Inverse Gaussian Process Interpolation for High-Quality Assumed Gaussian Filtering

Jiachen Zhou, Daniel Frisch, Uwe D. Hanebeck

***Abstract*— In this paper, we propose a novel Gaussian Assumed Density Filter (GADF) for high-quality state estimation of nonlinear dynamic systems. Our approach focuses on the measurement update, utilizing a non-Gaussian local approximation of the true joint measurement/prior state density. This is achieved through the sequential use of two Inverse Gaussian Processes (IGPs): the first IGP interpolates the means, and based on these results, the second IGP is trained to interpolate the covariances. Together, they fully characterize the conditional Gaussian densities of the hidden state on concrete measurements. Consequently, our method does not require the second Gaussian assumption for the joint density anymore, thereby enhancing filter performance. Moreover, our approach eliminates the need for an explicit likelihood function within the filter step, making it a higher-quality plug-in replacement for the commonly used Linear Regression Kalman Filter (LRKF).**

I. INTRODUCTION

Context: We consider general state estimation for discrete-time stochastic nonlinear dynamic systems on the basis of noise-corrupted measurements. This problem arises in the field of engineering such as robotics, target tracking, and satellite navigation.

Exact Bayesian solutions in closed form are, however, only available for a limited number of special cases, e.g., linear systems corrupted by additive Gaussian noise. In that case, the classic Kalman Filter [1] emerges as the optimal estimator in the sense of a Minimum Mean Square Error (MMSE). For most nonlinear systems suffering from non-additive noise, closed-form analytic solutions are not available. Consequently, approximate approaches are needed to develop efficient and powerful nonlinear Bayesian filters.

A common approach is the use of Gaussian Assumed Density Filters (GADFs) as maintaining the true, in general multi-modal or non-Gaussian, state Probability Density Function (PDF) is impractical. Instead, a reasonable simplification is to represent them as Gaussians recursively over time [2]. During each update, only mean and covariance are propagated to fully characterize the Gaussian distribution, which does not lead to an increasing computational complexity over time as compared to filters operating on the more complex state densities. However, it is still challenging to derive closed-form solutions for the time update, and especially the measurement update. Within a Bayesian measurement update, an explicit likelihood function is required, but deriving it for non-additive measurement noise is difficult. Even if one is available, performing an analytic filter step remains nearly impossible except for linear measurement models suffering from additive Gaussian noise.

All authors are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany jiachen.zhou@kit.edu, daniel.frisch@kit.edu, uwe.hanebeck@kit.edu.

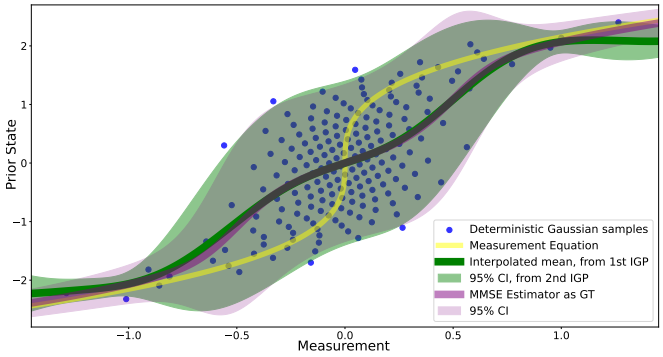

Fig. 1: Example of the proposed non-Gaussian local approximation (green) of the true joint measurement and state density through two IGPs based on the samples (blue) drawn from the true joint. The ground truth (purple) for comparison is the family of conditional Gaussian densities for any concrete measurement (horizontal axis), with mean and variance identical to the true posterior density.

State-of-the-art: The Gaussian Particle Filter (GPF) [3] is a special sequential importance sampling Particle Filter (PF) [4], [5] that approximates posterior distributions by single Gaussians. Monte Carlo integration is employed to approximate the desired moments for moment matching. The GPF converges towards the MMSE state estimator with an infinite number of random samples, making it asymptotically optimal in terms of sample size. However, it becomes computationally intractable in larger state spaces due to the curse of dimensionality. Moreover, an explicit likelihood function is still needed for sample re-weighting.

To further ease the measurement update, likelihood-free approaches have been adopted. They involve a linear approximation of the nonlinear mapping between the prior state and noisy measurements, followed by the direct usage of the Kalman Filter formulas, resulting in Nonlinear Kalman Filters (NKFs). Although the need for an explicit likelihood function is avoided, the problem is that it might be an oversimplification depending on the "strength" of the nonlinearity, leading to reduced estimation performance compared to the general GADFs without such linearization.

One way to perform such linearization is explicit linearization based on Taylor series approximation, commonly used by Extended Kalman Filters (EKFs) and its variants [6], [7]. In contrast, NKFs based on statistical linearization implicitly linearize the measurement model by approximating the joint

density of measurement and state with a Gaussian. When the required first-order and second-order moments are calculated by sample-based density representations, all corresponding NKFs fall into the class of Linear Regression Kalman Filters (LRKFs) [8], [9]. Examples are the Unscented Kalman Filter (UKF) [10], the Gaussian Filter (GF) [11] and the Smart Sampling Kalman Filter (S^2KF) [12], [13].

The advantages of the LRKFs include efficiency and ease of implementation. Compared to PFs and GPFs, the sample degeneration problem is avoided because of their likelihood-free nature. Additionally, their generic design is applicable to any system and measurement equation. However, the main drawback is further decrease of state estimation quality due to the approximation of continuous state and noise densities using only a limited number of samples.

A more advanced GADF, called Progressive Gaussian Filter (PGF), avoids the second Gaussian assumption, thereby improving estimation quality without such linearization errors [14], [15]. It decomposes the measurement update into several sub-updates, gradually incorporating measurement information into state estimates. In [15], an explicit likelihood function is needed within the progression mechanism. Both PGF approaches, however, may introduce cumulative errors due to multiple intermediate Gaussian approximations within each recursion step. Moreover, the proper determination of step sizes may increase computational complexity.

Contribution: In this work, we propose a GADF with a novel likelihood-free measurement update. Our method does not rely on the second Gaussian assumption for the true joint measurement/state density, leading to better performance than the LRKFs, which heavily depend on this simplification. Instead, we perform a non-Gaussian local approximation of the true joint density through a sequential use of two Inverse Gaussian Processs (IGPs). The first IGP interpolates the means, and the second IGP, trained based on these results, interpolates the covariances. Together, they fully characterize conditional Gaussian densities of the hidden state on concrete measurements. This approach uses a limited number of high-quality equally weighted deterministic samples from the true joint density. A suitable sampling strategy is employed to ensure reproducible results and improve sample efficiency.

