Development of Levenberg-Marquardt Method Based Iteration Square Root Cubature Kalman Filter ant its applications

Jing Mu * , Changyuan Wang

School of Computer Science and Engineering

Xi'an Technological University, Xi'an, 710032, China;

*** Corresponding author, Email: mujing1977@163.com;**

Abstract. Levenberg-Marquardt(abbr. L-M) method based iterative square root cubature Kalman filter (ISRCKFLM) is proposed to improve the low state estimation accuracy of nonlinear state estimation due to large initial estimation error and nonlinearity of measurement equation. The measurement update of square root cubature Kalman filter (SRCKF) is transformed to the problem of nonlinear least square error, then we use L-M method to solve it and obtain the optimal state estimation and covariance, so the ISRCKFLM algorithm has the virtues of global convergence and numerical stability. We apply the ISRCKFLM algorithm to state estimation for re-entry ballistic target; the simulation results demonstrate the ISRCKFLM algorithm has better accuracy of state estimation.

Keywords: Nonlinear filtering, Cubature Kalman filter, Levenberg-Marquardt method

1. Introductiong

A series of nonlinear filters have been developed to apply to state estimation for the last decades. Up to now the commonly used non-linear filtering is the extended Kalman filter (EKF) [1,2]. The EKF is based on first-order Taylor approximations of state transition and observation equation about the estimated state trajectory under Gaussian assumption, so EKF may introduce significant bias, or even convergence problems due to the overly crude approximation [3].

Recently, one type of suboptimal nonlinear filters based on numerical multi-dimensional integral were introduced in [4-6], such as cubature rules based cubature Kalman filter (CKF) and the interpolatory cubature Kalman filters (ICKFs), which used numerical multi-dimensional integral to approximate the recursive Bayesian estimation integrals under the Gaussian assumption. The CKF can solve high-dimensional nonlinear filtering problems with minimal computational effort and can be deemed as special case of ICKFs. Furthermore, the stability of CKF for non-linear systems with linear measurement is analyzed and the certain conditions to ensure that the estimation error of the CKF remains bounded are proved in [7]. On the other hand, in order to decrease the effect of initial estimation error and nonlinearity of measurement equation, Levenberg-Marquardt method based iteration cubature Kalman filter was developed on the basis of the CKF in Reference [8]. In fact, singular matrix occurs in the implementation of the above filters mentioned if the initial estimation is selected improperly. So the cubature rule is exploited as square root cubature information filter [9] and the square root cubature Kalman filter (SRCKF) was developed in order to mitigate ill effects and improve the numerical stability [5].

However, because of the large initial error and measurement error in the state estimation for re-entry ballistic target with unknown ballistic coefficient, which is the nonlinear dynamics system with the feature of hidden markov model , the SRCKF also shows its weakness in the robustness and estimation accuracy. As we know Levenberg-Marquardt (abbr. L-M) method has the global convergence and fast covergence. Making use of L-M method and the superiority of the SRCKF algorithm, we develop the L-M method based iterative square root cubature Kalman filter (ISRCKFLM), in which, we transform the measurement update of SRCKF to the problem of nonlinear least square error, then use L-M method to solve it and obtain the optimal state estimation and covariance to improve the low state estimation accuracy of nonlinear state estimation due to large initial estimation error and nonlinearity of measurement equation.

The rest of the paper is organized as follows. We begin in Section 2 with a description of square root cubature Kalman filter (SRCKF). The L-M method based iterative square root cubature Kalman filter (ISRCKFLM) is developed in Section 3. Then we apply the ISRCKFLM algorithm to track re-entry ballistic target (RBT) with unknown ballistic coefficient and discuss the simulation results in Section 4. Finally, Section 5 concludes the paper.

2. Square Root Cubature Kalman Filter

Consider the following nonlinear dynamics system:

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

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

Where $\boldsymbol{f}$ and $\boldsymbol{h}$ are some known nonlinear functions; $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ and $\boldsymbol{z}_k \in \mathbb{R}^{n_z}$ is state and the measurement vector, respectively; $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$ are process and measurement Gaussian noise sequences with zero means and covariance $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$, respectively, and $\{\boldsymbol{w}_{k-1}\}$ and $\{\boldsymbol{v}_k\}$ are mutually uncorrelated.

