Conditional Densities and Likelihoods for Hypertoroidal Densities Based on Trigonometric Polynomials

Florian Pfaff, **Kailai Li**, and **Uwe D. Hanebeck**

Abstract— **Recently, trigonometric polynomials have been used to approximate densities or their square roots in the context of Bayesian estimation. Trigonometric polynomials were also used to interpolate function values on grids on hypertori. In this paper, we derive formulae for conditional densities and likelihoods for multivariate densities parameterized by grid values or Fourier coefficients. Efficient formulae are proposed for both representations that involve no more than $O(n \log n)$ operations. The conditional densities can be described using a single parameter vector. For the likelihoods, formulae are given that allow for a precise evaluation using two parameter vectors. Furthermore, formulae involving only a single parameter vector are provided for approximations of the likelihoods.**

I. Introduction

Statistical modeling of periodic quantities is important in many fields, such as geology [1], biology [2], and particularly protein bioinformatics [3], where it led to a major breakthrough in the protein folding problem [4]. The Cartesian product of two circles is a torus, and the Cartesian product of even more circles is a hypertorus. Several distributions exist for toroidal and hypertoroidal domains. One is the bivariate von Mises distribution [5]. Furthermore, a multivariate version of the generalized von Mises distribution has been proposed [6]. Another parametric distribution is the wrapped normal distribution for arbitrary dimensions [7]. There are also multivariate projected Gaussian distributions for hypertoroidal domains [8]. Additional distributions were specifically tailored to problems in the field of protein bioinformatics [9], [10].

Very versatile classes of distributions, which are the focus of this paper, are those used by the Fourier filters [11] and the grid filter [12]. In the Fourier identity filter (IFF), the density is represented using a trigonometric polynomial, which is a Fourier series with a finite number of nonzero coefficients. Densities in this representation may have negative function values. In the Fourier square root filter (SqFF), a trigonometric polynomial describes the square root of the density instead. By squaring the values of the trigonometric polynomial, the nonnegativity of the function values can be ensured. In the grid filter, one approximates function values of the density on a grid. In our grid filter, function values at other points are provided by interpolating the square roots of the function values using a trigonometric polynomial and squaring the result. Similar to the SqFF, the nonnegativity of the function values is ensured.

Florian Pfaff, Kailai Li, and Uwe D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany. {florian.pfaff, kailai.li, uwe.hanebeck}@kit.edu

In this paper, we explain representation-specific approaches to obtain conditional densities and likelihoods from multivariate densities in the representations based on trigonometric polynomials and grids. Before going into the technical details, we illustrate the differences between likelihoods and conditional densities and explain why likelihoods and conditional densities are of interest in filtering contexts.

We now consider the difference between likelihood functions and conditional densities for two random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ with the joint density $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$. Denoting (scalar) variables as x and y and a fixed value as $\hat{y}$, we distinguish

$$f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y}) = \frac{f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})}{f^{\boldsymbol{y}}(\hat{y})} \ , \quad f^{\boldsymbol{y}|\boldsymbol{x}}(\hat{y}|x) = \frac{f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})}{f^{\boldsymbol{x}}(x)} \ .$$

The former is a likelihood, and the latter is a conditional density. Both $f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y})$ and $f^{\boldsymbol{y}|\boldsymbol{x}}(\hat{y}|x)$ are related to a slice of the joint density $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$ that is obtained by using the fixed value $\hat{y}$ as the second input argument. For scalar x, $f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})$ is a function of a single scalar variable. To obtain $f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y})$, $f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})$ is divided by the constant $f^{\boldsymbol{y}}(\hat{y})$ that ensures the normalization of $f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y})$ (unless $f^{\boldsymbol{y}}(\hat{y})$ is 0, but this is not practically relevant since one normally only conditions on values that have nonzero density). For the likelihood, the function $f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})$ is divided by the function $f^{\boldsymbol{x}}(x)$, which is a marginal of $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$ and also a function of x. The result of the division is not necessarily a function that integrates to one.

In summary, for $f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y})$, one takes a slice of the joint density and normalizes it by dividing it by a scalar. For $f^{\boldsymbol{y}|\boldsymbol{x}}(\hat{y}|x)$, one takes the same slice of the joint density and then divides it by the function $f^{\boldsymbol{x}}(x)$ that is obtained by marginalizing y out of $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$.

There are several situations in which one needs to derive a likelihood from a joint density. For example, one may want to employ Bayes' formula for an update step of a Bayesian filter based on the joint density $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$ and the measurement $\hat{y}$. In filtering contexts, it may be necessary to derive the likelihood $f^{\boldsymbol{y}|\boldsymbol{x}}(\hat{y}|x)$ in a compatible parametric form to perform the update step. Therefore, we present formulae for directly determining the parameters of the likelihoods from the parameters of the joint density in this paper. If one is only given $f^{\boldsymbol{x},\boldsymbol{y}}(x, y)$ as a function, we can determine the required parameters once before running the filter.

Conditional densities are particularly relevant when $\boldsymbol{x}$ and $\boldsymbol{y}$ are the two components of the state of a system. When obtaining uncertainty-free information on one component of the state, one may be interested in the distribution of the other component of the state. Further, it can be of interest to describe the distribution of a specific component when considering different values for the other components of the state.

The paper is structured as follows. In Sec. II, we detail the representations used in the Fourier and grid filters. In Sec. III, we present the approaches to obtain conditional densities and likelihoods. A conclusion and an outlook are provided in the last section.

II. Density Representations

In this section, we focus on the density representations used in the Fourier and grid filters. We also include some basic information on the filters, but refer the reader to [11], [13] for details on the operations for the hypertoroidal Fourier filter and to [14] for the hypertoroidal grid filter. The formulae given in this section are suitable for arbitrary-dimensional vectors $\underline{x}$. In the next section, we focus on the two-dimensional case, in which the vector (denoted as an underlined small letter) $\underline{x}$ is assumed to be $[x, y]^\top$.

