\varphi_1, \ldots, \varphi_m\}$. We say that $F(\theta)$ is approximable in $D_m$ if for all $\theta \in D(F)$ and $\epsilon > 0$, there exists $m$ (depending on $\epsilon$) and an element $\theta_m \in D_m$ such that

$$\| F(\theta) - F(\theta_m) \| \leq \epsilon.$$  \hspace{1cm} (70)

Clearly if $F$ is continuous and $\theta_m$ is close to $\theta$, i.e., the deviation (68) between $D(F)$ and $D_m$ is small, then we expect $\epsilon$ to be small. It is important to emphasize that the approximation error and the computational complexity of approximating a nonlinear functional depends on the choice of $D_m$. In particular, a functional may be low-dimensional in one function space and high-dimensional in another. The following example clarifies this question.

Example 1: Consider the sine functional

$$F(\theta) = \sin \left( \int_0^{2\pi} \theta(x) dx \right)$$  \hspace{1cm} (71)

in the space $D(F)$ of periodic functions in $[0, 2\pi]$. If we represent $\theta$ in terms of orthonormal Fourier modes, i.e., we consider

$$D_{2m+1} = \text{span}\left\{ \frac{1}{\sqrt{2\pi}}, \frac{\sin(x)}{\sqrt{\pi}}, \ldots, \frac{\sin(mx)}{\sqrt{\pi}}, \frac{\cos(x)}{\sqrt{\pi}}, \ldots, \frac{\cos(mx)}{\sqrt{\pi}} \right\}$$  \hspace{1cm} (72)

and

$$\theta_{2m+1}(x) = \frac{a_0}{\sqrt{2\pi}} + \frac{1}{\sqrt{\pi}} \sum_{j=1}^{m} a_j \sin(jx) + \frac{1}{\sqrt{\pi}} \sum_{j=1}^{m} b_j \cos(jx)$$  \hspace{1cm} (73)

In this setting, we obtain

$$F([\theta_{2m+1}]) = \sin(\sqrt{2\pi}a_0).$$  \hspace{1cm} (74)

This means that (71) is approximable in the function space (72). Moreover, the approximation is exact and just one-dimensional. On the other hand, if we set the space $D_{2m+1}$ to be the span of a normalized nodal
Fourier basis \{\varphi_0, ..., \varphi_{2m}\}, e.g., the normalized odd expansion discussed in [84], then the functional (71) technically requires an infinite number of variables. In fact, in this case we have
\[
F([\theta_{2m+1}]) = \sin \left( \eta \sum_{k=0}^{2m} a_k \right), \quad \text{where} \quad \eta = \int_0^{2\pi} \varphi_k(x)dx = \left( \frac{2\pi}{2m+1} \right)^{\frac{1}{2}}. \tag{75}
\]

**Example 2:** Consider the characteristic functional of zero-mean Gaussian white noise (see equation (15)),
\[
\Phi([\theta]) = \exp \left[ -\frac{1}{2} \int_{-\pi}^{\pi} \theta(x)^2 dx \right], \tag{76}
\]
where \(D(\Phi)\) is the space of periodic functions in \([0, 2\pi]\). Let \(D_m\) be the space spanned by any finite orthonormal set of periodic functions. The deviation between \(D(\Phi)\) and \(D_m\) this case yields a functional approximation error of order 1. To show this in a simple way, evaluate the functional (76) in both \(D(\Phi)\) and \(D_m\). This yields
\[
\Phi([\theta]) = \exp \left[ -\frac{1}{2} \sum_{k=1}^{\infty} a_k^2 \right], \quad \Phi([\theta_m]) = \exp \left[ -\frac{1}{2} \sum_{k=1}^{m} a_k^2 \right]. \tag{77}
\]
If we measure the error between \(\Phi([\theta])\) and \(\Phi([\theta_m])\) in the uniform operator norm then we have
\[
\|\Phi([\theta_m]) - \Phi([\theta])\|_{\infty} = 1, \tag{78}
\]
independently on \(m\). In other words, (76) is not approximable in any finite-dimensional subset of \(D(\Phi)\). This result is consistent with white-noise theory [211]. Recall, in fact, that a delta-correlated Gaussian process has a flat Fourier power spectrum. This implies that any finite truncation of the Fourier series of such process yields a systematic error that is not small.

**Remark:** In some cases the effects of the distance between \(D(F)\) and \(D_m\) can be mitigated by the presence of smooth functions appearing the in functional \(F\). For example, consider the sine functional
\[
F([\theta]) = \exp \left( \int_{-1}^{1} \sin(x)\theta(x)dx \right) \tag{79}
\]
and let \(D(F)\) be the space of infinitely differentiable functions in \([-1, 1]\). If we expand \(\theta\) in terms of Legendre polynomials \{\varphi_k\}, i.e.,
\[
\theta_m(x) = \sum_{k=0}^{m} a_k \varphi_k(x) \tag{80}
\]
then,
\[
\int_{-1}^{1} \sin(x)\theta(x)dx = \sum_{k=0}^{m} a_k \int_{-1}^{1} \sin(x)\varphi_k(x)dx. \tag{81}
\]
As is well known, the coefficients \(\int_{-1}^{1} \sin(x)\varphi_k(x)dx\) decay to zero exponentially fast with \(m\) [84]. This implies that convergence of \(F([\theta_m])\) to \(F([\theta])\) is exponentially fast in the number of dimensions \(m\), that is the error (70) goes to zero exponentially fast with \(m\).

**Example 3:** Consider the Hopf characteristic functional of a zero-mean correlated Gaussian process in \([0, 2\pi]\)
\[
\Phi([\theta]) = \exp \left[ -\frac{1}{2} \int_{0}^{2\pi} \int_{0}^{2\pi} C(x, y)\theta(x)\theta(y)dxdy \right]. \tag{82}
\]
where \( C(x, y) \) is a smooth covariance function, and \( D(\Phi) \) is the space of periodic functions in \([0, 2\pi]\). If the projection of \( C(x, y) \) onto the span of an orthonormal set \( D_m \subseteq D(F) \) decays with \( m \), then the functional \( \Phi \) is approximable in \( D_m \). Note that this is indeed the case if the covariance is smooth (and periodic) and \( \varphi_j(x) \) are the Fourier modes in \([72]\). The smoother the covariance the smaller the number of Fourier modes we need to achieve a certain accuracy \([84]\), i.e., the smaller the number of dimensions. If \( C(x, y) = 1 \) then the Hopf functional \((82)\) is effectively one-dimensional.

### 3.2 Functional Interpolation Methods

In this Section we discuss how to construct an approximation of a nonlinear functional \( F([\theta]) \) in terms of a functional interpolant \( \Pi([\theta]) \), i.e., a functional that interpolates \( F([\theta]) \) at at a given set of nodes \( \{\theta_1(x), ..., \theta_m(x)\} \subseteq D(F) \)

\[
F([\theta_j]) = \Pi([\theta_j]) \quad j = 1, ..., m. \tag{83}
\]

Differently from interpolation methods in spaces of finite dimension (e.g., \( d \)-dimensional Euclidean spaces), interpolation here is in a space of functions, i.e., the interpolation nodes \( \theta_k(x) \) are functions in a Hilbert or a Banach space. Over the years, the problem of constructing a functional interpolant through suitable nodes in Hilbert or Banach spaces has been studied by several authors and convergence results were established in rather general cases \([134, 103, 183, 178, 99, 4, 106, 104, 179, 216]\).

Before discussing functional interpolation in detail, let us provide some geometric intuition on what functional interpolation is and what kind of representations we should expect. To this end, let us first recall that a hyperplane in a \( d \)-dimensional space is a linear manifold defined uniquely by \( d \) interpolation nodes, each node being a vector of \( \mathbb{R}^d \). If we send \( d \) to infinity then the hyperplane intuitively becomes a linear functional, which is defined uniquely by an infinite number of \( \infty \)-dimensional nodes, i.e., an infinite number of functions. This suggests that if we consider any finite number of nodes in a function space, say \( \{\theta_1(x), ..., \theta_m(x)\} \), then we cannot even represent linear functionals in an exact way\(^6\) i.e., functionals in the form

\[
P_i([\theta]) = \int_a^b K_1(x_1)\theta(x_1)dx_1, \tag{84}
\]

where \( K_1(x) \) is a given kernel. The same conclusion obviously holds for nonlinear functionals, with the aggravating factor that the number of test functions theoretically required for the exact representation grows significantly. For example, quadratic and cubic forms in \( d \)-dimensions are identified by \( d^2 \) and \( d^3 \) interpolation nodes, respectively, where each node is vector of \( \mathbb{R}^d \). When we send \( d \) to infinity, we intuitively obtain homogeneous polynomial functionals of second- and third-order, respectively. These functionals are in the form

\[
P_2([\theta]) = \int_a^b \int_a^b K_2(x_1, x_2)\theta(x_1)\theta(x_2)dx_1dx_2, \tag{85}
\]

\[
P_3([\theta]) = \int_a^b \int_a^b \int_a^b K_3(x_1, x_2, x_3)\theta(x_1)\theta(x_2)\theta(x_3)dx_1dx_2dx_3. \tag{86}
\]

Thus, to represent \( P_2 \) and \( P_3 \) exactly we need \( \infty^2 \) and \( \infty^3 \) nodes in a function space. Why \( \infty^2 \) and \( \infty^3 \)?

Consider \( P_2([\theta]) \) and assume that the kernel function \( K_2(x_1, x_2) \) is in a separable Hilbert space. Represent \( K_2 \) relative to any complete orthonormal basis \( \{\varphi_k\}_{k=1, ..., \infty} \)

\[
K_2(x_1, x_2) = \sum_{i,j=1}^{\infty} a_{ij}\varphi_i(x_1)\varphi_j(x_2). \tag{87}
\]

\(^{6}\) We recall that the variational form of nonlinear PDEs is defined by linear functionals on test function spaces. In this setting, classical Galerkin methods to solve PDEs (see Section 4.1) are basically identification problems for linear functionals.
A substitution of (87) into (85) yields

$$P_2(\theta) = \sum_{i,j=1}^{\infty} a_{ij} \int_a^b \varphi_i(x) \theta(x) dx \int_a^b \varphi_j(x) \theta(x) dx. \quad (88)$$

Without loss of generality we can assume that $K_2$ is symmetric, i.e., that $a_{ij}$ is a symmetric matrix. To represent $P_2(\theta)$ exactly by means of a functional interpolant we need enough nodes $\{\theta_p(x)\}$ to determine each $a_{ij}$ in (87) uniquely. Clearly, the choice $\theta_p(x) = \varphi_p(x)$ ($p = 1, \ldots, \infty$) is not sufficient for this purpose, since it allows us to determine only the diagonal entries $a_{pp}$. Therefore we need to construct a larger set of collocation nodes, e.g., the set $\theta_p(x) = \varphi_p(x) + \varphi_{p_j}(x)$, where $p_i \geq p_j$ and $p_i \in \{1, \ldots, \infty\}$. This is the $\infty^2$ number of functions we have mentioned above. Similarly, to identify $P_3(\theta)$ through functional interpolation we need $\infty^3$ nodes, e.g., in the form $\theta_p(x) = \varphi_p(x) + \varphi_{p_j}(x) + \varphi_{p_k}(x)$, where $p_i \geq p_j \geq p_k$ and $p_k \in \{1, \ldots, \infty\}$. As we shall see later in this Section, determining a polynomial interpolant of an unknown functional $F(\theta)$, i.e., determining the kernels $K_j(x_1, \ldots, x_j)$ in (44) from input-output relations is a linear problem that involves high-dimensional systems and big data.

### 3.2.1 Interpolation Nodes in Function Spaces

Let $F(\theta)$ be a continuous functional with domain $D(F)$. Within $D(F)$ we define the spaces of functions

$$S_q^{(m)} = \{ \theta(x) \in D(F) \mid \theta(x) = a_{i_1} \varphi_{i_1}(x) + \cdots + a_{i_q} \varphi_{i_q}(x) \}.$$

(89)

where $i_j \in \{1, \ldots, m\}$, $a_{i_j} \in \mathbb{R}$ and $\{\varphi_1(x), \ldots, \varphi_m(x)\} \in D(F)$. The elements of $S_1^{(m)}$, $S_2^{(m)}$ and $S_3^{(m)}$ are in the form

- $S_1^{(m)}$: $\theta(x) = a_{i_1} \varphi_{i_1}(x),$
- $S_2^{(m)}$: $\theta(x) = a_{i_1} \varphi_{i_1}(x) + a_{i_2} \varphi_{i_2}(x),$
- $S_3^{(m)}$: $\theta(x) = a_{i_1} \varphi_{i_1}(x) + a_{i_2} \varphi_{i_2}(x) + a_{i_3} \varphi_{i_3}(x).$

Clearly, if $\theta_1 \in S_1^{(m)}$ and $\theta_2 \in S_q^{(m)}$, then $(\theta_1 + \theta_2) \in S_{j+q}^{(m)}$ (if $q < m$). Also, note that the sequence of spaces $S_j^{(m)}$, $j = 1, 2, \ldots$ is hierarchical in the sense that the following chain of embeddings hold

$$S_1^{(m)} \subset S_2^{(m)} \subset \cdots \subset S_{m}^{(m)} \subset D(F). \quad (90)$$

The function space $S_q^{(m)}$ admits a simple yet powerful graphical representation in terms of trajectories of functions $[133, 213, 214, 225]$. To illustrate such representation, consider a complex-valued functional $\Phi$. A trajectory of functions in the space $D(\Phi)$ is a curve in $D(\Phi)$, which is mapped to a curve in $\mathbb{C}$ (see Figure 3). Furthermore, if the functional is continuous and differentiable, a smooth curve in $D(F)$ is mapped onto a smooth curve in $\mathbb{C}$. The set of trajectories in the complex plane associated with $S_1^{(m)}$ is shown in Figure 4. Each curve is parametrized by only one parameter $a_j$ and it cannot branch into two distinct curves. On the contrary, if we consider $S_2^{(m)}$, we are adding one more degree of freedom and each curve departing from 1 can branch, but only once. Similarly, we can have three branches in $S_3^{(m)}$, etc. A remarkable distribution of nodes in $S_2^{(m)}$ is associated with networks of test functions, i.e., graphs in the function space $D(F)$. Vertex and edges are elements of $S_2^{(m)}$. A simple example is shown in Figure 3. In mathematics such network is called complete graph, i.e., an undirected graph in which every pair of distinct nodes is connected by a unique edge – the edge being the trajectory (straight line) of functions connecting $\theta_i$ to $\theta_j$. If we discretize each edge with $p$ collocation points (including the endpoints) and we have $m$ nodes then the number of degrees of freedom is $m(m-1)(p-2)/2 + m$. 

19
Figure 3: Network of test functions: A closed path in the function space $D(\Phi)$ made of four lines (e.g., the red polygon) is mapped through the functional $\Phi$ into a closed curve in the complex plane. Vertex and edges of the network are elements of $S^{(m)}_2$. We also show a complete graph with $n = 5$ nodes, where each edge is discretized with $p = 4$ points.

Figure 4: Trajectories in the range of a complex valued functional $\Phi$ corresponding to functions in the sets $S^{(m)}_1$ (left) and $S^{(m)}_3$ (right). In the case of $S^{(m)}_1$ each curve is parametrized by only one parameter $a_j$ and it cannot branch into two distinct curves. On the contrary, if we consider $S^{(m)}_3$ we are adding two more degrees of freedom and each curve can branch at most twice.

Another interesting set of interpolation nodes in $D(F)$ is the one obtained by setting all coefficients $a_{i\nu}$ in (89) equal to 1, i.e.,

$$\tilde{S}^{(m)}_q = \{ \theta_i(x) \in D(F) \mid \theta_i(x) = \varphi_{i_1}(x) + \cdots + \varphi_{i_q}(x) \}, \quad i_q = 0, ..., m, \quad (91)$$

If the set $\{\varphi_j\}$ includes the null element $\{0\}$, then by symmetry (91) is equivalent to

$$\tilde{S}^{(m)}_q = \{ \{0\}, \{\varphi_1, ..., \varphi_m\}, \{2\varphi_1, ..., 2\varphi_m\}, \{(\varphi_1 + \varphi_2), ..., (\varphi_1 + \varphi_m)\}, \{(\varphi_2 + \varphi_3), ..., (\varphi_2 + \varphi_m)\}, ... \}. \quad (92)$$
In this case, the number of elements (cardinality) of $\tilde{S}_q^{(m)}$ is

$$\#\tilde{S}_q^{(m)} = \sum_{j=0}^{q} \binom{j + m - 1}{j}, \quad \text{where } \binom{i}{j} \text{ is the binomial coefficient.} \quad (93)$$

For example,

$$\#\tilde{S}_1^{(10)} = 11, \quad \#\tilde{S}_2^{(10)} = 66, \quad \#\tilde{S}_3^{(10)} = 286, \quad \#\tilde{S}_5^{(10)} = 3003, \quad \#\tilde{S}_{10}^{(10)} = 184756. \quad (94)$$

In addition, the cardinality of $\tilde{S}_q^{(m)}$ satisfies the recursion relation

$$\#\tilde{S}_q^{(m)} = \#\tilde{S}_{q-1}^{(m)} + \binom{q + m - 1}{q}. \quad (95)$$

The set of functions $\tilde{S}_q^{(m)}$ is sufficient to uniquely identify a polynomial functional of order $q$ in which each kernel function is represented relative to tensor product basis with $m$ elements in each variable. However, $\tilde{S}_q^{(m)}$ is, in general, not sufficient to accurately interpolate nonlinear functionals. The main problem is that the set of nodes (91) may not be large enough or may not cover the function space $D(F)$ appropriately. Another open question is related to the selection of optimal interpolation nodes in $D(F)$ yielding highly accurate representations. This question is addressed in Section 3.4.1.

In a finite-dimensional setting, we can sample the coefficients $a_{ij}$ in (89), e.g., at sparse grids locations [11][24] that is at unions of appropriate tensorizations of one-dimensional point sets such as Gauss-Hermite, Clenshaw-Curtis, Chebyshev or Leja [151]. This yields the set

$$\tilde{S}_q^{(m)} = \left\{ \theta_i(x) \in D(F) \mid \theta_i(x) = a_{i_1} \varphi_{i_1}(x) + \cdots + a_{i_q} \varphi_{i_q}(x) \right\}, \quad i_s = 1, \ldots, m, \quad (96)$$

where the vector $(a_{i_1}, \ldots, a_{i_q})$ takes discrete values at sparse grid nodes. As an example, in Figure 5 we plot three Clenshaw-Curtis grids and few samples of the corresponding interpolation nodes in $\tilde{S}_2^{(2)}$. For illustration purposes, the basis elements $\varphi_1(x)$ and $\varphi_2(x)$ here are chosen as

$$\varphi_1(x) = \sin(x)e^{\cos(x)}, \quad \varphi_2(x) = \sin(2x)e^{\cos(2x)}. \quad (97)$$

The construction of sparse grids usually follows the Smolyak algorithm. Other dimension-adaptive schemes and greedy Leja rules were recently proposed in [32][151][153].

### 3.2.2 Polynomial Interpolation of Nonlinear Functionals

Let $F : D(F) \to Y$ ($Y = \mathbb{R}$ or $\mathbb{C}$) be a nonlinear real- or complex-valued functional on $D(F)$. Consider the set of polynomial functionals of degree $n$

$$\Pi_n(\theta) = L_0 + L_1(\theta) + L_2(\theta, \theta) + \cdots + L_n(\theta, \ldots, \theta), \quad (98)$$

where $L_0 \in Y$, and $L_k : D(F)^k \to Y$ are $k$-linear symmetric functionals

$$L_k(\theta_1, \ldots, \theta_p) = \int \cdots \int K_k(x_1, \ldots, x_k)\theta_1(x_1)\cdots\theta_k(x_k)dx_1 \cdots dx_k, \quad k = 1, \ldots, n. \quad (99)$$

A comparison between equations (98) and (44) yields

$$P_k(\theta) = L_k(\theta, \ldots, \theta), \quad (100)$$
Figure 5: Clenshaw-Curtis point sets and corresponding nodes in $\theta_i(x) = a_1 \varphi_1(x) + a_2 \varphi_2(x)$ ($i = 1, 2, 3$) in the test function space $S_2^3$. Specifically, we plot $\theta_i(x)$ corresponding to the nodes 1, 2 and 3 marked in red in the top right Figure. The basis elements $\varphi_k(x)$ here are chosen as $\sin(kx) \exp(\cos(kx))$ ($k = 1, 2$).

and therefore we can equivalently write (98) as

$$\Pi_n([\theta]) = \sum_{k=1}^{n} P_k([\theta]).$$  

(101)

The symmetry assumption on $L_k$ implies that $K_k(x_1, ..., x_k)$ are symmetric kernels, i.e., any permutation of $x_1, ..., x_n$ leaves $K_k$ unchanged. It is obviously possible to define polynomial functionals with non-symmetric kernels. However, such functionals can be always written in a symmetric form by rearranging the kernel functions appropriately. For example, let $H_2(x_1, x_2)$ be non-symmetric. It is easy to verify that

$$\int_a^b \int_a^b H_2(x_1, x_2)\theta(x_1)\theta(x_2)dx_1dx_2 = \int_a^b \int_a^b K_2(x_1, x_2)\theta(x_1)\theta(x_2)dx_1dx_2, \quad \forall \theta \in D(F),$$  

(102)

where

$$K_2(x_1, x_2) = \frac{1}{2}(H_2(x_1, x_2) + H_2(x_2, x_1)).$$  

(103)

In other words, the value of the integral does not change if we replace $H_2(x_1, x_2)$ with its symmetrized version $K_2(x_1, x_2)$. More generally, we can symmetrize any kernel $H_p(x_1, ..., x_p)$ by summing up all terms corresponding to all possible permutations of $(x_1, ..., x_p)$ and then dividing up by the factorial of $p$. For example,

$$K_3(x_1, x_2, x_3) = \frac{1}{3!}(H_3(x_1, x_2, x_3) + H_3(x_1, x_3, x_2) + H_3(x_2, x_1, x_3) + H_3(x_2, x_3, x_1) + H_3(x_3, x_1, x_2) + H_3(x_3, x_2, x_1)).$$  

(104)

The symmetry of the operators $L_k$ significantly reduces the number of collocation nodes in the function space $D(F)$ needed to identify kernels $K_k$, provided these are of finite-rank.
The polynomial functional interpolation problem can be stated as follows: Given a set of \( m \) nodes \( \{\theta_1(x), \ldots, \theta_m(x)\} \) in \( D(F) \), find a polynomial functional in the form (98) satisfying the interpolation conditions

\[
\Pi_n([\theta_i]) = F([\theta_i]), \quad i = 1, \ldots, m. \tag{105}
\]

**The Stone-Weierstrass Approximation Theorem**  
The possibility of approximating an arbitrary continuous functional in Hilbert or Banach spaces in terms polynomial functionals is justified by theorems analogous to the classical Weierstrass theorem for continuous functions. We recall that such theorem states that if \( f(x) \) is a continuous, real-valued function on the closed interval \([a, b]\), then given any \( \varepsilon > 0 \) there exists a real polynomial \( p(x) \) such that \( |f(x) - p(x)| < \varepsilon \) for all \( x \in [a, b] \). A remarkable generalization of this result has its roots in the Stone-Weierstrass theorem [210], which can be stated as follows: suppose that \( X \) is a compact metric space and \( K \) is an algebra of continuous, real-valued functions on \( X \) that separates points\(^7\). Then for any continuous, real-valued functional \( F \) on \( X \) and for any \( \varepsilon > 0 \) there exists a polynomial functional \( \Pi \in K \) such that \( \|F([u]) - \Pi([u])\| < \varepsilon \) for all \( u \in X \). The first paper dealing with this subject is due Fréchet [65]. He showed that any continuous functional can be represented by a series of polynomial functionals whose convergence is uniform in all compact sets of continuous functions. This result was generalized to compact sets of functions in Hilbert and Banach spaces by Prenter [182] and Istratescu [94], respectively. Other relevant work in this area is [70, 180, 17, 28, 170, 178].

### 3.2.3 Porter Interpolants

An effective way to construct finite-order polynomial functionals with minimal norm interpolating arbitrary continuous functionals in Hilbert spaces was proposed by W. Porter in [179]. The key idea relies on minimizing the norm of (98) subject to the interpolation conditions (105). A natural way to impose such conditions is through Lagrange multipliers. This yields the variational principle

\[
\min_{K_1, \ldots, K_n} \|\Pi_n\|^2 + \sum_{j=1}^m \lambda_i \left( \Pi_n([\theta_i]) - F([\theta_i]) \right), \tag{106}
\]

The minimum is relative to arbitrary variations of the kernel functions \( K_j(x_1, \ldots, x_j) \). Also, \( \|\Pi_n\|^2 \) is the norm of the polynomial functional (98), which is defined as

\[
\|\Pi_n\|^2 = \sum_{p=0}^n \int \cdots \int |K_p(x_1, \ldots, x_p)|^2 \, dx_1 \cdots dx_p < \infty \tag{107}
\]

The solution to the variational principle (106) allows us to identify the kernel functions \( K_j(x_1, \ldots, x_j) \) and, correspondingly, the polynomial functional with minimal norm interpolating \( F([\theta]) \) at the \( m \) nodes \( \{\theta_1, \ldots, \theta_m\} \). Specifically, we obtain

\[
K_p(x_1, \ldots, x_p) = \sum_{j,k=1}^m \pi_j^{(p)}(x_1, \ldots, x_p) H_{jk}^{-1} F([\theta_k]), \quad i = 0, \ldots, n. \tag{108}
\]

In this equation,

\[
\pi_j^{(0)} = 1, \quad \pi_j^{(p)}(x_1, \ldots, x_p) = \theta_j(x_1) \cdots \theta_j(x_p) \quad p = 1, 2, \ldots \tag{109}
\]

\(^7\)The algebra \( K \) separates points if for any two distinct elements \( u_1, u_2 \in X \) there exists \( \Pi_n \in K \) such that \( \Pi_n([u_1]) - \Pi_n([u_2]) \neq 0 \) of \( X \) and that contains the constant function.
while the (symmetric) matrix $H_{ij}$ is defined as

$$H_{ij} = 1 + (\theta_i, \theta_j) + (\theta_i, \theta_j)^2 + \cdots + (\theta_i, \theta_j)^n,$$

(110)

where $(\cdot, \cdot)$ denotes the $L_2(V)$ inner product, $V$ being the domain of the interpolation nodes $\theta_j(x)$. The polynomial functional $\Pi_n$ constructed in this way exists if $F([\theta_i])$ is in the range of the matrix $H$ (see [179] for further details). If one wants to approximate $F([\theta])$ in terms of a superimposition of monomials with orders defined by an index set $I$ then

$$H_{ij} = \sum_{p \in I} (\theta_i, \theta_j)^p$$

(111)

The total degree of the polynomial functional is the largest number in the index set $I$. It is convenient to write Porter’s interpolant in terms of basis functionals $g_i([\theta])$ as

$$\Pi_n ([\theta]) = \sum_{k=1}^m F ([\theta_k]) g_k([\theta]),$$

(112)

where

$$g_k([\theta]) = \sum_{j=1}^m H_{jk}^{-1} \sum_{p \in I} (\theta_j, \theta)^p.$$  

(113)

The functional interpolant (112)-(113) has the following properties:

1. $\{g_k([\theta])\}$ is a set of cardinal basis functionals, i.e., $g_k([\theta_q]) = \delta_{kq}$. This implies that Porter’s interpolant is a cardinal Lagrangian interpolant.

2. If the interpolation nodes $\{\theta_1, ..., \theta_m\}$ are orthonormal with respect to the inner product $(\cdot, \cdot)$ then $H_{ii} = \#I$ (cardinality of the index set $I$) and $H_{ij} (i \neq j)$ either equal to one or zero, depending on whether we have $\{0\}$ in the set $I$ or not. In every case, $H$ is a matrix with diagonal entries equal to $\#I$ and off-diagonal entries equal to either zero or one. Such matrix is always invertible provided $I$ does not reduce to the single element $\{0\}$.

3. Porter’s interpolant is degenerate for $I = \{0\}$ as the matrix $H$ is rank one and therefore it is not invertible. The Moore-Penrose pseudoinverse $H^+$, however, exists and it provides the correct form of the interpolant. To show this in a simple case, consider the constant functional $F([\theta]) = c \in \mathbb{R}$ and the zero-order polynomial interpolant at $\{\theta_1, ..., \theta_m\}$

$$\Pi_0([\theta]) = K_0 = \sum_{k,j=1}^m H_{jk}^{-1} F([\theta_j]). \quad H_{ij} = 1.$$  

(114)

Clearly, $H_{jk}^{-1}$ does not exist since $H_{ij} = 1$ and therefore $\text{rank}(H) = 1$. However, the Moore-Penrose pseudoinverse of $H$ has components $H_{ij}^+ = 1/m^2$, and therefore $\Pi_0([\theta]) = c = F([\theta])$.

4. The polynomial functional (112)-(113) is an interpolant if and only if the matrix (111) is invertible, i.e., full rank. This depends on both the choice of interpolation nodes and on the index set $I$. The Moore-Penrose pseudoinverse $H^+$, in general, does not allow to satisfy the interpolation condition. Indeed, by evaluating (112) at the interpolation nodes $\theta_k$ we obtain

$$\Pi_n ([\theta_i]) = \sum_{k=1}^m F ([\theta_k]) \sum_{j=1}^m H_{jk}^{-1} H_{ji} = F([\theta_i]).$$

(115)

However, if we replace $H_{jk}^{-1}$ with $H_{jk}^+$ then we get $\Pi_n ([\theta_i]) \neq F([\theta_i])$, since $H^+ H \neq I$. 

24
5. Consider the approximation of constant functionals \( F(\theta) = c \) by polynomial functionals of order one, i.e., \( \Pi_1(\theta) = L_0 + L_1(\theta) \). Assume that the interpolation nodes \( \{\theta_k\}_{k=1}^m \) are orthogonal functions with norm that decreases with \( m \) as \( 1/m \) (see Section 6.1). Let \( H_{ij} = 1 + h\delta_{ij} \) where \( h = \|\theta_j\|^2 \) be the matrix (111) corresponding to the index set \( I = \{0, 1\} \). By using the identity

\[
\lim_{h \to 0} \frac{1}{h} \left( 1 + h\delta_{ij} \right)^{-1} = \delta_{ij} - \frac{1}{m}
\]

we obtain

\[
\Pi_1(\theta) \approx c \sum_{k,j=1}^m \left( h\delta_{jk} - \frac{h}{m} \right) \left( 1 + (\theta, \theta_j) \right) \quad \text{as} \quad h \to 0 \quad \text{and} \quad m \to \infty.
\]

This means that \( \lim_{m \to \infty} \Pi_1(\theta) = c \), i.e., the polynomial interpolant is consistent with the functional \( F \) in the sense that the linear term becomes smaller and smaller as we increase the number of test functions \( \theta_k \). In the limit \( m \to \infty \) (infinite number of test functions) we see the linear term is absent, and we correctly recover the constant functional.

By extending these arguments to higher-order polynomial functionals in Hilbert spaces, one can show that Porter’s interpolants of order \( n \) converge pointwise to entire functionals or any polynomial functional of order \( n \) or less as the number of interpolation nodes \( \theta_k \) goes to infinity (see [105] and Theorem 1 in [103]).

**Functional Derivatives** The functional derivatives of Porter interpolants can be easily determined by computing the functional derivatives of the basis functionals \( g_i(\theta) \) defined in (113). To this end, we first notice that

\[
\frac{\delta (\theta, \theta_k)^p}{\delta \theta(x)} = (\theta, \theta_k)^{p-1} p \theta_k(x), \quad p \geq 1.
\]

A substitution of this formula into (113) yields

\[
\frac{\delta g_k(\theta)}{\delta \theta(x)} = \sum_{j=1}^m \theta_j(x) H_{jk}^{-1} \sum_{p \in \mathcal{I}} (\theta_j, \theta)^{p-1} p
\]

By evaluating \( g_k(\theta) \) at the nodes \( \theta_j(x) \) we obtain a functional generalization of the classical differentiation matrix [84]

\[
D_{ji}^{(1)}(x) = \frac{\delta g_i(\theta_j)}{\delta \theta(x)}.
\]

Similarly, the second-order functional derivative of \( g_k(\theta) \) is

\[
\frac{\delta^2 g_k(\theta)}{\delta \theta(x) \delta \theta(y)} = \sum_{j=1}^m \theta_j(x) \theta_j(y) H_{jk}^{-1} \sum_{p \in \mathcal{I}} p(p-1) (\theta_j, \theta)^{p-2},
\]

and it yields the following second-order functional differentiation matrix

\[
D_{ji}^{(2)}(x,y) = \frac{\delta^2 g_i(\theta_j)}{\delta \theta(x) \delta \theta(y)}.
\]

At this point it is useful to provide simple examples of functional interpolation in Hilbert spaces. Example
Figure 6: Structure of the $H$-matrix (111) associated with the function set $\hat{S}_1^{(10)}$, $\hat{S}_2^{(10)}$ and $\hat{S}_3^{(10)}$ defined in Eq. (127). We consider $I = 1$, $I = 2$ and $I = 3$, respectively.

1: Consider the first-order polynomial functional

$$P_1([\theta]) = \int_a^b K_1(x)\theta(x)dx.$$  \hspace{1cm} (123)

To represent $P_1([\theta])$ in terms of a functional interpolant it is sufficient to consider the set of orthonormal functions $\hat{S}_1^{(m)} = \{\varphi_1, ..., \varphi_m\}$ (see Eq. (91)). In this case, Porter’s cardinal basis functionals (113) reduce to

$$g_i([\theta]) = (\varphi_i, \theta),$$ \hspace{1cm} (124)

and the functional interpolant can be written as

$$\Pi_1([\theta]) = \sum_{i=1}^m P_1([\varphi_i])g_i([\theta]).$$ \hspace{1cm} (125)

Clearly, we have that $\Pi_1([\theta]) \rightarrow P_1$ as $m \rightarrow \infty$.

Example 2: Consider the second-order polynomial functional

$$P_2([\theta]) = \int_a^b \int_a^b K_2(x,y)\theta(x)\theta(y)dxdy.$$ \hspace{1cm} (126)

To represent $P_2([\theta])$ in terms of a functional interpolant it is sufficient to consider the set of orthonormal functions

$$\hat{S}_2^{(m)} = \{\varphi_1, ..., \varphi_m, (\varphi_1 + \varphi_2), ..., (\varphi_1 + \varphi_m), (\varphi_2 + \varphi_3), ..., (\varphi_2 + \varphi_m), ...\}$$ \hspace{1cm} (127)

which is similar (but not equal) to (91). The matrix (111) associated with this set has the structure shown in Figure 6. The cardinal basis functionals (113) reduce to

$$g_k([\theta]) = \sum_{j=1}^{\#S_2^{(m)}} H_{jk}^{-1}(\theta_j, \theta)^2,$$ \hspace{1cm} (128)
where \( \#\hat{S}_2^{(m)} \) is the number of elements of \( \hat{S}_2^{(m)} \), \( \theta_k = \varphi_k \) \((k=1,\ldots,m)\), \( \theta_{m+1} = \varphi_1 + \varphi_2 \), etc. The functional interpolant can be written as

\[
\Pi_2([\theta]) = \sum_{i=1}^{\#\hat{S}_2^{(m)}} P_2([\theta_i])g_i([\theta]),
\]
(129)

and it converges \( P_2 \) as \( m \to \infty \) (see Section 6).

Example 3: Consider the third-order polynomial functional

\[
P_3([\theta]) = \int_a^b \int_a^b \int_a^b K_3(x_1, x_2, x_3)\theta(x_1)\theta(x_2)\theta(x_3)dx_1dx_2dx_3.
\]
(130)

To represent \( P_3([\theta]) \) in terms of a functional interpolant it is sufficient to consider the set of functions \( \hat{S}_3^{(m)} \subset \hat{S}_3^{(m)} \) defined as

\[
\hat{S}_3^{(m)} = \hat{S}_2^{(m)} \cup \{(\varphi_i + \varphi_j + \varphi_k), \ k > j > i\}
\]
(131)

where \( \hat{S}_2^{(m)} \) is as in (127). The matrix (111) associated with this set has the structure shown in Figure 6. The cardinal basis functionals (113) reduce to

\[
g_k([\theta]) = \sum_{j=1}^{\#\hat{S}_3^{(m)}} H^{-1}_{jk} (\theta_j, \theta)^3.
\]
(132)

The functional interpolant can be written as

\[
\Pi_3([\theta]) = \sum_{i=1}^{\#\hat{S}_3^{(m)}} P_3([\theta_i])g_i([\theta]).
\]
(133)

More-Penrose Pseudoinverse and Non-Cardinal Basis Functionals We emphasized that the matrix \( H_{ij} \) defined in (111) may be not invertible in some cases. This happens, for example, if the interpolation nodes \( \theta_k(x) \) are linearly dependent or if there exist a symmetry such the inner product of \( \theta_k \) and \( \theta_j \) yields linearly dependent rows/columns in (111). In this cases we can still construct a polynomial functional with minimal norm which, however, does not interpolate \( F \) at \( \theta_k \). To this end, we simply use the Moore-Penrose pseudoinverse of \( H_{ij} \) to obtain a representation of Porter’s polynomial functionals in terms of a non-cardinal basis \( g^+_k([\theta]) \) as

\[
\Pi_n([\theta]) = \sum_{k=1}^{m} F([\theta_k])g^+_k([\theta]).
\]
(134)

where

\[
g^+_k([\theta]) = \sum_{j=1}^{m} H^{+}_{jk} \sum_{p \in \mathbb{I}} (\theta_j, \theta)^p.
\]
(135)

In the last equation \( H^{+}_{ij} \) denotes the Moore-Penrose pseudoinverse of \( H_{ij} \). The approximation properties of polynomial functionals in the form (134) will be studied in Section 6.3.2 and Section 6.4.
Recursive Porter Interpolation  The number of interpolation nodes \( \theta_k \) required to represent exactly a polynomial functional of order \( n \) is given in (93). For example, if we set \( m = 10 \) elementary functions and polynomial order \( n = 12 \) such formula yields \( 646646 \) interpolation nodes! Such large number of nodes may be an issue when computing Porter’s interpolants. In fact, computing the inverse of (111) rapidly becomes intractable as we increase either \( m \) or \( n \). To overcome this problem we can split the process of inverting the matrix (111) into a recursive algorithm, e.g., by using Schur complements and blockwise inversion. To this end, consider the set of nodes

\[
\left\{ \{ \theta_1, \ldots, \theta_m \}, \{ \theta_{m+1}, \ldots, \theta_{2m} \}, \{ \theta_{2m+1}, \ldots, \theta_{3m} \}, \ldots, \{ \theta_{(N-1)m+1}, \ldots, \theta_{Nm} \} \right\},
\]

and define the matrices

\[
H_{ij}^{(I,J)} = \sum_{p \in I} (\theta_i, \theta_j)^p \quad i = (I - 1)m + 1, \ldots, Im \quad j = (J - 1)m + 1, \ldots, Jm,
\]

where \( I \) denotes the index set of Porter’s monomials while \( I \) and \( J \) run from 1 to \( N \). The \( H \)-matrix (111) corresponding to the set (136) can be represented in a block-wise form as

\[
H_N = \begin{bmatrix}
H^{(1,1)} & \cdots & H^{(1,N)} \\
\vdots & \ddots & \vdots \\
H^{(N,1)} & \cdots & H^{(N,N)}
\end{bmatrix}.
\]

The computational cost of inverting such matrix is prohibitive if \( Nm \) is large. However, we can use the following recursive procedure. We first build and invert \( H_1 = H^{(1,1)} \), corresponding to the first set of \( m \). This allows us to determine Porter’s interpolant on the first set of \( m \) nodes in (136). Next we add the second set, i.e., the nodes \( \{ \theta_{m+1}, \ldots, \theta_{2m} \} \). The matrix (111) corresponding to the nodal set \( \{ \theta_1, \ldots, \theta_{2m} \} \) is

\[
H_2 = \begin{bmatrix}
H_1 & C_1 \\
C_1^T & H^{(2,2)}
\end{bmatrix}, \quad C_1 = H^{(1,2)}
\]

and it can be inverted by using the block-wise formula (142)

\[
H_2^{-1} = \begin{bmatrix}
H_1^{-1} + B_1 A_1 B_1^T & -B_1 A_1 \\
-A_1 B_1 & A_1
\end{bmatrix}
\]

where

\[
B_1 = H_1^{-1} C_1, \quad A_1 = \left( H^{(2,2)} - C_1^T B_1 \right)^{-1} \quad \text{(inverse of the Schur complement)}.
\]

Now we bring in the third set of nodes \( \{ \theta_{2m+1}, \ldots, \theta_{3m} \} \). The matrix (111) corresponding to the nodal set \( \{ \theta_1, \ldots, \theta_{3m} \} \) is

\[
H_3 = \begin{bmatrix}
H_2 & C_2 \\
C_2^T & H^{(3,3)}
\end{bmatrix}, \quad C_2 = \begin{bmatrix} H^{(1,3)} \\ H^{(2,3)} \end{bmatrix},
\]

and its inverse is, as before,

\[
H_3^{-1} = \begin{bmatrix}
H_2^{-1} + B_2 A_2 B_2^T & -B_2 A_2 \\
-A_2 B_2 & A_2
\end{bmatrix}
\]

where

\[
B_2 = H_2^{-1} C_2, \quad A_2 = \left( H^{(3,3)} - C_2^T B_2 \right)^{-1}.
\]
At this point it is clear that the procedure can be iterated as many times as needed. This generates a sequence of basis functionals, and an interpolating polynomial functional with minimal norm that passes through an increasing number of nodes. In this way, we have reduced the problem of computing Porter’s interpolant through a very large number of nodes into a sequence of matrix inversions of dimension at most $m \times m$.

The storage requirements of the algorithm just described, however, is not small as because Porter’s basis functionals are ultimately defined by $H^{-1}_N$. An open question is the identification of interpolation nodes $\theta_j$ leading to minimal complexity/storage requirements for $H^{-1}_N$.

An alternative interpolation method makes use of residuals. The main idea is sketched in Figure 7. The functional $F([\theta])$ is interpolated by a polynomial of order $n$, denoted as $\Pi_n^{(1)}([\theta])$, at just three nodes $\{\theta_1, \theta_2, \theta_3\}$. Subtracting $\Pi_n^{(1)}([\theta])$ from $F([\theta])$ yields the functional residual

$$R_n^{(1)}([\theta]) = F([\theta]) - \Pi_n^{(1)}([\theta])$$

which is zero at $\theta_1$, $\theta_2$ and $\theta_3$ because of the interpolation condition. Now we add three more nodes $\{\theta_4, \theta_5, \theta_6\}$ and construct a Porter’s interpolant of $R_n^{(1)}([\theta])$ at $\{\theta_1, ..., \theta_6\}$. We denote such interpolant by $\Pi_n^{(2)}([\theta])$. The computation of $\Pi_n^{(2)}([\theta])$ can be carried out as above by using Schur complements and block-wise inversion of $H^{-1}_N$. The polynomial functional $\Pi_n^{(1)}([\theta]) + \Pi_n^{(2)}([\theta])$ interpolates $F([\theta])$ at $\{\theta_1, ..., \theta_6\}$. The recursive construction proceeds with the definition of the new residual

$$R_n^{(2)}([\theta]) = F([\theta]) - \Pi_n^{(1)}([\theta]) - \Pi_n^{(2)}([\theta])$$

three more nodes $\{\theta_7, \theta_8, \theta_9\}$, and a Porter’s polynomial $\Pi_n^{(3)}([\theta])$, interpolating $R_n^{(2)}([\theta])$ at $\{\theta_1, ..., \theta_9\}$. Proceeding recursively with higher-order residuals up to order $r$, we obtain the polynomial functional

$$Q_n([\theta]) = \sum_{k=1}^{r} \Pi_n^{(k)}([\theta]).$$

Clearly $Q_n([\theta])$ interpolates $F([\theta])$ at all nodes $\theta_k$ and therefore it is completely equivalent to $\Pi_n([\theta])$, i.e., $Q_n([\theta]) = \Pi_n([\theta])$.

**Hierarchical Matrices** The algorithm we just described aim at reducing the computational cost of computing polynomial functional interpolants by inverting the matrix $H$ defined in (111) in a block-wise fashion.
or recursively. The structure of such matrix obviously depends on how we select the interpolation nodes in the function space $D(F)$. An interesting open question is whether we can determine sets of nodes for which the interpolation problem can be solved at a minimal cost. Stated in matrix terms, can we identify sets of nodes yielding structured matrices that can be easily inverted, e.g., hierarchical matrices? The matrices shown in Figure 6 have indeed a self-similar structure which can be used to speed up their inversion. We leave this question open for future research. For uniquely solvable interpolation problems we could equivalently construct Khlobystov polynomial functionals (see Section 3.2.5), and determine the coefficients of the expansion by using the method of moments.

3.2.4 Prenter Interpolants

Another method to determine polynomial interpolants in Hilbert and Banach spaces was introduced by Prenter [183]. She proved that if $F([\theta])$ is a functional in a Hilbert space $D(F)$, and $\theta_j \in D(F)$ are interpolation nodes, then there exists a $n$th-order functional interpolant in the form

$$\Pi_n([\theta]) = \sum_{i=1}^{n} F([\theta_i])g_i([\theta]),$$

where

$$g_i([\theta]) = \prod_{j=1, j \neq i}^{n} \frac{(\theta - \theta_j, \theta_i - \theta_j)}{(\theta_i - \theta_j, \theta_i - \theta_j)} \quad \text{(cardinal basis).}$$

As before, $(,)$ denotes the inner product in $L_2(V)$, where $V$ is the domain of $\theta_k(x)$. Note that each basis element $g_i([\theta])$ is a polynomial functional of order $n$. On the other hand, Porter’s method yields polynomial basis functionals of total degree $\max(I)$, where the index set $I$ does not depend on the number of collocation points. Porter [178] applied Prenter’s theorems to causal systems, while Bertuzzi, Gandolfi and Germani [16, 17] extended Prenter’s results to causal approximation of input-output maps in Hilbert spaces. Generalizations to Banach spaces can be found in Chapter 3 of [216] (see also [134] and the references therein). The functional derivatives of Prenter’s polynomial functionals can be obtained by computing the functional derivatives of (150). This yields

$$\frac{\delta g_i([\theta])}{\delta \theta(x)} = \sum_{k=1 \atop k \neq i}^{n} \frac{\theta_i(x) - \theta_k(x)}{\|\theta_i - \theta_k\|^2} g_{ik}([\theta]),$$

$$\frac{\delta^2 g_i([\theta])}{\delta \theta(x) \delta \theta(y)} = \sum_{k=1 \atop k \neq i}^{n} \frac{\theta_i(x) - \theta_k(x)}{\|\theta_i - \theta_k\|^2} \sum_{s=1 \atop s \neq k}^{n} \frac{\theta_i(y) - \theta_s(y)}{\|\theta_i - \theta_s\|^2} g_{iks}([\theta]),$$

where

$$g_{ik}([\theta]) = \prod_{j=1, j \neq i, k}^{n} \frac{(\theta - \theta_j, \theta_i - \theta_j)}{\|\theta_i - \theta_j\|^2}, \quad g_{iks}([\theta]) = \prod_{j=1, j \neq i, k, s}^{n} \frac{(\theta - \theta_j, \theta_i - \theta_j)}{\|\theta_i - \theta_j\|^2}.$$

These expressions can be easily proved by noting that

$$\frac{d}{d \epsilon} g_i([\theta + \epsilon \eta]) \bigg|_{\epsilon=0} = \sum_{k=1 \atop k \neq i}^{n} \frac{(\eta, \theta_i - \theta_k)}{\|\theta_i - \theta_k\|^2} \prod_{j=1 \atop j \neq k, i}^{n} \frac{(\theta - \theta_j, \theta_i - \theta_j)}{\|\theta_i - \theta_j\|^2}. \quad (151)$$
Note that the functional derivatives (152)-(153) are more complicated than the ones we obtained in the case Porter’s basis (119). As noted by Allasia and Bracco in [4], Prenter’s interpolants are badly conditioned as the number of interpolation nodes in function space increases. This unfortunate feature is common to many Lagrange interpolants.

