Square Root Receding Horizon Information Filters for Nonlinear Dynamic System Models

Du Yong Kim, Member, IEEE, Moongu Jeon, Member, IEEE

Abstract—New nonlinear filtering algorithms are designed based on a receding horizon strategy, i.e., a finite impulse response (FIR) structure, and square root information filtering to achieve high accuracy and good performance in empirical error covariance tests. The new nonlinear receding horizon filters reduce approximation errors in nonlinear filtering by considering a set of recent observations with non-informative initial conditions. By applying information filtering, we are able to manage the non-informative initial conditions, and thus propose the square root version of the algorithm as a means of retaining the positive definiteness of the error covariance. Based on the proposed strategy, we then implement known nonlinear filtering frameworks. Simulation results confirm that the new nonlinear receding horizon filters outperform existing nonlinear filters in well-known nonlinear examples.

Index Terms—receding horizon estimation, unscented Kalman filtering, square root filtering.

I. INTRODUCTION

ONLINEAR state estimation is of great importance in the estimation field. For example, in Kalman filtering techniques, the extended Kalman filter (EKF) and unscented Kalman filter (UKF) have been used for nonlinear dynamic models underlying a single Gaussian assumption [1]. However, though EKF is simple to implement it is not applicable if the model is non-differentiable. Furthermore, it is known that EKF only can handle mild nonlinearity; therefore, UKF has been proposed as an alternative for more severe nonlinear problems such as bearings-only target tracking [1]. In contrast to EKF-based algorithms, UKF does not require model linearization; instead, it performs a direct approximation of the state propagation using a set of deterministically selected points. More recently, a quadrature rule, e.g., cubature rule, has been introduced in order to derive a new single Gaussian filter called the cubature Kalman filter (CKF) [20]. Because of the cubature rule, no parameter tuning during point selection is needed for CKF implementation. However, nonlinear Kalman filters have an inherent performance limitation if the true posterior density is far from a single Gaussian density. Hence, a particle filter (PF) that approximates the probability density function (pdf) of the state with randomly chosen particles has been proposed so as to infer more general types of nonlinear dynamic systems/non-Gaussian type noise [1], [5]. Variants of this type of particle filtering have been studied rigorously. Interested readers are referred to [1] for details.

When the aforementioned stochastic filters are used in practice, however, filtering failures are sometimes inevitable [2]. A filtering failure usually originates from unexpected modeling uncertainty, accumulation of approximation error, round-off errors, or some combination thereof. In the literature a type of finite impulse response (FIR) filter, referred to as a receding horizon estimation (RHE), has been proposed and investigated to circumvent these issues.

RHE was initially studied as an alternative for nonlinear models having unexpected modeling uncertainty where the EKF fails [2]. This idea was revisited by Kwon et al. [3] for linear dynamic systems, who subsequently implemented the FIR filter underlying Kalman filtering techniques. Recent works in linear FIR filters can also be found in [18] and [19], which consider the optimal solution and unknown initial conditions, respectively.

In linear systems, RHE via Kalman filtering is suboptimal in situations in which there is no uncertainty in the given model. However, this is not the case for nonlinear systems because optimal filter implementation is impossible; studies about nonlinear RHE can be found in the literature but are quite limited in scope and number. As a brief overview, RHE has been formulated under constrained optimization methods for nonlinear filtering problems [9], and UKF-based RHE was applied to INS/GPS integration [10]. However, these works suffer from the use of time-consuming batch quadratic programming [9] and their accuracy is dependent upon heuristically determined initial conditions for receding horizon windows [10].

In this paper, we consider “unmodeled uncertainty” in the system as an approximation error for a nonlinear dynamic system or non-Gaussian distribution, and then propose nonlinear receding horizon filters (RHF)s based on the information form of UKF [15]. For this task, we first derive the square root version of the algorithm by pursuing the positive definiteness of a covariance matrix, and then extend the proposed algorithm to PF [5] and Gaussian mixture (GM) frameworks. Another contribution of this paper is its extensive

Manuscript received x x, 2011. This research is supported in part by the Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2012, and the Systems biology infrastructure establishment grant provided by Gwangju Institute of Science and Technology in 2012.

Du Yong Kim and Moongu Jeon are with the School of Information and Communications, Gwangju Institute of Science and Technology, Gwangju, South Korea, (e-mail: duyongkim@gmail.com, mgjeon@gist.ac.kr).
comparison with known nonlinear filters from literature through well-known benchmarking examples with severe nonlinearity. Compared to other state-of-the-art nonlinear filters, the proposed RHF's are superior for the particular problems that we tested in this paper. Indeed, a preliminary result of our work has been implemented in a visual tracking application [14], which verifies the practical usefulness of the algorithm.