Suppose that the state distribution is $\boldsymbol{x}_{k-1} \sim \mathcal{N}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{P}_{k-1})$, and a square root $\boldsymbol{S}_{k-1}$ of $\boldsymbol{P}_{k-1}$ such that $\boldsymbol{P}_{k-1} = \boldsymbol{S}_{k-1}\boldsymbol{S}_{k-1}^T$ is obtained. The square root cubature Kalman filter (SRCKF) algorithm is summarized as follows.

Step 1. Time Update

1) Calculate the cubature points and propagate the cubature points through the state equation

$$\mathbf{X}_{i,k-1} = \mathbf{S}_{k-1}\boldsymbol{\xi}_i + \hat{\mathbf{x}}_{k-1} \tag{3}$$

$$\boldsymbol{X}_{i,k}^* = \boldsymbol{f}(\boldsymbol{X}_{i,k-1}) \tag{4}$$

where $\hat{\boldsymbol{\imath}}_i = \sqrt{m/2}[1]_i, \omega_i = 1/m, i = 1, \cdots m = 2n_x$, the $[1]_i$ is a n_x dimensional vector and is generated according to the way described in [5].

2) Evaluate the predicted state and square root of the predicted covariance

$$\bar{\boldsymbol{x}}_k = \sum_{i=1}^{m} \omega_i \boldsymbol{X}_{i,k}^* \tag{5}$$

$$\bar{\boldsymbol{S}}_k = Tria([\chi_k^* \ \boldsymbol{S}_{Q,k-1}]) \tag{6}$$

here, $\boldsymbol{S}_{Q,k-1}$ denotes a square-root factor of $\boldsymbol{Q}_{k-1}$ and Tria() is denoted as a general triagularization algorithm. The matrix $\div_k^*$ is defined as:

$$\chi_k^* = 1/\sqrt{m}[\boldsymbol{X}_{1,k}^* - \bar{\boldsymbol{x}}_k \ \ \boldsymbol{X}_{2,k}^* - \bar{\boldsymbol{x}}_k, \cdots, \boldsymbol{X}_{m,k}^* - \bar{\boldsymbol{x}}_k] \tag{7}$$

Step 2. Measurement Update

1) Calculate the predicted cubature points and evaluate the propagated cubature points

$$\boldsymbol{X}_{i,k} = \bar{\boldsymbol{S}}_k \xi_i + \bar{\boldsymbol{x}}_k \tag{8}$$

$$\boldsymbol{Z}_{i,k} = \boldsymbol{h}(\boldsymbol{X}_{i,k}) \tag{9}$$

2) Evaluate the predicted measurement, a square root of the innovation covariance and cross-covariance

$$\bar{z}_k = \sum_{i=1}^{m} \omega_i \boldsymbol{Z}_{i,k} \tag{10}$$

$$\boldsymbol{S}_{zz,k} = Tria([\Upsilon_k \ \boldsymbol{S}_{R,k}]) \tag{11}$$

$$\boldsymbol{P}_{xz,k|k-1} = \ _k\Upsilon_k^T \tag{12}$$

here

$$\Upsilon_k = 1/\sqrt{m}[\boldsymbol{Z}_{1,k} - \bar{z}_k \ \ \boldsymbol{Z}_{2,k} - \bar{z}_k, \cdots, \boldsymbol{Z}_{m,k} - \bar{z}_k] \tag{13}$$

$$\chi_k = 1/\sqrt{m}[\mathbf{X}_{1,k} - \bar{\mathbf{x}}_k \ \ \mathbf{X}_{2,k} - \bar{\mathbf{x}}_k, \cdots, \mathbf{X}_{m,k} - \bar{\mathbf{x}}_k] \tag{14}$$

where $\boldsymbol{S}_{R,k}$ denotes a square root factor of $\boldsymbol{R}_k$.

3) Evaluate the state estimation and the square-root of the covariance at instant time.

$$\boldsymbol{W}_k = (\boldsymbol{P}_{xz,k} / \boldsymbol{S}_{zz,k}^T) / \boldsymbol{S}_{zz,k} \tag{15}$$

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \boldsymbol{W}_k (z_k - \bar{z}_k) \tag{16}$$

$$\boldsymbol{S}_k = Tria(\left[\chi_k - \boldsymbol{W}_k \Upsilon_k \ \ \boldsymbol{W}_k \boldsymbol{S}_{R,k}\right]) \tag{17}$$

where symbol “/” represents the matrix right divide operator.