3.2.5 Khlobystov Interpolants

We have seen that any continuous functional in a Hilbert or a Banach space can be approximated uniformly in terms of polynomial functionals [181, 17, 94, 134, 92, 216]. Such polynomial functionals may be built based on an interpolation process (see Section 3.2.3 and Section 3.2.4). What are the convergence properties of such expansions? To address this question, let us consider a polynomial interpolation problem of a given polynomial functional $\Pi_n(\{\theta\})$ in the form (101). Assume that $D(\Pi_n)$ is a Hilbert space of test functions, introduce the orthonormal basis $\{\varphi_1, \ldots, \varphi_m\}$ in $D(\Pi_n)$, and consider the following expansion

$$\Pi_n^I(\{\theta\}) = \sum_{k=1}^{n} \sum_{i_1=1}^{m} \cdots \sum_{i_k=1}^{m} (\theta, \varphi_{i_1}) \cdots (\theta, \varphi_{i_k}) a_k(\varphi_{i_1}, \ldots, \varphi_{i_k}),$$

(155)

where $(,)$ is an inner product in $D(\Pi_n)$ and $a_k(\varphi_{i_1}, \ldots, \varphi_{i_k})$ are real- or complex-valued coefficients. Clearly $\Pi_n^I(\{\theta\})$ is an interpolant of $\Pi_n(\{\theta\})$ at $\varphi_k$, i.e.,

$$\Pi_n^I(\varphi_k) = \Pi_n(\varphi_k).$$

(156)

The next step is to write the coefficients $a_k(\varphi_{i_1}, \ldots, \varphi_{i_k})$ in terms of $\Pi_n(\{\varphi_j\})$. To this end, we can use the method for finding orthonormal moments of regular polynomial functionals (Theorem 1 in [102]). This yields

$$n! a_n(\varphi_{i_1}, \ldots, \varphi_{i_n}) = \Pi_n(\{\varphi_{i_1} + \cdots + \varphi_{i_n}\}) - \Pi_n(\{\varphi_{i_1} + \cdots + \varphi_{i_{n-1}}\}) + \Pi_n(\{\varphi_{i_1} + \cdots + \varphi_{i_{n-2}} + \varphi_{i_n}\}) + \cdots + \Pi_n(\{\varphi_{i_2} + \cdots + \varphi_{i_n}\}) + \cdots + \Pi_n(\{\varphi_{i_3} + \cdots + \varphi_{i_n}\}) + \cdots + (-1)^{n-1} \Pi_n(\{\varphi_{i_1}\}) + \cdots + \Pi_n(\{\varphi_{i_n}\}) + (-1)^n \Pi_n(\{0\}).$$

(157)

Once $a_n(\varphi_{i_1}, \ldots, \varphi_{i_n})$ are available, we can construct the polynomial

$$\Pi_n - a_n(\varphi_{i_1}, \ldots, \varphi_{i_n}) \prod_{z=1}^{n} (\theta, \varphi_{i_z})$$

and apply (157) again to determine $a_{n-1}(\varphi_{i_1}, \ldots, \varphi_{i_{n-1}})$. After $n$ iterations, we have available all coefficients to construct the polynomial functional (155). It was shown in [102] (Theorem 2) that the following error estimate holds

$$\|\Pi_n^I(\{\theta\}) - \Pi_n(\{\theta\})\| \leq \sum_{k=1}^{n} \|P_k\| \left(\|\theta\| + \epsilon_m(\{\theta\})\right)^k - \|\theta\|^k,$$

(158)

where

$$\|P_k\| = \int \cdots \int |K_k(x_1, \ldots, x_k)| \, dx_1 \cdots dx_k,$$

(159)

$$\epsilon_m(\{\theta\}) = \left\|\theta - \sum_{k=1}^{m} (\theta, \varphi_k) \varphi_k\right\|.$$

(160)
This means that the interpolant (155) converges pointwise to the polynomial functional \( \Pi_n([\theta]) \) in (98) as the number of nodes \( \{\varphi_k\} \) in the function space \( D(\Pi_n) \) goes to infinity.\(^9\) As noted by Khlobystov in [103], Porter’s interpolant (112) can be written exactly in the form (155) if we consider test functions in the form \( \theta_i = \varphi_i + \ldots + \varphi_n \) (see Lemma 1 in [103]). In this sense, Porter’s interpolants represent the Lagrangian form of Khlobystov’s interpolants. An interesting and very important question is the approximation of \( \omega \) in terms interpolants over nodal sets \( S_q^{(m)} \), with \( q < n \) (see Eq. (97)).

A specific example would be the approximation of \( P_3([\theta]) \) in (130) by using an interpolant built on the set of nodes \( S_1^{(m)} \). In some very special cases we may get uniform convergence as \( m \to \infty \), for example when \( K_3 \) is diagonal

\[
K_3(x_1, x_2, x_3) = \sum_{j=1}^{m} a_j \varphi_j(x_1) \varphi_j(x_2) \varphi_j(x_3).
\]

However, this won’t happen in general, and therefore we will have to accept a systematic truncation error in representing continuous nonlinear functionals. Such error is similar the error committed when we approximate infinite-variate functions, e.g., by second-order HDMR [237] (see Section 3.5).

### Khlobystov interpolants with Hilbert-Schmidt Kernels

In this Section we present a procedure to construct interpolants of polynomial operators in Hilbert spaces with separable kernels. To this end, let us consider an orthonormal basis \( \{\varphi_1, \varphi_2, \ldots, \varphi_m\} \) and represent each kernel in (99) as

\[
K_1(x_1) = \sum_{i=1}^{m} a_i \varphi_i(x),
\]

\[
K_2(x_1, x_2) = \sum_{i,j=1}^{m} a_{ij} \varphi_i(x_1) \varphi_j(x_2),
\]

\[
K_3(x_1, x_2, x_3) = \sum_{i,j,k=1}^{m} a_{ijk} \varphi_i(x_1) \varphi_j(x_2) \varphi_k(x_3),
\]

\[\ldots\]

Without loss of generality we can assume that \( K_2, K_3, \ldots \) are symmetric, i.e., that the coefficients \( a_{ijk} \ldots \) are invariant under any permutation of the indices \( i, j, k, \ldots \). A substitution of (162), (163), etc., into (98) yields the polynomial functional

\[
\Pi_n([\theta]) = K_0 + \sum_{i=1}^{m} a_i(\varphi_i, \theta) + \sum_{i,j=1}^{m} a_{ij}(\varphi_i, \theta)(\varphi_j, \theta) + \sum_{i,j,k=1}^{m} a_{ijk}(\varphi_i, \theta)(\varphi_j, \theta)(\varphi_k, \theta) + \ldots.
\]

At this point we pose the following question: How many interpolation nodes do we need to identify the coefficients \( a_i, a_{ij}, a_{ijk}, \ldots \), uniquely? To clarify the question, consider the following second-order polynomial functional

\[
\Pi_2([\theta]) = K_0 + \sum_{i=1}^{m} a_i(\varphi_i, \theta) + \sum_{i,j=1}^{m} a_{ij}(\varphi_i, \theta)(\varphi_j, \theta).
\]

The total number of degrees of freedom (number of independent coefficients \( K_0, a_i \) and \( a_{ij} \)) is

\[
1 + m + m + \binom{m}{2} = 1 + \frac{3m + m^2}{2}.
\]

\(^9\)Note that \( \epsilon_n([\theta]) \to 0 \) as \( m \to \infty \). Thus, the right hand side of the error estimate (158) goes to zero as \( m \to \infty \), i.e., \( \Pi_n([\theta]) \to \Pi_n([\theta]) \) pointwise as \( m \to \infty \).
To determine such coefficients, we need to test $\Pi_2([\theta])$ at $(2 + 3m + m^2)/2$ distinct nodes\footnote{Note, that testing a second-order polynomial functional in $\varphi_i(x)$ or $2\varphi_i(x)$ yields different results.}, e.g.,

$$\left\{0, \{\varphi_i\}_{i=1}^m, \{\varphi_i + \varphi_j\}_{i,j=1}^m \right\}.$$  \hfill (168)

This yields the linear system

$$\begin{align*}
\Pi_2([0]) &= K_0 \\
\Pi_2([\varphi_p]) &= K_0 + a_p + a_{pp} \\
\Pi_2([\varphi_p + \varphi_q]) &= K_0 + a_p + a_q + a_{pp} + a_{qq} + 2a_{pq}
\end{align*}$$

which can be immediately solved for $a_{pq}, a_p$ and $K_0$

$$\begin{align*}
a_{pq} &= \frac{1}{2} \left( \Pi_2([\varphi_p + \varphi_q]) - \Pi_2([\varphi_p]) - \Pi_2([\varphi_q]) + \Pi_2([0]) \right), \\
a_p &= -\frac{1}{2} \Pi_2([2\varphi_p]) + 2 \Pi_2([\varphi_p]) - \frac{3}{2} \Pi_2([0]), \\
K_0 &= \Pi_2([0]).
\end{align*}$$  \hfill (169)-(171)

In this way, we can identify the kernels (162)-(164) and therefore any polynomial functional in the form (166). Note that if the basis elements in (168) are not normalized (but still orthogonal) then we simply need to replace $a_{pq}$ and $a_p$ in (169)-(171) with $||\varphi_p||^2 ||\varphi_q||^2 a_{pq}$ and $||\varphi_p||^2 a_p$, respectively. Higher-order polynomial functionals can be constructed in a similar way. However, the number of degrees of freedom may increase significantly with the order of the polynomial (see equation (93)). For example, a twelve-order polynomial functional in which each kernel is represented relative to $m = 10$ basis functions (tensor product) yields $646646$ degrees of freedom!

**Remark:** The fact that we can get analytical expressions for the coefficients of the polynomial interpolant means that Porter’s matrix (120) is invertible analytically for uniquely solvable interpolation problems and orthonormal bases.

The procedure we just described to identify the coefficients of the symmetric kernel functions $K_n$ relies on tensor product representations, i.e., series expansions in the form

$$K_n(x_1, x_2, \ldots, x_n) = \sum_{i_1, \ldots, i_n=1}^{m} a_{i_1 \cdots i_n} \varphi_{i_1}(x_1) \cdots \varphi_{i_n}(x_n).$$  \hfill (172)

The number of independent coefficients $a_{i_1 \cdots i_n}$ is

$$\binom{n + m - 1}{n} = \frac{(n + m - 1)!}{n!(m-1)!}.$$  \hfill (173)

Such number can pose serious computational challenges even for moderate values of $m$ and $n$. To alleviate this problem one could use HDMR expansions \cite{183,126,125}, i.e., represent $K_n(x_1, x_2, \cdots, x_n)$ in terms of a superimposition of functions involving a lower number of variables (interaction terms). This yields, for example

$$K_n(x_1, x_2, \ldots, x_n) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + \sum_{i<j<k} f_{ijk}(x_i, x_j, x_k) + \cdots.$$  \hfill (174)
The function $f_0$ is a constant. The functions $f_i(x_i)$ (first-order interactions) give us the overall effects of the variables $x_i$ in $f$ as if they were acting independently of the other input variables. The functions $f_{ij}(x_i, x_j)$ (second-order interactions) describe the cooperative effect of the variables $x_i$ and $x_j$. Similarly, higher-order terms reflect the cooperative effects of an increasing number of variables. Representing $f_i, f_{ij}, f_{ijk}$ relative to the orthonormal basis $\{\varphi_j(x)\}$ yields the series

$$K_n(x_1, x_2, ..., x_n) = f_0 + \sum_{i=1}^{n} \sum_{s=1}^{m} a^i_s \varphi_s(x_i) + \sum_{i<j}^{n} \sum_{s,q=1}^{m} a^{ij}_{sq} \varphi_s(x_i) \varphi_q(x_j) + \sum_{i<j<k}^{n} \sum_{s,q,r=1}^{m} a^{ijk}_{sq} \varphi_s(x_i) \varphi_q(x_j) \varphi_r(x_j) + \cdots .$$

(175)

Given the symmetry of each function $f_i, f_{ij}, f_{ijk}$, the total number of degrees of freedom of an HDMR expansion of order $Z$ is

$$\sum_{k=0}^{Z} \binom{n}{k} \binom{m + k - 1}{k} = 1 + \binom{n}{1} \binom{m}{1} + \binom{n}{2} \binom{m+1}{2} + \binom{n}{3} \binom{m+2}{3} + \cdots.$$

(176)

For example, the second- and third-order truncations of a $n = 10$ dimensional kernel relative to $m = 10$ yield 2576 and 28976 degrees of freedom, respectively. On the other hand, the tensor product representation yields 92378 degrees of freedom. Alternatively, one can use tensor expansions (see Section (3.3)), e.g., canonical polyadic or hierarchical-Tucker series, of each kernel to reduce the number of degrees of freedom. The tensor expansion can be fit to data by interpolation, least-squares or projection [159, 158, 156, 157].

### 3.3 Functional Approximation by Tensor Methods

Computing high-order polynomial functional expansions requires representing kernel functions in many independent variables. To get an feeling of how serious this problem could be, simply consider that representing a polynomial functional of order 6 in the Hilbert Space

$$K_6(x_1, x_2, ..., x_n) = f_0 + \sum_{i=1}^{n} \sum_{s=1}^{m} a^i_s \varphi_s(x_i) + \sum_{i<j}^{n} \sum_{s,q=1}^{m} a^{ij}_{sq} \varphi_s(x_i) \varphi_q(x_j) + \sum_{i<j<k}^{n} \sum_{s,q,r=1}^{m} a^{ijk}_{sq} \varphi_s(x_i) \varphi_q(x_j) \varphi_r(x_j) + \cdots ,$$

(176)

is as computationally expensive as representing the solution to the steady-state Boltzmann equation [41], a well-known challenging problem in computational physics. Expanding each kernel of the polynomial functional in terms of HDMR [185] or canonical polyadic decompositions i.e. separated series expansions [18] – can mitigate the dimensionality problem, but it may not be the most efficient way to proceed. In this Section we discuss nonlinear functional approximation based on tensor methods. To introduce these methods, consider the Hilbert space of functions spanned by the finite-dimensional basis $\{\varphi_1, ..., \varphi_m\}$ (e.g., an orthonormal basis) and look for an approximant of $F(\theta)$, say $f$, in the form

$$F(\theta) = f(a_1(\theta), ..., a_m(\theta)) + R(\theta), \quad a_k(\theta) = (\theta, \varphi_k).$$

(177)

In this equation, $f$ is a multivariate function of $a_k(\theta)$ (linear functionals of $\theta$), $(\cdot, \cdot)$ denotes an inner product in the Hilbert Space $D(F)$, e.g.,

$$(\theta, \varphi_k) = \int_a^b \theta(x) \varphi_k(x) dx, \quad k = 1, ..., m$$

(178)

and $R$ is a (functional) reminder term. The functionals $a_k(\theta) = (\theta, \varphi_k)$ can be either real or complex-valued. In the theory of stochastic processes the set

$$\{\theta \in D(F) | ((\theta, \varphi_1), ..., (\theta, \varphi_m)) \subseteq \mathbb{R}^m\}$$

(179)
is known as *cylindrical set* (see, e.g., [221] p. 56 or [207] p. 45). Therefore, according to Eq. (177), we are looking for an approximant of $F([\theta])$ in the space of *cylindrical functionals*, i.e., functionals defined on cylindrical sets. Thanks to the Stone-Weierstrass theorem, cylindrical functionals in the form (180) can approximate any continuous functional in a Hilbert or a Banach space. The representation (177) is very general. For example, it includes the case where $f$ is a polynomial functional, e.g., (155) or (112), or the case where the functional is defined on a finite-dimensional function space (see Section 3.1). For notational convenience we will often drop the functional dependence of $a_k([\theta])$ and write (177) as

$$F([\theta]) \simeq f(a_1, \ldots, a_m), \quad a_j = (\theta, \varphi_j). \quad (180)$$

Hereafter, we discuss effective numerical algorithms to compute an approximation of the multivariate function $f(a_1, \ldots, a_m)$ based on high-dimensional model representations (HDMR), and *tensor methods*\footnote{If we ask the question “what is a tensor?” to an engineer, a physicist or a mathematician we usually get different answers. The engineer usually says “a tensor is a matrix”. On the other hand, the physicist would say that a tensor is a mathematical object that has the fundamental property of transforming in a very specific way when the coordinate system is changed. He or she would point out that the laws of physics are built upon the principle of general covariance \cite{238, 223} (invariance of physical laws relative to coordinate transformations) which is formulated in a natural way in terms of tensors. The word “tensor” has recently spilled in the multi-linear algebra community to represent multi-dimensional arrays. Hereafter we will adopt such terminology, and refer to tensors as multi-dimensional arrays.}, including canonical tensor decompositions, hierarchical Tucker formats, and tensor networks. Such algorithms rely on optimization (e.g., the alternating least squares methods \cite{1, 188}), or multilinear algebra techniques such as high-order singular value decomposition \cite{77}, randomized block sampling, or generalized Schur decompositions.

**Functional Derivatives** Let us compute the functional derivatives of the cylindrical functional approximation (180). To this end, we evaluate the Gâteaux differential of $f(a_1([\theta]), \ldots, a_m([\theta]))$ in the direction of an arbitrary function $\eta(x)$ to obtain

$$\int_a^b \frac{\delta F([\theta])}{\delta \theta(x)} \eta(x) dx \simeq \frac{d}{d\epsilon} [f((\theta + \epsilon \eta, \varphi_1), \ldots, (\theta + \epsilon \eta, \varphi_m))]_{\epsilon=0}$$

$$= \sum_{k=1}^m \frac{\partial f}{\partial a_k}(\eta, \varphi_k). \quad (181)$$

Setting $\eta(x) = \delta(x - y)$ yields the following approximation for the first-order functional derivative

$$\frac{\delta F([\theta])}{\delta \theta(y)} \simeq \sum_{k=1}^m \frac{\partial f}{\partial a_k} \varphi_k(y), \quad (183)$$

where $a_k = (\theta, \varphi_k)$. On the other hand, by projecting (183) onto $\varphi_j$ we obtain

$$\int_a^b \frac{\delta F([\theta])}{\delta \theta(x)} \varphi_j(x) dx \simeq \frac{\partial f}{\partial a_j}. \quad (184)$$

This means that partial derivative of $f$ relative to $a_k = (\theta, \varphi_k)$ approximates the projection of the functional derivative of $F$ along $\varphi_k$. Equations (183) and (184) are consistent with previous results on functional derivatives in finite-dimensional spaces (see Eqs. (66) and (67)). By following the same procedure, we can...
construct functional derivatives of \( f \) of higher-order, e.g.,

\[
\frac{\delta^2 F(\theta)}{\delta \theta(x) \delta \theta(y)} \simeq \sum_{k,j=1}^{m} \frac{\partial^2 f}{\partial a_k \partial a_j} \varphi_k(x) \varphi_j(y),
\]

(185)

\[
\frac{\delta^3 F(\theta)}{\delta \theta(x) \delta \theta(y) \delta \theta(z)} \simeq \sum_{k,j,i=1}^{m} \frac{\partial^3 f}{\partial a_k \partial a_j \partial a_i} \varphi_k(x) \varphi_j(y) \varphi_i(z).
\]

(186)

**Choice of the Number of Active Dimensions** The choice of the basis functions \( \varphi_k \) and the number of active dimensions, i.e., the parameter \( m \) in (177), is critical for the accuracy of the cylindrical representation. For a fixed basis \( \{ \varphi_j \} \), smaller values of \( m \) yield faster computations but at the same time can lead to functional approximation problems with poor approximation errors.

### 3.3.1 Canonical Tensor Decomposition

The canonical tensor decomposition of the multivariate function \( f(a_1, ..., a_m) \) in (177) is a series expansion in the form

\[
f(a_1, ..., a_m) = \sum_{l=1}^{r} \prod_{i=1}^{m} G^l_i(a_i), \quad a_j = (\theta, \varphi_j),
\]

(187)

where \( G^l_i(a_i) \) are one-dimensional functions usually represented relative to a known basis \( \{ \phi_1, ..., \phi_Q \} \), i.e.,

\[
G^l_i(a_i) = \sum_{k=1}^{Q} \beta^l_{ik} \phi_k(a_i).
\]

(188)

The quantity \( r \) in (187) is called *separation rank*. In the statistics literature, representations of the form (187) are known as *parallel factorizations* (see [115, 123]). They are also known as proper generalized decompositions [31], canonical polyadic expansions (CP) [100], separated series [33], and Kruskal tensor formats [113]. Although there are at present no useful theorems on the size of the separation rank \( r \) needed to represent with accuracy general classes of functionals \( f \), there are cases where the expansion (187) is exponentially more efficient than one would expect a priori. The basis functions \( \phi_k \) appearing in (188) represent the variability of the functional \( f \) along different directions \( a_k = (\theta, \varphi_k) \) in the test function space \( D_m \). As such, they have to satisfy appropriate boundary conditions. For example, if \( f \) is periodic in the hypercube \([-b, b]^m\) then we could use rescaled trigonometric polynomials

\[
\phi_k(a) = l_k \left( \pi \left( \frac{a}{b} + 1 \right) \right) \quad a \in [-b, b],
\]

(189)

where

\[
l_{k+1}(x) = \frac{1}{Q} \frac{\sin \left( \frac{Q}{2} (x - x_k) \right)}{\sin \left( \frac{x - x_k}{2} \right)}, \quad x_k = \frac{2\pi k}{Q}, \quad k = 0, ..., (Q - 1).
\]

(190)

For more general boundary conditions we can use a polynomial basis, e.g., rescaled Legendre orthogonal polynomials

\[
\phi_k(a) = L_k \left( \frac{a}{b} \right) \quad a \in [-b, b],
\]

(191)

where

\[
L_0(x) = 1, \quad L_1(x) = x, \quad \cdots, \quad L_{n+1}(x) = \frac{2n+1}{n+1} x L_n(x) - \frac{n}{n+1} L_{n-1}(x), \quad (n = 1, ..., Q - 1).
\]

(192)
The $L_2$ norm of (189) and (191) is easily obtained as
\[ \eta_k = \frac{b}{\pi} \frac{2\pi}{Q} \quad \text{(Fourier)}, \quad \eta_k = \frac{2b}{2k + 1} \quad \text{(Legendre)}, \quad k = 1, \ldots, Q. \quad (193) \]

**Example 1:** Consider the sine functional (63), hereafter rewritten for convenience
\[ F(\theta) = \sin \left( \int_a^b K(x) \theta(x) \, dx \right). \quad (194) \]
Assuming that the kernel $K(x)$ admits the finite-dimensional expansion
\[ K(x) = \sum_{j=1}^m k_j \varphi_j(x), \quad (195) \]
and substituting it into (194) we obtain
\[
\begin{align*}
f(a_1, \ldots, a_m) &= \sin \left( \sum_{j=1}^m k_j a_j \right) \\
&= \sum_{j=1}^m \sin(k_j a_j) \prod_{i=1 \atop i \neq j}^m \frac{\sin(k_i a_i + \chi_i - \chi_j)}{\sin(\chi_i - \chi_j)},
\end{align*} \quad (196)
\]
for all $\chi_i$ such that $\chi_i \neq \chi_j$. The last equality was derived in [143], and it shows that the separation rank of the canonical tensor decomposition of (194)-(195) is exactly $r = m$. In other words, we can represent the nonlinear functional (194) exactly in terms of superimposition of $m$ terms. Furthermore, if we allow the $G^l_i(a_i)$ to be complex-valued\(^{12}\), then
\[
\sin \left( \sum_{j=1}^m k_j a_j \right) = \prod_{j=1}^m \frac{e^{ik_j a_j}}{2i} - \prod_{j=1}^m \frac{e^{-ik_j a_j}}{2i},
\quad (197)
\]
i.e., we can reduce the separation rank to $r = 2$. In general, the separation rank depends on the complexity of the nonlinear function $f(a_1, \ldots, a_m)$.

**Functional Derivatives** With the canonical tensor decomposition (187) available, it is straightforward to compute an approximation of the functional derivatives of $F(\theta)$. Recalling that the canonical tensor decomposition is a cylindrical representation of the functional $F(\theta)$, we have the expressions (183), (185) and (186). The unknowns are the partial derivatives of $f$ with respect to $a_j(\theta)$, which can be computed based on (187) as
\[
\frac{\partial f}{\partial a_i} = \sum_{l=1}^r \frac{\partial G^l_i}{\partial a_i} \prod_{k=1 \atop k \neq i}^m G^l_k(a_k),
\quad (198)
\]
\(^{12}\)Constraints on the functions $G^l_i$ such as positivity can be also imposed, e.g., if one is interested in probability functionals.
and
\[
\frac{\partial f}{\partial \alpha_i \partial \alpha_j} = \begin{cases} 
\sum_{l=1}^{r} \frac{\partial^2 G^l_i}{\partial \alpha_i^2} \prod_{k=1}^{m} G^l_k(\alpha_k) & i = j, \\
\sum_{l=1}^{r} \frac{\partial G^l_i}{\partial \alpha_i} \frac{\partial G^l_j}{\partial \alpha_j} \prod_{k=1}^{m} G^l_k(\alpha_k) & i \neq j.
\end{cases}
\tag{199}
\]

These derivatives are evaluated at \( \alpha_j(\theta) = (\theta, \varphi_j) \).

**Alternating Least Squares (ALS) Formulation**  The development of robust and efficient algorithms to compute \((187)\) to any desirable accuracy is still a relatively open question (see \([1, 54, 100, 45, 33]\) for recent progresses). Computing the tensor components \(G^l_k(\alpha_k)\) usually relies on (greedy) optimization techniques such as alternating least squares (ALS) \([188, 12, 1, 18]\) or regularized Newton methods \([54]\), which are only locally convergent \([219]\) (i.e., the final result may depend on the initial condition of the algorithm). Hereafter we describe the simplest form of the ALS algorithm. To this end, consider the functional residual \((177)\), with \(f\) given in \((187)\)
\[
R(\theta) = F(\theta) - \sum_{l=1}^{r} \prod_{k=1}^{m} G^l_k((\theta, \varphi_k)).
\tag{200}
\]

We aim at computing the tensor components \(G^l_k(\alpha_k)\) by minimizing the norm of \(R(\theta)\) relative to independent variations of \(G^l_k\). In particular, if we assume that \(G^l_k\) are in the form \((188)\), then variations of \(G^l_k\) are generated by variations of \(\beta^{n}_{hj}\). Therefore, the canonical tensor decomposition of \(F(\theta)\) can be computed by the variational principle
\[
\min_{\beta^{n}_{hj}} \|R(\theta)\|^2_W.
\tag{201}
\]

The norm in \(\| \cdot \|_W\) may be defined by a weighted functional integral (see Appendix \([8]\) in the form
\[
\|R(\theta)\|^2_W = \int R(\theta)^2 W(\theta) D\theta,
\tag{202}
\]
where \(W(\theta)\) is the functional integration measure, or by a discrete sum (functional collocation setting)
\[
\|R(\theta)\|^2_W = \sum_{j=1}^{N} R(\theta_j)^2 w_j,
\tag{203}
\]
where \(\{\theta_1, ..., \theta_N\}\) are collocation nodes in the function space \(D(F)\), and \(w_j\) are integration weights. In the alternating least squares paradigm, we compute the minimizer of the residual \((200)\) by splitting the non-convex optimization problem \((201)\) into a sequence of convex low-dimensional optimization problems (see Figure \([8]\)). To illustrate the method, let us define the vectors
\[
\beta_1 = [\beta^{1}_{11}, ..., \beta^{1}_{1Q}, ..., \beta^{1}_{r1}, ..., \beta^{1}_{rQ}]^T,
\beta_2 = [\beta^{2}_{21}, ..., \beta^{2}_{2Q}, ..., \beta^{2}_{r1}, ..., \beta^{2}_{rQ}]^T,
\vdots
\beta_m = [\beta^{m}_{m1}, ..., \beta^{m}_{mQ}, ..., \beta^{m}_{r1}, ..., \beta^{m}_{rQ}]^T.
\tag{204}
\]

\[\text{The Euler-Lagrange equations associated with (201) are nonlinear } \beta^{n}_{hj}.\]
Figure 8: Sketch of the alternating least squares (ALS) algorithm for the minimization of the functional residual $R(\theta)$. Convergence analysis of ALS be found in [55, 19, 191, 219].

Note that $\beta_i$ collects the degrees of freedom of all functions $\{G_1, ..., G_r\}$ depending on $a_i = (\theta, \varphi_i)$. Next, we split the optimization problem (201) into the following sequence of convex optimization problems

$$
\min_{\beta_1} \|R(\theta)\|_W^2 \quad \rightarrow \quad \min_{\beta_2} \|R(\theta)\|_W^2 \quad \rightarrow \quad \cdots \quad \rightarrow \quad \min_{\beta_m} \|R(\theta)\|_W^2.
$$

We emphasize that the system of equations (205) is not equivalent to the full problem (201). In other words, the sequence of low-dimensional optimization problems (205) in general does not allow us to compute the minimizer of (201) [55, 19, 191, 219]. The Euler-Lagrange equations associated with (205) are in the form

$$
A_j \beta_j = f_j, \quad j = 1, ..., m,
$$

where

$$
A_j = \begin{bmatrix}
A_{11}^j & A_{12}^j & \cdots & A_{1r}^j \\
A_{21}^j & A_{22}^j & \cdots & A_{2r}^j \\
\vdots & \vdots & \ddots & \vdots \\
A_{r1}^j & A_{r2}^j & \cdots & A_{rr}^j
\end{bmatrix}, \quad f_j = \begin{bmatrix}
f_{j1} \\
f_{j2} \\
\vdots \\
f_{jr}
\end{bmatrix},
$$

and

$$
A_{jhs}^{n} = \int \mathcal{D}[\theta]W([\theta])\phi_h((\theta, \varphi_j))\phi_n((\theta, \varphi_j)) \prod_{k=1 \atop k \neq j}^m G_k^n((\theta, \varphi_k)) G_k^m((\theta, \varphi_k)),
$$

$$
f_{jh}^n = \int \mathcal{D}[\theta]W([\theta])F([\theta])\phi_h((\theta, \varphi_j)) \prod_{k=1 \atop k \neq j}^m G_k^m((\theta, \varphi_k)).
$$

The matrices $A_j$ are symmetric, positive definite and of size $rQ \times rQ$. Also, the functional integrals defining the matrix entries can be simplified and eventually computed by using techniques for high-dimensional...
integration, such as the quasi-Monte Carlo method \([40]\). Indeed, if we restrict the residual \((200)\) to the finite-dimensional function space \(D_m = \text{span}\{\varphi_1, ..., \varphi_m\}\) (Section 3.1), and assume that \(F(\{\theta\})\) restricted to \(D_m\) is compactly supported within the hypercube \([-b, b]^m\), then \(A_{jh}^{ln}\) and \(f_{jh}^n\) can be written in the form

\[
A_{jh}^{ln} = \int_{-b}^{b} \phi_h(a_j)\phi_s(a_j) \prod_{k=1}^{m} \int_{-b}^{b} G_k^l(a_k)G_k^n(a_k)da_k,
\]

\[
f_{jh}^n = \int_{-b}^{b} \cdots \int_{-b}^{b} f(a_1, ..., a_m)\phi_h(a_j) \prod_{k=1}^{m} G_k^n(a_k)da_1 \cdots da_m,
\]

provided we select the integration measure \(W(\{\theta\})\) appropriately (see Appendix B).

**Remark:** Minimizing the residual \((200)\) with respect to \(\beta_{hj}^m\) is equivalent to imposing orthogonality relative to the space spanned by the functionals

\[
\phi_h((\theta, \phi_j)) \prod_{k=1}^{m} G_k^n((\theta, \varphi_k))
\]

To show this in a simple and intuitive way, consider the following example in just two dimensions. Let \(f(a_1, a_2)\) be a regular function defined on the unit square \([0, 1]^2\). We look for a canonical tensor decomposition of \(f(a_1, a_2)\) in the form

\[
f(a_1, a_2) = \sum_{j=1}^{r} G_1^j(a_1)G_2^j(a_2), \quad \text{where} \quad G_k^j(a_k) = \sum_{p=1}^{P} \beta_{kp}^j \phi_p(a_k).
\]

To determine \(\beta_{kp}^j\) we first define the residual,

\[
R(a_1, a_2) = f(a_1, a_2) - \sum_{j=1}^{r} G_1^j(a_1)G_2^j(a_2)
\]

and then minimize its weighed norm

\[
\|R\|^2_w = \int_0^1 \int_0^1 R(a_1, a_2)^2 w(a_1, a_2)da_1 da_2
\]

relative to independent variations of \(\beta_{1p}^j\) and \(\beta_{2p}^j\). This yields

\[
\int_0^1 \int_0^1 w(a_1, a_2)R(a_1, a_2)\phi_p(a_1)G_2^j(a_2)da_1 da_2 = 0 \quad \text{linear system for} \ \beta_{1p}^j,
\]

\[
\int_0^1 \int_0^1 w(a_1, a_2)R(a_1, a_2)G_1^j(a_1)\phi_p(a_2)da_1 da_2 = 0 \quad \text{linear system for} \ \beta_{2p}^j.
\]

Thus, minimizing the residual with respect to independent variations of \(\beta_{1p}^j\) and \(\beta_{2p}^j\) is equivalent impose Galerkin orthogonality relative to a space spanned by the basis functions \(\phi_p(a_1)G_2^j(a_2)\) and \(G_1^j(a_1)\phi_p(a_2)\), respectively.
**Convergence of the ALS Algorithm** The ALS algorithm we just described is an alternating optimization scheme, i.e., a nonlinear block Gauss–Seidel method ([167], §7.4). There is a well-developed local convergence theory for this type of method (see [167] [19]). In particular, it can be shown that ALS is locally equivalent to the linear block Gauss–Seidel iteration applied to the Hessian matrix. This implies that ALS is linearly convergent in the iteration number [219], provided that the Hessian of the residual is positive definite (except on a trivial null space associated with the scaling non-uniqueness of the canonical tensor decomposition). The last assumption may not be always satisfied. Therefore, convergence of the ALS algorithm cannot be granted in general. Another potential issue of the ALS algorithm is the poor conditioning (decomposition). The last assumption may not be always satisfied. It was shown in [219] that the classical alternating least squares algorithm does not converge in the iteration number for (217), provided that the Hessian of the residual is positive definite (except on a trivial null space associated with the scaling non-uniqueness of the canonical tensor decomposition). The last assumption may not be always satisfied. Therefore, convergence of the ALS algorithm cannot be granted in general. Another potential issue of the ALS algorithm is the poor conditioning of the matrices $A_j$ in [206], which can addressed by regularization [188]. The canonical tensor decomposition (187) in $m$ dimensions has relatively small memory requirements. In fact, the number of degrees of freedom that we need to store is $r \times m \times Q$, where $r$ is the separation rank, and $Q$ is the number of degrees of freedom employed in each tensor component $G_k^j(a_k)$ [188]. Despite the relatively low-memory requirements, it is often desirable to employ scalable parallel versions the ALS algorithm [100] to compute the canonical tensor expansion (187).

### 3.3.2 Tucker Decomposition

The Tucker decomposition of the cylindrical functional (180) is a series expansion in the form

$$f(a_1, ..., a_m) = \sum_{l_1=1}^{r_1} \cdots \sum_{l_m=1}^{r_m} C_{l_1 \cdots l_m} T_{l_1}^{1}(a_1) \cdots T_{l_m}^{m}(a_m),$$

(217)

where $C_{l_1 \cdots l_m}$ is a $r_1 \times \cdots \times r_m$ real- or complex-valued tensor – often referred to as core tensor [113] – and $T_{l_m}^{m}(a_m)$ are unknown functions. Tucker decomposition is known as high-order Schmidt decomposition in the context of quantum mechanics [27]. It is important to emphasize such decomposition is, in general, are not unique [15]. As pointed out by Kolda and Bader in [113], this freedom opens the possibility to choosing transformations that simplify the core structure in some way so that most of the elements of the core tensor $C_{l_1 \cdots l_m}$ are zero, thereby eliminating interactions between corresponding components. Diagonalization of the core is, in general, impossible [16] [171], but it is possible to try to make as many elements either zero or as small as possible (see, e.g., [27] or [147]). For general tensors we have that the multilinear rank $r = r_j$ ($j = 1, ..., m$) is upper semi-continuous, i.e., the Tucker expansion is closed. Several efficient algorithms are currently available to compute the series expansion (217). For instance, Lathauer et al. proposed in [118] a high-order singular value decomposition method to determine the components $T_{l_k}^{k}$ and the core tensor $C_{l_1 \cdots l_m}$ in a discrete setting. Such algorithm is simple, robust and it yields quasi-optimal low-rank approximations.

To illustrate the procedure to compute the Tucker decomposition let us first assume that the basis functions $T_k^{l_k}(a_k)$ in (217) are orthonormal and known. In this case, the expansion (217) is simply a tensor

\[ f(a_1, ..., a_m) = \sum_{l_1=1}^{r_1} \cdots \sum_{l_m=1}^{r_m} C_{l_1 \cdots l_m} G_1^{l_1}(a_1) \cdots G_m^{l_m}(a_m). \]

(218)

The fact that diagonalization of $C_{l_1 \cdots l_m}$ is, in general, impossible in dimension larger than 2 implies that it is impossible to compute canonical tensor decompositions by standard linear algebra techniques. Indeed, the best low-rank approximation problem is ill-posed for real tensors with dimension $m > 2$ (see, e.g., [36] [113] [85], and for complex tensors [222].
product representation of a multivariate function relative to the basis $T_k^{l_k}$. The core tensor $C_{l_1 \cdots l_m}$ can be immediately obtained by projecting $f(a_1, \ldots, a_m)$ onto the basis $T_k^{l_k}(a_k)$, i.e.,

$$
C_{l_1 \cdots l_m} = \int f(a_1(\theta), \ldots, a_m(\theta)) T_1^{l_1}(a_1(\theta)) \cdots T_m^{l_m}(a_m(\theta)) W(\theta) D(\theta).
$$

(219)

Evaluating the functional integral in $D_m = \text{span}\{\phi_1, \ldots, \phi_m\}$ and rescaling the integration measure $W(\theta)$ properly (see Appendix B) allows us to rewrite (219)

$$
C_{l_1 \cdots l_m} = \int_{-b}^{b} \cdots \int_{-b}^{b} f(a_1, \ldots, a_m) T_1^{l_1}(a_1) \cdots T_m^{l_m}(x_m) da_1 \cdots da_m,
$$

(220)

where we assumed $f$ to be compactly supported in $[-b, b]^m$. Next, suppose that each function $T_k^{l_k}(a_k)$ is a linear combination of $Q$ orthonormal basis functions $\phi_j$, i.e.,

$$
T_k^{l_k}(a_k) = \sum_{s=1}^{Q} \alpha_{sk}^{l_k} \phi_s(a_k) \quad \text{(fixed } k\text{)}.
$$

(221)

If $\phi_s$ is a cardinal basis associated with a set of interpolation nodes along $a_k$, then the matrix $\alpha_{sk}^{l_k}$ represents the set of functions $T_k^{l_k}(a_k)$ for fixed $k$. We can sort arrange the matrix $\alpha_{sk}^{l_k}$ in a way where the $l_k$-column represents the value of $T_k^{l_k}(a_k)$ at $Q$ collocation nodes along $a_k$. This yields the matrix with entries $[T_k^{l_k}]_{il}$ ($i = 1, \ldots, Q$). If we evaluate the multivariate field $f(a_1, \ldots, a_m)$ at the same collocation nodes we employed to construct the interpolants of $T_k^{l_k}$, then we can rewrite (217) in a full tensor notation as

$$
f_{i_1 \cdots i_m} = \sum_{l_1=1}^{r_1} \cdots \sum_{l_m=1}^{r_m} C_{l_1 \cdots l_m} [T_1]_{i_1l_1} \cdots [T_m]_{i_ml_m},
$$

(222)

where $i_j = 1, \ldots, Q$ are indices identifying the interpolation node along the axis $a_j$.

Remark: The expansion (222) is a “matricization” of continuum series (217) obtained by representing each basis element in terms of collocation nodes. Clearly, we can also set up a matricization of (217) based on Galerkin projection. To this end, it is sufficient to project both the left and the right hand sides of (217) onto the orthonormal basis elements $\phi_i(a_1) \phi_{i_2}(a_2) \cdots \phi_{i_m}(a_m)$ to obtain an expression in the form (222). The meaning of $f_{i_1 \cdots i_m}$ in the case is the Fourier coefficients of the projection, i.e.,

$$
f_{i_1 \cdots i_m} = \int_{-b}^{b} \cdots \int_{-b}^{b} f(a_1, \ldots, a_m) \phi_{i_1}(a_1) \cdots \phi_{i_m}(a_m) dx_1 \cdots dx_m.
$$

(223)

Thus, (222) represents a multivariate expansion of Fourier coefficients in a Tucker tensor format. In such finite-dimensional setting, we basically transformed the problem of decomposing a multivariate function to a multi-linear algebra problem. It is immediate to see that the discrete Tucker format a two-dimensional function $f(a_1, a_2)$ is

$$
f_{i_1, i_2} = \sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} C_{l_1l_2}[T_1]_{i_1l_1}[T_2]_{i_2l_2},
$$

(224)

We can obviously diagonalize the core tensor by using singular value decomposition.

A drawback of the Tucker decomposition is the storage requirement of the core tensor $C_{l_1 \cdots l_m}$, which is $O(r^m)$. Such problem can be mitigated by attempting to make zero as many entries of $C_{l_1 \cdots l_m}$ as possible.
through suitable linear transformations. Another option is to introduce further separability properties of the core tensor. This yields a multitude of possible expansions, including hierarchical Tucker [75, 8] and Tucker tensor train [159, 191]. All these series expansions can be conveniently visualized by suitable graphs, and as such they fall within the setting of tensor-networks.

3.3.3 Hierarchical Tucker Decomposition

We have seen that the canonical tensor decomposition of a nonlinear functional has a relatively small number of degrees of freedom, but its computation is not straightforward. The most reliable algorithms are based on regularized optimization and randomized methods (see [188, 12] for recent results), which often yield different series expansions for different initial conditions. On the other hand, we have seen the that high-order Schmidt decomposition suffers from the curse of dimensionality (dimension of the core tensor), but it is straightforward to compute by nearly optimal and robust techniques such as high-order singular value decomposition [118]. The hierarchical Tucker format was introduced by Hackbusch and Kühn in [79] (see also [78, 75]) to mitigate the dimensionality problem in the core-tensor of the classical high-order Schmidt decomposition. The key idea is to perform a sequence of classical bi-variate Schmidt decompositions until the approximation problem is reduced to a product of one-dimensional functions. To illustrate the method is a simple way, consider the following cylindrical functional

\[ f(a_1, a_2, a_3) = \sum_{i,j=1}^{r} A_{ij}^{(1)} T_{i}^{(1)}(a_1) T_{j}^{(2,3)}(a_2, a_3), \quad a_j = (\theta, \varphi_j). \]  

(225)

The matrix representation of \( f(a_1, a_2, a_3) \) (either Galerkin or collocation form) relative to the given bases \( T_{i}^{(1)} \) and \( T_{j}^{(2,3)} \) is \( A_{ij}^{(1)} \), i.e., it has two indices. \( \) We decompose \( T_{j}^{(2,3)} \) further by another Schmidt expansion

\[ T_{j}^{(2,3)}(a_2, a_3) = \sum_{l,n=1}^{r} A_{jln}^{(2)} T_{l}^{(2)}(a_2) T_{n}^{(3)}(a_3), \]  

(226)

to obtain

\[ f(a_1, a_2, a_3) = \sum_{i,j,l,n=1}^{r} A_{ijn}^{(2)} A_{ij}^{(1)} T_{i}^{(1)}(a_1) T_{l}^{(2)}(a_2) T_{n}^{(3)}(a_3). \]  

(227)

The procedure just described is at the basis of the hierarchical Tucker decomposition and it yields an expansion in the form [217], where the core tensor is factored as a product of at most three-dimensional tensors. To show this in a higher-dimensional case, consider the six-dimensional cylindrical functional \( f(a_1, ..., a_6) \). By performing a sequence of Hilbert-Schmidt factorizations, while keeping the separation rank \( r \) constant in each factorization, we obtain

\[ f(a_1, ..., a_6) = \sum_{i_7, i_8=1}^{r} A_{i_7i_8}^{(1)} T_{i_7}^{(1,2,3)}(a_1, a_2, a_3) T_{i_8}^{(4,5,6)}(a_4, a_5, a_6), \]

\[ = \sum_{i_7, i_8=1}^{r} A_{i_7i_8}^{(1)} \sum_{i_1, i_9=1}^{r} A_{i_7i_1i_9}^{(2)} T_{i_1}^{(1)}(a_1) T_{i_9}^{(2,3)}(a_2, a_3) \sum_{i_4, i_{10}=1}^{r} A_{i_8i_4i_{10}}^{(3)} T_{i_4}^{(4)}(a_4) T_{i_{10}}^{(5,6)}(a_5, a_6), \]

\[ = \sum_{i_1, ..., i_6=1}^{r} C_{i_1, ..., i_6} T_{i_1}^{(1)}(a_1) T_{i_2}^{(2)}(a_2) T_{i_3}^{(3)}(a_3) T_{i_4}^{(4)}(a_4) T_{i_5}^{(5)}(a_5) T_{i_6}^{(6)}(a_6), \]  

(228)

\( ^{17} \)The second index labels points on the plane \( (a_2, a_3) \) (collocation setting), or the projection of \( f \) onto the two-dimensional basis function \( T_{j}^{(2,3)}(a_2, a_3) \) (Galerkin setting).
where the 6-dimensional core tensor is explicitly given
\[
C_{i_1...i_6} = \sum_{i_7, i_8=1}^{r} A_{i_7 i_8}^{(1)} \sum_{i_9=1}^{r} A_{i_7 i_8 i_9}^{(2)} A_{i_9 i_{10}}^{(4)} \sum_{i_{10}=1}^{r} A_{i_{10} i_{11} i_{12} i_{13}}^{(3)} A_{i_{11} i_{12} i_{13} i_{14} i_{15} i_{16}}^{(5)}.
\] (229)

Note that such kernel has a recursive structure and it is factored as a product of at most three-dimensional matrices. Parallel algorithms to perform basic operations on hierarchical Tucker expansions such as addition, multiplication, and rank reduction were recently proposed in [77, 56]. Also, an optimization framework that leverages on the on recursive subspace factorizations of Hierarchical Tucker expansions was developed in [206].

### 3.3.4 Tensor Train Decomposition

If we single-out one variable at a time and perform a sequential Schmidt decomposition of the remaining variables we obtain the so-called tensor-train decomposition [168, 191]. Tensor train decomposition is a subcase of hierarchical Tucker decomposition. For example, the tensor train decomposition of the four-dimensional cylindrical functional \( f(a_1, ..., a_4) \) reads

\[
f(a_1, ..., a_4) = \sum_{i_1, i_2=1}^{r} A_{i_1 i_2}^{(1)} T_{i_1}^{(1)}(a_1) T_{i_2}^{(2, 3, 4)}(a_2, a_3, a_4),
\]

\[
= \sum_{i_3, i_4=1}^{r} A_{i_3 i_4}^{(2)} T_{i_3}^{(2)}(a_2) T_{i_4}^{(3, 4)}(a_3, a_4),
\]

\[
= \sum_{i_5, i_6=1}^{r} A_{i_5 i_6}^{(3)} T_{i_5}^{(3)}(a_3) T_{i_6}^{(4)}(a_4),
\]

\[
= \sum_{i_1, ..., i_6=1}^{r} A_{i_1 i_2}^{(1)} A_{i_2 i_3 i_4}^{(2)} A_{i_4 i_5 i_6}^{(3)} T_{i_1}^{(1)}(a_1) T_{i_3}^{(2)}(a_2) T_{i_5}^{(3)}(a_3) T_{i_6}^{(4)}(a_4). \tag{230}
\]

Tensor train and hierarchical Tucker expansions can be conveniently visualized by graphs.

