Hybrid Cubature Filter: Theory and Tracking Application

Ienkaran Arasaratnam, and Simon Haykin, *Fellow, IEEE,*

Abstract

In a recent paper [1], **we derived a new discrete-time Bayesian filter, which we have named the cubature Kalman filter (CKF). In this paper, we extend this nonlinear filter to deal with a classical state estimation problem, whose state-space model is described by a continuous-time process equation and a discrete-time measurement equation. For a reliable implementation of the CKF in a finite word-length machine, the CKF is structurally modified to propagate the square-roots of the covariances; the resulting filter is named the square-root cubature Kalman filter (SCKF). For the 'hybrid' state-space model under consideration, the time-update of the SCKF is computationally expensive. To mitigate this practical issue, we modify the SCKF to obtain a cost-reduced SCKF, whose time-update recursively propagates a set of cubature points without estimating the predicted mean and covariance at every small time-step. The two new formulations– the SCKF and the cost reduced SCKF– are validated in tracking a ballistic target on reentry. The results, presented herein, indicate that the SCKF and its cost reduced variant perform equally well and markedly outperform existing algorithms.**

Index Terms

Bayesian filters, Cubature Kalman filter, Itô-Taylor expansion, Nonlinear filtering, Square-root filtering.

I. Introduction

Nonlinear-filtering problems are typically described by a state-space model consisting of a pair of equations: process equation and measurement equation. In many practical problems, the process equation is often derived from the underlying physics of a dynamic system, and expressed in the form of a set of differential equations. Since the sensors are digital devices, the measurements are available at discrete times. In this 'hybrid' setting, we have a state-space model with a continuous-time process equation and a discrete-time measurement equation. This hybrid-setting often arises in signal processing [5], control [3] and finance [18].

Consider a state process generated from

$$d\mathbf{x}_t \quad = \quad \mathbf{f}(\mathbf{x}_t, t)dt + \sqrt{\mathbf{Q}}d\boldsymbol{\beta}_t, \tag{1}$$

where

- $\mathbf{x}_t \in \mathbb{R}^n$ is the state of the system to be estimated at time t;
- $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a known function;
- $\mathbf{Q} \in \mathbb{R}^{n \times n}$ is the diffusion matrix, also called the spectral density matrix or the gain matrix of process noise; and
- $\boldsymbol{\beta}_t \in \mathbb{R}^n$ denotes the standard Brownian motion with increments $d\boldsymbol{\beta}_t$ being independent of $\mathbf{x}_t$.

Equation (1) is the simplest case of a more general Itô's stochastic differential equation. It can equivalently be written in the form:

$$\frac{d\mathbf{x}_t}{dt} \quad = \quad \mathbf{f}(\mathbf{x}_t, t) + \sqrt{\mathbf{Q}}\mathbf{w}_t, \tag{2}$$

where $\mathbf{w}_t \in \mathbb{R}^n$ is the standard Gaussian noise and is interpreted as the time-derivative of the Brownian motion. This second equation (2) looks physically more meaningful than (1)– It is easy to communicate the physics of the problem that the plant is modeled by a deterministic equation driven by stochastic noise. However, (2) is mathematically less rigorous than (1) and it does not have a solution because $\int \mathbf{w}_t dt$ is not defined, whereas $\int d\boldsymbol{\beta}_t$ is well defined [18]. In the nonlinear filtering literature, Equations (1) and (2) have been interchangeably used.

In the Bayesian filtering paradigm, the conditional density of the state given the measurements, also called the posterior density of the state, provides a complete statistical description of the state at that time. The optimal continuous-discrete Bayesian filter consists of the following:

- Propagation of the 'old' posterior density between the measurement instants
- Use of Bayes' rule to update this propagated density at the measurement instants

Now, let us look at the propagation of the old posterior density. It is well known that for the process equation (1), the propagation or the temporal evolution of the probability density of the state at time t obeys the famous Fokker-Planck equation, also called Kolmogorov's forward equation [12]:

$$\frac{\partial p}{\partial t} \quad = \quad -\sum_{i=1}^{n} \frac{\partial (p f_i)}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^{n} \frac{\partial^2 \big((\mathbf{Q})_{ij} p\big)}{\partial x_i \partial x_j}, \tag{3}$$

where $p = p(\mathbf{x}_t|\mathbf{D}_k)$ with $t \leq (k+1)$ and the measurement history up to time k, $\mathbf{D}_k = \{\mathbf{z}_i,\ i = 1, 2 \ldots k\}$. Let the measurement $\mathbf{z}_{k+1}$ be generated from the measurement equation:

$$\mathbf{z}_{k+1} \;=\; \mathbf{h}(\mathbf{x}_{k+1}, k+1) + \mathbf{v}_{k+1}, \tag{4}$$

where $\mathbf{v}_{k+1}$ is Gaussian noise with mean zero and covariance $\mathbf{R}_{k+1}$.

On the receipt of a measurement $\mathbf{z}_{k+1}$, using Bayes' rule, we get the posterior density

$$p(\mathbf{x}_{k+1}|\mathbf{D}_{k+1}) \;=\; \frac{1}{c_{k+1}} p(\mathbf{x}_{k+1}|\mathbf{D}_k) p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}), \tag{5}$$

where the measurement likelihood function

$$p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}) \;=\; \mathcal{N}(\mathbf{z}_{k+1}; \mathbf{h}(\mathbf{x}_{k+1}, k+1), \mathbf{R}_{k+1}),$$

with $\mathcal{N}(.,.)$ being the conventional symbol for a Gaussian density, the normalizing constant

$$c_{k+1} \;=\; \int_{\mathbb{R}^n} p(\mathbf{x}_{k+1}|\mathbf{D}_k) p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}) d\mathbf{x}_{k+1}, \tag{6}$$

and the predictive density $p(\mathbf{x}_{k+1}|\mathbf{D}_k)$ is obtained from (3).

The pair of equations (3)-(5) describes the continuous-discrete Bayesian filter only in conceptual terms. The reason is that there is no guarantee that the new posterior will remain closed with a finite summary statistic expressed in terms of (quasi-)moments. Specifically, the Fokker-Planck equation has to be approximately solved except in the following two cases– For a linear process equation, (3) reduces to the time-update of the Kalman-Bucy filter [14]; whereas for a process equation exhibiting nonlinearity of Beneš-type, it reduces to the time-update of the Beneš filter [6]. In [8], Daum has further extended the class of nonlinear dynamic systems that admit a sufficient statistic of a constant finite dimension.

For a generic nonlinear setting, we have to resort to numerical methods to approximately compute both the predictive density and the posterior density. The novel contributions of this paper are as follows:

- Use of the Itô-Taylor expansion of order 1.5 to transform a continuous-discrete filtering problem into a more familiar discrete-time filtering problem for the first time in the nonlinear filtering literature.
- Development of a new square-root cubature filter for improved reliability.
- Development of a cost-reduced square-root cubature filter that significantly reduces the computational cost associated with the time-update of the square-root cubature filter.

The paper is structured as follows: Section II reviews the existing approximate solution to nonlinear continuous-discrete stochastic systems based on the extended Kalman filter. Section III contains the development of the cubature Kalman filter (CKF) for the considered 'hybrid' stochastic nonlinear systems. Section IV presents a square-root extension of the CKF for improved reliability in a limited precision system. The time-update of the square-root CKF is computationally the most costly part of the algorithm. In Section V, we derive a cost-reduced square-root CKF, whose time-update recursively propagates a set of cubature points without estimating the predicted mean and covariance at every time-step. Section VI is devoted to an experiment, which compares the proposed solutions with existing solutions in tracking a ballistic target on reentry. Section VII concludes the paper with some final remarks.

II. Approximate Solutions: A Brief Survey

In this section, we first review the most established, traditional filtering algorithm known as the extended Kalman filter (EKF). Suppose that the predictive density is Gaussian with the mean $\hat{\mathbf{x}}_{t|k} = \mathbb{E}[\mathbf{x}_t|\mathbf{D}_k]$ and covariance $\mathbf{P}_{t|k} = \mathbb{E}[(\mathbf{x}_t - \hat{\mathbf{x}}_{t|k})(\mathbf{x}_t - \hat{\mathbf{x}}_{t|k})^T|\mathbf{D}_k]$, where $t \geq k$. In the time-update, from (3), the following mean-covariance pair is obtained in a differential equation form [9], [12]:

$$\frac{d\hat{\mathbf{x}}_t}{dt} \approx \mathbf{f}(\mathbf{x}_t, t) \tag{7}$$

$$\frac{d\mathbf{P}_t}{dt} \approx P_t^T\mathbf{J}_{f,t} + \mathbf{J}_{f,t}^T\mathbf{P}_t + \mathbf{Q}, \tag{8}$$

where $\mathbf{J}_f$ is the Jacobian of $\mathbf{f}(.)$ with respect to $\mathbf{x}_t$. Assuming $\mathbf{J}_f$ remains constant over the time interval $[k, k+\delta]$, an approximate solution to (8) at time $(k+\delta)$ is given by

$$\mathbf{P}_{k+\delta|k} \approx \Phi\mathbf{P}_{k|k}\Phi^T + \delta\mathbf{Q}, \tag{9}$$

where

$$\Phi = \mathbf{I}_n + \delta\mathbf{J}_{f,k} + \frac{1}{2}\delta^2\mathbf{J}_{f,k}^2$$

with $\mathbf{I}_n$ denoting the conventional symbol for an n-dimensional identity matrix. To compute the predicted mean $\hat{\mathbf{x}}_{k+\delta|k}$ from (7), numerical methods such as the Euler method and the family of Runge-Kutta methods can be used. When the Euler method is used, the predicted state estimate at time $(k+\delta)$ is given by

$$\hat{\mathbf{x}}_{k+\delta|k} = \hat{\mathbf{x}}_{k|k} + \mathbf{f}(\hat{\mathbf{x}}_{k|k}, k). \tag{10}$$

The predicted state estimate (10) and its covariance (9) collectively form the time-update of the standard EKF. They are recursively propagated until time $(k+1)$. On the receipt of a new measurement $\mathbf{z}_{k+1}$ at time $(k+1)$, the measurement-update of the standard EKF is used to obtain the first two moments of the posterior density [9], [12].