The remainder of this paper is organized as follows. Section II addresses the basic formulation of the nonlinear dynamic system estimation. Section III introduces the preliminaries used in the proposed algorithm. Then, a square root nonlinear RHF based on the unscanted information filtering (UIF) with a non-informative initial condition is proposed in Section IV. The derived algorithm is then extended to hybrid PF and GM structures. Verifications of the algorithms are subsequently provided through well-known nonlinear filtering problems in Section V. Finally, we present the conclusion of this study in Section VI.

II. STATE ESTIMATION PROBLEM

Consider a nonlinear dynamic system state-space model as follows:

\[ x_t = f(x_{t-1}) + u_t , \quad z_t = h(x_t) + v_t , \]  

where \( x \in \mathbb{R}^n \) is the state vector, \( z \in \mathbb{R}^m \) is its observation, and \( f(x) \) and \( h(x) \) are nonlinear functions of the state vector. We assume that the pdf for the initial state vector is \( p(x_0) \). Gaussian process and observation noises, \( u_t \sim N(0;Q) \) and \( v_t \sim N(0;R) \), respectively, are assumed to be zero-mean with a corresponding covariance \( (Q, R) \) and are uncorrelated. Given this model, the main goal is to determine the best estimate of the true state. In Bayesian filtering, the formulation of this problem is solved using a two-step recursion (2)-(3), for a conditional pdf of the state given observation history, i.e.,

\[ p(x_{t-1}|z_{t-1}) = \frac{p(z_t|x_{t-1}) p(x_{t-1}|z_{t-1})}{p(z_t)}, \]

where the transition distribution \( p(x_t|x_{t-1}) \) and observation likelihood \( p(z_t|x_{t-1}) \) can be obtained from the known model in (1). However, the optimal implementation is intractable, especially for nonlinear/non-Gaussian models. Therefore, many approximation algorithms have been suggested for the implementation of this conceptual solution.

III. PRELIMINARIES

To make this paper self-contained and introduce notations for use, we will first briefly summarize the point-based nonlinear Kalman filters (UKF and CKF) [4], [20] and its information form. A goal of Kalman filtering is to obtain the optimal estimate under minimum mean square error criterion, if the given dynamic system is linear and the noise is Gaussian. Hence, we can obtain the solution as the conditional mean estimate, i.e., \( \hat{x}_k = E(x_k|z_{1:k}) \), where \( E(\cdot) \) is the conditional expectation operator.

A. Point selection and filter equations

UKF and CKF approximate the Gaussian density by using a set of deterministically chosen points, called sigma-points and cubature-points, respectively. Typically, these points are selected for an augmented state vector \( x_k' \in \mathbb{R}^{n+k} \) along with a Gaussian system noise vector \( u_k \) as

\[ \tilde{x}_k = [\tilde{x}_k^T ~ \tilde{u}_k^T]^T, \quad P_k = diag(\tilde{P}_k), \]  

where \( \tilde{P}_k \) is the error covariance of the filter and \( diag(\cdot) \) is the diagonal block matrix. Note that the superscript ‘a’ is only used in the point selection procedure, as in (5), (7), (15), and (19).

In UKF, we select \( 2n+1 \) sigma-points of the augmented state (4) \( \{x'_i\} = \{(x'_i)^T \}_{i=0}^{2n+1} \), \( i = 0, \ldots, 2n \), \( n_1 = n + n_c \) as

\[ x'_i = \tilde{x}_k + \tilde{\xi}_i^a \sqrt{(n_1 + \lambda)P_k}, \quad i = 1, \ldots, n_1 \]

\[ x'_i = \tilde{x}_k - \tilde{\xi}_i^a \sqrt{(n_1 + \lambda)P_k}, \quad i = n_1 + 1, \ldots, 2n, \]

with corresponding weights

\[ W_{i}^{a} = \lambda / (n_1 + \lambda), \quad W_{i}^{o} = 1/2(n_1 + \lambda), \]

where \( \lambda \) denotes the number of points for weights for the mean and covariance, \( \lambda \geq \alpha^2(n_1 + \kappa) - n_c \) is a scale parameter for sigma-points with tuning parameter \( 0 \leq \alpha \leq 1 \), and \( \kappa \) is usually chosen to be \( 3 - n_c; \beta \) is a parameter used to relate higher order terms.

In contrast, cubature-points are selected using the third-degree radial cubature rule [20], such that

\[ \hat{x}_k = \hat{x}_k + \sqrt{n} \hat{P}_k e_i, \quad i = 1, \ldots, n_c, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

where \( e_i \) are points from the intersection between the Cartesian axes and the \( n_c \)-dimensional unit hypersphere; corresponding weights are simply, 1/2, \( i = 1, \ldots, 2n_c \). Compared to the sigma-points, there is no scaling of weights using tuning parameters and no center point. As such, the cubature-point selection can initially be thought to be a simplification of the sigma-point selection; however, the authors in [20] pointed out that it has a mathematical background in cubature rules in numerical integrations. In what follows, we provide UKF equations, though it should be noted that the CKF has the same structure, with the points replaced with \( \tilde{x}_k^a \).