### 3.3.5 Tensor Networks

During the last decades, the field of Tensor Networks has undergone an explosion of results in several different directions, e.g., in the study of quantum many-body systems, and more generally in multivariate functional approximation. Tensor networks can be conveniently represented by undirected graphs. To this end, we adopt the following standard convention:

- a node in a graph represents a tensor in as many variables as the number of the edges connected to it;
- connecting two tensors by an edge represents a tensor contraction over the index associated to certain variable.

In this way, a three-dimensional Tucker format is represented by one node with three edges emanating from it. If we connect two of such tensors with one edge, then we obtain a tensor in four variables. In particular, the connection operation here results in a Tucker format with the core tensor represented by a product of two three-dimensional tensors in which we contract one index (see Figure 9). The graph representation of the hierarchical Tucker and tensor train decomposition is shown in Figure 10. A more general tensor network representation of a cylindrical functional in five variables is shown in Figure 11. Note that each
Figure 9: Examples of tensor networks representing a cylindrical functional in three and four variables.

Figure 10: Graph representation of the Hierarchical Tucker (HT) and tensor train (TT) decomposition of a six-dimensional cylindrical functional.

node is connected with at most three edges (coordination number equal to three) as we employed a Schmidt decomposition to represent each function with more than one variable. Clearly, the graph can be reduced to one node with five leaves (i.e. a Tucker series) by eliminating the cycle and clustering all nodes that are connected by edges \[114\]. The graph can be therefore reduced to a node with five leaves – i.e. a Tucker series, with a particular structure of the core tensor. At this point the graph is irreducible (one node with five leaves). Efficient algorithms that implement basic operations between tensors, such as addition, orthogonalization, rank reduction, scalar products, multiplication, and linear transformations are discussed in \[114, 56, 76, 159, 113\].

3.4 Generalized Lagrangian Interpolation

We have seen that the classical Lagrange interpolation problem for multivariate functions can be generalized to functionals in Hilbert or Banach spaces, i.e., objects depending on an infinite number of variables. Given a real-valued continuous functional

\[ F : D(F) \to \mathbb{R}, \quad (231) \]

and a set of elements \( \{ \theta_1(x), ..., \theta_n(x) \} \) in \( D(F) \), the Lagrangian interpolant of \( F \) is a continuous functional

\[ \Pi : D(F) \to \mathbb{R}, \quad (232) \]

such that

\[ \Pi([\theta_i]) = F([\theta_i]). \quad (233) \]
Figure 11: Tensor network representation of a five-dimensional cylindrical functional. After reduction (elimination of the cycle and connection of nodes), it is easy to see the graph is equivalent to the graph corresponding to a Tucker format of dimension five (right). The core tensor of such Tucker format is expressed as products and contractions of at most three-dimensional tensors.

A simple representation of $\Pi([\theta])$ can be given in terms of cardinal basis functionals $g_i : D(F) \to \mathbb{R}$ as

$$\Pi([\theta]) = \sum_{i=1}^{n} F([\theta_i])g_i([\theta]), \quad \text{where} \quad g_i([\theta_j]) = \delta_{ij}. \quad (234)$$

Similarly to classical Lagrange interpolation for functions from $\mathbb{R}^n$ to $\mathbb{R}^m$, we have a great freedom in selecting $g_i([\theta])$. We have already seen, for example, Porter’s and Prenter’s cardinal bases defined in equation (113) and equation (150), respectively. Now, let

$$\kappa : D(F) \times D(F) \to \mathbb{R} \quad (235)$$

be any continuous (not necessarily symmetric) functional subject to the sole requirement that

$$\kappa([\theta],[\eta]) = 0 \quad \iff \quad \theta(x) = \eta(x). \quad (236)$$

Examples of such functional are

$$\kappa_1([\theta],[\eta]) = \|\theta(x) - \eta(x)\|^p, \quad p > 0 \quad (237)$$

$$\kappa_2([\theta],[\eta]) = \|\theta(x) - \eta(x)\| + \|\theta(x)\| - \|\eta(x)\|. \quad (238)$$

$$\kappa_3([\theta],[\eta]) = 1 - \exp \left( -\|\theta(x) - \eta(x)\|^2 \right). \quad (239)$$

By using $\kappa([\theta],[\eta])$ we define the following cardinal basis associated with the interpolation nodes $\{\theta_1(x), ..., \theta_n(x)\}$ in $D(F)$ (see [30], Ch. 10)

$$g_i([\theta]) = \prod_{j=1, j \neq i}^{n} \frac{\kappa([\theta_i],[\theta_j])}{\kappa([\theta],[\theta_j])} \quad i = 1, ..., n. \quad (240)$$

A substitution of (240) into (234) yields a Lagrangian functional interpolant that depends on the choice of $\kappa([\theta],[\eta])$. An advantage of such interpolant over Porter’s interpolant is that the basis functionals (240) are given analytically and do not require any computation such as the inversion (and storage) of the $H$-matrix (111). On the other hand, we have no guarantee that (240) have good approximation properties away from
the interpolation nodes. The Lagrange interpolation formula (234)-(240) can be written in a barycentric form. To this end, we first rewrite the numerator of (240) as \( \Omega([\theta])/\kappa([\theta], [\theta_i]) \), where

\[
\Omega([\theta]) = \prod_{i=k}^{n} \kappa([\theta], [\theta_k]).
\] (241)

Then define the barycentric weights as

\[
w_i = \prod_{j=1, j \neq i}^{n} \frac{1}{\kappa([\theta_i], [\theta_j])}.
\] (242)

Clearly,

\[
g_i([\theta]) = \frac{\Omega([\theta])}{\kappa([\theta], [\theta_i])} w_i.
\] (243)

Substituting this into (234) and applying it to the constant functional \( F([\theta]) = 1 \) yields

\[
\Omega([\theta]) = \frac{1}{\sum_{i=1}^{n} w_i \kappa([\theta], [\theta_i])}.
\] (244)

Therefore, we obtain the interpolant

\[
\Pi([\theta]) = \left( \sum_{i=1}^{n} \frac{w_i}{\kappa([\theta], [\theta_i])} \right)^{-1} \sum_{i=1}^{n} \frac{w_i F([\theta_i])}{\kappa([\theta], [\theta_i])}.
\] (245)

This expression can be evaluated in \( O(n) \) operations, provided we have available \( w_i \), which needs \( O(n^2) \) operations. What about updating? Adding a new interpolation node \( \theta_{n+1} \) entails two calculations: 1) divide each \( w_j \) by \( \kappa([\theta_j], [\theta_{n+1}]) \) and 2) compute \( w_{n+1} \) using the formula (242).

**Remark:** The interpolation processes defined by (234) and (240) has a variation known as Shepard’s method [60, 121]. Such method was studied by Allasia and Bracco [4] for functional interpolation in Banach spaces and it relies on the cardinal functionals

\[
g_i([\theta]) = \frac{1}{1 + \sum_{k=1, k \neq i}^{n} \kappa([\theta], [\theta_k])}, \quad i = 1, \ldots, n.
\] (246)

The kernel \( \kappa([\theta], [\eta]) \) here satisfies (236) and the additional positivity requirement

\[
\kappa([\theta], [\eta]) > 0 \quad \text{if} \quad \theta(x) \neq \eta(x).
\] (247)

This implies that the basis functionals (246) have the following properties

\[
0 \leq g_i([\theta]) \leq 1 \quad \forall \theta \quad g_i([\theta_j]) = \delta_{ij}.
\] (248)

One unfortunate consequence of these conditions is that the functional derivative of \( g_i([\theta]) \) (if it exists) is identically zero at all nodes \( \theta_j \). In fact, \( g_i([\theta]) \) is always positive, it has a maximum (equal to 1) at \( \theta_i \) and many minima (equal to 0) at all other nodes. This yields functional differentiation matrices (120) that are
identically zero. Functional Lagrangian interpolation can also be defined in terms of radial basis functionals 
\( \kappa(\|\theta - \theta_j\|) \), for suitable choices of \( \kappa \). In this case, the cardinal basis \( g_i([\theta]) \) can be obtained by standard techniques such as the ratio of two Vandermonde determinants. Interpolation using radial basis functionals can converge towards polynomial functional interpolants if we take increasingly flat radial bases (see, e.g., [202, 47, 208]).

The Lagrangian interpolation process defined by (234), (236) and (240) can be generalized further by con-
sidering a family of continuous functionals \( \kappa_i : D(F) \times D(F) \rightarrow \mathbb{R} \), \( i = 1, \ldots, n \) as many as the number of interpolation nodes, satisfying \( \kappa_i([\theta], [\theta_j]) \neq 0 \) for \( i \neq j \), \( \kappa_i([\theta], [\theta_i]) = 0 \) and \( \kappa_i([\theta], [\theta]) = 0 \). The cardinal basis corresponding to \( \{\kappa_1, \ldots, \kappa_n\} \) is

\[
g_i([\theta]) = \prod_{j=1, j \neq i}^{n} \frac{\kappa_i([\theta], [\theta_j])}{\kappa_i([\theta_i], [\theta_j])}.
\]

In particular, the choice

\[
\kappa_i([\theta], [\eta]) = (\theta - \eta, \theta_i - \eta)
\]

yields Prenter’s cardinal basis (150). It fact,

\[
\kappa_i([\theta], [\theta_j]) = (\theta - \theta_j, \theta_i - \theta_j),
\]

\[
\kappa_i([\theta_i], [\theta_j]) = \|\theta_i - \theta_j\|^2,
\]

\[
\kappa_i([\theta], [\theta_i]) = 0.
\]

Another generalization of the Lagrange interpolation process can be obtained along the lines of Porter’s functional interpolants. To this end, let \( \kappa([\theta], [\eta]) \) be any symmetric functional such that the matrix

\[
H_{ij} = \kappa([\theta_i], [\theta_j])
\]

is invertible for every set of distinct nodes \( \{\theta_j(x)\} \). Then the functional

\[
\Phi_n([\theta]) = \sum_{k=1}^{m} F(\theta_k) \sum_{j=1}^{m} \kappa([\theta_k], [\theta_j])^{-1} \kappa([\theta], [\theta_j]),
\]

where \( \kappa([\theta_k], [\theta_j])^{-1} \) denotes the inverse matrix of (255), interpolates \( F([\theta]) \) at \( \theta_k(x) \) \( (k = 1, \ldots, m) \). The question of how to select the symmetric functional \( \kappa \) depends on the requirements we impose on the behavior of the functional interpolant away from the interpolation nodes. In particular, the choice

\[
\kappa([\theta], [\eta]) = \sum_{p=0}^{n} (\theta, \eta)^p
\]

yields Porter’s interpolants, i.e., polynomial functionals of order \( n \) with minimal norm.

### 3.4.1 Optimal Interpolation Nodes

In this Section we briefly address the question of how to select the interpolation nodes in the function space
\( D(F) \) in such a way that the corresponding functional interpolant, e.g., the Porter’s one (112)-(113), exhibits good approximation properties.
Adaptive Leja Sequences  An effective approach to select sub-optimal interpolation nodes relies on adaptive Leja sequences [151][23]. In this setting, given the set of nodes \( \{ \theta_1(x), \ldots, \theta_n(x) \} \) in \( D(F) \), the problem is to find a new node \( \theta^*(x) \) satisfying the maximization principle

\[
\arg\max_{\theta^* \in D(F)} |\det(V)|
\]  

(258)

where \( V \) is a Vandermonde-like operator (infinite-dimensional matrix) (see [23][151][153][155] for further details). This results in the so-called Leja construction of the test function \( \theta^*(x) \), a greedy version of the well-known Fekete construction involving nonlinear optimization. There are also non-determinant versions of multivariate Leja interpolation nodes, which are related to potential theory [128][93]. In this setting, given a symmetric functional \( \kappa([\theta(x)], [\eta(x)]) \) one can construct a greedy \( \kappa \)-energy sequence in which \( \theta^*(x) \) is chosen to satisfy

\[
\sum_{i=1}^n \kappa([\theta^*(x)], [\theta_i(x)]) = \arg\min_{\theta^* \in D(F)} \sum_{i=1}^n \kappa([\theta^*(x)], [\theta_i(x)]),
\]

(259)

assuming \( \{ \theta_1(x), \ldots, \theta_n(x) \} \) are available. The element \( \theta^*(x) \) obtained in this way is not unique. Also, depending on the choice of the kernel function \( \kappa([\theta(x)], [\eta(x)]) \), we can have different sets of nodes. A relevant kernel for the \( \kappa \)-energy sequence is the so-called Riesz \( p \)-kernel

\[
\kappa_p([\theta(x)], [\eta(x)]) = \begin{cases} 
\frac{1}{||\theta(x) - \eta(x)||^p} & \text{if } p > 0 \\
-\ln(||\theta(x) - \eta(x)||) & \text{if } p = 0 
\end{cases}
\]

(260)

Such kernel avoids placing points too close to each other (the potential \( \kappa_p \) is rather large in such regions), while decreasing monotonically at larger distances. The asymptotic behavior of greedy \( \kappa \)-energy sequences has been studied theoretically by López-Garcia and Saff [128], in a finite-dimensional setting.

Minimization of Lebesgue Functionals  Another approach to determine sub-optimal nodes in the function space \( D(F) \) relies on greedy algorithms minimizing Lebesgue-like functionals. This problem has been recently addressed in a finite-dimensional setting by Narayan and Xiu [153][152], Maday et al. [131][132] and Van Barel et al. [10]. The basic idea is simple and can be generalized to the infinite-dimensional case. Suppose we have available \( \{ \theta_1(x), \ldots, \theta_n(x) \} \). We look for a new node \( \theta^*(x) \) maximizing a suitable objective functional, e.g., related to the well-known Lebesgue function in finite-dimensional interpolation problems. In particular, following the weighted approach of Narayan and Xiu [153], we can look for a new node \( \theta^*(x) \) satisfying the following (greedy) optimization problem

\[
\arg\max_{\theta^* \in D(F)} \chi([\theta^*]), \quad \chi([\theta]) = W([\theta])^2 \sum_{k=1}^n \frac{g_k([\theta])^2}{W([\theta_k])^2},
\]

(261)

where \( W([\theta]) \) is a positive functional and \( g_k([\theta]) \) is a cardinal basis. Other choices of \( \chi([\theta]) \) yield different sets of nodes. For example,

\[
\chi([\theta]) = W([\theta]) \sum_{k=1}^n \frac{g_k([\theta])^2}{W([\theta_k])}
\]

(262)

yields an infinite-dimensional extension of the Fejér points. The sequence of nodes \( \theta_i(x) \) we obtain in this way strongly depends on the initial set of nodes, on the weight \( W([\theta]) \) and on the cardinal basis \( \{ g_k([\theta]) \} \). Finding the maximum of (261) involves computing the solution to an optimization problem in infinite dimensions [14]. A necessary condition for a stationary point of (261) is

\[
\frac{\delta \chi([\theta])}{\delta \theta(x)} = 0.
\]

(263)
Example 1: The Euler-Lagrange equation (263) can be written down explicitly, e.g., for Porter’s interpolants. To this end, simply substitute (113) in (262) and set $W = 1$ to obtain

$$\chi([\theta]) = \sum_{k=1}^{n} \sum_{j,z=1}^{m} H_{jk}^{-1} H_{zk}^{-1} \sum_{p,q=0}^{h} (\theta_j, \theta)^p (\theta_z, \theta)^q.$$  \hspace{1cm} (264)

This allows us to write explicitly the conditions identifying stationary points of $\chi([\theta])$, e.g., in the function space

$$D_N = \left\{ \theta(x) \in D(F) \mid \theta(x) = \sum_{k=1}^{N} a_k \varphi_k(x) \right\}.$$  \hspace{1cm} (265)

3.5 High-Dimensional Model Representation

The high-dimensional model representation (HDMR) [185, 126, 125, 26] of the cylindrical functional (177) is a series expansion in the form

$$f(a_1, \ldots, a_m) = f_0 + \sum_{k=1}^{m} f_k(a_k) + \sum_{k,j=1}^{m} f_{kj}(a_k, a_j) + \cdots.$$  \hspace{1cm} (266)

The functional $f_0$ is simply a constant. The functionals $f_k(a_i)$, which we shall call first-order interaction terms, give us the overall effects of the variables $a_i = (\theta, \varphi_i)$ in $f$ as if they were acting independently of the other variables. The functions $f_{ij}(a_i, a_j)$ represent the interaction effects of $a_i$ and $a_j$, and therefore they are usually called second-order interactions. Similarly, higher-order terms reflect the cooperative effects of an increasing number of variables. The interaction terms in the HDMR expansion can be easily computed if we assume that the domain of $f$ is compactly supported and included in the hypercube $[-b, b]^m$ (see Appendix B). By introducing the finite-dimensional integration measure $w(a) = w(a_1, \ldots, a_m)$ and the vector notation $\mathbf{d}a_{-i} = da_{i-1} da_{i+1} \cdots da_m$, we have

$$f_0 = \int_{[-b, b]^m} w(a) f(a) da,$$  \hspace{1cm} (267)

$$f_i(a_i) = \int_{[-b, b]^{m-1}} w(a) f(a) da_{-i} - f_0,$$  \hspace{1cm} (268)

$$f_{ij}(a_i, a_j) = \int_{[-b, b]^{m-2}} w(a) f(a) da_{-ij} - f_i(a_i) - f_j(a_j) - f_0,$$  \hspace{1cm} (269)

$$\cdots.$$  

This procedure generates, by construction, terms that are orthogonal in the weighted $L_2$ sense. The HDMR series (266) with components (267)-(269) is often called ANOVA-HDMR expansion. If we consider a Dirac delta measure $w(a) = \delta(a_1 - c_1) \cdots \delta(a_m - c_m)$ with “anchor point” $c = (c_1, \ldots, c_m)$, then the HDMR series is called CUT-HDMR [39, 249]. There is also a random-sampling version of HDMR – namely the RS-HDMR [125] expansion – in which the high-dimensional integrals in (267)-(268) are computed by Monte-Carlo or quasi-Monte Carlo methods. The HDMR expansion (266) is usually truncated at some interaction order, and the interactions $f_i, f_{ij}$ are expanded relative to a certain basis. For example, the first-order ANOVA-HDMR expansion of $f$ reads

$$f(a_1, \ldots, a_m) \simeq f_0 + \sum_{k=1}^{m} f_k(a_k),$$  \hspace{1cm} (270)
where
\[ f_k(a_k) = \sum_{i=1}^{Q} \alpha_{ik} \phi_i(a_k), \quad (271) \]
and \( \phi_i \) are suitable basis functions in the variable \( a_k - (\theta, \varphi_k) \).

### 3.6 Cluster Expansion

The cluster expansion is a functional approximation method for Hopf characteristic functionals that leverages on the structure of the functional itself to provide an approximation that preserves important properties such as normalization and marginalization rules. To illustrate the method, let us consider the restriction of the Hopf functional to the finite-dimensional function space \( D_m \subseteq D(\Phi) \) (see Example 1 in Section 3.1 and Figure 1).

\[ \phi(a_1, \ldots, a_m) = \left\langle e^{i(a_1 U_1(\omega) + \cdots + a_m U_m(\omega))} \right\rangle, \quad U_k(\omega) = \int_a^b u(x; \omega) \varphi_k(x) dx. \quad (272) \]

To represent the high-dimensional (complex-valued) function \( \phi(a_1, \ldots, a_m) \) one can use techniques such as HDMR (Section 3.5), functional tensor methods (Section 3.3), or sparse collocation [24, 44, 160]. One of the problems with such techniques is that they do not preserve important properties of the characteristic function, normalization \( \phi(0, \ldots, 0) = 1 \), and marginalization rules. However, in the case of (272), we do know the structure of cylindrical functional we are approximating. Therefore, we can leverage on such structure to build an expansion that preserves marginalization rules and other properties. To this end, let us first define the following reduced-order characteristic functions

\[ \phi_n(a_n) = \left\langle e^{ia_n U_n(\omega)} \right\rangle, \quad (273) \]
\[ \phi_{nm}(a_n, a_m) = \left\langle e^{ia_n U_n(\omega) + ia_m U_m(\omega)} \right\rangle, \quad (274) \]
\[ \phi_{nmk}(a_n, a_m, a_k) = \left\langle e^{ia_n U_n(\omega) + ia_m U_m(\omega) + ia_k U_k(\omega)} \right\rangle, \quad (275) \]
\[ \cdots. \]

Clearly, we have
\[ \phi_{nmk}(a_n, 0, 0) = \phi_{nm}(a_n, 0) = \phi_n(a_n), \quad \text{(marginalization rule).} \quad (276) \]

By using well-known cumulant series representations [116], we expand (272) as

\[ \left\langle \exp \left[ \sum_{j=1}^{N} a_j U_j(\omega) \right] \right\rangle = \exp \left[ \sum_{\nu_1, \ldots, \nu_N = 0}^{\infty} \frac{\langle U_1^{\nu_1} \cdots U_N^{\nu_N} \rangle}{\nu_1! \cdots \nu_N!} \right], \quad (277) \]

where the last summation is over \( \nu_1, \ldots, \nu_N \) excluding the case in which all indices are zero, i.e., excluding \( \nu_1 = \cdots = \nu_N = 0 \). The cumulant averages in (277) can be written in terms of moments of \( U_j \). For example, we have

\[ \langle U_j \rangle_e = \langle U_j \rangle, \]
\[ \langle U_j U_k \rangle_e = \langle U_j U_k \rangle - \langle U_j \rangle \langle U_k \rangle, \]
\[ \langle U_j U_k U_m \rangle_e = \langle U_j U_k U_m \rangle - \langle U_j \rangle \langle U_k U_m \rangle - \langle U_k \rangle \langle U_j U_m \rangle - \langle U_m \rangle \langle U_k U_j \rangle + 2 \langle U_j \rangle \langle U_k \rangle \langle U_m \rangle. \]
At this point, it is useful to write the (277) explicitly for simple cases, e.g., for (274) and (275). We have,

\[ \phi_{nm}(a_n, a_m) = \phi_n(a_n)\phi_m(a_m) \exp \left[ \sum_{j,k=1}^{\infty} \langle U_m^j U_n^k \rangle / j!k! \right] \quad \text{and} \quad (278) \]

\[ \phi_{nmk}(a_n, a_m, a_k) = \phi_n(a_n)\phi_m(a_m)\phi_k(a_k) \exp \left[ \sum_{j,z,q=1}^{\infty} \langle U_m^j U_n^z U_k^q \rangle / j!z!q! \right]. \quad (279) \]

By generalizing these results to \( N \) variables, we find the following \textit{cluster expansion}

\[ \phi(a_1, \ldots, a_N) = \prod_{n=1}^N \phi_n(a_n) \prod_{n<m}^{N} \frac{\phi_{nm}(a_n, a_m)}{\phi_n(a_n)\phi_m(a_m)} \times \prod_{n<m<k}^{N} \frac{\phi_{nmk}(a_n, a_m, a_k)}{\phi_{nm}(a_n, a_m)\phi_k(a_k)\phi_{nk}(a_n, a_k)} \times \cdots. \quad (280) \]

Any truncation of (280) to a certain order in the multi-point characteristic functions \( \phi_{lmn\ldots} \) yields approximations. In particular, the second-order truncation

\[ \phi(a_1, \ldots, a_N) \simeq \prod_{n=1}^N \phi_n(a_n) \prod_{n<m}^{N} \frac{\phi_{nm}(a_n, a_m)}{\phi_n(a_n)\phi_m(a_m)} \quad (281) \]

involves \( N(N-1)/2 \) functions \( \phi_n \) (\( \phi_1 \) can be determined from \( \phi_{lm} \)), and it can be defined as \textit{Gaussian approximation}. The reason for such definition is that if \( U_1, \ldots, U_N \) are jointly Gaussian, then (281) is exact (the \( n \)-th order cumulants of a multivariate Gaussian are zero if \( n \geq 3 \)). We can establish an interesting connection between the networks of test functions we discussed in Section 3.2.1 and the truncation order in (280).

\textbf{Representation in} \( S_1^{(m)} \) The representation of the Hopf functional in \( S_1^{(m)} \) (see Eq. (89)) is defined in terms of one-dimensional functions

\[ \Phi([a_j\varphi_j(x)]), \quad j = 1, \ldots, m. \quad (282) \]

The corresponding approximation takes the form

\[ \Phi([a_1\varphi_1(x) + \cdots + a_m\varphi_m(x)]) \simeq \prod_{j=1}^m \Phi([a_j\varphi_j(x)]). \quad (283) \]

In statistical physics this is known as \textit{mean-field approximation}, and it relies on a statistical independence hypothesis between the random variables \( U_k(\omega) \) in (272).

---

\( ^{18} \)The rationale behind this approximation relies on the fact that the joint cumulants of \( U_k \) often decay with the order, and therefore the exponential function (see (279)) tends to 1 quickly.
**Representation in $S_2^{(m)}$** The representation of the Hopf functional in $S_2^{(m)}$ involves two-dimensional functions in the form

$$
\Phi([a_j\varphi_j(x) + a_k\varphi_k(x)]), \quad k, j = 1, \ldots, m.
$$

With this set we consider the second-order truncation of (280), i.e.,

$$
\Phi([a_1\varphi_1(x) + \cdots + a_m\varphi_m(x)]) \approx \prod_{j=1}^{m} \Phi([a_j\varphi_j(x)]) \prod_{j<k} \frac{\Phi([a_j\varphi_j(x) + a_k\varphi_k(x)])}{\Phi([a_j\varphi_j(x)]\Phi([a_k\varphi_k(x)])

= \prod_{j<k} \Phi([a_j\varphi_j(x) + a_k\varphi_k(x)]) \prod_{j=1}^{m} \Phi([a_j\varphi_j(x)])^{2-m}. 
$$

Clearly, the one-dimensional functions $\Phi([a_j\varphi_j(x)])$ can be obtained from (284) by setting $a_k = 0$.

### 3.7 Functional Approximation Based on Random Processes

So far we discussed representations of nonlinear functionals based on power series, Lagrangian interpolants, and tensor methods. In this Section we briefly discuss another way of constructing polynomial functional expansions by using stochastic processes. The method was pioneered by N. Wiener in [239]. Suppose we are given a random function $u(x; \omega)$, with know statistical properties and a real-valued nonlinear functional $F([\theta])$. If $u(x; \omega)$ is in the domain of $F$ then we can evaluate $F([u(x; \omega)])$, which is a real-valued random variable. The set

$$
\{u(x; \omega), F([u(x; \omega)])\}
$$

can be considered as an infinite collection of input-output signals from which we would like to determine a polynomial approximation of $F$. The key point is that the stochastic signal $u(x; \omega)$ is equivalent to an infinite collection of functions that span the domain of $F$, hopefully in a way that is sufficient to identify $F([\theta])$. To this end, we look for an expansion in the form

$$
F([\theta]) = \sum_{k=0}^{\infty} G_k([\theta]),
$$

where $G_k([\theta])$ $k = 0, 1, \ldots$ is a complete set of orthogonal polynomial functionals [239, 204, 164, 53]. Orthogonality here is relative to the probability measure $P[u]$ of the random process $u(x; \omega)$, i.e., relative to the inner product

$$
(G_k([u]), G_j([u]))_{dP[u]} = \int_{\Omega} G_k([u]) G_j([u]) dP[u]. 
$$

It was shown in [239] that if $u(x; \omega)$ is Brownian motion, then $G_k$ are Hermite polynomial functionals, and (287) becomes the celebrated Wiener-Hermite expansion [239, 25, 242]. For completeness, we recall that the first- and the second-order Hermite polynomial functionals are defined as (see [239], p. 32)

$$
G_1([u]) = \int K_1(x_1) du(x_1; \omega),
$$

$$
G_2([u]) = \int \int K_2(x_1, x_2) du(x_1; \omega) du(x_2; \omega) - \int K_2(x_1, x_1) dx_1,
$$

where $K_2(x_1, x_2)$ is subject to the normalization condition

$$
\int K_2(x_1, x_2)^2 dx_1 dx_2 = \frac{1}{2}.
$$

53
Clearly, if \( u(x; \omega) \) is Brownian motion then the integrals in (289)-(290) do not exist in the ordinary Stieltjes sense because \( u(x; \omega) \) is nowhere differentiable. However, we can interpret the derivative of \( u(x; \omega) \) in a distributional sense to obtain a generalized process, i.e., the white noise. Wiener has shown that in such a generalized setting, the integral (289)-(290) are perfectly well defined for kernel functions \( K_1(x_1) \) and \( K_2(x_1, x_2) \) in \( L_2 \). A detailed mathematical analysis of the Wiener-Hermite expansion can be found in \([53, 25, 239, 170]\).

The process of determining the kernels of \( G_1(\theta) \), \( G_2(\theta) \), etc., is known as identification process in the theory of nonlinear systems, and it has been studied extensively for obvious reasons (see \([155, 199, 205]\)). By leveraging on the orthogonality of \( G_k \) relative to the inner product (288) one can show that (see, e.g., Eq. (4.4) in [229])

\[
K_p(x_1, \ldots, x_p) = \frac{1}{p!\epsilon^p} \left( F([u]), G_p(H_p, [u]) \right)_{dP[u]}, \quad H_p(z_1, \ldots, z_p) = \begin{cases} 
1 & x_i \leq z_i \leq x_i + \epsilon \\
0 & \text{otherwise}
\end{cases} . \quad (292)
\]

This result relies on the fact that orthogonality of \( G_k \) and \( G_j \) in the sense of (288) is not dependent on the actual kernel functions appearing in \( G_j \) and \( G_k \). In fact, if we consider the inner product of (287) with \( G_p(H_p, [u]) \) - arbitrary kernel \( H_p(x_1, \ldots, x_n) \) - then the only term that survives at the right hand side is \( (G_p(K_p, [u])G_p(H_p, [u]))_{dP[u]} \). Such inner product can be written as

\[
(G_p(K_p, [u])G_p(H_p, [u]))_{dP[u]} = p! \int_a^b \cdots \int_a^b K_p(x_1, \ldots, x_p)H_p(x_1, \ldots, x_p)dx_1 \cdots dx_p, \quad (293)
\]

i.e., all contributions of lower-order terms are identically zero. Thus, if we select \( H_p \) properly (e.g., \( H_p(x_1, \ldots, x_p) = 1 \) in a small hypercube centered at \((x_1, \ldots, x_p)\) with side length \( \epsilon \) and zero otherwise) then we can extract \( K_p(x_1, \ldots, x_p) \) as in equation (292).

This means that if we know the response \( F([u]) \) corresponding to the stochastic process \( u(x; \omega) \), then we can identify the kernels (292) and therefore construct the corresponding polynomial functional expansion (287). The path integrals in (292) can be evaluated numerically, e.g., by using Monte-Carlo or quasi-Monte Carlo methods (see Appendix B). To this end, suppose we have available a collection of \( N_x \) of response excitation signals \( \{u(x; \omega_i), F([u(x; \omega_i)])\} (i = 1, \ldots, N) \), where \( u(x; \omega_i) \) is a realization of the process \( u(x; \omega) \) obtained by sampling the probability functional \( P[u] \). The Monte-Carlo estimate of the kernels (292) is simply

\[
K_p(x_1, \ldots, x_p) \approx \frac{1}{p!\epsilon^p N_x} \sum_{i=1}^{N} F(\xi(x, \omega_i))G_k(H_p, [\xi(x, \omega_i)]) \quad p = 0, 1, \ldots \quad (294)
\]

where \( \xi(x, \omega_i) \) is a realization of the distributional derivative of \( u(x; \omega_i) \). In particular, if \( u(x; \omega) \) is Brownian motion then \( \xi(x; \omega) \) is spatial white noise\(^{19} \). In this case, we have that the first two functionals \( G_1(H_1, [\xi]) \)

\(^{19} \)A simple numerical approximation of white noise \( \xi(x; \omega) = du(x; \omega)/dx \) can be obtained as

\[
\xi_{\Delta x}(x; \omega) = \frac{u(x + \Delta x; \omega) - u(x; \omega)}{\Delta x}, \quad (u \text{ Brownian motion}). \quad (295)
\]

This is a Gaussian process with zero mean, variance \( 1/\Delta x \) and covariance

\[
\langle \xi_{\Delta x}(x; \omega)\xi_{\Delta x}(y; \omega) \rangle = \begin{cases} 
\frac{1}{\Delta x} \left( 1 - \frac{|x - y|}{\Delta x} \right) & \text{if } |x - y| \leq \Delta x \\
0 & \text{otherwise}
\end{cases} \quad (296)
\]

The Fourier transform of \( \langle \xi_{\Delta x}(x; \omega)\xi_{\Delta x}(y; \omega) \rangle \) yields a power spectrum which is not flat as it is supposed to be for white noise, but decays at sufficiently large frequencies.
and \( G_2(H_2, \xi) \) in (294) are given by
\[
G_1(H_1, [\xi]) = \epsilon \xi(x_1; \omega),
\]
\[
G_2(H_2, [\xi]) = \epsilon^2 (\xi(x_1; \omega)\xi(x_2; \omega) - 1),
\]
and therefore
\[
K_1(x_1) \approx \frac{1}{N} \sum_{i=1}^{N} F([\xi(i)]) h_1(\xi(i)(x_1; \omega)),
\]
\[
K_2(x_1, x_2) \approx \frac{1}{N} \sum_{i=1}^{N} F([\xi(i)]) h_2(\xi(i)(x_1; \omega), \xi(i)(x_2; \omega)).
\]

Here \( h_1(x_1) = x_1 \) and \( h_2(x_1, x_2) = (x_1 x_2 - 1)/2 \) are the classical first- and second-order Hermite polynomials. A very insightful discussion on how to compute the kernels \( K_j \) by using white noise is given by Rugh in [199], §7.4. The expansion (287) obtained in this way can, in principle, be used to compute the value of the functional \( F([\theta]) \) corresponding to any deterministic function \( \theta(x) \). However, since the kernels (292) are built upon stochastic processes and their averages, it is not clear in what sense (287) will converge for deterministic input functions \( \theta(x) \). This question was addressed by Palm and Poggio in [170] (Theorems 4 and 5), where necessary and sufficient conditions for pointwise convergence of Wiener-Hermite expansions are provided (see also [53]). The functional expansion in terms of orthogonal polynomial functionals can be generalized to random processes other than Brownian motion and white noise\(^{20}\). However, one has to be very careful when performing such generalizations. There are indeed random processes that do not allow for a complete representation of the functional \( F([\theta]) \) (see [53, 204, 164] for details).

### 4 Functional Differential Equations

A functional differential equation (FDE) is an equation involving a nonlinear functional (i.e., a nonlinear operator), derivatives with respect to functions (functional derivatives) and derivatives with respect to independent variables, e.g., space and time variables. In this report we will study linear FDEs in the form
\[
\frac{\partial F([\theta], t)}{\partial t} = L([\theta], t) F([\theta], t) + H([\theta], t) \quad F([\theta], 0) = F_0([\theta]),
\]
where \( F_0([\theta]) \) is an initial condition, \( H([\theta], t) \) is a known forcing term and \( L([\theta], t) \) is a linear operator in the space of functionals. This class of FDEs is very broad, and it includes many well-known functional equations of theoretical and quantum physics. The solution to initial value problem (300) (assuming it exists) depends on the initial condition \( F_0 \) as well as on the choice of the function space \( D(F) \) (domain of the functional). The following examples clarifies this point.

**Example 1 (Functional Advection Equation):** Consider the following FDE
\[
\frac{\partial F([\theta], t)}{\partial t} + \int_0^{2\pi} \theta(x) \frac{\partial}{\partial x} \frac{\delta F([\theta], t)}{\delta \theta(x)} dx = 0, \quad F([\theta], 0) = F_0([\theta]),
\]
\(^{20}\) The construction of such generalized expansion proceeds as follows. Starting from the constant functional \( F_0 \) we look for
\[
F_i([u]) = C_1 \int K_1(x) du(x; \omega) + F_0,
\]
where \( u(x; \omega) \) is a generalized random process, and we make it orthogonal to \( F_0 \) in the sense of \( dP[u(x; \omega)] \). This yields \( C_1 \). Then we construct \( F_2([u]) \) and we make it orthogonal to both \( F_1([u]) \) and \( F_0 \). All these conditions are ultimately expressed analytically in terms of multi-point averages of the random process \( u(x; \omega) \).
Clearly, this equation is in the form \((300)\), with \(H([\theta], t) = 0\) and
\[
L([\theta], t) = -\int_0^{2\pi} \theta(x) \frac{\partial}{\partial x} \frac{\delta}{\delta \theta(x)} dx.
\] (302)

The solution to \((301)\) depends on the initial condition \(F_0([\theta])\) as well as on the choice of the function space \(D(F)\). To show this, let us set
\[
F_0([\theta]) = \int_0^{2\pi} K(x) \theta(x) dx,
\] (303)
where \(K(x)\) is a given kernel function\(^{21}\). As we will see in Section 7.1 if we solve \((302)\) in the space of periodic functions
\[
D(F) = \{ \theta(x) \in C^{(\infty)}([0, 2\pi]) \mid \theta(0) = \theta(2\pi) \},
\] (307)
then we obtain the solution
\[
F([\theta], t) = F_0(\theta),
\] (308)
i.e., the constant functional. On the other hand, if we consider the function space
\[
D(F) = \{ \theta(x) \in C^{(\infty)}([0, 2\pi]) \mid \theta(x) = 0 \},
\] (309)
then the solution to \((302)\) is
\[
F([\theta], t) = \begin{cases} 
\int_t^{2\pi} K(x) \theta(x) dx & t \leq 2\pi, \\
0 & t > 2\pi.
\end{cases}
\] (310)

**Existence and Uniqueness of the Solution** A fundamental question that has lasted over the years is whether the solution to linear FDEs are unique or not, given the initial state \(F_0([\theta])\) and the function space \(D(F)\). This is an unsolved mathematical problem which we hope will be addressed systematically in near future. At today, there are few general theorems and results on the existence and the uniqueness of solutions to functional differential equations (see, e.g., \([63, 80, 241]\)).

Before addressing the question of numerical approximation functional differential equations, it is useful to show how such equations look like and, more importantly, how they arise in the context of well-known mathematical theories.

\(^{21}\) The solution functional \(F([\theta], t)\) corresponding to the initial condition \((303)\) is linear and homogeneous for all \(t \geq 0\), i.e., it is in the form
\[
F([\theta], t) = \int_0^{2\pi} R(x, t) \theta(x) dx, \quad R(x, 0) = K(x).
\] (304)

To see this, it is sufficient to write the first-order Euler scheme
\[
F([\theta], \Delta t) = F([\theta], 0) - \Delta t \int_0^{2\pi} \theta(x) \frac{\partial K(x, 0)}{\partial x} dx.
\] (305)

and note that the term within the integral at the right hand side is a linear functional of \(\theta\). This implies that \(F([\theta], \Delta t)\) is a linear functional of \(\theta\). By applying this argument over and over we see that \(F([\theta], t)\) is a linear functional of \(\theta\) for all \(t \geq 0\), i.e., it is in the form \((304)\). A substitution of \((304)\) into \((301)\) yields
\[
\int_0^{2\pi} \theta(x) \left( \frac{\partial R(x, t)}{\partial t} + \frac{\partial R(x, t)}{\partial x} \right) dx = 0, \quad \text{i.e.} \quad \frac{\partial R(x, t)}{\partial t} + \frac{\partial R(x, t)}{\partial x} = 0.
\] (306)

Therefore, \((304)\) is a solution to \((301)-(303)\) if \(R(x, t)\) solves a simple linear advection equation on the real line, with initial condition \(R(x, 0) = K(x)\). On the other hand, if the initial condition \(F_0([\theta])\) is a nonlinear functional of \(\theta\), then \(F([\theta], t)\) is nonlinear functional of \(\theta(x)\). We will discuss this case extensively in Section 7.
4.1 Variational Form of PDEs

Perhaps, the simplest example of a (algebraic) functional equation is the variational form of a PDE. To show how such equation looks like, consider the scalar PDE

\[ N(u) = 0, \] \hspace{1cm} (311)

where \( N \) is a nonlinear differential operator subject to appropriate initial/boundary conditions. For example,

\[ N(u) = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2}, \quad u(x, 0) = u_0(x), \quad u(0, t) = u(L, t). \] \hspace{1cm} (312)

We multiply (311) by the test function \( \theta(x) \) in the space

\[ D(F) = \{ \theta \in C^2([0, 2\pi]) \mid \theta(0) = \theta(2\pi) \} \] \hspace{1cm} (313)

and integrate over \([0, 2\pi]\) we obtain

\[ F(\theta) = (N(u), \theta) = 0, \quad \forall \theta \in D(F). \] \hspace{1cm} (314)

This is the starting point of the well-known method of weighed residuals [62], from which classical Galerkin, collocation, least-squares and finite volumes schemes can be derived (see [101], p.18). As we will see in Section 3, to identify the functional \( F \) in this case it is sufficient to test it relative to a set of linearly independent functions \( \theta_i \) (i = 1, 2, ...), e.g., an orthonormal basis of \( D(F) \).

4.2 Schwinger-Dyson Equations

The Schwinger-Dyson equations are functional differential equations for the generating functional of a field theory. They arise in both classical statistical physics as well as in quantum field theory. Hereafter we review the main aspects of such equations.

**Statistical Physics** The functional integral approach to classical statistical dynamics [96, 174, 97, 117] allows us derive formally exact evolution equations for phase space functions such as the mean and the correlation function of the solution to SODEs and SPDEs. The standard approach relies on a generating functional \( Z \). For stochastic dynamical systems in the form

\[ \frac{d\psi(t)}{dt} = \Lambda(\psi(t), t) + f(t; \omega), \] \hspace{1cm} (315)

\( Z \) can be written as a functional integral (see [174, 96, 252, 5, 211])

\[ Z([\xi, \eta]) = Z_0 \int \int D[\psi]D[\chi]A([\psi, \chi]) \exp \left( \int_0^t d\tau (\xi(\tau) \cdot \psi(\tau) + \eta(\tau) \cdot \chi(\tau)) \right), \] \hspace{1cm} (316)

where

\[ \frac{1}{Z_0} = \int \int D[\psi]D[\chi]A([\psi, \chi]), \] \hspace{1cm} (317)

and

\[ A([\psi, \chi]) = C([\chi]) \exp \left( -\frac{1}{2} \int_0^t d\tau \nabla \cdot \Lambda(\psi(\tau), \tau) - i \int_0^t d\tau \chi(\tau) \cdot \left[ \frac{d\psi(\tau)}{dt} - \Lambda(\psi(\tau), \tau) \right] \right). \] \hspace{1cm} (318)
Here $C([\chi])$ denotes the (known) characteristic functional of the external random noise $f(t; \omega)$. Clearly, if we have available the solution to the stochastic dynamical system \((315)\) then we can construct the functional $Z([\xi, \eta])$, and from it compute any statistical property we may be interested in. On the other hand, it is straightforward to show that $Z([\xi, \eta])$ satisfies a coupled system of linear functional differential equations, known as Schwinger-Dyson equations in quantum field theory [95]. The equations are in the form

\[\frac{\partial}{\partial \tau} \left( \frac{1}{i} \frac{\delta Z}{\delta \eta_k(\tau)} \right) = \eta_k(\tau) Z + \Lambda_k \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)} \right) Z - i D_k \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)} \right) Z, \]

\[\frac{\partial}{\partial \tau} \left( \frac{1}{i} \frac{\delta Z}{\delta \eta_k(\tau)} \right) = -\xi_k(\tau) Z + i \frac{\delta}{\delta \eta_j(\tau)} \frac{\partial \Lambda_k}{\partial \psi_j} \left( \frac{1}{i} \frac{\delta}{\delta \xi(\tau)} \right) Z, \]

where

\[D_i([\chi], \tau) = \frac{\delta}{\delta \chi_i(\tau)} \ln C[\chi]. \tag{321}\]

The quantities $\delta/\delta \xi_k$ and $\delta/\delta \eta_k$ are first-order functional derivative operators, defined in Appendix 2.1. By solving \((319)\)-(320) we can identify $Z([\xi, \eta])$ without any knowledge of the stochastic process $\psi(t; \omega)$. By generalizing \((316)\), it is possible to derive a functional integral formalism to classical statistical dynamics yielding Schwinger-Dyson equations for generating functionals associated with SPDEs (see, e.g., [96][136]). In particular, if the SPDE admits an action functional $A[\phi]$ (see [225][59][74][68][246][58][5]), then the construction of the generating functional as well as the derivation of the corresponding Schwinger-Dyson equations are standard. In this setting, the Schwinger-Dyson functional differential equations provide a non-perturbative formulation of the problem of computing the statistical properties of nonlinear random systems, including stochastic ODEs and stochastic PDEs.

**Quantum Field Theory** In quantum field theory, the Schwinger-Dyson equations govern the dynamics of the Green functions and they characterize the propagation field interactions [49][252][238]. Such functional equations can be employed in a perturbation setting [165] (weak coupling regime), but they show their true strength in the strong coupling regime [212][120]. The starting point to derive the Schwinger-Dyson equations is the generating functional of the correlation functions (Green functions), which can be often expressed as a functional integral

\[Z([j(x)]) = \int \mathcal{D}[\phi] e^{i A([\phi]) + i \int j(x)\phi(x)dx}. \tag{322}\]

where $A([\phi])$ is the action functional. In the context of quantum $\phi^4$-theory [110] the action $A$ for a field with mass $m$ is given by

\[A([\phi]) = \int \left[ \frac{1}{2} (\nabla \phi(x))^2 - \frac{1}{2} m^2 \phi^2(x) - \frac{\lambda}{4!} \phi^4(x) \right] dx. \tag{323}\]

By employing \((322)\), we can express the Green functions of the quantum field theory at any order by functional differentiation i.e.,

\[Z([0])G(x_1, \ldots, x_n) = \int \mathcal{D}[\phi] \phi(x_1) \cdots \phi(x_n) e^{i A([\phi])}, \]

\[= \frac{1}{i^n} \frac{\delta^n Z([j(x)])}{\delta j(x_1) \cdots \delta j(x_n)} \bigg|_{j(x) = 0}. \tag{324}\]

\[22\]The so-called connected Green functions of the field theory are obtained by functional differentiation of $\log Z([j])$. The main reason for such definition is that if we represent the functional derivatives of $\log Z$ in terms of Feynman diagrams then only the connected diagrams contribute to the expansion.
Roughly speaking, the correlation functions are averages of products of fields \( \phi(x_1) \cdots \phi(x_n) \) with respect to the functional measure \( \exp iA(\phi) \). From a physical viewpoint they represent the transition amplitude for the propagation of a particle or excitation between different points in space-time. The generating functional \( Z \) satisfies the Schwinger-Dyson equation

\[
\frac{\delta A([\phi])}{\delta \phi(x)} \left( \left[ -i \frac{\delta}{\delta j(x)} \right] Z([j]) + j(x)Z([j]) \right) = 0.
\]  
(325)

For example, if the action \( A \) is the one given in (323), then the Schwinger-Dyson equation takes the form

\[
\Box \frac{\delta Z([j])}{\delta j(x)} + m^2 \frac{\delta Z([j])}{\delta j(x)} - \frac{\lambda}{3!} \frac{\delta^3 Z([j])}{\delta j(x)^3} - i j(x)Z([j]) = 0.
\]  
(326)

A substitution of the functional Taylor expansion of \( Z([j]) \) into this equation yields an infinite-dimensional coupled PDE system for the correlation functions \( G(x_1, \ldots, x_n) \). Here \( x_i \) is a quadruple of coordinates. Solution methods for the Schwinger-Dyson equations rely on truncated series expansions \cite{165, 13}, or renormalized expansions of the generating functional \( Z([\eta]) \) (see \cite{108} and \cite{5}, p. 385), or numerical algorithms \cite{49, 120, 119}.