We exploit the interpolation capabilities of Gaussian Processs (GPs) to infer hidden system states from concrete measurements. This method suggests an inverse application of standard GPs, which typically predict outputs. Consequently, we refer to this method as the IGPs. Employing IGP models enables us to derive an approximate Gaussian posterior state densities. It requires, however, utilizing measurement realizations as model inputs, and realizations of the prior state as outputs. To improve visualization, we adopt a rotated Cartesian coordinate system throughout this paper, positioning measurements on the horizontal axis, and prior states or their computations on the vertical axis.

We train two IGPs sequentially and utilize only the predictive mean results of each posterior IGP model as the interpolated mean and covariance values, respectively. These interpolation results are then used to directly parameterize Gaussian posterior distributions through moment matching.

Our proposed filter can be integrated as a higher-quality plug-in replacement for the commonly used LRKF during an online filter step. It can also operate independently of actual received measurements, enabling offline pre-interpolations for a set of different prior state and measurement noise densities. These results can be stored in a cache for online application, which is outside the scope of this paper.

As demonstrated through evaluations by means of a canonical benchmark example, the performance of the proposed filter closely approaches that of the optimal MMSE estimator. Moreover, its sample-based nature significantly enhances filter versatility. In other words, the assumption of additive Gaussian noise can be relaxed and can be non-Gaussian and non-additive. This capability is further validated by an additional evaluation involving multiplicative measurement noise. Additionally, state-correlated Gaussian measurement noise can be effectively handled with an appropriate Gaussian sampling strategy, provided that the joint Gaussian density of the prior state and measurement noise is known in advance.

II. Problem Formulation

We consider estimating the hidden state $\underline{\boldsymbol{x}}_k$ of a discrete-time stochastic dynamic system based on noisy measurements, consisting of a time update (or prediction step) and a measurement update (or filter step). This work specifically addresses the measurement update. The relationship between the measurement random vector $\underline{\boldsymbol{y}}_k$, the hidden system state $\underline{\boldsymbol{x}}_k$, and the measurement noise $\underline{\boldsymbol{v}}_k$ is described by the given generative nonlinear measurement equation

$$\underline{\boldsymbol{y}}_k = \underline{h}_k(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{v}}_k) \,, \tag{1}$$

where $\underline{h}_k(\cdot,\cdot)$ denotes the vector-valued measurement nonlinearity and maps the state and noise to the concrete measurement $\underline{\tilde{y}}_k$, a realization of the random vector $\underline{\boldsymbol{y}}_k$.

A predicted state density is received by performing a time update and then approximating it as a Gaussian through moment matching, i.e., the Gaussian PDF of the state at time step k conditioned on the measurements $\underline{\tilde{y}}_1, \ldots, \underline{\tilde{y}}_{k-2}, \underline{\tilde{y}}_{k-1}$

$$f_k^{\mathrm{p}}(\underline{\boldsymbol{x}}_k) = f(\underline{\boldsymbol{x}}_k \mid \underline{\tilde{y}}_{1:k-1}) \approx \mathcal{N}(\underline{\boldsymbol{x}}_k; \hat{\underline{x}}_k^{\mathrm{p}}, \boldsymbol{\Sigma}_k^{\mathrm{p}}) \,. \tag{2}$$

The goal is to correct this Gaussian prior state estimate by incorporating a newly received measurement $\underline{\tilde{y}}_k$ at time step k. In general, it is done by using the Bayes' rule. The generative measurement model (1) is first converted into a probabilistic model as the conditional density $f(\underline{\boldsymbol{y}}_k \mid \underline{\boldsymbol{x}}_k)$ over $\underline{\boldsymbol{y}}_k$ on $\underline{\boldsymbol{x}}_k$, which turns into a likelihood function $f_k^{\mathrm{h}}(\underline{\tilde{y}}_k \mid \underline{\boldsymbol{x}}_k)$ for a given specific measurement $\underline{\tilde{y}}_k$. However, as mentioned before, an explicit description of the likelihood function is hard to derive in the case of non-additive measurement noise. To address this, we adopt a likelihood-free point of view, which instead operates directly on the joint density of prior state and measurement $f_k^{x,y}(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{y}}_k \mid \underline{\tilde{y}}_{1:k-1})$, so the corrected state estimate can be written as

$$f_k^{\mathrm{e}}(\underline{\boldsymbol{x}}_k) = f(\underline{\boldsymbol{x}}_k \mid \underline{\tilde{y}}_{1:k}) = \frac{f_k^{x,y}(\underline{\boldsymbol{x}}_k, \underline{\tilde{y}}_k \mid \underline{\tilde{y}}_{1:k-1})}{f_k^{y}(\underline{\tilde{y}}_k \mid \underline{\tilde{y}}_{1:k-1})} \,, \tag{3}$$

in which a newly received concrete measurement $\underline{\tilde{y}}_k$ determines where to condition the joint distribution in order to get the posterior state density as the corrected state estimate.

Considered problem: Directly deriving the true joint density is still burdensome. Nevertheless, samples drawn from it are easily available, regardless of whether deterministic or random sampling methods are used. As a result, the problem at hand is to locally approximate this true joint density using high-quality limited-quantity samples drawn from it. The key idea to solve this problem is given in the next section. By conditioning this approximate joint density representation on the actual received measurement subsequently, we aim to derive an approximate posterior state density. In addition, this posterior state density can take an arbitrary form and is not necessarily Gaussian, even when starting with a Gaussian prior. Therefore, it is re-approximated as a single Gaussian distribution through moment matching. This approach ensures computational consistency in recursive processing.

III. Key Idea And Groundwork

In this paper, we address the approximation for the underlying true joint density of prior state and measurement in the context of nonlinear Bayesian recursive state estimation. The core idea is to fully characterize a conditional Gaussian distribution over $\underline{\boldsymbol{x}}_k$ on the given specific measurement $\underline{\tilde{y}}_k$ using the interpolated mean vector and covariance matrix. To do that, interpolation techniques are essential. They work with samples drawn from the true joint measurement/state density to construct a continuous and smooth representation of the mean vector and covariance matrix, both as functions of the measurement random variable.

