

A NOVEL SQUARE ROOT CUBATURE KALMAN FILTER USING MODEL-ENHANCED GAUSSIAN PROCESS STATE-SPACE MODELS

ZHIKUN HE, GUANGBIN LIU, XIJING ZHAO, ZHICHENG YAO, BO ZHANG

Xi'an Research Institute of High-Tech, Xi'an, Shaanxi 710025, China

ABSTRACT

A novel nonlinear filter called model-enhanced Gaussian process square root cubature Kalman filter (MEGP-SRCKF) is proposed to estimate the state of nonlinear dynamic systems where their state-space models are unknown or insufficiently accurate. The algorithm integrates Gaussian process regression (GPR) into square root cubature Kalman filter (SRCKF). Given the training data, GPR model is used to learn and represent the residual of system after factoring the contributions of the parametric model. The combination of GPR and parametric models enhances the performance of either model alone. It improves the accuracy of the transition and measurement models. The resulting MEGP-SRCKF algorithm has several advantages over standard extended Kalman filter (EKF), nonaugmented unscented Kalman filter (UKF), augmented UKF, and SRCKF. Two cases are used to test these filters and the superiority of the proposed filter is demonstrated, where the MEGP-SRCKF can obtain the better results whether an accurate parametric state-space model is obtained.

Keywords: *Bayesian Filter, Cubature Rule, Gaussian Process Regression, Model-Enhanced, Parametric Model, State-Space model, Nonlinear Dynamic System*

1 INTRODUCTION

Filtering in dynamical systems is frequently used in many areas from inertial guidance of aircrafts and spacecrafts to weather and climate prediction. One of the basic problems of filtering is to process noisy measurements to obtain estimates of all of the state variables. In the case where the transition and measurement models of dynamic systems depend on the past information and when we have the assumptions that both the models in the state-space are linear and that the noise terms are normally distributed, the standard linear recursive Kalman filter algorithm can be derived^[1]. The Kalman filter is a well-known recursive state estimator for linear systems. However, in many cases interesting dynamic systems are not linear by nature, and the traditional Kalman filter cannot be applied in estimating the state of such these systems. In this kind of systems, both the models can be nonlinear or only one of them.

Many nonlinear filtering methods, such as the extended Kalman filter (EKF)^[2,3], the unscented Kalman filter (UKF)^[4], the assumed density filter (ADF)^[5], and the quadrature Kalman filter (QKF)^[6,7], have been proposed for filtering in dynamical systems. Based on the first order Taylor series expansion of the transition and measurement models

about the estimated state trajectory under the Gaussian assumption, the EKF may introduce large error and even diverge when the nonlinearities become severe. As shown in [5], in order to transmit the mean and covariance of state more accurately than the EKF, the UKF computes the carefully selected sigma-points through the nonlinear transform. However, the conventional UKF can be viewed as a second order EKF of special form and the unscented transform does not always preserve the first and second moments through the nonlinear transform^[8]. Based on the Gauss-Hermite quadrature rules, the complexity of the QKF increases exponentially with the state-space dimension, that is, QKF suffers from the curse of dimensionality. More recently, by employing a third-degree spherical-radical cubature rule for those multi-dimensional integral to numerically calculate the mean and covariance of the states, an alternative estimation algorithm called cubature Kalman filter (CKF) has been proposed for general nonlinear systems^[9]. The CKF uses a total of cubature points with the same weight to numerically compute Gaussian weighted integrals involved in the Bayesian filter. Under the additive Gaussian noise assumption, CKF can preserve the first and second moments more accurately than the UKF with the similar computational complexity. Meanwhile, for



numerical stability, the square-root format of CKF (SRCKF) is also developed. However, like the traditional filters, the performance of CKF relies on an accurate parametric transition and measurement models, which include the properly selected process and measurement noise statistics. For many applications, the accurate parametric models are difficult to obtain. Even though such parametric models are very efficient, they often ignore hard to model aspects of dynamic systems so that their predictive capabilities may be limited.

To overcome the limitations of parametric models, we introduce non-parametric, Gaussian process regression (GPR) models to enhance the parametric transition and measurement models for dynamical systems. Meanwhile, GPR models can provide uncertainty estimates for their predictions so that they are seamlessly incorporated into Bayesian filters [10]. In this paper, we integrate GPR models into the SRCKF to enhance the parametric models. The combination of GPR and parametric models, which alleviates some of the problems with either model alone, can obtain higher prediction accuracy. The resulting filtering algorithm is thus referred to as model-enhanced Gaussian process square-root cubature Kalman filter (MEGP-SRCKF).