### 4.3 Hopf Characteristic Functional Equations

The Hopf characteristic functional of a random field is the functional Fourier transform of the probability density functional (see \cite{195, 228} and Appendix A). To introduce this mathematical object in a simple way, let us consider an integrable random function \( u(x; \omega) \) on an interval \( x \in [a, b] \). The Hopf functional associated with \( u(x; \omega) \) is defined as

\[
\Phi([\theta(x)]) = \exp \left\{ i \int_a^b u(x; \omega) \theta(x) dx \right\},
\]  
(327)

where \( \theta(x) \) is a deterministic function (test function), \( i \) is the imaginary unit and the average is defined as a functional integral over the probability functional of \( u(x; \omega) \). Equation (327) assigns to each function \( \theta(x) \) a complex number \( \Phi([\theta(x)]) \) (see Figure 1). Similarly to the probability density functional, the Hopf characteristic functional (327) encodes the full statistical information of the random function \( u(x; \omega) \), including multi-point moments, joint characteristic functions and probability density functions (see \cite{144, 145}). If we consider instead of a random function \( u(x; \omega) \) a random vector field \( u(x, t; \omega) \), e.g., a stochastic solution to the Navier-Stokes equations, then we define

\[
\Phi([\theta(x)], t) = \exp \left\{ i \int_V u(x, t; \omega) \cdot \theta(x) dx \right\},
\]  
(329)

where \( V \) is a spatial domain in \( \mathbb{R}^d \) (\( d = 2, 3 \)). By the Riemann-Lebesgue lemma we have that

\[
\Phi([\theta(x)], t) \to 0 \quad \text{as} \quad \| \theta(x) \| \to \infty,
\]  
(330)

with a rate that depends on the regularity of the underlying probability density functional. The derivation of the Hopf characteristic functional equation is relatively straightforward if the random field \( u(x, t; \omega) \) satisfies a nonlinear PDE with polynomial nonlinearities. Hereafter we provide some examples.

\footnote{Lewis and Kraichnan \cite{124, 195} introduced a space-time generalization of (329), namely

\[
\Phi([\theta(x), t]) = \left\{ \exp \left\{ i \int_V u(x, t; \omega) \cdot \theta(x, t) dx dt \right\} \right\}.
\]  
(328)

This functional allows us to determine joint multi-point statistics of the random field \( u(x, t; \omega) \) at different times. }
Burgers-Hopf Equation  Consider the Burgers equation
\[
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2},
\]
(331)
in a periodic spatial domain \([0, 2\pi]\), and let the initial condition \(u_0(x, \omega)\) be random. By differentiating the Hopf functional
\[
\Phi([\theta(x)], t) = \left\langle \exp \left[ i \int_0^{2\pi} \theta(x) u(x, t; \omega) dx \right] \right\rangle
\]
(332)
with respect to time and using (331) we obtain
\[
\frac{\partial \Phi([\theta], t)}{\partial t} = i \int_0^{2\pi} \theta(x) \left( \frac{\partial u(x, t; \omega)}{\partial t} \exp \left[ i \int_0^{2\pi} u(x, t; \omega) \theta(x) dx \right] \right) dx
\]
(333)
i.e.,
\[
\frac{\partial \Phi([\theta], t)}{\partial t} = \int_a^b \theta(x) \left( i \frac{\partial}{\partial x} \delta^2 \Phi([\theta], t) + \nu \frac{\partial^2}{\partial x^2} \delta \Phi([\theta], t) \right) dx.
\]
(334)
This equation is known as Burgers-Hopf equation and it has been the subject of numerous investigations (see, e.g., \[2, 91, 145\]).

Navier-Stokes-Hopf Equation  The problem of determining the evolution of the Hopf characteristic functional for the Navier-Stokes equations
\[
\frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\nabla p + \nu \nabla^2 u, \quad \nabla \cdot u = 0
\]
(335)
was deemed by Monin and Yaglom as the most compact formulation of the turbulence problem, which is the problem of determining the statistical properties of the velocity and pressure field given statistical information on the initial condition. The Hopf functional differential equation corresponding to the Navier-Stokes equations is derived in \[145\] Ch. 10 (see also \[15\], §3.1.4), and it is hereafter summarized for convenience
\[
\frac{\partial \Phi([\theta], t)}{\partial t} = \sum_{k=1}^{3} \int_V \theta_k(x) \left( i \frac{\partial}{\partial x} \frac{\partial^2 \Phi([\theta], t)}{\partial \theta_k(x) \partial \theta_j(x)} + \nu \nabla \frac{\partial^2 \Phi([\theta], t)}{\partial \theta_k(x)} \right) dx.
\]
(336)
Here \(V\) is a periodic three-dimensional box and \(\theta(x)\) is chosen in a divergence free space of test functions (see Section (7.4)). Functional formulations of non-isothermal turbulent reactive flow have been also considered, leading to more complicated Hopf equations \[46\]. As we will see in Section C.2.1 Hopf equations are equivalent to PDEs in an infinite number of variables or to infinite-dimensional systems of coupled PDEs, e.g., the Monin-Lundgren-Novikov hierarchy \[90, 146, 233, 130\]. It is interesting to note that the structure of the Hopf equation somehow resembles the weak form of a PDE. However, there is a remarkable difference: in the Hopf equation, both the solution and its functional derivatives depend on the test function. In other words, the test function appears in the functional equation a nonlinear way. In addition, the equation involves derivatives with respect to functions (functional derivatives) which are not present in classical PDEs.
It is straightforward to derive Hopf functional differential equations corresponding to linear or nonlinear evolution PDEs with polynomial nonlinearities, such as the Kuramoto-Sivashinsky equation, the nonlinear wave equation, and Maxwell’s equations subject to random boundary or random initial conditions.

**Hopf Equations Defining Random Processes**  Hopf functional equations arise naturally also in the context of random processes. For example, the Hopf equation defining the characteristic functional of a zero-mean Gaussian process is (see [112], p. 61)

\[
\frac{d}{dt}\Phi(\theta(t)), t = -\frac{1}{2}\Phi(\theta(t)), t\int_{0}^{t} C(t, \tau)\theta(\tau)d\tau, \quad \Phi(\theta(t)), 0 = 1, \quad (337)
\]

where \(C(t, \tau)\) is the covariance function of the process. In fact, the solution to (337) is the well-known Hopf functional

\[
\Phi(\theta(t)), t = \exp \left[-\frac{1}{2}\int_{0}^{t}\int_{0}^{\tau} C(\tau_{1}, \tau_{2})\theta(\tau_{1})\theta(\tau_{2}) \right]. \quad (338)
\]

Similar equations can be derived for other processes, such as the telegrapher’s random process and general Markov processes. Note that (337) does not involve any functional derivative.

### 4.4 Probability Density Functional Equations

In statistical mechanics, a system of \(n\) particles can be described by the joint probability density function \(p(x_{1}, ..., x_{n}, v_{1}, ..., v_{n}, t)\) where \((x_{i}, v_{i})\) denotes the position and the velocity of the \(i\)-th particle. Similarly, the phase space associated with any finite-dimensional approximation of the solution to a stochastic PDE can be described by the joint probability density function of the corresponding phase variables, e.g., the Fourier coefficients of the series expansion of the solution [50] [83] [146]. Consider a scalar random field \(u(x, t, \omega)\). The full statistical information of the random field \(u(x, t; \omega)\) at time \(t\) is encoded in the probability density functional\(^{24}\)

\[
P([a(x)], t) = \langle \delta [a(x) - u(x, t; \omega)] \rangle, \quad (340)
\]

where \(\delta[\cdot]\) denotes a Dirac delta functional (see Appendix A), and the average \(\langle \cdot \rangle\) is a functional integral over the probability measure of \(u(x, t; \omega)\). Any well-defined nonlinear stochastic PDE for the scalar random field \(u(x, t; \omega)\) can be rewritten as a linear functional differential equation for \(P([a(x)], t)\). Such equation can be obtained by taking an appropriate continuum limit of a finite-dimensional joint probability density equation [228] [227]. In practice, we can replace the discrete set of variables \(\{a_{1} = a(x_{1}), a_{2} = a(x_{2}), ..., a_{n} = a(x_{n})\}\) in the joint PDF equation by a continuous set denoted by a continuously indexed set \(a(x)\), i.e., a function. Partial derivatives with respect to \(a_{i} = a(x_{i})\) can then be replaced by functional derivatives with respect to \(a(x)\) (see Section 3.1), etc. An alternative method to derive the probability functional equation was proposed by Beran in [15]. The derivation parallels classical the Liouville theory, where a nonlinear dynamical system is converted into a linear PDE for the joint probability density function of the state vector [228]. To convert a nonlinear PDE into a linear equation for the probability density functional, we need to move one level up and look for a functional differential equation. Perhaps, the simplest way to derive

\(^{24}\) The probability density functional \(P([a(x)], t)\) allows us to compute all moments, cumulants and joint PDFs by using functional integration. For example,

\[
p(b_{1}, b_{2}, t) = \int \delta(b_{1} - a(x_{1}))\delta(b_{2} - a(x_{2}))P([a(x)], t)D[a(x)]. \quad (339)
\]

Computing functional integrals often requires a careful definition of the integration measure as it may be possible to run into convergence issues (see Appendix B and [15] §2.2.4).
a probability density functional equation is to inverse Fourier transform the corresponding Hopf equation, and use functional integration by parts. To illustrate the procedure, we first differentiate the Hopf functional \( \Phi([\theta], t) \) with respect to time to obtain\(^\text{25}\)

\[
\frac{\partial \Phi([\theta], t)}{\partial t} = \int \exp \left[ i \int \theta(x)a(x)dx \right] \frac{\partial P([a(x)], t)}{\partial t} D[a]. \tag{342}
\]

Then we set the equality with the functional equation that defines the evolution of \( \Phi([\theta], t) \) for a particular nonlinear PDE. For instance, if we consider the Burgers equation, then we have that \( \partial \Phi / \partial t \) is given by equation (334), i.e.,

\[
\int \exp \left[ i \int \theta(x)a(x)dx \right] \frac{\partial P([a(x)], t)}{\partial t} D[a] = \int_0^{2\pi} \int \left( -\frac{1}{2} \frac{\partial (a(x)^2)}{\partial x} + \nu \frac{\partial^2 a(x)}{\partial x^2} \right) \frac{\delta}{\delta a(x)} \left( \exp \left[ i \int_0^{2\pi} a(x)\theta(x)dx \right] \right) dxP([a], t)D[a] \tag{343}
\]

Performing a functional integration by parts and assuming that the boundary terms are zero yields

\[
\frac{\partial P([a(x)], t)}{\partial t} = - \int_0^{2\pi} \frac{\delta}{\delta a(x)} \left( \left[ -a(x) \frac{\partial a(x)}{\partial x} + \nu \frac{\partial^2 a(x)}{\partial x^2} \right] P([a], t) \right) dx. \tag{344}
\]

**The Method of Continuum Limits** The formal procedure to derive probability density functional equations we just described can be justified in a finite dimensional setting. To this end, let us consider the one-dimensional diffusion problem

\[
\frac{\partial u(x, t; \omega)}{\partial t} = \frac{\partial^2 u(x, t; \omega)}{\partial x^2}, \quad u(x, 0; \omega) = u_0(x; \omega) \quad \text{(random)} \tag{345}
\]

in the real line \( x \in \mathbb{R} \). The probability density functional equation of the solution is a subcase of equation (344), namely

\[
\frac{\partial P([a(x)], t)}{\partial t} = - \int_{-\infty}^{\infty} \frac{\delta}{\delta a(x)} \left( \frac{\partial^2 a(x)}{\partial x^2} P([a], t) \right) dx. \tag{346}
\]

To derive (346), we first discretize (345) in space, e.g., on an spatial grid with evenly spaced nodes \( x_j \) \((j = 1, \ldots, n)\), the spacing between the nodes being \( \Delta x \). If we use second-order finite differences we obtain

\[
\frac{du(x_k, t; \omega)}{dt} = \frac{u(x_{k+1}, t; \omega) - 2u(x_k, t; \omega) + u(x_{k-1}, t; \omega)}{\Delta x^2}. \tag{347}
\]

Now, let \( p(a_1, \ldots, a_n, t) \) be the joint PDF of \( \{u(x_1, t; \omega), \ldots, u(x_n, t; \omega)\} \), i.e.,

\[
p(a_1, \ldots, a_n, t) = \prod_{k=1}^{n} \delta(a_k - u(x_k, t; \omega)). \tag{348}
\]

We think of \( a_k \) as the value of some function \( a(x) \) at \( x_k \), that is \( a_k = a(x_k) \) (see Figure 12). By using

\[\int \exp \left[ i \int_0^{2\pi} \theta(x)u(x, t; \omega)dx \right] P([u_0])D[u_0] = \int \exp \left[ i \int_0^{2\pi} \theta(x)a(x)dx \right] P([a], t)D[a], \tag{341}\]

where \( P([u_0]) \) and \( P([a], t) \) are, respectively, the probability functionals of \( u_0(x; \omega) \) (initial condition) and \( u(x, t; \omega) \).

\(^{25}\) In equation (342), we employed the identity

\[
\int \exp \left[ i \int_0^{2\pi} \theta(x)u(x, t; \omega)dx \right] P([u_0])D[u_0] = \int \exp \left[ i \int_0^{2\pi} \theta(x)a(x)dx \right] P([a], t)D[a], \tag{341}\]
Figure 12: Sketch of the variables \(a(x_k)\) representing the random field \(u(x, t; \omega)\) at \(x_k\). When we send \(\Delta x\) to zero, the number of variables \(a(x_k)\) goes to infinity and the joint probability density function becomes a probability density functional (see equation (340)).

Well-known identities involving the Dirac delta function \([107]\) it can be shown that

\[
\frac{a(x_{k+1}) - 2a(x_k) + a(x_{k-1})}{\Delta x^2} \left( \prod_{k=1}^{n} \delta (a(x_k) - u(x_k, t; \omega)) \right) = \left( \frac{\partial^2 u(x_k, t; \omega)}{\partial x^2} \prod_{j=1}^{n} \delta (a(x_j) - u(x_j, t; \omega)) \right).
\]

This yields,

\[
\frac{\partial^2 a(x_k)}{\partial x^2} \left( \prod_{k=1}^{n} \delta (a(x_k) - u(x_k, t; \omega)) \right) = \left( \frac{\partial^2 u(x_k, t; \omega)}{\partial x^2} \prod_{j=1}^{n} \delta (a(x_j) - u(x_j, t; \omega)) \right),
\]

where \(\partial^2 / \partial x^2\) is the numerical differentiation operator, i.e., the approximation of the second-order derivative operator by using finite differences, or other differentiation schemes such as pseudospectral collocation \([84]\]. By extending these arguments to higher-order derivatives, we obtain

\[
\frac{\partial^s a(x_k)}{\partial x^s} \left( \prod_{k=1}^{n} \delta (a(x_k) - u(x_k, t; \omega)) \right) = \left( \frac{\partial^s u(x_k, t; \omega)}{\partial x^s} \prod_{j=1}^{n} \delta (a(x_j) - u(x_j, t; \omega)) \right), \quad s = 1, 2, ...
\]

The joint probability density function of the state vector \(u(x_i, t; \omega) (i = 1, ..., n)\) satisfies the equation

\[
\frac{\partial p}{\partial t} = - \sum_{k=1}^{n} \frac{\partial}{\partial a_k} \left( \frac{\partial^2 u(x_i, t; \omega)}{\partial x^2} \prod_{j=1}^{n} \delta (a_j - u(x_j, t; \omega)) \right) = - \sum_{k=1}^{n} \frac{\partial}{\partial a_k} \left( \frac{\partial^2 a(x_i)}{\partial x^2} p \right), \quad (352)
\]

where \(p = p(a_1, ..., a_n, t)\). The last equality follows from (350). By taking the continuum limit, i.e., by sending \(\Delta x\) to zero (and correspondingly \(n\) to infinity), we obtain the following functional differential equation for the probability density functional of the solution to equation (345)

\[
\frac{\partial P([a(x)]), t)}{\partial t} = - \int_{-\infty}^{\infty} \frac{\delta}{\delta a(x)} \left( \frac{\partial^2 a(x)}{\partial x^2} P([a(x)], t) \right) dx.
\]
This equation is in agreement with (346).

Remark: The functional equation (353) is linear in \( P([a(x)], t) \), but it involves a singular term. Such term is generated by the derivative

\[
\frac{\delta}{\delta a(x)} \left( \frac{\partial^2 a(x)}{\partial x^2} \right) = \frac{\delta}{\delta a(x)} \int_{-\infty}^{\infty} a(y) \delta''(x-y)dy,
\]

\[
= \int_{-\infty}^{\infty} \delta(x-y) \delta''(x-y)dy.
\] (354)

The last integral is equivalent to the second derivative of the Dirac delta function evaluated at zero \( x = 0 \). Such singularity can also be seen from a purely discrete viewpoint. To this end, substitute the (second-order) finite-difference approximation to the second-order derivative into (352). This yields the equation

\[
\frac{\partial p}{\partial t} = -\sum_{k=1}^{n} \frac{a(x_{k+1}) - 2a(x_k) + a(x_{k-1})}{\Delta x^2} \frac{\partial p}{\partial a_k} - \frac{2}{\Delta x^2} p.
\] (355)

It is clear that as \( \Delta x \) goes to zero (continuum limit), the term \( 2p/\Delta x^2 \) generates a singularity.

**Regularity of the Probability Functional** The solution to a probability functional equation may be an irregular functional. To understand why, consider Figure 12. When we send \( \Delta x \) to zero we have that \( x_k \) approaches \( x_{k+1} \). Correspondingly the random variables \( u(x_k; t, \omega) \) and \( u(x_{k+1}; t, \omega) \) tend to be the same random variable. In this situation, the joint PDF of \( u(x_k; t, \omega) \) and \( u(x_{k+1}; t, \omega) \) involves a Dirac delta function as \( x_k \rightarrow x_{k+1} \) (see [130], p. 970). In a continuum setting, the phenomenon we just described happens at each point \( x \). Therefore the probability density functional can be an irregular mathematical object.

**Example 1:** The probability density functional of a zero mean Gaussian random function \( u(x; \omega) \) \( (x \in \mathbb{R}) \) with covariance \( C(x, y) \) is proportional to

\[
P([a(x)]) \sim \exp \left[ -\frac{1}{2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} C^{-1}(x, y) a(x) a(y) dx dy \right],
\] (356)

where \( C^{-1}(x, y) \) is the inverse covariance function. Such inverse covariance may be obtained by solving the Fredholm integral equation of the first kind

\[
\int_{-\infty}^{\infty} C(x, y) C^{-1}(y, z) dy = \delta(x - z).
\] (357)

If \( C(x, y) \) is smooth then its differential inverse \( C^{-1}(y, z) \) must have serious singularities in order for the integral in (357) to yield a Dirac delta function (see Table 1). If \( C(x, y) \) is homogeneous, i.e., if \( C(x, y) = C(-x, -y) \), then \( C^{-1}(y, z) \) is called convolution inverse \([88, 150]\). This method was pioneered by Hirschman and Widder \([86]\) in the late forties. Relevant cases of convolution inverses are summarized in Table 1. Note that the convolution inverses of smooth covariance functions — such as the Matérn covariance in 2D — are rough functions involving Laplacians and bi-harmonic operators applied to Dirac delta functions. However, such rough functions appear within integrals in (356), and therefore we expect some regularization. For instance, if we assume that the covariance function \( C(x - y) \) is exponential (see Table 1), then from (356), we obtain

\[
P([a(x)]) \sim \exp \left[ -\frac{1}{4\hbar^2} \int_{-\infty}^{\infty} \left( a(x)^2 + \hbar^2 \frac{da(x)}{dx} \right)^2 dx \right].
\] (358)
Covariance | Inverse Covariance
---|---
Exponential (1D) | $\sigma^2 e^{-|x|/h}$ | $\frac{1}{2h\sigma^2} (\delta(x) - h\delta''(x))$
Matérn (2D) (polar coordinates) | $\sigma^2 rB_1\left(\frac{r}{h}\right)$ | $\frac{1}{4\pi\sigma^2 h^2} \left(\frac{\delta(r)}{\pi r} - 2h^2 \nabla^2 \frac{\delta(r)}{\pi r} + h^4 \nabla^4 \frac{\delta(r)}{\pi r}\right)$
Exponential (3D) (polar coordinates) | $\sigma^2 e^{-r/h}$ | $\frac{1}{8\pi\sigma^2 h^3} \left(\frac{\delta(r)}{\pi r} - 2h^2 \nabla^2 \frac{\delta(r)}{\pi r} + h^4 \nabla^4 \frac{\delta(r)}{\pi r}\right)$

Table 1: Convolution inverses of well-known covariance functions [166]. Here $B_1$ denotes the modified Bessel function of the first kind. It is seen that the convolution inverse of smooth functions are rough functions involving Laplacians and bi-harmonic operators applied to Dirac delta functions.

Probability density functional equations represent an excellent starting point to obtain effective approximations. To this end, one needs to follow (by analogy) the route taken in classical statistical mechanics in which we start with the Liouville equation, and make approximations in order to derive an computable equation for a quantity of interest. Such coarse-graining process for functional differential equations is discussed by McComb [139] in the context of fluid turbulence. Probability density functional equations were derived and studied in the context turbulent flows by Dopazo and O’Brien [46], and Rosen [193, 197, 194].

**Hopf Functionals and Probability Density Functionals:** We have seen that Hopf equations and probability density functional equations are related by a functional Fourier transform. Therefore, from a purely mathematical viewpoint they are completely equivalent. However, from the viewpoint of approximation theory they are not equivalent at all. Hopf functionals may be hard to resolve due to high-frequencies related to the complex exponential. On the other hand, probability density functional equations may have non-smooth solutions. The statistical properties of a random field can be equivalently computed by using the Hopf functional or the probability density functional. In the first case, we simply need to take functional derivatives and evaluate them at $\theta(x) = 0$ (see Section 2.1). In the second case, we need to compute functional integrals, i.e., integrals in an infinite number of variables. This requires requires a careful definition of the integration measure ([15], §2.2.4).

### 4.5 Effective Fokker-Planck Systems

Consider the stochastic dynamical system (315). Suppose we are interested in determining an evolution equation for the joint probability density function of the state vector $\psi(t)$. To this end, we think of $\psi(t)$ as a nonlinear functional of the random noise $f(t)$, i.e., we can consider the map $\psi(t) = \Psi(t; [f(t)])$. The specific form of $\Psi$ depends on the system, in particular on the nonlinear map $\Lambda(\psi, t)$ in (315). The probability density function of the $\Psi(t; [f(t)])$ can be expressed as a functional integral over the probability density functional of the noise (assuming it exists)

$$p(\psi, t) = \int \delta(\psi - \Psi(t; [f])) P([f]) D[f],$$

(359)

where $\delta$ here is a multivariate Dirac delta function. From this expression, it is clear that the random noise $f(t)$ determines $p(\psi, t)$, and therefore the structure of probability density function equation that evolves $p(\psi, t)$ in time. Such equation can be derived by using functional calculus. [228, 64, 82, 149], and it takes...
the form
\[
\frac{\partial p(\psi, t)}{\partial t} + \nabla \psi \cdot \left( \Lambda(\psi, t) p(\psi, t) + \langle \delta (\psi - \Psi(t; [f])) f(t) \rangle_f \right) = 0, \tag{360}
\]
where
\[
\langle \delta (\psi - \Psi(t; [f])) f(t) \rangle_f = \int \delta (\psi - \Psi(t; [f])) f(t) P([f]) D[f]. \tag{361}
\]

The quantity (361) represents the correlation between two functionals of the random noise, namely \(\delta (\psi - \Psi(t; [f]))\) and the noise itself \(f(t)\). Such correlation can be disentangled by using functional integral techniques (see, e.g., [21, 228, 111]). In particular, if \(f(t)\) is Gaussian then (361) can be expressed by the well-known Furutsu-Novikov-Donsker formula [228, 69, 161, 43]. Similarly, if \(f(t)\) is Gaussian white noise then (361) reduces to a simple diffusion term [228]. In this case (360) coincides with the classical Fokker-Plank equation [190]. We remark that computing the solution to (360) in the general (non-Markovian) setting is very challenging. Possible techniques rely on data-driven models that employ random paths of the SODE (315), e.g., path integral methods [81, 172, 138].

**Non-Markovian Random Processes on Random Graphs**  Consider a non-Markovian stochastic process \(\psi(t; \omega) \in \mathbb{R}^n\) evolving on a random graph with \(n\) nodes (see Figure 13). Such process could model, e.g., the propagation of epidemics in interacting individuals [6, 35]. Introducing uncertainty in the graph allows us to take into account uncertainty in the interconnections between different nodes, which is fundamentally important when modeling dynamics of social networks and disease propagation. We can characterize a random graph mathematically in terms of random edges defined by mixed random variables, i.e., random variables with continuous/discrete probability distribution. To this end, let \(g_{ij}(\omega)\) represent the interaction between the node \(i\) and the node \(j\), i.e., the flow of information from \(i\) to \(j\).\(^{26}\) Such interaction can be of different types, but roughly speaking it just characterizes how the random process \(\psi_i(t)\) (at node \(i\)) influences the random process \(\psi_j(t)\) (at node \(j\)). The type of influence is defined by a stochastic model on the random graph, e.g., a system of stochastic differential equations in the form
\[
\frac{d\psi}{dt} = \Lambda(\psi, t; G(\omega)) + f(t), \tag{362}
\]

\(^{26}\)For example, the probability density of \(g_{ij}\) could be in the form \(p(g_{ij}) = \delta(0)/2 + U_{[1,2]}(g_{ij})/2\), in which case we have 50% of chances that \(i\) does not influence \(j\) at all (term \(\delta(0)/2\)), and 50% of chance that \(g_{ij}\) is uniformly distributed in \([1, 2]\). If the graph has a time-evolving random structure, i.e., some nodes in the network have a time-varying set of neighbors, then we can introduce time-dependence in the random variables \(g_{ij}\). This makes \(g_{ij}\) a set of stochastic processes.
\[
\frac{\partial u}{\partial t} + (u \nabla) u = -\nabla p + \nu \Delta u
\]

\[
\frac{\partial \Phi(\theta, t)}{\partial t} = \sum_{k=1}^{3} \int_{V} \theta_k(x) \left( \sum_{j=1}^{3} \frac{\partial}{\partial x_j} \frac{\delta^2 \Phi(\theta, t)}{\delta \theta_j(x)} + \nu \nabla^2 \frac{\delta \Phi(\theta, t)}{\delta \theta_k(x)} \right) dx
\]

\[
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}
\]

\[
\Box + m^2 \phi = \frac{\lambda}{6} \phi^3
\]

where \( \Lambda \) is a nonlinear map defining the model, \( G(\omega) = \{g_{12}(\omega), g_{21}(\omega), \ldots\} \in \mathbb{R}^{n(n-1)/2} \), is the random vector representing all interactions among the nodes in the random graph, and \( f(t) \) is colored random noise. The stochastic system \( \Box \) can be obviously generalized to cases where we have multiplicative noises, such as in tumororal cell growth models \( \{248, 236, 61\} \). The solution to \( \Box \) (assuming it exists) is a nonlinear functional of the random vector \( G \) defining the graph, and a nonlinear functional of the stochastic process \( f(t) \). We write such functional as \( \hat{\psi}(t; G, [f(t)]) \). The probability density function of \( \psi(t) \) then can be obtained by integrating out the graph and the noise over the corresponding probability distribution, i.e.,

\[
p(\psi, t) = \int \delta \left( \psi - \hat{\psi}(t; G, [f]) \right) p(G)P([f])D[f]dG. \tag{363}
\]

Table 2: Examples of functional differential equations. In this table, \( \Phi \) denotes the Hopf characteristic functional, \( P \) is the probability density functional, and \( Z \) is the generating functional of the quantum \( \phi^4 \)-theory.

| Field Equation | Functional Differential Equation |
|---------------|----------------------------------|
| \( \frac{\partial u}{\partial t} + (u \nabla) u = -\nabla p + \nu \Delta u \) | \( \frac{\partial \Phi(\theta, t)}{\partial t} = \sum_{k=1}^{3} \int_{V} \theta_k(x) \left( \sum_{j=1}^{3} \frac{\partial}{\partial x_j} \frac{\delta^2 \Phi(\theta, t)}{\delta \theta_j(x)} + \nu \nabla^2 \frac{\delta \Phi(\theta, t)}{\delta \theta_k(x)} \right) dx \) |
| \( \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} \) | \( \frac{\partial P([a], t)}{\partial t} = -\int_{-\infty}^{\infty} \frac{\delta}{\delta a(x)} \left( -a(x) \frac{\partial a(x)}{\partial x} + \nu \frac{\partial^2 a(x)}{\partial x^2} \right) P([a], t) \right) dx \) |
| \( \Box + m^2 \phi = \frac{\lambda}{6} \phi^3 \) | \( \Box + m^2 \phi = \frac{\lambda}{6} \phi^3 \)

Note that here we assumed that the noise and the graph are statistically independent. Integration over noise is (in general) a functional integral. The exact evolution equation for \( p(X, t) \) can be obtained by using a functional calculus approach \( \{228, 64, 82, 149\} \). This yields,

\[
\frac{\partial p(\psi, t)}{\partial t} + \int \left[ \nabla_{\psi} \cdot (\Lambda(\psi, t; G)p(\psi, G, t)) + \nabla_{\psi} \cdot \left\{ \delta \left( \psi - \hat{\psi}(t; G, [f]) \right) f(t) \right\}_{f} \right] dG = 0, \tag{364}
\]

where \( p(\psi, G, t) \) is the joint probability density of \( \psi(t, \omega) \) and \( G(\omega) \), while \( \langle \cdot \rangle_f \) is defined in \( \{361\} \). As before, the correlation \( \langle \delta \left( \psi - \hat{\psi}(t; G, [f]) \right) f(t) \rangle_f \) can be disentangled by using functional calculus \( \{21, 228, 111\} \), or computed by using data-driven methods.

### 4.6 Conjugate Flow Action Functionals

In a recent paper \( \{225\} \), we have shown how to construct an action functional for a non-potential field theory by using methods of differential geometry and nonlinear functional analysis \( \{220, 154\} \). The key idea is to represent the governing equations of the field theory relative to a diffeomorphic flow of curvilinear coordinates which is assumed to be functionally dependent on the field equations, i.e., on their solution. Such flow evolves in space and time similarly to a physical fluid flow of classical mechanics and it can be chosen to symmetrize the Gâteaux derivative of the field equations relative to suitable local bilinear forms. This is equivalent to require that the governing equations of the field theory can be derived from a principle
of stationary action on a flow, which we called the \textit{conjugate flow of the theory}. The determining equations of the conjugate flow are functional differential equations. In particular, for a second-order nonlinear scalar field theory

\[
f(u; u,_{\mu}; u,_{\mu\nu}; \tilde{x}^{\mu}_{,\mu}; \tilde{x}^{\mu\nu}_{,\mu\lambda}) = 0, \quad (365)
\]

we obtain

\[
R^{\mu\nu}_{,\nu} + R^{\mu\nu}_{,\nu} \Gamma^{\lambda}_{\mu\nu} = Z^{\nu}, \quad (366)
\]

where the comma denotes differentiation with respect to the independent variable \(\sigma\), \(\Gamma^{\lambda}_{\mu\nu}\) is the Christoffel symbol of the second kind and

\[
Z^{\nu} = \frac{\partial f}{\partial u^{,\nu}} + \frac{\partial f}{\partial \tilde{x}^{\mu}_{,\nu}} \delta \tilde{x}^{\mu}_{,\nu} + \left( \frac{\partial f}{\partial \tilde{x}^{\mu}_{,\nu\rho}} + \frac{\partial f}{\partial \tilde{x}^{\mu}_{,\nu\rho}} \right) \left( \frac{\partial}{\partial \sigma^{\rho}} \frac{\delta \tilde{x}^{\mu}_{,\nu}}{\delta u^{,\nu}} + \frac{\delta^2 \tilde{x}^{\mu}_{,\nu}}{\delta u^{,\nu} \delta \sigma^{\rho}} \right), \quad (367)
\]

\[
R^{\nu\rho} = \frac{\partial f}{\partial u^{,\nu\rho}} + \frac{\partial f}{\partial \tilde{x}^{\mu}_{,\nu\rho}} \delta \tilde{x}^{\mu}_{,\nu\rho}. \quad (368)
\]

Given a solution to the field equation (365), the system of functional differential equations (366) allows us to identify the \textit{conjugate flow} \(\tilde{x}^{\mu}(\sigma^{\nu}; [u])\), \(i.e., the functional relation between the flow \tilde{x}^{\mu} and the solution u for which the PDE (365) can be derived from a principle of least action (see [225] for further details). The identification of transformation groups leaving the conjugate flow action functional invariant could lead to the discovery of new conservation laws.

\section*{4.7 Large Deviation Theory and Minimum Action Methods}

Large deviations theory deals with the probabilities of rare events that are exponentially small as a function of some parameter. To illustrate the theory, consider a nonlinear PDE perturbed by space-time additive random noise of small amplitude \(\epsilon\)

\[
\frac{\partial u}{\partial t} + G(u) = \sqrt{\epsilon} f(x, t; \omega). \quad (369)
\]

We define the set trajectories connecting two arbitrary states \{\(u_1, u_2\)\}

\[
B = \{u(x, t; \omega) : u(x, 0; \omega) = u_1, \quad u(x, T; \omega) = u_2\} \quad (370)
\]

If \(f\) is white noise, then the Freidlin-Wentzell theory gives us the following large deviation principle

\[
\lim_{\epsilon \to 0^+} \epsilon \Pr(u \in B) = \inf_{\mathcal{u} \in B} \frac{1}{2} \int_0^T \left\| \frac{\partial \mathcal{u}}{\partial t} + G(\mathcal{u}) \right\|^2_{L_2} dt \quad (371)
\]

where \(\Pr(A)\) denotes the probability of the event \(A\), \(\|\cdot\|_{L_2}^2\) indicates the \(L_2\) norm in space. The large deviation principle is equivalent to minimum action principle

\[
\min_{\begin{subarray}{c} \mathcal{u}(0, x; \omega) = u_1 \\ \mathcal{u}(T, x; \omega) = u_2 \end{subarray}} S_T([\mathcal{u}]), \quad S_T([\mathcal{u}]) = \frac{1}{2} \int_0^T \left\| \frac{\partial \mathcal{u}}{\partial t} + G(\mathcal{u}) \right\|^2_{L_2} dt. \quad (372)
\]

The minimizer of (372) is called \textit{minimum action path}, and it satisfies the functional differential equation

\[
\frac{\delta S_T([\mathcal{u}])}{\delta \mathcal{u}(x, t)} = 0 \quad \mathcal{u} \in B. \quad (373)
\]

The minimum action path is the most probable transition path from \(u_1\) to \(u_2\). A method to solve (373) is based on a direct discretization of \(S_T(u)\) [48].
5 Approximation of Functional Differential Equations

In this Section we address the numerical approximation of linear FDEs in the form (300). To this end, we develop a method of weighted residuals [62, 101] in the space of functionals that allows us to derive functional least squares, functional Galerkin and functional collocation methods to FDEs in a unified and straightforward way.

5.1 The Method of Weighted Functional Residuals

The method of weighted residuals illustrates how the choice of different weight (or test) functionals can be used to construct different classes of methods extending Galerkin, least-squares and collocation methods for PDEs to functional differential equations. The general framework resembles the classical one for PDEs, in which one minimizes a residual (least-squares method) or imposes its orthogonality relative to a suitable space of test functions (Galerkin or collocation methods). To describe the weighted residuals technique, let us consider the linear functional differential equation (300). In approximating its solution numerically we are typically replacing

\[
F(\theta, t) \approx \tilde{F}(\theta, t),
\]

(374)
e.g., a tensor canonical tensor decomposition (Section 3.3.1) or a Lagrangian interpolant (Section 3.2) with \( N \) degrees of freedom. Substitution of the approximation (374) into equation (300) yields the (functional) residual

\[
R(\theta, t) = \frac{\partial \tilde{F}(\theta, t)}{\partial t} - L(\theta, t)\tilde{F}(\theta, t) - H(\theta, t).
\]

(375)

At this point, we introduce the following inner product in the space of functionals\(^{27}\) (see Appendix B)

\[
(F, G)_W = \int F(\theta)G(\theta)W(\theta)D\theta,
\]

(376)

where \( W(\theta) \) is a known weight functional, and consider the set of equations

\[
(R(\theta, t), h_k(\theta))_W = 0 \quad k = 1, \ldots, N
\]

(377)

where \( h_k(\theta) \) are test functionals. There is no particular restriction on \( h_k(\theta) \). For example, they can be cardinal basis functionals, orthogonal polynomial functionals or other basis functionals. The system (377) allows us to determine the \( N \) degrees of freedom in the functional approximation \( \tilde{F}(\theta, t) \). Specifically, we are imposing that the residual of the FDE is orthogonal to the span of the functionals \( \{h_1, \ldots, h_N\} \). The nature of the numerical scheme is determined by the choice of the test functionals \( h_j(\theta) \) in (377).

Evaluating the functional integrals in (377) is challenging, but there are approximation methods that allow us to compute them. For example, several algorithms have been recently proposed for high-dimensional (possibly infinite-dimensional) integration [9, 176, 237, 40, 39] (see also Appendix B and Chapter 4 in [51]). The system of equations (377) defines a functional Galerkin method\(^{28}\).

\(^{27}\) The inner product (375) is a functional integral, which is usually defined in terms of a limiting procedure [177, 51]. From a mathematical viewpoint, the limiting procedure defining the functional integral measure in terms of an infinite products of elementary measurable should be handled with care. In fact, the classical Lebesgue measure does not exist in spaces of infinite dimension [132]. On the other hand, Gaussian measures are still well defined in such setting. This is why we included \( W(\theta) \) in (376). The argument leading to the result on non-existence of an analogue to the Lebesgue measure in infinite dimension is related to the argument showing that the Heine-Borel theorem does not hold in infinite-dimensional normed linear spaces.

\(^{28}\) Stochastic Galerkin methods [242] are functional Galerkin methods. Essentially, these approaches are based on stochastic representations of the solution functional (Section 3.7), and functional inner products involving probability measures. Stochastic Galerkin methods have been studied extensively in the theory of turbulence [140, 141, 122, 22], and in uncertainty quantification [243, 235].
5.1.1 Functional Collocation Methods

In this class of methods the test functionals \( h_j(\theta) \) are chosen to be Dirac delta functionals centered at \( \theta_j(x) \), i.e., \( h_j(\theta) = \delta[\theta_j(x) - \theta(x)] \). In this setting, the orthogonality condition (377) can be written as

\[
\int R([\theta],t) \delta[\theta_j(x) - \theta(x)] W([\theta]) D[\theta] = 0 \Rightarrow R([\theta_j],t) = 0
\]

(378)

In other words, in the functional collocation method we impose that the residual \( R([\theta],t) \) vanishes at \( N \) collocation nodes in \( D(F) \), i.e., \( N \) functions \( \{\theta_1(x), ..., \theta_N(x)\} \). This yields a system of \( N \) equations for the unknowns \( \{\alpha_1(t), ..., \alpha_N(t)\} \). The solution we obtain from the functional collocation method obviously interpolates the exact solution at the nodes \( \theta_i(x) \).

5.1.2 Functional Least Squares

In this class of methods we look for an approximate solution functional that minimizes the norm of the residual \( R([\theta],t) \). Such norm may be defined in terms of the functional inner product (376), i.e., \( \|R\|^2_W = (R,R)_W \). In this case, we obtain the following variational principle involving a functional integral

\[
\min_{\hat{F} \in D_N(F)} \|R([\theta],t)\|^2_W = \min_{\hat{F} \in D_N(F)} \int R([\theta],t)^2 W([\theta]) D[\theta].
\]

(379)

The stationary points of (379) corresponding to variations of the degrees of freedom \( \alpha_k(t) \) in (374) satisfy the the Euler-Lagrange equations

\[
\left( R([\theta],t), \frac{\partial R([\theta],t)}{\partial \alpha_k(t)} \right)_W = 0, \quad k = 1, ..., N.
\]

(380)

A comparison between (377) and (380) suggests that the test functionals \( h_j([\theta]) \) in this case are equation-dependent, i.e., they depend on the residual \( R([\theta],t) \) through the formula

\[
h_k([\theta],t) = \frac{\partial R([\theta],t)}{\partial \alpha_k}.
\]

(381)

Remark: Error analysis, stability and consistency of functional Galerkin, functional collocation and functional least squares methods is an open question.

Remark: If we restrict \( D(F) \) to a finite-dimensional function space, e.g., the span of a finite-dimensional basis, then weighted residual formulation we just discussed reduces to the weighted residual formulation for multivariate linear PDEs.

5.2 Temporal Discretization

The FDE (300) can be discretized in time with different numerical schemes such as Adams-Bashforth, Adams-Multon or BDF methods [184]. Such discretization is quite classical in numerical analysis, and it represents an important building block in the development of efficient algorithms to compute the numerical solution to FDEs. Hereafter we discuss functional tensor methods built upon explicit and implicit linear multistep schemes (see [184], p. 497). As we will see, such algorithms have significant advantages over
other approaches in terms of accuracy and computational cost. Given an evenly-spaced sequence of time instants \( t_k = k\Delta t \) \((k = 0, 1, \ldots)\) we write the formal solution to to the FDE (300) as

\[
F([\theta], t_n) = F([\theta], t_{n-1}) + \int_{t_{n-1}}^{t_n} (L([\theta], \tau)F([\theta], \tau) + H([\theta], \tau)) \, d\tau. \tag{382}
\]

By approximating the temporal integral with a quadrature rule we obtain a fully discrete time-integration scheme. For example, if we replace

\[
S([\theta], \tau) = L([\theta], \tau)F([\theta], \tau) + H([\theta], \tau) \tag{383}
\]

by the interpolating polynomial at \( t_{n-1}, ..., t_{n-q}, \text{ex}trapolate in} [t_{n-1}, t_n], \text{and integrate in time we obtain the} q\text{-th order Adams-Bashforth scheme. Hereafter we provide some examples.}

### 5.2.1 Second-order Adams-Bashforth (AB2) method

We replace (383) with the polynomial interpolating \( S([\theta], t) \) at \( \{t_{n-1}, t_{n-2}\} \), extrapolate such polynomial to to \( [t_{n-1}, t_n] \) and compute the integral in (382). This yields the second-order explicit explicit scheme

\[
F([\theta], t_n) = F([\theta], t_{n-1}) + \frac{\Delta t}{2} [3S([\theta], t_{n-1}) - S([\theta], t_{n-2})] + \Delta t \tau_n([\theta]), \tag{384}
\]

where

\[
\tau_n([\theta]) = \frac{5\Delta t^3}{12} \frac{\partial^3}{\partial^3} F([\theta], t_{n-2}) + \mathcal{O}(\Delta t^4). \tag{385}
\]

The quantity \( \tau_n \) is the local truncation error at time \( t_n \) (184, p. 499). Clearly, if the operator \( L([\theta], t) \) is time-independent and \( H = 0 \), then (382) has the simpler form

\[
F([\theta], t_n) = F([\theta], t_{n-1}) + \frac{\Delta t}{2} L([\theta]) [3F([\theta], t_{n-1}) - F([\theta], t_{n-2})] + \Delta t \tau_n([\theta]). \tag{386}
\]

### 5.2.2 Crank-Nicolson Method

The Crank Nicolson method is an implicit second-order method of the Adams-Multon family. The scheme can be easily derived by discretizing the time integral in (382) with the trapezoidal rule. This yields

\[
F([\theta], t_n) = F([\theta], t_{n-1}) + \frac{\Delta t}{2} [S([\theta], t_n) + S([\theta], t_{n-1})] + \Delta t \tau_n([\theta]), \tag{387}
\]

where

\[
\tau_n([\theta]) = -\frac{\Delta t^3}{12} \frac{\partial^3}{\partial^3} F([\theta], t_{n-1}) + \mathcal{O}(\Delta t^4) \tag{388}
\]

is the local truncation error (184, p. 499). The scheme (387) can be rewritten as

\[
\left[ I - \frac{\Delta t}{2} L([\theta], t_n) \right] F([\theta], t_n) = \left[ I + \frac{\Delta t}{2} L([\theta], t_{n-1}) \right] F([\theta], t_{n-1}) + \frac{\Delta t}{2} [H([\theta], t_n) + H([\theta], t_{n-1})] + \Delta t \tau_n([\theta]), \tag{389}
\]

The integration process proceeds as follows: Given \( F([\theta], t_{n-1}) \) we build the right hand side of (389) and then solve for \( F([\theta], t_n) \). This involves inverting the following (functional differential) linear operator

\[
A([\theta], t_n) = \left[ I - \frac{\Delta t}{2} L([\theta], t_n) \right]. \tag{390}
\]

71
It is convenient to rewrite (389) as
\[ A([\theta], t_n) F([\theta], t_n) = E([\theta], t_n) + \Delta t \tau_n([\theta]), \]  
(391)
where
\[ E([\theta], t_n) = \left[ I + \frac{\Delta t}{2} L([\theta], t_{n-1}) \right] F([\theta], t_{n-1}) + \frac{\Delta t}{2} [H([\theta], t_n) + H([\theta], t_{n-1})]. \]  
(392)
is a known functional, provided \( F([\theta], t_{n-1}) \) is known (solution at time \( t_{n-1} = (n-1)\Delta t \)).

### 5.3 Functional Approximation

The solution to the FDE (300) can be approximated at each time step by using the functional approximation methods we discussed in Section 3. For example, if we restrict the domain the solution functional \( F \) to the finite-dimensional Hilbert space spanned by the orthonormal basis \( \{ \varphi_1, \ldots, \varphi_m \} \), i.e.,
\[ D_m = \text{span}\{ \varphi_1, \ldots, \varphi_m \} \subseteq D(F), \]  
(393)
then the functional becomes a multivariate function. Alternatively, we can look for an approximant of \( F \) in the space of cylindrical functionals. This yields a functional in the form (see Section 5.3)
\[ f(a_1, \ldots, a_m, t_n) \simeq F([\theta], t_n), \quad a_k = (\theta, \varphi_k). \]  
(394)

For example, in a canonical tensor decomposition setting we have
\[ f(a_1, \ldots, a_m, t_n) \simeq \sum_{l=1}^{r} \prod_{j=1}^{m} G_j^l(a_j, t_n), \quad G_j^l(a_j, t_n) = \sum_{p=1}^{P} \beta_{jp}^l(t_n) \phi_p(a_j), \]  
(395)
where \( r \) is the separation rank. Replacing with \( F([\theta], t) \) with \( f(a_1, \ldots, a_m, t) \) in (384) or (391) yields a functional residual \( R([\theta], t_n) \). For example, a substitution of (395) into the Crank-Nicolson scheme (391) yields
\[ R([\theta], t_n) = \sum_{l=1}^{r} A([\theta], t_n) G_1^l((\theta, \varphi_1), t_n) \cdots G_m^l((\theta, \varphi_m), t_n) - E([\theta], t_n). \]  
(396)
Note that we have incorporated the local truncation error \( \Delta t \tau_n \) within the residual \( R([\theta], t_n) \).