To achieve effective interpolation, a rotated Cartesian coordinate system of the joint measurement/prior state space is first employed, placing measurements on the horizontal axis, and states or their computations on the vertical axis. Following this characterization procedure, the resulting approximate joint measurement/state density is then obtained by introducing additional prefactors.

GPs are selected as the interpolation technique for the purpose of backward inference. We provide a concise introduction to their notation and prediction[1] models. As a nonparametric model, GPs do not assume a specific functional form with a fixed number of parameters, but instead, they define a distribution over possible functions, which allows them to dynamically and flexibly scale their complexity based on the data. Generally, one aims to infer a scalar latent function h from noisy observations $y_i = h(\underline{x}_i) + \varepsilon_i$, $i = 1, \ldots, N$, where inputs are $\underline{x}_i \in \mathbb{R}^n, n \in \mathbb{Z}^+$ and noise $\varepsilon_i \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. A GP is completely specified by its mean function $m(\cdot)$ and positive semi-definite covariance function $k(\cdot, \cdot)$, also called kernel. Throughout this paper, we consider a zero prior mean function and the widely used Squared Exponential (SE) kernel, parameterized by a diagonal matrix of characteristic length scales for each input dimension $\boldsymbol{\Lambda}$, and a scalar variance α^2 of the latent function, as its hyperparameters $\underline{\theta}$,

$$k(\underline{x}, \underline{x}') = \alpha^2 \cdot \exp\left(-\frac{1}{2}(\underline{x} - \underline{x}')^\top \boldsymbol{\Lambda}^{-1}(\underline{x} - \underline{x}')\right) . \quad (4)$$

It yields smooth inference results and its structure is mathematically straightforward, making more intuitive hyperparameter tuning. The posterior GP model's predictive distribution for the random variable, specifically the function value $\boldsymbol{h}(\underline{x}_*)$ at an arbitrary input $\underline{x}_*$, is a Gaussian distribution characterized by exact mean and variance representations

$$\mathbb{E}[\boldsymbol{h}(\underline{x}_*)] = m_h(\underline{x}_*) = \underline{k}_*^\top (\mathbf{K} + \sigma_\varepsilon^2 \cdot \mathbf{I})^{-1} \underline{Y} , \quad (5)$$

$$\mathrm{var}[\boldsymbol{h}(\underline{x}_*)] = \sigma_h^2(\underline{x}_*) = k_{**} - \underline{k}_*^\top (\mathbf{K} + \sigma_\varepsilon^2 \cdot \mathbf{I})^{-1} \underline{k}_* , \quad (6)$$

with $\underline{k}_* \coloneqq k(\underline{X}, \underline{x}_*)$, $k_{**} \coloneqq k(\underline{x}_*, \underline{x}_*)$. $\mathbf{K} \in \mathbb{R}^{N \times N}$ is the quadratic kernel matrix with elements $K_{ij} \coloneqq k(\underline{x}_i, \underline{x}_j)$. Moreover, $\underline{X} = [\underline{x}_1^\top, \ldots, \underline{x}_N^\top]^\top$ are the N inputs for training, and $\underline{Y} = [y_1, \ldots, y_N]^\top$ are the corresponding outputs.

Various standard libraries available in many programming languages, such as `GPyTorch` [16], `GPy` [17] in `Python` and `GaussianProcesses` [18] in `Julia`, not only facilitate the standard implementation of GPs, but also allow for extensive modifications to meet our specific purposes.

IV. Inverse Gaussian Process Interpolation

So far, we established the foundational basis for this work. We will now introduce the proposed novel likelihood-free measurement update. The first step is to justify the sequential use of two IGPs needed to interpolate the mean and covariance values, respectively, for moment matching.

A. Sequential Use of Two Inverse Gaussian Processes

Our goal is to derive conditional Gaussians of the hidden state by conditioning the approximate non-Gaussian joint density on received measurements. However, a single posterior IGP model, trained on samples drawn from the true joint density, provides a suboptimal joint density approximation by directly using its inference results, i.e., both mean and covariance, for moment matching conditional Gaussians. In regions where samples are densely scattered, a single IGP can interpolate confidently, leading to lower uncertainty in its predictive mean results, as represented by the standard deviations of the individual conditional Gaussians. Conversely, in areas where samples are sparse, the model has less informative data for training, leading to higher uncertainty in its predictions. Therefore, more samples should have been drawn from the resulting approximate joint density, despite the sparsity of the actual data in these areas. Based on these observations, one can infer that the covariances of the individual conditional Gaussians must be independently estimated with additional care.

Hence, two successive IGPs are required to accurately capture the true joint density: one dedicated to interpolating the means, followed by the other exclusively for the covariances.

[1]The term "prediction" here refers to the process of estimating the values of a latent function at new, unseen inputs based on training data. In contrast, the expression "prediction step" in the sense of Bayesian state estimation involves propagating the last known state estimate to the present through the use of state-transition density and Chapman-Kolomogorov equation.

B. Deterministic Gaussian Samples

Training two IGPs necessitates samples derived from the true joint measurement/state density $f_k^{x,y}(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{y}}_k \mid \underline{\tilde{y}}_{1:k-1})$, while directly drawing samples from it is troublesome. For that reason, we choose to draw samples from the joint density of the prior state and measurement noise first, because this approach is more tractable by assuming a commonly-used density for the noise. Similar to LRKFs, the drawn samples are then propagated through the measurement equation to yield corresponding measurement realizations, which enables an indirect sampling from the true joint density.

The measurement noise is assumed to be Gaussian distributed. Various methods exist for computing deterministic Gaussian Dirac mixture approximations, such as those employed by the UKF, the GF and the Cubature Kalman Filter (CKF) [19]. We use the sampling method with the generalized Fibonacci grids [20] for sample efficiency with fewer equally weighted samples to homogeneously cover the joint measurement/state space with low cost. This approach was inspired by the unique spiral packing found in sunflower heads, known as the 2D Fibonacci grid, and its generalization to higher dimensions [21]. A Fibonacci grid in higher dimensions is first generated by a uniformly distributed sample set, and then transformed in order to approximate arbitrary multivariate Gaussian distributions deterministically.

C. Online Implementation Strategy

The employed implementation strategy involves a specialized approach for the online phase, termed measurement-targeted sampling. Upon receiving an actual measurement, a window of defined width is centered on it. Deterministic samples are deliberately drawn within this window to ensure homogeneous and dense placement around this concrete measurement. It allows the IGP models to more effectively leverage local information through the kernel function.