A. Representations Used in the Fourier Filters

To represent densities in the IFF, we consider a Fourier coefficient tensor $\mathbf{C}^{\text{id}}$ of size m along each dimension, which leads to a total of $n = m^d$ coefficients for a d-dimensional density. The assumption of equal sizes along all dimensions is not a requirement and is only used to simplify the explanations. The density is then given by

$$f^{\text{id}}(\underline{x}) = \sum_{\underline{k} \in \mathcal{K}} c^{\text{id}}_{\underline{k}} \mathrm{e}^{i(\underline{k} \cdot \underline{x})} \qquad (1)$$

with $\underline{k} \in \mathcal{K} = \{-\frac{m-1}{2}, \dots, \frac{m-1}{2}\}^d$. Closed-form formulae exist for calculating the coefficients for certain classes of densities, e.g., the multivariate wrapped normal distribution [11, Appendix A]. If no such formula exists, the coefficients can be approximated based on function values on a regular grid (as used in the grid filters) using a multidimensional fast Fourier transform [15] (FFT). As mentioned in the introduction, directly representing a density using a trigonometric polynomial can lead to negative function values.

In the representation used for the SqFF, a trigonometric polynomial with coefficient tensor $\mathbf{C}^{\text{sqrt}}$ is squared. This leads to the formula

$$f^{\text{sqrt}}(\underline{x}) = \left(\sum_{\underline{k} \in \mathcal{K}} c^{\text{sqrt}}_{\underline{k}} \mathrm{e}^{i(\underline{k} \cdot \underline{x})} \right)^2 .$$

In this case, the function is always guaranteed to be nonnegative.

The update step of the IFF and SqFF can both be performed in $O(n \log n)$. The prediction step for the identity system model with additive noise is in $O(n)$ for the IFF and in $O(n \log n)$ for the SqFF. A general prediction step was presented in [13]. While the computational complexity in the most general case is in $O(n^2 \log n)$, a complexity in $O(n^2)$ can be achieved under certain circumstances.

B. Representation Used in the Grid Filter

Grid filters have previously been used as filters for bounded regions of real domains [16, Sec. 4.1]. In our grid filter for the hypertorus [17], a potentially multivariate density is represented using function values on a grid.

An equidistant grid on the circle is given by $2\pi \frac{k}{m}$ with $k \in \mathbb{Z}/m\mathbb{Z} = \{0, \dots, m-1\}$. We only consider grids for the hypertorus that can be generated by taking the d-fold Cartesian product of such a grid for the circle. Such grids have an equal resolution along all dimensions. Similar as for the Fourier filters, this is not a requirement of the filter but simplifies the explanations. The set of grid points $\mathcal{B}$ for the hypertorus is thus

$$\mathcal{B} = \left\{ \left[2\pi \frac{k_1}{m}, 2\pi \frac{k_2}{m}, \cdots, 2\pi \frac{k_d}{m} \right] \middle| \underline{k} \in (\mathbb{Z}/m\mathbb{Z})^d \right\} .$$

This results in a total number of $n = m^d$ grid points.

In this paper, we say the function values on the grid are stored in a tensor $\mathbf{\Lambda}$ as in the fast prediction step in [17]. To obtain the function values of the density for inputs that are not grid points, the function values at the nearest grid point could be used. However, this does not lead to a smooth density. Furthermore, experiments in which we reconstructed a given density based on the function values on the grid showed bad reconstruction performance.

Therefore, we consider interpolations based on trigonometric polynomials. One way would be to interpolate the function values on the grid. To this end, we can perform a multidimensional FFT to obtain the Fourier coefficient tensor $\mathbf{C}^{\text{id}}$. This leads to the density function

$$f^{\text{id}}(\underline{x}) = \sum_{\underline{k} \in (\mathbb{Z}/m\mathbb{Z})^d} c^{\text{id}}_{\underline{k}} \mathrm{e}^{i(\underline{k} \cdot \underline{x})}, \qquad \mathbf{C}^{\text{id}} = \text{FFT}(\mathbf{\Lambda}) .$$

A downside to this interpolation is that it may have negative function values.

To ensure all function values are nonnegative, we can first take the square root of the function values before performing the FFT to obtain the Fourier coefficient tensor $\mathbf{C}^{\text{sqrt}}$. As in the SqFF, the trigonometric polynomial is squared to obtain the function

$$f^{\text{sqrt}}(\underline{x}) = \left(\sum_{\underline{k} \in (\mathbb{Z}/m\mathbb{Z})^d} c^{\text{sqrt}}_{\underline{k}} \mathrm{e}^{i(\underline{k} \cdot \underline{x})} \right)^2 , \quad \mathbf{C}^{\text{sqrt}} = \text{FFT}(\sqrt{\mathbf{\Lambda}}) .$$

To prevent negative function values, we use the latter as our default interpolation.

III. Determining Conditional Densities and Likelihoods

The derivations in this section are not strictly limited to the two-dimensional case, which we focus on in this paper for clarity. The approaches provided in this paper are also valid when dealing with $d_{\underline{x}}$-dimensional vectors $\underline{x}$ and $d_{\underline{y}}$-dimensional vectors $\underline{y}$. We will point out the key differences for the higher dimensional case throughout this section. We always assume $\hat{y}$ is given. Thus, the likelihood to be derived is $f^{\boldsymbol{y}|\boldsymbol{x}}(\hat{y}|x)$, and the conditional density is $f^{\boldsymbol{x}|\boldsymbol{y}}(x|\hat{y})$.

In the previous section, we used that the Fourier coefficients of a density (or its square root) can be obtained from grid values via an FFT. The FFT and IFFT allow deriving the parameters for one representation from the parameters of a different representation, as shown in Fig. 1. The grid values $\mathbf{\Lambda}$ can be converted into the Fourier coefficient matrix $\mathbf{C}^{\text{id}}$ describing a function via an FFT, while the opposite conversion is possible via an IFFT. By applying an FFT to the square roots of the grid values, which we denote by $\sqrt{\mathbf{\Lambda}}$, we obtain the Fourier coefficient matrix $\mathbf{C}^{\text{sqrt}}$ for the square root of the function. The grid values for the density and the density's square root can be converted into one another by taking the square roots of the values or squaring them.

The convolution of two Fourier coefficient matrices corresponds to the multiplication of the original functions (see [18, Ch. 4.4] for the 1-D case). Hence, by convolving $\mathbf{C}^{\text{sqrt}}$ with itself, we can obtain a Fourier coefficient matrix representing the same function as $\mathbf{C}^{\text{sqrt}}$ without the need to square the trigonometric polynomial. For true equivalence of the functions, the full convolution result has to be used, which is larger than the original matrix.