As an alternative to the time-update just mentioned, the *Itô-Taylor expansion of order 0.5* can be used to discretize the continuous-time Itô process equation (1) [15]. Subsequently, numerical integration methods can be used to estimate the predicted state and its covariance. According to Section 10.2 of [15], for the time interval $[k,\ k+\delta]$, applying the Itô-Taylor expansion to (1) leads to

$$\mathbf{x}_{k+\delta} \ \approx \ \mathbf{x}_k + \delta \mathbf{f}(\mathbf{x}_k, k) + \sqrt{\delta \mathbf{Q}}\mathbf{q}_k, \tag{11}$$

where $\mathbf{q}_k = (\boldsymbol{\beta}_{k+\delta} - \boldsymbol{\beta}_k)$ is the standard Gaussian random variable. The approximate process equation (11) can be considered as a discrete-time counterpart of the Itô-type process equation (1). This discretization scheme is strongly convergent with the order of 0.5. That is, we may say that

$$\mathbb{E}[|\mathbf{x}_t - \hat{\mathbf{x}}_t|] \ \leq \ c\delta^{0.5},$$

where

- $\mathbf{x}_t$ is the exact solution;
- $\hat{\mathbf{x}}_t$ is the estimate due to the Itô-Taylor expansion of order 0.5 at time t; and
- c is a constant that does not depend on δ.

In the literature on nonlinear filtering, researchers in the past have often referred to this expansion in the name of the *Euler expansion* [11], [16].

To compute the predicted state estimate more accurately before receiving the measurement at time step $(k+1)$, several integration steps are performed within the measurement-time interval $[k, k+1]$ as described below:

Time Update: From time step k to $(k+1)$

1) Divide the time interval $[k, k+1]$ into M equal subintervals, each of length δ.
2) Given $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$ at time $t = k$, recursively estimate the predicted state at the m-th

intermediate time step as $(m = 1, 2, \dots M)$:

$$\begin{aligned}\hat{\mathbf{x}}_{k+m\delta|k} &= \mathbb{E}\big[\mathbf{x}_{k+m\delta}|\mathbf{D}_k\big] \\ &\approx \int_{\mathbb{R}^n} [\mathbf{x} + \delta \mathbf{f}(\mathbf{x}, k + (m-1)\delta)] \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_{k+(m-1)\delta|k}, \mathbf{P}_{k+(m-1)\delta|k}) d\mathbf{x} \qquad (12)\end{aligned}$$

3) Recursively estimate the corresponding predicted error covariance $(m = 1, 2, \dots M)$:

$$\begin{aligned}\mathbf{P}_{k+m\delta|k} &= \mathbb{E}\big[(\mathbf{x}_{k+m\delta} - \hat{\mathbf{x}}_{k+m\delta|k})(\mathbf{x}_{k+m\delta} - \hat{\mathbf{x}}_{k+m\delta|k})^T|\mathbf{D}_k\big] \\ &\approx \delta\mathbf{Q} + \int_{\mathbb{R}^n} ([\mathbf{x} + \delta \mathbf{f}(\mathbf{x}, k + (m-1)\delta)] - \hat{\mathbf{x}}_{k+m\delta|k})([\mathbf{x} + \delta \mathbf{f}(\mathbf{x}, k + (m-1)\delta)] - \hat{\mathbf{x}}_{k+m\delta|k})^T \\ &\quad \times \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_{k+(m-1)\delta|k}, \mathbf{P}_{k+(m-1)\delta|k}) d\mathbf{x}. \qquad (13)\end{aligned}$$

To numerically compute the integrals in (12)-(13), the Gauss-Hermite quadrature rule is used in [16]. As a computationally efficient alternative to the Gauss-Hermite quadrature rule, we may also use the third-degree spherical-radial cubature rule, which is briefly described in the next section. When $m = M$ or at time $(k+1)$, we have the Gaussian-distributed predicted state variable with the mean $\hat{\mathbf{x}}_{k+1|k}$ and covariance $\mathbf{P}_{k+1|k}$, which is finally updated using the measurement $\mathbf{z}_{k+1}$.

III. Cubature Kalman Filtering

In this section, we present a powerful numerical tool, called the Cubature Kalman Filter (CKF) for continuous-discrete systems. The CKF is the closest known approximation to the discrete-time Bayesian filter that could be designed in a nonlinear setting under the Gaussian assumption [1], [2]. The heart of the CKF is a third-degree spherical-radial cubature rule that approximates integrals, whose integrands are of the form:

nonlinear function × *Gaussian.*

The cubature rule uses a set of $2n$ equally weighted cubature points, where n is the dimension of the integration domain. For example, we use the cubature rule to approximate an n-dimensional Gaussian weighted integral as follows:

$$\int_{\mathbb{R}^n} \mathbf{f}(\mathbf{x}) \mathcal{N}(\mathbf{x}; \mu, \Sigma) d\mathbf{x} \approx \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{f}(\mu + \sqrt{\Sigma}\xi_i).$$

where $\Sigma = \sqrt{\Sigma}\sqrt{\Sigma}^T$ and the cubature points [1], [2]:

$$\xi_i = \begin{cases} \sqrt{n}\mathbf{e}_i, & i = 1, 2 \ldots n \\ -\sqrt{n}\mathbf{e}_{i-n}, & i = n+1, n+2 \ldots 2n. \end{cases}$$

The cubature rule is exact for nonlinear functions belonging to monomials of degree three or less. Next, we discuss on how to transform the CKF into a useful tool in a continuous-discrete system under two different steps: time-update and measurement-update.

A. Time Update

In this first step, we update the old posterior density of the state before receiving a new measurement, and obtain the predictive density. To discretize the process equation (1) in the time domain, we apply the Itô-Taylor expansion that is strongly convergent with the order of 1.5 to (1) at time $t = (k + \delta)$ and get (Section 10.4 of [15])

$$\mathbf{x}_{k+\delta} = \underbrace{\mathbf{x}_k + \delta\mathbf{f}(\mathbf{x}_k) + \frac{1}{2}\delta^2(\mathbb{L}^0\mathbf{f}(\mathbf{x}_k))}_{\mathbf{J}(\mathbf{x}_k)} + \sqrt{\mathbf{Q}}\delta\mathbf{w} + (\mathbb{L}\mathbf{f}(\mathbf{x}_k))\delta\mathbf{y}, \quad (14)$$

where

- The two operators $\mathbb{L}^0$ and $\mathbb{L}^j$ $(j = 1, 2 \ldots n)$ are defined as follows:

$$\mathbb{L}^0 = \frac{\partial}{\partial t} + \sum_{i=1}^{n} \mathbf{f}^i \frac{\partial}{\partial \mathbf{x}^i} + \frac{1}{2} \sum_{j,p,q=1}^{n} \sqrt{\mathbf{Q}}^{p,j} \sqrt{\mathbf{Q}}^{q,j} \frac{\partial^2}{\partial \mathbf{x}^p \partial \mathbf{x}^q}$$
$$\mathbb{L}^j = \sum_{i=1}^{n} \sqrt{\mathbf{Q}}^{i,j} \frac{\partial}{\partial \mathbf{x}^i};$$

- By the notation $\mathbb{L}\mathbf{f}$, we denote a square matrix with its $(i,j)-$th element being $\mathbb{L}^j\mathbf{f}^i$, $(i, j = 1, \ldots, n)$;
- The function

$$\mathbf{J}(\mathbf{x}_k) = \mathbf{x}_k + \delta\mathbf{f}(\mathbf{x}_k) + \frac{1}{2}\delta^2(\mathbb{L}^0\mathbf{f}(\mathbf{x}_k)); \quad (15)$$

- The pair of correlated Gaussian random variables $(\delta\mathbf{w}, \delta\mathbf{y})$ can be generated from a pair of independent standard Gaussian random variables $(\mathbf{u}_1, \mathbf{u}_2)$ as follows:

$$\delta\mathbf{w} = \sqrt{\delta}\mathbf{u}_1 \sim \mathcal{N}(\mathbf{0}, \delta\mathbf{I}_n)$$
$$\delta\mathbf{y} = \frac{1}{2}\delta^{3/2}(\mathbf{u}_1 + \frac{\mathbf{u}_2}{\sqrt{3}}) \sim \mathcal{N}(\mathbf{0}, \frac{\delta^3}{3}\mathbf{I}_n)$$

with

$$\mathbb{E}[\delta\mathbf{w}\delta\mathbf{y}^T] \;=\; \frac{1}{2}\delta^2\mathbf{I}_n.$$

Given the statistic of $\mathbf{x}_k$ such that $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$, we write the predicted state estimate

$$\begin{aligned}\hat{\mathbf{x}}_{k+\delta|k} &= \mathbb{E}[\mathbf{x}_{k+\delta}|\mathbf{D}_k] \\ &\approx \mathbb{E}[\mathbf{J}(\mathbf{x}_k) + \sqrt{\mathbf{Q}}\delta\mathbf{w} + (\mathbb{L}\mathbf{f}(\mathbf{x}_k))\delta\mathbf{y}|\mathbf{D}_k]\end{aligned}$$

Because the noise terms are independent and zero-mean Gaussian, we further simplify matters by writing

$$\begin{aligned}\hat{\mathbf{x}}_{k+\delta|k} &= \mathbb{E}[\mathbf{J}(\mathbf{x}_k)|\mathbf{D}_k] \\ &= \int_{\mathbb{R}^n} \mathbf{J}(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k. \qquad (16)\end{aligned}$$

Similarly, we write the predictive state-error covariance

$$\begin{aligned}\mathbf{P}_{k+\delta|k} &= \mathbb{E}\big[(\mathbf{x}_{k+\delta} - \hat{\mathbf{x}}_{k+\delta|k})(\mathbf{x}_{k+\delta} - \hat{\mathbf{x}}_{k+\delta|k})^T|\mathbf{D}_k\big] \\ &\approx \int_{\mathbb{R}^n} \mathbf{J}(\mathbf{x}_k)\mathbf{J}^T(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k + \frac{\delta^3}{3}\int_{\mathbb{R}^n} (\mathbb{L}\mathbf{f}(\mathbf{x}_k))(\mathbb{L}\mathbf{f}(\mathbf{x}_k))^T\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k \\ &\quad + \frac{\delta^2}{2}\Big[\sqrt{\mathbf{Q}}\big(\int_{\mathbb{R}^n} \mathbb{L}\mathbf{f}(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k\big)^T + \big(\int_{\mathbb{R}^n} \mathbb{L}\mathbf{f}(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k\big)\sqrt{\mathbf{Q}}^T\Big] \\ &\quad - \hat{\mathbf{x}}_{k+\delta|k}\hat{\mathbf{x}}_{k+\delta|k}^T + \delta\mathbf{Q}. \qquad (17)\end{aligned}$$

To numerically compute the integrals present in (16) and (17), we use the third-degree cubature rule.