The rest of this paper is organized as follows. Section 2.1 gives the general filtering framework of the Bayesian filter in the Gaussian domain and Section 2.2 derives the new cubature rule for Gaussian-weighted integrals in detail. GPR method is introduced briefly in Section 3. MEGP-SRCKF algorithm is derived and its implementation is described in Section 4. Section 5 demonstrates the advantages of MEGP-SRCKF over the other traditional filters via an example. Finally, some concluding remarks are given in Section 6.

2 CUBATURE KALMAN FILTER

2.1 Bayesian filter theory in the Gaussian Domain

Consider the general state-space model with additive noise:

$$\text{Transition model: } \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (1)$$

$$\text{Measurement model: } \mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad (2)$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ and $\mathbf{z}_k \in \mathbb{R}^{n_z}$ are the state and measurement of the system at discrete time k respectively; $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ are some known functions; $\mathbf{w}_{k-1} \sim \mathcal{N}(0, \mathbf{Q}_{k-1})$ and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k)$ are the independent process and measurement Gaussian noise with zero means and covariances \mathbf{Q}_{k-1} and \mathbf{R}_k respectively.

The Gaussian distribution has many distinctive mathematical properties and it approximates many physical random phenomena, so it is the most convenient and widely used density function assumption to approximate the Bayesian filter. We call the approximate Bayesian filter as Gaussian filter. Consider the discrete-time state-space model given by Eq. (1) and (2). For this model, due to the fact that Gaussian distribution is completely specified by its mean and covariance, the general filtering framework of the Gaussian filter is given as follows [6,11]:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}), \\ P_{k|k} &= P_{k|k-1} - \mathbf{W}_k P_{\mathbf{z}\mathbf{z},k|k-1} \mathbf{W}_k^T, \\ \mathbf{W}_k &= P_{\mathbf{z}\mathbf{x},k|k-1} P_{\mathbf{z}\mathbf{z},k|k-1}^{-1}, \end{aligned}$$

where

$$\hat{\mathbf{x}}_{k|k-1} = \int_{\mathbb{R}^{n_x}} \mathbf{f}(\mathbf{x}_{k-1}) \times \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}, P_{k-1|k-1}) d\mathbf{x}_{k-1}, \quad (3)$$

$$\begin{aligned} P_{k|k-1} &= \int_{\mathbb{R}^{n_x}} \mathbf{f}(\mathbf{x}_{k-1}) \mathbf{f}^T(\mathbf{x}_{k-1}) \\ &\times \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}, P_{k-1|k-1}) d\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}^T + \mathbf{Q}_{k-1}, \end{aligned} \quad (4)$$

$$\hat{\mathbf{z}}_{k|k-1} = \int_{\mathbb{R}^{n_x}} \mathbf{h}(\mathbf{x}_k) \times \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, P_{k|k-1}) d\mathbf{x}_k, \quad (5)$$

$$\begin{aligned} P_{\mathbf{z}\mathbf{z},k|k-1} &= \int_{\mathbb{R}^{n_x}} \mathbf{h}(\mathbf{x}_k) \mathbf{h}^T(\mathbf{x}_k) \\ &\times \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, P_{k|k-1}) d\mathbf{x}_k - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T + \mathbf{R}_k, \end{aligned} \quad (6)$$

$$\begin{aligned} P_{\mathbf{z}\mathbf{x},k|k-1} &= \int_{\mathbb{R}^{n_x}} \mathbf{x}_k \mathbf{h}^T(\mathbf{x}_k) \times \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, P_{k|k-1}) d\mathbf{x}_k \\ &- \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T. \end{aligned} \quad (7)$$

From Eq. (3) to Eq. (7), it shows that the key of the Gaussian filter is how to compute multi-dimensional Gaussian weighted integrals whose integrands are all of the form “nonlinear \times Gaussian density”.