Generally, there is no simple way to derive a coefficient matrix describing the square root of the density from the coefficient matrix describing the density. However, it is possible to convert $\mathbf{C}^{\text{id}}$ to grid values $\mathbf{\Lambda}$ via an FFT, take the square roots of these grid values, and then perform an IFFT to obtain the Fourier coefficients. In general, this approach involves approximation errors. These are generally inevitable when keeping the number of coefficients constant. For example, while a sine function can be described with $m = 3$ coefficients, this does not hold for the square root of a sine function.

The 2-D FFT can be computed using a 1-D FFT over one of the dimensions followed by an FFT over the other dimension [19]. In this paper, we also consider a representation for which only a 1-D FFT is applied along the dimension of y (or the dimensions of $\underline{y}$ for higher dimensions). As illustrated in Fig. 1, we denote the result of applying this 1-D FFT to $\mathbf{\Lambda}$ by $\mathbf{\Xi}^{\text{id}}$, and we use $\mathbf{\Xi}^{\text{sqrt}}$ to refer to the result of applying this 1-D FFT to $\sqrt{\mathbf{\Lambda}}$.

For both representations, we consider shifting $f^{\boldsymbol{x},\boldsymbol{y}}(x,y)$ along the y-axis by $\hat{y}$. This leads to the function

$$\underleftarrow{f}_{\hat{y}}^{\boldsymbol{x},\boldsymbol{y}}(x,y) = f^{\boldsymbol{x},\boldsymbol{y}}(x, y - \hat{y}) \ .$$

As we will see in this section, deriving the parameters of $\underleftarrow{f}_{\hat{y}}^{x,y}(x,0)$ is easier than directly deriving the parameters of $f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})$ for the considered representations. For the conditional density, we merely need to normalize $\underleftarrow{f}_{\hat{y}}^{x,y}(x,0)$. For the likelihood, we also need to derive the marginal density $f^{\boldsymbol{x}}(x)$. Thus, we also need an approach to marginalize y out of $f^{\boldsymbol{x},\boldsymbol{y}}(x,y)$ or, equivalently, $\underleftarrow{f}_{\hat{y}}^{\boldsymbol{x},\boldsymbol{y}}(x,y)$. In the subsections of this section, we first consider the grid-based representation and afterward regard densities in the representations used by the Fourier filters.

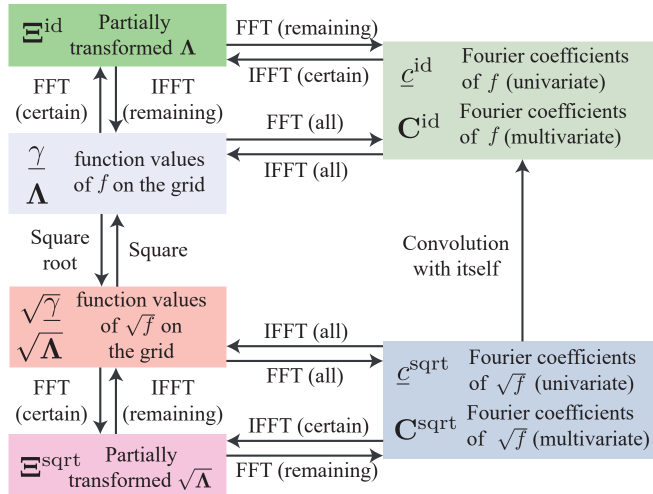


Fig. 1: Overview of the considered representations.

A. *Approach for Grid-Based Densities*

For both the conditional density and the likelihood, we will use the slice of the joint density at $\hat{y}$. For this, we regard how to get the function values of $f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})$ on the one-dimensional grid along the x-axis with $\hat{y}$ as the second argument. If $\hat{y} \in \mathcal{I} = \{2\pi\frac{k}{m} | k \in \mathbb{Z}/m\mathbb{Z}\}$, this is trivial because the grid points are already part of the grid $\mathcal{B}$ and we can simply take the function values associated with them.

In the following paragraphs, we first explain a naïve approach for the general case. Second, a faster approach is presented, which is further optimized afterward. After this, we will address deriving the likelihood and conditional density based on this intermediate result.

The naïve approach to obtaining the function values on the 1-D grid in the 2-D space is to evaluate the interpolation at $\mathcal{I} \times \{\hat{y}\}$. For this, we first take the square root of all values, leading to the matrix $\sqrt{\mathbf{\Lambda}}$. Afterward, we apply the FFT to the values to obtain the Fourier coefficient matrix $\mathbf{C}^{\text{sqrt}}$ for the square root of the interpolation. We can then obtain the desired grid values by evaluating the trigonometric polynomial at $\mathcal{I} \times \{\hat{y}\}$.

The considered lower-dimensional grid comprises $\sqrt{n}$ points in the 2-D case and $n^{d/d_{\underline{x}}}$ points for higher dimensions. For each function evaluation, we add up a weighted combination of all $O(n)$ Fourier coefficients, leading to a total complexity in $O(n^{1.5})$ for the 2-D case and $O(n^{1+d_{\underline{x}}/d})$ for the arbitrary-dimensional case.

To arrive at a faster approach, we shift the trigonometric polynomial to obtain $\underleftarrow{f}_{\hat{y}}^{x,y}(x,0)$ and then calculate the slice at 0, which is easier since $0 \in \mathcal{I}$. An overview of the approach is given in Fig. 2. Black arrows show the steps to derive the parameter of the density at the slice at $\hat{y}$. Blue and red arrows are used for the steps to derive the conditional density or likelihood, respectively. The dashed arrows show an optimization that we address after the basic approach.

In the grid filter, we start out with the function values $\mathbf{\Lambda}$ on the grid, as visualized for an example in Fig. 3a. The first step is to determine the function values of the square

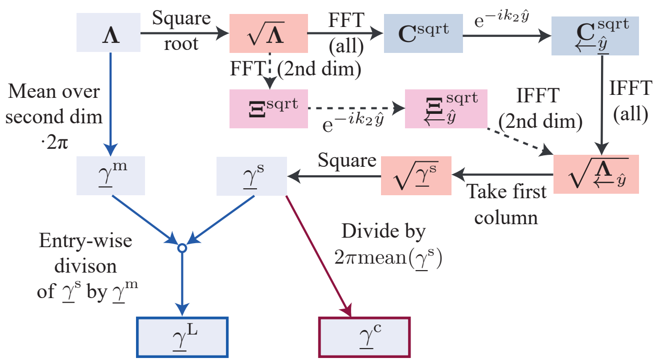


Fig. 2: Illustration of how to get the grid values of the conditional density and likelihood based on the grid values of the joint density.

root of the density on the grid (i.e., $\sqrt{\boldsymbol{\Lambda}}$). The square roots of the grid values are illustrated for the example in Fig. 3b. Then, we perform a 2-D FFT (in general d-D) to obtain the Fourier coefficient matrix $\mathbf{C}^{\mathrm{sqrt}}$. The resulting trigonometric polynomial (describing the square roof of the density) is visualized for our example in Fig. 3c. In this figure, we show the function values of $\sqrt{f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})}$ in red to visualize the relevant slice of the square root of the joint density.