### 5.4 CP-ALS Algorithm for FDEs with Implicit Time Stepping

We have seen in Section 5.1.2 that the functional least squares formulation of the FDE (300) relies on minimizing the norm of the residual. Such residual can have different forms. In particular, if we discretize the FDE (300) in time with the Crank-Nicolson method and represent its solution by a canonical polyadic (CP) tensor expansion then the residual takes the form (396). Its norm can be defined as
\[ \| R([\theta], t_n) \|_W^2 = \| \sum_{l=1}^{r} A([\theta], t_n) G_1^l((\theta, \varphi_1), t_n) \cdots G_m^l((\theta, \varphi_m), t_n) - E([\theta], t_n) \|_W^2, \]  
(397)
where \( \| \cdot \|_W^2 \) is induced by the functional inner product (376). Recall that the functions \( G_k^l((\theta, \varphi_k), t_n) \) are in the form
\[ G_k^l((\theta, \varphi_k), t_n) = \sum_{s=1}^{Q} \beta_{ks}^l(t_n) \phi_s((\theta, \varphi_k)), \]  
(398)
\( \beta_k^{\ell}(t_n) \) \((l = 1, \ldots, r, k = 1, \ldots, m, s = 1, \ldots, Q) \) being the degrees of freedom. We look for a minimizer of \( \beta_k^{\ell}(t_n) \) computed in a parsimonious way. The key idea is to split the large scale optimization problem

\[
\min_{\beta_k^{\ell}(t_n)} \| R([\theta], t_n) \|_W^2
\]

into a sequence of optimization problems of smaller dimension, which are solved sequentially and eventually in parallel \(^{[100]}\). To this end, we define

\[
\beta_k(t_n) = [\beta_k^{1}(t_n), \ldots, \beta_k^{3}(t_n), \ldots, \beta_k^{Q}(t_n), \ldots, \beta_k^{m}(t_n)]^T, \quad k = 1, \ldots, m.
\]

(400)

Note that the vector \( \beta_k(t_n) \) collects the degrees of freedom representing the solution functional along \( (\theta, \varphi_k) \) at time \( t_n \), i.e., the set of functions \( \{G_k^1((\theta, \varphi_k), t_n), \ldots, G_k^Q((\theta, \varphi_k), t_n)\} \). Minimization of \( \beta_k(t_n) \) with respect to independent variations of \( \beta_k(t_n) \) yields the sequence of convex optimization problems

\[
\min_{\beta_k(t_n)} \| R([\theta], t_n) \|_W^2, \quad \min_{\beta_2(t_n)} \| R([\theta], t_n) \|_W^2, \quad \cdots, \quad \min_{\beta_m(t_n)} \| R([\theta], t_n) \|_W^2.
\]

(401)

This is the set of equations defining the alternating least-squares (ALS) method. The Euler-Lagrange equations identifying the stationary points of \( \beta_k(t_n) \) are

\[
M_h(t_n)\beta_h(t_n) = f_h(t_n), \quad h = 1, \ldots, m
\]

(402)

where,

\[
[M_h(t_n)]_{qs}^{zl} = \int Q_{\theta h}^z([\theta], t_n)Q_{\theta q}^l([\theta], t_n)W([\theta])D[\theta],
\]

(403)

\[
[f_h(t_n)]_{s}^{zl} = \int E([\theta], t_n)Q_{\theta h}^z([\theta], t_n)W([\theta])D[\theta],
\]

(404)

\[
Q_{\theta h}^z([\theta], t_n) = A([\theta], t_n)\phi_q((\theta, \varphi_n)) \prod_{j=1}^{m} G_j^z((\theta, \varphi_j), t_n).
\]

(405)

The ordering of the matrix elements \( [M_h(t_n)]_{qs}^{zl} \) and the vector \( [f_h(t_n)]_{s}^{zl} \) is the same as in \( \text{[207]} \). Note that minimizing the norm of the residual \( \beta_h(t_n) \) with respect \( \beta_h(t_n) \) is equivalent to impose orthogonality of \( \beta_h(t_n) \) with respect to the space spanned by the basis functionals \( Q_{\theta h}^z([\theta], t_n) \). Indeed, the system \( \text{[402]} \) is equivalent to

\[
( R([\theta], t_n), Q_{\theta h}^z([\theta], t_n) )_W = 0 \quad \text{(fixed } h = 1, \ldots, m),
\]

(406)

where \( (,)_W \) is the functional inner product \( \text{[376]} \). The system of equations \( \text{[402]} \) is symmetric and positive definite \( \text{[29]} \). This allows us to use well-known high-performance algorithms to compute the solution, e.g., the conjugate gradient method \( \text{[184], p.152]} \). It is important to emphasize that the minimization of the residual \( \text{[397]} \) is basically a fixed point problem involving the vector

\[
\beta(t_n) = [\beta_1(t_n) \quad \cdots \quad \beta_m(t_n)].
\]

(407)

The ALS method aims at solving such fixed point problem by splitting it into a sequence of linear problems \( \text{[402]} \) which are solved iteratively for each \( h \). The criterion is to freeze all \( \beta_j(t_n) \) \((j = 1, \ldots, m, j \neq h)\) when solving for \( \beta_h(t_n) \). We recall that convergence of CP-ALS iterations is, in general, not granted (see Section 3.3.1). To overcome this problem, additional regularization terms may be necessary \( \text{[1] [12]} \).

\( \text{[29]} \)The alternating least-squares formulation \( \text{[402]} \) is very similar to alternating least squares formulation for nonlinear functionals we studied in Section 3.3.1. The main difference is that here we are trying to determine the canonical tensor decomposition of a linearly mapped functional, i.e., we solving the linear system \( A([\theta], t_n)F([\theta], t) = E([\theta], t_n) \) with ALS, \( F \) being represented as a canonical tensor decomposition. On the other hand, in Section 3.3.1 we addressed the problem of computing the canonical tensor decomposition of \( F([\theta], t) \) given \( E([\theta], t) \) (compare the residual \( \text{[396]} \) with \( \text{[204]} \)). These two problems are equivalent if the linear operator \( A \) is invertible.
Evaluation of the Functional Integrals The ALS coefficients (403) and (404) are defined by functional integrals involving cylindrical functionals. The computation of such integrals is briefly addressed in Appendix B.1. Hereafter we summarize the main results. We first restrict $\theta$-integrals involving cylindrical functionals. The computation of such integrals is briefly addressed in Appendix B.1. Hereafter we summarize the main results. We first restrict $\theta(x)$ to the space of functions $393$, i.e., we assume that $\theta(x)$ can be written as

$$\theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x), \quad a_k = (\theta, \varphi_k). \quad (408)$$

In this hypothesis, the basis functionals (405) become multivariate functions of $(a_1, \ldots, a_m)$, i.e.,

$$Q_{qh}^{z}(a_1, \ldots, a_m, t_n) = A(a_1, \ldots, a_m, t_n) \phi_q(a_h) \prod_{j=1, j \neq h}^{m} G_j^z(a_j, t_n). \quad (409)$$

Similarly, the coefficients (403) and (404) can be explicitly written as multivariate integrals:

$$[M_h(t_n)]_{qs}^{zl} = \int_{-b}^{b} \cdots \int_{-b}^{b} Q_{qh}^{z}(a_1, \ldots, a_m, t_n) Q_{q}^{l}(a_1, \ldots, a_m, t_n) da_1 \cdots da_m, \quad (410)$$

$$[f_h(t_n)]_{q}^{z} = \int_{-b}^{b} \cdots \int_{-b}^{b} E(a_1, \ldots, a_m, t_n) Q_{qh}^{z}(a_1, \ldots, a_m, t_n) da_1 \cdots da_m, \quad (411)$$

The quantity $A(a_1, \ldots, a_m, t_n)$ appearing in (409) is the discrete version of the functional differential operator $A([\theta], t_n)$, i.e., it is a linear operator in the form

$$A(a_1, \ldots, a_m, t_n) = I - \frac{\Delta t}{2} L(a_1, \ldots, a_m, t_n) \quad (412)$$

where $L(a_1, \ldots, a_m, t_n)$ is the finite-dimensional version of the operator $L([\theta], t_n)$ in (300). In general, the integrals (410) and (411) can be computed numerically only for a relatively small number of variables $a_k$ ($k = 1, \ldots, m$). However, if we assume that the operator $A(a_1, \ldots, a_m, t_n)$ is separable, i.e.,

$$A(a_1, \ldots, a_m, t_n) = \sum_{k=1}^{r_A} A^k_1(a_1, t_n) \cdots A^k_m(a_m, t_n), \quad (413)$$

then the cost of computing such integrals scales linearly with the dimension $m$ of the space, since (410) and (411) can be factored as products of one-dimensional integrals. In equation (413), $A^k_r(a_i, t_n)$ are one-dimensional linear operators, while $r_A$ is the separation rank of the operator $A(a_1, \ldots, a_m, t_n)$. With the operator decomposition (413) available, we can represent the multivariate fields (405) as

$$Q_{q}^{l}(a_1, \ldots, a_m, t_n) = \sum_{k=1}^{r_A} A^k_q(a_q, t_n) \phi_q(a_q) \prod_{j=1, j \neq q}^{m} A^k_j(a_j, t_n) G_j^l(a_j, t_n). \quad (414)$$

---

\(30\)In equations (410) and (411) we have set the weight function $W(a_1, \ldots, a_n)$ equal to one. This is always possible provided the support of the integrands is compact (see Appendix B.1).
This yields the following representation of the matrix coefficients \(^{(410)}\):

\[
[M_q(t_n)]_{sh}^{lz} = \sum_{j=1}^{m} \prod_{j \neq q}^{m} \left[ \int_{-b}^{b} A^k_j(a_j, t_n)G^e_j(a_j, t_n)J^z_j(a_j, t_n) da_j \right].
\]

\(^{(415)}\) i.e., sums of products of one-dimensional integrals. Let us provide a simple example.

**Example 1:** Consider the time-independent functional differential operator \(^{(302)}\) and the associated operator defined in \(^{(390)}\)

\[
A(\theta) = I + \frac{\Delta t}{2} \int_{0}^{2\pi} \theta(x) \frac{\partial}{\partial x} \frac{\delta(x)}{\partial \theta(x)} dx.
\]

Evaluating \(A(\theta)\) in the finite-dimensional function space \(^{(393)}\) yields

\[
A(a_1, ..., a_m) = I + \frac{\Delta t}{2} \sum_{k=1}^{m} \left( \sum_{j=1}^{m} a_j \int_{0}^{2\pi} \frac{d\varphi_j(x)}{dx} dx \right) \frac{\partial}{\partial a_k}.
\]

Therefore, \(A(a_1, ..., a_m)\) is a separable operator with separation rank \(r_A = m^2 + 1\).

### 5.4.1 Collocation Setting

Consider the sequence of linear systems \(^{(402)}\), and let \(\{\phi_s(a_k)\}\) be a cardinal basis associated with the set of collocation nodes \(\{a_{k1}, ..., a_{kQ}\}\). For simplicity, we consider the same set of nodes in each dimension. In this assumption, the integrals defining the matrix entries \(^{(415)}\) can be significantly simplified. For example,

\[
\int_{-b}^{b} A^k_q(a) \phi_s(a) A^e_q(a) \phi_h(a) da \simeq \underbrace{A^k_q W A^e_q}_{K^{ke}_q} \]

where \(A^k_q\) is the matrix representation of the operator \(A^k_q(a)\) (collocation version), and \(W\) is a diagonal matrix of integration weights. The matrix \(K^{ke}_q\) is \(Q \times Q\) for all \(k, e = 1, ..., r_A\) and all \(q = 1, ..., m\). With \(K^{ke}_q\) available, it is easy to determine the matrix representation of the integral

\[
\int_{-b}^{b} A^k_q(a) G^l_j(a) A^e_q(a) G^z_j(a) da = \sum_{s, h=1}^{Q} \beta_{jsh}^l \beta_{jsh}^e \int_{-b}^{b} A^k_q(a) \phi_s(a) A^e_q(a) \phi_h(a) da \simeq \beta_{j}^{T} K^{ke}_q \beta_{j}
\]

\(^{31}\)Note that we can express all integrals at the right hand side of \(^{(415)}\) in terms of the integrals

\[
\int_{-b}^{b} A^k_q(a_q, t_n) \phi_s(a_q) A^e_q(a_q, t_n) \phi_h(a_q) da_q
\]

where \(q = 1, ..., m, k, e = 1, ..., r_A, s, h = 1, ..., Q\). In fact,

\[
\int_{-b}^{b} A^k_q(a_j, t_n) G_j(a_j, t_n) A^e_q(a_j, t_n) G^z_j(a_j, t_n) da_j = \sum_{s, h=1}^{Q} \beta_{jsh}^l (t_n) \beta_{jsh}^e (t_n) \int_{-b}^{b} A^k_q(a_j, t_n) \phi_s(a_j) A^e_q(a_j, t_n) \phi_h(a_j) da_j.
\]

\(^{(417)}\)
where $\beta_j$ here is a matrix that has $G_j^z(x_{jp})$ ($p = 1, \ldots, Q$) as $z$-th column. This yields the following matrix

$$M_q = \sum_{k,e=1}^{r_A} [R_{ke}]^T \otimes K_{ke}.$$  

(422)

where

$$R_{ke}^q = \prod_{j=1\atop j \neq q}^m \beta_j^T K_{ke}^q \beta_j.$$  

(423)

### 5.5 Tensor Formats for FDEs with Explicit Time Stepping

Consider the following finite-dimensional form of the linear FDE

$$\frac{\partial f}{\partial t} = Lf + h,$$  

(424)

where $f(a_1, \ldots, a_m, t)$ is a tensor format that approximates a functional $F([\theta], t)$ and $L(a_1, \ldots, a_m, t)$ is the linear operator arising from the discretization of the functional linear operator $L([\theta], t)$, and $h(a_1, \ldots, a_m, t)$ is the tensor format that approximates $H([\theta], t)$. In particular, consider the case where $h = 0$, $L$ is time-independent and time-integration follows the Adams-Bashforth scheme (386), i.e.,

$$f_n = f_{n-1} + \Delta t L \left( \frac{3f_{n-1} - f_{n-2}}{2} \right).$$  

(425)

In the last equation we employed the shorthand notation $f_n = f(a_1, \ldots, a_m, t_n)$. Assuming, that the operator $L$ is separable, e.g.,

$$L = -\sum_{i,j=1}^m a_i C_{ij} \frac{\partial}{\partial a_j},$$  

(426)

then a greedy computation of the tensor format $f_n$ involves the following steps:

1. compute a low rank representation of $\hat{f}_{n-1} = 3f_{n-1}/2 - f_{n-2}/2$,
2. compute a low rank representation of $f_n = p_{n-1} + \Delta t L \hat{f}_{n-1}$.

The need for a low rank representation is clear: any algebraic operation between tensors, including the application of a linear operator, increases the separation rank. Therefore, efficient rank reduction methods are needed to avoid an explosion of the number of terms when solving the PDE (424) with tensor methods. Among them we recall methods based on alternating least squares [100, 12, 56], hierarchical Tucker formats [76, 113] or block coordinate descent methods [245]. Disregarding the particular tensor format employed to represent the solution functional, we emphasize that the development of robust and efficient rank-reduction algorithms is an active area of research [8, 158].

### 6 Numerical Results: Functionals

In this Section we provide numerical results and examples on functional approximation. In particular, we discuss polynomial functional interpolants and functional tensor methods.
6.1 Linear functionals

Consider the linear functional

\[ F(\theta) = \int_0^{2\pi} K_1(x)\theta(x)dx \]  \hspace{1cm} (427)

on the Hilbert space of square integrable periodic functions in \([0, 2\pi]\)

\[ D(F) = \{ \theta \in L_2([0, 2\pi]) \mid \theta(0) = \theta(2\pi) \} . \]  \hspace{1cm} (428)

Our aim is to represent \( F(\theta) \) in terms of a functional interpolant in \( D(F) \), i.e.,

\[ F(\theta) = \sum_{k=1}^{\infty} F(\theta_k)g_k(\theta). \]  \hspace{1cm} (429)

where \( g_k(\theta) \) are cardinal basis functionals and \( \theta_k(x) \) are interpolation nodes in \( D(F) \). In particular, we choose \( \theta_k(x) = \varphi_k(x) \) where \( \{ \varphi_1(x), \varphi_2(x), \ldots \} \) is an orthonormal basis in \( D(F) \). Assuming that \( K_1(x) \) is in \( D(F) \), i.e.,

\[ K_1(x) = \sum_{k=1}^{\infty} (K_1, \varphi_k)\varphi_k(x), \]  \hspace{1cm} (430)

it follows from (427) that

\[ F(\theta) = \sum_{k=1}^{\infty} (K_1, \varphi_k)(\varphi_k, \theta), \]

\[ = \sum_{k=1}^{\infty} F(\varphi_k)(\varphi_k, \theta). \]  \hspace{1cm} (431)

This can be written as

\[ F(\theta) = \sum_{k=1}^{\infty} F(\varphi_k)g_k(\theta), \hspace{0.5cm} \text{where} \hspace{0.5cm} g_k(\theta) = (\varphi_k, \theta). \]  \hspace{1cm} (432)

Note that this representation coincides with Porter’s series expansion (112)-(113) on the index set \( I = \{1\} \).

6.1.1 Polynomial Functional Interpolation

Let us study numerically an interpolation problem involving a specific kernel. To this end, we set

\[ K_1(x) = e^{\sin(x)}(1 + \sin(\cos(x) - 2) - \frac{1}{2}) \]  \hspace{1cm} (Fig. [14]),

\[ (433) \]

and define the following interpolation nodes in \( D(F) \)

\[ \varphi_{k+1}(x) = \frac{1}{m+1} \sin \left( \frac{m+1}{2} \frac{x - x_k}{\sin \left( \frac{x - x_k}{2} \right)} \right), \hspace{0.5cm} x_k = \frac{2\pi m+1}{m+1} k = 0, 1, \ldots, m \]  \hspace{1cm} (434)
Figure 14: (a) Kernel function defining the linear functional \((427)\). (b) Three elements of the orthogonal basis function set \((434)\) \((m = 20)\).

\(m\) being an even natural number. These are well-known nodal trigonometric polynomials ([84], p. 28) satisfying the orthogonality conditions

\[
(\varphi_j, \varphi_k) = \frac{2\pi}{m+1} \delta_{kj}, \quad k, j = 1, \ldots, m + 1. \tag{435}
\]

Clearly, if \(K_1(x)\) is in the span of \(\{\varphi_1, \ldots, \varphi_{m+1}\}\) then the exact representation of the linear functional \(F[\theta]\) involves no more than \(m + 1\) terms, i.e., a truncation of the series \((432)\) to \(m + 1\) terms is exact. The kernel \((433)\) is not in such span. Next, we construct a polynomial functional that interpolates \(F[\theta]\) at the nodes \((434)\). Specifically we consider Porter’s construction (Section 3.2.3), which is very easy to implement and equivalent to Khlobystov polynomials for uniquely solvable interpolation problems. Such interpolants are dense in the space of linear functionals if we consider the set of nodes \((91)\), i.e.,

\[
\hat{S}_1^{(m)} = \{\theta_i(x) \in D(F) \mid \theta_i(x) = \varphi_i(x) \quad i = 0, \ldots, m\}. \tag{436}
\]

This means that to identify linear functionals it is sufficient to represent them relative to orthogonal bases (see Section 4.1). As we have noticed in Section 3.2.2 this is not the case for higher-order polynomial functionals or general nonlinear functionals. Now, let us consider the general expression of Porter’s interpolants

\[
F(\theta) \simeq \sum_{k=1}^{m+1} F([\varphi_k]) g_k([\theta]), \quad g_k([\theta]) = \sum_{j=1}^{m+1} H_{jk}^{-1} \sum_{p \in \mathcal{I}} (\varphi_j, \theta)^p, \tag{437}
\]

where the matrix \(H_{ij}\) is given in \((111)\). Depending on how we choose the index set \(\mathcal{I}\) we have different expressions for the basis functionals \(g_i(\theta)\). Specifically,

1. **Constant polynomial functionals** (\(\mathcal{I} = \{0\}\)). This case is degenerate and it requires Moore-Penrose pseudo-inversion of the matrix \((111)\). This yields the basis functionals

\[
g_k([\theta]) = \frac{1}{m+1}. \tag{438}
\]

2. **Homogeneous polynomial functionals of first order** (\(\mathcal{I} = \{1\}\)):

\[
g_k([\theta]) = \left[ \frac{2\pi}{m+1} \right]^{-1} (\theta, \varphi_k). \tag{439}
\]
Figure 15: Linear functionals. Functional interpolation errors obtained by using Porter’s method. Shown are the pointwise errors (441) versus $m$ for the Gaussian ensemble $G_{300}(1)$ and Porter’s basis functionals (439) ($\mathcal{I} = \{1\}$), and (440) ($\mathcal{I} = \{1, 2\}$). The expansion corresponding to $\mathcal{I} = \{1\}$ converges exponentially for obvious reasons, while the expansion corresponding to $\mathcal{I} = \{1, 2\}$ has a $O(m^{-1/2})$ convergence rate due to insufficient interpolation nodes (Fig. (b)).

3. Quadratic polynomial functionals ($\mathcal{I} = \{1, 2\}$) on the basis set $\{\varphi_1, \ldots, \varphi_{m+1}\}$. This yields basis functionals

$$g_k([\theta]) = \left[\left(\frac{2\pi}{m+1}\right) + \left(\frac{2\pi}{m+1}\right)^2\right]^{-1}\left((\theta, \varphi_k) + (\theta, \varphi_k)^2\right).$$

(440)

In Figure 15 we plot the pointwise error

$$E_m = \sup_{\theta \in G_q} \left| F([\theta]) - \sum_{k=1}^{m+1} F([\varphi_k])g_k([\theta]) \right|$$

(441)

versus $m$ for test functions $\theta$ in the Gaussian ensemble

$$G_q(\sigma) = \left\{ \theta(x) \in D(F) \mid \theta(x) = \sigma \sum_{k=1}^{q+1} b_k \varphi_k(x), \quad \{b_1, \ldots, b_{m+1}\} \text{ i.i.d. Gaussian} \right\},$$

(442)

and interpolants corresponding to the index sets $\mathcal{I} = \{1\}$ and $\mathcal{I} = \{1, 2\}$ (bases (439) and (440)). The reason why we obtain exponential convergence with the index set $\mathcal{I} = \{1\}$ is obvious: convergence of the polynomial interpolant is basically defined by the convergence of the trigonometric series of $K_1(x)$. On the other hand, Porter’s interpolant corresponding to $\mathcal{I} = 1, 2$ shows an algebraic convergence at rate $O(m^{-1/2})$. The reason is that the basis function set $\{\varphi_1, \ldots, \varphi_{m+1}\}$ does not have enough elements to correctly identify the quadratic part of the interpolant, which is zero in this case. In fact, the off-diagonal terms in (85) or (88) cannot be identified by using an orthogonal basis, unless we consider a set of nodes in the form $\{\varphi_i, (\varphi_i + \varphi_j)\}$ where $i, j = 1, \ldots, m+1$, and $j \geq i$.

6.1.2 Canonical Tensor Decomposition

We look for a representation of (427) in the form (187). The basis functions $G^l_i$ can be equivalently determined in an alternating Galerkin or least squares setting by solving the system of equations (206) with
Figure 16: Quadratic functional (447). Kernel function (448) and its symmetrized version (449). Integrating \( H_2(x, y)\theta(y)\theta(x) \) or \( K_2(x, y)\theta(y)\theta(x) \) over \( x \) and \( y \) produces exactly the same result, independently on \( \theta \).

forcing given by

\[
f_{n}^{m} = \int_{-b}^{b} \cdots \int_{-b}^{b} \sum_{p=1}^{m+1} k_{p} a_{p} \phi_{p}(a_{j}) \prod_{k=1}^{m+1} G_{k}(a_{k}) da_{1} \cdots da_{m+1},
\]

\[k_{p} = \int_{0}^{2\pi} K_{1}(x)\varphi_{p}(x)dx.
\]

All integrals in (443) can be reduced to products of one-dimensional integrals (linear functionals are fully separable). In addition, if we use a polynomial basis for \( G_{l} \), then we can represent (427) exactly as a product of constants and linear polynomials. In fact,

\[
\sum_{l=1}^{r} G_{l}(a_{1}) \cdots G_{m+1}(a_{m+1}) = \sum_{l=1}^{m+1} k_{l} a_{l},
\]

which means that the exact separation rank is \( r = m + 1 \),

\[
G_{l}(a_{l}) = a_{l} \quad \text{and} \quad \prod_{k=1}^{m+1} G_{k}(a_{k}) = k_{l}.
\]

6.2 Quadratic Functionals

Consider the quadratic functional

\[
F(\theta(x)) = 5 + \int_{0}^{2\pi} K_{1}(x)\theta(x)dx + \int_{0}^{2\pi} \int_{0}^{2\pi} H_{2}(x, y)\theta(x)\theta(y)dxdy
\]

defined on the space of square integrable periodic functions (428). We set \( K_{1}(x) \) as in (433) and \( H_{2}(x, y) \) as

\[
H_{2}(x, y) = \sin(\cos(x) + \sin(y)) \sin(y) + \frac{1}{2} \cos(\cos(x)).
\]

Replacing \( H_{2}(x, y) \) with the symmetrized kernel (see Figure 16)

\[
K_{2}(x, y) = \frac{1}{2} (H_{2}(x, y) + H_{2}(y, x))
\]

does not change the functional (447).
Figure 17: Interpolation nodes in the space of periodic functions (each function is a node). Here we plot the elements in the sets $\hat{S}_p^{(m+1)}$ (see equation (92)) for $m = 10$ and $p = 1, 2, 3, 4$. The total number of elements within each set is $\#\hat{S}_1^{(11)} = 12$, $\#\hat{S}_2^{(11)} = 78$, $\#\hat{S}_3^{(11)} = 364$, and $\#\hat{S}_4^{(11)} = 1365$.

6.2.1 Polynomial Functional Interpolation

We first show that Porter’s interpolants corresponding to the index set $I = \{0, 1, 2\}$ and Khlobystov interpolants (166) coincide when the interpolation problem is uniquely solvable, e.g., when we consider the set of nodes (92), hereafter rewritten for convenience

$$\hat{S}_2^{(m+1)} = \left\{ 0, \{\varphi_i\}_{i=1}^{m+1}, \{\varphi_i + \varphi_j\}_{j \geq i}^{m+1} \right\}. \quad (450)$$

In Figure 17 we plot the elements of such set for $m = 10$. In Figure 18 we plot the pointwise error

$$E_m = \sup_{\theta \in \mathcal{G}_{50}(1)} |F(\theta) - \Pi_2(\theta)| \quad (451)$$

versus $m$ for functions $\theta(x)$ in the Gaussian ensemble $\mathcal{G}_{50}(1)$ (see equation (442)). In (451), $\Pi_2$ represents either Porter’s or Khlobystov’s interpolant at the set of nodes (450). It is seen that both methods achieve exponential convergence rate when interpolating the polynomial functional (447). This is due to the fact that such functional expansions are basically approximating the kernel functions (433) and (448) in terms of a Fourier spectral basis. This also establishes the full equivalence of Porter’s and Khlobystov’s interpolants for uniquely solvable interpolation problems.
Figure 18: Interpolation of the quadratic functional (447) using Porter and Khlobystov approaches. Shown are the pointwise errors (451) versus $m$ for the Gaussian ensemble $G_{50}(1)$. The small deviation between Porter’s and Khlobystov’s error plots observed at $m = 24$ is due to inaccuracies in the computation of the inverse of (111). The number of interpolation nodes required to achieve accuracy of about $10^{-7}$ ($m = 20$) is $\#\hat{S}_2(21) = 253$ (see equation 93).

6.2.2 Canonical Tensor Decomposition

Evaluation of (447) in the finite-dimensional function space $D_m = \text{span}\{\varphi_1, ..., \varphi_{m+1}\}$ yields the multivariate function

$$f(a_1, ..., a_m) = 5 + \sum_{p=1}^{m+1} k_p a_p + \sum_{q,p=1}^{m+1} q_{pq} a_p a_q,$$

where $a_j = (\varphi_j, \theta)$ and

$$k_p = \int_0^{2\pi} K_1(x) \varphi_p(x) dx, \quad q_{pq} = \int_0^{2\pi} \int_0^{2\pi} K_2(x,y) \varphi_p(x) \varphi_q(x) dxdy.\quad (453)$$

The forcing term at the right hand side of (206) can be written as

$$f_{jh}^n = \int_{-b}^{b} \cdots \int_{-b}^{b} \left( 5 + \sum_{p=1}^{m+1} k_p a_p + \sum_{q,p=1}^{m+1} q_{pq} a_p a_q \right) \phi_h(a_j) \prod_{k=1}^{m+1} G_k^n(a_k) da_1 \cdots da_m, \quad (454)$$

As before, the integrals (454) can be reduced to products of one-dimensional integrals. If we use a Legendre basis $\phi_h$, then we can represent (447) exactly as a product of constants, linear and quadratic polynomials. To show this, let us consider the quadratic part of the functional (447). We have,

$$\sum_{l=1}^{r} G_1^l(a_1) \cdots G_{m+1}^l(a_{m+1}) = \sum_{q,p=1}^{m+1} q_{pq} a_p a_q.\quad (455)$$

Given the symmetry of $q_{pq}$, the separation rank for such quadratic part is $(m+1)(m+2)/2$. More generally, the function (452) is separable, with separation rank $r = 1 + (m + 1) + (m + 1)(m + 2)/2$. In fact, it can be written in the form

$$f(a_1, ..., a_m) = \sum_{l=1}^{r} a_l G_1^l(a_1) \cdots G_m^l(a_m).\quad (456)$$
Table 3: Ordering of the terms in the canonical tensor expansion (456) of the quadratic functional (447).

A possible ordering of the series could be the one in Table 3. In Figure 19 we study the accuracy of the canonical tensor decomposition in representing the quadratic functional (447). Specifically, we plot the relative pointwise error

\[ e_m = \frac{|F(\theta) - f(a_1,\ldots,a_m)|}{F(\theta)} \]

at \( \theta(x) = \sin(\cos(2x)) + \sin(4x) \) versus the number of basis functions \( m \) for different separation ranks.

### 6.2.3 Hierarchical Tucker Expansion

The hierarchical Tucker expansion aims at mitigating the dimensionality of the core-tensor of multivariate Schmidt decompositions (Section 3.3.3). It has advantages over the canonical tensor decomposition in terms of robustness and computational efficiency. In Figure 19 we study convergence of the hierarchical Tucker expansion (see Section 3.3.3) in representing the quadratic functional (447). Specifically, we plot the relative pointwise error (457) at \( \theta(x) = \sin(\cos(2x)) + \sin(4x) \) versus the number of dimensions \( m \) for different separation ranks.

### 6.3 Hopf Characteristic Functionals

In this Section we study the accuracy of polynomial functional interpolation and tensor methods in representing Hopf functionals. To this end, consider the random function

\[ u_0(x;\omega) = \frac{1}{\sqrt{q}} \sum_{k=1}^{q} [\eta_k \sin(kx) + \xi_k \cos(kx)] , \]

where \( \{\eta_k, \xi_k\}_{k=1,\ldots,q} \) are i.i.d. random variables satisfying

\[ \langle \xi_k \rangle = 0, \quad \langle \eta_k \rangle = 0, \quad \langle \xi_k^2 \rangle = 1, \quad \langle \eta_k^2 \rangle = 1. \]

In Figure 20 we plot few samples of (458), for different values of \( q \) in the hypothesis that \( \{\xi_k\} \) and \( \{\eta_k\} \) are i.i.d. Gaussian random variables. From (459) it follows that the first two statistical moments of (458) are...
Figure 19: Accuracy of canonical tensor decomposition and hierarchical Tucker expansions in representing the quadratic functional (447). Specifically, we plot the relative pointwise error at $\theta(x) = \sin(\cos(2x)) + \sin(4x)$ versus the number of dimensions $m$, and for different separation ranks.

Figure 20: Samples of the random function (458) for different values of $q$ (Gaussian $\eta_k$ and $\xi_k$).

independent of $q$, i.e., we have

$$\langle u_0(x; \omega) \rangle = 0, \quad \langle u_0(x; \omega)^2 \rangle = 1, \quad \text{for all } q. \tag{460}$$

On the other hand, the covariance function

$$C_0(x, y) = \frac{1}{q} \sum_{k=1}^{q} \left[ \sin(kx) \sin(ky) + \cos(kx) \cos(ky) \right] \tag{461}$$

does depend on $q$, as shown in Figure 21. Higher-order moments and cumulants can be computed analytically in a similar way. The Hopf functional of $u_0(x; \omega)$ is defined as

$$\Phi([\theta(x)]) = \left\langle \exp \left[ i \int_0^{2\pi} \theta(x) u_0(x; \omega) \, dx \right] \right\rangle, \tag{462}$$

where the average $\langle \cdot \rangle$ is a multi-dimensional integral with respect to the joint probability density function of $\xi_k$ and $\eta_k$. By substituting (458) into (462) we obtain

$$\Phi([\theta(x)]) = \prod_{k=1}^{q} \int e^{i\alpha_k \theta} p_{\eta_k}(\alpha) \, d\alpha \prod_{k=1}^{q} \int e^{i\alpha_k \theta} p_{\xi_k}(\alpha) \, d\alpha, \tag{463}$$
where
\[
s_k[\theta] = \frac{1}{\sqrt{q}} \int_0^{2\pi} \theta(x) \sin(kx) dx, \quad c_k[\theta] = \frac{1}{\sqrt{q}} \int_0^{2\pi} \theta(x) \cos(kx) dx.
\] (464)

Depending on the probability density functions \( p_{\eta_k}(a) \) and \( p_{\xi_k}(a) \) appearing in (463), we have different expressions of \( F([\theta]) \).

**Gaussian Random Fields**

Let us assume
\[
p_{\eta_k}(a) = \frac{1}{\sqrt{2\pi}} e^{-a^2/2} \quad p_{\xi_k}(a) = p_{\eta_k}(a).
\] (465)

The integrals in (463) are in the form
\[
\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\alpha s_k[\theta] - a^2/2} da = e^{-s_k^2[\theta]/2}.
\] (466)

Therefore, we obtain
\[
\Phi([\theta(x)]) = \exp \left[ -\frac{1}{2} \sum_{k=1}^{q} \left( s_k^2[\theta] + c_k^2[\theta] \right) \right]
\] (467)

\[
= \exp \left[ -\frac{1}{2} \int_0^{2\pi} \int_0^{2\pi} \theta(x)\theta(y) C_0(x, y) dxdy \right],
\] (468)

in agreement with well-known results for Hopf characteristic functionals of Gaussian random fields.

**Uniform Random Fields**

Let us assume
\[
p_{\eta_k}(a) = \begin{cases} 1/(2\sqrt{3}) & a \in [-\sqrt{3}, \sqrt{3}] \\ 0 & \text{otherwise} \end{cases} \quad p_{\xi_k}(a) = p_{\eta_k}(a)
\] (469)

In this way the assumptions (459) are satisfied. The integrals in (463) are easily obtained as
\[
\frac{1}{2\sqrt{3}} \int_{-\sqrt{3}}^{\sqrt{3}} e^{i\alpha s_k[\theta]} da = \frac{\sinh(i\sqrt{3}s_k[\theta])}{i\sqrt{3}s_k[\theta]} = \frac{\sin(\sqrt{3}s_k[\theta])}{\sqrt{3}s_k[\theta]}.
\] (470)
A substitution of this formula into (463) yields

$$\Phi(\theta(x)) = \prod_{k=1}^{q} \frac{\sin(\sqrt{3}s_k[\theta]) \sin(\sqrt{3}c_k[\theta])}{s_k[\theta]c_k[\theta]}.$$  (471)

Note that in both cases we just discussed the Hopf functional turns out to be real-valued. Moreover, $\Phi \to 0$ as $\|\theta\| \to \infty$ (Riemann-Lebesgue lemma), at a rate that depends on the regularity of the underlying probability density functional. In particular, in the Gaussian case $\Phi$ goes to zero faster than in the uniform case. Note also, that both (468) are (471) are entire functionals (i.e., analytic on the complex plane). This implies that the polynomial interpolation process converges pointwise \[105\].

### 6.3.1 Effective Dimension

Let us represent $\theta(x)$ in the space of periodic functions in $[0, 2\pi]$. Possible bases are the discrete trigonometric polynomials (434) or the more classical Fourier modes

$$1, \sin(kx), \cos(kx), \quad k = 1, 2, \ldots.$$  (472)

Let us now ask the following question: what is the effective dimension of the Hopf functional (462) in the space of periodic functions? Such dimension is clearly determined by the dimension of the linear functionals $s_k$ and $c_k$ in (464). If we expand the test function $\theta$ in a classical Fourier series

$$\theta(x) = a_0 + \sum_{k=1}^{N} a_k \sin(kx) + \sum_{k=1}^{N} b_k \cos(kx)$$  (473)

and we substitute it into (464) then we obtain

$$s_k[\theta] = \frac{\pi}{\sqrt{q}} a_k, \quad \text{and} \quad c_k[\theta] = \frac{\pi}{\sqrt{q}} b_k.$$  (474)

This means that the effective dimension of the Hopf functional (462) is exactly $2q$. To show this numerically, we consider the Hopf functional (462) (Gaussian case) and plot the spectrum of the covariance matrix

$$Z_{ij} = \int_{0}^{2\pi} \int_{0}^{2\pi} C_0(x_1, y) \varphi_i(x) \varphi_j(y) dx dy,$$  (475)

obtained by projecting the covariance function (461) onto the the Fourier modes (472) or, equivalently, onto the discrete trigonometric polynomials (434). As it is clearly seen from Figure 22 the number of active components is exactly $2q$. Note also that the spectrum is flat in all cases, which means that all active variables are equally important. This has important consequences when approximating the functional (468) by polynomial functionals or SSE. In particular, if we use functionals involving less than $2q$ components, e.g., if $m < 2q$ in equation (187), then we cannot approximate (468) accurately, no matter how we push the expansion order.

### 6.3.2 Polynomial Functional Interpolation

Let us expand (468) in a power series

$$\Phi(\theta(x)) = 1 - \frac{1}{2} \int_{0}^{2\pi} \int_{0}^{2\pi} C_0(x_1, x_2) \theta(x_1) \theta(x_2) dx_1 dx_2 + \frac{1}{4} \int_{0}^{2\pi} \int_{0}^{2\pi} \int_{0}^{2\pi} \int_{0}^{2\pi} C_0(x_1, x_2) C_0(x_3, x_4) \theta(x_1) \theta(x_2) \theta(x_3) \theta(x_4) dx_1 dx_2 dx_3 dx_4 + \cdots.$$  (476)
Figure 22: Spectra of the covariance matrix (475) obtained by projecting the covariance function (461) onto the Fourier modes (472), or the discrete trigonometric polynomials (434). It is seen that the number of active components is exactly $2q$ and all variables are *equally important*. This has important consequences when we approximate the Hopf functional using tensor methods or polynomial functionals. In particular, any approximation in a function space with less than $2q$ dimensions yields a systematic error.

For small $\theta(x)$ we can truncate the series, and represent $\Phi$ in terms of an interpolating polynomial functional of relatively small order, provided we have enough interpolation nodes nearby $\theta(x) = 0$. This is demonstrated in Figure 23, where we study the accuracy of Porter’s interpolant through the set of nodes $\tilde{S}_n^{(m+1)}$ (see Eq. (92)). Specifically, we plot the error

$$E_m = \sup_{\theta \in \mathcal{G}_{50}(\sigma)} |\Phi([\theta]) - \Pi_n([\theta])|$$

(477) versus $m$ for $q = 5, n = 1, 2, 3, 4$ and $\sigma = 0.01$. It is seen that the interpolants converge in both $n$ (polynomial order) and $m$ (number of basis functions). Convergence in $m$ becomes monotonic for $m \geq 10$. This is related to the fact that the effective dimension of the Hopf functionals (467) and (471) is 10 for $q = 5$ (see Section 6.3.1). Therefore for $m \geq 10$ we have enough basis functions to fully resolve them. The accuracy of the polynomial interpolants then depends only on $n$ (polynomial order), the number of interpolation nodes and their location. The number of interpolation nodes in $\tilde{S}_n^{(m+1)}$ is given in (93). For example, the case $m = 20$ and $n = 4$ yields 12650 nodes and an interpolation matrix (111) of size $12650 \times 12650$.

When evaluating the error (477) it is important to select $\mathcal{G}_{50}(\sigma)$ within the convex hull of the interpolation nodes, e.g., by choosing $\sigma$ small enough. In this way we avoid inaccuracies due to polynomial extrapolation. To mitigate this phenomenon, we can also consider different sets of interpolation nodes, e.g., the set $\tilde{S}_n^{(m+1)}$ defined in (96) with $a_{i_q}$ sampled at Gauss-Hermite sparse-grid points. In Figure 24 we plot the elements in $\tilde{S}_1^{(m+1)}, \tilde{S}_2^{(m+1)}$ and $\tilde{S}_3^{(m+1)}$ for Gauss-Hermite sparse grids of level 5 (see also Table 4). The symmetry of the nodes $a_{i_j}$ in the Fourier space yields linearly dependent nodes in $\tilde{S}_n^{(m+1)}$. Correspondingly, the interpolation matrix (111) is rank-deficient, i.e., it cannot be inverted in a classical sense. We can overcome this issue by either removing some nodes from the set $\tilde{S}_n^{(m+1)}$, or by taking the More-Penrose pseudo-inverse of (111). In the latter case, we obtain a non-cardinal basis (135) and a polynomial functional in the form (134). In Figure 25 we demonstrate convergence of such polynomial

32Recall that the matrix (111) has to be inverted to compute the cardinal basis (113).
Figure 23: Accuracy of Porter’s polynomial functional interpolants in representing the Hopf functionals (467) and (471). Shown are the pointwise errors (477) versus \( m \) for different orders of the polynomial interpolant. The errors are computed by sampling 20000 functions from the set \( \mathcal{G}_{50}(\sigma) \) with \( \sigma = 0.01 \) and then computing the supremum (477).

| \( m \) | \#\( \tilde{S}_2^{(m+1)} \) |
|-------|-----------------|
| 2     | 121             |
| 4     | 341             |
| 6     | 673             |
| 8     | 1117            |
| 10    | 1673            |
| 16    | 4013            |
| 20    | 6133            |

Table 4: Number of interpolation nodes in \( \tilde{S}_2^{(m+1)} \) with \( a_{ij} \) sampled at Gauss-Hermite sparse grids (level 5).

6.3.3 Canonical Tensor Decomposition

We evaluate the Hopf functionals (468) and (471) in the function space spanned by the finite-dimensional orthonormal basis

\[
\varphi_0(x) = \frac{1}{2\pi}, \quad \varphi_{k-1}(x) = \frac{\sin(kx/2)}{\sqrt{\pi}}, \quad \varphi_k(x) = \frac{\cos((1+k/2)x)}{\sqrt{\pi}} \quad (k \text{ even}).
\]  

Hereafter we prove that this yields exact rank-one representations of both Gaussian and uniform Hopf functionals.

**Gaussian Functionals**  By evaluating (468) in the space spanned by (478) we obtain

\[
f(a_1, \ldots, a_m) = \exp \left[ -\frac{1}{2} \sum_{p,q=1}^{m} Z_{pq} a_p a_q \right],
\]

where

\[
Z_{ij} = \int_0^{2\pi} \int_0^{2\pi} C_0(x, y) \varphi_i(x) \varphi_j(y) dx dy.
\]

It can be verified that

\[
Z_{ij} = \begin{cases} 
\frac{\pi}{q} \delta_{ij} & i, j \leq 2q \\
0 & \text{otherwise}
\end{cases}
\]

88
Figure 24: Gauss-Hermite sparse grids (left) and corresponding interpolation nodes in the function space $\tilde{S}_n^{(m)}$ (right). Specifically, we plot all elements in $\tilde{S}_1^{(9)}$ (first row, 45 elements), $\tilde{S}_2^{(9)}$ (second row, 1117 elements) and $\tilde{S}_3^{(9)}$ (third row, 3949 elements).

and therefore the Gaussian functional (468) is rank one relative to the basis (478), i.e.,

$$\Phi(\theta) = \exp \left[ -\frac{\pi}{2q} \sum_{p=1}^{2q} a_p^2 \right], \quad a_p = (\theta, \varphi_p).$$  \hfill (481)

Thus, a rank one canonical decomposition with $m = 2q$ variables is exact in the case. It’s important to remark that the Gaussian functional (468) is not rank one relative to other bases, for example (434).
Figure 25: Approximation of the Hopf functionals (467) and (471) by using polynomial functionals of total degree \( n = 1 \) and \( n = 2 \). The polynomial are in the form (134) (i.e., non-interpolatory) and they are constructed by using the set of nodes \( \tilde{S}_1^{(m)} \) and \( \tilde{S}_2^{(m)} \) defined in equation (96), with \( a_{ij} \) sampled at Gauss-Hermite sparse grids of level 5 [24]. Shown are the pointwise errors (477) versus \( m \) for functions \( \theta(x) \) in the ensemble \( \mathcal{G}_{50}(1) \).

**Uniform Functionals**  By evaluating (471) in the space spanned by (478) we obtain

\[
f(a_1, \ldots, a_m) = \prod_{k=1}^q \sin(\sqrt{3}s_k(a_1, \ldots, a_m)) \sin(\sqrt{3}c_k(a_1, \ldots, a_m)) / (3s_k(a_1, \ldots, a_m)c_k(a_1, \ldots, a_m)),
\]

where \( s_k(a_1, \ldots, a_m) \) and \( c_k(a_1, \ldots, a_m) \) are defined as (see (464))

\[
s_k(a_1, \ldots, a_m) = \frac{1}{\sqrt{q}} \sum_{p=0}^{m} (\sin(kx), \varphi_p(x)) a_p,
\]

\[
c_k(a_1, \ldots, a_m) = \frac{1}{\sqrt{q}} \sum_{p=0}^{m} (\cos(kx), \varphi_p(x)) a_p.
\]

A substitution of (478) into (483)-(484) yields

\[
s_k(a_1, \ldots, a_m) = \sqrt{\frac{\pi}{q}} a_k, \quad k = 1, \ldots, q, \quad (\text{zero otherwise}),
\]

\[
c_k(a_1, \ldots, a_m) = \sqrt{\frac{\pi}{q}} a_{k+q}, \quad k = 1, \ldots, q, \quad (\text{zero otherwise}).
\]

This means that, relative to the basis (478) the functional (471) is rank one, i.e.,

\[
\Phi([\theta]) = \prod_{k=1}^{2q} \frac{\sin(\sqrt{3\pi} a_k / \sqrt{q})}{3\sqrt{\pi} a_k / \sqrt{q}}, \quad a_k = (\theta, \varphi_k).
\]

Therefore, a rank one canonical tensor decomposition with \( m = 2q \) variables is exact. Similarly to the Gaussian case, the uniform functional is not rank-one relative to other bases, for example (434).

---

Equations (485) and (486) suggest that all the \( 2q \) variables \( a_1, \ldots, a_{2q} \) are equally important.
6.3.4 Functional Derivatives

The first- and second-order functional derivatives of the Hopf functional (462) are defined as

\[
\frac{\delta \Phi(\theta)}{\delta \theta(x)} = i \left< u_0(x; \omega) \exp \left[ i \int_0^{2\pi} \theta(x)u_0(x; \omega)dx \right] \right>,
\]
(488)

\[
\frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} = -\left< u_0(x; \omega)u_0(y; \omega) \exp \left[ i \int_0^{2\pi} \theta(x)u_0(x; \omega)dx \right] \right>,
\]
(489)

Remarkably, these derivatives can be expressed analytically in terms of simple functions. To this end, we need a formula to compute the averages in (488) and (489). A lengthy calculation shows that

\[
\frac{\delta \Phi(\theta)}{\delta \theta(x)} = \frac{i}{\sqrt{q}} \left( I^{(c)}[\theta] \sum_{k=1}^{q} \sin(kx)I_k^{(s)}[\theta] + I^{(s)}[\theta] \sum_{k=1}^{q} \cos(kx)I_k^{(c)}[\theta] \right),
\]
(490)

\[
\frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} = -\frac{1}{q} \sum_{k,h=1}^{q} \left( \sin(kx) \cos(hy)I_k^{(s)}[\theta]I_h^{(c)}[\theta] + \cos(kx) \sin(hy)I_k^{(c)}[\theta]I_h^{(s)}[\theta] \right)
- \frac{1}{q} \sum_{k,h=1}^{q} \left( I^{(c)}[\theta] \sin(kx) \sin(hy)I_{kh}^{(s)}[\theta] + I^{(s)}[\theta] \cos(kx) \cos(hy)I_{kh}^{(c)}[\theta] \right)
- \frac{1}{q} \sum_{k=1}^{q} \left( I^{(c)}[\theta] \sin(kx) \sin(ky)J_k^{(s)}[\theta] + I^{(s)}[\theta] \cos(kx) \cos(ky)J_k^{(c)}[\theta] \right),
\]
(491)

where the functionals \( I^{(c)}[\theta], I^{(s)}[\theta], I_k^{(c)}[\theta], \) etc., are defined in Table 5. These expressions are general and they hold for any random function in the form (458), with i.i.d. random variables \( \eta_k \) and \( \xi_j \). Note that in the case of uniform PDF the coefficients in Table 5 satisfy

\[
\lim_{z \to 0} G_0(z) = 1, \quad \lim_{z \to 0} G_1(z) = 0, \quad \lim_{z \to 0} G_2(z) = 1.
\]
(492)

and therefore there are no singularities at \( z = 0 \). Also, all coefficients decay to 0 when \( z \) goes to infinity. The expressions (490) and (491) can be simplified significantly for Gaussian Hopf functionals (468). In particular, we obtain\(^{34}\)

\[
\frac{\delta \Phi(\theta)}{\delta \theta(x)} = -\left[ \int_0^{2\pi} C_0(x, y)\theta(y)dy \right] \exp \left[ -\frac{1}{2} \int_0^{2\pi} \int_0^{2\pi} C_0(x, y)\theta(x)\theta(y)dxdy \right],
\]
(493)

\[
\frac{\delta^2 \Phi(\theta)}{\delta \theta(x) \delta \theta(y)} = \left[ -C_0(x, y) + \int_0^{2\pi} C_0(x, y)\theta(y)dy \int_0^{2\pi} C_0(x, y)\theta(x)dxdy \right] \times \exp \left[ -\frac{1}{2} \int_0^{2\pi} \int_0^{2\pi} C_0(x, y)\theta(x)\theta(y)dxdy \right].
\]
(494)

In Figure 26 and Figure 27 we plot the first- and second-order functional derivatives of the Hopf functionals (467) and (471), evaluated at different test functions.