2.2 Cubature Rule For Gaussian-Weighted Integrals

Consider a multi-dimensional Gaussian weighted integral of the form

$$I(\mathbf{f}) = \int_{\mathbb{R}^n} \mathbf{f}(\mathbf{x}) \exp(-\mathbf{x}^T \mathbf{x}) d\mathbf{x} \quad (8)$$

where $\mathbf{f}(\cdot)$ is some arbitrary function. To compute Eq. (8), it is firstly converted into a spherical-radial integration form by the change of variables as follows: Let $\mathbf{x} = r\mathbf{y}$ with $\mathbf{y}^T \mathbf{y} = 1$, so that $\mathbf{x}^T \mathbf{x} = r^2$ for $r \in [0, \infty)$. Then Eq. (8) can be rewritten as the radial integral

$$\begin{aligned} I(\mathbf{f}) &= \int_0^\infty \int_{U_n} \mathbf{f}(r\mathbf{y}) r^{n-1} \exp(-r^2) d\sigma(\mathbf{y}) dr \\ &= \int_0^\infty S(r) r^{n-1} \exp(-r^2) dr \end{aligned} \quad (9)$$

with the spherical integral



$$S(r) = \int_{U_n} \mathbf{f}(r\mathbf{y})w(\mathbf{y})d\sigma(\mathbf{y}) \quad w(\mathbf{y}) \equiv 1 \quad (10)$$

where $U_n = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y}^T \mathbf{y} = 1\}$ denotes the surface of the unit sphere and $\sigma(\cdot)$ denotes the area element on U_n .

The spherical integral Eq. (9) and the radial integral Eq. (10) are numerically computed by the third-degree spherical cubature rule and the Gaussian quadrature rule, respectively. Hence, we have

$$I(\mathbf{f}) \approx \frac{\sqrt{\pi^n}}{2n} \sum_{i=1}^{2n} \mathbf{f}\left(\sqrt{\frac{n}{2}}[\mathbf{1}]_i\right)$$

where $[\mathbf{u}]$ represents a complete fully symmetric set of points that can be obtained by permutation and changing the sign of the generator \mathbf{u} in all possible ways, $[\mathbf{u}]_i$ denote the i -th point from the set $[\mathbf{u}]$. For example, when $n=3$ and $\mathbf{u}=(1,0,0)^T=(1)$ (zero coordinates are suppressed for brevity), $[\mathbf{u}]=[\mathbf{1}]$ represents the following set of points:

$$[\mathbf{1}] = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \right\}.$$

Further, a general Gaussian weighted integral is approximated as follows:

$$\begin{aligned} & \int_{\mathbb{R}^n} \mathbf{f}(\mathbf{x})\mathcal{N}(\mathbf{x};\boldsymbol{\mu},\boldsymbol{\Sigma})d\mathbf{x} \\ &= \frac{1}{\sqrt{\pi^n}} \int_{\mathbb{R}^n} \mathbf{f}(\sqrt{2} \cdot \sqrt{\boldsymbol{\Sigma}}\mathbf{x} + \boldsymbol{\mu}) \exp(-\mathbf{x}^T \mathbf{x})d\mathbf{x} \\ &\approx \sum_{i=1}^{2n} w_i \mathbf{f}(\sqrt{\boldsymbol{\Sigma}}\boldsymbol{\xi}_i + \boldsymbol{\mu}) \end{aligned}$$

where $\sqrt{\boldsymbol{\Sigma}}$ is the square-root of the covariance $\boldsymbol{\Sigma}$, i.e., $\sqrt{\boldsymbol{\Sigma}}\sqrt{\boldsymbol{\Sigma}}^T = \boldsymbol{\Sigma}$; $\{(\boldsymbol{\xi}_i, w_i), i=1,2,\dots,2n\}$ is the cubature-point set with $\boldsymbol{\xi}_i = \sqrt{n}[\mathbf{1}]_i$ and $w_i = w = 1/2n$.

The CKF algorithm uses the cubature-point set $\{(\boldsymbol{\xi}_i, w_i), i=1,2,\dots,2n\}$ to numerically compute the multi-dimensional Gaussian weighted integrals in Eq. (3)-Eq. (7). For numerical stability, the cubature-point set is also used to build the SRCKF, see [9] for more details.