The UKF prediction step (8) and update step (9) are given using sigma-point representation (5)-(6) as follows:

\[ x_{k+1} = f(x_k) + \tilde{x}_k' + \tilde{x}_{k+1} - \tilde{x}_k, \]

\[ P_{k+1} = \sum_{i=0}^{2n} W_{i}^{a} [x'_i - \tilde{x}_k'] [x'_i - \tilde{x}_k']^T, \]

\[ p_{k+1} = \sum_{i=0}^{2n} W_{i}^{o} [x'_i - \tilde{x}_k']^T [x'_i - \tilde{x}_k'] - P_{k+1} \]

where \( P_{k+1} \) is the error covariance of the filter and \( diag(\cdot) \) is the diagonal block matrix. Note that the superscript ‘a’ is only used in the point selection procedure, as in (5), (7), (15), and (19).

In UKF, we select \( 2n+1 \) sigma-points of the augmented state (4) \( \{x'_i\} = \{(x'_i)^T \}_{i=0}^{2n+1} \), \( i = 0, \ldots, 2n \), \( n_1 = n + n_c \) as

\[ x'_i = \tilde{x}_k + \tilde{\xi}_i^a \sqrt{(n_1 + \lambda)P_k}, \quad i = 1, \ldots, n_1 \]

\[ x'_i = \tilde{x}_k - \tilde{\xi}_i^a \sqrt{(n_1 + \lambda)P_k}, \quad i = n_1 + 1, \ldots, 2n, \]

with corresponding weights

\[ W_{i}^{a} = \lambda / (n_1 + \lambda), \quad W_{i}^{o} = 1/2(n_1 + \lambda), \]

where \( \lambda \) denotes the number of points for weights for the mean and covariance, \( \lambda \geq \alpha^2(n_1 + \kappa) - n_c \) is a scale parameter for sigma-points with tuning parameter \( 0 \leq \alpha \leq 1 \), and \( \kappa \) is usually chosen to be \( 3 - n_c; \beta \) is a parameter used to relate higher order terms.

In contrast, cubature-points are selected using the third-degree radial cubature rule [20], such that

\[ \hat{x}_k = \hat{x}_k + \sqrt{n} \hat{P}_k e_i, \quad i = 1, \ldots, n_c, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

where \( e_i \) are points from the intersection between the Cartesian axes and the \( n_c \)-dimensional unit hypersphere; corresponding weights are simply, 1/2, \( i = 1, \ldots, 2n_c \). Compared to the sigma-points, there is no scaling of weights using tuning parameters and no center point. As such, the cubature-point selection can initially be thought to be a simplification of the sigma-point selection; however, the authors in [20] pointed out that it has a mathematical background in cubature rules in numerical integrations. In what follows, we provide UKF equations, though it should be noted that the CKF has the same structure, with the points replaced with \( \tilde{x}_k^a \).

The UKF prediction step (8) and update step (9) are given using sigma-point representation (5)-(6) as follows:

\[ x_{k+1} = f(x_k) + \tilde{x}_k' + \tilde{x}_{k+1} - \tilde{x}_k, \]

\[ P_{k+1} = \sum_{i=0}^{2n} W_{i}^{a} [x'_i - \tilde{x}_k'] [x'_i - \tilde{x}_k']^T, \]

\[ \psi_{k+1} = \lambda / (n_1 + \lambda), \quad \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]

\[ \hat{x}_k = \hat{x}_k - \sqrt{n} \hat{P}_k e_i, \quad i = n_c + 1, \ldots, 2n, \]
B. UIF

Information filtering is a mathematically equivalent to standard Kalman filtering. Because of the additive update form and natural extension for multisensory observation using parallelization, it is widely used in the literature for decentralized data fusion [16], [17]. In this paper, we used information filtering to handle the unknown initial conditions of the RHF design, which is detailed in Section IV.

Instead of implementing recursions for the conditional mean vector \( \tilde{x}_k \) and its covariance \( P_k \), information filtering is composed of recursions for the information state \( \tilde{y}_k \) and information matrix \( Y_k \) defined as \( \tilde{y}_k \equiv P_k^{-1} \tilde{x}_k \), \( Y_k \equiv P_k^{-1} \).