B. Measurement Update

In this second step, on the receipt of a new measurement $\mathbf{z}_{k+1}$, we update the predictive density and obtain the posterior density of the current state. In order to proceed with this step, we must make certain assumptions as described in the sequel. It is well known that the innovations sequence is a *white* sequence with zero mean [5]. Under the reasonable assumption that it is Gaussian, the innovations become independent of each other, and we write the innovations density

$$p(\boldsymbol{\epsilon}_{k+1}) \;=\; \mathcal{N}(\boldsymbol{\epsilon}_{k+1}; \mathbf{0}, \mathbf{P}_{zz,k+1|k}), \qquad (18)$$

with the second-order moment of the innovations or simply the innovations covariance

$$\begin{aligned} \mathbf{P}_{zz,k+1|k} &= \mathbb{E}[(\mathbf{z}_{k+1}-\hat{\mathbf{z}}_{k+1|k})(\mathbf{z}_{k+1}-\hat{\mathbf{z}}_{k+1|k})^T|\mathbf{D}_{k+1}] \\ &= \int_{\mathbb{R}^n} \mathbf{h}(\mathbf{x},k+1)\mathbf{h}^T(\mathbf{x},k+1)\mathcal{N}(\mathbf{x};\hat{\mathbf{x}}_{k+1|k},\mathbf{P}_{k+1|k})d\mathbf{x} - \hat{\mathbf{z}}_{k+1|k}\hat{\mathbf{z}}^T_{k+1|k} + R_{k+1}, \end{aligned} \tag{19}$$

where the predicted measurement

$$\hat{\mathbf{z}}_{k+1|k} = \int_{\mathbb{R}^n} \mathbf{h}(\mathbf{x},k+1)\mathcal{N}(\mathbf{x};\hat{\mathbf{x}}_{k+1|k},\mathbf{P}_{k+1|k})d\mathbf{x}. \tag{20}$$

Rearranging (18), we get

$$\begin{aligned} p(\boldsymbol{\epsilon}_{k+1}) &= \mathcal{N}(\mathbf{z}_{k+1}-\hat{\mathbf{z}}_{k+1|k};\mathbf{0},\mathbf{P}_{zz,k+1|k}) \\ &= \mathcal{N}(\mathbf{z}_{k+1};\hat{\mathbf{z}}_{k+1|k},\mathbf{P}_{zz,k+1|k}). \end{aligned} \tag{21}$$

From (21), it is understood the innovations density $p(\boldsymbol{\epsilon}_{k+1})$, and the filter likelihood density $p(\mathbf{z}_{k+1}|\mathbf{D}_{k+1})$ are related by one-to-one transformation.

To develop an approximate Bayesian filter, we further assume that the state process and the measurements process given the past measurement history, can jointly be approximated by Gaussian. That is, we write

$$p\Big([\mathbf{x}^T_{k+1} \quad \mathbf{z}^T_{k+1}]^T|\mathbf{D}_{k+1} \Big) = \mathcal{N}\left(\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{z}_{k+1}\end{pmatrix};\begin{pmatrix} \hat{\mathbf{x}}_{k+1|k} \\ \hat{\mathbf{z}}_{k+1|k}\end{pmatrix},\begin{pmatrix} \mathbf{P}_{k+1|k} & \mathbf{P}_{xz,k+1|k} \\ P^T_{xz,k+1|k} & \mathbf{P}_{zz,k+1|k}\end{pmatrix}\right), \tag{22}$$

where the cross-covariance

$$\mathbf{P}_{xz,k+1|k} = \int_{\mathbb{R}^n} \mathbf{x}\mathbf{h}^T(\mathbf{x},k+1)\mathcal{N}(\mathbf{x};\hat{\mathbf{x}}_{k+1|k},\mathbf{P}_{k+1|k})d\mathbf{x} - \hat{\mathbf{x}}_{k+1|k}\hat{\mathbf{z}}^T_{k+1|k}. \tag{23}$$

The assumption that the joint state-measurement process given the past measurement history is Gaussian implies both the predictive and innovations densities are Gaussian. Using Bayes' rule, we write the posterior density

$$\begin{aligned} p(\mathbf{x}_{k+1}|\mathbf{D}_{k+1}) &= p(\mathbf{x}_{k+1}|\mathbf{z}_{k+1},\mathbf{D}_k) \\ &= \frac{p(\mathbf{x}_{k+1},\mathbf{z}_{k+1}|\mathbf{D}_k)}{p(\mathbf{z}_{k+1}|\mathbf{D}_k)}. \end{aligned} \tag{24}$$

On the receipt of a new measurement $\mathbf{z}_{k+1}$, substituting (21) and (22) into (24) yields

$$p(\mathbf{x}_{k+1}|\mathbf{D}_{k+1}) = \mathcal{N}(\mathbf{x}_{k+1}; \hat{\mathbf{x}}_{k+1|k+1}, \mathbf{P}_{k+1|k+1}), \tag{25}$$

where

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

As mentioned previously, the CKF solves the problem of how to compute Gaussian-weighted integrals whose integrands are all of the form *nonlinear function* × *Gaussian density* present in (12), (13), (19), (20) and (23) using the third-degree spherical-radial cubature rule [7], [19].

Below, we summarize the cubature Kalman filtering algorithm for continuous-discrete stochastic systems:

CKF– Time Update

Initialize m to be zero.

1) Factorize

$$\mathbf{P}_{k+m\delta|k} = \mathbf{S}_{k+m\delta|k}\mathbf{S}_{k+m\delta|k}^T.$$

2) Evaluate the cubature points (i=1,2,...,2n)

$$X_{i,k+m\delta|k} = \mathbf{S}_{k+m\delta|k}\xi_i + \hat{\mathbf{x}}_{k+m\delta|k}.$$

3) Evaluate the following pair of propagated cubature point sets (i=1,2,...,2n)

$$\begin{aligned}
X^*_{i,k+(m+1)\delta|k} &= \mathbf{J}(X_{i,k+m\delta|k}) \\
Y^*_{i,k+(m+1)\delta|k} &= \mathbb{L}\mathbf{f}(X_{i,k+m\delta|k}).
\end{aligned}$$

4) Estimate the predicted state

$$\hat{\mathbf{x}}_{k+(m+1)\delta|k} = \frac{1}{2n}\sum_{i=1}^{2n} X^*_{i,k+(m+1)\delta|k}.$$

5) Estimate the predicted error covariance

$$\begin{aligned}\mathbf{P}_{k+(m+1)\delta|k} &= \frac{1}{2n}\sum_{i=1}^{2n}\boldsymbol{X}^*_{i,k+(m+1)\delta|k}\boldsymbol{X}^{*T}_{i,k+(m+1)\delta|k} + \frac{\delta^3}{6n}\sum_{i=1}^{2n}\mathbf{Y}_{i,k+(m+1)\delta|k}\mathbf{Y}^T_{i,k+(m+1)\delta|k} + \\ &\frac{\delta^2}{4n}\big[\sqrt{\mathbf{Q}}(\sum_{i=1}^{2n}\mathbf{Y}^T_{i,k+(m+1)\delta|k}) + (\sum_{i=1}^{2n}\mathbf{Y}_{i,k+(m+1)\delta|k})\sqrt{\mathbf{Q}}^T\big] - \hat{\mathbf{x}}_{k+(m+1)\delta|k}\hat{\mathbf{x}}^T_{k+(m+1)\delta|k} \\ &+\delta\mathbf{Q}. \end{aligned} \tag{26}$$

6) Increase m by one and repeat the steps (1)-(5) until m reaches M (that is, until time $(k+1)$).