Then, we shift the function by $-\hat{y}$ along the y-axis to obtain the Fourier coefficients of $\sqrt{\underleftarrow{f}_{\hat{y}}^{x,y}(x,y)}$. The shifting operation can be performed directly based on the Fourier coefficients by applying the formula (see [20, Volume I, Ch. 2] for the 1-D case)

$$c_{k_1,k_2}^{\mathrm{shifted}} = c_{k_1,k_2}\mathrm{e}^{-ik_2\hat{y}} \tag{2}$$

to all Fourier coefficients, leading to the Fourier coefficient matrix $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{sqrt}}$. This operation is in $O(n)$. The formula can also be used for the d-dimensional case by adapting the term in the exponent. The resulting trigonometric polynomial for our example is shown in Fig. 3d along with the slice of the density, which is now at a different location.

Then, we apply an IFFT to $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{sqrt}}$ to obtain the function values $\sqrt{\underleftarrow{\boldsymbol{\Lambda}}_{\hat{y}}}$ on the original 2-D grid, as shown in Fig. 3e. Next, we take only the grid values for $y = 0$ and square them to obtain $\underline{\gamma}^{\mathrm{s}}$. The result for our example is shown along with the corresponding interpolation in Fig. 3f. In practice, one can omit to calculate all other function values on the grid when calculating the IFFT. The overall computational complexity is in $O(n \log n)$ because no more expensive operations than the FFT and IFFT are involved. This is better than $O(n^{1.5})$ or $O(n^{1+d_{\underline{x}}/d})$ required for the naïve approach.

There is also a more efficient way to determine the grid values for the slice of the density. We never need to perform FFTs and IFFTs over all dimensions. Instead, we can restrict this operation to the dimension (or dimensions) of $\hat{y}$, which are those we would like to shift. Thus, we apply a 1-D (or $d_{\underline{y}}$-D) FFT to obtain $\boldsymbol{\Xi}^{\mathrm{sqrt}}$ based on $\sqrt{\boldsymbol{\Lambda}}$. We can then perform the shifting operation using (2), as explained above. Afterward, we perform an IFFT to obtain $\sqrt{\underleftarrow{\boldsymbol{\Lambda}}_{\hat{y}}}$. The rest of the steps are identical to those described before. The complexity of this approach is also in $O(n \log n)$.

It should be noted that none of the operations involved introduces approximation errors. Thus, the function values on the 1-D grid are precisely the function values we would have obtained if we had directly evaluated the trigonometric polynomial, as described in the naïve approach.

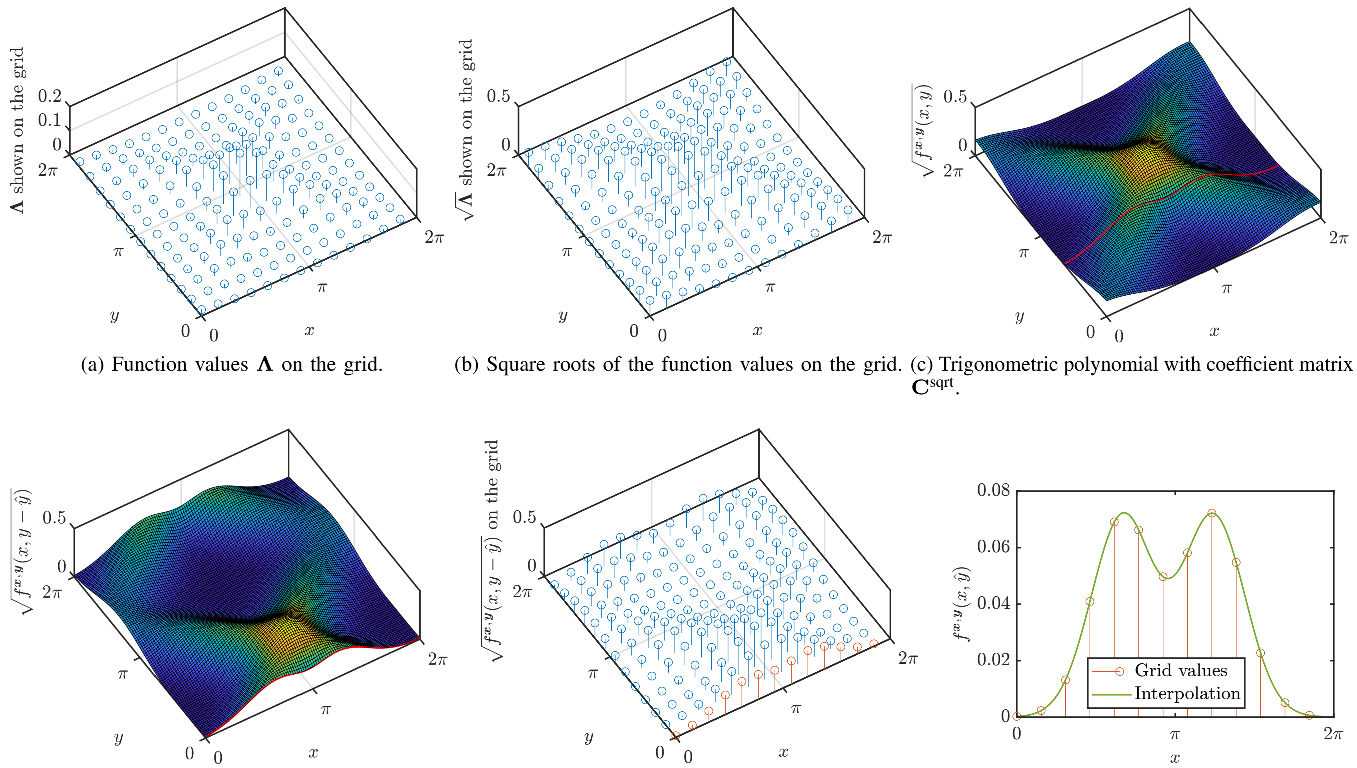
To obtain the conditional density, we normalize the new 1-D density. As proven in the appendix of [12], the integral over the interpolation of the density used in the grid filter (i.e., the square of the trigonometric polynomial interpolating the square roots of the grid values) can be determined by taking the mean over all grid values and multiplying the result by 2π. By dividing $\underline{\gamma}^{\mathrm{s}}$ by $2\pi\mathrm{mean}(\underline{\gamma}^{\mathrm{s}})$, one obtains the vector $\underline{\gamma}^{\mathrm{c}}$ representing the normalized conditional density. For higher dimensions, one may obtain a tensor $\boldsymbol{\Lambda}^{\mathrm{s}}$. In the higher dimensional case, the mean has to be calculated over all entries of the tensor, and the result needs to be scaled by $(2\pi)^{d_{\underline{x}}}$. Again, no approximations are involved. The complexity is in $O(n \log n)$ because of the FFT employed to obtain $\underline{\gamma}^{\mathrm{s}}$.

