Unscented Kalman Filter Tutorial

Gabriel A. Terejanu
Department of Computer Science and Engineering
University at Buffalo, Buffalo, NY 14260
terejanu@buffalo.edu

1 Introduction

The Unscented Kalman Filter belongs to a bigger class of filters called **Sigma-Point Kalman Filters** or **Linear Regression Kalman Filters**, which are using the **statistical linearization** technique [1, 5]. This technique is used to linearize a nonlinear function of a random variable through a linear regression between n points drawn from the prior distribution of the random variable. Since we are considering the spread of the random variable the technique tends to be more accurate than Taylor series linearization [7].

In the same family of filters we have The Central Difference Kalman Filter, The Divided Difference Filter, and also the Square-Root alternatives for UKF and CDKF [7].

In EKF the state distribution is propagated analytically through the first-order linearization of the nonlinear system due to which, the posterior mean and covariance could be corrupted. The UKF, which is a derivative-free alternative to EKF, overcomes this problem by using a deterministic sampling approach [9]. The state distribution is represented using a minimal set of carefully chosen sample points, called **sigma points**. Like EKF, UKF consists of the same two steps: model forecast and data assimilation, except they are preceded now by another step for the selection of sigma points.

2 UKF Algorithm

The UKF is founded on the intuition that it is easier to approximate a probability distribution that it is to approximate an arbitrary nonlinear function or transformation [4]. The sigma points are chosen so that their mean and covariance to be exactly $\mathbf{x}_{k-1}^a$ and $\mathbf{P}_{k-1}$. Each sigma point is then propagated through the nonlinearity yielding in the end a cloud of transformed points. The new estimated mean and covariance are then computed based on their statistics. This process is called unscented transformation. *The unscented transformation is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation* [9].

Consider the following nonlinear system, described by the difference equation and the observation model with additive noise:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1} \tag{1}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{2}$$

The **initial state** $\mathbf{x}_0$ is a random vector with known mean $\mu_0 = E[\mathbf{x}_0]$ and covariance $\mathbf{P}_0 = E[(\mathbf{x}_0 - \mu_0)(\mathbf{x}_0 - \mu_0)^T]$. In the case of non-additive process and measurement noise, the unscented transformation scheme is applied to the augmented state [9]:

$$\mathbf{x}_k^{aug} = [\mathbf{x}_k^T \quad \mathbf{w}_{k-1}^T \quad \mathbf{v}_k^T]^T \tag{3}$$

Set Selection of Sigma Points

Let $\mathbf{X}_{k-1}$ be a set of $2n+1$ sigma points (where n is the dimension of the state space) and their associated weights:

$$\mathbf{X}_{k-1} \quad = \quad \left\{ \left(\mathbf{x}_{k-1}^j, W^j \right) | j = 0 \ldots 2n \right\} \tag{4}$$

Consider the following selection of sigma points, selection that incorporates higher order information in the selected points [4]:

$$\mathbf{x}_{k-1}^0 \quad = \quad \mathbf{x}_{k-1}^a \tag{5}$$

$$-1 \quad < \quad W^0 < 1 \tag{6}$$

$$\mathbf{x}_{k-1}^i \quad = \quad \mathbf{x}_{k-1}^a + \left(\sqrt{\frac{n}{1 - W^0} \mathbf{P}_{k-1}} \right)_i \text{ for all } i = 1 \ldots n \tag{7}$$

$$\mathbf{x}_{k-1}^{i+n} \quad = \quad \mathbf{x}_{k-1}^a - \left(\sqrt{\frac{n}{1 - W^0} \mathbf{P}_{k-1}} \right)_i \text{ for all } i = 1 \ldots n \tag{8}$$

$$W^j \quad = \quad \frac{1 - W^0}{2n} \text{ for all } j = 1 \ldots 2n \tag{9}$$

where the weights must obey the condition:

$$\sum_{j=0}^{2n} W^j \quad = \quad 1 \tag{10}$$

and $\left(\sqrt{\frac{n}{1-W^0} \mathbf{P}_{k-1}} \right)_i$ is the row or column of the matrix square root of $\frac{n}{1-W^0} \mathbf{P}_{k-1}$. W^0 controls the position of sigma points: $W^0 \geq 0$ points tend to move further from the origin, $W^0 \leq 0$ points tend to be closer to the origin. A more general selection scheme for sigma points, called *scaled unscented transformation*, is given in [9, 2].