CKF– Measurement Update

1) Factorize

$$\mathbf{P}_{k+1|k} = \mathbf{S}_{k+1|k}\mathbf{S}^T_{k+1|k}.$$

2) Evaluate the cubature points (i=1,2,...,2n)

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

3) Evaluate the propagated cubature points (i=1,2,...,2n)

$$Z_{i,k+1|k} = \mathbf{h}(X_{i,k+1|k}, k+1).$$

4) Estimate the predicted measurement

$$\hat{\mathbf{z}}_{k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k+1|k}.$$

5) Estimate the innovation covariance matrix

$$\mathbf{P}_{zz,k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k+1|k}Z^T_{i,k+1|k} - \hat{\mathbf{z}}_{k+1|k}\hat{\mathbf{z}}^T_{k+1|k} + \mathbf{R}_{k+1}.$$

6) Estimate the cross-covariance matrix

$$\mathbf{P}_{xz,k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} X_{i,k+1|k}Z^T_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\hat{\mathbf{z}}^T_{k+1|k}.$$

7) Estimate the cubature Kalman gain

$$\mathbf{W}_{k+1} = \mathbf{P}_{xz,k+1|k}\mathbf{P}^{-1}_{zz,k+1|k}. \tag{27}$$

8) Estimate the updated state

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

9) Estimate the corresponding error covariance

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{W}_{k+1}\mathbf{P}_{zz,k+1|k}\mathbf{W}^{T}_{k+1}. \tag{28}$$

IV. Square-Root Filtering for Improved Reliability

The CKF performs numerically sensitive operations such as matrix inversion (27) and subtraction of two positive definite matrices (28). These operations can lead to numerical errors that are generally manifested in the appearance of covariances which fail to be symmetric and positive (semi-)definite. In order to preserve the symmetry and positive (semi-)definiteness and to improve numerical accuracy, various ad-hoc methods have been introduced in the literature on Bayesian filtering. Some of them include [10]:

- Measurement-update with a sequence of scalar measurements in a preferred order
- Decoupled or quasi-decoupled covariances
- Symmetrization of covariances based on the formula $\mathbf{P}' = \frac{1}{2}(\mathbf{P} + \mathbf{P}^T)$
- Computation of only upper triangular entries of covariances
- Tikhonov regularization
- Joseph's covariance update
- Use of large process and measurement noise covariances
- Use of doubled-precision arithmetic

In contrast, as a systematic solution to preserve the properties of a covariance matrix and improve numerical accuracy, square-root algorithms, which propagate the square-roots of various error covariance matrices, have been proposed [13]. Following this line of thinking, we can also structurally reformulate the CKF developed for a continuous-discrete state-space model. A covariance matrix $\mathbf{P}$ of the CKF is often written in the form

$$\mathbf{P} \;=\; \mathbf{S_o}\mathbf{S_o}^T, \tag{29}$$

where $\mathbf{P} \in \mathbb{R}^{n\times n}$, $\mathbf{S_o} \in \mathbb{R}^{n\times m}$, $m > n$, is a 'fat' matrix. Though $\mathbf{S_o}$ in (29) can be considered as a square-root of $\mathbf{P}$, we prefer to keep the square-root as a triangular matrix of the dimension $n \times n$ for computational reasons. The transformation of $\mathbf{S_o}$ into a triangular matrix $\mathbf{S_n} \in \mathbb{R}^{n\times n}$ is performed by a triangularization algorithm, e.g., Gram-Schmidt based QR decomposition. When the matrix $\mathbf{S_o}^T$ is decomposed into an orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{m\times n}$ and an upper triangular matrix $\mathbf{R} \in \mathbb{R}^{n\times n}$ such that $\mathbf{S_o}^T = \mathbf{QR}$, we get

$$\mathbf{P} \;=\; \mathbf{S_o}\mathbf{S_o}^T = \mathbf{R}^T\mathbf{Q}^T\mathbf{Q}\mathbf{R} = \mathbf{R}^T\mathbf{R} = \mathbf{S_n}\mathbf{S_n}^T,$$

where the 'new' square-root of $\mathbf{P}$, $\mathbf{S_n} = \mathbf{R}^T$. In this paper, we denote this procedure as

$$\mathbf{S_n} \;=\; \textbf{Tria}(\mathrm{S_o}),$$

where $\mathbf{S_o}$ could be referred to as the 'old' square-root of $\mathbf{P}$. In what follows, we illustrate how this triangularization algorithm can be fitted into the two steps of a *square-root cubature Kalman filter* (SCKF), namely, the time-update and the measurement-update.

A. Time update

From (26), we see that the predicted error covariance $P_{k+(m+1)\delta|k}$ does not admit a simple expression as a sum of some squared matrices, because $\{Y_i\}$s are matrices of dimension $(n \times n)$ rather than vectors. This suggests that finding a square-root of the predicted error covariance is difficult in its plain form. However, this difficulty can be circumvented without loss of accuracy by replacing the factor $\mathbb{L}\mathbf{f}(\mathbf{x}_k)$ by $\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})$ in (14). Then, the predicted mean is given by

$$\hat{\mathbf{x}}_{k+\delta|k} \;=\; \int_{\mathbb{R}^n} \mathbf{J}(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k \tag{30}$$

whereas the error covariance is given by

$$\begin{aligned}\mathbf{P}_{k+\delta|k} \;=\; & \int_{\mathbb{R}^n} \mathbf{J}(\mathbf{x}_k)\mathbf{J}^T(\mathbf{x}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})d\mathbf{x}_k + \frac{\delta^3}{3}(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k}))(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k}))^T \\ & + \frac{\delta^2}{2}\Big[\sqrt{\mathbf{Q}}\big(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})\big)^T + \big(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})\big)\sqrt{\mathbf{Q}}^T\Big] - \hat{\mathbf{x}}_{k+\delta|k}\hat{\mathbf{x}}_{k+\delta|k}^T + \delta\mathbf{Q}. \end{aligned} \tag{31}$$

The integrals present in the predicted state (30) and its error covariance (31) can be approximated by the cubature rule. Moreover, a square-root factor of the error covariance can be constructed in a straightforward way (see the time update of the SCKF shown at the end of this section).

B. Measurement-update

Recall from the cubature Kalman filtering algorithm that the following three covariance matrices can also be expressed in a squared-matrix form:

$$\begin{aligned}
\mathbf{P}_{k+1|k} &= \mathscr{X}_{k+1|k}\mathscr{X}^{T}_{k+1|k} \\
\mathbf{P}_{zz,k+1|k} &= \mathscr{Z}_{k+1|k}\mathscr{Z}^{T}_{k+1|k} + \mathbf{S}^{T}_{R,k+1}\mathbf{S}_{R,k+1} \\
\mathbf{P}_{xz,k+1|k} &= \mathscr{X}_{k+1|k}\mathscr{Z}^{T}_{k+1|k},
\end{aligned}$$

where the weighted-centered (prior mean is subtracted off) matrices

$$\begin{aligned}
\mathscr{X}_{k+1|k} &= \frac{1}{\sqrt{2n}}[\mathbf{X}_{1,k+1|k} - \hat{\mathbf{x}}_{k+1|k} \quad X_{2,k+1|k} - \hat{\mathbf{x}}_{k+1|k} \ldots X_{2n,k+1|k} - \hat{\mathbf{x}}_{k+1|k}] \\
\mathscr{Z}_{k+1|k} &= \frac{1}{\sqrt{2n}}[Z_{1,k+1|k} - \hat{\mathbf{z}}_{k+1|k} \quad Z_{2,k+1|k} - \hat{\mathbf{z}}_{k+1|k} \ldots Z_{2n,k+1|k} - \hat{\mathbf{z}}_{k+1|k}].
\end{aligned}$$

Equivalently, the above three equations can be combined together and written in the squared-matrix form:

$$\begin{pmatrix} \mathbf{P}_{zz,k+1|k} & \mathbf{P}_{zx,k+1|k} \\ \mathbf{P}_{xz,k+1|k} & \mathbf{P}_{k+1|k} \end{pmatrix} = \begin{pmatrix} \mathscr{Z}_{k+1|k} & \mathbf{S}_{R,k+1} \\ \mathscr{X}_{k+1|k} & \mathbf{O} \end{pmatrix} \begin{pmatrix} \mathscr{Z}_{k+1|k} & \mathbf{S}_{R,k+1} \\ \mathscr{X}_{k+1|k} & \mathbf{O} \end{pmatrix}^{T}, \tag{32}$$

where $\mathbf{O} \in \mathbb{R}^{n_x \times n_z}$ is the zero matrix. Applying the triangularization algorithm on the square-root factor available on the right-hand side yields

$$\textbf{Tria}\begin{pmatrix} \mathscr{Z}_{k+1|k} & \mathbf{S}_{R,k+1} \\ \mathscr{X}_{k+1|k} & \mathbf{O} \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B} & \mathbf{C} \end{pmatrix}, \tag{33}$$

where $\mathbf{A} \in \mathbb{R}^{n_z \times n_z}$, and $\mathbf{C} \in \mathbb{R}^{n_x \times n_x}$ are lower-triangular matrices, and $\mathbf{B} \in \mathbb{R}^{n_x \times n_z}$. Hence, we rewrite (32) in a 'new' squared-matrix form as follows:

$$\begin{aligned} \begin{pmatrix} \mathbf{P}_{zz,k+1|k} & \mathbf{P}_{zx,k+1|k} \\ \mathbf{P}_{xz,k+1|k} & \mathbf{P}_{k+1|k} \end{pmatrix} &= \begin{pmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B} & \mathbf{C} \end{pmatrix} \begin{pmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B} & \mathbf{C} \end{pmatrix}^T \\ &= \begin{pmatrix} \mathbf{A}\mathbf{A}^T & \mathbf{A}\mathbf{B}^T \\ \mathbf{B}\mathbf{A}^T & \mathbf{B}\mathbf{B}^T + \mathbf{C}\mathbf{C}^T \end{pmatrix}. \end{aligned} \tag{34}$$

Recall also from Section III that the cubature Kalman gain is given by

$$\mathbf{W}_{k+1} = \mathbf{P}_{xz,k+1|k}\mathbf{P}^{-1}_{zz,K+1|k}. \tag{35}$$

Substituting the results obtained in (34) into (35) yields

$$\begin{aligned} \mathbf{W}_{k+1} &= \mathbf{B}\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1} \\ &= \mathbf{B}\mathbf{A}^{-1} \end{aligned} \tag{36}$$

Because $\mathbf{A}$ is a lower-triangular matrix, we may efficiently compute $\mathbf{W}_{k+1}$– Assume that the new symbol / represents the matrix right-divide operator; by writing $\mathbf{B}\mathbf{A}^{-1}$ to be $\mathbf{B}/\mathbf{A}$, we essentially apply the *forward substitution* algorithm to compute $\mathbf{W}_{k+1}$. We therefore write the posterior state estimate

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + (\mathbf{B}/\mathbf{A})(\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1|k}). \tag{37}$$

Let us now see how the posterior error covariance $\mathbf{P}_{k+1|k+1}$ can be written in a matrix-squared form. Recall from Section III that $\mathbf{P}_{k+1|k+1}$ is given by

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{W}_{k+1}\mathbf{P}_{zz,k+1|k}\mathbf{W}^T_{k+1}. \tag{38}$$

Substituting the results obtained in (34) into (38) yields

$$\begin{aligned} \mathbf{P}_{k+1|k+1} &= (\mathbf{B}\mathbf{B}^T + \mathbf{C}\mathbf{C}^T) - \mathbf{B}\mathbf{A}^{-1}(\mathbf{A}\mathbf{A}^T)(\mathbf{B}\mathbf{A}^{-1})^T \\ &= \mathbf{C}\mathbf{C}^T. \end{aligned}$$

Hence, the 'new' square-root factor of $\mathbf{P}_{k+1|k+1}$ is $\mathbf{C}$.