If the given nonlinear state space model is linearized as EKF, i.e., \( f(x) \equiv Fx \) and \( h(x) \equiv Hx \), the corresponding recursions for prediction (10) and update steps (11) are as follows:

\[
\tilde{y}_{k+1} = Y_k \tilde{x}_k, \quad Y_{k+1} = \left[ F_k P_k^{-1} F^T_k + Q_k \right]^{-1},
\]

(10)
\[
\tilde{y}_{k+1} = \tilde{y}_k + H_k R^k \left( \tilde{z}_k - H_k \tilde{x}_k \right), \quad Y_{k+1} = Y_k + H_k R^k H^T_k, \quad \text{(11)}
\]

where \( \tilde{z}_k = z_k - h(\tilde{x}_k) \) is the residual vector.

Note that the linearized state-space model can be obtained by calculating Jacobians and can be used for EKF implementation. However, the EKF may be numerically unstable if the given linearization is not accurate. On the other hand, the implementation of (10)-(11) under the UKF framework is not straightforward because the prediction step in UKF is not explicitly represented as linear operations.

Recently, Lee [15] developed an unscented extension of information filtering, UIF, based on a statistical linear error propagation methodology. From this error propagation method, the pseudo-measurement matrix \( \mathbf{H} \) and its statistical error \( \tilde{v} \) can be given as

\[
\mathbf{H}^t = P_k^{-1} p_k^t, \quad \mathbb{H} = \tilde{H} + \tilde{v},
\]

where \( p_k^t \) and \( P_k^{-1} \) are defined in (8)-(9), \( \tilde{H} + \tilde{v} \) is the statistically approximated noise-free measurement, and \( \tilde{v} = h(\tilde{x}_k) - \tilde{H} \tilde{x}_k \) is the statistical approximation error. Then, prediction (13) and the update equations (14) for UIF are as follows:

\[
\tilde{y}_{k+1} = Y_k \tilde{x}_k, \quad Y_{k+1} = P_k^{-1},
\]

(13)
\[
\tilde{y}_{k+1} = \tilde{y}_k + H_k R^k \left( \tilde{z}_k + H_k \tilde{x}_k \right), \quad Y_{k+1} = Y_k + H_k R^k H^T_k.
\]

(14)

In the following section, a square root version of the UIF equations (12)-(14) is proposed so as to retain the positive definiteness of the covariance matrix. The Square Root Receding Horizon UIF and its extensions are also proposed. Thus, we propose a square root version of UIF (SRUIF) to improve the good performance in empirical error covariance.

Let us denote the square root of the information matrix as \( \Sigma_k = \Sigma_k^1 \) and assume that \( \Sigma_k^1 \) is known and that the square root matrices for noise covariances are \( Q_k = \Sigma_k^1 \Sigma_k^0 \), \( R_k = \Sigma_k^0 \Sigma_k^1 \). Then, instead of using the full information matrix, the square root matrix \( \Sigma_k^1 \) is recursively predicted (15)-(22) and updated (23)-(25).

Let us define \( \Sigma_k^1 = \left[ \Sigma_k^1 \right]^{1/2} \) and sigma-points according to following expressions:

\[
\begin{align*}
\tilde{X}_{0,i} &= \tilde{x}_{0,i} + \tilde{v}_i, \quad \Sigma_{0,i} = \Sigma_{0,i} + \tilde{v}_i, & i = 1, \ldots, n_a, \\
\tilde{X}_{0,i} &= \tilde{x}_{0,i} + \tilde{v}_i, \quad \Sigma_{0,i} = \Sigma_{0,i} + \tilde{v}_i, & i = n_a + 1, \ldots, 2n_a,
\end{align*}
\]

where \( \tilde{v}_i = \text{diag} \{ \tilde{v}_i, \ldots, \tilde{v}_i \} \) is the block diagonal matrix. The resulting centered sigma-point set of the predicted state (16) and predicted measurement (17) are

\[
\begin{align*}
\tilde{X}_{k+1,i} &= \tilde{X}_{0,i} + \tilde{v}_i, \quad \Sigma_{k+1,i} = \Sigma_{0,i} + \tilde{v}_i, & i = 1, \ldots, n_a, \\
\tilde{Z}_{k+1,i} &= \tilde{X}_{0,i} + \tilde{v}_i, \quad \Sigma_{k+1,i} = \Sigma_{0,i} + \tilde{v}_i, & i = n_a + 1, \ldots, 2n_a.
\end{align*}
\]

The predicted square root error matrix (18) and cross-covariance matrix (19)-(21) can then be obtained as

\[
\Sigma_{k+1} = \text{Triu} \left( \tilde{X}_{k+1}, \tilde{Z}_{k+1} \right), \quad \Sigma_{k+1} = \left( \Sigma_{k+1} \right)^{1/2},
\]