\(^{34}\)Note that the first-order derivative reduces to 0 (mean field) at \( \theta = 0 \) while the second-order derivative at \( \theta = 0 \) reduces to the opposite of the correlation function \( C_0(x, y) \).
Figure 26: First-order functional derivatives of the Hopf functionals (467) (Gaussian) and (471) (Uniform) for $q = 10$ evaluated at different test functions. The Hopf functionals we are considering here are real-valued and therefore the functional derivatives are real. Note that evaluating the first-order functional derivative at $\theta(x) = 0$ (first row) yields the mean field $\langle u_0(x) \rangle = 0$. 

\[
\theta(x) = 0
\]

\[
\theta(x) = 40 \sin(6x)^2 + 5 \cos(x)^4
\]

\[
\theta(x) = 2 \exp(-\sin(4x)) + \cos(\cos(4x))
\]
Figure 27: Second-order functional derivatives of the Hopf functionals (467) (Gaussian) and (471) (Uniform) for $q = 10$ evaluated at different test functions. Both functionals are real-valued and therefore the functional derivatives are real. Note that evaluating the second-order functional derivative at $\theta(x) = 0$ (first row) yields the opposite of the correlation function $C_0(x, y)$ in (461).
Coefficients Appearing in the Functional Derivatives (490) and (491)

\[ G_n(z) = \int_{-\infty}^{\infty} a^n e^{iza} p(a) da \]

**Gaussian PDF**
- \( G_0(z) = e^{-z^2/2} \)
- \( G_1(z) = iz e^{-z^2/2} \)
- \( G_2(z) = (1 - z^2) e^{-z^2/2} \)

**Uniform PDF**
- \( G_0(z) = \frac{\sin(\sqrt{3}z)}{\sqrt{3}z} \)
- \( G_1(z) = i \frac{\sin(\sqrt{3}z)}{\sqrt{3}z^2} - i \frac{\cos(\sqrt{3}z)}{z} \)
- \( G_2(z) = \frac{2 \cos(\sqrt{3}z)}{z^2} + \frac{\sqrt{3} \sin(\sqrt{3}z)}{z} - \frac{2 \sin(\sqrt{3}z)}{\sqrt{3}z^3} \)

\[
\begin{align*}
I^{(s)}[\theta] & = \prod_{r=1}^{q} G_0(s_r[\theta]) \\
I^{(c)}[\theta] & = \prod_{r=1}^{q} G_0(c_r[\theta]) \\
I^{(s)}_k[\theta] & = G_1(s_k[\theta]) \prod_{r=1, r \neq k}^{q} G_0(s_r[\theta]) \\
I^{(c)}_k[\theta] & = G_1(c_k[\theta]) \prod_{r=1, r \neq k}^{q} G_0(c_r[\theta]) \\
J^{(s)}_k[\theta] & = G_2(s_k[\theta]) \prod_{r=1, r \neq k}^{q} G_0(s_r[\theta]) \\
J^{(c)}_k[\theta] & = G_2(c_k[\theta]) \prod_{r=1, r \neq k}^{q} G_0(c_r[\theta]) \\
I^{(s)}_{kh}[\theta] & = G_1(s_k[\theta]) G_1(s_h[\theta]) \prod_{r=1, r \neq k, h}^{q} G_0(s_r[\theta]) \\
I^{(c)}_{kh}[\theta] & = G_1(c_k[\theta]) G_1(c_h[\theta]) \prod_{r=1, r \neq k, h}^{q} G_0(c_r[\theta])
\end{align*}
\]

Table 5: Coefficients appearing in the functional derivatives (490) and (491).

**Canonical Tensor Expansion of Functional Derivatives**

We have seen in Section 3.3 that the functional derivatives of any cylindrical approximant of \( \Phi \) can be expressed as

\[
\frac{\delta \Phi([\theta])}{\delta \theta(x)} \approx \sum_{k=1}^{m} \frac{\partial f}{\partial a_k} \varphi_k(x), \tag{495}
\]

\[
\frac{\delta^2 \Phi([\theta])}{\delta \theta(x) \delta \theta(y)} \approx \sum_{k,j=1}^{m} \frac{\partial^2 f}{\partial a_k \partial a_j} \varphi_k(x) \varphi_j(y), \tag{496}
\]

where \( a_k = (\theta, \varphi_k) \). In particular, if we consider a canonical tensor expansion of \( \Phi \) then \( f \) has the form (187), and the partial derivatives in (495) and (496) can be easily computed. In Figure 28 we show that (495) and (496) provide a very accurate approximation of the functional derivatives (493) and (494).
6.4 Sine Functional

Consider the nonlinear functional

$$F(\theta) = \sin((K_1, \theta)), \quad (K_1, \theta) = \int_0^{2\pi} K_1(x) \theta(x) dx,$$  \hspace{1cm} (497)

where $K_1(x)$ is as in equation (433). We represent $F(\theta)$ in the function space

$$D_m = \text{span}\{\varphi_1, \ldots, \varphi_m\}$$  \hspace{1cm} (498)

spanned by a finite-dimensional orthonormal basis. This yields the cylindrical representation (see Section 3.3)

$$f(a_1, \ldots, a_m) = \sin\left(\sum_{k=1}^m s_k(\varphi_k, \theta)\right), \quad s_k = \int_0^{2\pi} K_1(x) \varphi_k(x) dx.$$  \hspace{1cm} (499)

In Figure 29 we study the accuracy of second-order Porter’s polynomial functionals and canonical tensor decomposition (187) in approximating the functional (499). Specifically, the polynomial functionals are constructed by using the set of nodes $S_{2(m+1)}$ defined in equation (92), where $a_{ij}$ are sampled at Gauss-Hermite sparse grids of level 5. We recall that this set yields a rank-deficient matrix (111), which requires More-Penrose pseudoinversion (see Section 6.3.2). Correspondingly, the polynomial functionals do not interpolate (499). The canonical tensor decomposition, on the other hand, is based on Legendre polynomials of order $Q = 6$. The functionals $G_k^i(\theta, \varphi_k)$ are shown in Figure 30.
Polynomial functionals

Canonical Tensor Decomposition

Figure 29: Approximation of the sine functional (499). Shown are the $L_2([-1, 1]^m)$ errors obtained by using Porter’s polynomial functionals (134) of second-order (left) with non-cardinal basis, and canonical tensor decomposition (right). Specifically, we study convergence as a function of the number of interpolation nodes in Porter’s method: $m = 10$ and sparse grids level 5 yields $\#S_2(m+1) = 1673$ nodes (see Table 4). In the canonical tensor decomposition method we show convergence of (187) as a function of the separation rank ($r$). The dimension of the test function space is chosen to be $m = 10$.

Figure 30: Tensor components of the sine functional (499) versus $(\theta, \varphi_k)$. 

96
6.4.1 Functional Derivatives

The first- and second-order functional derivatives of (497) are easily obtained as

$$\frac{\delta F(\theta)}{\delta \theta(x)} = K_1(x) \cos((K_1, \theta)),$$

(500)

$$\frac{\delta^2 F(\theta)}{\delta \theta(x) \delta \theta(y)} = -K_1(x)K_1(y) \sin((K_1, \theta)).$$

(501)

Note that for each $\theta(x)$, such functional derivatives are basically a rescaled version of the functions $K_1(x)$ and $K_1(x)K_1(y)$. In Figure 31 we compare the exact functional derivatives versus those obtained by the canonical tensor decomposition with separation rank $r = 4$. Recall that the functional derivatives can be approximated in the space of cylindrical functionals as (see Section 3.3)

$$\frac{\delta F(\theta)}{\delta \theta(x)} = \sum_{k=1}^{m} \frac{\partial f}{\partial a_k} \varphi_k(x),$$

(502)

$$\frac{\delta^2 F(\theta)}{\delta \theta(x) \delta \theta(y)} = \sum_{k,j=1}^{m} \frac{\partial^2 f}{\partial a_k \partial a_j} \varphi_k(x) \varphi_j(y).$$

(503)

In the case of canonical tensor expansions, $\partial f/\partial a_k$ and $\partial^2 f/\partial a_j \partial a_k$ are defined in (198) and (199), respectively.

Figure 31: Sine functional (497). First- and second-order functional derivatives evaluated at $\theta(x) = 9(1 + \sin(x) + \sin(2x))/10$. Specifically, we compare the analytical results (500) and (501) versus results obtained by canonical tensor decomposition (equations (183) and (185)).
7 Numerical Results: Functional Differential Equations

Computing the numerical solution to a functional differential equation is a long standing open problem in mathematical physics. In this Section we address this problem with reference to linear functional equations in the form (300). In particular, we study the initial value problem for a prototype functional advection-reaction equation. We also develop the numerical discretization of the Navier-Stokes-Hopf functional equation (see Section 1), and discuss its computational complexity.

7.1 Advection-Reaction Functional Differential Equation

Consider the following advection-reaction functional differential equation

\[ \frac{\partial F([\theta], t)}{\partial t} + \int_0^{2\pi} \theta(x) \frac{\partial}{\partial x} \left( \frac{\delta F([\theta], t)}{\delta \theta(x)} \right) dx = H([\theta], t) \]  

(504)

evolving from the initial condition

\[ F([\theta], 0) = F_0([\theta]). \]  

(505)

Here \( H([\theta], t) \) is a given functional reaction term. We assume that \( D(F) \) (the domain of the functional \( F \)) is a suitable space of functions\(^35\), e.g., the space of periodic functions in \([0, 2\pi]\) or the space of infinitely differentiable functions in \([0, 2\pi]\) such that \( \theta(0) = 0 \). Evaluation of \( F([\theta], t) \) and \( H([\theta], t) \) in the finite-dimensional subspace

\[ D_m = \left\{ \theta(x) \in D(F) \mid \theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x) \right\}, \]  

(506)

where \( \{\varphi_1, ..., \varphi_m\} \) is an orthonormal basis, yields the following multivariate functions

\[ f(a_1, ..., a_m, t) = F([\theta], t) \quad h(a_1, ..., a_m, t) = H([\theta]), \quad \theta \in D_m. \]  

(507)

We also recall that the functional derivative \( \delta F([\theta], t)/\delta \theta(x) \) can be expressed in \( D_m \) as (see Eq. (67))

\[ \frac{\delta F([\theta], t)}{\delta \theta(x)} = \sum_{j=1}^{m} \frac{\partial f}{\partial a_j} \varphi_j(x). \]  

(508)

A substitution of these expression back into (504) yields the following initial value problem for a multivariate first-order PDE

\[ \frac{\partial f}{\partial t} + \sum_{j=1}^{m} \left( \sum_{k=1}^{m} C_{jk} a_k \right) \frac{\partial f}{\partial a_j} = h, \quad f(a_1, ..., a_m, 0) = f_0(a_1, ..., a_m), \]  

(509)

where

\[ C_{jk} = \int_0^{2\pi} \varphi_k(x) \frac{d\varphi_j(x)}{dx} dx \]  

(510)

The entries \( C_{ij} \) depend on the choice of \( D(F) \), and correspondingly \( D_m \). For instance, if we assume that \( D(F) \) is the space of infinitely differentiable periodic functions in \([0, 2\pi]\) then the matrix (510) is centro-skew-symmetric.

\(^{35}\)We have seen in Section 4 that the solution to the initial value problem (504)-(505) is strongly dependent on the choice of function space \( D(F) \).
Example 1: Another example of advection FDE is
\[ \frac{\partial F}{\partial t} + \int_0^{2\pi} \theta(x) \frac{\partial}{\partial x} \left[ \frac{\delta F}{\delta \theta(x)} \right] \, dx = \int_0^{2\pi} \theta(x) \frac{\partial^2}{\partial x^2} \left[ \frac{\delta F}{\delta \theta(x)} \right] \, dx. \]  
(511)

The discrete form such equation is
\[ \frac{\partial f}{\partial t} + \sum_{k,j=1}^m a_k (C_{jk} - F_{jk}) \frac{\partial f}{\partial a_j} = 0, \]  
(512)
where \( C_{jk} \) is defined in (510) while
\[ F_{kj} = \int_0^{2\pi} \varphi_k(x) \frac{d^2 \varphi_j(x)}{dx^2} \, dx. \]  
(513)

7.1.1 Analytical Solution

The analytical solution to the initial value problem (509) can be computed by using the method of characteristics [189]. To this end, let \( C \) be the matrix with entries (510), and \( a \) be the vector of coordinates \( (a_1, \ldots, a_m) \). Then the solution to (509) is
\[ f(a,t) = f_0(e^{-tC}a) + \int_0^t h(e^{sC}a) \, ds. \]  
(514)
In the particular case where the reaction term \( h \) is zero we obtain
\[ f(a_1, \ldots, a_m, t) = f_0(a_1(t), \ldots, a_m(t)), \]  
(515)
where the (inverse) flow map \( a_i(t; a_1, \ldots, a_m) \) is given by
\[ a_i(t; a_1, \ldots, a_m) = \sum_{j=1}^m Z_{ij}(t) a_j, \quad Z(t) = e^{-tC}. \]  
(517)

The solution to the functional equation (504) can be obtained by taking the continuum limit of (515), i.e., by sending \( m \) to infinity. This yields
\[ F([\theta], t) = F_0([\theta(x,t)]), \]  
(518)
where
\[ \theta(x) = \sum_{k=1}^{\infty} a_k \varphi_k(x), \quad \theta(x,t) = \sum_{k=1}^{\infty} a_k(t) \varphi_k(x). \]  
(519)

The coefficients \( a_k(t) \) are obtained by applying the semigroup \( Z_{ij}(t) \) to \( a_k \) (see equation (517)). The analytical expression (518) can be written more rigorously in terms of the action of a semigroup \( U(t) \) [52] to \( \theta(x) \), i.e.,
\[ F([\theta], t) = F_0([U(t)\theta(x)]). \]  
(520)
As we will see, such semigroup defines a translation in the space of functions \( D(F) \). Such translation can be generated by rotations or contractions, depending on the space of functions \( D(F) \) we consider. Hereafter we discuss this matter in more detail.

---

\[ \text{36} \text{We recall that the matrix exponential appearing in (517) can be represented as} \]
\[ e^{-tC} = U^T e^{-t} U, \]  
(516)
where \( U \) is the matrix of eigenvectors of \( C \) (columnwise) and \( \Lambda \) is the diagonal matrix of eigenvalues.
Periodic Function Spaces  Assume that the function space $D(F)$ (domain of the solution functional $F$), is the space of infinitely differentiable periodic functions in $[0, 2\pi]$. In this case, the matrix $C_{ij}$ defined in (510) is skew-symmetric, thanks to the periodicity of $\varphi_k$ (just integrate (510) by parts ). Therefore, by the spectral theorem, $C_{ij}$ has purely imaginary eigenvalues $\lambda_k = ki, k \in \mathbb{Z}$. Since $C$ is skew-symmetric we have that $\exp[-tC]$ is orthogonal, i.e., it defines an isometry in $\mathbb{R}^m$. Such isometry generates a translation in the space of periodic functions with group velocity equal to one. In other words, we have

$$\theta(x,t) = \theta(x - t). \quad (521)$$

Therefore, the analytical solution to the functional differential equation (504) (with $H = 0$) in the space of periodic functions $D(F)$ is

$$F([\theta], t) = F_0([\theta(x - t)]). \quad (522)$$

From this equation, we see that if $F_0$ is invariant under translation, i.e., $F_0([\theta(x - t)]) = F_0([\theta(x)])$ then the solution functional is constantly equal to the initial condition $F_0([\theta])$. This is discussed in more detail the following two examples.

Example 1: Consider the initial condition

$$F_0([\theta]) = \sin \left( \int_0^{2\pi} \theta(x) dx \right). \quad (523)$$

We have seen in Section 3.1.2 that this nonlinear functional is approximable by a one-dimensional function relative to the standard Fourier (modal) basis in $[0, 2\pi]$. Specifically, we obtained

$$f_0(a_0, ..., a_m) = \sin \left( \sqrt{2\pi} a_0 \right), \quad (524)$$

independently on $m$. This implies that the analytical solution to the initial value problem (509) is

$$f(a_0, ..., a_m, t) = \sin \left( \sqrt{2\pi} \sum_{k=0}^m Z_{0k}(t) a_k \right), \quad \text{where} \quad Z_{0k}(t) = [e^{-tC}]_{0k}. \quad (525)$$

It can be shown that if we sort the basis elements as in (72) then $Z_{0k}(t) = \delta_{k0}$. Therefore, the solution to (504)-(523) (with $H = 0$) in the space of periodic functions is

$$F([\theta], t) = F_0([\theta]); \quad (526)$$

i.e., the constant functional. In general, the solution to a FDE is independent on the way represent the test function space $D(F)$. Thus, it shouldn’t be surprising that we obtain exactly the same result if we consider a finite-dimensional expansion in terms of nodal trigonometric polynomials. In this case, the initial condition functional can be discretized as

$$f_0(a_0, ..., a_m) = \sin \left( \eta \sum_{k=0}^m a_k \right), \quad \text{where} \quad \eta = \int_0^{2\pi} \varphi_k(x) dx = \left( \frac{2\pi}{m + 1} \right) \quad (527)$$

and the solution is

$$f(a_0, ..., a_m, t) = \sin \left( \eta \sum_{k=0}^m a_k(t) \right), \quad (528)$$
where \( a_k(t) \) is defined in (525). Since \( \exp(-tC) \) is an orthogonal matrix we have that
\[
\sum_{k=0}^{m} a_k(t) = \sum_{k=0}^{m} a_k.  \tag{529}
\]
In the limit \( m \to \infty \) (528) and (529) imply (526). Such result can also be obtained by directly noting that if \( \theta \) is periodic in \([0, 2\pi]\) then
\[
\int_0^{2\pi} \theta(x) dx = \int_0^{2\pi} \theta(x - t) dx \quad \text{for all} \; t \in \mathbb{R}. \tag{530}
\]
Substituting this into (523) and (522) yields (526).

**Example 2:** Consider the initial condition
\[
F_0([\theta]) = \exp \left[ -\int_0^{2\pi} \theta(x)^2 dx \right]. \tag{531}
\]
where \( \theta(x) \) is in the space of infinitely differentiable periodic functions in \([0, 2\pi]\). Represent \( \theta(x) \) in a finite-dimensional space spanned by any orthonormal periodic basis in \([0, 2\pi]\). This yields the following multivariate function corresponding to \( F_0([\theta]) \) (see also Section 3.1.2)
\[
f_0(a_1, \ldots, a_m) = \prod_{k=1}^{m} e^{-a_k^2}. \tag{532}
\]
The analytical solution to the multivariate PDE (509) with initial condition (532) and \( h = 0 \) is (see equation (515))
\[
f(a_1, \ldots, a_m, t) = \prod_{k=1}^{m} e^{-a_k(t)^2}, \tag{533}
\]
where
\[
a_k(t) = \sum_{j=1}^{m} Z_{kj}(t)a_j \quad \text{and} \quad Z_{kj}(t) = \left[e^{-tC}\right]_{kj}. \tag{534}
\]
In a collocation setting where the the basis functions are normalized nodal trigonometric polynomials it is easy to see that
\[
f(a_1, \ldots, a_m, t) = f(a_1, \ldots, a_m) \quad \text{for all} \; t \geq 0 \tag{535}
\]
In fact, \( a_j \) are the rescaled values of \( \theta(x) \) at node \( x_j \in [0, 2\pi] \), the rescaling coefficient being the norm of the trigonometric polynomial. Since \( \theta(x) \) is periodic, when \( t \) increases we have that values \( a_j \) are just shifted to another location in \([0, 2\pi]\), leaving the product in (533) constant. In other words, the solution to the FDE (504) (with \( H = 0 \)) corresponding to the initial condition (531) is
\[
F([\theta], t) = F_0([\theta]), \tag{536}
\]
i.e., the constant functional. As before, this result can be obtained by noting that if \( \theta(x) \) is periodic in \([0, 2\pi]\) then
\[
\int_0^{2\pi} \theta(x)^2 dx = \int_0^{2\pi} \theta(x - t)^2 dx \quad \text{for all} \; t \in \mathbb{R}. \tag{537}
\]
A substitution of (537) into (531) yields (536).
Polynomial Function Spaces

To obtain non-trivial solutions to equation (504), we consider the following space of functions

\[ D(F) = \{ \theta \in C^\infty([0, 2\pi]) \mid \theta(0) = 0 \} \tag{538} \]

and the initial condition \( (531) \). We generate an orthonormal polynomial basis spanning \( D(F) \) by orthonormalizing the modified Chebyshev basis \( xT_k(x/\pi - 1) \) in \([0, 2\pi]\) though the Gram-Schmidt procedure. The basis functions we obtain are shown in Figure 32. This allows us to define the finite-dimensional function space

\[ D_m = \left\{ \theta \in C^\infty([0, 2\pi]) \mid \theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x) \right\} \subseteq D(F), \tag{539} \]

where \( \varphi_k \) are the orthonormal basis functions shown in Figure 32. The matrix \( C_{ij} \) in this case is not skew-symmetric, and it has eigenvalues with positive real part (see Figure 32). This implies that \( Z(t) = \exp[-tC] \) is a contraction map that takes any function \( \theta(x) \in D(F) \) and continuously deforms it to \( \theta(x) = 0 \) (see Figure 33). From a dynamical system viewpoint, \( \exp[-tC]a \) is in fact stable spiral. This means that if we set the initial condition as in \( (533) \), then we obtain \( f(a_1, \ldots, a_m, t) \to 1 \) everywhere as \( t \to \infty \). The speed
at which \( f \) goes to one can be bounded by the spectral radius of \( C \). The solution to the functional equation \((504)\), with initial condition \((521)\) is

\[
F(\theta, t) = \exp \left[ - \int_0^{2\pi} \theta(x, t)^2 dx \right],
\]

where \( \theta(x, t) \) is defined in \((519)\). More explicitly,

\[
F(\theta, t) = \begin{cases} 
\exp \left[ - \int_t^{2\pi} \theta(x)^2 dx \right] & t \in [0, 2\pi] \\
1 & t > 2\pi
\end{cases}
\]

**Regularity of the Solution Functional**  
The characteristic system associated with the first-order PDE \((509)\) is

\[
\frac{d a_k}{dt} = \sum_{j=1}^{m} a_j C_{kj} \quad C_{kj} = \int_0^{2\pi} \frac{d\varphi_j(x)}{dx} \frac{d\varphi_k(x)}{dx} dx
\]

Given any test function \( \theta(x) \) in the function space \((538)\), e.g., the function \((544)\), the semigroup \( \exp(-tC) \) pushes forward in time its Fourier coefficients, yielding the function \( \theta(x, t) \) defined in equation \((519)\). In particular, if we consider the initial condition \((544)\) then \( \theta(x, t) \) is shown in Figure 33(a). As easily seen, such function has a shock discontinuity moving leftwards with velocity equal to one towards the origin as time increases. Specifically we have \( \theta(x, t) = \theta(x - t) \). Remarkably, \( \theta(x, t) \) is not in \( D_m \) if \( t > 0 \). In fact, such function does not satisfy the boundary condition \( \theta(0, t) = 0 \) \( (t > 0) \) and it has a shock discontinuity. In other words, the semigroup generated by the FDE \((504)\) immediately pushes \( \theta(x) \in D_m \) out of \( D_m \). This has important consequences when we aim at approximating \( \theta(x, t) \) with elements of \( D_m \). In particular, we need to use a high resolution to resolve the jump at \( x = 0 \) and the shock in \( [0, 2\pi] \) (see Figure 33(b)). However, we emphasize that the singularities we just mentioned do not have any serious effect on the regularity of the solution functional \((540)\). In fact, such functional involves integration in \( x \), which is very-well defined for bounded functions with a finite number of discontinuities (see Figure 34). Also, \( F(\theta, t) \) is continuous in \( \theta \) and smooth in time, thanks to the properties of the exponential semigroup. In particular, from equation \((541)\) we see that

\[
\theta_m \to \theta^* \Rightarrow |F(\theta_m, t) - F(\theta^*, t)| \to 0
\]

i.e., the solution functional is continuous. Moreover, if we restrict the set of admissible test functions to a function space \( D(F) \) that is “close enough” to \( D_m \), e.g. in the sense of \((68)\) or \((69)\), then we are allowed to say that the functional is approximable \( D_m \). In Figure 34 we show convergence of \( F(\theta_m, t) \) to \( F(\theta^*, t) \) as we increase \( m \), where

\[
\theta^*(x) = \frac{x}{4} \exp \left[ - \sin \left( \frac{x}{2} \right)^2 \right] \cos(x + 4),
\]

and \( \theta_m(x) \) is the projection of \( \theta^*(x) \) in the finite dimensional function space \( D_m \) spanned by the orthonormal polynomial basis shown in Figure 32. The function \((544)\) is shown in Figure 33 (case \( t = 0 \)).

**Functional Derivatives**  
The first-order functional derivative of the solution functional \((541)\) is

\[
\frac{\delta F(\theta, t)}{\delta \theta(x)} = \begin{cases} 
-2\theta(x) \exp \left[ - \int_t^{2\pi} \theta(x)^2 dx \right] & t \in [0, 2\pi] \\
-2\theta(x) & t > 2\pi
\end{cases}
\]
Figure 34: Functional convergence with the number of dimensions \( m \). We set \( \theta^* \) as in (544) and evaluate \( F([\theta_m], t) \) and the \( L_\infty \) error \( |F([\theta^*], t) - F([\theta_m], t)| \) versus time for different \( m \), where \( \theta_m(x) \) is the projection of \( \theta^*(x) \) in the finite dimensional function space \( D_m \) spanned by the orthonormal polynomial basis shown in Figure 32. It is seen that \( m = 10 \) yields reasonably accurate results.

Figure 35: Solution to the multivariate PDE (509) (with \( h = 0 \)) in \( m = 2 \) dimensions. The stable spiral at the origin of the characteristic system attracts every curve in the space of modes and ultimately yields \( f = 1 \) everywhere after a transient (see also Figure 34).

Note that the functional derivative at time \( t \) evaluated at \( \theta(x) \) is simply a rescaled version of \( \theta(x) \), where the scaling factor grows from \( 2 \exp \left[ \int_0^{2\pi} \theta(x) dx \right] \) (at \( t = 0 \)) to \( 2 \) (at \( t = 2\pi \)). Such derivative can be expressed in \( D_m \) as

\[
\frac{\delta F([\theta], t)}{\delta \theta(x)} \bigg|_{D_m} = \sum_{j=1}^{m} \frac{\partial f}{\partial a_j} \varphi_j(x),
\]

where

\[
f(a_1, ..., a_m, t) = \prod_{j=1}^{m} \exp \left[ - \left( \sum_{k=1}^{m} Z_{jk}(t) a_k \right)^2 \right].
\]

More explicitly,

\[
\frac{\delta F([\theta], t)}{\delta \theta(x)} \bigg|_{D_m} = -2f(a_1, ..., a_m, t) \sum_{p=1}^{m} \varphi_p(x) \sum_{k,j=1}^{m} Z_{jp}(t) Z_{jk}(t) a_k.
\]

In Figure 35 we plot the analytical solution (540) in two dimensions. The side length of the hypercube cube that encloses any level set of the solution at time \( t \) depends on the number of dimensions \( m \). In particular, if
we are interested in estimating the size of the hypercube that encloses the set \( \{ a \in \mathbb{R}^m \mid f(a, t) \geq \epsilon \} \), then we can use the formula\(^{37}\)
\[
\frac{1}{2} \sqrt{-\log(\epsilon)} \leq b \leq \frac{1}{2} \sqrt{\frac{-\log(\epsilon)}{\lambda_{\text{min}}}}.
\] (554)

where \( \lambda_{\text{min}} \) is the smallest eigenvalue of the matrix \( Z(t)^T Z(t) \) (see Equation (517)). In Figure 36 we plot the upper and the lower bound estimates of half of the side length of the hypercube that encloses the \( 10^{-10} \) level set of the solution at \( t = 1 \) versus the number of dimensions \( m \). It is seen that the size of the hypercube increases exponentially fast with \( m \). This has important consequences when it comes to numerical simulations. In particular, if we perform simulations with far field boundary conditions, then the size of the computational domain should be chosen large enough to accommodate the support of the solution throughout the simulation time interval of interest.

### 7.2 Numerical Discretization

Consider the multivariate PDE (509) with \( h = 0 \), hereafter written in the operator form
\[
\frac{\partial f}{\partial t} = Lf, \tag{555}
\]
where
\[
L = - \sum_{j,k=1}^{m} a_k C_{jk} \frac{\partial}{\partial a_j}.
\] (556)

Note that \( L \) is a separable linear operator with separation rank \( r_L = m^2 \). In fact, \( L \) can be written in the form
\[
L = \sum_{q=1}^{m^2} \alpha_q L_q^1(a_1) \cdots L_q^m(a_m), \tag{557}
\]
for suitable one-dimensional linear operators \( L_q^j(a_j) \) defined in Table 6. The solution to the multivariate PDE (555) can be represented by using any tensor series expansion. In particular, hereafter we consider the hierarchical Tucker format (Section 3.3.1) and the canonical polyadic tensor decomposition (Section 7.2.2).

\(^{37}\)The inequality (554) can be obtained by noticing that the solution \( f(a, t) \) is in the form
\[
f(a, t) = e^{-a^T S(t) a} \quad S(t) = Z(t)^T Z(t),
\] (549)
i.e.,
\[
- \log(f(a, t)) = a^T S(t) a. \tag{550}
\]

By diagonalizing \( S(t) \) we obtain
\[
(V(t)a)^T \Lambda(t) (V(t)a) = - \log(f(a, t)).
\] (551)

Upon definition of the rotated coordinate system \( y(t) = V(t)a \), we find that the largest semi-axis of the ellipse representing the \( \epsilon \) level set of \( f(a, t) \) is
\[
\frac{1}{2} \sqrt{-\log(\epsilon)} \leq b \leq \frac{1}{2} \sqrt{\frac{-\log(\epsilon)}{\lambda_{\text{min}}}}.
\] (552)

where \( \lambda_{\text{min}} \) is the smallest eigenvalue of \( S(t) \). This formula assumes that there is no rotation in the Gaussian function \( f(a, t) \) during the dynamics and therefore it provides a conservative upper bound \( b \) that coincides with the largest semi-axis of the ellipsoidal level set. On the other hand, there exist a rotation of the ellipsoid that minimizes the size of the aforementioned hypercube. Such rotation aligns the largest semi-axis with the diagonal of the hypercube. We recall that the diagonal of a hypercube in dimension \( m \) has length \( \sqrt{m} \ell \), where \( \ell \) is the side length of the hypercube. Therefore, the upper and the lower bound estimates for \( b \), i.e., the half side length of the hypercube that encloses the \( - \log(\epsilon) \) level set of the solution are
\[
\frac{1}{2} \sqrt{-\log(\epsilon)} \leq b \leq \frac{1}{2} \sqrt{\frac{-\log(\epsilon)}{\lambda_{\text{min}}}}.
\] (553)
Figure 36: Upper and lower bound estimates \([554]\) of the half side length of the hypercube that encloses the \(10^{-10}\) level set of the solution at \(t = 1\) and \(t = 3\) versus the number of dimensions \(m\). It is seen that \(b\) increases exponentially fast with \(m\). The computational domain should be at least as big as \([-b, b]^m\), where \(b\) is the number given by the upper curve. For example, if we are aiming at resolving the numerical solution in the case \(m = 6\) within the time interval \([0, 3]\), then we should consider the computational domain \([-b, b]^6\) with \(b \approx 1600\).

7.2.1 HT and CP Algorithms with Explicit Time Stepping

Let us discretize the PDE \((555)\) in time by using any explicit time stepping scheme, for example the second-order Adams-Bashforth scheme (see Section 5.2)

\[
f_{n+2} = f_{n+1} + \frac{\Delta t}{2} L (3f_{n+1} - f_n).
\] (558)

In this setting, we see that the only operations needed to compute \(f_{n+2}\) with tensor methods are: i) addition, ii) application of a (separable) linear operator, and iii) rank reduction, the last operation being the most important among all three. From a computational viewpoint, it would be also very useful if we could split the tensor operations yielding \(f_{n+2}\) into sequences of tensor operations followed by rank reduction. In this way we could minimize the storage requirements and the overall computational cost. For example, we could split \((558)\) as

1. Compute a low rank representation of \(w_{n+1} = (3f_{n+1} - f_n)\),

2. Compute a low rank representation of \(q_{n+1} = Lw_{n+1}\),

3. Compute a low rank representation of \(f_{n+2} = f_{n+1} + \Delta t q_{n+1}/2\).

Are we allowed to do so? Unfortunately no. Splitting tensor operations into sequences of tensor operations followed by rank reduction usually yields severe cancellation errors. In some cases, this problem can be overcome. For example, an efficient and robust algorithm that allows us to split sums and rank reduction operations was recently proposed in \([114]\) in the context of hierarchical Tucker formats. The algorithm leverages on the block diagonal structure that arises when adding hierarchical Tucker format. Also, the vector resulting from application of the separable linear operator \(L\) to the hierarchical Tucker format \(w_{n+1}\) at

\[38\] On the other hand, if we employ implicit time-discretization schemes such as \((559)\) we end up solving linear systems with tensor methods. This was addressed, e.g., in \([167]\,[56]\,[188]\).
Table 6: Ordering of the linear operators defined in equation (557).

| $q$ | $\alpha_q$ | $L_1^q$ | $L_2^q$ | $\cdots$ | $L_m^q$ |
|-----|-------------|---------|---------|----------|---------|
| 1   | $-C_{11}$   | $a_1 \partial_{a_1}$ | 1       | $\cdots$ | 1       |
| 2   | $-C_{12}$   | $\partial_{a_1}$     | $a_2$   | $\cdots$ | 1       |
| $\vdots$ | $\vdots$    | $\vdots$         | $\vdots$| $\vdots$ | $\vdots$ |
| $m$ | $-C_{1m}$   | $\partial_{a_1}$    | 1       | $\cdots$ | $a_m$   |
| $m+1$ | $-C_{21}$   | $a_1$             | $\partial_{a_2}$ | $\cdots$ | 1       |
| $m+2$ | $-C_{22}$   | 1                 | $a_2 \partial_{a_2}$ | $\cdots$ | 1       |
| $\vdots$ | $\vdots$    | $\vdots$         | $\vdots$| $\vdots$ | $\vdots$ |
| $2m$ | $-C_{2m}$   | 1                 | $\partial_{a_2}$   | $\cdots$ | $a_m$   |
| $\vdots$ | $\vdots$    | $\vdots$         | $\vdots$| $\vdots$ | $\vdots$ |
| $m^2-m+1$ | $-C_{m1}$   | $a_1$             | 1       | $\cdots$ | $\partial_{a_m}$ |
| $m^2-m+2$ | $-C_{m2}$   | 1                 | $a_2$   | $\cdots$ | $\partial_{a_m}$ |
| $\vdots$ | $\vdots$    | $\vdots$         | $\vdots$| $\vdots$ | $\vdots$ |
| $m^2$ | $-C_{mm}$   | 1                 | 1       | $\cdots$ | $a_m \partial_{a_m}$ |

point ii) above can be computed very efficiently if we have available a hierarchical Tucker representation of the operator $L$. Such representation can be easily constructed in a Fourier collocation setting by vectorizing all operators in table 6. Each one-dimensional operator is represented relative to a basis trigonometric (nodal) polynomials (Fourier spectral collocation method) [84]. In practice, we simply need to convert the vectorized canonical polyadic series of $L$ into a hierarchical Tucker expansion, which is a relative standard operation.

### 7.2.2 CP-ALS Algorithm with Implicit Time Stepping

Let us discretize the PDE (555) in time by using the Crank-Nicolson method. This yields

$$
\left[ I - \frac{\Delta t}{2} L \right] f_{n+1} = \left[ I + \frac{\Delta t}{2} L \right] f_n.
$$

(559)

This equation can be written in a compact notation as

$$
Af_{n+1} = Bf_n,
$$

(560)

where

$$
A = I - \frac{\Delta t}{2} L, \quad \text{and} \quad B = I + \frac{\Delta t}{2} L.
$$

(561)

By using the definition of $L$ given in (557) (see also Table 6), it is clear that both $A$ and $B$ are separable operators in the form

$$
A = 1 \cdots 1 + \frac{\Delta t}{2} \sum_{i,j=1}^{m} a_i C_{ji} \frac{\partial}{\partial a_j},
$$

(562)

$$
B = 1 \cdots 1 - \frac{\Delta t}{2} \sum_{i,j=1}^{m} a_i C_{ji} \frac{\partial}{\partial a_j}.
$$

(563)
Minimization of the into equation (559) yields the residual

These operators have separation rank \( r_A = r_B = m^2 + 1 \) and can be conveniently written as

\[
A = \sum_{q=0}^{m^2} \eta_q E^q_1(a_1) \cdots E^q_m(a_m), \quad B = \sum_{q=0}^{m^2} \zeta_q E^q_1(a_1) \cdots E^q_m(a_m), \tag{564}
\]

where all quantities are defined in Table 7. The difference with Table 6 is that we added one row (the zeroth one) to represent the identity operator 1 \( \cdots \) 1, and we rescaled all coefficients \( C_{ij} \). A substitution of the CP decomposition

\[
\hat{f}_{n+1} = \sum_{l=1}^{r} \prod_{k=1}^{m} C^l_k(a_k, t_{n+1})
\]

into equation (559) yields the residual

\[
R = A\hat{f}_{n+1} - B\hat{f}_n. \tag{566}
\]

Minimization of the \( L_2 \) norm of (566) with respect to \( \beta_q(t_{n+1}) \) yields the linear systems of equations

\[
M_q^L \beta_q(t_{n+1}) = M_q^R \beta_q(t_n), \quad q = 1, \ldots, m \tag{567}
\]

where

\[
M_q^L = \sum_{e,z=0}^{m^2} K^L_{ez} \left[ \sum_{k=1}^{m} \left( \hat{\beta}_k(t_{n+1})^T E^{ez}_k \hat{\beta}_k(t_{n+1}) \right) \otimes [ E^e_q ]^T \right],
\]

\[
M_q^R = \sum_{e,z=0}^{m^2} K^R_{ez} \left[ \sum_{k=1}^{m} \left( \hat{\beta}_k(t_n)^T E^{ez}_k \hat{\beta}_k(t_{n+1}) \right) \otimes [ E^e_q ]^T \right],
\]

and

\[
[E^e_q]_{sh} = \int_{-b}^{b} E^e_q(a) \phi_s(a) E^e_q(a) \phi_h(a) da. \tag{570}
\]

| \( q \) | \( \eta_q \) | \( \zeta_q \) | \( E^q_1 \) | \( E^q_2 \) | \( \cdots \) | \( E^q_m \) |
|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 | \cdots | 1 |
| 1 | \( \Delta t C_{11}/2 \) | \( -\Delta t C_{11}/2 \) | \( a_1 \partial_{a_1} \) | 1 | \cdots | 1 |
| 2 | \( \Delta t C_{12}/2 \) | \( -\Delta t C_{12}/2 \) | \( \partial_{a_1} \) | \( a_2 \) | \cdots | 1 |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| \( m \) | \( \Delta t C_{1m}/2 \) | \( -\Delta t C_{1m}/2 \) | \( \partial_{a_1} \) | 1 | \cdots | \( a_m \) |
| \( m+1 \) | \( \Delta t C_{21}/2 \) | \( -\Delta t C_{21}/2 \) | \( a_1 \) | \( \partial_{a_2} \) | \cdots | 1 |
| \( m+2 \) | \( \Delta t C_{22}/2 \) | \( -\Delta t C_{22}/2 \) | 1 | \( a_2 \partial_{a_2} \) | \cdots | 1 |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| \( 2m \) | \( \Delta t C_{2m}/2 \) | \( -\Delta t C_{2m}/2 \) | 1 | \( \partial_{a_2} \) | \cdots | \( a_m \) |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| \( m^2 - m + 1 \) | \( \Delta t C_{m1}/2 \) | \( -\Delta t C_{m1}/2 \) | \( a_1 \) | 1 | \cdots | \( \partial_{a_m} \) |
| \( m^2 - m + 2 \) | \( \Delta t C_{m2}/2 \) | \( -\Delta t C_{m2}/2 \) | 1 | \( a_2 \) | \cdots | \( \partial_{a_m} \) |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| \( m^2 \) | \( \Delta t C_{mm}/2 \) | \( -\Delta t C_{mm}/2 \) | 1 | 1 | \cdots | \( a_m \partial_{a_m} \) |

Table 7: Ordering of the linear operators \( A \) and \( B \) defined in (564).
In equations (568)-(569), \(\odot\) denotes the Hadamard matrix product, \(\otimes\) is the Kronecker matrix product, \(\hat{\beta}_k\) is the matrix version of \(\beta_k\), i.e.,

\[
\hat{\beta}_k(t_n) = \begin{bmatrix}
\beta_{k1}^1(t_n) & \cdots & \beta_{k1}^r(t_n) \\
\vdots & \ddots & \vdots \\
\beta_{kQ}^1(t_n) & \cdots & \beta_{kQ}^r(t_n)
\end{bmatrix},
\] (571)

\(E_q^{e,z}\) is the \(Q \times Q\) matrix (570), and \(K_L^{e,z}\) and \(K_R^{e,z}\) are entries of the matrices

\[
K_L = \eta \eta^T, \quad \text{and} \quad K_R = \zeta \zeta^T,
\] (572)

where \(\eta\) and \(\zeta\) are column vectors with entries \(\eta_q\) and \(\zeta_q\) defined in Table [7].

**Computing the Matrix System** There are many of symmetries we can exploit when constructing the separated series expansion of the operators \(A\) and \(B\) in (564). Indeed, a closer look at Table [7] suggests that if we employ the same series expansion in each variable \(a_j\) (e.g., a trigonometric series) then the number of operators in (564) that we effectively need to compute reduces to the following four

\[1, a_j, \frac{\partial}{\partial a_j}, a_j \frac{\partial}{\partial a_j} .\] (573)

This means that the number of terms that are effectively different in the fundamental matrix (570) are only 12 (9 if we are willing to employ matrix transposes). Specifically,

\[
1. \int_{-b}^{b} \phi_s \phi_h da, \quad 2. \int_{-b}^{b} a \phi_s \phi_h da, \quad 3. \int_{-b}^{b} \phi_s \frac{d\phi_h}{da} da, \quad 4. \int_{-b}^{b} a \phi_s \frac{d\phi_h}{da} da,
\]

\[
5. \int_{-b}^{b} a^2 \phi_s \phi_h da, \quad 6. \int_{-b}^{b} a^2 \phi_s \frac{d\phi_h}{da} da, \quad 7. \int_{-b}^{b} \frac{d\phi_s}{da} \phi_h da, \quad 8. \int_{-b}^{b} a \frac{d\phi_s}{da} \phi_h da,
\]

\[
9. \int_{-b}^{b} a \frac{d\phi_s}{da} \frac{d\phi_h}{da} da, \quad 10. \int_{-b}^{b} a \frac{d\phi_s}{da} \frac{d\phi_h}{da} da, \quad 11. \int_{-b}^{b} a^2 \frac{d\phi_s}{da} \phi_h da, \quad 12. \int_{-b}^{b} a^2 \frac{d\phi_s}{da} \phi_h da.
\] (574)

All these integrals can be pre-computed and stored as \(Q \times Q\) matrices (see Eq. (420)). For each \(e, z\) and \(q\), the tensor (570) corresponds to one of the 12 integrals above. Such map, denoted as \(g^{e,z}_q\), takes in the triple \((e, z, q)\), where \(q \in \{1, ..., m\}\) and \(e, z \in \{0, ..., m^2\}\), and it returns a number between 1 and 12 identifying which integral in the set (557) corresponds to the tensor entry in (570). For example, if we sort the operators as in Table [7] then in \(m = 2\) dimensions we have

\[
g_1^{e,z} = \begin{bmatrix} 1 & 8 & 7 & 2 & 1 \\ 1 & 2 & 7 & 8 & 4 \\ 3 & 10 & 9 & 4 & 3 \\ 2 & 11 & 8 & 5 & 2 \\ 1 & 8 & 7 & 2 & 1 \end{bmatrix}, \quad g_2^{e,z} = \begin{bmatrix} 1 & 1 & 2 & 7 & 8 \\ 1 & 1 & 2 & 7 & 8 \\ 2 & 2 & 5 & 8 & 11 \\ 3 & 3 & 4 & 9 & 10 \\ 4 & 4 & 6 & 10 & 12 \end{bmatrix},
\] (575)
while in dimension $m = 3$ we have, e.g.,

$$g^{e, z}_1 = \begin{bmatrix} 1 & 8 & 7 & 7 & 2 & 1 & 1 & 2 & 1 & 1 \\ 4 & 12 & 10 & 10 & 6 & 4 & 4 & 6 & 4 & 4 \\ 3 & 10 & 9 & 9 & 4 & 3 & 3 & 4 & 3 & 3 \\ 3 & 10 & 9 & 9 & 4 & 3 & 3 & 4 & 3 & 3 \\ 2 & 11 & 8 & 8 & 5 & 2 & 5 & 2 & 5 & 2 \\ 1 & 8 & 7 & 7 & 2 & 1 & 1 & 2 & 1 & 1 \\ 1 & 8 & 7 & 7 & 2 & 1 & 1 & 2 & 1 & 1 \\ 2 & 11 & 8 & 8 & 5 & 2 & 5 & 2 & 5 & 2 \\ 1 & 8 & 7 & 7 & 2 & 1 & 1 & 2 & 1 & 1 \\ 1 & 8 & 7 & 7 & 2 & 1 & 1 & 2 & 1 & 1 \\ \end{bmatrix}. \quad (576)$$

A combinatorial argument shows that the number of entries equal to 1, 2, 3, etc., in each matrix $g^{e, z}_1, g^{e, z}_2, \ldots$, $g^{e, z}_m$ is the same (for fixed $m$). For instance, in (575) we have 4 ones, 2 threes, 3 fours, 1 five, etc. This is very useful when we break the sum in $e$ and $z$ in (568) and (569) into multiple sums, and use the associative property of the tensor product to reduce the number of operations. By using the map $g^{e, z}_q$ we can immediately identify each matrix $E^{e, z}_q$. For example, in the case $m = 2$ we have (see Eq. (575))

$$[E^{11}_1]_{sh} = \int_{-b}^{b} \phi_s(a) \phi_h(a) da, \quad [E^{12}_1]_{sh} = \int_{-b}^{b} a \frac{d \phi_s(a)}{da} \phi_h(a) da, \quad \ldots. \quad (577)$$

**Summary of the Algorithm** We first compute all integrals in (574) and store them in $12 \times 12$ matrices $Q \times Q$, $Q$ being the number of degrees of freedom in each variable (e.g., collocation points of Fourier modes). We also set up the map between such set of matrices and any element of the tensor (570). Such map basically takes the number of degrees of freedom in each variable (e.g., collocation points of Fourier modes). We also identify which integral in the set (557) corresponds to the tensor entry in (570). In a matrix setting, this basically allows us to efficiently compute each tensor $E^{e, z}_q$, appearing in (568) and (569). Next, we compute the canonical tensor decomposition of the initial condition $f_0(a_1, \ldots, a_m)$, by applying the methods we described in Section 3.3.1. This gives us the set of vectors $\{\beta_1(t_0), \ldots, \beta_m(t_0)\}$. With such vectors available, we can build the matrices $M^L_1$ and $M^R_1$ defined in (568) and (569). To this end, we need an initial guess for $\{\beta_1(t_1), \ldots, \beta_m(t_1)\}$ which we can take to be equal to $\{\beta_1(t_0), \ldots, \beta_m(t_0)\}$, or a small random perturbation of it. With $M^L_1$ and $M^R_1$ in place and set, we can solve the linear system (567) and update $\beta_1(t_1)$. At this point we recompute $M^L_2$ and $M^R_2$ (with the updated $\beta_1(t_1)$) and solve for $\beta_2(t_1)$. We repeat this process for $q = 3, \ldots, m$ and iterate over and over among all variables until convergence. Parallel versions of the ALS algorithm were recently proposed by Karlsson et al. in [100].