To derive the likelihood, we first determine the grid values of the marginalized density $f^{\boldsymbol{x}}(x)$. For this, we use the formula for the integral explained earlier. To marginalize out the second dimension, we take the mean over all grid values along the second dimension and multiply the result by 2π. This leads to the coefficient vector $\underline{\gamma}^{\mathrm{m}}$ for the marginalized result. For vector-valued $\underline{y}$, all $d_{\underline{y}}$ dimensions have to be marginalized out by taking the average over all these dimensions and multiplying the result by $(2\pi)^{d_{\underline{y}}}$.

To obtain the function values of the likelihood at arbitrary points, one can calculate the Fourier coefficient vectors $\underline{c}^{\mathrm{s,sqrt}} = \mathrm{FFT}(\sqrt{\underline{\gamma}^{\mathrm{s}}})$ and $\underline{c}^{\mathrm{m,sqrt}} = \mathrm{FFT}(\sqrt{\underline{\gamma}^{\mathrm{m}}})$ and evaluate both trigonometric polynomial at the input points. By dividing the squared function values of the trigonometric polynomial with coefficient vector $\underline{c}^{\mathrm{s,sqrt}}$ by the squared function values of the trigonometric polynomial with the coefficient vector $\underline{c}^{\mathrm{m,sqrt}}$, the function values of the likelihood can be obtained without introducing approximation errors.

It can also be of interest to have a single vector of grid values for, e.g., the update step of the grid filter. For this, we divide the vector $\underline{\gamma}^{\mathrm{s}}$ comprising the function values on the slice of the density entry-wise by the function values of the marginalized density $\underline{\gamma}^{\mathrm{m}}$. The result is the vector $\underline{\gamma}^{\mathrm{L}}$ comprising the precise function values of the likelihood at the grid points. However, unlike the approach discussed in the previous paragraph, applying our standard interpolation scheme to the grid values does generally not lead to the precise values of the likelihood. The reason for this is that it may not be possible to precisely describe the result of the division using the given number of grid points. The complexity is in $O(n \log n)$ both for determining the parameter vectors $\underline{c}^{\mathrm{s,sqrt}}$ and $\underline{c}^{\mathrm{m,sqrt}}$ for the precise solution and obtaining $\underline{\gamma}^{\mathrm{L}}$ for the approximation.

Entries of $\underline{\gamma}^{\mathrm{m}}$ may be zero, which makes it impossible to perform the division. In this case, it may not be possible to approximate the likelihood well using a single vector. To be



(a) Function values $\mathbf{\Lambda}$ on the grid.

(b) Square roots of the function values on the grid.

(c) Trigonometric polynomial with coefficient matrix $\mathbf{C}^{\mathrm{sqrt}}$.

(d) Shifted trigonometric polynomial with coefficient matrix $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{sqrt}}$.

(e) Square root of the function values on the grid of the shifted function $\sqrt{\underleftarrow{\mathbf{\Lambda}}_{\hat{y}}}$. The values of $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ are highlighted in red.

(f) The values of $\underline{\gamma}^{\mathrm{s}}$ are shown in red. The interpolation disallowing negative function values is shown in green.

Fig. 3: Illustration of the steps to obtain a conditional density for grid-based densities.

able to perform the division and thus obtain a single vector nonetheless, one may consider shifting the grid by an offset so that no value is zero anymore. The shifting operation can be implemented by transforming the vectors $\underline{\gamma}^{\mathrm{s}}$ and $\underline{\gamma}^{\mathrm{m}}$ to Fourier coefficients via an FFT, performing the shifting operation via (2), and then transforming the results back via an IFFT.

B. Approach for the Density Representations of the Fourier Filters

In this subsection, we first consider the representation used in the IFF and then the one used in the SqFF.

1) Representation Used in the IFF: In a naïve approach, one may consider transforming the Fourier coefficients to grid values, then performing the steps in Sec. III-A, and transforming the grid values back at the end. However, this only works if applying an IFFT to $\mathbf{C}^{\mathrm{id}}$ results only in nonnegative values, which is not guaranteed in this representation. If the Fourier coefficients were obtained by applying an FFT to values of the density on the grid, the IFFT will (when disregarding numerical imprecisions) always yield nonnegative values. However, for example, when the coefficient matrix is the result of an update step, negative values may be obtained. In this case, one cannot take the square root of the grid values. Also note that even when this approach is applicable, it may not yield the expected result. By employing the approach of Sec. III-A, one uses the interpolation involving the square root, even though the density is represented directly using a trigonometric polynomial in the IFF.

A solution specifically tailored to the density representation of the IFF is shown in Fig. 4. To obtain the Fourier coefficients of $f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})$, we first use (2) to shift the density and obtain $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{id}}$ (which describes $\underleftarrow{f}_{\hat{y}}^{x,y}(x,y)$) based on $\mathbf{C}^{\mathrm{id}}$. Obtaining the parameters of a slice of the joint density is easier in the representation based on grid values. Therefore, we perform a 1-D IFFT along the second dimension, leading to $\underleftarrow{\boldsymbol{\Xi}}_{\hat{y}}^{\mathrm{id}}$. In this representation, we can obtain the slice of the density in a similar way as in the previous subsection. By taking the first column (i.e., the column describing the function at $y=0$), we obtain the coefficient vector $\underline{c}^{\mathrm{s,id}}$ describing $\underleftarrow{f}_{\hat{y}}^{x,y}(x,0)$. No approximation errors were introduced in any of the operations so far. While the function described by $\underline{c}^{\mathrm{s,id}}$ can have negative function values, this may only occur if the function described by $\mathbf{C}^{\mathrm{id}}$ also has negative function values.

