A Brief Tutorial On Recursive Estimation With Examples From Intelligent Vehicle Applications (Part III): Handling Nonlinear Estimation Problems And The Unscented Kalman Filter

Hao Li

A Brief Tutorial On Recursive Estimation With Examples From Intelligent Vehicle Applications (Part III): Handling Nonlinear Estimation Problems And The Unscented Kalman Filter

Hao Li

Abstract

This is the third article of the series "*A brief tutorial on recursive estimation*". In this article, we focus on the problem of how to handle model nonlinearity (concerning both the system model and the measurement model) in recursive estimation. We will review an important variant of the KF, i.e. the unscented Kalman filter (UKF), which is dedicated to nonlinear estimation problems.

Keywords: recursive estimation, model nonlinearity, unscented Kalman filter (UKF), intelligent vehicles

1 Introduction

This is the third article of the series "*A brief tutorial on recursive estimation*" initiated by [1]. In this article, we focus on the problem of how to handle model **nonlinearity** (concerning both the system model and the measurement model) in recursive estimation.

A question arises naturally: what is *nonlinearity*? It means the opposite of *linearity*. Then what is linearity? Simply speaking, **linearity** refers to a kind of relationship between two variables where one variable changes in proportion to the change of the other. For example, in equations $y = 3.14x$ and $y = x+2.78$, the relationship between x and y is linear—it is worth noting that the meaning of the "variable" mentioned here is rather general; it can refer to a vector, a matrix, a polynomial, a functional etc—Any relationship that can not be expressed as a linear relationship is *nonlinear*.

On can often encounter the concept of linearity in literature. Why does linearity attract so much attention from researchers? Because linearity possesses many desirable characteristics which make problems with linear relationships comparatively easy to handle. For estimation problems, an important characteristic of linearity is: a Gaussian distribution, after a linear transformation, is still a Gaussian distribution; besides, we can conveniently and precisely compute the Gaussian distribution after the transformation from that before the transformation. This desirable characteristic is a basis for the famous Kalman filter [2].

For many estimation problems, the system model and the measurement model are nonlinear. In the context of intelligent vehicles, the application of vehicle localization in a 2D case as presented in [1] is an example where the system model is nonlinear. To handle the model nonlinearity, be it the nonlinearity of the system model or of the measurement model, one may linearize the system model and the measurement model locally and then apply the original Kalman filter to the locally linearized system. This is the basic spirit of the extended Kalman filter (EKF) [3].

However, the local linearization strategy in the EKF has its drawbacks. Here, we would highlight one point: it may result in **inconsistent** statistics [4]—**Consistency** means that for an estimate the estimated covariance matrix is no smaller than the true covariance of the estimated state vector—We give a simple example here:

Consider the one-dimensional function $f(x) = x^2$ and let the system model be $x_t = f(x_{t-1})$; suppose last estimate is $\hat{x}_{t-1} = 0$ with certain variance (i.e. covariance for multivariate state cases) $\hat{\Sigma}_{t-1} > 0$. We perform the prediction step as in the EKF, with local linearization of f at $\hat{x}_{t-1} = 0$, and we have $\bar{x}_t = f(\hat{x}_{t-1}) = 0$ and $\bar{\Sigma}_t = f'(\hat{x}_{t-1})\hat{\Sigma}_{t-1}f'(\hat{x}_{t-1}) = 0$. Attention! The predicted variance $\bar{\Sigma}_t$ becomes zero! This predicted variance is apparently smaller than the true variance of x_t and hence the prediction result is inconsistent—It is worth noting that this kind of inconsistency caused by local linearization of the system model is essentially different from the inconsistency caused by inappropriate fusion of correlated data: the former kind of inconsistency can be regarded as being due to a kind of model mismatch, whereas the latter kind of inconsistency is due to the fusion itself; one can refer to [5] [6] [7] for data fusion methods that handle the latter kind of inconsistency.

In order to handle inconsistent statistics that harass the EKF in nonlinear estimation, Julier and Uhlmann proposed in [4] [8] a useful variant of the KF, i.e. the unscented Kalman filter (UKF), which plays an important role in the KF family since its birth. In this article, we review the UKF, explain its idea of handling nonlinear factors in the estimation, and demonstrate its advantage over the EKF with an example of vehicle localization.

2 The Unscented Kalman Filter (UKF)

2.1 Review of the Kalman filter (KF)

We have already reviewed in [1] formulas of the KF and explained its essence from "information" perspective—the KF, in essence, is a fusion method that forms the fusion estimate by a **linear weighted combination** of source estimates—Here, we first review again the formulas of the KF that we have already reviewed and then review another formalism of the KF. The KF can be given as (see [1]):
Prediction:

$$\begin{aligned}\bar{\mathbf{x}}_t &= \mathbf{A}\hat{\mathbf{x}}_{t-1} + \mathbf{B}\hat{\mathbf{u}}_t \\ \bar{\boldsymbol{\Sigma}}_t &= \mathbf{A}\hat{\boldsymbol{\Sigma}}_{t-1}\mathbf{A}^T + \mathbf{B}\hat{\boldsymbol{\Sigma}}_{\mathbf{u}}\mathbf{B}^T + \boldsymbol{\Sigma}_\epsilon\end{aligned} \tag{1}$$

Update:

$$\begin{aligned}\mathbf{K} &= \bar{\boldsymbol{\Sigma}}_t\mathbf{H}^T(\mathbf{H}\bar{\boldsymbol{\Sigma}}_t\mathbf{H}^T + \boldsymbol{\Sigma}_\gamma)^{-1} \\ \hat{\mathbf{x}}_t &= \bar{\mathbf{x}}_t + \mathbf{K}(\mathbf{z}_t - \mathbf{H}\bar{\mathbf{x}}_t) \\ \hat{\boldsymbol{\Sigma}}_t &= (\mathbf{I} - \mathbf{K}\mathbf{H})\bar{\boldsymbol{\Sigma}}_t\end{aligned} \tag{2}$$

2.1.1 Concepts related to the KF

We review several concepts related to the formulas (1) and (2) in the KF.