Model Forecast Step

Each sigma point is propagated through the nonlinear process model:

$$\mathbf{x}_k^{f,j} \quad = \quad \mathbf{f}(\mathbf{x}_{k-1}^j) \tag{11}$$

The transformed points are used to compute the mean and covariance of the forecast value of $\mathbf{x}_k$:

$$\mathbf{x}_k^f \quad = \quad \sum_{j=0}^{2n} W^j \mathbf{x}_k^{f,j} \tag{12}$$

$$\mathbf{P}_k^f \quad = \quad \sum_{j=0}^{2n} W^j \left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f \right) \left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f \right)^T + \mathbf{Q}_{k-1} \tag{13}$$

We propagate then the sigma points through the nonlinear observation model:

$$\mathbf{z}_{k-1}^{f,j} = \mathbf{h}(\mathbf{x}_{k-1}^{j}) \tag{14}$$

With the resulted transformed observations, their mean and covariance (innovation covariance) is computed:

$$\mathbf{z}_{k-1}^{f} = \sum_{j=0}^{2n} W^j \mathbf{z}_{k-1}^{f,j} \tag{15}$$

$$Cov(\widetilde{\mathbf{z}}_{k-1}^{f}) = \sum_{j=0}^{2n} W^j \left(\mathbf{z}_{k-1}^{f,j} - \mathbf{z}_{k-1}^{f}\right)\left(\mathbf{z}_{k-1}^{f,j} - \mathbf{z}_{k-1}^{f}\right)^T + \mathbf{R}_k \tag{16}$$

The cross covariance between $\widetilde{\mathbf{x}}_k^f$ and $\widetilde{\mathbf{z}}_{k-1}^f$ is:

$$Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^{f}) = \sum_{j=0}^{2n} W^j \left(\mathbf{x}_{k}^{f,j} - \mathbf{x}_{k}^{f}\right)\left(\mathbf{z}_{k-1}^{f,j} - \mathbf{z}_{k-1}^{f}\right)^T \tag{17}$$

Data Assimilation Step

We like to combine the information obtained in the forecast step with the new observation measured $\mathbf{z}_k$. Like in KF assume that the estimate has the following form:

$$\mathbf{x}_k^a = \mathbf{x}_k^f + \mathbf{K}_k(\mathbf{z}_k - \mathbf{z}_{k-1}^f) \tag{18}$$

The gain $\mathbf{K}_k$ is given by:

$$\mathbf{K}_k = Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^f) Cov^{-1}(\widetilde{\mathbf{z}}_{k-1}^f) \tag{19}$$

The posterior covariance is updated after the following formula:

$$\mathbf{P}_k = \mathbf{P}_k^f - \mathbf{K}_k Cov(\widetilde{\mathbf{z}}_{k-1}^f)\mathbf{K}_k^T \tag{20}$$

3 Square-Root UKF

Note that in order to compute the new set of sigma points we need the square root matrix of the posterior covariance each time ($\mathbf{P}_k = \mathbf{S}_k\mathbf{S}_k^T$). Since the update is applied to the full posterior covariance we can change the algorithm to propagate directly the square root matrix, $\mathbf{S}_k$.

The selection scheme of sigma points becomes:

$$\mathbf{x}_{k-1}^0 = \mathbf{x}_{k-1}^a \tag{21}$$

$$-1 < W^0 < 1 \tag{22}$$

$$\mathbf{x}_{k-1}^i = \mathbf{x}_{k-1}^a + \left(\sqrt{\frac{n}{1-W^0}}\mathbf{S}_{k-1}\right)_i \quad \text{for all } i = 1 \ldots n \tag{23}$$

$$\mathbf{x}_{k-1}^{i+n} = \mathbf{x}_{k-1}^a - \left(\sqrt{\frac{n}{1-W^0}}\mathbf{S}_{k-1}\right)_i \quad \text{for all } i = 1 \ldots n \tag{24}$$

$$W^j = \frac{1-W^0}{2n} \quad \text{for all } j = 1 \ldots 2n \tag{25}$$

The filter is initialized by computing the initial square root matrix via a Cholesky factorization of the full error covariance matrix.

$$\mathbf{S}_0 = chol\left(E[(\mathbf{x}_0 - \mu_0)(\mathbf{x}_0 - \mu_0)^T]\right) \tag{26}$$