3 GAUSSIANP ROCESS REGRESSION

A Gaussian process (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution in the statistics areas. It is often thought of as a ‘‘Gaussian distribution over functions’’ [12] in machine learning areas. It can also be thought of as the generalization of a Gaussian

distribution over a finite vector space to a function space of infinite dimension. Similar to a Gaussian distribution, a GP is completely specified by a mean function and a positive definite covariance function. Suppose we have a training data $D = \{(\mathbf{x}_i, y_i) \mid i=1,\dots,N\} = \{X, \mathbf{y}\}$ which is drawn from a noisy process:

$$y_i = f(\mathbf{x}_i) + \varepsilon$$

where $\mathbf{x}_i \in \mathbb{R}^d$ is an input vector of dimension d , $y_i \in \mathbb{R}$ is a corresponding scalar output and ε is an independent, identically distributed Gaussian distribution with zero mean and variance σ_N^2 . For brevity, $X = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_N]$ denotes a $d \times N$ matrix which aggregates the column vector inputs for all N cases and $\mathbf{y} = [y_1 \ y_2 \ \dots \ y_N]$ aggregates the N outputs. The key task of GPR is to generate an output prediction of a function $f(\cdot): \mathbb{R}^d \mapsto \mathbb{R}$ at a new arbitrary test input \mathbf{x}_* .

A GP estimates posterior distribution over functions $f(\cdot)$ from training data D . In terms of the training points, the posterior distribution is represented non-parametrically. A key idea underlying GPs is the requirement that the function values at different points are correlated, that is, the covariance between two function values, $f(\mathbf{x}_i)$ and $f(\mathbf{x}_j)$, depends on the input values, \mathbf{x}_i and \mathbf{x}_j . This dependency is specified via an arbitrary covariance function, or kernel $k(\mathbf{x}_i, \mathbf{x}_j)$. The most widely used kernel function is the squared exponential (SE) kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^T \Lambda^{-1}(\mathbf{x}_i - \mathbf{x}_j)\right)$$

where σ_f^2 is the signal variance, $\Lambda = \text{diag}([l_1^2, l_2^2, \dots, l_d^2])$ and l_i is the length scale which reflect the relative smoothness along the different input dimensions.

Given the training data D and a new test input \mathbf{x}_* , we can obtain a Gaussian predictive distribution over the corresponding output y_*

$$y_* \sim \mathcal{N}(\text{GP}_\mu(\mathbf{x}_*, D), \text{GP}_\Sigma(\mathbf{x}_*, D)) := \text{GP}(\mathbf{x}_*, D)$$

with mean and variance

$$\text{GP}_\mu(\mathbf{x}_*, D) = \mathbf{k}_*^T [K + \sigma_N^2 I_N]^{-1} \mathbf{y} \quad (11)$$

$$\text{GP}_\Sigma(\mathbf{x}_*, D) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T [K + \sigma_N^2 I_N]^{-1} \mathbf{k}_* \quad (12)$$

where $\mathbf{k}_* = K(X, \mathbf{x}_*) = K(\mathbf{x}_*, X)^T$ is the $N \times 1$ covariance matrix between the inputs X and the test point \mathbf{x}_* such that $\mathbf{k}_*(i,1) = k(\mathbf{x}_*, \mathbf{x}_i)$, $i=1,2,\dots,N$, $k(\mathbf{x}_*, \mathbf{x}_*)$ the variance of the test point \mathbf{x}_* , and I the $n \times n$ identity matrix.

The parameters of the kernel function and the output noise are aggregated to $\boldsymbol{\theta} = \{\Lambda, \sigma_f^2, \sigma_N^2\}$, called the hyperparameters. They are typically

determined by maximizing the log likelihood of the training outputs, that is

$$\hat{\theta} = \arg \max_{\theta} \log p(y | \mathbf{X}, \theta).$$

After determining the optimal hyperparameters, we can compute the predicted output $GP_{\mu}(\mathbf{x}_*, D)$ and its variance $GP_{\Sigma}(\mathbf{x}_*, D)$ corresponding to the test point \mathbf{x}_* using Eq. (11) and Eq. (12).

Most GP models are defined for modeling only a single output variable. An independent model for each output dimension is used to represent the vectorial outputs generally involve in the transition and measurement models. The resulting process and measurement noise covariances are diagonal matrices, since the output dimensions are now independent of each other.

4 MEGP-SRCKF

Since purely non-parametric GPR model is data-driven, it lacks interpretability for the system. The another drawback of the model is that when the new test point is far away from the training data, the corresponding output will quickly tend toward zero, as shown in Figure 1. It results high requirement for the training data. Parametric model is use to attempt to represent a particular phenomenon. However, to build these models, substantial domain expertise is required. It results that these accurate parametric models are difficult to obtain, which are always very complicated and are often simplified to represent the actual systems.