3. Development of L-M Based Iterative Square Root Cubature Kalman Filter

3.1 L-M method based iterative measurement update

In the time update of the SRCKF algorithm, we get the predicted state $^{-}$ and a square root of corresponding covariance $\bar{\boldsymbol{S}}_k$ [5], and we can obtain $\bar{\boldsymbol{P}}_k = \bar{\boldsymbol{S}}_k \bar{\boldsymbol{S}}_k^T$. Assuming $\bar{\boldsymbol{x}}_k \sim \mathcal{N}(\boldsymbol{x}_k, \bar{\boldsymbol{P}}_k)$, the current measurement is $\boldsymbol{z}_k$, and $\boldsymbol{z}_k \sim \mathcal{N}(\boldsymbol{h}(\boldsymbol{x}_k), \boldsymbol{R}_k)$). Defining the augmented matrix: $\boldsymbol{Y}_k = [\bar{\boldsymbol{x}}_k \ \boldsymbol{z}_k]^T$, and $\boldsymbol{H}(\boldsymbol{x}_k) = [\boldsymbol{x}_k \ \boldsymbol{h}(\boldsymbol{x}_k)]^T$.

Defining the residual function:

$$\boldsymbol{\Psi}(\boldsymbol{x}_k) = \boldsymbol{W}_k^{-1} \boldsymbol{V}_k \tag{18}$$

here

$$\boldsymbol{W}_k = \begin{bmatrix} \bar{\boldsymbol{P}}_k^{1/2} & \boldsymbol{0}_{n_x \times n_z} \\ \boldsymbol{0}_{n_x \times n_z} & \boldsymbol{R}_k^{1/2} \end{bmatrix} \tag{19}$$

$$\boldsymbol{C}_k = \boldsymbol{W}_k \boldsymbol{W}_k^T = \begin{bmatrix} \bar{\boldsymbol{P}}_k & \boldsymbol{0}_{n_x \times n_z} \\ \boldsymbol{0}_{n_x \times n_z} & \boldsymbol{R}_k \end{bmatrix} \tag{20}$$

$$\boldsymbol{V}_k = \boldsymbol{Y}_k - \boldsymbol{H}(\boldsymbol{x}_k) \tag{21}$$

Defining the cost function:

$$C_{LS}(\boldsymbol{x}_k) = 0.5 \boldsymbol{\Psi}^T(\boldsymbol{x}_k) \boldsymbol{\Psi}(\boldsymbol{x}_k) \tag{22}$$

Defining the matrix: $\tilde{\boldsymbol{P}}_k^{-1} = \left[\bar{\boldsymbol{P}}_k^{-1} + \mu_i \boldsymbol{I} \right]$, and using L-M method and manipulations, the iterate formula can be obtained:

$$\hat{\boldsymbol{x}}_k^{(i+1)} = \bar{\boldsymbol{x}}_k + \boldsymbol{L}_k^{(i)} \left\{ \boldsymbol{z}_k - \boldsymbol{h}(\boldsymbol{x}_k^{(i)}) - \boldsymbol{J}_h(\boldsymbol{x}_k^{(i)})(\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k^{(i)}) \right\} - \mu_i \left\{ \boldsymbol{I} - \boldsymbol{L}_k^{(i)} \boldsymbol{J}_h(\boldsymbol{x}_k^{(i)}) \right\} \tilde{\boldsymbol{P}}_k (\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k^{(i)}) \tag{23}$$

where $\boldsymbol{J}_h(\hat{\boldsymbol{x}}_k^{(i)}) = \partial \boldsymbol{h}(\boldsymbol{x}_k) / \partial \boldsymbol{x}_k |_{\boldsymbol{x}_k = \hat{\boldsymbol{x}}_k^{(i)}}$, $\boldsymbol{I}$ is identity matrix and μ_i is the tuning parameter, the gain $\boldsymbol{L}_k^{(i)}$ is defined:

$$\boldsymbol{L}_k^{(i)} = \tilde{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\hat{\boldsymbol{x}}_k^{(i)}) \left[\boldsymbol{J}_h(\boldsymbol{x}_k^{(i)}) \tilde{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\boldsymbol{x}_k^{(i)}) + \boldsymbol{R}_k \right]^{-1} \tag{24}$$