Prediction:
Predicted (*a priori*) state estimate: $\bar{\mathbf{x}}_t$ in (1).
Predicted (*a priori*) covariance estimate: $\bar{\boldsymbol{\Sigma}}_t$ in (1).

Update:

Predicted (*a priori*) measurement: the expectation (or mean) of the measurement predicted with *a priori* state distribution according to the measurement model; denoted as $\hat{\mathbf{z}}_t$.

Innovation or measurement residual: the difference between the measurement $\mathbf{z}_t$ and the predicted measurement; denoted as $\hat{\mathbf{y}}_t$ i.e. $\hat{\mathbf{y}}_t = \mathbf{z}_t - \hat{\mathbf{z}}_t$.

Innovation covariance: the covariance of the innovation; denoted as $\mathbf{\Sigma}_{yy}$.

(State-measurement) cross covariance: the cross covariance between the predicted state estimate $\bar{\mathbf{x}}_t$ and the predicted measurement $\hat{\mathbf{z}}_t$; denoted as $\mathbf{\Sigma}_{xz}$.

Kalman gain: $\mathbf{K}$ in (2).

Updated (*a posteriori*) state estimate: $\hat{\mathbf{x}}_t$ in (2).

Updated (*a posteriori*) covariance estimate: $\hat{\mathbf{\Sigma}}_t$ in (2).

2.1.2 Another formalism of the KF

It seems that above concepts "predicted measurement", "innovation", "innovation covariance", and "cross covariance" are not related to the KF. To understand their roles in the KF, we have to note following equations (valid for cases of a linear measurement model).

Given a linear measurement model with the observation matrix $\mathbf{H}$, the predicted measurement is given as $\hat{\mathbf{z}}_t = \mathbf{H}\bar{\mathbf{x}}_t$. Then we have the innovation as $\hat{\mathbf{y}}_t = \mathbf{z}_t - \mathbf{H}\bar{\mathbf{x}}_t$ and derive the innovation covariance as:

$$\begin{aligned}
\mathbf{\Sigma}_{yy} &= E[(\Delta\mathbf{z}_t - \Delta\hat{\mathbf{z}}_t)(\Delta\mathbf{z}_t - \Delta\hat{\mathbf{z}}_t)^T] \\
&= E[(\Delta\mathbf{z}_t - \mathbf{H}\Delta\bar{\mathbf{x}}_t)(\Delta\mathbf{z}_t - \mathbf{H}\Delta\bar{\mathbf{x}}_t)^T] \\
&= E[\Delta\mathbf{z}_t\Delta\mathbf{z}_t^T] + \mathbf{H}E[\Delta\bar{\mathbf{x}}_t\Delta\bar{\mathbf{x}}_t^T]\mathbf{H}^T \\
&= \mathbf{\Sigma}_\gamma + \mathbf{H}\bar{\mathbf{\Sigma}}_t\mathbf{H}^T
\end{aligned}$$

In above derivation, the notation Δ represents "error" (a random variable). For example, $\Delta\hat{x}$ means $\chi - \hat{x}$, where χ represents the random variable associated with the estimate $\hat{x}$. We can derive the cross covariance as:

$$\begin{aligned}
\mathbf{\Sigma}_{xz} &= E[\Delta\bar{\mathbf{x}}_t\Delta\hat{\mathbf{z}}_t^T] \\
&= E[\Delta\bar{\mathbf{x}}_t\Delta\bar{\mathbf{x}}_t^T\mathbf{H}^T] \\
&= \bar{\mathbf{\Sigma}}_t\mathbf{H}^T
\end{aligned}$$

Besides, we have the following relationship:

$$\begin{aligned}\mathbf{KH}\bar{\mathbf{\Sigma}}_t &= \mathbf{K\Sigma}_{yy}\mathbf{\Sigma}_{yy}^{-1}\mathbf{H}\bar{\mathbf{\Sigma}}_t \\ &= \mathbf{K\Sigma}_{yy}\mathbf{K}^T\end{aligned}$$

With above equations, we can reformulate the update procedures (2) in the KF as in (3):

$$\begin{aligned}\mathbf{K} &= \mathbf{\Sigma}_{xz}\mathbf{\Sigma}_{yy}^{-1} \\ \hat{\mathbf{y}}_t &= \mathbf{z}_t - \hat{\mathbf{z}}_t \\ \hat{\mathbf{x}}_t &= \bar{\mathbf{x}}_t + \mathbf{K}\hat{\mathbf{y}}_t \\ \hat{\mathbf{\Sigma}}_t &= \bar{\mathbf{\Sigma}}_t - \mathbf{K\Sigma}_{yy}\mathbf{K}^T\end{aligned} \tag{3}$$

From (3) we can see how the concepts "predicted measurement", "innovation", "innovation covariance", and "cross covariance" are related to the KF. We can also notice the generality of the formalism (3) which does not contain factors that are necessarily based on assumptions of model linearity. Let the system model be represented abstractly as f and the measurement model be represented abstractly as h (be them linear or nonlinear):

$$\begin{aligned}\mathbf{x}_t &= f(\mathbf{x}_{t-1}, \mathbf{u}_t) \\ \mathbf{z}_t &= h(\mathbf{x}_t)\end{aligned}$$

We further assume that we have suitable models to compute the predicted state, the predicted state covariance, the predicted measurement covariance, and the cross covariance, which are denoted as g_x, g_{xx}, g_{zz}, and g_{xz} respectively—It is worth noting that the state predicted with $\hat{\mathbf{x}}_{t-1}$ and $\hat{\mathbf{u}}_t$ via f is not necessarily to be the mean of the predicted distribution (though for linear systems it is)—Then we can reformulate the KF in a rather generic way as in (4) and (5):