To obtain the conditional density, we normalize the density represented by $\underline{c}^{\mathrm{s,id}}$. As discussed in [11], the integral over a trigonometric polynomial is equal to $2\pi c_0^{\mathrm{s,id}}$ (or $(2\pi)^{d_{\underline{x}}} c_{\underline{0}}^{\mathrm{s,id}}$ in the multidimensional case, with $\underline{0}$ being the index vector comprising only zeros). As evident from (1), we can scale the trigonometric polynomial directly describing the function by

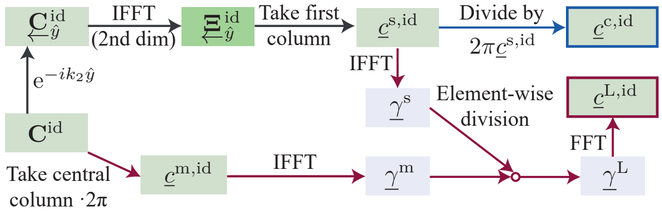


Fig. 4: Approach to obtain the coefficient vectors of the conditional density and likelihood for the representation used in the IFF.

multiplying or dividing all coefficients by a constant. Thus, we can perform the normalization and obtain the coefficients for the conditional density via

$$\underline{c}^{\mathrm{c,id}} = \frac{1}{2\pi c_0^{\mathrm{s,id}}} \underline{c}^{\mathrm{s,id}} .$$

For the likelihood, we first derive the coefficients of the marginalized density. As discussed in [21], one can marginalize out the second dimension of a trigonometric polynomial by taking the central column of the Fourier coefficient tensor and multiplying it by 2π (in the multidimensional case, we need to take a subtensor and multiply the result by $(2\pi)^{d_{\underline{y}}}$). This leads to the coefficient vector $\underline{c}^{\mathrm{m,id}}$ for the marginalized density. Then, we have the coefficient vectors $\underline{c}^{\mathrm{s,id}}$ for $f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})$ and $\underline{c}^{\mathrm{m,id}}$ for $f^{\boldsymbol{x}}(x)$. The optimal function value for any point x can be obtained by evaluating both trigonometric polynomials at x and then performing the division.

In this representation, obtaining a single coefficient vector, e.g., for use in the update step of the IFF, is non-trivial since there is no simple formula for the Fourier coefficients of the result of the division of two trigonometric polynomials. However, we can apply an IFFT to both vectors to obtain the vectors $\underline{\gamma}^{\mathrm{s}}$ and $\underline{\gamma}^{\mathrm{m}}$ comprising the function values on the grid. Then, we perform a division to obtain a vector of function values on the grid $\underline{\gamma}^{\mathrm{L}}$. This vector can be transformed to the Fourier coefficient vector $\underline{c}^{\mathrm{L,id}}$ via an FFT. As before, the vector $\underline{\gamma}^{\mathrm{m}}$ may include negative values or zero. While negative values do not prevent any operation, zero values make the division impossible. If any value is zero, a similar strategy could be considered as for the grid filter. Also note that as in the grid filter, $\underline{\gamma}^{\mathrm{L}}$ may not be sufficient to describe the precise result of the division, and thus, $\underline{c}^{\mathrm{L,id}}$ does generally not describe the likelihood precisely.

Obtaining the parameter vector $\underline{c}^{\mathrm{s,id}}$ is in $O(n \log n)$ due to the IFFT involved. Determining the conditional density and likelihood does not involve any more expensive operation, and thus, the total complexity for both is also in $O(n \log n)$.

2) Representation Used in the SqFF: In the representation used by the SqFF, one can obtain nonnegative function values on the grid by applying an IFFT to the grid values and squaring them. This way, one can temporarily switch to the grid-based representation. In this representation, one can perform the operations according to Fig. 2, take the square roots of the values, and apply an FFT to obtain a vector of Fourier coefficients for the square root of the function again. While this approach is not optimal, we will first discuss it before explaining an approach similar to that for the IFF. Note that despite the differences, both approaches are in $O(n \log n)$.

Approach Based on that for the Grid Filter: By applying a 2-D IFFT to $\mathbf{C}^{\mathrm{sqrt}}$ and then squaring the resulting function values, we would obtain the matrix $\boldsymbol{\Lambda}$. If we then proceeded as in Fig. 2, we would calculate the square roots of the values and perform an FFT again, which would undo the previous steps. Therefore, we first apply the shifting operation to obtain $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{sqrt}}$ from $\mathbf{C}^{\mathrm{sqrt}}$. Then, we can perform an IFFT to obtain the function values on the grid $\sqrt{\underleftarrow{\boldsymbol{\Lambda}}_{\hat{y}}}$. From these, we can obtain the vector comprising the square roots of the grid values $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ by discarding all columns of the matrix except the first.

To obtain the conditional density, one can determine the Fourier coefficient vector $\underline{c}^{\mathrm{s,sqrt}}$ from $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ via an FFT. As proven in [13, Sec. 6.2.2], one can normalize a density represented by a Fourier coefficient vector by dividing it by $\sqrt{2\pi}$ times its norm. Thus,

$$\underline{c}^{\mathrm{c,sqrt}} = \frac{1}{\sqrt{2\pi}\, \|\underline{c}^{\mathrm{s,sqrt}}\|} \underline{c}^{\mathrm{s,sqrt}} \tag{3}$$

can be used to obtain the parameters of the conditional density.

For the likelihood, we only consider how to get a single vector describing an approximate result. For this, we perform a 2-D IFFT and square the values to obtain $\boldsymbol{\Lambda}$. Then, we perform the marginalization by calculating the mean over the second dimension of $\boldsymbol{\Lambda}$ and multiplying the result by 2π. By taking the square root afterward, we obtain $\sqrt{\underline{\gamma}^{\mathrm{m}}}$. By dividing $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ by $\sqrt{\underline{\gamma}^{\mathrm{m}}}$ entry-wise (see above how to handle the case in which an entry is zero) we can obtain the square roots of the function values of the likelihood on the grid. By applying an FFT, we can obtain the Fourier coefficient vector $\underline{c}^{\mathrm{L,sqrt}}$ for the likelihood.