(18)

where "Triu" denotes a general triangularization algorithm (e.g., QR decomposition). Sigma-points are then drawn using \( \tilde{X}_{k+1,i} \),

\[
\begin{align*}
\tilde{X}_{k+1,i}^* &= \tilde{X}_{k+1,i} + \tilde{v}_i, , \tilde{X}_{k+1,i}^* &= \tilde{X}_{k+1,i} + \tilde{v}_i + \left( \tilde{v}_i + \mathbf{L} \right) \tilde{v}_i, & i = 1, \ldots, n_a, \\
\tilde{X}_{k+1,i}^* &= \tilde{X}_{k+1,i} + \tilde{v}_i, , \tilde{X}_{k+1,i}^* &= \tilde{X}_{k+1,i} + \tilde{v}_i + \left( \tilde{v}_i + \mathbf{L} \right) \tilde{v}_i, & i = n_a + 1, \ldots, 2n_a.
\end{align*}
\]

(19)

And the corresponding centered sigma-point set of the predicted state is given by

\[
\tilde{X}_{k+1,i} = \tilde{X}_{k+1,i} + \tilde{v}_i, \quad \tilde{X}_{k+1,i}^* = \tilde{X}_{k+1,i} + \tilde{v}_i + \left( \tilde{v}_i + \mathbf{L} \right) \tilde{v}_i, \quad \Sigma_{k+1,i} = \left( \Sigma_{k+1,i} \right)^{1/2},
\]

(20)

where \( \tilde{X}_{k+1,i}^* \) is the sigma-point of the augmented state vector (19). The predicted cross covariance and predicted information state are then given as

\[
\begin{align*}
P_k^P^{1/2} &= X_k^1 Z_k^T, \\
P_k^P^* &= S_k^1 X_k^1 Z_k^T,
\end{align*}
\]

(21)
\[
\begin{align*}
\tilde{X}_{k+1,i}^* &= \tilde{X}_{k+1,i}^*, \\
P_k^P^* &= S_k^1 X_k^1 Z_k^T,
\end{align*}
\]

(22)
\[
\tilde{S}_k = \text{Triu} \left( S_k, \tilde{H}_k S_k^T \right), \quad \Sigma_k = \left( \Sigma_k \right)^{1/2},
\]

(23)
\[
\tilde{S}_k = \text{Triu} \left( S_k, \tilde{H}_k S_k^T \right), \quad \Sigma_k = \left( \Sigma_k \right)^{1/2},
\]

(24)

where \( \tilde{S}_k \) and \( \Sigma_k \) are lower triangular matrices. Finally, the information state is updated using

\[
\tilde{y}_{k+1} = \tilde{y}_{k+1} + H_k R^k \left( \tilde{z}_k + H_k \tilde{x}_k \right)
\]

(25)

Proof of (24): With simple decompositions, the prediction step (15)-(22) is straightforward. For the update step from (14), the update equation of information filtering with respect to the square root matrix is given as
\[ S_{p} = S_{q} + \hat{H}R_{\text{in}}\hat{H}^{\top}, \]  
\[ (26) \]
where \( \hat{H} \) is calculated from (23). Then, considering the square root matrix of measurement noise covariance \( S_{z_{0}} \), (26) can be simply decomposed to the following expression:
\[ S_{p}^{\top} = [S_{q}, \hat{H}S_{z_{0}}\hat{H}^{\top}] = [S_{q}, \hat{H}S_{z_{0}}^{\top}]. \]  
\[ (27) \]
Therefore, when \( S_{q}, \hat{H} \) and \( S_{z_{0}} \) are available, we can obtain the updated square root information matrix \( S_{q} \) by using triangularization which leads to (24).

### B. Square root receding horizon UIF

Existing nonlinear filtering algorithms such as EKF, UKF, and PF try to approximate the optimal Bayesian solution with full information consideration \( \tilde{z} \). In contrast, the conditional pdf in RHE is approximated using limited information such as \( p(x_{\iota} | z_{\iota}) = p(x_{\iota} | z_{\iota+1}) \), where \( \Delta \) is the length of the horizon window. This condition implies that the filter has a FIR structure, and could be implemented in a semi-batch manner in which the complexity of the filter linearly increases proportional to \( \Delta \).

When each horizon window is receding, the initial conditions are needed. In designing the RHFs, determination of the initial condition is an important parameter, because when it is not determined properly filters may drift and fail to give accurate estimates. Specifically, if we implement the RHF using the standard Kalman filtering technique, we need the initial conditions, \( \tilde{x}_{\iota-1}, \hat{P}_{\iota-1} \). Given these initial conditions, the Kalman filter begins to calculate the sequence of intermediate estimates \( \{\tilde{x}_{0:1}, \tilde{x}_{1:1}, \tilde{x}_{1:2}, \ldots, \tilde{x}_{1:1}\} \) and their covariance.