For simplicity, within the scope of this paper, we temporarily assume a scalar bijective nonlinear measurement equation of a scalar hidden state, corrupted by additive Gaussian noise

$$\boldsymbol{y}_k = h_k(\boldsymbol{x}_k) + \boldsymbol{v}_k \ . \tag{7}$$

For multidimensional scenarios, IGPs are also applicable and can be constructed using multi-output GPs. This aspect will be explored in detail in an upcoming paper.

Upon receiving a concrete measurement $\tilde{y}_k$, the inverse measurement equation is used to find the corresponding state value $\tilde{x}_k = h^{-1}(\tilde{y}_k)$. An interval of width $2l$ along the x-axis centered at $\tilde{x}_k$ is then defined, spanning $[\tilde{x}_k - l, \tilde{x}_k + l]$, $l \in \mathbb{R}^+$, to represent the range of values for the Gaussian prior state. This interval is then linearly transformed into $\left[\frac{\tilde{x}_k - l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}, \frac{\tilde{x}_k + l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}\right]$ by converting an arbitrary 1D Gaussian into a standard Gaussian. Subsequent transformation of this interval into a range of the $[0, 1]$-uniform distribution is achieved by using the Cumulative Distribution Function (CDF) of the standard Gaussian $\Phi(\cdot)$. It results in the domain of $\left[\Phi(\frac{\tilde{x}_k - l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}), \Phi(\frac{\tilde{x}_k + l - \hat{x}_k^{\mathrm{p}}}{\sigma_k})\right]$, where the Fibonacci grids is placed. This strategy is summarized in Algorithm 1.

Algorithm 1 Derive a domain for uniform Fibonacci grids.

function BUILDWINDOW($\tilde{y}_k$, h, $\hat{x}_k^{\mathrm{p}}$, σ_k^{p}; a=1.5)
input: $\tilde{y}_k$: actual received measurement value
h: bijective measurement equation
$\hat{x}_k^{\mathrm{p}}$: Gaussian prior mean at k^{th} time step
σ_k^{p}: Gaussian prior standard deviation
a: positive constant proportional coefficient
$\tilde{x}_k \leftarrow h^{-1}(\tilde{y}_k)$, $l \leftarrow a \cdot \sigma_k^{\mathrm{p}}$;
$[\tilde{x}_k - l, \tilde{x}_k + l]$ of arbitrary Gaussian prior ;
$\left[\frac{\tilde{x}_k - l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}, \frac{\tilde{x}_k + l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}\right]$ of std. Gauss. $\leftarrow$ Linear transform;
return $\left[\Phi(\frac{\tilde{x}_k - l - \hat{x}_k^{\mathrm{p}}}{\sigma_k}), \Phi(\frac{\tilde{x}_k + l - \hat{x}_k^{\mathrm{p}}}{\sigma_k})\right] \leftarrow \Phi(\cdot)$ std. Gauss.;
end function

D. 1st Single-Output IGP for Scalar Mean Interpolation

The first single-output IGP model is employed for scalar mean value interpolation, targeting the optimal MMSE estimator $\mathbb{E}_{f_k^{\mathrm{e}}(\boldsymbol{x}_k)}[\boldsymbol{x}_k \mid \underline{\tilde{y}}_{1:k}]$. It involves numerically computing the expected value of the true posterior state density, thus minimizing the mean square error between estimates and true states. However, the training of the 1st IGP focuses on maximizing the log marginal likelihood of the drawn samples to determine the optimal hyperparameters of interest [22]

$$\underline{\theta}_{1\text{st}}^* = \underset{\underline{\theta}_{1\text{st}}}{\operatorname{argmax}} \ \log \ p\Big(\underline{X}_k \ \Big| \ \underline{Y}_k, \ \underline{\theta}_{1\text{st}}\Big) \ , \tag{8}$$

$$= \underset{\underline{\theta}_{1\text{st}}}{\operatorname{argmax}} \ -\underline{X}_k^\top \ \mathbf{K}_x^{-1} \ \underline{X}_k - \log \left|\mathbf{K}_x\right| \ , \tag{9}$$

where $\mathbf{K}_x = \mathbf{K}^\theta + \sigma_\varepsilon^2 \cdot \mathbf{I}$ contains the hyperparameters to be optimized. The N samples are $D = \{(y_k^i, \ x_k^i)\}_{i=1}^{\mathrm{N}}$ with inputs $\underline{Y}_k = [y_k^1, \ldots, y_k^N]^\top$ and outputs $\underline{X}_k = [x_k^1, \ldots, x_k^N]^\top$.

Unlike standard GP approaches, our method above, however, fixes the length scale of the SE kernel during training. The SE kernel is characterized by its stationarity and isotropy, serving as a similarity measure based on the distance between any two samples. For us, the similarity among samples in the local surroundings of the actual received measurement is of particular interest.

This is achieved by first calculating a heuristic length scale in a sample-specific manner. The process involves calculating the average absolute distance between each sample and every other sample along the input axis. The overall average of these values indicates the all-inclusive similarity among samples centered on the concrete measurement. This length scale is then fixed during training, which controls the smoothness of the inference results provided by the posterior IGP. It ensures that a specific level of smoothness or flexibility is maintained without easily underfitting the given samples while searching for other optimal hyperparameters.

Through a gradient-ascent based optimization tool for training in (9), we employ only the predictive mean results of the posterior IGP in (5) with optimal hyperparameters $\underline{\theta}_{1\text{st}}^*$. They serve as the interpolated mean values for subsequent moment matching of conditional Gaussians. The interpolated mean values, derived from the 1st posterior IGP model, are depicted in a rotated Cartesian coordinate system in Figure 2.