Approach Similar to That for the IFF: As an alternative, we shall now consider a more direct approach based on the approach for the representation used in the IFF. An overview of the approach is given in Fig. 5.

In the first step, we shift the function via (2) to obtain $\underleftarrow{\mathbf{C}}_{\hat{y}}^{\mathrm{sqrt}}$. Then, we apply a 1-D IFFT along the second dimension to obtain $\underleftarrow{\boldsymbol{\Xi}}_{\hat{y}}^{\mathrm{sqrt}}$. As in Sec. III-B.1, we take the first column to obtain the Fourier coefficient vector $\underline{c}^{\mathrm{s,sqrt}}$ for the function $\sqrt{f^{\boldsymbol{x},\boldsymbol{y}}(x, \hat{y})}$. As before, we can derive the Fourier coefficient vector for the conditional density according to (3).

The marginalization cannot be performed directly based on the Fourier coefficients because the coefficients describe the square root of the density. Therefore, we calculate the Fourier coefficients directly representing the function from the Fourier coefficients for the square root of the function by convolving the matrix $\mathbf{C}^{\mathrm{sqrt}}$ with itself to obtain $\mathbf{C}^{\mathrm{id}}$. We use the full result of the discrete convolution, which has $(2m-1)^2 \leq 4n$ (for d-D $(2m-1)^d \leq 2^d n$) entries. Because the factor by which the number of coefficients increases only depends on the dimension, we shall treat it as a constant. The discrete convolution is thus in $O(n \log n)$. Using the full

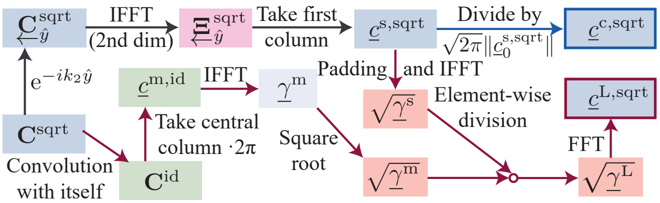


Fig. 5: Approach to obtain the coefficient vectors of the conditional density and likelihood for the representation used in the SqFF.

result provides the advantage that the matrix $\mathbf{C}^{\mathrm{id}}$ describes the same function as the one represented by $\mathbf{C}^{\mathrm{sqrt}}$. Then, we can perform the marginalization by taking the central column and multiplying the result by 2π, leading to $\underline{c}^{\mathrm{m,id}}$.

Now, we can evaluate both $f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})$ and $f^{\boldsymbol{x}}(x)$ precisely for any x. For the former, we evaluate the trigonometric polynomial with the coefficient vector $\underline{c}^{\mathrm{s,sqrt}}$ and square the result. For the latter, we evaluate the trigonometric polynomial $\underline{c}^{\mathrm{m,id}}$ (no squaring is involved in this). By dividing the function values of the $f^{\boldsymbol{x},\boldsymbol{y}}(x,\hat{y})$ by those of $f^{\boldsymbol{x}}(x)$, the precise function values of the likelihood can be obtained.

To approximate the likelihood using a single coefficient vector, we can proceed as follows. We first perform an IFFT to obtain the vector $\underline{\gamma}^{\mathrm{m}}$ comprising the function values of $f^{\boldsymbol{x}}(x)$ on the grid. Since no approximation errors were introduced so far, the entries of the vector $\underline{\gamma}^{\mathrm{m}}$ are guaranteed to be nonnegative. Hence, we can calculate the square roots of the function values.

Note that the vector $\underline{\gamma}^{\mathrm{m}}$ is of length $2m-1$ due to the discrete convolution employed to obtain $\mathbf{C}^{\mathrm{id}}$. Therefore, we first pad $\underline{c}^{\mathrm{s,sqrt}}$ with zeros so that the result of the IFFT is a vector of the same length as $\underline{\gamma}^{\mathrm{m}}$. Then, we divide $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ entry-wise by the square roots of the values in the vector $\underline{\gamma}^{\mathrm{m}}$ to obtain the vector $\sqrt{\underline{\gamma}^{\mathrm{L}}}$. By applying an FFT to $\sqrt{\underline{\gamma}^{\mathrm{L}}}$, we obtain $\underline{c}^{\mathrm{L,sqrt}}$. As for the other representations, $\underline{c}^{\mathrm{L,sqrt}}$ does generally not describe the precise likelihood. Also, the case in which an entry of $\underline{\gamma}^{\mathrm{m}}$ is zero may require a strategy similar to the one explained for the grid-based representation.

In some cases, one may be interested in a vector of length m instead of $2m-1$. Note that simply discarding every second grid value in $\underline{\gamma}^{\mathrm{m}}$ after the IFFT leads to incorrect results. Such an approach would only work if one had $2m$ grid points, but in this case, one has $2m-1$ grid points. As a different approach, one may attempt to only determine a part of the full convolution result of $\mathbf{C}^{\mathrm{sqrt}}$ with itself. However, the steps that follow may then not always work as the IFFT can then lead to negative function values.

A working approach is as follows. One applies a 2-D FFT to $\mathbf{C}^{\mathrm{sqrt}}$ to obtain $\sqrt{\boldsymbol{\Lambda}}$ on a grid with m^2 grid points. One can then square the values to obtain $\boldsymbol{\Lambda}$ and then calculate the mean over the second dimension and multiply the result by 2π to obtain a vector $\underline{\gamma}^{\mathrm{m}}$ with m entries describing the marginalized result. Note that calculating the square of the grid values may introduce an approximation error as it may not be possible to perfectly describe the result of the squared function using only the obtained grid values. Since $\underline{\gamma}^{\mathrm{m}}$ has m entries, it is not necessary to pad $\underline{c}^{\mathrm{s,sqrt}}$, and one can directly apply an IFFT to $\underline{c}^{\mathrm{s,sqrt}}$ and divide $\sqrt{\underline{\gamma}^{\mathrm{s}}}$ by $\sqrt{\underline{\gamma}^{\mathrm{m}}}$ to obtain a vector for the likelihood with m entries. Finally, by applying an IFFT to the result, one obtains a coefficient vector of length m.

IV. Conclusion and Outlook

In this paper, we derived formulae for conditional densities and likelihoods for the representations employed in the Fourier and grid filters. Similar results were obtained for all the considered representations. For both conditional densities and likelihoods, we first considered how to parameterize the 1-D slice of the joint density obtained by setting the second input argument to $\hat{y}$. The key idea is that one can first shift the joint density by $-\hat{y}$ along the second dimension and then consider the slice obtained by setting the second input argument of the shifted density to 0.