Since $W^j > 0$ for all $i \geq 1$, in the *time update* step the forecast covariance matrix can be written as:

$$\begin{aligned}
\mathbf{P}_k^f &= \sum_{j=0}^{2n} W^j \left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)^T + \mathbf{Q}_{k-1} \\
&= \sum_{j=1}^{2n} \sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)\sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)^T + \sqrt{\mathbf{Q}_{k-1}}\sqrt{\mathbf{Q}_{k-1}}^T \\
&\quad + W^0\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)^T \\
&= \left[\sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right) \quad \sqrt{\mathbf{Q}_{k-1}}\right]\left[\begin{array}{c} \sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)^T \\ \\ \sqrt{\mathbf{Q}_{k-1}}^T \end{array}\right] \\
&\quad + W^0\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)^T \qquad \text{for } j = 1 \ldots 2n
\end{aligned} \tag{27}$$

where $\sqrt{\mathbf{Q}_{k-1}}$ is the square-root matrix of the process noise covariance matrix. This form is computationally undesirable since we have tripled the number of columns.

$$\left[\sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right) \quad \sqrt{\mathbf{Q}_{k-1}}\right] \in \mathcal{R}^{n\times 3n} \qquad \text{for } j = 1 \ldots 2n$$

We can use the QR-decomposition to express the transpose of the above matrix in terms of an orthogonal matrix $\mathbf{O}_k \in \mathcal{R}^{3n\times n}$ and an upper triangular matrix $(\mathbf{S}_k^f)^T \in \mathcal{R}^{n\times n}$.

$$\left[\sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right) \quad \sqrt{\mathbf{Q}_{k-1}}\right]^T = \mathbf{O}_k(\mathbf{S}_k^f)^T \qquad \text{for } j = 1 \ldots 2n$$

Hence the error covariance matrix:

$$\begin{aligned}
\mathbf{P}_k^f &= \mathbf{S}_k^f\mathbf{O}_k^T\mathbf{O}_k(\mathbf{S}_k^f)^T + W^0\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)^T \\
&= \mathbf{S}_k^f(\mathbf{S}_k^f)^T + W^0\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)^T
\end{aligned} \tag{28}$$

In order to include the effect of the last term in the square-root matrix, we have to perform a rank 1 update to Cholesky factorization.

$$\mathbf{S}_k^f = cholupdate\left(\mathbf{S}_k^f, \quad (\mathbf{x}_k^{f,0} - \mathbf{x}_k^f), \quad sgn\{W^0\}\sqrt{W^0}\right) \tag{29}$$

where *sgn* is the sign function and *cholupdate* returns the Cholesky factor of

$$\mathbf{S}_k^f(\mathbf{S}_k^f)^T + W^0\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)\left(\mathbf{x}_k^{f,0} - \mathbf{x}_k^f\right)^T$$

Therefore the forecast covariance matrix can be written $\mathbf{P}_k^f = \mathbf{S}_k^f(\mathbf{S}_k^f)^T$. The same way the posterior covariance can be expressed as $\mathbf{P}_k = \mathbf{S}_k(\mathbf{S}_k)^T$ and the innovation covariance as $Cov(\widetilde{\mathbf{z}}_{k-1}^f) = \mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}^T$.

Time-update summary

$$
\begin{aligned}
\mathbf{x}_k^{f,j} &= \mathbf{f}(\mathbf{x}_{k-1}^j) &(30)\\
\mathbf{x}_k^f &= \sum_{j=0}^{2n} W^j \mathbf{x}_k^{f,j} &(31)\\
\mathbf{S}_k^f &= qr\left(\left[\sqrt{W^j}\left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right) \quad \sqrt{\mathbf{Q}_{k-1}}\right]\right) \qquad \text{for } j = 1\ldots 2n &(32)\\
\mathbf{S}_k^f &= cholupdate\left(\mathbf{S}_k^f, \ \ (\mathbf{x}_k^{f,0} - \mathbf{x}_k^f), \ \ sgn\{W^0\}\sqrt{W^0}\right) &(33)
\end{aligned}
$$

Redraw sigma points to incorporate effect of process noise:

$$
\begin{aligned}
\mathbf{x}_k^{f,0} &= \mathbf{x}_k^f &(34)\\
\mathbf{x}_k^{f,i} &= \mathbf{x}_k^f + \left(\sqrt{\frac{n}{1-W^0}}\mathbf{S}_k^f\right)_i \ \text{for all } i = 1\ldots n &(35)\\
\mathbf{x}_k^{f,i+n} &= \mathbf{x}_k^f - \left(\sqrt{\frac{n}{1-W^0}}\mathbf{S}_k^f\right)_i \ \text{for all } i = 1\ldots n &(36)
\end{aligned}
$$