E. 2nd Single-Output IGP for Scalar Variance Interpolation

Based on the first posterior IGP model $p(\boldsymbol{x}_k|y_k^*, \underline{\theta}_{1\text{st}}^*)$ trained on the drawn samples $D = \{(y_k^i,\ x_k^i)\}_{i=1}^{\text{N}}$, we infer the predictive mean value for each measurement realization y_k^i in D. Afterwards, squared residuals between these predictions and the corresponding state realizations x_k^i in D are computed and serve as new "observations" or "outputs"

$$\Delta_{k,i}^2 = \left(x_k^i - \mathbb{E}\left[p(\boldsymbol{x}_k \mid y_k^i,\ \underline{\theta}_{1\text{st}}^*)\right]\right)^2 \ , \ i = 1, \ldots, N. \quad (10)$$

Together with the measurement realizations in D as "inputs", they form a new dataset for training the second IGP model $D' = \{(y_k^i,\ \Delta_{k,i}^2)\}_{i=1}^{\text{N}}$. The training procedure follows a similar methodology as previously described, but has a different optimization goal to determine $\underline{\theta}_{2\text{nd}}^*$ for the 2nd IGP:

$$\underline{\theta}_{2\text{nd}}^* = \underset{\underline{\theta}_{2\text{nd}}}{\text{argmax}}\ \log\ p\Big(\underline{\Delta}_k \ \Big|\ \underline{Y}_k, \underline{\theta}_{2\text{nd}}\Big)\ , \quad (11)$$

$$= \underset{\underline{\theta}_{2\text{nd}}}{\text{argmax}}\ -\underline{\Delta}_k^\top\ \mathbf{K}_\delta^{-1}\ \underline{\Delta}_k - \log\left|\mathbf{K}_\delta\right|\ , \quad (12)$$

where $\mathbf{K}_\delta = \mathbf{K}^\theta + \sigma_\varepsilon^2 \cdot \mathbf{I}$ contains the hyperparameters to be optimized, and the training dataset is $D' = \{(y_k^i,\ \Delta_{k,i}^2)\}_{i=1}^{\text{N}}$ with $\underline{Y}_k = [y_k^1, \ldots, y_k^N]^\top$ and $\underline{\Delta}_k = [\Delta_{k,1}^2, \ldots, \Delta_{k,N}^2]^\top$.

To ensure the interpolated variance remains positive for the specific measurement, we check this value after each optimization iteration. If it is nonpositive, we penalize the next optimization iteration by assigning a significantly small scalar, e.g. $-1\text{e}5$, to the objective function in (12), which opposes the goal of maximizing the marginal likelihood. This penalty guides the optimization process to find suitable hyperparameters that ensure positive interpolation results.

Similar to the first IGP, we use only the predictive mean results of the 2nd posterior IGP model, as shown in Figure 3. They serve as the interpolated variance values along the interpolated means provided by the 1st posterior IGP. The final parameterized conditional Gaussians of the hidden state on arbitrary measurements, obtained through moment matching, are illustrated in Figure 4. The detailed steps of our novel measurement update are summarized in Algorithm 2.

The computational complexity in (9) and (12) is mainly driven by the need for matrix inversion. Standard methods for inverting an $n \times n$ positive definite symmetric covariance matrix require $O(n^3)$ time and $O(n^2)$ space. This computational burden restricts the number of drawn samples from the true joint density. Moreover, with insufficient samples, the posterior IGP tends to revert to its predefined prior mean function, leading to suboptimal extrapolations. Therefore, it is essential in real applications to ensure homogeneous placement of limited samples around the concrete measurement. As a result, the use of IGP models is often constrained to local joint density approximations only.

V. Evaluation

We evaluate the performance of the proposed GADF by means of two simulation examples, which were all implemented in `Julia`. We compare our filter to several state-of-the-art GADFs. The GPF draws 10^6 random samples in each prediction and filtering step to determine the mean and covariance of the state. Despite its high computational cost and non-deterministic results, this large sample size ensures asymptotic convergence to the optimal MMSE estimator, making the GPF a benchmark for the estimation quality achievable with a GADF. An LRKF is considered. Unlike the GPF, it uses the same deterministic Gaussian sampling strategy [20] and sample size as our filter to draw from the

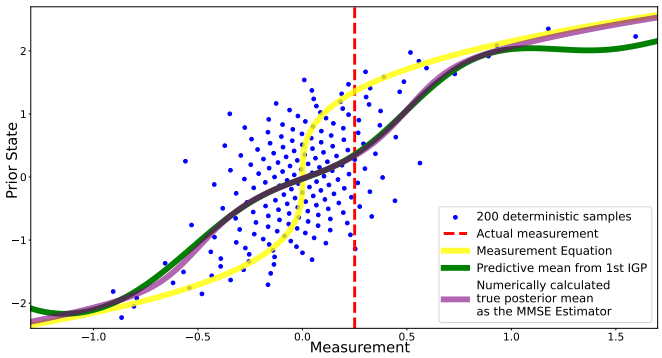

Fig. 2: Predictive mean values (green) of the 1st IGP, based on the deterministic samples (blue) from the true joint density, compared with the MMSE estimator (purple).

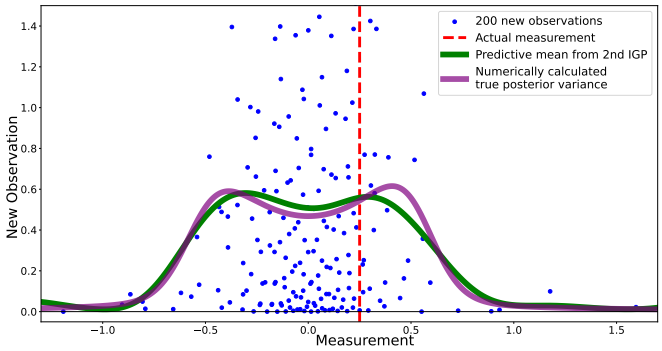

Fig. 3: Predictive mean values (green) of the 2nd IGP, based on the new dataset D' (blue), compared with the numerically calculated variances of the true posterior density (purple).

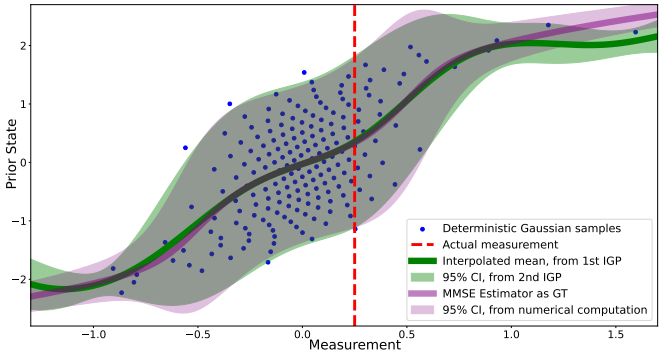

Fig. 4: Conditional Gaussians of the hidden state on arbitrary measurements (horizontal axis) are parameterized by interpolated mean and variance values (green). By introducing individual prefactors, a non-Gaussian approximation of the joint measurement/state density is obtained.