To sum up, given the weighted-centered matrices $\mathscr{X}_{k+1|k}$, and $\mathscr{Z}_{k+1|k}$, and the square-root factor of the measurement noise covariance $\mathbf{S}_{R,k+1}$, the heart of the measurement-update step of the SCKF resides in computing the matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ as shown by (33), using a triangualriation algorithm. Subsequently, we compute the updated state estimate $\hat{\mathbf{x}}_{k+1|k+1}$ as shown by (37) and the square-root of the corresponding posterior error covariance, which is simply given by $\mathbf{C}$.

Below, we summarize the square-root cubature Kalman filter (SCKF) writing the steps only when they differ from the CKF:

SCKF– Time Update

Initialize m to be zero.

1) Skip the factorization step (1) because the square-root of the error covariance $\mathbf{S}_{k+m\delta|k}$ is available. Compute steps (2)-(4).
2) Estimate the square-root factor of the predicted error covariance

$$\mathbf{S}_{k+(m+1)\delta|k} \quad = \quad \mathbf{Tria}\big([\mathscr{X}^{*}_{k+(m+1)\delta|k} \quad \sqrt{\delta}\big(\sqrt{\mathbf{Q}} + \frac{\delta}{2}\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k+m\delta|k})\big) \quad \sqrt{\frac{\delta^3}{12}}\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k+m\delta|k})]\big),$$

where the weighted centered matrix

$$\begin{aligned}\mathscr{X}^{*}_{k+(m+1)\delta|k} \quad = \quad & \frac{1}{\sqrt{2n}}[\boldsymbol{X}^{*}_{1,k+(m+1)\delta|k} - \hat{\mathbf{x}}_{k+(m+1)\delta|k} \quad \boldsymbol{X}^{*}_{2,k+(m+1)\delta|k} - \hat{\mathbf{x}}_{k+(m+1)\delta|k} \dots \\ & \boldsymbol{X}^{*}_{2n,k+(m+1)\delta|k} - \hat{\mathbf{x}}_{k+(m+1)\delta|k}].\end{aligned}$$

3) Increase m by one and repeat the above steps until m reaches M (that is, until time $(k+1)$).

SCKF– Measurement Update

1) Evaluate the cubature points (i=1,2,. . . 2n)

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

2) Evaluate the propagated cubature points (i=1,2,. . . 2n)

$$Z_{i,k+1|k} \quad = \quad \mathbf{h}(X_{i,k+1|k}, k+1).$$

3) Estimate the predicted measurement

$$\hat{\mathbf{z}}_{k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k+1|k}.$$

4) Construct the following two weighted-centered matrices:

$$\mathscr{X}_{k+1|k} = \frac{1}{\sqrt{2n}}[X_{1,k+1|k} - \hat{\mathbf{x}}_{k+1|k} \quad X_{2,k+1|k} - \hat{\mathbf{x}}_{k+1|k} \ldots X_{2n,k+1|k} - \hat{\mathbf{x}}_{k+1|k}].$$
$$\mathscr{Z}_{k+1|k} = \frac{1}{\sqrt{2n}}[Z_{1,k+1|k} - \hat{\mathbf{z}}_{k+1|k} \quad Z_{2,k+1|k} - \hat{\mathbf{z}}_{k+1|k} \ldots Z_{2n,k+1|k} - \hat{\mathbf{z}}_{k+1|k}].$$

5) Compute the matrices $\mathbf{A}, \mathbf{B}$, and $\mathbf{C}$ using the triangularization algorithm:

$$\textbf{Tria}\begin{pmatrix} \mathscr{Z}_{k+1|k} & \mathbf{S}_{R,k+1} \\ \mathscr{X}_{k+1|k} & \mathbf{O} \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B} & \mathbf{C} \end{pmatrix}.$$

6) Estimate the cubature Kalman gain

$$\mathbf{W}_{k+1} = \mathbf{B}/\mathbf{A}.$$

7) Estimate the updated state

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

8) The square-root factor of the corresponding error covariance is given by

$$\mathbf{S}_{k+1|k+1} = \mathbf{C}.$$

Remarks:

- A comparison of the measurement-update step of the existing SCKF derived in [1] with that of the new variant derived in this paper reveals that both algorithms approximately require a same computational cost of $(6n^3 + 10n^2m)$ flop counts (or simply flops) per update cycle, where n and m denote the dimensions of the state vector and the measurement vector, respectively. In computing these costs, note that we do not account for flops associated with problem-specific function evaluations, which are common to both algorithms (see Appendix A). However, the derivation of the new variant is more elegant– it applies the triangularization algorithm to the array of matrices only once; in so doing, the

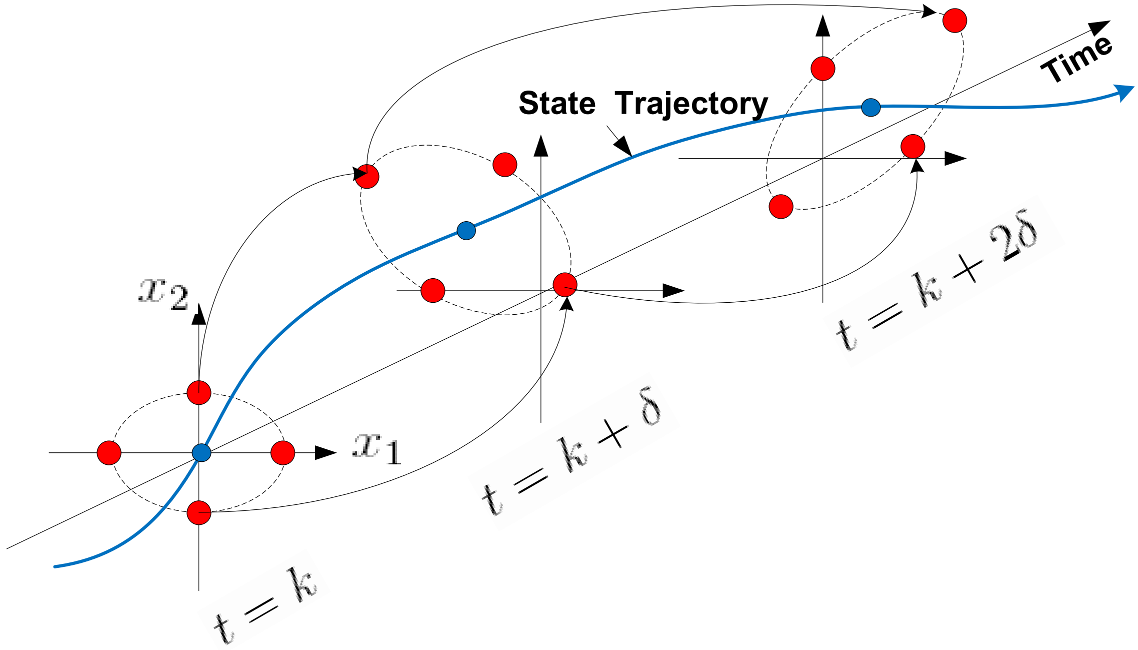


Fig. 1. Time-update of the CR-SCKF illustrating how the cubature points in the two-dimensional state-space are propagated in the intermediate steps of the measurement interval $[k, k+2\delta]$; The circles represent cubature points; the new cubature point set at time $t = k+\delta$ is computed by simply propagating the old cubature point set at time k through the truncated Itô-Taylor expansion of the continuous-time process equation.

new variant avoids an explicit computation of the cross-covariance matrix and a repeated use of the forward substitution algorithm in computing the cubature Kalman gain.

- It is also interesting to note that the new variant closely resembles the square-root formulation of the linear Kalman filter proposed by Kaminski *et al.* [13]. A key exception is that in the new SCKF, various covariances are expressed in the form of the outer products of weighted-centered matrices.

V. Cost-Reduced Square-root Cubature Kalman Filtering

In this section, we derive a *Cost-Reduced Square-root Cubature Kalman Filter* (CR-SCKF) that drastically reduces the computational cost of the SCKF. The only difference between the SCKF and the CR-SCKF lies in propagating second-order statistics in the time-update. Assuming the use of an M-step Itô-Taylor approximation scheme, the CR-SCKF performs the following:

CR-SCKF– Time Update: From time k to $(k+1)$

1) Given the cubature point set representing the posterior density at time k, recursively propagate this point set through the noise-free process model (15) up to M steps forward (see Fig. 1).
2) Compute the predicted state using the set of cubature pints available at time $(k+1)$.