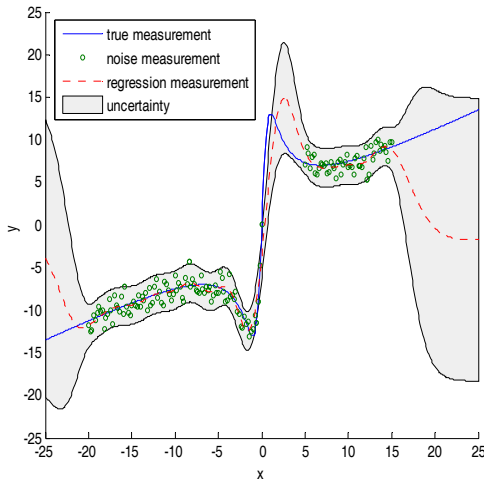


Figure 1: Gaussian Process Regression For A One-Dimensional Linear Regression Problem, $y_t = x_t/2 + 25x_t/(1 + x_t^2) + w_t$, With $w_t \sim N(0,1)$ I.I.D. Noise.

The Blue Line Denotes The Noise-Free Measurements. The Red Dashed Denotes The

Output Of Trained GPR Model Using The Training Data (Green Circle), Which Are Collected In Both Areas [-20, 0] And [5, 15]. The Gray Region Denotes The 95% Confidence Region For Regression Measurements. Note That Both The Error Of Regression Measurement And Its Uncertainty (The Vertical Width Of The Gray Region) Become Larger Beyond The Training Areas.

The combined model of GPR and parametric models can alleviate some of the problems with either model alone. Since the GPR model is used to represent the aspect of the dynamic system that is still not modeled by the parametric model, the accuracy of the combined model for the actual system is improved. Meanwhile, due to the parametric model can capture most of the physical process underlying the dynamic system, the generality of the combined model is improved beyond the training data. The combined model is called Model-enhanced GP (MEGP) model.

The GPR-model branch of MEGP model is used to represent the residual output after factoring the contribution of the parametric model. If $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1})$ and $\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k)$ are the parametric state-space models for the dynamic system, then the MEGP model can be written as

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}) + GP_{\mu}^f(\mathbf{x}_{k-1}, D^f) \\ \mathbf{z}_k &= \mathbf{h}(\mathbf{x}_k) + GP_{\mu}^h(\mathbf{x}_k, D^h) \end{aligned}$$

where $GP^f(\mathbf{x}_{k-1}, D^f)$ and $GP^h(\mathbf{x}_k, D^h)$ are the GPR-model branches of MEGP models for the transition and measurement models, respectively. The training outputs for the two MEGP models are

$$\begin{aligned} \Delta \mathbf{x}_k &= \mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}) \\ \Delta \mathbf{z}_k &= \mathbf{z}_k - \mathbf{h}(\mathbf{x}_{k-1}). \end{aligned}$$

Their corresponding training data are

$$\begin{aligned} D^f &= (\mathbf{X}_0, \Delta \mathbf{X}), \\ D^h &= (\mathbf{X}, \Delta \mathbf{Z}) \end{aligned}$$

where $\mathbf{X}_0 = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{N-1}]$, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$, $\Delta \mathbf{X} = [\Delta \mathbf{x}_1, \Delta \mathbf{x}_2, \dots, \Delta \mathbf{x}_N]$, $\Delta \mathbf{Z} = [\Delta \mathbf{z}_1, \Delta \mathbf{z}_2, \dots, \Delta \mathbf{z}_N]$.

Now the MEGP models can be incorporated into the SRCKF algorithm. Details of the resulting filtering algorithm are presented in Appendix A.

5 APPLICATIONS

To illustrate some of the advantages of MEGP-SRCKF over other filters, a model called Univariate Nonstationary Growth Model (UNGM)

is considered in this paper. This model is highly nonlinear and bimodal that is really challenging for traditional filtering techniques. The dynamic state space model for UNGM can be written as

$$x_k = \frac{1}{2}x_{k-1} + 25\frac{x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2(k-1)) + w_{k-1}, \quad (13)$$

$$y_k = \frac{1}{20}x_k^2 + v_k, \quad k = 1, 2, \dots, K, \quad (14)$$

where $w_{k-1} \sim \mathcal{N}(0, \sigma_w^2)$ and $v_k \sim \mathcal{N}(0, \sigma_v^2)$. The cosine term in the state transition model represents the effect of time-varying noise.