Propagate the new sigma points through measurement model:

$$
\begin{aligned}
\mathbf{z}_{k-1}^{f,j} &= \mathbf{h}(\mathbf{x}_{k-1}^{f,j}) &(37)\\
\mathbf{z}_{k-1}^f &= \sum_{j=0}^{2n} W^j \mathbf{z}_{k-1}^{f,j} &(38)\\
\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f} &= qr\left(\left[\sqrt{W^j}\left(\mathbf{z}_{k-1}^{f,j} - \mathbf{z}_{k-1}^f\right) \quad \sqrt{\mathbf{R}_k}\right]\right) \qquad \text{for } j = 1\ldots 2n &(39)\\
\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f} &= cholupdate\left(\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}, \ \ (\mathbf{z}_{k-1}^{f,0} - \mathbf{z}_{k-1}^f), \ \ sgn\{W^0\}\sqrt{W^0}\right) &(40)\\
Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^f) &= \sum_{j=0}^{2n} W^j \left(\mathbf{x}_k^{f,j} - \mathbf{x}_k^f\right)\left(\mathbf{z}_{k-1}^{f,j} - \mathbf{z}_{k-1}^f\right)^T &(41)
\end{aligned}
$$

The qr function returns only the lower triangular matrix.

Measurement-update summary

$$
\begin{aligned}
\mathbf{x}_k^a &= \mathbf{x}_k^f + \mathbf{K}_k(\mathbf{z}_k - \mathbf{z}_{k-1}^f) &(42)\\
\mathbf{K}_k &= \left(Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^f)/\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}^T\right)/\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f} &(43)\\
\mathbf{S}_k &= cholupdate\left(\mathbf{S}_k^f, \ \ \mathbf{K}_k Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^f), \ \ -1\right) &(44)
\end{aligned}
$$

where / denotes a back-substitution operation. This is a better alternative to the matrix inversion. Since the Cholesky factor is a lower triangular matrix, we can find $\mathbf{K}_k$ using two back-substitution operations in the equation:

$$
\mathbf{K}_k(\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}\mathbf{S}_{\widetilde{\mathbf{z}}_{k-1}^f}^T) = Cov(\widetilde{\mathbf{x}}_k^f, \widetilde{\mathbf{z}}_{k-1}^f) \qquad (45)
$$

In eqn. (44) since the middle argument of the *cholupdate* function is a matrix $\in \mathcal{R}^{n\times n}$, the result is n consecutive updates of the Cholesky factor using the n columns of the matrix.

Since QR-decomposition and Cholesky factorization tend to control better the round off errors and there are no matrix inversions, the SR-UKF has better numerical properties and it also guarantees positive semi-definiteness of the underlying state covariance [8].

4 Conclusion

The UKF represents the extra uncertainty on a linearized function due to linearization errors by the covariance of the deviations between the nonlinear and the linearized function in the regression points [6].

The approximations obtained with at least $2n + 1$ sampling points are accurate to the 3rd order of Gaussian inputs for all nonlinearities and at least to the 2nd for non-Gaussian inputs. The Reduced Sigman Point Filters [3] uses only $n + 1$ sampling points but this time it does not take into account the linearization errors.

References

[1] Arthur Gelb. *Applied Optimal Estimation*. M.I.T. Press, 1974.

[2] S. Julier. The Scaled Unscented Transformation, 1999.

[3] S. Julier, J. Jeffrey, and K. Uhlmann. Reduced Sigma Point Filters for the Propagation of Means and Covariances Through Nonlinear Transformations. 2002.

[4] S. J. Julier and J. K. Uhlmann. Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.

[5] Tine Lefebvre and Herman Bruyninckx. Kalman Filters for Nonlinear Systems: A Comparison of Performance.

[6] Tine Lefebvre and Herman Bruyninckx. Comment on "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators". *IEEE Trans.Automatic Control*, 2002.

[7] R. van der Merwe. Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. Technical report, 2003.

[8] R. van der Merwe and E. Wan. The Square-Root Unscented Kalman Filter for State and Parameter-Estimation, 2001.

[9] E. Wan and R. van der Merwe. *The Unscented Kalman Filter*. Wiley Publishing, 2001.