### 7.2.3 Long-Term Integration

A rigorous error analysis of the HT and CP-ALS algorithms to solve the multivariate PDE (555) goes beyond the scope of this report (see [8] for a recent account). It is useful, however, to point out a few things on the nature of the discretization error, in particular on how the temporal local truncation error depends on the dimension $m$. To this end, let us first recall that the local truncation error at time $t_{n+1}$ of the second-order Adams Bashforth (AB2) scheme applied to the linear PDE (555) is $5\Delta t^3 L^3 f(\eta, a)/12$, where $\eta$ is some time instant between $t_n$ and $t_{n+1}$, $f(\eta, a)$ is the exact solution (515), and $L$ is defined in (556). The operator
Figure 37: Norm of $L^3 f$ at initial time versus $m$. The operator $L^3$, defined \((579)\), has been discretized in space by using an accurate Fourier spectral method with $N = 600$ points in each dimension. It is seen that the norm of $L^3 f_0$ grows faster than $m^6$. This has important consequences on the local truncation error generated by AB2 time-integrator applied to the PDF \((555)\).

$L^2$ can be explicitly written as

$$L^2 = \sum_{i,j,l,p=1}^{m} x_i C_{pl} \frac{\partial}{\partial a_p} \left( x_j C_{ij} \frac{\partial}{\partial a_j} \right),$$

while $L^3$ has the form

$$L^3 = \sum_{i,j,l,p,q,z=1}^{m} x_q C_{zq} C_{pl} C_{ij} \delta_{lq} \left( \delta_{jp} \frac{\partial}{\partial a_j} + x_j \frac{\partial^2}{\partial a_j \partial a_p} \right) + \sum_{i,j,l,p,q,z=1}^{m} x_q x_i C_{zq} C_{pl} C_{ij} \left( \delta_{jp} \frac{\partial^2}{\partial a_j \partial a_z} + \delta_{jz} \frac{\partial^2}{\partial a_j \partial a_p} + x_j \frac{\partial^3}{\partial a_j \partial a_p \partial a_z} \right). \quad (579)$$

From the last expression it is clear that any inaccuracy in the computation of the derivatives adds up to the temporal truncation error with at least with factor $m^6$. Indeed, as shown in Figure 37 the norm of $L^3 f$ grows faster than $m^6$. This has important consequences on the accuracy attainable with the AB2 time-integration scheme applied to high-dimensional linear PDEs. In fact, suppose that $\| L^3 f \|$ grows like $m^6$ not just at initial time but at each time step (it actually grows faster than that), i.e.,

$$\| L^3 f \| \sim \kappa m^6, \quad (580)$$

where $\kappa$ is a suitable constant. If we want the AB2 scheme to operate at constant truncation error for different number of dimensions $m$ then we need to guarantee that

$$m_1^6 \Delta t_1^3 = m_2^6 \Delta t_2^3. \quad (581)$$
For example, if $m_1 = 2$ and $m_2 = 10$ then we have

$$\Delta t_2 = \frac{\Delta t_1}{25}. \quad (582)$$

In other words, if we want our time integrator to operate at constant truncation error, then we need to run the simulation in 10 dimensions with a time step that is roughly 25 times smaller than the one we employ in the simulation in 2 dimensions. This can tax the computational resources quite substantially. In fact, suppose we are interested in integrating our PDE up to $T = 1$, and we set $\Delta t_1 = 10^{-4}$ in 2 dimensions, i.e., $10^4$ time steps. Assuming that the local truncation error is the same at each time step (see Eq. (580)), at the end of the integration period we accumulated an error of approximately $10^{-6}\kappa$. The same error is roughly attained at $T = 0.04$ if we integrate the PDE in 10 dimensions. In fact, the time step $\Delta t_2 = \Delta t_1/25$ guarantees a constant local truncation error which adds up to $10^{-6}\kappa$ after just 0.04 time units. The local truncation error manifest itself as numerical diffusion which eventually dissipates the numerical solution to zero. Note that in this simple calculation we did not take into account the accuracy of the rank reduction process in the CP-ALS and HT algorithms, which takes place at each time step.

### 7.3 Numerical Results

We solve the multivariate PDE (555) in the hypercube $[-b, b]^m$, with $b = 60$ and variable $m$. Such domain is chosen large enough to accommodate periodic (zero) boundary conditions in the integration period of interest. We study both the HT (hierarchical Tucker) and the CP-ALS schemes we discussed in Section 7.2.1 and Section 7.2.2. Specifically, we implemented a Fourier collocation method with 600 nodes in each variable and explicit AB2 time stepping. To study the accuracy of the numerical solution, we consider the time-dependent relative error

$$\epsilon_m(t) = \frac{|F(\theta_m^*, t) - \hat{F}(\theta_m^*, t)|}{|F(\theta_m^*, t)|}, \quad (583)$$

where $F$ is the analytical solution (540), $\hat{F}$ is the numerical solution we obtained by using the CP-ALS or the HT algorithms in the test function space $D_m$ (i.e., $m$ with independent variables) and with separation rank $r$. The test function $\theta^*$ in (583) is defined as

$$\theta_m^*(x) = h \sum_{j=1}^m \varphi_j(x), \quad h = 0.698835274542439, \quad (584)$$

where $\varphi_j(x)$ are the orthonormal polynomials shown in Figure 32. The accuracy of the CP-ALS and the HT algorithms is studied in Figure 38 where we plot the relative pointwise error (583) for different separation ranks $r$ and for different number of dimensions. It is seen that, as expected, the accuracy of the numerical solution increases as we increase the separation rank. Also, as we increase the number of dimensions from 2 to 6 the relative error increases, in agreement with the results of Section 7.2.3 (we employ a constant $\Delta t = 2.55 \times 10^{-4}$ in all our simulations). It is worthwhile emphasizing that the CP-ALS is a randomized algorithm which requires initialization at each time-step. This means that results of simulations with the same nominal parameters may be different. On the other hand, tensor methods based on multivariate/distributed singular value decomposition, such as the hierarchical Tucker decomposition [77], do not suffer from this issue. The CP-ALS algorithm is faster than HT but, as we just said, accuracy control may be an issue. The separation rank of both the CP-ALS and HT algorithms are computed adaptively up to the maximum value $r_{max}$ specified in the legend of Figure 38. Note that in two dimensions the CP-ALS algorithm results in error plots that looks very similar when $r_{max} = 8$ and $r_{max} = 12$. This is because the separation rank is less than 8 in both cases throughout the simulation up to $t = 1$. The variability of the results is related to the
Figure 38: Accuracy of the numerical solution to the FDE (504) in the finite dimensional function space (539) for different number of dimensions. Specifically, we plot the relative $L_\infty$ error (583) versus time we obtained by using the CP-ALS and HT algorithms we discussed in Section 7.2. The separation rank of both CP-ALS and HT is computed adaptively at each time step up to the maximum value $r_{\text{max}}$. 

\[ r_{\text{max}} = 4 \quad \text{and} \quad r_{\text{max}} = 8 \quad \text{and} \quad r_{\text{max}} = 12 \]
random initialization required by the ALS algorithm at each time step. On the other hand, in three and six dimensions the error plots we obtain for $r_{m \alpha x} = 4$ and $r_{m \alpha x} = 8$ are of the same order of magnitude because such separation ranks are achieved after just few time steps and they are not sufficient to accurately represent the multivariate solution. A similar phenomenon attributable to the separation rank is observed in the HT simulations. In particular, the time instant at which the HT tensor series requires a separation rank higher than 8 can be clearly identified, i.e., at $t = 0.2$ in two dimensions, at $t = 0.1$ in three dimensions, and at $t = 0.05$ in six dimensions. In Figure 39 we plot the time evolution of the first three CP modes $G_{16}^1$, $G_{16}^2$ and $G_{16}^3$ (see equation (565)) representing the dynamics of the solution functional in the variable $a_6 = (\theta, \varphi_6)$.

Remark: If $\theta(x)$ is not in the function space $D_m$ (see Eq. (539)), but can be represented in $D_m$ with accuracy then the solution to the multivariate PDE (555) provides an approximation of the solution to the full FDE (504) at such $\theta(x)$. For instance, consider the following test function

$$\theta(x) = \sin(x).$$

(585)

The Fourier coefficients of $\theta$ relative to the orthonormal basis shown in Figure 32 are plotted in Figure 40. Clearly $\theta(x)$ is not in $D_6$, but it can be approximated well in $D_6$. This means that the solution to the FDE at $\sin(x)$ can be approximated by the solution of the six-dimensional PDE arising when we evaluate the FDE (504) $D_6$.

Computing Functional Derivatives In Figure 41 we compare the exact first-order functional derivative (545) at $t = 0.4$ with the numerical approximation we obtained with the HT algorithm in $m = 3$ and $m = 6$ dimensions. The separation rank of the tensor series expansions is set to $r = 12$. As expected, the numerical approximation converges to the exact solution as we increase the number of dimensions.
7.4 The Navier-Stokes-Hopf Functional Equation

In this Section we discuss approximation of the Navier-Stokes-Hopf functional equation

\[
\frac{\partial \Phi(\theta, t)}{\partial t} = \sum_{k=1}^{3} \int_{V} \theta_k(x) \left[ i \sum_{j=1}^{3} \frac{\partial}{\partial x_j} \left( \frac{\delta^2 \Phi(\theta, t)}{\delta \theta_k(x) \delta \theta_j(x)} \right) \right] dx. \tag{586}
\]

In this formulation, \( V \) is a periodic box and \( \theta(x) \) is chosen in a divergence-free space of test functions. The main advantage of using such divergence-free space is that the pressure term drops out, just as in the classical Navier-Stokes equations\(^{39}\). To develop a discretization of the Hopf equation (586) it is convenient to first address the question of how to represent divergence-free spaces of periodic functions in 2D and 3D.

7.4.1 Symmetries of the Solution Functional

The divergence-free constraint in the velocity field induces a certain number of symmetries in the Hopf functional. Let first assume that the flow develops in a bounded region \( V \), i.e., \( u \cdot \hat{n} = 0 \), where \( \hat{n} \) is the outward unit vector orthogonal to the boundary of \( V \).

\(^{39}\)We recall that the pressure functional in the Navier-Stokes-Hopf equation is defined as (see \(^{145}\), p. 749)

\[
\Pi(\theta, x, t) = \left\langle p(x, t; \omega) \exp \left[ i \int_{V} \theta(x) \cdot u(x, t; \omega) dx \right] \right\rangle.
\]

Therefore,

\[
\int_{V} \theta \cdot \left\langle \nabla p \exp \left[ i \int_{V} \theta \cdot u dx \right] \right\rangle dx = \left\langle \int_{V} \theta \cdot \nabla p dx \exp \left[ i \int_{V} \theta \cdot u dx \right] \right\rangle.
\]

If \( \theta \) is divergence-free, i.e., \( \nabla \cdot \theta = 0 \) in \( V \), and satisfies the tangency condition \( \theta \cdot \hat{n} = 0 \) (\( \hat{n} \) is the outward unit vector orthogonal to the boundary of \( V \)) we have

\[
\int_{V} \theta(x) \cdot \nabla p(x, t; \omega) dx = \int_{\partial V} p(x, t; \omega) \theta(x) \cdot \hat{n} dx = 0. \tag{589}
\]

The integral (589) is zero also if \( \theta \) is divergence-free and the domain \( V \) is a periodic box. In fact, in this case \( \theta(x) \) is periodic, \( p \) is periodic, and therefore the integral along the boundary \( \partial V \) vanishes.
The first-order functional derivative of the solution functional (541) at $t = 0.4$, evaluated at the test functions $\theta(x) = \sin(x)$ and $\theta(x) = xe^{-x^2/4}$. We plot the exact result (545) and the numerical approximation (548) we obtained in the function spaces $D_3$ and $D_6$ (see Eq. (539)). The numerical approximation is computed by solving the multivariate PDE (509) in three and six dimensions.

outward unit vector normal to the boundary of $V$. Such region could be any volume co-moving with the fluid. In this assumption,

$$
\Phi([\theta + \nabla \varphi]) = \Phi([\theta])
$$

(590)

for any $\theta(x)$ and any $\varphi(x)$. In fact,

$$
\int_V \mathbf{u} \cdot \nabla \varphi \, dx = \int_{\partial V} \varphi \mathbf{u} \cdot \mathbf{n} \, dx = 0.
$$

(591)

If the boundary conditions assigned to $\mathbf{u}$ on $\partial V$ (boundary of $V$) are different from $\mathbf{u} \cdot \mathbf{n} = 0$, then (590) is still valid, but not for any $\varphi(x)$. For example, if $\mathbf{u}$ is not orthogonal to $\mathbf{n}$ along some part of the boundary $\partial V$, then it is sufficient to set $\varphi = 0$ along such boundary and use the fact that $\nabla \cdot \mathbf{u} = 0$ to conclude that (590) is still valid. Now let us consider the Helmholtz decomposition

$$
\theta = \eta + \nabla \varphi,
$$

(592)

where $\nabla \cdot \eta = 0$ in the domain $V$, and $\varphi = 0$ at the boundary $\partial V$. By taking the divergence of (592) we conclude that $\varphi$ satisfies the Dirichlet boundary value problem

$$
\nabla^2 \varphi = \nabla \cdot \theta \quad \text{(in $V$)}, \quad \varphi = 0 \quad \text{(in $\partial V$)},
$$

(593)

which has a solution. With the decomposition (592) available, we have

$$
\Phi([\eta + \nabla \varphi]) = \Phi([\eta]).
$$

(594)

In fact,

$$
\int_V (\eta + \nabla \varphi) \cdot \mathbf{u} \, dx = \int_V \eta \cdot \mathbf{u} \, dx + \int_V \nabla \varphi \cdot \mathbf{u} \, dx.
$$

(595)

However, by applying the Gauss theorem (recall that $\nabla \cdot \mathbf{u} = 0$)

$$
\int_V \nabla \varphi \cdot \mathbf{u} \, dV = \int_{\partial V} \varphi \mathbf{u} \cdot \mathbf{n} \, dx = 0
$$

(596)
Therefore (594) holds. The boundary conditions for \( \eta = \theta - \nabla \varphi \) depend on the gradient of the solution to (593) at the boundary \( \partial V \). Next, consider the unique Helmholtz-Hodge decomposition (see [20] or [34], p. 36) of the field \( \theta \) in the form (592), where \( \nabla \cdot \eta = 0 \) and \( \eta \) is tangent to \( \partial V \), i.e., \( \eta \cdot \hat{n} = 0 \) (\( \hat{n} \) outward unit vector normal to \( \partial V \)). If there is no flow across the boundary \( \partial V \) (\( u \cdot \hat{n} = 0 \)), or if the boundary conditions of \( \varphi \) are chosen such that the integral at the right hand side of (596) is zero then we have the symmetry (594).

For example, if \( \varphi \) is constant along the boundary then the divergence-free requirement on \( u \) implies that the integral at the right hand side of (596) is zero. Also, if \( \theta \) and \( u \) are periodic on a box then \( \varphi \) is periodic and the boundary integral in (596) is zero. In fact, the field \( \varphi \) arising from the Helmoltz-Hodge decomposition of \( \theta \) is the unique solution to the Neuman problem (see [34], p. 36)

\[
\nabla^2 \varphi = \nabla \cdot \theta \quad \text{(in } V), \quad \frac{\partial \varphi}{\partial n} = \theta \cdot \hat{n} \quad \text{(in } \partial V). \tag{597}
\]

Clearly, if \( \theta \) is periodic then \( \varphi \) is periodic and therefore \( \eta \) is periodic too. In other words, If we evaluate \( \Phi \) on a space of periodic functions then we have the invariance

\[
\Phi([\theta]) = \Phi([\eta]), \tag{598}
\]

where \( \eta \) is periodic and divergence free (\( \eta \) is the divergence-free part arising from the unique Helmholtz-Hodge decomposition of \( \theta \).

### 7.4.2 Divergence-Free Function Spaces

There has been a significant research activity in identifying bases for divergence-free spaces of functions. For example, Deriaz and Perrier [37, 38] have developed an effective algorithm to construct divergence-free and curl-free wavelets in 2D and 3D with various types of boundary conditions. Other divergence-free bases can be constructed in terms of radial basis functions [240], trigonometric polynomials [200], or eigenvalue problems with appropriate boundary conditions conditions [226, 169]. Hereafter we discuss how to construct a divergence-free basis for two-dimensional periodic flows. To this end, we consider the tensor product basis

\[
\psi_n(x, y) = l_{j(n)}(x)l_{i(n)}(y), \tag{599}
\]

where \( j(n) \) and \( i(n) \) are suitable sequences of integer number while \( l_k(x) \) are trigonometric polynomials. Any scalar-valued periodic function on the square (such as the streamfunction) can be represented as

\[
\Psi(x, y) = \sum_{k=1}^{M} \alpha_k \psi_j(x, y). \tag{600}
\]

Next define the divergence-free basis

\[
\Gamma_k(x, y) = \left( \frac{\partial \psi_k(x, y)}{\partial y}, -\frac{\partial \psi_k(x, y)}{\partial x} \right). \tag{601}
\]

It is clear that each basis element \( \Gamma_k \) is divergence-free by construction, i.e., \( \nabla \cdot \Gamma_k = 0 \). However, the basis (601) is not orthogonal nor normalized relative to the \( L_2([-b, b]^2) \) inner product

\[
(\Gamma_i, \Gamma_j) = \int_{-b}^{b} \int_{-b}^{b} \Gamma_i \cdot \Gamma_j dxdy,
\]

\[
= \int_{-b}^{b} \int_{-b}^{b} \left( \frac{\partial \psi_i}{\partial y} \frac{\partial \psi_j}{\partial y} + \frac{\partial \psi_i}{\partial x} \frac{\partial \psi_j}{\partial x} \right) dxdy.
\]

117
To generate a divergence-free and orthonormal basis one could use the Gram-Schmidt orthogonalization. As we shall see hereafter, such procedure preserves the divergence-free character of each basis element. We first normalize $\Gamma_1$ as

$$
\hat{\Gamma}_1(x, y) = \frac{\Gamma_1(x, y)}{\|\Gamma_1(x, y)\|}, \quad \|\Gamma_1\| = \sqrt{\langle \Gamma_1, \Gamma_1 \rangle}.
$$

(602)

Clearly, $\nabla \cdot \hat{\Gamma}_1 = 0$ since the $L_2$ norm of $\Gamma_1$ is just a real number. Next, define

$$
\Theta_2 = \Gamma_2 - (\Gamma_2, \hat{\Gamma}_1)\hat{\Gamma}_1, \quad \hat{\Theta}_2(x, y) = \frac{\Theta_2(x, y)}{\|\Theta_2(x, y)\|}.
$$

(603)

As before, $\nabla \cdot \hat{\Theta}_2 = 0$. Moreover $\hat{\Theta}_2$ is orthogonal to $\hat{\Gamma}_1$, i.e., $(\hat{\Theta}_2, \hat{\Gamma}_1) = 0$. The algorithm proceeds with the computation of

$$
\Theta_3 = \Gamma_3 - (\Gamma_3, \hat{\Theta}_2)\hat{\Theta}_2 - (\Gamma_3, \hat{\Gamma}_1)\hat{\Gamma}_1, \quad \hat{\Theta}_3(x, y) = \frac{\Theta_3(x, y)}{\|\Theta_3(x, y)\|}.
$$

(604)

Thanks to the fact that $\Gamma_3$, $\hat{\Gamma}_2$ and $\hat{\Gamma}_1$ are divergence free, we have that $\nabla \cdot \hat{\Theta}_3 = 0$. Moreover $\hat{\Theta}_3$ is orthogonal to both $\hat{\Theta}_2$ and $\hat{\Gamma}_1$. In other words, $\{\hat{\Theta}_1, \hat{\Theta}_2, \hat{\Theta}_3\}$ is a divergence-free orthonormal system. Proceeding in a similar way, we can construct the divergence-free orthonormal basis we were looking for, and define the following finite-dimensional divergence-free space of functions

$$
D_M = \text{span}\{\Gamma_1, \cdots, \Gamma_M\}.
$$

(605)

An element of $D_M$ is in the form

$$
\theta(x, y) = \sum_{k=1}^{M} a_k \hat{\Gamma}_k(x, y).
$$

(606)

### 7.4.3 Analytical Solution to the Characteristic Function Equation

A substitution of (606) into the Navier-Stokes-Hopf equation (586) yields the multivariate (complex-valued) PDE\textsuperscript{40}

$$
\frac{\partial \phi}{\partial t} = i \sum_{p, j, k=1}^{M} A_{pjk} a_p \frac{\partial^2 \phi}{\partial a_k \partial a_j} + \nu \sum_{k, p=1}^{M} a_p B_{pk} \frac{\partial \phi}{\partial a_k},
$$

(609)

where

$$
B_{pk} = \int_{V} \hat{\Gamma}_p \cdot \nabla^2 \hat{\Gamma}_k d\mathbf{x}, \quad A_{pjk} = \int_{V} \hat{\Gamma}_p \cdot \left[ \left( \hat{\Gamma}_k \cdot \nabla \right) \hat{\Gamma}_j \right] d\mathbf{x}.
$$

(610)

\textsuperscript{40}We emphasize that equation (609) has exactly the same structure as the characteristic function equation we obtain for the one-dimensional Burgers equation. To show this, it is sufficient to discretize the Burgers-Hopf equation

$$
\frac{\partial \phi(\theta, t)}{\partial t} = \int_{-b}^{b} \theta(x) \left[ \frac{\partial}{\partial x} \frac{\partial^2 \Phi(\theta, t)}{\partial \theta(x)^2} + \nu \frac{\partial^2}{\partial x^2} \frac{\partial \Phi(\theta, t)}{\partial \theta(x)} \right] dx,
$$

(607)

in the finite-dimensional space $D_m$ of periodic functions in $[-b, b]$. To this end, consider the series expansion

$$
\theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x).
$$

(608)

where $\varphi_k(x)$ are orthonormal trigonometric polynomials. Substituting (608) into (607) and evaluating all functional derivatives in $D_m$ yields an equation in the form (609).
By using integration by parts we can simplify $B_{pk}$ to

$$B_{pk} = \int_V \left[ \hat{\Gamma}_p^{(x)} \left( \frac{\partial^2 \hat{\Gamma}_k^{(x)}}{\partial x^2} + \frac{\partial^2 \hat{\Gamma}_k^{(x)}}{\partial y^2} \right) + \hat{\Gamma}_p^{(y)} \left( \frac{\partial^2 \hat{\Gamma}_k^{(y)}}{\partial x^2} + \frac{\partial^2 \hat{\Gamma}_k^{(y)}}{\partial y^2} \right) \right] dx,$$

$$= - \int_V \left[ \frac{\partial \hat{\Gamma}_p^{(x)}}{\partial x} \frac{\partial \hat{\Gamma}_k^{(x)}}{\partial x} + \frac{\partial \hat{\Gamma}_p^{(x)}}{\partial y} \frac{\partial \hat{\Gamma}_k^{(x)}}{\partial y} + \frac{\partial \hat{\Gamma}_p^{(y)}}{\partial x} \frac{\partial \hat{\Gamma}_k^{(y)}}{\partial x} + \frac{\partial \hat{\Gamma}_p^{(y)}}{\partial y} \frac{\partial \hat{\Gamma}_k^{(y)}}{\partial y} \right] dx. \quad (611)$$

Similarly, the coefficients $A_{pjk}$ can be written as

$$A_{pjk} = \int_V \left[ \Gamma_p^{(x)} \left( \frac{\partial \Gamma_k^{(x)}}{\partial x} + \frac{\partial \Gamma_k^{(y)}}{\partial y} \right) + \Gamma_p^{(y)} \left( \frac{\partial \Gamma_k^{(y)}}{\partial x} + \frac{\partial \Gamma_k^{(y)}}{\partial y} \right) \right] dx. \quad (612)$$

By taking the inverse Fourier transform of the characteristic function equation (609) we obtain

$$\frac{\partial p(u,t)}{\partial t} + \sum_{p=1}^M \frac{\partial}{\partial u_p} \left[ \left( \nu \sum_{k=1}^M u_k B_{pk} - \sum_{k,j=1}^M u_k u_j A_{pkj} \right) p(u,t) \right] = 0. \quad (613)$$

This is the multivariate first-order PDE that governs the evolution of the finite-dimensional approximation of the probability density functional $\Phi$. The formal solution to (613) is

$$p(u,t) = p_0(U(t,u)) \exp \left( - \int_0^t \nabla \cdot G(u(\tau,U)) d\tau \right), \quad (614)$$

where

$$G_p(u) = \nu \sum_{k=1}^M u_k B_{pk} - \sum_{k,j=1}^M u_k u_j A_{pkj}, \quad p = 1, \ldots, M. \quad (615)$$

The flow map $u(t,U)$ and its inverse $U(t,u)$ are defined by the solution to the ODE system

$$\frac{du}{dt} = G(u), \quad u(0) = U. \quad (616)$$

By taking the Fourier transform of (614), we obtain the following analytical solution to the discretized Navier-Stokes-Hopf equation (609)

$$\phi(a,t) = \int_{-\infty}^\infty \cdots \int_{-\infty}^\infty e^{iu \cdot a} p_0(U(t,u)) \exp \left( - \int_0^t \nabla \cdot G(u(\tau,U)) d\tau \right) du. \quad (617)$$

Numerical methods to compute the flow map $u(t,U)$ and its inverse $U(t,u)$ in equations (614) and (617) often rely on ray tracing, i.e., numerical solutions to the (616) on a Cartesian mesh of initial conditions. Such methods are not efficient in high-dimensions [247], and may be improved significantly by representing flow maps using, e.g., tensor formats.

$^{41}$Equation (613) can be obtained by evaluating the Navier-Stokes probability density functional equation [46, 163] in the finite-dimensional test function space (605).
**Functional Derivatives**  The first-order functional derivatives of the Hopf characteristic functional (evaluated in the space of divergence-free functions) can be approximated as

$$\frac{\delta \Phi([\theta], t)}{\delta \theta(x)(x)} = \sum_{k=1}^{M} \frac{\partial \Phi}{\partial a_k} \hat{\Gamma}_k(x), \quad \frac{\delta \Phi([\theta], t)}{\delta \theta(y)(y)} = \sum_{k=1}^{M} \frac{\partial \Phi}{\partial a_k} \hat{\Gamma}_k(y).$$

(618)

Similarly, the second-order derivatives can be written as

$$\frac{\delta^2 \Phi([\theta], t)}{\delta \theta(x)(x) \delta \theta(x)(y)} = \sum_{k,j=1}^{M} \frac{\partial^2 \Phi}{\partial a_k a_j} \hat{\Gamma}_k(x) \hat{\Gamma}_j(y),$$

(619)

$$\frac{\delta^2 \Phi([\theta], t)}{\delta \theta(y)(x) \delta \theta(y)(y)} = \sum_{k,j=1}^{M} \frac{\partial^2 \Phi}{\partial a_k a_j} \hat{\Gamma}_k(x) \hat{\Gamma}_j(y),$$

(620)

$$\frac{\delta^2 \Phi([\theta], t)}{\delta \theta(x)(x) \delta \theta(y)(y)} = \sum_{k,j=1}^{M} \frac{\partial^2 \Phi}{\partial a_k a_j} \hat{\Gamma}_k(x) \hat{\Gamma}_j(y).$$

(621)

Evaluating these derivatives at $\theta = 0$ is equivalent to evaluate the derivatives of the characteristic function at $(a_1, ..., a_M) = (0, ..., 0)$. This yields, the following representation of the mean and cross correlation of the velocity field

$$\langle u(x, t) \rangle = \sum_{k=1}^{M} \frac{\partial \phi(a, t)}{\partial a_k} \bigg|_{(0, ... ,0)} \hat{\Gamma}_k(x),$$

(622)

$$\langle u(x)(x)u(y)(y) \rangle = \sum_{k,j=1}^{M} \frac{\partial^2 \phi(a, t)}{\partial a_k a_j} \bigg|_{(0, ... ,0)} \hat{\Gamma}_k(x) \hat{\Gamma}_j(y).$$

(623)

**Computational Complexity**  The number of dimensions $M$ appearing in the characteristic function equation (609) and the joint PDF equation (613) coincides with the number of number of degrees of freedom we employ in the discretization of the velocity field. For instance, if we consider the classical two-dimensional Kolmogorov flow [129] represented on a $128 \times 128$ Fourier basis, then $M = 16384 (128^2)$. Computing the solution to such high-dimensional linear PDEs (i.e., (609) or (613)) obviously requires parallel algorithms and a highly-efficient tensor methods. If we employ operator splitting in time, then we can easily take care of the linear part in (613) – i.e., the one depending linearly on $u_k$ – by integrating out the corresponding dynamics with the method of characteristics. To this end, let us first consider an orthonormal divergence-free basis that diagonalizes the matrix $B_{pk}$ in (610). Such basis exists and it can be computed by standard linear algebra techniques [251] (simultaneous diagonalization of two quadratic forms). Relative to the new basis, the joint PDF equation (613) can be written as

$$\frac{\partial p(u, t)}{\partial t} = -\sum_{p=1}^{M} \frac{\partial}{\partial u_p} \left[ \nu u_p B_{pp} - \sum_{j,k=1}^{M} u_k u_j A_{pkj} \right] p(u, t),$$

(624)

where, with some abuse of notation, we denoted by $B_{pp}$ and $A_{pkj}$ the entries of $B_{pk}$ and $A_{pkj}$ in (610) relative to the new basis. The last equation can be written in the operator form

$$\frac{\partial p(u, t)}{\partial t} = L(u)p(u, t) + Q(u)p(u, t),$$

(625)
where

\[ L(u) = - \sum_{p=1}^{M} \nu u_p B_{pp} \frac{\partial}{\partial u_p} - B_{pp} \nu - \sum_{j=1}^{M} u_j (A_{ppj} + A_{mp}) , \quad Q(u) = \sum_{k,j=1}^{M} u_k u_j A_{pkj} \frac{\partial}{\partial u_p}. \]  

(626)

Note that \( L(u) \) depends linearly on \( u \) while \( Q(u) \) is quadratic in \( u \). Both \( L \) and \( Q \) are separable linear operators. The formal solution to (625) is\(^{42}\)

\[ p(u, t) = e^{t(L(u) + Q(u))} p(u, 0). \]  

(627)

By using operator splitting in time, e.g., the classical second-order Strang splitting, we can write

\[ p(u, t_n) = e^{tL(u)/2} e^{tQ(u)} e^{tL(u)/2} p(u, t_{n-1}). \]  

(628)

Clearly, the action of the semigroup \( \exp[tL(u)] \) can be computed exactly since the flow map corresponding characteristic system associated with the first order linear PDE \( \dot{g} = Lg \) is trivial. Specifically, for any \( g_0 \) and any \( t \) we have

\[ e^{tL(u)} g_0(u) = g_0 \left( e^{-\nu B_{11}} u_1, ..., e^{-\nu B_{M}} u_M \right) \exp \left[ -t\nu \sum_{j=1}^{M} B_{jj} + \sum_{j=1}^{M} \int_{0}^{t} e^{\nu B_{jj}} u_j \sum_{p=1}^{M} (A_{ppj} + A_{mp}) \right]. \]  

(629)

The action of the semigroup \( \exp[tQ(u)] \) is much more complicated to compute. Indeed, the flow map of quadratic dynamical systems is, in general, not known explicitly. For small \( \Delta t \) one may introduce the approximation \( \exp[\Delta Q(u)] \simeq I + \Delta t Q(u) \) (or any higher-order one arising, e.g., from multi-step methods) and leverage on the separability of the operator \( Q(u) \) (rank \( M^3 \)).

**Acknowledgments**

This research was supported by the Air Force Office of Scientific Research grant FA9550-16-1-0092.

**A Functional Fourier Transform**

Functional Fourier transforms can be defined as a continuum limit of multi-variate Fourier transforms. To introduce this concept in a simple and intuitive way, let us consider the expression of the joint probability density function of \( n \) random variables in terms of the joint characteristic function

\[ p(a_1, ..., a_n) = \frac{1}{(2\pi)^{n/2}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} e^{-i(a_1 b_1 + \cdots + a_n b_n)} \phi(b_1, ..., b_n) db_1 \cdots db_n. \]  

(630)

If we think of \( a_i \) and \( b_i \) as values of two continuous functions \( a(x) \) and \( b(x) \) at locations \( x_i \in \mathbb{R} \), then it makes sense to consider what happens to \( p \) and \( \phi \) as we send the number of collocation points to infinity. In this limit, the joint probability density function \( p(a_1, ..., a_n) \) becomes a probability density functional \( P([a(x)]) \), while the joint characteristic function \( \phi(b_1, ..., b_n) \) becomes a (Hopf) characteristic functional. This allows us to define the functional Fourier transform as

\[ P([a(x)]) = \int e^{-i \int a(x)b(x) dx} \Phi([b(x)])[D[b(x)]], \]  

(631)

\(^{42}\)We recall that \( \exp[t(L + Q)] \) is the Frobenious-Perron operator associated with the quadratic dynamical system (616) (see 42).
where
\[ D[b(x)] = \lim_{n \to \infty} \frac{1}{(2\pi)^{n/2}} \prod_{i=1}^{n} db(x_i). \] (632)
Equation (631) establishes the connection between the Hopf functional and the probability density functional of a random field. It also allows us to define special functionals, such as the Dirac delta functional
\[ \delta[a(x)] = \int e^{-i \int a(x)b(x)dx} D[b(x)]. \] (633)
By using this definition, it is easy to show that the probability density functional can be expressed as an average of Dirac delta functional
\[ P([a(x)]) = \langle \delta[a(x) - u(x; \omega)] \rangle \]
\[ = \langle \int e^{-i \int a(x)b(x)dx + i \int u(x; \omega)b(x)dx} D[b(x)] \rangle \]
\[ = \int e^{-i \int a(x)b(x)dx} \left< e^{i \int u(x; \omega)b(x)dx} \right> D[b(x)] \]
\[ = \int e^{-i \int a(x)b(x)dx} \Phi([b(x)]) D[b(x)]. \]

B Evaluation of Functional Integrals

Functional integrals in the form
\[ \int_{D(F)} F([\theta]) W([\theta]) D[\theta] \] (634)
arise naturally in many branches of physics and mathematics, e.g., in quantum and statistical mechanics, field theory, quantum optics, solid state physics, and financial mathematics [109, 252, 15, 5]. Analytical results for functional integrals are available only in few exceptional cases, and therefore one usually has to resort to numerical approximation techniques [51, 177]. Perhaps, the most classical one is Monte Carlo (MC), which gives results as an ensemble average over a large number of realizations of the test function \( \theta(x) \), i.e.,
\[ \int F([\theta]) W[\theta] D[\theta] \simeq \frac{1}{N} \sum_{k=1}^{N} F([\theta_k(x)]). \] (635)
Each term in the series corresponds to a specific path \( \theta_k(x) \), which is drawn from the probability measure \( W([\theta]) \). Monte Carlo and quasi-Monte Carlo methods [40] have slow convergence rate and therefore they require significant computational resources to achieve good accuracy. In a renormalized perturbation theory setting [139], the functional integral can be expanded relative to a certain parameter and the terms in the series expansion have a structure determined by the order of the perturbation parameter. This yields well-know diagrammatic representations of functional (path) integrals, as a sum over all possible paths consistent with the order of the perturbation parameter [95, 109]. Alongside statistical methods and perturbation series expansions, deterministic techniques were also proposed to evaluate functional integrals. These include, path summation based on short time propagators [192], fast Fourier transforms [57], HDMR expansions [176] and functional approximation techniques [127]. An intuitive way to calculate functional integrals is to

43 In (634) \( F \) is a measurable functional on a complete metric space \( D(F) \) and \( W > 0 \) is a functional integral measure.
consider a finite-dimensional subset of $D_m \subset D(F)$, e.g., the linear span of the basis $\{\varphi_1, \ldots, \varphi_m\}$. In this setting, $\theta(x)$ admits the representation

$$
\theta(x) = \sum_{k=1}^{m} a_k \varphi_k(x), \quad a_k = (\theta, \varphi_k).
$$

(636)

If we substitute (636) into (634) we obtain the multivariate integral

$$
\int \cdots \int f(a_1, \ldots, a_m) w(a_1, \ldots, a_m) da_1 \cdots da_m,
$$

(637)

where

$$
f(a_1, \ldots, a_m) = F\left(\sum_{k=1}^{m} a_k \varphi_k(x)\right), \quad w(a_1, \ldots, a_m) = W\left(\sum_{k=1}^{m} a_k \varphi_k(x)\right).
$$

(638)

This form includes both “modal” and “nodal” discretizations of the functional integral (634). For example, if (636) is an interpolant through $m$ nodes along the $x$ axis, then $a_j = \theta(x_j)$ and (637) becomes

$$
\int \cdots \int f(\theta(x_1), \ldots, \theta(x_m)) w(\theta(x_1), \ldots, \theta(x_m)) d\theta(x_1) \cdots d\theta(x_m).
$$

(639)

In this formulation, we obtain the functional integral (634) as the limit of an infinite number of nodes $m \to \infty$. This is the viewpoint taken in quantum mechanics and field theory to define path integrals [109, 252], i.e., integrals over suitable trajectories of functions. From a mathematical viewpoint, the limiting procedure defining the functional integral measure in terms of an infinite products of elementary measures should be handled with care. In fact, the classical Lebesgue measure does not exist in spaces of infinite dimension [137]. On the other hand, Gaussian measures are still well defined in such setting. This is why we included $W(\theta)$ in (376). The argument leading to the result on non-existence of an analogue to the Lebesgue measure in infinite dimension is related to the argument showing that the Heine-Borel theorem does not hold in infinite-dimensional normed linear spaces.

**Remark** Roughly speaking, the meaning of $D[\theta]$ in (634) is: integrate over all possible degrees of freedom defining $\theta$ in the class of functions $D(F)$. If $\theta$ is in the span of a finite-dimensional basis (e.g., (636)), then there is no problem in replacing (640) with an integral over the range of the expansion coefficients, as we did in (637). A more tricky situation arises when $\theta$ is truly infinite-dimensional. In this case, the functional integral (634) is an integral over an infinite number of variables. To ensure convergence the functional integral measure $W(\theta)$ must be carefully chosen.

**B.1 Functional Integrals of Cylindrical Functionals**

Functional integrals involving cylindrical functionals (see Section 3.3) can be written in the general form

$$
\int_{D(f)} f((\theta, \varphi_1), \ldots, (\theta, \varphi_m)) W(\theta) D[\theta].
$$

(640)

**Example 1:** Consider the nonlinear functional

$$
f((\theta, \sin(x))) = \frac{1}{1 + (\theta, \sin(x))^2}, \quad (\theta, \sin(x)) = \int_0^{2\pi} \theta(x) \sin(x) dx,
$$

(641)
in the space of infinitely-differentiable periodic functions in $[0, 2\pi]$, i.e.,

$$\theta \in D(f) = \{ \theta \in C^\infty([0, 1]), \theta(0) = \theta(2\pi) \}. \quad (642)$$

Also, consider the functional integral measure

$$W([\theta]) = \kappa e^{-(\theta, \theta)^2}, \quad (643)$$

where $\kappa$ is a (possibly infinite) normalization constant. We expand $\theta$ in a Fourier series,

$$\theta(x) = a_0 + \sum_{k=1}^{\infty} a_k \sin(kx) + \sum_{k=1}^{\infty} b_k \cos(kx). \quad (644)$$

In this representation, the functional integral measure becomes

$$w(a_0, a_1, b_1, ...) = 2\sqrt{\pi} e^{-\pi^2 a_0^2} \lim_{j \to \infty} \prod_{k=1}^{j} \pi^k e^{-\pi^2 (a_k^2 + b_k^2)} \quad (645)$$

and the integral (640) can be computed analytically. The result is

$$\int_{D(f)} W([\theta]) \frac{1}{1 + (\theta, \sin(x))^2} \approx \int \cdots \int f(a_1, ..., a_N) w(a_1, ..., a_N) da_1 \cdots da_N, \quad (648)$$

where erf is the error function and $e$ is the Napier number.

An interesting question is whether integrals in the form (640) can be simplified and reduced to integrals over the range of $\{(\theta, \varphi_1), ..., (\theta, \varphi_m)\}$, i.e., a subset of $\mathbb{R}^m$. The answer to this question is, in general, negative, as have just shown in the previous example. However, if we pick $\theta$ in a finite-dimensional function space, e.g., spanned by the basis $\{\xi_1(x), ..., \xi_N(x)\}$, i.e.,

$$\theta(x) = \sum_{k=1}^{N} a_k \xi_k(x), \quad (647)$$

then the functional integral effectively reduces to a finite-dimensional integral over the range of the expansion coefficients $a_1, ..., a_N$. A substitution of (647) into (640) yields

$$\int_{D_N(f)} f ((\theta, \varphi_1), ..., (\theta, \varphi_m)) W([\theta]) D[\theta] \approx \int \cdots \int g(a_1, ..., a_N) w(a_1, ..., a_N) da_1 \cdots da_N, \quad (648)$$

where

$$g(a_1, ..., a_N) = f \left( \sum_{k=1}^{N} a_k \alpha_{1k}, ..., \sum_{k=1}^{N} a_k \alpha_{mk} \right), \quad \alpha_{ij} = (\xi_k, \varphi_j). \quad (649)$$

In particular, if the functions $\xi_k(x)$ coincide with $\varphi_k(x)$, and $\{\varphi_1, ..., \varphi_N\}$ is orthonormal with respect to the inner product $(,)$, then we obtain

$$\int_{D_N(f)} f ((\theta, \varphi_1), ..., (\theta, \varphi_m)) W([\theta]) D[\theta] = \int \cdots \int f (a_1, ..., a_m) w(a_1, ..., a_N) da_1 \cdots da_N. \quad (650)$$
If \( w(a_1, \ldots, a_N) \) is a probability density, then (650) implies that

\[
\int_{D_N(f)} f((\theta, \varphi_1), \ldots, (\theta, \varphi_m)) W(\theta) D[\theta] = \int \cdot \int f(a_1, \ldots, a_m) w(a_1, \ldots, a_m) da_1 \cdots a_m. \tag{651}
\]

The last integral can be computed if the measure \( w \) is separable, and \( f \) is represented in terms of a tensor decomposition, e.g., a canonical tensor series or an HDMR expansion (see Section 3.3 and Section 3.5).

C Derivation of Functional Differential Equations

In this Appendix we briefly discuss how to derive functional differential equations, and their equivalence to systems of PDEs, or PDEs in an infinite number of variables.

C.1 The Generating Functional Approach

The seminal work of Martin, Siggia and Rose [136] opened the possibility to apply quantum field theoretic methods, such as Feynman diagrams and Schwinger-Dyson equations, to classical and statistical physics. The key idea is to construct a generating functional (action functional) based on the equations of motion [96, 174, 225] and then apply a Heisenberg operator theory which parallels the Schwinger formalism of quantum field theory. This allows to obtain closed equations for quantities of interest such as the correlation functions and response functions (averaged Green’s functions). For a thorough description of the method see the excellent review paper of Jensen [96].

C.2 From PDEs to Functional Equations: The Method of Continuum Limits

In Section 4 we have discussed how derive probability density functional equations and Hopf functional equations as continuum limits of joint PDF equations and joint characteristic function equations, respectively. Such procedure is rather formal, but easy to follow and effective in many cases. More importantly, it can be applied to many different linear and nonlinear PDEs.

Example 1: Let us consider the Kuramoto-Sivashinsky equation

\[
\frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial u^2}{\partial x} + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^4 u}{\partial x^4} = 0 \tag{652}
\]

with random initial data \( u(x, 0; \omega) = u_0(x; \omega) \). In a collocation setting, the solution to (652) can be written as

\[
u(x, t; \omega) = \sum_{k=1}^{n} u_k(t; \omega) l_k(x), \tag{653}
\]

where \( u_k(t; \omega) = u(x_k, t; \omega) \) while \( l_j(x) \) are suitable basis functions, e.g., Lagrange characteristic polynomials through the points \( \{x_1, \ldots, x_n\} \). Substituting (653) into (652) and setting the residual at collocation points equal to zero yields the system

\[
\frac{d u_k}{d t} + \frac{1}{2} \sum_{j=1}^{n} D_{kj}^{(1)} u_j^2 + \sum_{j=1}^{n} \left( D_{kj}^{(2)} + D_{kj}^{(4)} \right) u_j = 0 \quad k = 1, \ldots, n \tag{654}
\]
where \(D_{kj}^{(1)}, D_{kj}^{(2)}\) and are first-, second-, and fourth-order differentiation matrices, respectively. The joint characteristic function of \(\{u_1, \ldots, u_n\}\) is

\[
\phi(\theta_1, \ldots, \theta_n, t) = \left\langle \exp \left[ i \sum_{j=1}^{n} \theta_j u_j(t; \omega) \right] \right\rangle
\]  
(655)

and it satisfies the evolution equation

\[
\frac{\partial \phi}{\partial t} = -\frac{i}{2} \sum_{k,j=1}^{n} \theta_k D_{kj}^{(1)} \left\langle u_j^2 \exp \left[ i \sum_{j=1}^{n} \theta_j u_j \right] \right\rangle - i \sum_{k,j=1}^{n} \theta_k (D_{kj}^{(2)} + D_{kj}^{(4)}) \left\langle u_j \exp \left[ i \sum_{j=1}^{n} \theta_j u_j \right] \right\rangle
\]

\[
= \frac{i}{2} \sum_{k,j=1}^{n} \theta_k D_{kj}^{(1)} \frac{\partial^2 \phi}{\partial \theta_j^2} - \sum_{k,j=1}^{n} \theta_k (D_{kj}^{(2)} + D_{kj}^{(4)}) \frac{\partial \phi}{\partial \theta_j}.
\]  
(656)

By taking the continuum limit of this equation, i.e., by sending \(n\) to infinity, we formally obtain

\[
\frac{\partial \Phi}{\partial t} = \int_{a}^{b} \theta(x) \left[ \frac{i}{2} \frac{\partial}{\partial x} \frac{\delta^2 \Phi(\theta)}{\delta \theta(x)^2} - \frac{\partial^2}{\partial x^2} \frac{\delta \Phi(\theta)}{\delta \theta(x)} - \frac{\partial^4}{\partial x^4} \frac{\delta \Phi(\theta)}{\delta \theta(x)} \right] dx.
\]  
(657)

The continuum-limit argument just invoked is formal and can be used for other finite-dimensional equations (see [232]). In the next Section we discuss the inverse operation, i.e., how to transform functional differential equations into PDEs with a finite number of variables.