First, we assume that the accurate parametric model is obtained as Eq. (13) and Eq. (14). EKF, nonaugmented UKF, augmented UKF [13], and SRCKF are also tested in this case. In figure 2 we have plotted the mean square errors (MSEs) of each tested methods of 50 Monte Carlo runs, of which the mean and standard deviation is present in figure 3.

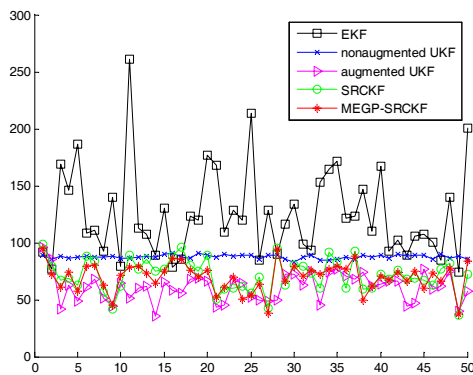


Figure 2: MSEs of different methods in 50 Monte Carlo runs

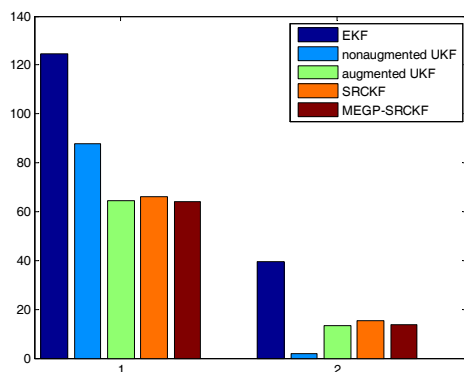


Figure 3: The Means (Left) And Standard Deviations (Right) Of Mses Of Different Methods In 50 Monte Carlo Runs

Since the bimodality of UNGM in general can not be applied well using Gaussian approximations, EKF don't work well in this case. The

MEGP-SRCKF can obtain similar results to the augmented UKF and the SRCKF. They give clearly better performance from the nonaugmented UKF. In the case that we can obtain appropriate parametric models, the MEGP-SRCKF can obtain similar performance to its corresponding parametric filter.

We further assume that an insufficiently accurate state-space model is obtained as follows:

$$x_k = \frac{1}{3}x_{k-1} + 30\frac{x_{k-1}}{1+x_{k-1}^2} + w_{k-1},$$

$$y_k = \frac{1}{10}x_k^2 + v_k,$$

Before filtering, two groups data are assumed to obtain for training the GPR-model branch of MEGP models for the transition and measurement models, respectively. We also test the above filters in this case. Similarly, in Figure 4 we have plotted the MSEs of each tested methods of 50 Monte Carlo runs, of which the mean and standard deviation is present in Figure 5.

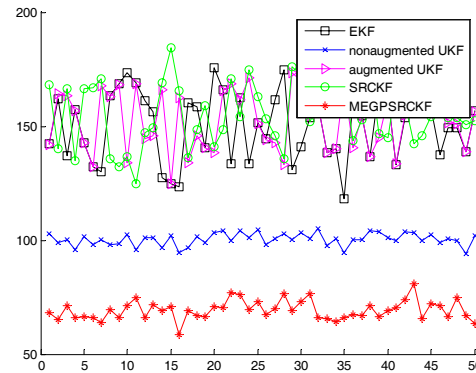


Figure 4: Mses Of Different Methods In 50 Monte Carlo Runs

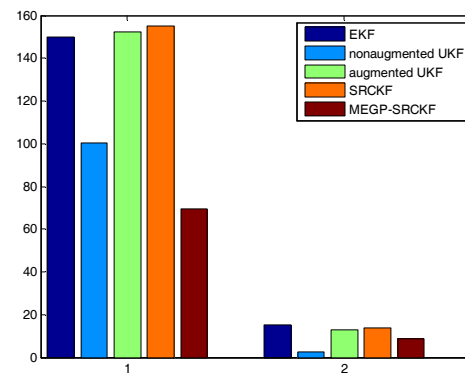


Figure 5: The Means (Left) And Standard Deviations (Right) Of Mses Of Different Methods In 50 Monte Carlo Runs

As shown in Figure 4 and 5, we can see that the MEGP-SRCKF is effective to estimate the state in this case, whereas standard EKF, nonaugmented UKF, augmented UKF, and SRCKF are disabled. Compare the results of the above four figures, it also shows that MEGP-SRCKF can obtain the similar results whether the accurate parametric state-space model can be obtained. Therefore, the MEGP-SRCKF does not require an accurate, parametric model of the nonlinear dynamic system.