Based on the parameters of the function along this slice, one can obtain the parameters of the conditional density. For this, one only has to perform an operation on the parameters that ensures the normalization of the function. No approximations are required, and the resulting parameter vector for each considered representation precisely describes the conditional density.

For the likelihood, the precise function values can only be obtained by considering two functions in the respective parametric form and dividing the function values of the first function by those of the second at all considered points. We further explored ways to provide a single parameter vector for the likelihood. While the obtained vectors do not describe the likelihoods perfectly, they are useful for, e.g., the update steps of the filters.

The formulae presented in this paper extend our theory on the Fourier and grid filters by allowing us to obtain conditional densities and likelihoods from joint densities. For all representations, Matlab code to obtain the parameters of the function along the slice, the marginal density, the conditional density, and the likelihood is available as part of the latest version of libDirectional [22].

Future work could investigate other operations and properties concerning the parametric families of densities used in the Fourier and grid filters. Further, one may consider densities based on grids or orthogonal basis functions on other manifolds.

Acknowledgments

We thank Kanti Mardia for raising awareness of this missing piece of theory on these kinds of densities. This work is partially funded by the Helmholtz AI Cooperation Unit within the scope of the project *Ubiquitous Spatio-Temporal Learning for Future Mobility (ULearn4Mobility)*.

References

[1] W. Schmidt, "Statistische Methoden beim Gefügestudium krystalliner Schiefer," *Sitzungsberichte der Kaiserlichen Akademie der Wissenschaften in Wien – mathematisch-naturwissenschaftliche Classe*, vol. 126, pp. 515–539, Jul. 1917.

[2] J. A. Carta, C. Bueno, and P. Ramírez, "Statistical Modelling of Directional Wind Speeds Using Mixtures of von Mises Distributions: Case Study," *Energy Conversion and Management*, vol. 49, no. 5, pp. 897–907, May 2008.

[3] K. V. Mardia, J. I. Foldager, and J. Frellsen, "Directional Statistics in Protein Bioinformatics," in *Applied Directional Statistics*. Chapman & Hall/CRC, 2018, pp. 1–23.

[4] A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. W. R. Nelson, A. Bridgland, H. Penedones, S. Petersen, K. Simonyan, S. Crossan, P. Kohli, D. T. Jones, D. Silver, K. Kavukcuoglu, and D. Hassabis, "Improved Protein Structure Prediction Using Potentials From Deep Learning," *Nature*, vol. 577, no. 7792, pp. 706–710, Jan. 2020, number: 7792 Publisher: Nature Publishing Group.

[5] K. V. Mardia and J. Voss, "Some Fundamental Properties of a Multivariate von Mises Distribution," *Communications in Statistics—Theory and Methods*, vol. 43, no. 6, pp. 1132–1144, Mar. 2014, publisher: Taylor & Francis _eprint: https://doi.org/10.1080/03610926.2012.670353.

[6] A. K. W. Navarro, J. Frellsen, and R. E. Turner, "The Multivariate Generalised von Mises Distribution: Inference and Applications," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI'17. San Francisco, California, USA: AAAI Press, Feb. 2017, pp. 2394–2400.

[7] C. Ferrari, "The Wrapping Approach for Circular Data Bayesian Modeling," Ph.D. dissertation, Università di Bologna, Mar. 2009.

[8] F. Wang and A. E. Gelfand, "Modeling Space and Space-Time Directional Data Using Projected Gaussian Processes," *Journal of the American Statistical Association*, vol. 109, no. 508, pp. 1565–1580, Oct. 2014.

[18] G. Strang, *Computational Science and Engineering*. Wellesley-Cambridge Press, 2007.

[9] S. Kim, A. SenGupta, and B. C. Arnold, "A Multivariate Circular Distribution with Applications to the Protein Structure Prediction Problem," *Journal of Multivariate Analysis*, vol. 143, pp. 374–382, Jan. 2016.

[10] M. Arashi, N. N. Rad, A. Bekker, and W. D. Schubert, "Protein Structure Parameterization via Mobius Distributions on the Torus," *arXiv:2011.13027 [q-bio, stat]*, Nov. 2020, arXiv: 2011.13027.

[11] F. Pfaff, G. Kurz, and U. D. Hanebeck, "Multivariate Angular Filtering Using Fourier Series," *Journal of Advances in Information Fusion*, vol. 11, no. 2, pp. 206–226, Dec. 2016.

[12] F. Pfaff, K. Li, and U. D. Hanebeck, "Fourier Filters, Grid Filters, and the Fourier-Interpreted Grid Filter," in *Proceedings of the 22nd International Conference on Information Fusion (Fusion 2019)*, Ottawa, Canada, Jul. 2019.

[13] F. Pfaff, "Multitarget Tracking Using Orientation Estimation for Optical Belt Sorting," Ph.D. dissertation, 2019.

[14] F. Pfaff, K. Li, and U. D. Hanebeck, "A Hyperhemispherical Grid Filter for Orientation Estimation," in *Proceedings of the 23rd International Conference on Information Fusion (Fusion 2020)*, Virtual, Jul. 2020.

[15] M. Frigo and S. G. Johnson, "The Design and Implementation of FFTW3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005.

[16] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent Robotics and Autonomous Agents Series. The MIT Press, 2005.

[17] F. Pfaff, K. Li, and U. D. Hanebeck, "Estimating Correlated Angles Using the Hypertoroidal Grid Filter," in *Proceedings of the 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2020)*, Virtual, Sep. 2020.

[19] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C—The Art of Science Computing*, 2nd ed., 1992.

[20] A. Zygmund, *Trigonometric Series*, 3rd ed. Cambridge University Press, 2003, vol. 1 and 2.

[21] F. Pfaff, G. Kurz, and U. D. Hanebeck, "Nonlinear Prediction for Circular Filtering Using Fourier Series," in *Proceedings of the 19th International Conference on Information Fusion (Fusion 2016)*, Heidelberg, Germany, Jul. 2016.

[22] G. Kurz, I. Gilitschenski, F. Pfaff, L. Drude, U. D. Hanebeck, R. Haeb-Umbach, and R. Y. Siegwart, "Directional Statistics and Filtering Using libDirectional," *Journal of Statistical Software*, May 2019.