Prediction:

$$\begin{aligned}\bar{\mathbf{x}}_t &= g_x(\hat{\mathbf{x}}_{t-1}, \hat{\mathbf{u}}_t, f) \\ \bar{\mathbf{\Sigma}}_t &= g_{xx}(\hat{\mathbf{x}}_{t-1}, \hat{\mathbf{u}}_t, \hat{\mathbf{\Sigma}}_{t-1}, \hat{\mathbf{\Sigma}}_{\mathbf{u}}, f)\end{aligned} \tag{4}$$

Update:

$$\begin{aligned} \mathbf{\Sigma}_{xz} &= g_{xz}(\bar{\mathbf{x}}_t, \bar{\mathbf{\Sigma}}_t, h) \\ \mathbf{\Sigma}_{yy} &= \mathbf{\Sigma}_\gamma + g_{zz}(\bar{\mathbf{x}}_t, \bar{\mathbf{\Sigma}}_t, h) \\ \mathbf{K} &= \mathbf{\Sigma}_{xz}\mathbf{\Sigma}_{yy}^{-1} \\ \hat{\mathbf{y}}_t &= \mathbf{z}_t - \hat{\mathbf{z}}_t \\ \hat{\mathbf{x}}_t &= \bar{\mathbf{x}}_t + \mathbf{K}\hat{\mathbf{y}}_t \\ \hat{\mathbf{\Sigma}}_t &= \bar{\mathbf{\Sigma}}_t - \mathbf{K}\mathbf{\Sigma}_{yy}\mathbf{K}^T \end{aligned} \tag{5}$$

Above formalism in (4) and (5) holds for arbitrary f and h and the key problem is how do we know the functions g_x, g_{xx}, g_{zz}, and g_{xz}? The essential difference among different variants of the KF consists in how they define these functions. For the EKF, g_x is defined as f; g_{xx}, g_{zz}, and g_{xz} are derived from f and h using the local linearization technique. However, g_x, g_{xx}, g_{zz}, and g_{xz} obtained in this way may result in inconsistent statistics [4], which degrade the ability of the EKF to handle nonlinear estimation problems. How to define g_x, g_{xx}, g_{zz}, and g_{xz} better and use them in the generic KF formalism (4) and (5) is the motivation of Julier and Uhlmann to propose the UKF in [4] [8].We review the UKF in the next section.

2.2 The unscented transformation (UT)

As we have explained previously, variants of the KF differ from each other in how they define the functions g_x, g_{xx}, g_{zz}, and g_{xz}. So our review here focuses on how these functions are defined in the UKF, i.e. how the predicted state, the predicted state covariance, the innovation covariance, and the cross covariance are computed in the UKF. The idea of the UKF which is based on the **unscented transformation** (UT) [9] can be summarized as:

First, we create a set of sample points to approximate the distribution characterized by the state estimate $\hat{\mathbf{x}}_{t-1}$ and the covariance estimate $\hat{\mathbf{\Sigma}}_{t-1}$. These points are called **sigma points** [4] [8]. For each *sigma* point, we can have a predicted state point and a predicted measurement point accordingly by substituting the *sigma* point into the system model and the measurement model. We use statistics of the predicted state points and the predicted measurement points to approximate the predicted state $\bar{\mathbf{x}}_t$, the predicted state covariance $\bar{\mathbf{\Sigma}}_t$, the innovation covariance $\mathbf{\Sigma}_{yy}$, and the cross covariance $\mathbf{\Sigma}_{xz}$—With $\bar{\mathbf{\Sigma}}_t$, $\bar{\mathbf{\Sigma}}_t$, $\mathbf{\Sigma}_{yy}$, and $\mathbf{\Sigma}_{xz}$ available, we can then update the estimate via (5).

2.2.1 Create the *sigma* points

The n-dimensional random variable $\mathbf{x}$ with mean $\bar{\mathbf{x}}$ and covariance $\mathbf{\Sigma}_{xx}$ is approximated by $2n+1$ weighted points $\{\mathbf{x}_i, w_i\}$ $(i = 0, 1, ..., 2n)$:

$$\begin{aligned} \mathbf{x}_0 &= \bar{\mathbf{x}} & w_0 &= \kappa/(n+\kappa) \\ \mathbf{x}_j &= \bar{\mathbf{x}} + (\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}})_j & w_j &= 1/2(n+\kappa) \\ \mathbf{x}_{j+n} &= \bar{\mathbf{x}} - (\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}})_j & w_{j+n} &= 1/2(n+\kappa) \end{aligned} \tag{6}$$

where $\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}}$ with the square root symbol $\sqrt{\ }$ does not really denote the square root matrix of the matrix $(n+\kappa)\mathbf{\Sigma}_{xx}$; it represents a kind of "quasi" square root of $(n+\kappa)\mathbf{\Sigma}_{xx}$ satisfying

$$\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}}\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}}^T = (n+\kappa)\mathbf{\Sigma}_{xx}$$

$j = 1, 2, ..., n$ and $(\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}})_j$ denotes the j-th column of the matrix $\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}}$. One can easily verify

$$\begin{aligned} E[\mathbf{x}] &\approx \sum_{i=0}^{2n} w_i \mathbf{x}_i = \bar{\mathbf{x}} \\ E[(\mathbf{x}-\bar{\mathbf{x}})(\mathbf{x}-\bar{\mathbf{x}})^T] &\approx \sum_{i=0}^{2n} w_i (\mathbf{x}-\bar{\mathbf{x}}_i)(\mathbf{x}-\bar{\mathbf{x}}_i)^T = \mathbf{\Sigma}_{xx} \end{aligned}$$

This shows that the mean and the covariance of the *sigma* points coincide exactly with $\bar{\mathbf{x}}$ and $\mathbf{\Sigma}_{xx}$. One can compute $\sqrt{(n+\kappa)\mathbf{\Sigma}_{xx}}$ by performing the *Cholesky decomposition* [10] [11] on $(n+\kappa)\mathbf{\Sigma}_{xx}$.