In this paper, we design nonlinear RHFs that do not require specific initial conditions, by using a trick from the SRUIF. UIF is utilized for the main framework in designing RHF so that the initial conditions could be ignored by simply setting as \( \tilde{x}_{\iota-1} = 0, \hat{P}_{\iota-1} = \infty \). The physical meaning of this choice was discussed in the previous work [3], which also employs the information Kalman filter. In brief, it can be understood that we are able to ignore the initial conditions by assuming that the zero information matrix implies the covariance matrix becomes infinite, i.e., \( \hat{P}_{\iota-1} = \infty \), then allowing \( \tilde{x}_{\iota-1} \) to be any value.

For simplicity, we use the function notation of SRUIF to summarize equations (15)-(25): \( \{\tilde{x}_{\iota}, S_{\iota}\} = \text{SRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \). Algorithm 1 then describes a new RHF based on the proposed SRUIF as Step 1-2, denoted by the function notation as \( \{\tilde{x}_{\iota}, S_{\iota}\} = \text{SRHRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \). Note that a square root receding horizon UIF (SRHRUIF) is defined for \( \iota > \Delta \), because RHF requires \( \Delta \) observations to start. Thus, if \( \iota < \Delta \), SRUIF notation is used; otherwise, SRHRUIF notation is utilized.

#### Algorithm 1. Square Root Receding Horizon UIF:

Given \( \tilde{y}_{\iota}, S_{\iota} \), \( \tilde{y}_{\iota} \neq \tilde{P}_{\iota-1} \).

For \( \iota = 1, 2, \ldots \):

- If \( \iota < \Delta \), \( \{\tilde{x}_{\iota}, S_{\iota}\} = \text{SRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \). \( \tilde{P}_{\iota} = (S_{\iota}^{\top})^{-1} \). \( \tilde{x}_{\iota} = \tilde{P}_{\iota-1} \).
- Else, \( \{\tilde{x}_{\iota}, S_{\iota}\} = \text{SRHRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \). Step 1, 2

### C. Receding horizon particle filter

In this subsection, we propose another type of RHF based on the PF framework. One of the key issues in an efficient PF is how to design the importance sampling (IS) function used to sample particles [11]. Generally, a state transition pdf, i.e., \( p(x_{\iota} | x_{\iota-1}) \) is used for the IS function in a conventional PF. However, this choice leads to the particle degeneracy problem when the shape of the state transition pdf is broad and the shape of the observation likelihood is peaky [6]. In the literature, a hybrid PF approach has been discussed in an attempt to address this issue by employing Kalman filters in PF frameworks [6], [11].

Specifically, an optimal IS function is proposed as the following form:
\[ q(x_{\iota} | x_{\iota-1}, z_{\iota}) = p(x_{\iota} | x_{\iota-1}, z_{\iota}) \]
This minimizes the variance of the importance weight conditional on \( x_{\iota-1} \) and \( z_{\iota} \), where \( \iota \) is the particle index [11]. The optimal IS function is then implemented using UKF to avoid degeneracy and filter drift problems, this is known as unscented PF (UPF) [6].

The implementation of RHF under PF framework is similar to the UPF, and it is simple and efficient in terms of memory requirement because knowledge of the initial conditions for the particles is not required. In particular, the IS function is approximated as \( q(x_{\iota} | x_{\iota-1}, z_{\iota}) = p(x_{\iota} | x_{\iota-1}, z_{\iota}) \) and implemented via Algorithm 1. The new hybrid PF, the receding horizon PF (RHPF), is then proposed as summarized in Algorithm 2.

#### Algorithm 2. Receding Horizon PF

For \( \iota = 1, 2, \ldots \):

For \( i = 1, \ldots, M \) (\( M \) : number of particles in PF)

- If \( \iota < \Delta \), \( \{x_{\iota}^{(i)}, S_{\iota}^{(i)}\} = \text{SRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \).
- \( x_{\iota}^{(i)} = \tilde{x}_{\iota-1}^{(i)} \). \( S_{\iota}^{(i)} = \tilde{P}_{\iota}^{(i)} \).

Else, \( \{x_{\iota}^{(i)}, S_{\iota}^{(i)}\} = \text{SRHRUIF}(\tilde{x}_{\iota-1}, S_{\iota-1}, z_{\iota}) \). End If

Set \( q(x_{\iota}^{(i)} | x_{\iota-1}^{(i)}, z_{\iota}) = N(x_{\iota}^{(i)}, P_{\iota}^{(i)}) \).

Draw particle \( x_{\iota}^{(i)} - q(x_{\iota}^{(i)} | x_{\iota-1}^{(i)}, z_{\iota}) \).