Algorithm 2 Perform measurement-targeted online filtering.

function FILTERING($\tilde{y}_k$, h, $\hat{x}_k^{\mathrm{p}}$,σ_k^{p}, μ_v,σ_v;N=200,a=1.5)
input: μ_v: mean of measurement noise
  σ_v: standard deviation of measurement noise
  N: number of samples drawn
$[u_1, u_2] \leftarrow$ BuildWindow$(\tilde{y}_k, h, \hat{x}_k^{\mathrm{p}}, \sigma_k^{\mathrm{p}}; a = 1.5)$;
$D = \{(y_k^i,\ x_k^i)\}_{i=1}^{\mathrm{N}} \leftarrow$
  Fibonacci$(N, [u_1, u_2], [\hat{x}_k^{\mathrm{p}}, \sigma_k^{\mathrm{p}}], [\mu_v, \sigma_v], h)$;
$\hat{x}_k^{\mathrm{e}} \leftarrow$ pred. mean of 1st posterior IGP $p(\boldsymbol{x}_k \mid \tilde{y}_k,\ \underline{\theta}_{1^{\mathrm{st}}}^*)$;
$\Delta_{k,i}^2 \leftarrow \left(x_k^i - \mathbb{E}\left[\ p(\boldsymbol{x}_k \mid y_k^i,\ \underline{\theta}_{1^{\mathrm{st}}}^*)\ \right]\right)^2$,
  $D' = \{(y_k^i,\ \Delta_{k,i}^2)\}_{i=1}^{\mathrm{N}}$;
$\sigma_k^{\mathrm{e}} \leftarrow$ pred. mean of 2nd post. IGP $p(\boldsymbol{\sigma}_k^2 \mid \tilde{y}_k,\ \underline{\theta}_{2^{\mathrm{nd}}}^*)$;
return $[\hat{x}_k^{\mathrm{e}},\ \sigma_k^{\mathrm{e}}]$ as Gaussian posterior state density ;
end function

joint state/measurement noise density. Moreover, a PGF [15] is also analyzed, as it, like our approach, avoids linearizing the measurement model and outperforms other filters.

A. Cubic Sensor Problem

The discrete-time cubic sensor problem is a popular benchmark to evaluate filter performance effectively and is described by the measurement equation

$$\boldsymbol{y}_k = 0.1 \cdot \boldsymbol{x}_k^3 + \boldsymbol{v}_k\ , \tag{13}$$

where $\boldsymbol{v}_k$ is zero-mean state-independent Gaussian measurement noise. This nonlinear measurement equation is a polynomial function of the state variable, which enables analytic calculations of all required moments. For that reason, a NKF based on analytic statistical linearization, denoted as Analytic Linear Regression Kalman Filter (ALRKF), is derived as the highest-quality variant among all the LRKFs.

The first experiment aims to compare the average filter performance in recursive hidden state estimation scenarios. 50 independent true state sequences $x_k^{(\mathrm{i})}, i = 1, \ldots, 50$ over 500 time steps $k = 1, \ldots, 500$ are generated according to the identity system equation $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{w}_k$, i.e., random walk, corrupted by Gaussian system noise $\boldsymbol{w}_k$ with a standard deviation of 1.0. The initial true state is always set to $x_1 = 0.0$. The corresponding measurement sequences $\tilde{y}_k^{(\mathrm{i})}, k = 1, \ldots, 500, i = 1, \ldots, 50$ are generated according to the measurement equation in (13). The resulting state estimate sequences are given as $\hat{x}_k^{\mathrm{e},(\mathrm{i})}, k = 1, \ldots, 500, i = 1, \ldots, 50$. Thus, the Root Mean Square Error (RMSE) between the state estimates and the true states at each time step is calculated as a measure for filter performance assessment.

Figure 5 demonstrates that the estimation outcomes of the proposed filter closely match the ground truth from the GPF. Notably, our filter exhibits a much smaller RMSE between steps 250 and 300, as 10^6 samples might not be sufficient enough for the GPF to fully converge. Furthermore, our filter significantly outperforms both the LRKF and the ALRKF by a large margin. These results are expected. Because an implicit linearization of the measurement model through a second Gaussian assumption comes at the price of diminished filter performance compared with other GADFs working without such simplification.

The following analysis compares the computational complexity of various filters during the measurement update, as shown in table I, and highlights the differences in average measurement update runtimes in Figure 6. While faster than the GPF, our filter has lengthy runtimes, which arise from two sequential matrix inversions, each with cubic time complexity. Despite this computational burden, it delivers higher-quality Gaussian filtering than the PGF and LRKFs, ensuring more accurate state estimation.

TABLE I: Computational complexity of different filters. n is the number of drawn samples in one measurement update.

Filter	Sample-Based	Time Complex.	Space Complex.
GPF	Yes	$O(n)$	$O(n)$
IGP	Yes	$O(n^3)$	$O(n^2)$
PGF	Yes	$O(n)$	$O(n)$
LRKF	Yes	$O(n)$	$O(n)$
ALRKF	No	$O(1)$	$O(1)$

B. Non-Additive Measurement Noise Example

So far, we considered a nonlinear measurement equation suffering from additive noise and explored recursive state estimation scenarios under the same level of nonlinearity. To demonstrate the proposed filter's enhanced versatility and robustness in handling non-additive measurement noise along with increasingly uncertain prior information, we consider a bilinear measurement equation, as found in [23], [24],

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{v} \cdot \boldsymbol{x}, \tag{14}$$

where $\boldsymbol{v}$ is zero-mean state-independent multiplicative Gaussian noise with a standard deviation of 0.2. The experiment evaluates average filter performance under varying levels of prior uncertainty. To simplify the analysis, we only consider a single filter step and take up the procedures in [14].

The true state is drawn from a uniform distribution over $[-1, 1]$, and the measurement is generated using the measurement equation in (14). The prior estimate $\hat{x}^{\mathrm{p}}$ is formed by adding a zero-mean Gaussian sample with variance σ^{p} to the true state. The experiment is repeated 200 times for each of the 200 different prior density "widths", σ^{p}, ranging from 0.1 to 5. As the uncertainty in prior information increases, the effective nonlinearity of a nonlinear dynamic system intensifies, requiring advanced filters for higher-quality state estimation.

The estimation results are depicted in Figure 7, showing the RMSE at each prior uncertainty level, together with the minimum and maximum errors. Our proposed filter delivers nearly the same results as the reference GPF, showing its potential to maintain high estimation quality even with significant nonlinearities. In contrast, the results from the LRKF are unsatisfactory. The rapid increase in RMSE and maximum errors indicates significant degradation in filter performance with increasing prior uncertainty.