In the estimation, the *sigma* points are created from the **augmented state vector** $\hat{\mathbf{x}}^a_{t-1}$ which means the state vector plus the system input error vector and the system model error vector and from the corresponding **augmented state covariance** $\hat{\mathbf{\Sigma}}^a_{t-1}$:

$$\hat{\mathbf{x}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{x}}_{t-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\hat{\mathbf{\Sigma}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{\Sigma}}_{t-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{\Sigma}}_{\mathbf{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Sigma}_\epsilon \end{bmatrix}$$

Here we assume the state error, the system input error, and the system model error are mutually independent. Let the *sigma* points created from $\hat{\mathbf{x}}^a_{t-1}$ and $\hat{\mathbf{\Sigma}}^a_{t-1}$ according to (6) be denoted as $\hat{\mathbf{x}}^a_{t-1,i}$ $(i = 0, 1, ..., 2n)$.

2.2.2 Compute the predicted mean and the predicted covariance

Compute the predicted state points with the *sigma* points $\hat{\mathbf{x}}^a_{t-1,i}$ $(i = 0, 1, ..., 2n)$ according to the system model:

$$\bar{\mathbf{x}}_{t,i} = f(\hat{\mathbf{x}}^a_{t-1,i}, \hat{\mathbf{u}}_t)$$

Then the predicted mean is computed as

$$\bar{\mathbf{x}}_t = \sum_{i=0}^{2n} w_i \bar{\mathbf{x}}_{t,i} \tag{7}$$

and the predicted covariance is computed as

$$\bar{\mathbf{\Sigma}}_t = \sum_{i=0}^{2n} w_i (\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)(\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)^T \tag{8}$$

The idea of (7) and (8) is to use the statistics of the predicted state points to approximate the *a priori* state and covariance.

2.2.3 Compute the predicted measurement, the innovation covariance, and the cross covariance

The predicted measurement points are computed with the predicted state points $\bar{\mathbf{x}}_{t,i}$ $(i = 0, 1, ..., 2n)$ as

$$\hat{\mathbf{z}}_{t,i} = h(\bar{\mathbf{x}}_{t,i})$$

The predicted measurement i.e. the mean of the predicted measurement points is computed as

$$\hat{\mathbf{z}}_t = \sum_{i=0}^{2n} w_i \hat{\mathbf{z}}_{t,i}$$

Then the innovation covariance is computed as

$$\mathbf{\Sigma}_{yy} = \mathbf{\Sigma}_\gamma + \sum_{i=0}^{2n} w_i (\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T \tag{9}$$

and the cross covariance is computed as

$$\mathbf{\Sigma}_{xz} = \sum_{i=0}^{2n} w_i(\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T \tag{10}$$

2.3 Why "unscented"?

One may pose the question: why is the UKF called "unscented"? The creator of the unscented transformation, Uhlmann, might have certain special inspiration as he gave his method this name in [9]. In our understanding, we see it as a nice figurative expression. The word root "scented" means literally "emitting something (usually a nice odor)". From this sense we can "deduce" an implicit fact that a scented object is also an object which gradually "loses" something as it emits this thing. So the term "unscented" in the UKF implies that something is well kept without being gradually lost in the estimation. Then what is this thing not lost? The answer is true statistics of the state (which may be gradually lost if the EKF is used).

Besides, as already mentioned, "scented" usually implies "emitting something *desirable*" in opposition to a similar word "stinky" which means "emitting something undesirable and offensive". As true statistics of the state are something desirable (like the desirable thing contained in a scented object), it is apparently more proper to call the UKF as it is than to call it in some way such as the "unstinky" KF.

2.4 Summary of the UKF

We have reviewed the key idea of the UKF in previous subsections; here, we give a complete summary of the UKF.

Prediction:

1. Create the *sigma* points $\hat{\mathbf{x}}^a_{t-1,i}$ $(i = 0, 1, ..., 2n)$ from the **augmented state vector** $\hat{\mathbf{x}}^a_{t-1}$ and from the corresponding **augmented state covariance** $\hat{\mathbf{\Sigma}}^a_{t-1}$ according to (11):

$$\begin{aligned}
\hat{\mathbf{x}}^a_{t-1,0} &= \hat{\mathbf{x}}^a_{t-1} & w_0 &= \kappa/(n+\kappa) \\
\hat{\mathbf{x}}^a_{t-1,j} &= \hat{\mathbf{x}}^a_{t-1} + (\sqrt{(n+\kappa)\hat{\mathbf{\Sigma}}^a_{t-1}})_j & w_j &= 1/2(n+\kappa) \\
\hat{\mathbf{x}}^a_{t-1,j+n} &= \hat{\mathbf{x}}^a_{t-1} - (\sqrt{(n+\kappa)\hat{\mathbf{\Sigma}}^a_{t-1}})_j & w_{j+n} &= 1/2(n+\kappa)
\end{aligned} \tag{11}$$

$$\hat{\mathbf{x}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{x}}_{t-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\hat{\mathbf{\Sigma}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{\Sigma}}_{t-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{\Sigma}}_{\mathbf{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Sigma}_{\epsilon} \end{bmatrix}$$