3) Compute the square-root of the predicted state error covariance after adding a discrete-time process noise covariance

$$\mathbf{Q}_d \;=\; \Delta\mathbf{Q} + \frac{\Delta^3}{3}(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k}))(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k}))^T + \frac{\Delta^2}{2}\Big[\sqrt{\mathbf{Q}}\big(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})\big)^T + \big(\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})\big)\sqrt{\mathbf{Q}}^T\Big]$$

to the filter-estimated error covariance, where Δ is the measurement sampling time, and the process noise gain matrix $\mathbf{Q}$ is defined in Section I. To be specific, the square-root covariance is given by

$$\mathbf{S}_{k+1|k} \;=\; \mathbf{Tria}([\mathscr{X}^*_{k+M\delta|k} \quad \sqrt{\Delta}\big(\sqrt{\mathbf{Q}} + \frac{\Delta}{2}\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})\big) \quad \sqrt{\frac{\Delta^3}{12}}\mathbb{L}\mathbf{f}(\hat{\mathbf{x}}_{k|k})]),$$

where the weighted centered matrix

$$\mathscr{X}^*_{k+M\delta|k} \;=\; \frac{1}{\sqrt{2n}}[\mathbf{X}^*_{1,k+M\delta|k} - \hat{\mathbf{x}}_{k+1|k} \quad \mathbf{X}^*_{2,k+M\delta|k} - \hat{\mathbf{x}}_{k+1|k} \ldots \mathbf{X}^*_{2n,k+M\delta|k} - \hat{\mathbf{x}}_{k+1|k}].$$

The above mentioned steps are mathematically equivalent to the following statement:

Given $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$ *at time* k, *compute the second-order statistics of the composite function* $\underbrace{\mathbf{J}(\mathbf{J}(\mathbf{J}\ldots\mathbf{J}(\mathbf{x}_k)))}_{M \text{ times}}$ *using the cubature rule and fudge the estimated covariance with noise covariances.*

In so doing, we perform only one time-update instead of M time-updates during the interval of two consecutive measurement instants. Let us now focus on the amount of computational saving achieved by the CR-SCKF. Under the reasonable assumption that the dimension of the state-vector is higher than that of the measurement-vector, the time-update of the SCKF is computationally more expensive than the measurement-update (see Appendix A). To get a handle on the computational cost of the time-update of the SCKF, let us assume that the state vector dimension n is 10 and an M-step Itô-Taylor scheme is used to approximate the continuous-time process equation. Because the SCKF performs M time-updates between measurement instants, it requires $8M$ kflops $(= 8 \times n^3M)$. On the other hand, the CR-SCKF performs only one time-update and thus requires 8 kflops $(= 8 \times n^3)$. For the example of $M = 64$, the SCKF costs nearly 64 times the CR-SCKF. Note that in computing these costs, we ignore the cost associated with function-evaluations because it is problem-dependant. However, the function-evaluations cost is almost the same for both the SCKF and the the CR-SCKF.

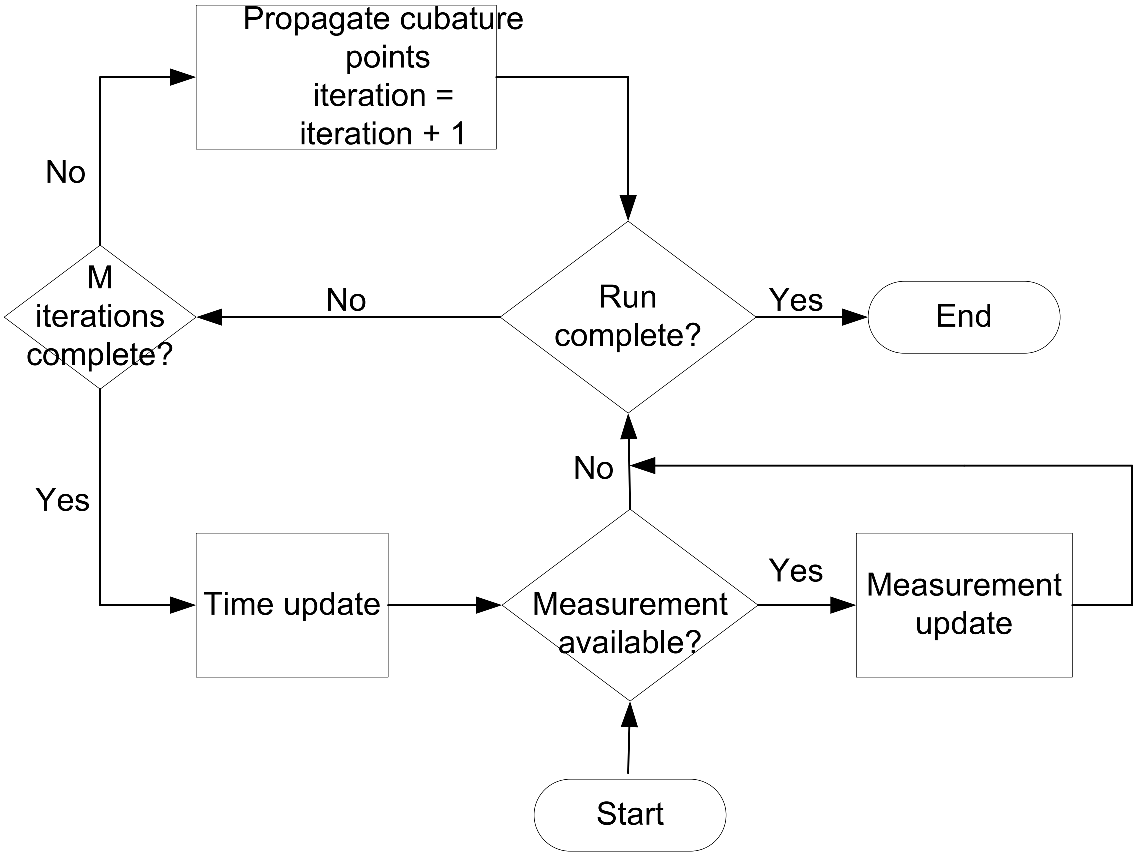


Fig. 2. Flowchart of the continuous-discrete CR-SCKF.

To be more specific in terms of computation time, let us assume that the SCKF and the CR-SCKF are committed to an Intel Pentium Core Duo processor with a cycle speed of 2Gflops/s. In this case, the computation times of the time-updates of the SCKF and the CR-SCKF are 256 micro seconds ($=$ 512kflops$/2$GHz), and 4 micro seconds ($=$ 8kflops$/2$GHz), respectively. We thus save a computation time of 0.25 milliseconds in each recursion cycle. This could be a monumental time reduction for a radar operating at a measurement sampling frequency of hundreds of kHz. Indeed, for an increased number of steps, that is $M \gg 64$, the use of the CR-SCKF will save a considerable computation time.

A block diagram/flowchart of the CR-SCKF is shown in Fig. 2. Note that the number of iterations between two consecutive measurements, M, does not need to be fixed. For aperiodically sampled data, we perform the time update and then the measurement update as soon as a new measurement is received. Generally speaking, the state estimates can also be made available at points on as fine a division of the time line as we desire, independently of the measurement sampling period, which may not be up to us to choose. This can be quite useful in a situation where a slowly scanning radar measures range and/or azimuth in substantially longer time intervals.

VI. Application to Tracking a Ballistic Target on Reentry

Tracking ballistic targets is one of the most extensively studied applications considered by the aerospace engineering community (see the survey paper [17]). The goal of this research is to track, intercept, and destroy ballistic targets before they hit the ground. The flight of a ballistic target, from launch to impact, consists of three phases: the boost phase, the coast phase and the reentry phase. In this paper, we limit our focus to tracking a ballistic target on reentry.

Reentry Scenario. When a ballistic target reenters the atmosphere after having traveled a long distance, its speed is high and the remaining time to ground impact is relatively short. In the experiment, we consider a ballistic target falling vertically as shown in Fig. 3. In the reentry phase, two types of forces are in effect– the most dominant is drag, which is a function of speed and has a substantial nonlinear variation in altitude; the second force is due to gravity, which accelerates the target toward the center of the earth. This target tracking problem is highly difficult because the target's dynamics change rapidly. Under the influence of drag and gravity acting on the target, the following differential equation governs its motion [4], [9]:

$$\begin{aligned} \dot{x}_1 &= -x_2 \\ \dot{x}_2 &= \underbrace{-\rho(x_1)x_2^2x_3}_{\text{drag}}+g \\ \dot{x_3} &= 0, \end{aligned}$$

where

- x_1 is altitude;
- x_2 is velocity;
- x_3 is a constant ballistic coefficient that depends on the target's mass, shape and cross-sectional area;
- $\rho(x_1)$ is air density and modeled as an exponentially decaying function of altitude x_1:

  $$\rho(x_1) = \rho_0 \text{exp}(-\gamma x_1),$$

  where the proportionality constant $\rho_0 = 1.754$ and $\gamma = 1.49 \times 10^{-4}$; and
- g is gravity ($g = 9.8\text{ms}^{-2}$).