6 CONCLUSIONS

An alternative estimation algorithm called CKF has been proposed recently for nonlinear dynamic systems. It can preserve the first and second moments accurately with low computation. However, its performance relies on the accurate state-space models as same as the traditional filters, for instance EKF and UKF. In this paper, a novel filter algorithm called MEGP-SRCKF is presented to estimate the state of nonlinear dynamic systems where their transition or measurement or both models are unknown or insufficiently accurate. It integrated a combination model of GPR and parametric models into the SRCKF, where the state-space model is replaced by the combination model. The GPR-model branch of the combination model is used to learn and represent the residual aspects of the system that are not captured by the parametric model. The novel algorithm alleviates the disadvantages associated with the traditional filters, and its performance is tested via two examples.

Appendix A: Table 1 Megp-Srckf Algorithm

Before Filtering

1) Build the training data for the transition MEGP model and train this model:

$$\Delta \mathbf{x}_k \sim \text{GP}^f(\mathbf{x}_{k-1}, D^f).$$

2) Build the training data for the measurement MEGP model and train this model:

$$\Delta \mathbf{z}_k \sim \text{GP}^h(\mathbf{x}_k, D^h).$$

3) Evaluate the initial cubature points ($i = 1, 2, \dots, m, m = 2n_x$)

$$\xi_i = \sqrt{n_x} [1]_i.$$

Initialization

1) Initialization of the state $\mathbf{x}_{0|0}$ and the square-root factor of the predicted error covariance $S_{0|0}$ such that $P_{0|0} = S_{0|0} S_{0|0}^T$.

Time Update

1) Evaluate the cubature points of the current state ($i = 1, 2, \dots, m$)

$$X_{i,k-1|k-1} = S_{k-1|k-1} \xi_i + \hat{\mathbf{x}}_{k-1|k-1}.$$

2) Evaluate the cubature points of the predicted state ($i = 1, 2, \dots, m$)

$$X_{i,k|k-1}^* = \mathbf{f}(X_{i,k-1|k-1}) + \text{GP}_\mu^f(X_{i,k-1|k-1}, D^f).$$

3) Estimate the predicted state

$$\hat{\mathbf{x}}_{k|k-1} = \frac{1}{m} \sum_{i=1}^m X_{i,k|k-1}^*.$$

4) Estimate the square-root factor of the predicted error covariance

$$S_{k|k-1} = \text{Tria} \left(\begin{bmatrix} X_{k|k-1}^* & S_{Q,k-1} \end{bmatrix} \right)$$

where the weighted, centered matrix

$$X_{k-1|k-1}^* = \frac{1}{\sqrt{m}} \begin{bmatrix} X_{1,k-1|k-1}^* - \hat{\mathbf{x}}_{k-1|k-1} & X_{2,k-1|k-1}^* - \hat{\mathbf{x}}_{k-1|k-1} \\ \dots & X_{m,k-1|k-1}^* - \hat{\mathbf{x}}_{k-1|k-1} \end{bmatrix}$$

and $S_{Q,k-1}$ denotes a square-root factor of

$$Q_{\Sigma} = \text{GP}_\Sigma^f(\hat{\mathbf{x}}_{k-1|k-1}, D^f) \quad \text{such that}$$

$$Q_{k-1} = S_{Q,k-1} S_{Q,k-1}^T.$$

Measurement Update

1) Evaluate the cubature points of the update state ($i = 1, 2, \dots, m$)

$$X_{i,k|k-1} = S_{k|k-1} \xi_i + \hat{\mathbf{x}}_{k|k-1}.$$

2) Evaluate the cubature points of the predicted measurement ($i = 1, 2, \dots, m$)