2. Compute the predicted state points with the *sigma* points $\hat{\mathbf{x}}^a_{t-1,i}$ $(i = 0, 1, ..., 2n)$ according to the system model:

$$\bar{\mathbf{x}}_{t,i} = f(\hat{\mathbf{x}}^a_{t-1,i}, \hat{\mathbf{u}}_t)$$

3. Compute the predicted mean $\bar{\mathbf{x}}_t$ as

$$\bar{\mathbf{x}}_t = \sum_{i=0}^{2n} w_i \bar{\mathbf{x}}_{t,i}$$

4. Compute the predicted covariance $\bar{\mathbf{\Sigma}}_t$ as

$$\bar{\mathbf{\Sigma}}_t = \sum_{i=0}^{2n} w_i (\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)(\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)^T$$

Update:

5. Compute the predicted measurement points $\hat{\mathbf{z}}_{t,i}$ $(i = 0, 1, ..., 2n)$ with the predicted state points $\bar{\mathbf{x}}_{t,i}$ $(i = 0, 1, ..., 2n)$ as

$$\hat{\mathbf{z}}_{t,i} = h(\bar{\mathbf{x}}_{t,i})$$

6. Compute the predicted measurement $\hat{\mathbf{z}}_t$ as

$$\hat{\mathbf{z}}_t = \sum_{i=0}^{2n} w_i \hat{\mathbf{z}}_{t,i}$$

7. Compute the innovation covariance $\mathbf{\Sigma}_{yy}$ as

$$\mathbf{\Sigma}_{yy} = \mathbf{\Sigma}_{\gamma} + \sum_{i=0}^{2n} w_i (\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T$$

8. Compute the cross covariance $\mathbf{\Sigma}_{xz}$ as

$$\mathbf{\Sigma}_{xz} = \sum_{i=0}^{2n} w_i(\bar{\mathbf{x}}_{t,i} - \bar{\mathbf{x}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T$$

9. Compute the *a posteriori* state estimate and the *a posteriori* covariance estimate as

$$\begin{aligned}
\mathbf{K} &= \mathbf{\Sigma}_{xz}\mathbf{\Sigma}_{yy}^{-1} \\
\hat{\mathbf{y}}_t &= \mathbf{z}_t - \hat{\mathbf{z}}_t \\
\hat{\mathbf{x}}_t &= \bar{\mathbf{x}}_t + \mathbf{K}\hat{\mathbf{y}}_t \\
\hat{\mathbf{\Sigma}}_t &= \bar{\mathbf{\Sigma}}_t - \mathbf{K}\mathbf{\Sigma}_{yy}\mathbf{K}^T
\end{aligned}$$

3 Vehicle Localization In A 2D Case

3.1 Application description

We consider the same application of vehicle localization in a 2D case as described in [1]. Suppose a vehicle is navigating on a 2D plane and needs to estimate its pose i.e. the position (x, y) and the orientation θ. In other words, we treat the pose of the vehicle as its state to be estimated; this state is denoted compact as $\mathbf{p}$ i.e. $\mathbf{p} = (x, y, \theta)$. The system model is given as the following kinematic model:

$$\begin{cases} x_t &= x_{t-1} + v_t \Delta T cos(\theta_{t-1} + \phi_t \Delta T/2) \\ y_t &= y_{t-1} + v_t \Delta T sin(\theta_{t-1} + \phi_t \Delta T/2) \\ \theta_t &= \theta_{t-1} + \phi_t \Delta T \end{cases} \tag{12}$$

where ΔT denotes the system period; v and ϕ denote respectively the speed and the yawrate of the vehicle. Suppose the vehicle is equipped with devices that monitor its speed and its yawrate. Speed measurements are denoted as $\hat{v}$, and yawrate measurements are denoted as $\hat{\phi}$. Their errors are assumed to follow the Gaussian distribution as $\Delta v_t \sim N(0, \Sigma_v)$ and $\Delta \phi_t \sim N(0, \Sigma_\phi)$.

Suppose the vehicle is also equipped with a component that outputs measurements on the vehicle position (x, y)—vehicle position measurements may be provided by commonly used GPSs [12] [13] or by some *ad hoc* techniques [14] [15] [16]. Here we eliminate any consideration of *ad hoc* factors and just suppose abstractly that we have vehicle position measurements with certain level of accuracy. Our intention is not to validate the feasibility of certain concrete solution for certain concrete application, but to give a fair comparison between performances of the EKF and the UKF. So what we have to guarantee in the tests is that the EKF and the UKF are applied to the **same** data, whereas how these data may be actually provided in real practice is out of our focus here—Let vehicle position measurements be denoted as $\mathbf{z}$ and the measurement model is given as:

$$\mathbf{z}_t = \mathbf{H}\mathbf{p}_t + \gamma_t \tag{13}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

where γ denotes the measurement error which is assumed to follow the Gaussian distribution with zero mean and covariance Σ_γ, i.e. $\gamma \sim N(\mathbf{0}, \Sigma_\gamma)$. The measurement model given in (13) is a partial measurement model.

The system model (12) is a nonlinear model with respect to the vehicle orientation θ and the yawrate input ϕ. To implement the EKF, this nonlinear system model needs to be locally linearized and the locally linearized system model is rewritten as follows:

$$\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} \approx \begin{bmatrix} \bar{x}_t \\ \bar{y}_t \\ \bar{\theta}_t \end{bmatrix} + \mathbf{A}(\mathbf{p}_{t-1}, \mathbf{u}_t) \begin{bmatrix} \Delta x_{t-1} \\ \Delta y_{t-1} \\ \Delta \theta_{t-1} \end{bmatrix} + \mathbf{B}(\mathbf{p}_{t-1}, \mathbf{u}_t) \begin{bmatrix} \Delta v_t \\ \Delta \phi_t \end{bmatrix} \tag{14}$$