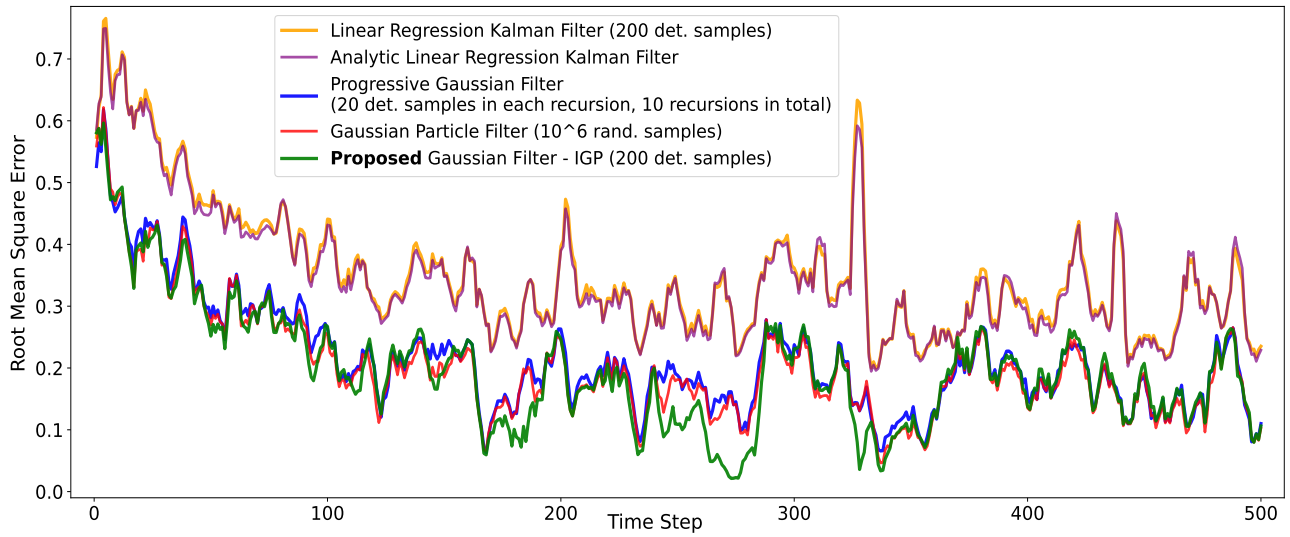

Fig. 5: Comparison of the average estimation quality by filtering 50 independent measurement sequences over 500 time steps. The performance of the proposed filter (green), the GPF (red), the ALRKF (purple), the LRKF (orange), and the PGF (blue) are depicted. All simulation results have been smoothed with a window size of 4 time steps for a better illustration.

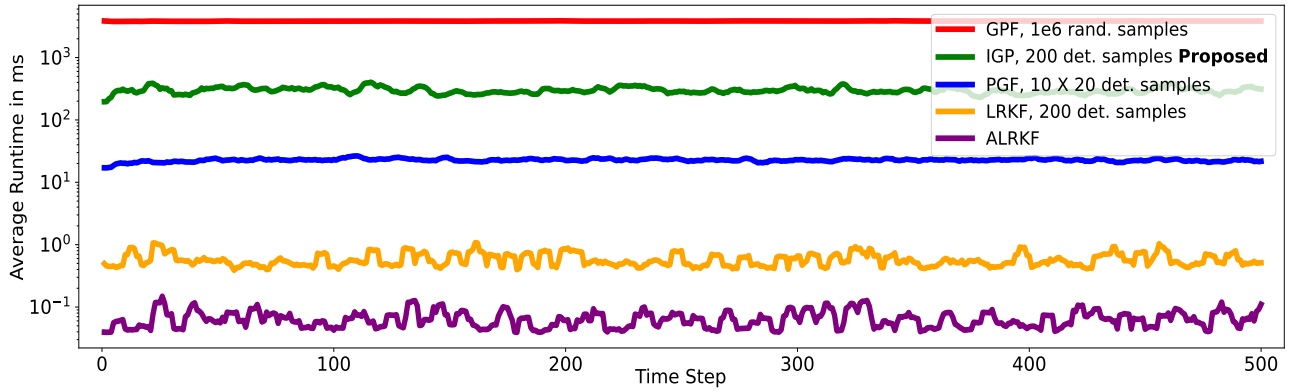

Fig. 6: Average measurement update runtimes in Milliseconds of different filters at each time step.

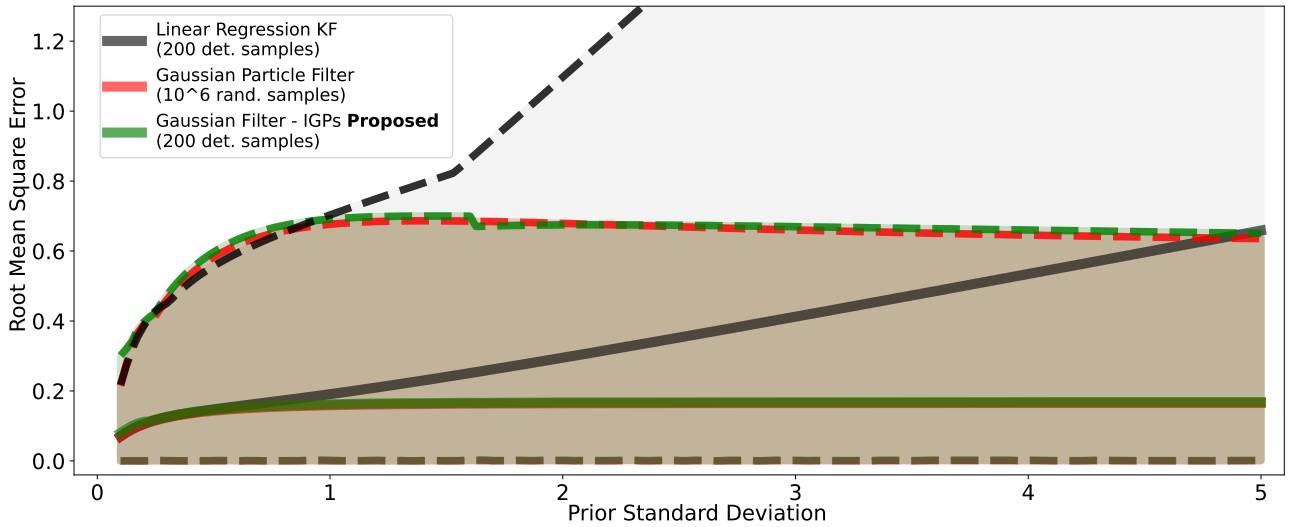

Fig. 7: Estimation quality of three diverse Gaussian filters at different prior uncertainty levels. The RMSE (solid lines) with maximum and minimum errors (dashed lines) from the GPF (red), the proposed filter (green) and the LRKF (grey) are depicted by performing 200 independent single filter steps at 200 different levels of prior uncertainty, respectively.