Update weight \( a_{\iota}^{(i)} = p(z_{\iota} | x_{\iota}^{(i)} z_{\iota}) q(x_{\iota}^{(i)} | x_{\iota-1}^{(i)}, z_{\iota}) \).

End For, Resampling Step in [5]: \( \tilde{x}_{\iota} = \arg \max_{i} (a_{\iota}^{(i)}) \)

End For

### D. Extension to GM model

In visual tracking applications, due to relatively accurate
observations, UPF has been popular for appearance model based tracking [12] where the standard PF often fails. However, when the dimension of the state vector is high and the search space in the prediction step becomes larger, a huge number of particles is required. In such cases, UPF is not appropriate, especially in time-critical applications.

To overcome this limitation, the GM model (GMM) is adopted [7], [13]. By the same manner in [7], Gaussian mixture receding horizon PF (GMRHPF) is additionally implemented. When the UPF and RHPF assign UKF or SRRHUIF for every particle to approximate the IS function, the GMRHPF employs SRRHUIF for each GM. Here, the number of GMMs is significantly less than that of particles, so a substantial reduction in computational time can be achieved. Moreover, performance of the filter is significantly improved when the multi-mode of distribution is distinctive.

It is assumed that the GMM of the initial state and noise pdfs are known (e.g., number of GMs, weight, mean and covariance). Then, the posterior pdf of state is recursively approximated as in Algorithm 3.

Algorithm 3 Gaussian Mixture Receding Horizon PF

For $t = 1, 2, ...$

Given GMMs of the posterior pdf and noise pdfs as

$$p(x_{t-1}, \xi_{t-1}) = \sum_{i=1}^{G} \mu_{i, t-1}^{(0)} N(x_{t-1}; \mu_{i, t-1}^{(0)}, \Sigma_{i, t-1}^{(0)}),$$

$$p(u_{t-1}) = \sum_{i=1}^{G} \mu_{i, t-1}^{(0)} N(u_{t-1}; \mu_{i, t-1}^{(0)}, \Sigma_{i, t-1}^{(0)}),$$

where $G$, $I$, $J$ are the number of components in GMMs; $\mu_{i, t-1}^{(0)}$, $\Sigma_{i, t-1}^{(0)}$ are the mixture weights. Here, GM parameters of noise pdfs are known and fixed while GM parameters of posterior pdf are recursively estimated from initial values as Step 1-3.

Step 1. Prediction

For $g = 1, ..., G$, (for every $g$ and $i, \ g' = g + (i-1)G : : G = GI$)

Use the prediction step of Algorithm 1 to calculate

$$p(x_{t} | \xi_{t-1}) = \sum_{i=1}^{G} \mu_{i, t-1}^{(0)} N(x_{t}; \mu_{i, t-1}^{(0)}, \Sigma_{i, t-1}^{(0)}),$$

whose mixture weight is

$$\mu_{i, t}^{(0)} = \mu_{i, t-1}^{(0)} \rho_{i}^{(0)} \left( \sum_{i=1}^{G} \sum_{j=1}^{J} \mu_{j, t-1}^{(0)} \rho_{j}^{(0)} \right).$$

Step 2. IS function generation: Measurement update for local GMs

For $g = 1, ..., G$, (for every $g$ and $j, \ g' = g + (j-1)G : : G = GJ$)

Complete the measurement update step of Algorithm 1 to obtain the IS function

$$\tilde{p}(x_{t} | z_{t}) = \sum_{i=1}^{G} \mu_{i, t}^{(0)} \rho_{i}^{(0)} N(z_{t}; \mu_{i, t}^{(0)}, \Sigma_{i, t}^{(0)}),$$

whose mixture weight is

$$\mu_{i, t}^{(0)} = \mu_{i, t}^{(0)} \rho_{i}^{(0)} \left( \sum_{i=1}^{G} \sum_{j=1}^{J} \mu_{j, t}^{(0)} \rho_{j}^{(0)} \right) p_{j} \left( z_{t} \mid \mu_{i, t}^{(0)}, \Sigma_{i, t}^{(0)} \right),$$

where the $j$th likelihood

$$p_{j} \left( z_{t} \mid \mu_{i, t}^{(0)}, \Sigma_{i, t}^{(0)} \right)$$

is evaluated at $\mu_{i, t}^{(0)}$.

End For

Step 3: Measurement update of PF and reformulation of GMM

1. Draw $M$ particles $[\varphi_{l}^{(0)}, l = 1, ..., M]$, by importance sampling using the IS function, i.e.,

$$\tilde{p}(x_{t} | z_{t}) = \sum_{l=1}^{M} \alpha_{l}^{(t)} N(x_{t}; \mu_{l, t}^{(t)}, \Sigma_{l, t}^{(t)}),$$

and the calculate corresponding importance weights [1].