$$\begin{cases} \bar{x}_t & = x_{t-1} + v_t \Delta T cos(\theta_{t-1} + \phi_t \Delta T/2) \\ \bar{y}_t & = y_{t-1} + v_t \Delta T sin(\theta_{t-1} + \phi_t \Delta T/2) \\ \bar{\theta}_t & = \theta_{t-1} + \phi_t \Delta T \end{cases}$$

$$\mathbf{A}(\mathbf{p}_{t-1}, \mathbf{u}_t) = \begin{bmatrix} 1 & 0 & -v_t \Delta T sin(\theta_{t-1} + \phi_t \Delta T/2) \\ 0 & 1 & v_t \Delta T cos(\theta_{t-1} + \phi_t \Delta T/2) \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{B}(\mathbf{p}_{t-1}, \mathbf{u}_t) = \begin{bmatrix} \Delta T cos(\theta_{t-1} + \phi_t \Delta T/2) & -v_t \Delta T^2 sin(\theta_{t-1} + \phi_t \Delta T/2)/2 \\ \Delta T sin(\theta_{t-1} + \phi_t \Delta T/2) & v_t \Delta T^2 cos(\theta_{t-1} + \phi_t \Delta T/2)/2 \\ 0 & \Delta T \end{bmatrix}$$

where $\mathbf{u}_t = (v_t, \phi_t)$. The matrices $\mathbf{A}(\mathbf{p}_{t-1}, \mathbf{u}_t)$ and $\mathbf{B}(\mathbf{p}_{t-1}, \mathbf{u}_t)$ are actually the Jacobian matrices of the state evolution function (specified in (12)) with respect to $\mathbf{p}_{t-1}$ and $\mathbf{u}_t$ respectively. With this locally linearized system model (14) and the measurement model (13), we can apply the EKF and details can be referred to in [1].

To apply the UKF, we are exempt from the trouble of locally linearizing the system model; we can just treat the system model (12) and the measurement model (13) as **black boxes**. The UKF is carried out as follows:

Prediction:

1. Create the *sigma* points $\hat{\mathbf{p}}^a_{t-1,i} = [\hat{\mathbf{p}}_{t-1,i}, \Delta v_{t,i}, \Delta\phi_{t,i}]^T$ $(i = 0, 1, ..., 2n)$ from the **augmented state vector** $\hat{\mathbf{p}}^a_{t-1}$ and the corresponding **augmented state covariance** $\hat{\mathbf{\Sigma}}^a_{t-1}$ via (15):

$$\begin{aligned}
\hat{\mathbf{p}}^a_{t-1,0} &= \hat{\mathbf{p}}^a_{t-1} & w_0 &= \kappa/(n+\kappa) \\
\hat{\mathbf{p}}^a_{t-1,j} &= \hat{\mathbf{p}}^a_{t-1} + (\sqrt{(n+\kappa)\hat{\mathbf{\Sigma}}^a_{t-1}})_j & w_j &= 1/2(n+\kappa) \\
\hat{\mathbf{p}}^a_{t-1,j+n} &= \hat{\mathbf{p}}^a_{t-1} - (\sqrt{(n+\kappa)\hat{\mathbf{\Sigma}}^a_{t-1}})_j & w_{j+n} &= 1/2(n+\kappa)
\end{aligned} \tag{15}$$

$$\hat{\mathbf{p}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{p}}_{t-1} \\ 0 \\ 0 \end{bmatrix}$$

$$\hat{\mathbf{\Sigma}}^a_{t-1} = \begin{bmatrix} \hat{\mathbf{\Sigma}}_{t-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Sigma_v & 0 \\ \mathbf{0} & 0 & \Sigma_\phi \end{bmatrix}$$

Here we only consider the system input error while neglecting the system model error. As stated in [4] [8], carefully tuning the parameter κ may enhance the performance of the UKF. In the tests, our focus was not on examining the optimal performance of the UKF, but on comparison between the performances of the UKF and the EKF. So we simply set κ to 1/2 (so that the weights for the *sigma* points are the same)—if the UKF with a κ not carefully tuned can achieve better performance than the EKF does, then the UKF with a carefully tuned κ will achieve even better performance.

2. Compute the predicted state points $\bar{\mathbf{p}}_{t,i} = [\bar{x}_{t,i}, \bar{y}_{t,i}, \bar{\theta}_{t,i}]^T$ with the *sigma* points $\hat{\mathbf{p}}^a_{t-1,i}$ $(i = 0, 1, ..., 2n)$ according to the system model (12):

$$\begin{cases}
\bar{x}_{t,i} &= \hat{x}_{t-1,i} + (\hat{v}_t + \Delta v_{t,i})\Delta T cos(\hat{\theta}_{t-1,i} + (\hat{\phi}_t + \Delta\phi_{t,i})\Delta T/2) \\
\bar{y}_{t,i} &= \hat{y}_{t-1,i} + (\hat{v}_t + \Delta v_{t,i})\Delta T sin(\hat{\theta}_{t-1,i} + (\hat{\phi}_t + \Delta\phi_{t,i})\Delta T/2) \\
\bar{\theta}_{t,i} &= \hat{\theta}_{t-1,i} + (\hat{\phi}_t + \Delta\phi_{t,i})\Delta T
\end{cases}$$

3. Compute the predicted mean $\bar{\mathbf{p}}_t$ as

$$\bar{\mathbf{p}}_t = \sum_{i=0}^{2n} w_i \bar{\mathbf{p}}_{t,i}$$

4. Compute the predicted covariance $\bar{\mathbf{\Sigma}}_t$ as

$$\bar{\mathbf{\Sigma}}_t = \sum_{i=0}^{2n} w_i (\bar{\mathbf{p}}_{t,i} - \bar{\mathbf{p}}_t)(\bar{\mathbf{p}}_{t,i} - \bar{\mathbf{p}}_t)^T$$

Update:

5. Compute the predicted measurement points $\hat{\mathbf{z}}_{t,i}$ $(i = 0, 1, ..., 2n)$ with the predicted state points $\bar{\mathbf{p}}_{t,i}$ $(i = 0, 1, ..., 2n)$ as

$$\hat{\mathbf{z}}_{t,i} = \mathbf{H}\bar{\mathbf{p}}_{t,i}$$

6. Compute the predicted measurement $\hat{\mathbf{z}}_t$ as

$$\hat{\mathbf{z}}_t = \sum_{i=0}^{2n} w_i \hat{\mathbf{z}}_{t,i}$$

7. Compute the innovation covariance $\mathbf{\Sigma}_{yy}$ as

$$\mathbf{\Sigma}_{yy} = \mathbf{\Sigma}_\gamma + \sum_{i=0}^{2n} w_i (\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T$$

8. Compute the cross covariance $\mathbf{\Sigma}_{pz}$ as

$$\mathbf{\Sigma}_{pz} = \sum_{i=0}^{2n} w_i (\bar{\mathbf{p}}_{t,i} - \bar{\mathbf{p}}_t)(\hat{\mathbf{z}}_{t,i} - \hat{\mathbf{z}}_t)^T$$

9. Compute the *a posteriori* state estimate and the *a posteriori* covariance estimate as

$$\begin{aligned}
\mathbf{K} &= \mathbf{\Sigma}_{pz}\mathbf{\Sigma}_{yy}^{-1} \\
\hat{\mathbf{y}}_t &= \mathbf{z}_t - \hat{\mathbf{z}}_t \\
\hat{\mathbf{p}}_t &= \bar{\mathbf{p}}_t + \mathbf{K}\hat{\mathbf{y}}_t \\
\hat{\mathbf{\Sigma}}_t &= \bar{\mathbf{\Sigma}}_t - \mathbf{K}\mathbf{\Sigma}_{yy}\mathbf{K}^T
\end{aligned}$$

3.2 Simulation

We tested performances of the EKF and the UKF using same synthetic data generated according to the system model (12) and the measurement model (13). In the simulation, let $\Delta T = 1(s)$; let $\Sigma_v = 0.1^2(m^2/s^2)$; let $\Sigma_\phi = 0.1^2(rad^2/s^2)$; let $\Sigma_\gamma = diag(2.0^2, 2.0^2)(m^2)$. Set the ground-truth $p_0 = [0(m), 0(m), -\pi/2(rad)]^T$; $v_t = 10(m/s)$ and $\phi_t = 0.0(rad/s)$. The speed measurements and the yawrate measurements were synthesized according to $\hat{v}_t \sim N(v_t, \Sigma_v)$ and $\hat{\phi}_t \sim N(\phi_t, \Sigma_\phi)$. The vehicle position measurements were synthesized according to $z_t \sim N(p_t, \Sigma_\gamma)$.

The EKF and the UKF were applied to the same synthetic data and their estimates on the vehicle state were obtained respectively. The results of 100 Monte Carlo trials are shown in Fig.1 and Fig.2, in both of which the red lines and the blue lines represent respectively the errors of the EKF estimates (after convergence) and those of the UKF estimates (after convergence). The black crosses in Fig.1 represent the position measurement errors.

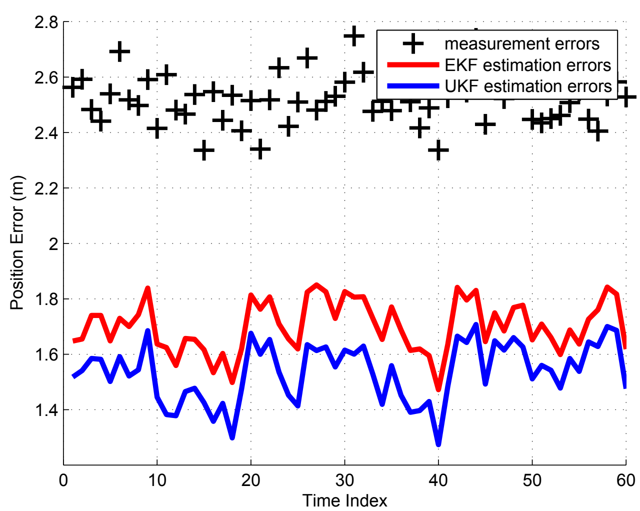

Figure 1: Position estimate errors for 100 Monte Carlo trials

As shown in Fig.1, the position estimates provided by both the EKF and the UKF were considerably more accurate than the position measurements, whereas the UKF brought marginal improvement on position estimate accuracy further compared with the EKF. As shown in Fig.2, the UKF also

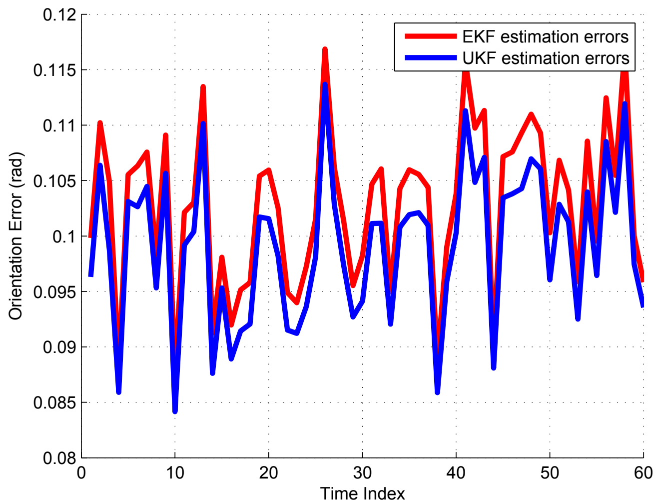

Figure 2: Orientation estimate errors for 100 Monte Carlo trials

brought marginal improvement on orientation estimate accuracy compared with the EKF.