To obtain the covariance, we derive the equation (3) to get its extremum and get:

$$\left(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^{(N)} \right) = \left\{ \boldsymbol{J}_V^T(\boldsymbol{x}_k^{(N)}) \boldsymbol{C}_k^{-1} \boldsymbol{J}_V(\boldsymbol{x}_k^{(N)}) \right\}^{-1} \boldsymbol{J}_V^T(\boldsymbol{x}_k^{(N)}) \boldsymbol{C}_k^{-1} \boldsymbol{V}_k \tag{25}$$

where $\boldsymbol{J}_V(\hat{\boldsymbol{x}}_k) = \partial \boldsymbol{V}(\boldsymbol{x}_k) / \partial \boldsymbol{x}_k |_{\boldsymbol{x}_k = \hat{\boldsymbol{x}}_k}$. Using the matrix inversion' s lemmas, the covariance $\boldsymbol{P}_k$ at k time can be obtained:

$$\boldsymbol{P}_k = \left[\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{J}_h(\hat{\boldsymbol{x}}_k^{(N)}) \right] \bar{\boldsymbol{P}}_k \tag{26}$$

where the gain $\boldsymbol{K}_k$ is defined as:

$$\boldsymbol{K}_k = \bar{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\hat{\boldsymbol{x}}_k^{(N)}) \left\{ \boldsymbol{J}_h(\boldsymbol{x}_k^{(N)}) \bar{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\boldsymbol{x}_k^{(N)}) + \boldsymbol{R}_k \right\}^{-1} \tag{27}$$

Covariance and cross-covariance are defined as:

$$\boldsymbol{P}_{xz} = \overline{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\hat{\boldsymbol{x}}_k^{(N)}) \tag{28}$$

$$\boldsymbol{P}_{zz} = \boldsymbol{J}_h(\hat{\boldsymbol{x}}_k^{(N)})\overline{\boldsymbol{P}}_k \boldsymbol{J}_h^T(\boldsymbol{x}_k^{(N)}) + \boldsymbol{R}_k = \left[\boldsymbol{J}_h(\boldsymbol{x}_k^{(N)})\overline{\boldsymbol{S}}_k \quad \boldsymbol{S}_{R,k}\right]\left[\boldsymbol{J}_h(\boldsymbol{x}_k^{(N)})\overline{\boldsymbol{S}}_k \quad \boldsymbol{S}_{R,k}\right]^T \tag{29}$$

Using a series of manipulations, we can obtain a square root $\boldsymbol{S}_k$ of covariance $\boldsymbol{P}_k$:

$$\boldsymbol{S}_k = Chol(\left[\overline{\boldsymbol{S}}_k - \boldsymbol{K}_k \boldsymbol{J}_h(\hat{\boldsymbol{x}}_k^{(N)})\overline{\boldsymbol{S}}_k \quad \boldsymbol{K}_k \boldsymbol{S}_{R,k}\right]) \tag{30}$$

3.2 L-M based Iterative square root cubature Kalman filter

Suppose that the state distribution at k-1 time is $\mathbf{x}_{k-1} \sim \mathcal{N}(\hat{\mathbf{x}}_{k-1}, \mathbf{S}_{k-1}\mathbf{S}_{k-1}^T)$, L-M based iteration square root cubature Kalman filter (ISRCKFLM), which includes two process: time update and measurement update, is describe as follows.

1) Time Update

(1) Calculate the cubature points and the predicted $\overline{\boldsymbol{x}}_k$ and $\overline{\boldsymbol{S}}_k$ using Equations (3)-(6) at k-1 time.

(2) Evaluate the modified covariance:

$$\tilde{\boldsymbol{P}}_k = \left[\boldsymbol{I} - \overline{\boldsymbol{S}}_k \overline{\boldsymbol{S}}_k^T \left(\overline{\boldsymbol{S}}_k \overline{\boldsymbol{S}}_k^T + \frac{1}{\mu_i}\boldsymbol{I}\right)^{-1}\right]\overline{\boldsymbol{S}}_k \overline{\boldsymbol{S}}_k^T \tag{31}$$

2) Measurement update

(1) Set the initial value as: $\hat{\boldsymbol{x}}_k^{(0)} = \overline{\boldsymbol{x}}_k$.

(2) Assuming the i-th iterate $\hat{\boldsymbol{x}}_k^{(i)}$, using equation (24) to calculate the matrix $\boldsymbol{L}_k^{(i)}$.

(3) Using equation (23) to calculate the iterate $\hat{\boldsymbol{x}}_k^{(i+1)}$.

(4) Calculate the iteration termination condition

$$\left\|\hat{\boldsymbol{x}}_k^{(i+1)} - \boldsymbol{x}_k^{(i)}\right\| \le \varepsilon \text{ or } i = N_{\max} \tag{32}$$

ε and $N_{\max}$ are predetermined threshold and maximum iterate number, respectively. If the termination condition meets, continue to 5); otherwise: set $\hat{\boldsymbol{x}}_k^{(i)} = \boldsymbol{x}_k^{(i+1)}$ and return to 3).