### C.2.1 From Functional Equations to PDEs

The restriction of functionals and functional differential equations to finite-dimensional function spaces yields multivariate fields and multivariate PDEs, respectively. Such PDEs can be obtained directly by using the finite dimensional theory, i.e., functional equations on finite-dimensional function spaces are in toto equivalent to multivariate PDEs. To show this, let us consider the finite-dimensional Hilbert space

\[
D_m = \text{span}\{\varphi_1(x), \ldots, \varphi_m(x)\},
\]  
(658)

where \(\varphi_1, \ldots, \varphi_m\) are orthonormal basis functions. Let us represent the test function \(\theta(x)\) relative to such orthonormal basis

\[
\theta_m(x) = \sum_{k=1}^{m} a_k \varphi_k(x), \quad a_k = (\theta, \varphi_k)
\]  
(659)

Restricting the domain of the Hopf functional (327) to the finite-dimensional space (658) (see Figure 1) yields the joint characteristic function

\[
\phi(a_1, \ldots, a_m) = \left\langle e^{i(a_1 U_1(\omega) + \cdots + a_m U_m(\omega))} \right\rangle, \quad U_k(\omega) = \int_{a}^{b} u(x; \omega) \varphi_k(x) dx.
\]  
(660)

Note that the linear combination of \(U_k(\omega)\) yields a finite-dimensional approximation of \(u(x, \omega)\), i.e.,

\[
\widehat{u}(x; \omega) = \sum_{k=1}^{m} U_k(\omega) \varphi_k(x).
\]  
(661)
Based on this expansion, the mean field and the correlation function of \(\tilde{u}(x, \omega)\) can be easily expressed as

\[
\langle \tilde{u}(x, \omega) \rangle = \sum_{k=1}^{m} \langle U_k(\omega) \rangle \varphi_k(x) = \sum_{k=1}^{m} \frac{1}{i} \left. \frac{\partial \phi}{\partial a_k} \right|_0 \varphi_k(x),
\]

\[
\langle \tilde{u}(x, \omega) \tilde{u}(y; \omega) \rangle = \sum_{k,j=1}^{m} \langle U_k(\omega) U_j(\omega) \rangle \varphi_k(x) \varphi_j(x) = -\sum_{k,j=1}^{m} \left. \frac{\partial^2 \phi}{\partial a_k \partial a_j} \right|_0 \varphi_k(x) \varphi_j(x).
\]

(662)  

(663)

Similar equations hold for higher-order moments. Next we consider the restriction of functional equations to finite-dimensional function spaces.

Example 1: Evaluating the Hopf functional equation in the finite-dimensional space \(D_m\) yields a characteristic function equation that can be obtained directly by using simpler methods, e.g., by differentiating the characteristic function. To show this, consider the diffusion equation \(\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}\), in a bounded domain \([a, b]\), with Dirichlet boundary conditions and a random initial condition. The corresponding Hopf equation is

\[
\frac{\partial \Phi(\theta, t)}{\partial t} = \int_{a}^{b} \theta(x) \frac{\partial^2}{\partial x^2} \left[ \delta \Phi(\theta, t) \right] dx.
\]

(664)

Now, we restrict \(\theta\) to the finite-dimensional function space \(D_m\). Evaluating the Hopf functional for \(\theta \in D_m\) yields the multivariate characteristic function

\[
\phi(a_1, \ldots, a_m, t) = \Phi([\theta_m], t) \quad \theta_m \in D_m.
\]

(665)

A substitution of (55) and (665) into (664) yields

\[
\frac{\partial \phi}{\partial t} = \sum_{k,j=1}^{m} a_k \frac{\partial \phi}{\partial a_j} H_{kj}, \quad H_{kj} = \int_{a}^{b} \varphi_k(x) \frac{\partial^2 \varphi_j(x)}{\partial x^2} dx.
\]

(666)

This is an equation for the joint characteristic function of the coefficients \(\{U_k(\omega)\}_{k=1}^{m}\) defined in (660). Such equation can be obtained directly by using the semidiscrete (Galerkin) form of the diffusion equation

\[
\frac{du_j(t; \omega)}{dt} = \sum_{k=1}^{m} u_k(t; \omega) \int_{a}^{b} \varphi_k(x) \frac{\partial^2 \varphi_j(x)}{\partial x^2} dx \quad j = 1, \ldots, m.
\]

(667)

Indeed, by differentiating the joint characteristic function of \(\{u_1, \ldots, u_m\}\) we obtain

\[
\frac{\partial \phi}{\partial t} = \sum_{k=1}^{m} i \left. \frac{du_k}{dt} \exp \left[ i \sum_{j=1}^{m} u_k(t; \omega) a_k \right] \right|,
\]

\[
= \sum_{k,j=1}^{m} a_k H_{kj} \frac{\partial \phi}{\partial a_j},
\]

(668)

which coincides with (666).

C.2.2 From Functional Equations to Systems of PDEs

In this Section we discuss the connection between functional differential equations and systems of PDEs. The main idea is to evaluate the FDE on function spaces of increasing dimension (see, e.g., Section 3.2.1). To
such equation may be obtained from (664). We have seen already that evaluating the Hopf functional at \( \theta(x) = \alpha \delta(y - x) \) yields the one-point characteristic function of \( u(y, t; \omega) \)

\[
\Phi(\theta, t) = \left\langle e^{i \alpha \int_0^b u(x,t;\omega)\delta(y-x)dx} \right\rangle = \left\langle e^{i \alpha u(y,t;\omega)} \right\rangle.
\] (669)

Similarly, if we consider \( \theta(x) \in \text{span}\{\delta(y_1 - x), ..., \delta(y_m - x)\} \) then we have that \( \Phi(\theta, t) \) coincides with the joint characteristic function of \{\( u(y_k, t; \omega) \)\}_{k=1,...,m}. Now, let us see what happens to the Hopf equation when we evaluate it at \( \theta(x) = \alpha_1 \delta(y_1 - x) \). We have

\[
\frac{\partial \phi(\alpha_1, y_1, t)}{\partial t} = \alpha_1 \int_a^b \delta(y_1 - x) \frac{\partial^2}{\partial x^2} \left[ \frac{\delta \Phi(\theta, t)}{\delta \theta(x)} \right]_{\theta = \alpha_1 \delta(y_1 - x)} \, dx,
\]

\[
= i \alpha_1 \left\langle \frac{\partial^2 u(y_1, t; \omega)}{\partial y_1^2} e^{i \alpha u(y_1, t; \omega)} \right\rangle.
\] (671)

This equation can also be obtained directly by differentiating the one-point characteristic function \( \phi(\alpha, y, t) \) with respect to time and then using the diffusion equation. Note that the term at the right-hand side of (671) cannot be easily expressed in terms of the one-point characteristic function. However, if we use the two-point function we have

\[
\left[ \frac{\partial^3 \phi(\alpha_1, \alpha_2, y_1, y_2, t)}{\partial \alpha_2 \partial y_2^2} \right]_{\alpha_2 = 0, y_2 = y_1} = \left\langle \frac{\partial^2 u(y, t; \omega)}{\partial y^2} e^{i \alpha u(y, t; \omega)} \right\rangle.
\] (672)

Therefore, we obtain

\[
\frac{\partial \phi(\alpha_1, y_1, t)}{\partial t} = i \alpha_1 \left[ \frac{\partial^3 \phi(\alpha_1, \alpha_2, y_1, y_2, t)}{\partial \alpha_2 \partial y_2^2} \right]_{\alpha_2 = 0, y_2 = y_1}
\] (673)

At this point we need an evolution equation for the two-point characteristic function \( \phi(\alpha_1, \alpha_2, y_1, y_2, t) \). Such equation may be obtained from (664) by using the test function \( \theta(x) = \alpha_1 \delta(y_1 - x) + \alpha_2 \delta(y_2 - x) \). If we perform the calculation we find that such evolution depends on the three-point characteristic function. If we keep going on we obtain an infinite hierarchy of finite-dimensional characteristic function PDEs. Taking the Fourier transform of such hierarchy yields the well-known Bogoliubov-Born-Green-Kirkwood-Yvon (BBGKY) hierarchy of probability density function equations [146, 153]. Other functional equations yield other types of hierarchies of PDEs which should be studied on a case-by-case basis. Another approach to convert an FDE to an infinite hierarchy of PDEs was developed by Thomas Lundgren in [130]. The hierarchy of PDEs is known as Lundgren-Monin-Novikov hierarchy, and it can be constructed by a direct approach [233, 90, 66] or by functional integration of the probability density functional equation (see Section 4.4).

References

[1] E. Acar, D. M. Dunlavy, and T. G. Kolda. A scalable optimization approach for fitting canonical tensor decompositions. *J. Chemometrics*, 25(2):67–86, 2011.

[2] G. Ahmadi. An approximate method for solving Hopf’s equation of Burgers’s model of turbulence. *Appl. Sci. Res.*, 32:207–215, 1976.
[3] T. Alankus. The generating functional for the probability density functions of Navier-Stokes turbulence. *J. Stat. Phys.*, 53(5-6):1261–1271, 1988.

[4] G. Allasia and C. Bracco. Lagrange interpolation on arbitrarily distributed data in Banach spaces. *Numerical Functional Analysis and Optimization*, 32(2):111–125, 2011.

[5] D. J. Amit and V. Martín-Mayor. *Field theory, the renormalization group and critical phenomena*. World Scientific Publishing, 2005.

[6] H. Andersson and T. Britton. *Stochastic epidemic models and their statistical analysis*, volume 151. 2012.

[7] N. V. Azbelev, V. P. Makasimov, and L. F. Rakhmatullina. *Introduction to the theory of functional differential equations: methods and applications*. Hindawi Publishing Corporation, first edition, 2007.

[8] M. Bachmayr, R. Schneider, and A. Uschmajew. Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations. *Foundations of Computational Mathematics*, 16(6), 2016.

[9] J. Baldeaux and M. Gnewuch. Optimal randomized multilevel algorithms for infinite-dimensional integration on function spaces with ANOVA-type decomposition. *SIAM J. Numer. Anal.*, 52(3):1128–1155, 2014.

[10] M. Van Barel, M. Humet, and L. Sorber. Approximating optimal point configurations for multivariate polynomial interpolation. *Electronic Transactions on Numerical Analysis*, 42:41–63, 2014.

[11] V. Barthelmann, E. Novak, and K. Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Computational Mechanics*, 12:273–288, 2000.

[12] C. Battaglino, G. Ballard, and T. G. Kolda. A practical randomized CP tensor decomposition. *ArXiv*, 1701.06600:1–26, 2017.

[13] C. M. Bender, K. A. Milton, and V. M. Savage. Solution of Schwinger-Dyson equation for $pt$-symmetric quantum field theory. *Phys. Rev. D*, 62:085001(1–13), 2000.

[14] A. Bensoussan, G. Da Prato, M. C. Delfour, and S. K. Mitter. *Representation and control of infinite dimensional systems*. Birkhäuser, second edition, 2006.

[15] M. J. Beran. *Statistical continuum theories*. New York: Interscience Publishers, 1968.

[16] A. Bertuzzi, A. Gandolfi, and A. Germani. Causal polynomial approximation for input-output maps on Hilbert spaces. *Math. Systems Theory*, 14:339–392, 1981.

[17] A. Bertuzzi, A. Gandolfi, and A. Germani. A Weierstrass-like theorem for real, separable Hilbert spaces. *Journal of Approximation Theory*, 32:76–81, 1981.

[18] G. Beylkin, J. Garcke, and M. J. Mohlenkamp. Multivariate regression and machine learning with sums of separable functions. *SIAM J. Sci. Comput.*, 31:1840–1857, 2009.

[19] J. C. Bezdek and R. J. Hathaway. Convergence of alternating optimization. *Neural Parallel Sci. Comput.*, 11:351–368, 2003.
[20] H. Bhatia, G. Norgard, V. Pascucci, and P.-T. Bremer. The Helmholtz-Hodge decomposition - a survey. IEEE Transactions on Visualization and Computer Graphics, 18(8):1386–1404, 2013.

[21] G. N. Bochkov, A. A. Dubkov, and A. N. Malakhov. Structure of the correlation dependence of nonlinear stochastic functionals. Radiophysics and Quantum Electronics, 20(3):276–280, 1977.

[22] S. E. Bodner. Turbulence theory with a time-varying Wiener-Hermite basis. Phys. Fluids, 12(1):33–38, 1969.

[23] L. Bos, S. De Marchi, A. Sommariva, and M. Vianello. Computing multivariate Fekete and Leja points by numerical linear algebra. SIAM J. Numer. Anal., 48(5):1984–1999, 2010.

[24] H. J. Bungartz and M. Griebel. Sparse grids. Acta Numerica, 13:147–269, 2004.

[25] R. H. Cameron and W. T. Martin. The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals. Annals of Mathematics, 48(2):385–392, 1947.

[26] Y. Cao, Z. Chen, and M. Gunzburger. ANOVA expansions and efficient sampling methods for parameter dependent nonlinear PDEs. Int. J. Numer. Anal. Model., 6:256–273, 2009.

[27] H. A. Carteret, A. Higuchi, and A. Sudbery. Multipartite generalization of the Schmidt decomposition. J. Math. Phys., 41(12):7932–7939, 2000.

[28] M. Chaika and S. I. Perlman. A Weierstrass theorem for a complex separable Hilbert space. Journal of Approximation Theory, 32:76–81, 1981.

[29] H. Chen, S. Chen, and R. H. Kraichnan. Probability distribution of a stochastically advected scalar field. Phys. Rev. Lett., 63(24):2657–2660, 1989.

[30] W. Cheney and W. Light. A course in approximation theory. Brooks/Cole publishing company, 2000.

[31] F. Chinesta, R. Keunings, and A. Leygue. The Proper Generalized Decomposition for Advanced Numerical Simulations. Springer, 2014.

[32] A. Chkifa, A. Cohen, and C. Schwab. High-dimensional adaptive sparse polynomial interpolation and applications to parametric PDEs. Found. Comput. Math., 14:601–633, 2014.

[33] H. Cho, D. Venturi, and G. E. Karniadakis. Numerical methods for high-dimensional probability density function equation. J. Comput. Phys, 315:817–837, 2016.

[34] A. Chorin and J. E. Marsden. A mathematical introduction to fluid mechanics. Springer-Verlag, 1992.

[35] J. Gani D. J. Daley. Epidemic modelling: an introduction, volume 151. 2001.

[36] V. de Silva and L.-H. Lim. Tensor rank and ill-posedness of the best low-rank approximation problem. SIAM J. Matrix Anal. Appl., 30:1084–1127, 2008.

[37] E. Deriaz and V. Perrier. Divergence-free and curl-free wavelets in two dimensions and three dimensions: application to turbulent flows. Journal of Turbulence, 7(3):1–37, 2006.

[38] E. Deriaz and V. Perrier. Direct numerical simulation of turbulence using divergence-free wavelets. Multiscale Model. Simul., 7(3):1101–1129, 2008.
[39] J. Dick and M. Gnewuch. Infinite-dimensional integration in weighted Hilbert spaces: anchored decompositions, optimal deterministic algorithms, and higher-order convergence. *Foundations Comput. Math.*, 14(5):1027–1077, 2014.

[40] J. Dick, F. Y Kuo, and I. H. Sloan. High-dimensional integration: The quasi-Monte Carlo way. *Acta Numerica*, 22:133–288, 2013.

[41] G. Dimarco and L. Pareschi. Numerical methods for kinetic equations. *Acta Numerica*, 23(4):369–520, 2014.

[42] J. Dominy and D. Venturi. Duality and conditional expectation in the Nakajima-Mori-Zwanzig formulation. *J. Math. Phys.*, 58(082701):1–26, 2017.

[43] M. D. Donsker. On function space integrals. In W. T. Martin an I. Segal, editor, *Proceedings of a Conference on the Theory and Applications of Analysis in Function Space, Dedham (MA), June 913, 1963*, pages 17–30. MIT Press, 1963.

[44] A. Doostan and H. Owhadi. A non-adapted sparse approximation of PDEs with stochastic inputs. *J. Comput. Phys.*, 230(8):3015–3034, 2011.

[45] A. Doostan, A. Validib, and G. Iaccarino. Non-intrusive low-rank separated approximation of high-dimensional stochastic models. *Comput. Methods Appl. Mech. Engrg.*, 263:42–55, 2013.

[46] C. Dopazo and E. E. O’Brien. Functional formulation of nonisothermal turbulent reactive flow. *Physics of Fluids*, 17(11):1968–1975, 1998.

[47] T. A. Driscoll and B. Fornberg. Interpolation in the limit of increasingly flat radial basis functions. *Comput. Math. Appl.*, 43:413–422, 2002.

[48] W. E, E. Ren, and E. Vanden-Eijden. Minimum action method for the study of rare events. *Comm. Pure Appl. Math.*, (57):637–656, 2004.

[49] R. Easther, D. D. Ferrante, G. S. Guralnik, and D. Petrov. A review of two novel numerical methods in QFT. *ArXiv*, hep-lat/0306038:1–16, 2003.

[50] S. F. Edwards. The statistical dynamics of homogeneous turbulence. *J. Fluid Mech.*, 18:239–273, 1964.

[51] A. D. Egorov, P. I. Sobolevsky, and L. A. Yanovich. *Functional integrals: approximate evaluation and applications*. Springer, 1993.

[52] K.-J. Engel and R. Nagel. *One-parameter semigroups for linear evolution equations*. Springer, 2000.

[53] O. G. Ernst, A. Mugler, H.-J. Starkloff, and E. Ullmann. On the convergence of generalized polynomial chaos expansions. *ESAIM: Math. Model. Numer. Anal.*, 46(2):317–339, 2012.

[54] M. Espig and W. Hackbusch. A regularized Newton method for the efficient approximation of tensors represented in the canonical tensor format. *Numerische Mathematik*, 122(3):489–525, 2012.

[55] M. Espig, W. Hackbusch, and A. Khachatryan. On the convergence of alternating least squares optimisation in tensor format representations. *ArXiv*, page 1506.00062, 2015.

[56] A. Etter. Parallel ALS algorithm for solving linear systems in the hierarchical Tucker representation. *SIAM J. Sci. Comput.*, 38(4):A2585A2609, 2016.
[57] A. Eydeland. A fast algorithm for computing integrals in function spaces: financial applications. *Computational Economics*, 7:277–285, 1994.

[58] G. L. Eyink. Action principle in nonequilibrium statistical mechanics. *Phys. Rev. E*, 54(4):955–1019, 1996.

[59] G. L. Eyink. Turbulence noise. *J. Stat. Phys.*, 83(5/6):3419–3435, 1996.

[60] R. Farwig. Rate of convergence of Shepard’s global interpolation formula. *Math. Comp.*, 46:577–590, 1986.

[61] A. Fiasconaro, B. Spagnolo, A. Ochab-Marcinek, and E. Gudowska-Nowak. 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):041904 (10pp), 2006.

[62] B. A. Finlayson. *The method of weighted residuals and variational principles*. Academic Press, 1972.

[63] C. Foias, O. Manley, R. Rosa, and R. Temam. *Navier-Stokes equations and turbulence*. Cambridge Univ. Press, first edition, 2008.

[64] R. F. Fox. Functional-calculus approach to stochastic differential equations. *Phys. Rev. A*, 33(1):467–476, 1986.

[65] M. Fréchet. Sur les fonctionelles continues. *Annales Scientifiques de l’Ecole Normale Superieure*, 27:193–216, 1910.

[66] R. Friedrich, A. Daitche, O. Kamps, J. Lülf, M. Voβkühle, and M. Wilczek. The Lundgren-Monin-Novikov hierarchy: kinetic equations for turbulence. *Comptes Rendus Physique*, 13(9-10):929–953, 2012.

[67] U. Frisch. *Turbulence: the legacy of A. N. Kolmogorov*. Cambridge University Press, 1995.

[68] T. Funaki and A. Inoue. On a new derivation of the Navier-Stokes equation. *Comm. Math. Phys.*, 65:83–90, 1979.

[69] K. Furutsu. On the statistical theory of electromagnetic waves in fluctuating medium (i). *J. Res. Natl. Bur. Stand. (Sect. D)*, 67(3):303–323, 1963.

[70] P. G. Galman and K. S. Narendra. Representations of nonlinear systems via the Stone-Weierstrass theorem. *Automatica*, 12(2):619–622, 1976.

[71] W. Gautschi. On generating orthogonal polynomials. *SIAM J. Sci. and Stat. Comput.*, 3(3):289–317, 1982.

[72] W. Gautschi. *Orthogonal polynomials: computation and approximation*. Oxford University Press, 2004.

[73] M. J. Giles. Probability distribution functions for Navier-Stokes turbulence. *Phys. Fluids*, 7(11):2785–2795, 1995.

[74] D. A. Gomes. A variational formulation for the Navier-Stokes equation. *Comm. Math. Phys.*, 257:227–234, 2005.

[75] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029–2054, 2010.
[76] L. Grasedyck, R. Kriemann, C. Löbbert, A. Nägel, G. Wittum, and K. Xylouris. Parallel tensor sampling in the hierarchical Tucker format. *Comput. Vis. Sci.*, 17(2):67–78, 2015.

[77] L. Grasedyck and C. Löbbert. Distributed hierarchical SVD in the hierarchical Tucker format. *ArXiv*, 1708.03340:1–25, 2017.

[78] W. Hackbusch. *Tensor spaces and numerical tensor calculus*. Springer, 2012.

[79] W. Hackbusch and S. Kühn. A new scheme for the tensor representation. *J. Fourier Anal. Appl.*, 15(5):706–722, 2009.

[80] J. K. Hale and S. M. Verduyn Lunel. *Introduction to functional differential equations*. Springer-Verlag, first edition, 1993.

[81] P. Hänggi. Path integral solutions for non-Markovian processes. *Z. Phys. B*, 31:407–416, 1978.

[82] P. Hänggi. The functional derivative and its use in the description of noisy dynamical systems. In L. Pesquera and M. Rodriguez, editors, *Stochastic processes applied to physics*, pages 69–95. World Scientific, 1985.

[83] J. R. Herring. Self-consistent-field approach to nonstationary turbulence. *Phys. Fluids*, 9(11):2106–2110, 1966.

[84] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral methods for time-dependent problems*. Cambridge Univ. Press, 2007.

[85] C. Hillar and L.-H. Lim. Most tensor problems are NP-hard. *ACM (JACM)*, 60(6):1–35, 2013.

[86] I. I. Hirschman and D. V. Widder. *The convolution transform*. Princeton University Press, 1955.

[87] D. Hochberg, C. Molina-París, J. Pérez-Mrcander, and M. Visser. Effective action for stochastic partial differential equations. *Phys. Rev. E*, 60(6):6343–6360, 1999.

[88] R. Hohlfeld, J. I. F. King, T. W. Drueding, and G. V. H. Sandri. Solution of convolution integral equations by the method of differential inversion. *SIAM J. Appl. Math.*, 53(1):154–167, 1993.

[89] E. Hopf. Statistical hydromechanics and functional calculus. *J. Rat. Mech. Anal.*, 1(1):87–123, 1952.

[90] I. Hosokawa. Monin-Lundgren hierarchy versus the Hopf equation in the statistical theory of turbulence. *Phys. Rev. E*, 73:067301(1–4), 2006.

[91] I. Hosokawa and K. Yamamoto. A Monte Carlo approach to the Hopf characteristic functional for 3d homogeneous turbulence. *J. Phys. Soc. Jpn.*, 56:521–534, 1987.

[92] P. Howlett, C. Pierce, and A. Torokhti. On nonlinear operator approximation with preassigned accuracy. *Journal of Computational Analysis and Applications*, 5(3):273–297, 2003.

[93] K. Ishizaka. Weak*-convergence to minimum energy measure and dispersed-dot halftoning. *SIAM J. Imaging Sciences*, 7(2):1035–1079, 2014.

[94] V. I. Istrătescu. A Weierstrass theorem for real Banach spaces. *Journal of Approximation Theory*, 19:118–122, 1977.

[95] C. Itzykson and J. B. Zuber. *Quantum field theory*. Dover, 2005. Republication of the work originally published by McGraw-Hill, Inc., NY, 1980.
[96] R. V. Jensen. Functional integral approach to classical statistical dynamics. *J. Stat. Phys.*, 25(2):183–210, 1981.

[97] B. Jouvet and R. Phythian. Quantum aspects of classical and statistical fields. *Phys. Rev. A*, 19:1350–1355, 1979.

[98] R. P. Kanwal. *Generalized functions: theory and technique.* Birkhäuser Boston, second edition, 1998.

[99] V. M. Kaplitskiï. Interpolation of nonlinear operators in weighted $l_p$-spaces. *Siberian Mathematical Journal*, 51(2):255–266, 2010.

[100] L. Karlsson, D. Kressner, and A. Uschmajew. Parallel algorithms for tensor completion in the CP format. *Parallel computing*, 57:222–234, 2016.

[101] G. E. Karniadakis and S. Sherwin. *Spectral/hp element methods for computational fluid dynamics.* Oxford University Press, second edition, 2005.

[102] E. F. Kashpur and V. V. Khlobystov. Interpolation of polynomial operators in a Hilbert space. *Journal of Mathematical Sciences*, 86(1):2455–2458, 1997.

[103] V. V. Khlobystov. On convergence of interpolation processes in a Hilbert space. *Cybernetics and Systems Analysis*, 36(6):936–941, 2000.

[104] V. V. Khlobystov. On the convergence of interpolation process in a Hilbert space. *Cybernetics and Systems Analysis*, 36(6):936–941, 2000.

[105] V. V. Khlobystov. On the convergence of an interpolation processes to an entire operator in a Hilbert space. *Journal of Mathematical Sciences*, 104(6):1682–1684, 2001.

[106] V. V. Khlobystov and E. F. Kashpur. On the accuracy of polynomial interpolation in Hilbert space with disturbed nodal values of the operator. *Cybernetics and Systems Analysis*, 38(1):143–147, 2002.

[107] A. I. Khuri. Applications of Dirac’s delta function in statistics. *Int. J. Math. Educ. Sci. Technol.*, 35(2):185–195, 2004.

[108] H. Kleinert. *Critical properties of $\phi^4$ theories.* World Scientific Publishing Company, 2001.

[109] H. Kleinert. *Path integrals in quantum mechanics, statistics, polymer physics, and financial markets.* World Scientific Publishing Company, 2009.

[110] H. Kleinert and V. Schulte-Frohlinde. *Critical properties of $\phi^4$-theories.* World Scientific, 2001.

[111] V. I. Klyatskin. Statistical theory of light reflection in randomly inhomogeneous medium. *Sov. Phys. JETP*, 38:27–34, 1974.

[112] V. I. Klyatskin. *Dynamics of stochastic systems.* Elsevier Publishing Company, 2005.

[113] T. Kolda and B. W. Bader. Tensor decompositions and applications. *SIREV*, 51:455–500, 2009.

[114] D. Kressner and C. Tobler. Algorithm 941: htucker – a Matlab toolbox for tensors in hierarchical Tucker format. *ACM Transactions on Mathematical Software*, 40(3):1–22, 2014.

[115] P. M. Kroonenberg and J. de Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45:69–97, 1980.
[116] R. Kubo. Generalized cumulant expansion method. *Journal of the Physical Society of Japan*, 17(7):1100–1120, 1962.

[117] F. Langouche, D. Roekaerts, and E. Tirapegui. Functional integral methods for stochastic fields. *Physica A: Statistical and Theoretical Physics*, 95:252–274, 1979.

[118] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

[119] J. W. Lawson and G. S. Guralnik. The source Galerkin method: fermionic formulation. *Nuclear Physics B*, 459(3):612–627, 1996.

[120] J. W. Lawson and G. S. Guralnik. The source Galerkin method for scalar field theory. *Nuclear Physics B*, 459(3):589–611, 1996.

[121] D. Lazzaro and L. B. Montefusco. Radial basis functions for the multivariate interpolation. *Journal of Computational and Applied Mathematics*, 140:521–536, 2002.

[122] C. P. Lee, W. C. Meecham, and H. D. Hogge. Application of the Wiener-Hermite expansion to turbulence of moderate Reynolds number. *Phys. Fluids*, 25(8):1322–1327, 1982.

[123] S. E. Leugarans, R. A. Moyeed, and B. W. Silverman. Canonical correlation analysis when the data are curves. *J. Roy. Soc. Ser. B*, 55:725–740, 1993.

[124] R. M. Lewis and R. H. Kraichnan. A space-time functional formalism for turbulence. *Communications on Pure and Applied Mathematics*, 15:397–411, 1962.

[125] G. Li and H. Rabitz. Regularized random-sampling high dimensional model representation (RS-HDMR). *Journal of Mathematical Chemistry*, 43(3):1207–1232, 2008.

[126] G. Li, S.-W. Wang, H. Rabitz, S. Wang, and P. Jaffé. Global uncertainty assessments by high dimensional model representations (HDMR). *Chemical Engineering Science*, 57(21):4445–4460, 2002.

[127] Yu. Yu. Lobanov. Deterministic computation of functional integrals. *Computer Physics Communications*, 99:59–72, 1996.

[128] A. Lópezc-García and E. B. Saff. Asymptotics of greedy energy points. *Math. Comp.*, 79:2287–2316, 2010.

[129] D. Lucas and R. Kerswell. Spatiotemporal dynamics in two-dimensional kolmogorov flow over large domains. *J. Fluid Mech.*, 750:518–554, 2014.

[130] T. S. Lundgren. Distribution functions in the statistical theory of turbulence. *Phys. Fluids*, 10(5):969–975, 1967.

[131] Y. Maday, O. Mula, A. T. Patera, and M. Yano. The generalized empirical interpolation method: stability theory on Hilbert spaces with an application to the Stokes equation. *Comput. Meth. Appl. Engng.*, (In Review):1–25, 2014.

[132] Y. Maday, O. Mula, and G. Turinici. Convergence analysis of the generalized empirical interpolation method. *SIAM J. Numer. Anal.*, (In Review):1–24, 2014.

[133] F. Magri. An operator approach to Poisson brackets. *Annals of Physics*, 99:196–228, 1976.
[134] V. L. Makarov and V. V. Khlobystov. Polynomial interpolation of operators. *Journal of Mathematical Sciences*, 84(4):1244–1289, 1997.

[135] S. De Marchi. On Leja sequences: some results and applications. *Applied Mathematics and Computation*, 58(198):785–727, 2004.

[136] P. C. Martin, E. D. Siggia, and H. A. Rose. Statistical dynamics of classical systems. *Phys. Rev. A*, 8:423–437, 1973.

[137] S. Marzucchi. *Mathematical Feynman path integrals and their applications*. World Scientific, 2009.

[138] A. J. McCane, H. C. Luckock, and A. J. Bray. Path integrals and non-Markov processes. 1. general formalism. *Phys. Rev. A*, 41(2):644–656, 1990.

[139] W. D. McComb. *The Physics of Fluid Turbulence*. Oxford University Press, 1990.

[140] W. C. Meecham and D. T. Jeng. Use of the Wiener-Hermite expansion for nearly normal turbulence. *J. Fluid Mech.*, 32(2):225–249, 1968.

[141] W. C. Meecham and A. Siegel. Wiener-Hermite expansion in model turbulence at large Reynolds numbers. *Phys. Fluids*, 7(8):1178–1190, 1964.

[142] J.-M. Miao. General expressions for the Moore-Penrose inverse of a $2 \times 2$ block matrix. *Linear Algebra and its Applications*, 151(1):1–15, 1991.

[143] M. J. Mohlencamp and L. Monzón. Trigonometric identities and sums of separable functions. *Math. Intelligencer*, 27:65–69, 2005.

[144] A. S. Monin and A. M. Yaglom. *Statistical Fluid Mechanics, Volume I: Mechanics of Turbulence*. Dover, 2007.

[145] A. S. Monin and A. M. Yaglom. *Statistical Fluid Mechanics, Volume II: Mechanics of Turbulence*. Dover, 2007.

[146] D. Montgomery. A BBGKY framework for fluid turbulence. *Phys. Fluids*, 19(6):802–810, 1976.

[147] C. D. Moravitz-Martin and C. F. Van Loan. A Jacobi-type method for computing orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.*, 30(3):1219–1232, 2007.

[148] C. S. Morawetz, J. B. Serrin, and Y. G. Sinai, editors. *Selected works of Eberhard Hopf with commentaries*. American Mathematical Society, 2002.

[149] F. Moss and P. V. E. McClintock, editors. *Noise in nonlinear dynamical systems. Volume I: theory of continuous Fokker-Planck systems*. Cambridge Univ. Press, 1995.

[150] A. S. Vasudeva Murthy. a note on the differential inversion method of Hohlfeld et al. *SIAM J. Appl. Math.*, 55(3):719–722, 1995.

[151] A. Narayan and J. Jakeman. Adaptive Leja sparse grid constructions for stochastic collocation and high-dimensional approximation. *SIAM J. Sci. Comput.*, 36(6):A2952–A2983, 2014.

[152] A. Narayan and D. Xiu. Stochastic collocation methods on unstructured grids in high-dimensions via interpolation. *SIAM J. Sci. Comput.*, 34(3):A1729–A1752, 2012.
[153] A. Narayan and D. Xiu. Constructing nested nodal sets for multivariate polynomial interpolation. *SIAM J. Sci. Comput.*, 35(5):A2293–A2315, 2013.

[154] M. Z. Nashed. Differentiability and related properties of non-linear operators: some aspects of the role of differentials in non-linear functional analysis. In L. B. Rall, editor, *Nonlinear Functional Analysis and Applications*. Academic Press, 1971.

[155] O. Nelles. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer, 2001.

[156] A. Nouy. Generalized spectral decomposition method for solving stochastic finite element equations. *Comput. Methods Appl. Mech. Engrg.*, 197:4718–4736, 2008.

[157] A. Nouy. A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 199:1603–1626, 2010.

[158] A. Nouy. Low-rank tensor methods for model order reduction. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of uncertainty quantification*, pages 1–26. Springer, 2016.

[159] A. Nouy. Higher-order principal component analysis for the approximation of tensors in tree-based low-rank formats. 2017.

[160] E. Novak and K. Ritter. High dimensional integration of smooth functions over cubes. *Numer. Math.*, 75:79–97, 1996.

[161] E. A. Novikov. Functionals and the random-force method in turbulence. *Sov. Phys. JETP*, 20:1290–1294, 1965.

[162] M. Oberlack and M. Waclawczyk. On the extension of Lie group analysis to functional differential equations. *Arch. Mech.*, 58:597–618, 2006.

[163] E. E. O’Brien. The probability density function (pdf) approach to reacting turbulent flows. In *Topics in Applied Physics (Turbulent Reacting Flows)*, volume 44, pages 185–218. Springer, 1980.

[164] H. Ogura. Orthogonal functionals of the Poisson process. *IEEE Trans. Inf. Theory*, 4:473–481, 1972.

[165] A. Okopińska. Solving Schwinger-Dyson equations by truncation in zero-dimensional scalar quantum field theory. *Phys. Rev. D*, 43(10):3561–3564, 1991.

[166] D. S. Oliver. Calculation of the inverse of the covariance. *Mathematical Geology*, 30(7):911–933, 1975.

[167] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. Academic Press (New York), 1970.

[168] I. V. Oseledets. Tensor-train decomposition. *SIAM J. Sci. Comput.*, 33(5):2295–2317, 2011.

[169] M. N. Özişik. *Heat conduction*. John Wiley & Sons, second edition, 1993.

[170] G. Palm and T. Poggio. The Volterra representation and the Wiener expansion: validity and pitfalls. *SIAM J. Appl. Math.*, 33(2):195–216, 1977.

[171] A. Peres. Higher-order Schmidt decompositions. *Phys. Lett. A*, 202(1):16–17, 1995.
[172] L. Pesquera, M. A. Rodriguez, and E. Santos. Path integrals for non-Markovian processes. *Physics Letters*, 94(6-7):287–289, 1983.

[173] R. Phythian. The operator formalism of classical statistical dynamics. *J. Phys A: Math. Gen.*, 8(9):1423–1432, 1975.

[174] R. Phythian. The functional formalism of classical statistical dynamics. *J. Phys A: Math. Gen.*, 10(5):777–788, 1977.

[175] A. Pinkus. *n-widths in approximation theory*. Springer-Verlag, 1985.

[176] L. Plaskota and G. W. Wasilkowski. Tractability of infinite-dimensional integration in the worst case and randomized settings. *Journal of Complexity*, 27(6):505–518, 2011.

[177] V. N. Popov. *Functional integrals in quantum field theory and statistical physics*. D. Reider publishing company, 1983.

[178] W. A. Porter. Nonlinear systems in Hilbert space. *Int. J. Contr.*, 13:593–602, 1971.

[179] W. A. Porter. Synthesis of polynomic systems. *SIAM J. Math. Anal.*, 11(2):308–315, 1980.

[180] W. A. Porter and T. M. Clark. Causality structure and the Weierstrass theorem. *J. Math. Anal. Appl.*, 52:351–363, 1975.

[181] P. M. Prenter. A Weierstrass theorem for real, separable Hilbert spaces. *Journal of Approximation Theory*, 3:341–351, 1970.

[182] P. M. Prenter. A Weierstrass theorem for real, separable Hilbert spaces. *Journal of Approximation Theory*, 3:341–351, 1970.

[183] P. M. Prenter. Lagrange and Hermite interpolation in Banach spaces. *Journal of Approximation Theory*, 4:419–432, 1971.

[184] A. Quarteroni. *Numerical mathematics*. Springer, second edition, 2007.

[185] H. Rabitz, Ö. F. Aliş, J. Shorter, and K. Shim. Efficient input–output model representations. *Computer Physics Communications*, 117(1-2):11–20, 1999.

[186] T. M. Rassias. *Functional equations in mathematical analysis*. Springer, 2011.

[187] T. M. Rassias. *Handbook of functional equations: stability theory*. Springer, 2014.

[188] M. J. Reynolds, A. Doostan, and G. Beylkin. Randomized alternating least squares for canonical tensor decompositions: application to a PDE with random data. *SIAM J. Sci. Comput.*, 38:A2634–A2664, 2016.

[189] H.-K. Rhee, R. Aris, and N. R. Amundson. *First-order partial differential equations, volume 1: theory and applications of single equations*. Dover, 2001.

[190] H. Risken. *The Fokker-Planck equation: methods of solution and applications*. Springer-Verlag, second edition, 1989. Mathematics in science and engineering, vol. 60.

[191] T. Rohwedder and A. Uschmajew. On local convergence of alternating schemes for optimization of convex problems in the tensor train format. *SIAM J. Numer. Anal.*, 51(2):1134–1162, 2013.
[192] M. Rosa-Clot and S. Taddei. Deterministic technique of path summation. *Phys. Rev. C*, 50:627–629, 1994.

[193] G. Rosen. Turbulence theory and functional integration I. *Phys. Fluids*, 3(4):519–524, 1960.

[194] G. Rosen. Functional integration theory for incompressible fluid turbulence. *Phys. Fluids*, 10(12):2614–2619, 1967.

[195] G. Rosen. Dynamics of probability distributions over classical fields. *International Journal of Theoretical Physics*, 4(3):189–195, 1971.

[196] G. Rosen. Functional calculus theory for incompressible fluid turbulence. *J. Math. Phys.*, 12(5):812–820, 1971.

[197] G. Rosen, J. A. Okolowski, and G. Eckstut. Functional integration theory for incompressible fluid turbulence II. *J. Math. Phys.*, 10(3):415–421, 1967.

[198] A. Rosteck and M. Oberlack. Lie algebra of the symmetries of the multi-point equations in statistical turbulence theory. *J. Nonlin. Math. Phys.*, 18:251–264, 2011.

[199] W. J. Rugh. *Nonlinear system theory: the Volterra/Wiener approach*. Johns Hopkins University Press, 1981.

[200] G. Sacchi-Landi and H. Vandeven. Polynomial approximation of divergence-free functions. *Mathematics of Computation*, 185:103–130, 1989.

[201] P. K. Sahoo and P. Kannappan. *Introduction to functional equations*. CRC Press, 2011.

[202] R. Schaback. Multivariate interpolation by polynomials and radial basis functions. *Constructive Approximation*, 21:293–317, 2005.

[203] J. T. Schwartz. *Nonlinear functional analysis*. Gordon and Breach Science Publishers, 1969.

[204] A. Segall and T. Kailath. Orthogonal functionals of independent-increment processes. *IEEE Trans. Inf. Theory*, 22(3):287–298, 1976.

[205] M. Shetzen. *The Volterra and Wiener theories of nonlinear systems*. Wiley, New York, 1980.

[206] C. Da Silva and F. J. Herrmann. Optimization on the Hierarchical Tucker manifold – applications to tensor completion. *Linear Algebra and its Applications*, 481:131–173, 2015.

[207] I. I. Gikhman A. V. Skorokhod. *The Theory of Stochastic Processes I*. Springer, 2004.

[208] G. Song, J. Riddle, G. E. Fasshauer, and F. J. Hickernell. Multivariate interpolation with increasingly flat radial basis functions of finite smoothness. *Adv. Comput. Math.*, 36:485–501, 2012.

[209] M. M. Stanišić. *The Mathematical Theory of Turbulence*. Springer Verlag, 1985.

[210] M. H. Stone. The generalized Stone-Weierstrass approximation theorem. *Mathematics Magazine*, 21(4):167–184, 1948.

[211] R. L. Stratonovich. Some Markov methods in the theory of stochastic processes in nonlinear dynamical systems. In F. Moss and P. V. E. McClintock, editors, *Noise in nonlinear dynamical systems (Vol. 1)*, pages 16–68. Cambridge Univ. Press, 1989.
[212] E. Swanson. A primer on functional methods and the Schwinger-Dyson equations. *ArXiv*, 1008.4337:1–38, 2010.

[213] E. Tonti. Variational formulation for nonlinear differential equations (i). *Académie royale de Belgique, bulletin de la classe des sciences, 5 Série -Tome LV*, 3:137–165, 1969.

[214] E. Tonti. Variational formulation for nonlinear differential equations (ii). *Académie royale de Belgique, bulletin de la classe des sciences, 5 Série -Tome LV*, 4:262–278, 1969.

[215] E. Tonti. Variational formulation for every nonlinear problem. *Int. J. Engng. Sci.*, 2(11/12):1343–1371, 1984.

[216] A. Torokhti and P. Howlett. *Computational methods for modelling of nonlinear systems*. Elsevier, first edition, 2007.

[217] M. Ueda. Probability-density-functional description of photoelectron statistics. *Phys. Rev. A*, 40(2):1096–1108, 1989.

[218] M. Ueda. Probability-density-functional description of quantum photodetection process. *Quantum Opt.*, 1:131–151, 1989.

[219] A. Uschmajew. Local convergence of the alternating least squares algorithm for canonical tensor approximation. *SIAM J. Matrix Anal. Appl.*, 33(2):639–652, 2012.

[220] M. M. Vainberg. *Variational methods for the study of nonlinear operators*. Holden-Day, 1964.

[221] J. van Neerven. *Stochastic Evolution Equations*. 2008. ISEM Lecture Notes.

[222] N. Vannieuwenhoven, J. Nicaise, R. Vandebriil, and K. Meerbergen. On generic nonexistence of the Schmidt–Eckart–Young decomposition for complex tensors. *SIAM. J. Matrix Anal. & Appl.*, 35(3):886903, 2014.

[223] D. Venturi. Convective derivatives and Reynolds transport in curvilinear time-dependent coordinate systems. *J. Phys. A: Math. Gen.*, 42:125203(1–16), 2009.

[224] D. Venturi. A fully symmetric nonlinear biorthogonal decomposition theory for random fields. *Physica D*, 240(4-5):415–425, 2011.

[225] D. Venturi. Conjugate flow action functionals. *J. Math. Phys.*, 54:113502(1–19), 2013.

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

[227] D. Venturi and G. E. Karniadakis. New evolution equations for the joint response-excitation probability density function of stochastic solutions to first-order nonlinear PDEs. *J. Comput. Phys.*, 231:7450–7474, 2012.

[228] D. Venturi, T. P. Sapsis, H. Cho, and G. E. Karniadakis. A computable evolution equation for the joint response-excitation probability density function of stochastic dynamical systems. *Proc. R. Soc. A*, 468(2139):759–783, 2012.

[229] D. Venturi, D. M. Tartakovsky, A. M. Tartakovsky, and G. E. Karniadakis. Exact PDF equations and closure approximations for advective-reactive transport. *J. Comput. Phys.*, 243:323–343, 2013.
[230] D. Venturi, X. Wan, R. Mikulevicius, B. L Rozovskii, and G. E. Karniadakis. Wick-Malliavin approximation to nonlinear stochastic partial differential equations: analysis and simulations. Proc. R. Soc. A, 469(2158):1–20, 2013.

[231] V. Volterra. Theory of functionals and of integral and integro-differential equations. Dover, 1959.

[232] M. Waclawczyk and M. Oberlack. Application of the extended Lie group analysis to the hopf functional formalism of the Burgers equation. J. Math. Phys., 54:07291(119), 2013.

[233] M. Waclawczyk, N. Staffolani, M. Oberlack, A. Rosteck, M. Wilczek, and R. Friedrich. Statistical symmetries of the Lundgren-Monin-Novikov hierarchy. Phys. Rev. E, 90:013022(111), 2014.

[234] X. Wan and G. E. Karniadakis. Long-term behavior of polynomial chaos in stochastic flow simulations. Comput. Methods Appl. Mech. Engrg., 195:5582–5596, 2006.

[235] X. Wan and G. E. Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. SIAM J. Sci. Comput., 28(3):901–928, 2006.

[236] C.-J. Wang. Effects of colored noise on stochastic resonance in a tumor cell growth system. Phys. Scr., 80:065004 (5pp), 2009.

[237] G. W. Wasilkowski. Liberating the dimension for $l_2$-approximation. Journal of Complexity, 28:304–319, 2012.

[238] S. Weinberg. The quantum theory of fields: volume I. Cambridge University Press, 2002.

[239] N. Wiener. Nonlinear problems in random theory. MIT Press, 1966.

[240] E. J. Fuselier G. B. Wright. A radial basis function method for computing helmholtzhodge decompositions. IMA J. Numer. Anal., 37(2):774–797, 2017.

[241] J. Wu. Theory and applications of partial functional differential equations. Springer, first edition, 1996.

[242] D. Xiu. Numerical methods for stochastic computations: a spectral approach. Princeton University Press, 2010.

[243] D. Xiu and G. E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput., 24(2):619–644, 2002.

[244] D. Xiu and G. E. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. J. Comp. Phys., 187:137–167, 2003.

[245] Y. Xu and W. Yin. A block coordinate decent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J. of Imaging Sci., 6(3):17581789, 2013.

[246] K. Yasue. A variational principle for the Navier-Stokes equation. Journal of Functional Analysis, 51:133–141, 1983.

[247] G You and S. Leung. Eulerian based interpolation schemes for flow map construction and line integral computation with applications to Lagrangian coherent structures extraction. Journal of Scientific Computing, To appear, 2017.
[248] C. Zeng and H. Wang. Colored noise enhanced stability in a tumor cell growth system under immune response. *J. Stat. Phys.*, 141(5):889–908, 2010.

[249] Z. Zhang, M. Choi, and G. E. Karniadakis. Anchor points matter in ANOVA decomposition. *Proceedings of ICOSAHOM’09*, Springer, eds. E. Ronquist and J. Hesthaven, 2010.

[250] Z. Zhang and G. E. Karniadakis. *Numerical methods for stochastic partial differential equations with white noise*. Springer, 2017.

[251] X. Zheng and S. Dong. An eigen-based high-order expansion basis for structured spectral elements. *J. Comput. Phys. Foundations of Computational Mathematics*, 230, 2011.

[252] J. Zinn-Justin. *Quantum field theory and critical phenomena*. Oxford Univ. Press, fourth edition, 2002.