2. Use a weighted expectation maximization (EM) algorithm to fit a GMM to the set of weighted particles $[\varphi_{l}^{(0)}, \varphi_{l}^{(t)}, l = 1, ..., M]$, representing the updated GMM approximate as $p(x_{t} | z_{t}) = \sum_{i=1}^{G} \mu_{i, t}^{(t)} N(x_{t}; \mu_{i, t}^{(t)}, \Sigma_{i, t}^{(t)})$, where EM is seeded by the $G$ means, covariances and mixing weights of the prior state GMM, $p(x_{t-1}, \xi_{t-1})$, and iterated until a certain convergence.

End For

V. EXPERIMENTAL RESULTS

In this section, we evaluate the proposed algorithms introduced in Section IV by comparing them with well-known numerical scalar benchmarking examples. Overall, Example 1 is found suitable to show the superiority of PF compared to the Kalman filtering algorithms (e.g., UKF, EKF) [5]. Example 2 is taken from a well-known model [6], [7] and effectively shows the performance differences between the PF-based algorithms (e.g., PF [5], Gaussian Sum PF (GSPF) [13], and UPF [6]).

Example 1: Univariate non-stationary growth model (UNGM). Through a nonlinear scalar example, the improved accuracy and good performance of SRRHUIF in empirical error covariance are verified by comparing it with the classical augmented UIF and PF. The following expression describes UNGM [5]:

$$x_{t} = 0.5x_{t-1} + 2 \frac{\Delta t}{1 + x_{t-1}^{2}} \cos(2 \pi t) + x_{t-1}, \quad z_{t} = \frac{x_{t}^{2}}{20} + v_{t}, \quad t = 1, ..., 100,$$

where the process and observation noises $u_{t}$ and $v_{t}$ are white Gaussian, i.e., $u_{t} \sim N(u_{t}, 0.10)$ and $v_{t} \sim N(v_{t}, 0.1)$, respectively.

Table I and Fig. 1 show that over the entire duration of the experiment, SRRHUIF outperformed UIF and is competitive with a PF having 1000 particles. The results show that RHE improved the accuracy because the approximation errors in nonlinear problems can be reduced using FIR filtering. Then, according to the value of the time-averaged MSE, SRRHUIF facilitated the good performance in Monte Carlo runs. Note that the results of PF-algorithms are not provided because they have a similar performance to PF.

Example 2: Nonlinear model with non-Gaussian process noise and non-stationary observation model [6], [7]. A nonlinear scalar time series is generated using the following models:

$$x_{t} = 0.5x_{t-1} + 1 + 8 \sin(0.04 \pi (t-1)) + u_{t},$$

$$z_{t} = \begin{cases} 0.2z_{t-1}^{2} + v_{t}, & t \leq 50, \\ 0.5x_{t-1}^{2} - 2 + v_{t}, & t > 50, \quad t = 1, ..., 60, \end{cases}$$

where $u_{t}$ is a Gamma($3, 2$) random variable for process noise and the observation noise is normal $v_{t} \sim N(v_{t}, 0.10)$.

In Table 2, PF, its variants, and the proposed algorithms are compared with respect to their accuracy and computational time. Here, results of the nonlinear Kalman filters are not shown because their performances are far inferior to the PF algorithms. Note that the same number of particles (200) was
used in all algorithms to ensure a fair comparison and that the root mean square error (RMSE) was used as the better scale. From the experimental results, GMRHPF outperformed all existing algorithms including GSPF [13]. In all experiments, we also implemented the proposed algorithms via cubature-points [20]. However, only slight performance improvements were achieved. The results of the cubature-point based algorithm were straightforward and its simple implementation without tuning was only meaningful for practitioners.

UKF parameters were set as $\alpha = 0.5$, $\beta = 2$, and $\kappa = 3 - \eta$. The size of window $\Delta$ ranges from 2 to 5 depending on the given model. In Example 2, the number of GM components was set as $G = 5$, $I = 1$, $J = 1$. Tuning for best performance was required; however, we obtained good performance for these particular examples, i.e., low variance in the time-averaged MSE. This technical note limits our discussion of the particular examples, i.e., low variance in the time-averaged MSE. This technical note limits our discussion of the performance evaluation through numerical examples due to the limitation of space, and the convergence analysis remains as our forthcoming work.

VI. CONCLUSION

Novel nonlinear estimation algorithms were presented using a receding horizon approach and square root information filtering. By considering a set of recent observations and ignoring initial conditions for filter, the FIR structure was implemented. SRRHUIF, RHPF, and GMRHPF were derived and evaluated with popular nonlinear and non-Gaussian models. Well-known benchmarking examples confirm that the proposed RHPFs are effective modifications of classical nonlinear filters.

REFERENCES