(5) Mark iterate number N when the termination condition meets and calculate the state estimation at k time instant

$$\hat{\boldsymbol{x}}_k = \boldsymbol{x}_k^{(N)} \tag{33}$$

(6) Evaluate the cross-covariance and square root of innovation covariance at k time

$$\boldsymbol{P}_{xz} = \overline{\boldsymbol{S}}_k \overline{\boldsymbol{S}}_k^T \boldsymbol{J}_h^T(\hat{\boldsymbol{x}}_k^{(N)}) \tag{34}$$

$$\boldsymbol{S}_{zz} = Chol(\left[\boldsymbol{J}_h(\hat{\boldsymbol{x}}_k^{(N)})\bar{\boldsymbol{S}}_k \quad \boldsymbol{S}_{R,k}\right]) \tag{35}$$

(7) Calculate the square root of covariance at k time

$$\boldsymbol{K}_k = \boldsymbol{P}_{xz} / \boldsymbol{S}_{zz}^T / \boldsymbol{S}_{zz} \tag{36}$$

And $\boldsymbol{S}_k$ is calculated using the Equation (30).

The ISRCKFLM algorithm inherits the virtues of SRCKF which has better numerical stability. The measurement update of the ISRCKLM algorithm is transformed to the nonlinear least-square problem; the optimum state estimation and covariance are solved using L-M method with better performance. The sequences obtained have the global convergence.

4. Applications to State Estimation for Re-Entry Ballistic Target

To demonstrate the performance of the ISRCKFLM algorithm, we apply the ISRCKFLM to estimate state of re-entry ballistic target with unknown ballistic coefficient and compare its performance against the SRCKF and iterate square root cubature Kalman filter using Gauss-Newton method (ISRCKF) algorithms. All the simulations were done in MATLAB on a ThinkPad PC with an Intel (R) CORE i5 M480 processor with the 2.67GHz clock speed and 3GB physical memory.

In the simulation, the parameters and the initial state estimate are the same as in [10]. To demonstrate the performance of the ISRCKFLM algorithm, we use the root-mean square error (RMSE) and average accumulated mean-square root error (AMSRE) in the position, velocity and ballistic coefficient introduced in [8]. Figure. 1, Figure. 2 and Figure. 3 show the RMSEs for the SRCKF, ISRCKF and ISRCKFLM (u =10-10) in position, velocity and ballistic coefficient in an interval of 15s-58s. The AMSREs of the three filters in position, velocity and and ballistic coefficient are listed in Table. 1. The iteration number selected in the ISRCKFLM and ISRCKF algorithms is 4. All performance curves and figures in this subsection were obtained by averaging over 100 independent Monte Carlo runs. All the filters are initialized with the same condition in each run.