By choosing the state vector $\mathbf{x} = [x_1\ x_2\ x_3]^T$, the process equation in continuous time t can be expressed

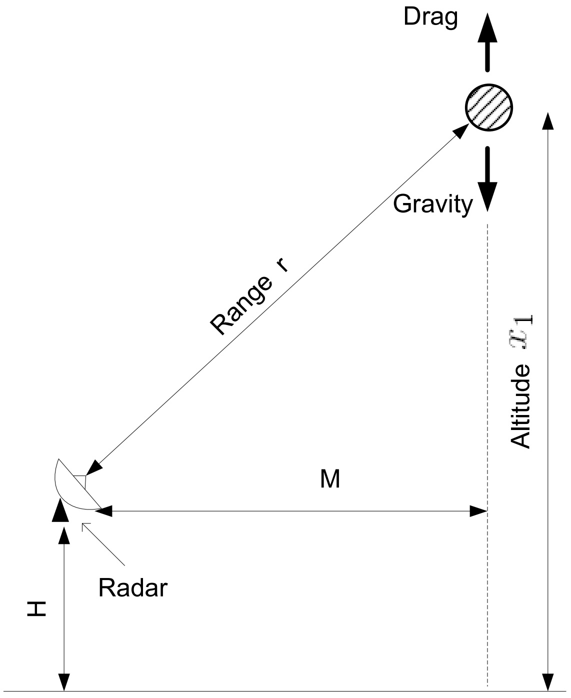


Fig. 3. Geometry of the ballistic target tracking scenario.

by

$$\dot{\mathbf{x}}_t \quad = \quad \mathbf{f}(\mathbf{x}_t), \tag{39}$$

where $\mathbf{f}(\mathbf{x}_t) = [-x_2 \; -\rho(x_1)x_2^2 x_3 + g \; 0]^T$. In order to account for imperfections in the process model (e.g., lift force, small variations in the ballistic coefficient, and spinning motion), we add zero-mean Gaussian process noise to (40) and obtain a new process equation:

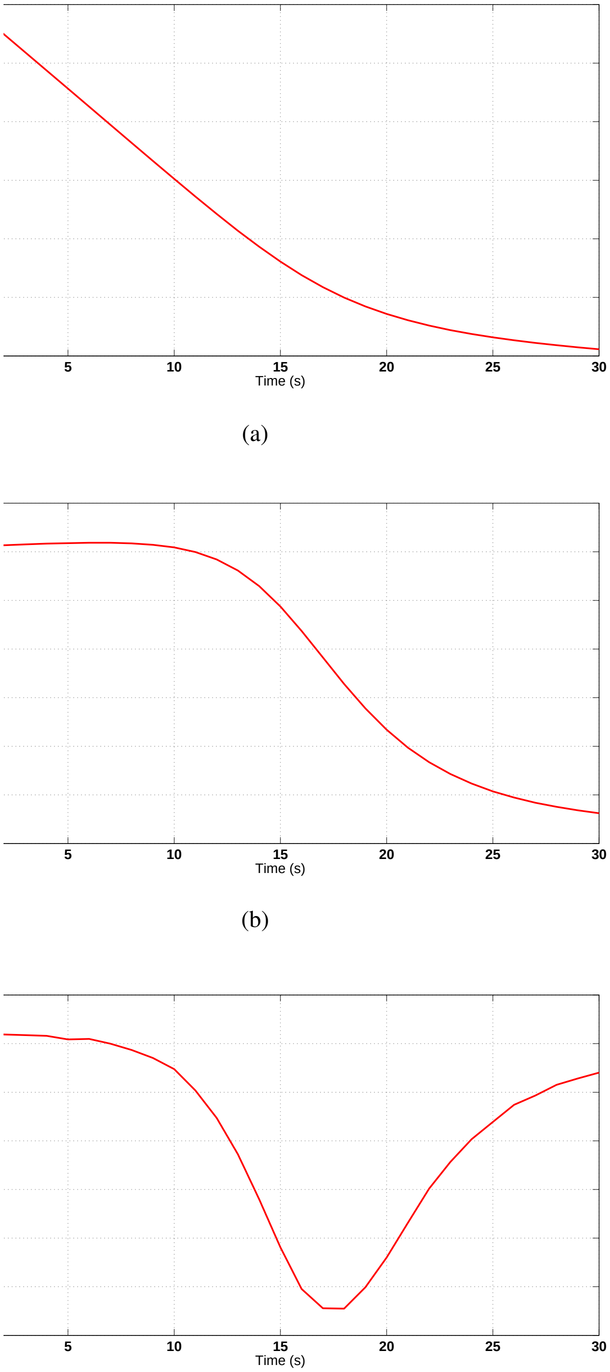
$$\dot{\mathbf{x}}_t \quad = \quad \mathbf{f}(\mathbf{x}_t) + \sqrt{\mathbf{Q}}\mathbf{w}_t, \tag{40}$$

where the gain matrix $\mathbf{Q} = \mathrm{diag}([\sigma_1^2 \; \sigma_2^2 \; \sigma_3^2])$.

For the experiment at hand, a radar was located at $(0, H)$ and equipped to measure the range r at a measurement time interval of Δ. Hence, the measurement equation is given by

$$r_k \quad = \quad \sqrt{M^2 + (\mathbf{x}_k[1] - H)^2} + v_k, \tag{41}$$

where the measurement noise $v_k \sim \mathcal{N}(0, R)$; and M is the horizontal distance (see Fig. 3).



(a)

(b)

(c)

Fig. 4. An example trajectory of the ballistic object Vs. time.

Experimental Data.

$$H = 30\text{m}$$

$$M = 30\text{km}$$

$$\mathbf{Q} = \text{diag}([(50\text{m})^2 s^{-1} \ (10\text{ms}^{-1})^2\text{s}^{-1} \ 0\text{s}^{-1}])$$

$$R = (30\text{m})^2$$

$$\Delta = 1\text{s}$$

The true initial state was assumed to be

$$\mathbf{x}_0 \;=\; [61\text{km } 3048\text{m/s } 2.56 \times 10^{-4}]^T$$

For the purpose of generating the ballistic target's trajectory, we used the Itô-Taylor expansion order 1.5 with 64 time-steps between two consecutive measurements. For this problem, the matrices $\mathbb{L}\mathbf{f}$ and $\mathbb{L}^0\mathbf{f}$ are given by

$$\begin{aligned}\mathbb{L}\mathbf{f} &= \begin{pmatrix} 0 & -\sigma_2 & 0 \\ \rho(x_1)x_2^2x_3\gamma\sigma_1 & -2\rho(x_1)x_2x_3\sigma_2 & -\rho(x_1)x_2^2\sigma_3 \\ 0 & 0 & 0 \end{pmatrix} \\ \mathbb{L}^0\mathbf{f} &= \begin{pmatrix} \rho(x_1)x_2^2x_3 - g \\ \rho(x_1)x_2x_3\big(2\rho(x_1)x_2^2x_3 - \gamma x_2^2 - 2g\big) \\ 0 \end{pmatrix}.\end{aligned}$$

An example of kinematic parameters for a representative target trajectory is shown in Fig. 4. As can be seen from this figure, the velocity remains constant in the first 10 seconds of the ballistic flight on reentry, followed by a sharp drop due to deceleration caused by the air resistance at lower altitudes. To track the ballistic target, the following Bayesian filters were employed:

- The extended Kalman filter (EKF)
- The square-root cubature Kalman filter using the Itô-Taylor expansion of order 0.5 (SCKF (IT-0.5))
- The square-root cubature Kalman filter using the Itô-Taylor expansion of order 1.5 (SCKF (IT-1.5))
- The cost reduced square-root cubature Kalman filter using the Itô-Taylor expansion of order 1.5 (CR-SCKF (IT-1.5))

To initialize the Bayesian filters, the initial state density was assumed to be Gaussian with the mean $\hat{\mathbf{x}}_{0|0}$ and covariance $\mathbf{P}_{0|0}$:

$$\begin{aligned}\hat{\mathbf{x}}_{0|0} &= [62\text{km } 3100\text{m/s } 10^{-5}]^T \\ \mathbf{P}_{0|0} &= \text{diag}([(1\text{km})^2 \; (100\text{m/s})^2 \; 10^{-4}]).\end{aligned}$$

For the purpose of implementing the time-update of the cubature filter, we used the Itô-Taylor expansion of 4 time-steps between any two consecutive measurement instants.

Performance metric. For this problem, we used the average absolute error as a performance measure.

The average absolute error in the i-th state component at time k is defined by

$$\epsilon_k[i] \;=\; \frac{1}{N}\sum_{n=1}^{N} |\mathbf{x}_k^{(n)}[i] - \hat{\mathbf{x}}_{k|k}^{(n)}[i]|,$$

where N is the total number of Monte Carlo simulation runs. Each trajectory was simulated for 30s; and a total of $N = 50$ independent Monte Carlo runs was made.

Observations and Conclusion.

- The average absolute errors in altitude, velocity and ballistic coefficient are shown in Figs. 5(a)-5(c). As expected, both the SCKF (IT-1.5) and the CR-SCKF (IT-1.5) perform almost equally, and markedly outperform the EKF and the SCKF (IT-0.5).
- Though the computational costs of all four Bayesian filters grow cubically with the state-vector dimension, their scaling factors are different. In terms of relative computational time, the EKF takes the shortest time and the SCKF (IT-1.5) takes the longest time; The CR-SCKF (IT-1.5) and the SCKF (IT-0.5) take the same time approximately.