3.3 Discussion

The simulation results presented in the previous subsection demonstrate the advantage of the UKF over the EKF in handling **nonlinear** estimation problems—It is worth noting that **nolinearity** mentioned here does not necessarily refer to the evolution of the state, but refer to the system model and the measurement model. For example, in previously presented simulation, we deliberately set the yawrate ground-truth ϕ to be zero, so that the vehicle was actually set to be moving on a straight line. However, the system model that describes the vehicle movement with potential state errors and system input errors taken into account is nonlinear, so the presented estimation problem is still nonlinear.

It is also worth noting that we should not exaggerate the advantage of the UKF over the EKF. In the example given previously, compared with the EKF, the marginal improvement brought by the UKF is slight (yet existing). For linear estimation problems, the UKF and the EKF will have same performances. In fact, the more exists nonlinearity in an estimation problem, the more noticeable may be the advantage of the UKF over the EKF.

One implementation convenience brought by the UKF is that for implementing the UKF we no longer have the trouble of explicitly computing the Jacobian matrices of the system model function and the measurement model function as we do for implementing the EKF. In the formalism of the UKF, the system model and the measurement model are totally treated as **black boxes**. Although the formalism of the UKF seems to possess more formulas than that of the EKF, its implementation is actually easier than that the EKF.

So, on one hand, the UKF tends to perform at least no worse than the EKF. On the other hand, it is easier to implement the UKF than to implement the EKF. This is why Julier and Uhlmann advocate ubiquitous use of the UKF in applications where the EKF may be used. The UKF do have aroused more and more interests of researchers; examples of its application in the intelligent vehicle domain can be referred to in [17] [18] [19].

If the UKF has disadvantages compared with the EKF, one point may be that the UKF is not as deterministic as the EKF because it contains heuristic factors. Another point may be that its mechanism of uncertainty propagation is not as intuitive as that of the EKF.

4 Conclusion

In this article i.e. the third article of the series "*A brief tutorial on recursive estimation*", we have reviewed an important variant of the KF, i.e. the unscented Kalman filter (UKF), for its advantage over the extended Kalman filter (EKF) in handling nonlinear estimation problems. First, we have revisited the formalism of the KF. Then, we have reviewed a more generic formalism of the KF. Based on the generic formalism of the KF, we have explained the key idea of the unscented Kalman filter. We have also give simulation based comparison between performances of the UKF and the EKF.

Besides the UKF, another powerful method for handling nonlinear estimation problems is the *particle filter*, explanations on which will be postponed to a further article of the series.

References

[1] H. Li. *A Brief Tutorial On Recursive Estimation With Examples From Intelligent Vehicle Applications (Part I): Basic Spirit And Utilities*. HAL Open Archives, 2014.

[2] R.E. Kalman. A new approach to linear filtering and prediction problem. *ASME Trans, Ser. D, J. Basic Eng.*, 82:35–45, 1960.

[3] M.S. Grewal and A.P. Andrews. *Kalman filtering: Theory and practice*. New York, USA: Wiley, 2000.

[4] S.J. Julier and J.K. Uhlmann. A new extension of the kalman filter to nonlinear systems. *Int. symp. aerospace defense sensing, simul. and controls*, 3(26), 1997.

[5] S.J. Julier and J.K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of American Control Conference*, pages 2369–2373, 1997.

[6] S.J. Julier and J.K. Uhlmann. General decentralized data fusion with covariance intersection (ci). *Handbook of Data Fusion*, 2001.

[7] H. Li, F. Nashashibi, and M. Yang. Split covariance intersection filter: Theory and its application to vehicle localization. *IEEE Transactions on Intelligent Transportation Systems*, 14(4):1860–1871, 2013.

[8] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.

[9] J.K. Uhlmann. *Dynamic map building and localization: New theoretical foundations*. Ph.D. Dissertation, University of Oxford, 1995.

[10] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge University Press, 2012.

[11] G.H. Golub and C.F. Van Loan. *Matrix computations*. Johns Hopkins University Press, 1996.

[12] S. Rezaei and R. Sengupta. Kalman filter-based integration of dgps and vehicle sensors for localization. *IEEE Transactions on Control Systems Technology*, 15(6):1080–1088, 2007.

[13] P. Bonnifait, P. Bouron, D. Meizel, and P. Crubille. Dynamic localization of car-like vehicles using data fusion of redundant abs sensors. *The Journal of Navigation*, 56(3):429–441, 2003.

[14] T.K. Xia, M. Yang, R.Q. Yang, and C. Wang. Cyberc3: A prototype cybernetic transportation system for urban applications. *IEEE Transactions on Intelligent Transportation Systems*, 11(1):142–152, 2010.

[15] H. Li, F. Nashashibi, and G. Toulminet. Localization for intelligent vehicle by fusing mono-camera, low-cost gps and map data. In *IEEE International Conference on Intelligent Transportation Systems*, pages 1657–1662, 2010.

[16] I. Skog and P. Handel. In-car positioning and navigation technologiesa survey. *IEEE Transactions on Intelligent Transportation Systems*, 10(1):4–21, 2009.

[17] H. Fang, M. Yang, R.Q. Yang, and C. Wang. Ground-texture-based localization for intelligent vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 10(3):2009, 463-468.

[18] M. Meuter, U. Iurgel, S.B. Park, and A. Kummert. The unscented kalman filter for pedestrian tracking from a moving host. In *IEEE Intelligent Vehicles Symposium*, pages 37–42, 2008.

[19] M. Doumiati, A.C. Victorino, A. Charara, and D. Lechner. Onboard real-time estimation of vehicle lateral tire-road forces and sideslip angle. *IEEE/ASME Transactions on Mechatronics*, 16(4):601–614, 2011.