$$Z_{i,k|k-1} = \mathbf{h}(X_{i,k|k-1}) + \text{GP}_\mu^h(X_{i,k|k-1}, D^h).$$

3) Estimate the predicted measurement

$$\hat{\mathbf{z}}_{k|k-1} = \frac{1}{m} \sum_{i=1}^m Z_{i,k|k-1}.$$

4) Estimate the square-root factor of the innovation covariance matrix

$$S_{\mathbf{z}\mathbf{z},k|k-1} = \text{Tria} \left(\begin{bmatrix} Z_{k|k-1} & S_{R,k} \end{bmatrix} \right)$$

where the weighted, centered matrix

$$Z_{k|k-1} = \frac{1}{\sqrt{m}} \begin{bmatrix} Z_{1,k|k-1} - \hat{\mathbf{z}}_{k|k-1} & Z_{2,k|k-1} - \hat{\mathbf{z}}_{k|k-1} \\ \dots & Z_{m,k|k-1} - \hat{\mathbf{z}}_{k|k-1} \end{bmatrix}$$

and $S_{R,k}$ denotes a square-root factor of

$$R_k = \text{GP}_\Sigma^h(\hat{\mathbf{x}}_{k|k-1}, D^h) \quad \text{such that} \quad R_k = S_{R,k} S_{R,k}^T.$$

5) Estimate the cross-covariance matrix

$$P_{\mathbf{z}\mathbf{x},k|k-1} = X_{k|k-1} Z_{k|k-1}^T$$

where the weighted, centered matrix

$$X_{k|k-1} = \frac{1}{\sqrt{m}} \begin{bmatrix} X_{1,k|k-1} - \hat{\mathbf{x}}_{k|k-1} & X_{2,k|k-1} - \hat{\mathbf{x}}_{k|k-1} \\ \dots & X_{m,k|k-1} - \hat{\mathbf{x}}_{k|k-1} \end{bmatrix}$$

6) Estimate the Kalman gain of CKF



$$W_k = (P_{xz,k|k-1} / S_{zz,k|k-1}^T) / S_{zz,k|k-1}$$

7) On the receipt of a new measurement \mathbf{z}_k , estimate the updated state

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + W_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1})$$

8) Estimate the square-root factor of the update error covariance matrix

$$S_{k|k} = \mathbf{Tria} \left(\begin{bmatrix} X_{k|k-1} & & \\ & Z_{k|k-1} & \\ & & W_k S_{R,k} \end{bmatrix} \right)$$

[13] Y. Wu, D. Hu, M. Wu, and X. Hu, "Unscented Kalman Filtering for Additive Noise Case: Augmented versus Nonaugmented", IEEE Signal Processing Letters, Vol. 12, No. 5, 2005, pp. 357-360.

REFERENCES:

[1] R. E. Kalman, R. S. Bucy, "New results in linear filtering and prediction theory", Trans. ASME Ser. D J. Basic Eng., Vol. 83, 1961, pp. 95-108.

[2] A. H. Jazwinski, "Stochastic Processes and Filtering Theory", New York: Academic Press, 1970.

[3] P. S. Maybeck, "Stochastic Models, Estimation, and Control", vol. 141 of Mathematics in Science and Engineering, New York: Academic Press, 1979.

[4] S. J. Julier, J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems", Proceedings of AeroSense: 11th Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, FL, USA, 1997, pp. 182-193.

[5] M. Opper, "A Bayesian approach to online learning", In Online Learning in Neural Networks, Cambridge University Press, 1998, pp. 363-378.

[6] K. Ito, K. Xiong, "Gaussian filters for nonlinear filtering problems," IEEE Trans. Automatic Control, Vol. 45, No. 5, 2000, pp. 910-927.

[7] I. Arasaratnam, S. Haykin, R. J. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature", Proc. IEEE, Vol. 95, No. 5, 2007, pp. 953-977.

[8] F. Gustafsson, G. Hendeby, "Some Relations Between Extended and Unscented Kalman Filters," IEEE Trans. Signal Processing, Vol. 60, No. 2, 2012, pp. 545-555.

[9] I. Arasaratnam, S. Haykin, "Cubature kalman filters", IEEE Trans. on Automatic Control, Vol. 54, No. 6, 2009, pp. 1254-1269.

[10] J. Ko, D. Fox, "GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models", Autonomous Robots, Vol. 27, No. 1, 2009, pp. 75-90.

[11] F. L. Lewis, "Optimal Estimation", New York: Wiley, 1986.

[12] C. E. Rasmussen, C. K. I. Williams, "Gaussian Processes for Machine Learning", MIT Press, 2006.