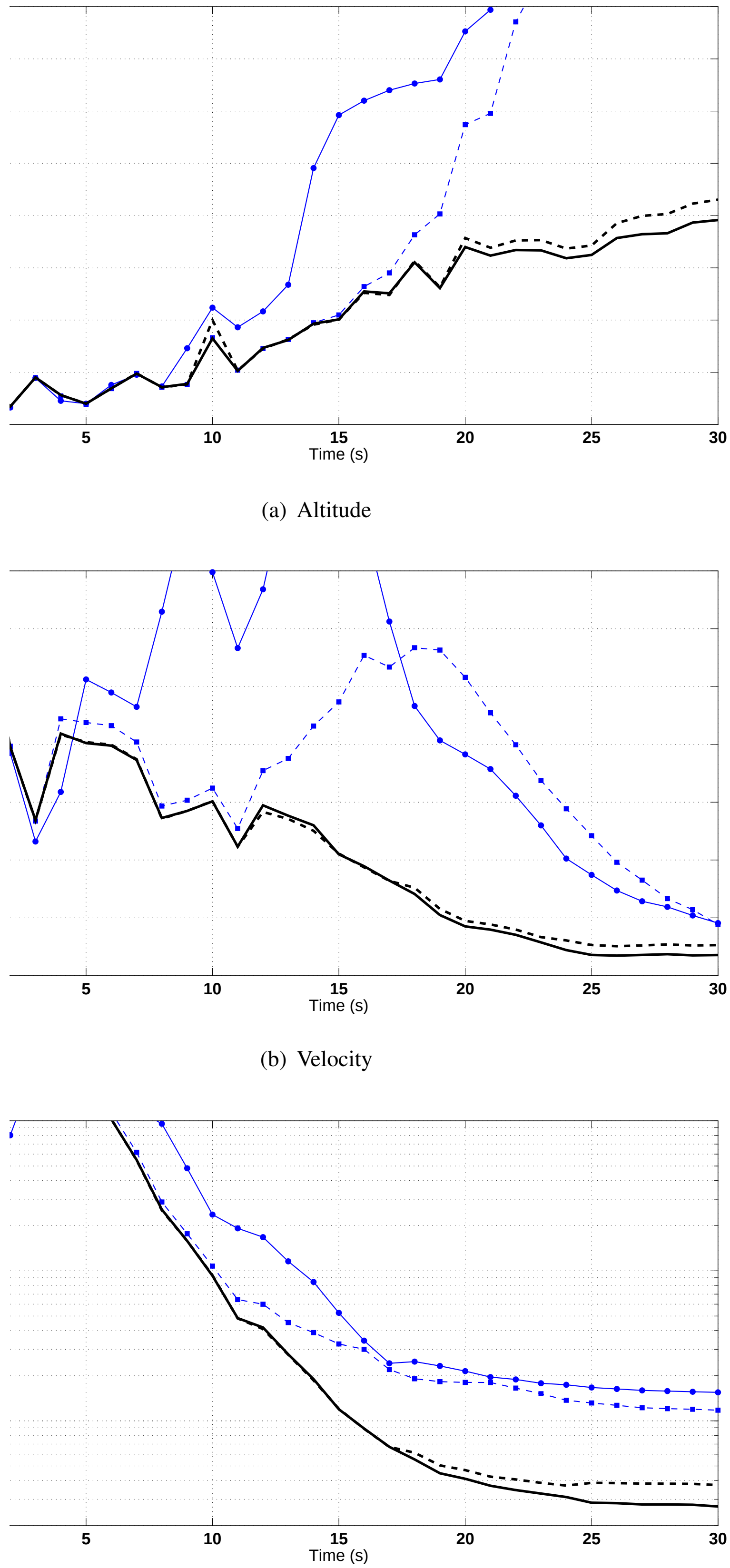
To sum up, based on the computational complexity-accuracy tradeoff, we may say that the CR-SCKF (IT-1.5) is the best choice for this considered tracking problem.

VII. Concluding Remarks

In this paper, we have developed a square-root cubature Kalman filter (SCKF) for continuous-discrete stochastic nonlinear systems. In the time-update step of the SCKF, we have employed two different numerical integration tools:

- In the *time domain*, we discretize the continuous-time state process using the *Itô-Taylor expansion of order 1.5*. This step transforms a continuous-discrete time filtering problem into a familiar discrete-time filtering problem.
- Subsequently, in the *state-space domain*, we compute the second-order information of the predictive density using the *third-degree cubature rule* that numerically integrates the time-discretized process equation.

Note that one of the main reasons for why we do not choose an Itô-Taylor expansion of higher order (higher than 1.5) is that the conditional density of (14) given $\mathbf{x}_k$ is no longer Gaussian. The higher-order expansion brings in squared Gaussian terms, which make the subsequent computations of second-order statistics extremely difficult. To markedly reduce the computational cost of the SCKF, we have also derived a cost-reduced SCKF. We have demonstrated that the SCKF and its cost reduced variant perform equally well and outperform the existing solutions in tracking a ballistic target on reentry.



(a) Altitude

(b) Velocity

(c) Ballistic Coefficient

Fig. 5. Comparison of Bayesian filters' performances averaged over 50 runs (solid line with circles- EKF, dotted line with squares- SCKF (IT-0.5), solid line- SCKF (IT-1.5), dotted line- CR-SCKF (IT-1.5))

REFERENCES

[1] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Automatic Cont.*, vol. 54, pp. 1254-1269, June 2009.

[2] I. Arasaratnam, "Cubature Kalman Filtering: Theory & Applications," Ph.D. Thesis, Department of Electrical & Computer Eng., McMaster University, April 2009.

[3] K. J. Aström, *Introduction to stochastic control theory*, NY: Academic Press, 1970.

[4] M. Athans, R. P. Wishner and A. Bertolini, "Suboptimal state estimation for continuous-time nonlinear systems from discrete noise measurements," *IEEE Trans. Automatic Cont.*, vol. 13, pp. 504-514, 1968.

[5] Y. Bar Shalom, X. R. Li and T. Kirubarajan, *Estimation with applications to tracking and navigation,* NY: Wiley & Sons, 2001.

[6] V. E. Beneš, "Exact finite-dimensional filters with certain diffusion nonlinear drift," *Stochastics*, vol. 5, pp. 65-92, 1981.

[7] R. Cools, "Constructing cubature formulas: the science behind the art," *Acta Numerica 6*, pp. $1-54$, 1997.

[8] F. Daum, "Exact finite-dimensional nonlinear filters," *IEEE Trans. Autom. Contr.,* vol. 31, pp. $616-622$, 1986.

[9] A. Gelb, ed., *Applied Optimal Estimation*, MA: MIT press, 1974.

[10] M. Grewal and A. Andrews, *Kalman filtering: Theory and practice using Matlab*, 2nd ed., NY: Wiley, 2001.

[11] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Trans. Automat. Control*, vol. 45, no. 5, pp. 910-927, May 2000.

[12] A. H. Jazwinski, *Stochastic processes and filtering theory*, NY: Academic, 1970.

[13] P. Kaminski, A. Bryson and S. Schmidt, "Discrete sqaure root filtering: A survey of current techniques," *IEEE Trans. Automat. control*, vol. ac-16, Dec. 1971.

[14] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory" *Trans. ASME, J. Basic Eng.*, vol. 83, pp. $95-108$, Mar. 1961.

[15] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations,* Springer: Berlin, 1999.

[16] H. J. Kushner and A. S. Budhiraja, "A nonlinear filtering algorithm based on an approximation of the conditional distribution," *IEEE Trans. Autom. Control*, vol. 45, no. 3, Mar. 2000.

[17] X. R. Li and V. P. Jilkov, "A survey of manuevering target tracking: Part II: Ballistic target models," *Proc. SPIE*, vol 4473, pp. 559-581, 2001.

[18] B. Øksendal, *Stochastic Differential Equations: An Introduction with Applications*, NY: Springer, 2003.

[19] A. H. Stroud, *Gaussian quadrature formulas*, NJ: Prentice Hall, 1966.

APPENDIX A

COMPUTATIONAL COST OF THE CR-SCKF

Below, we summarize the number of *flops* required by each step of the time and measurement updates of the CR-SCKF algorithm:

Time Update			
Steps	add/substract	multiply/divide	square root
1. $X_{i,k\|k} = \mathbf{S}_{k\|k}\xi_i + \hat{\mathbf{x}}_{k\|k} \quad (i = 1, \dots 2n)$	$n^3 + n^2$	$n^3 + n^2$	$-$
2. $\hat{\mathbf{x}}_{k+1\|k} = \frac{1}{2n}\sum_{i=1}^{2n} X^*_{i,k+1\|k}$	$2n^2 - n$	n	$-$
3. $\mathscr{X}^*_{k+1\|k} = \frac{1}{\sqrt{2n}}[\mathbf{X}^*_{1,k+1\|k} - \dots - \hat{\mathbf{x}}_{k+1\|k}]$	$2n^2$	$2n^2$	$-$
4. $\mathbf{S}_{k+1\|k} = \textbf{Tria}([\mathscr{X}^*_{k+1\|k} \quad \dots])$	$3n^3 - \frac{1}{2}n^2 - \frac{1}{2}n$	$3n^3 + 3n^2$	n
Total cost	$8n^3 + \frac{21}{2}n^2 + \frac{1}{2}n$		

Measurement Update			
Steps	add/substract	multiply/divide	square root
1. $X_{i,k\|k} = \mathbf{S}_{k\|k}\xi_i + \hat{\mathbf{x}}_{k\|k} \quad (i = 1, \dots 2n)$	$n^3 + n^2$	$n^3 + n^2$	$-$
2. $\hat{\mathbf{z}}_{k+1\|k} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k+1\|k}$	$(2n-1)m$	$-$	$-$
3. $\mathscr{X}_{k+1\|k} = \frac{1}{\sqrt{2n}}[\mathbf{X}_{1,k+1\|k} - \dots - \hat{\mathbf{x}}_{k+1\|k}]$	$2n^2$	$2n^2$	$-$
$\mathscr{Z}_{k+1\|k} = \frac{1}{\sqrt{2n}}[Z_{1,k+1\|k} - \dots - \hat{\mathbf{z}}_{k+1\|k}]$	$2nm$	$2nm$	$-$
4. $\textbf{Tria}\begin{pmatrix} \mathscr{Z}_{k+1\|k} & \mathbf{S}_{R,k+1} \\ \mathscr{X}_{k+1\|k} & \mathbf{O} \end{pmatrix}$	$2n^3 + (5m - \frac{1}{2})n^2$ $+(4m^2 - m - \frac{1}{2})n$ $+m^3 - \frac{1}{2}m^2 - \frac{1}{2}m$	$2n^3 + (5m + 2)n^2$ $+(4m^2 + 3m)n$ $+m^3 + m^2$	$n + m$
5. $\mathbf{W}_{k+1} = \mathbf{B}/\mathbf{A}$	$\frac{1}{2}m(m-1)n + \frac{1}{6}m^3$ $-\frac{1}{2}m^2 + \frac{1}{3}m$	$\frac{1}{2}m(m+1)n + \frac{1}{6}m^3$ $+\frac{1}{2}m^2 - \frac{2}{3}m$	m
6. $\hat{\mathbf{x}}_{k+1\|k+1} = \hat{\mathbf{x}}_{k+1\|k} + \mathbf{W}_{k+1} \dots$	$m(n+1)$	nm	$-$
Total cost	$6n^3 + (10m + \frac{15}{2})n^2 + (9m^2 + 10m + \frac{1}{2})n$ $+\frac{7}{3}m^3 + \frac{1}{2}m^2 + \frac{7}{6}m$		