VI. CONCLUSIONS

In this paper, we introduced a novel GADF for high-quality state estimation of nonlinear dynamic systems. The novelty lies in the measurement update step, which leverages a non-Gaussian local approximation for the true joint density of measurement and prior state. This is obtained through a sequential use of two IGPs, effectively eliminating the need for a second Gaussian assumption. As demonstrated in simulations, the performance of our proposed filter approaches that of the MMSE estimator in recursive state estimation scenarios and its enhanced filter versatility can handle non-additive measurement noise well.

Current research is focused on extending estimation scenarios to multidimensional cases by generalizing single-output IGPs to multi-output IGPs. This involves specifying the cross-correlations between multiple model outputs. Additionally, exploring alternative interpolation techniques presents a promising direction for further investigation.

REFERENCES

[1] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (1960), pp. 35–45. ISSN: 0021-9223. DOI: 10.1115/1.3662552.

[2] Dan Simon. *Optimal State Estimation*. 1st. Hoboken, NJ: Wiley & Sons, 2006.

[3] Jayesh H. Kotecha and Petar M. Djuric. "Gaussian Particle Filtering". In: *IEEE Transactions on Signal Processing* 51.10 (2003), pp. 2592–2601. DOI: 10.1109/TSP.2003.817204.

[4] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA: Artech House Publishers, 2004.

[5] Arnaud Doucet and Adam M. Johansen. "A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later". In: *The Oxford Handbook of Nonlinear Filtering*. Ed. by Dan Crisan and Boris Rozovskii. Oxford: Oxford University Press, 2009, pp. 656–704.

[6] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering*. 2nd. New York: Wiley, 2001.

[7] Harold W. Sorenson. *Recursive Estimation for Nonlinear Dynamic Systems*. New York: Wiley, 1988.

[8] Tom Lefebvre, Herman Bruyninckx, and Joris De Schutter. "The Linear Regression Kalman Filter". In: *Nonlinear Kalman Filtering for Force-Controlled Robot Tasks*. Ed. by Michel Verhaegen and Paul Van Dooren. Vol. 19. Springer, 2005, pp. 205–210.

[9] Tom Lefebvre, Herman Bruyninckx, and Joris De Schutter. "Kalman Filters for Non-Linear Systems: A Comparison of Performance". In: *International Journal of Control* 77.7 (2004), pp. 639–653. DOI: 10.1080/00207170410001713462.

[10] Simon J. Julier and Jeffrey K. Uhlmann. "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE* 92.3 (2004), pp. 401–422. DOI: 10.1109/JPROC.2003.823141.

[11] Marco F. Huber and Uwe D. Hanebeck. "Gaussian Filter Based on Deterministic Sampling for High Quality Nonlinear Estimation". In: *Proceedings of the 17th IFAC World Congress*. Vol. 17. Seoul, Republic of Korea, 2008.

[12] Jannik Steinbring and Uwe D. Hanebeck. "LRKF Revisited: The Smart Sampling Kalman Filter (S2KF)". In: *Journal of Advances in Information Fusion* 9.2 (2014), pp. 106–123.

[13] Jannik Steinbring and Uwe D. Hanebeck. "S2KF: The Smart Sampling Kalman Filter". In: *Proceedings of the 16th International Conference on Information Fusion (Fusion)*. Istanbul, Turkey, 2013, pp. 2089–2096.

[14] Uwe D. Hanebeck. "PGF 42: Progressive Gaussian Filtering with a Twist". In: *Proceedings of the 16th International Conference on Information Fusion (Fusion)*. Istanbul, Turkey, 2013, pp. 1103–1110.

[15] Jannik Steinbring and Uwe D. Hanebeck. "Progressive Gaussian Filtering Using Explicit Likelihoods". In: *Proceedings of the 17th International Conference on Information Fusion (Fusion)*. Salamanca, Spain, 2014.

[16] Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. "Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration". In: *Advances in neural information processing systems* 31 (2018).

[17] GPy. *GPy: A Gaussian process framework in python*. http://github.com/SheffieldML/GPy.

[18] Jamie Fairbrother, Christopher Nemeth, and Maxime Rischard. "GaussianProcesses. jl: A Nonparametric Bayes Package for the Julia Language". In: *Journal of Statistical Software* 102 (2022), pp. 1–36.

[19] Ienkaran Arasaratnam and Simon Haykin. "Cubature Kalman Filters". In: *IEEE Transactions on Automatic Control* 54.6 (2009), pp. 1254–1269. DOI: 10.1109/TAC.2009.2019800.

[20] Daniel Frisch and Uwe D. Hanebeck. "Deterministic Gaussian Sampling with Generalized Fibonacci Grids". In: *Proceedings of the 24th International Conference on Information Fusion (Fusion)*. Sun City, South Africa, 2021, pp. 1–8.

[21] Robert James Purser. *Generalized Fibonacci Grids; A New Class of Structured, Smoothly Adaptive Multi-Dimensional Computational Lattices*. May 2008.

[22] Christopher KI Williams and Carl Edward Rasmussen. Vol. 2. 3. MIT press Cambridge, MA, 2006.

[23] Ondřej Straka, Jindřich Duník, and Vítězslav Elvira. "Importance Gauss-Hermite Gaussian Filter for Models with Non-Additive Non-Gaussian Noises". In: *Proceedings of the 24th International Conference on Information Fusion*. Sun City, South Africa, 2021.

[24] Filip Tronarp, Raphael Hostettler, and Simo Särkkä. "Sigma-Point Filtering for Nonlinear Systems with Non-Additive Heavy-Tailed Noise". In: *Proceedings of the 19th International Conference on Information Fusion (Fusion)*. Heidelberg, Germany, 2016, pp. 1859–1866.