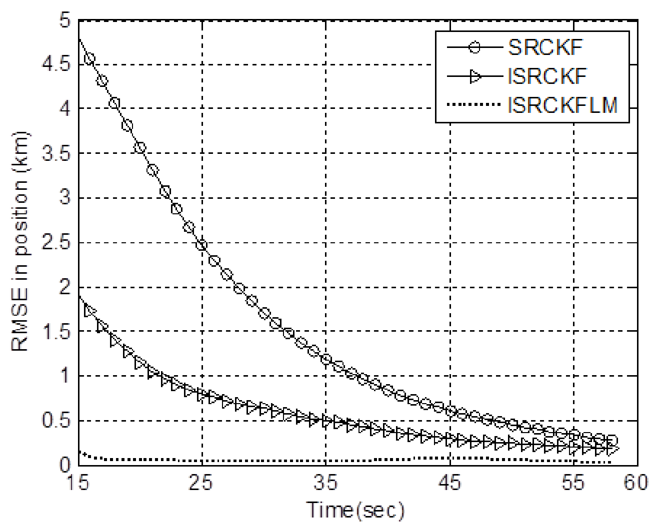

Figure.1 RMSEs in position for various filters

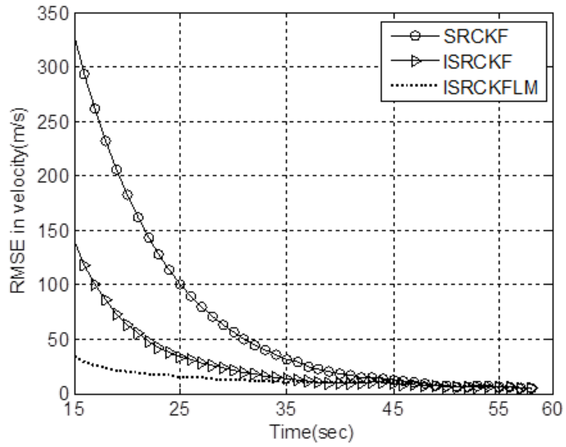

Figure.2 RMSEs in velocity for various filters

From Figure.1, we can see that the RMSE of ISRCKFLM in position is far less than that of SRCKF algorithm, and is less than that of ISRCKF algorithm. Moreover, the ISRCKFLM needs 14.5 seconds to make the RMSE in position reduce below 500 meters, the ISRCKF algorithm needs 34.6 seconds, and SRCKF algorithm needs about 47.6 seconds, so the ISRCKFLM algorithm has faster convergence rate than the SRCKF and ISRCKF algorithms. So the estimates provided by the ISRCKFLM in the position and velocity are markedly better than those of SRCKF and ISRCKF algorithms.

Observe from Figure.2, the RMSE of ISRCKFLM in velocity is far less than those of SRCKF and ISRCKF algorithm in the interval time (t < 35s), the ISRCKFLM still has faster convergence rate. And the RMSEs of the three filters lie at the lower level in the period (t >35s).

As to the estimation of the ballistic coefficient, in the Figure.3, the RMSEs of the three filters have less improvement in the interval time (0< t <35s) because of having less effective information about it from the noisy measurement. The RMSEs of the three filters begin to decrease at about t=37s because the measurements have the effective information on ballistic coefficient. In the period (35s< t <45s), the RMSE of the ISRCKFLM algorithm for the ballistic coefficient decreases more rapidly than that of SRCKF, and decreases at the same rate as that of ISRCKF. At the period 45s< t < 58s, the RMSE in the ISRCKFLM algorithm decreases most rapidly among the three algorithms. The ballistic coefficient estimate in the ISRCKFLM algorithm has the great improvement.

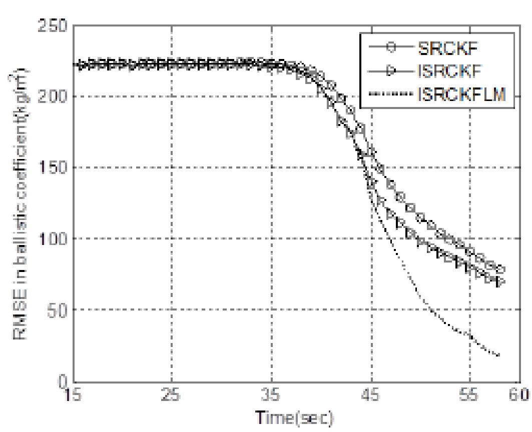

Figure .3 RMSEs in ballistic coefficient for various filters

Table 1 AMSREs in position, velocity and ballistic coefficient

Algorithms	AMSREp (m)	AMSREv (m/s)	AMSRE (kg/m2)
SRCKF	2693.096	306.133	165.363
ISRCKF	1457.078	250.900	162.530
ISRCKFLM	856.993	220.296	160.658

According to Figure.1-Figure.3, the RMSEs of ISRCKFLM in position and velocity markedly decrease, compared with those of the SRCKF and ISRCKF algorithm. Although the RMSE of ISRCKFLM in ballistic coefficient has less improvement, its RMSE significantly reduces in the last period. So the ISRCKFLM improves the state estimation accuracy of re-entry ballistic target.

From Table.1, it is seen that, the ISRCKFLM’s AMSRE in position reduces by about 68%, and its AMSRE in velocity reduces by about 28% compared to SRCKF. And compared to ISRCKF, the AMSRE of ISRCKFLM algorithm in position decreases by about 41%, and its AMSRE in velocity decreases by about 12%. Table.1 shows ISRCKFLM’s AMSRE in ballistic coefficient reduces marginally, but Figure.3 shows the ISRCKFLM’s RMSE is less than the other two filters in the interval of 40s-58s. Hence, the ISRCKFLM is to be preferred over the other filters in the light of AMSREs in the position, velocity and ballistic coefficient and has better performance.

Therefore, on the basis of the simulation results presented in Figure.1-Figure.3 and Table.1, one can draw a conclusion that the ISRCKFLM algorithm yields on the superior performance over the SRCKF and ISRCKF algorithms on state estimation of re-entry ballistic target.

5. Conclusion

In this study, we develop a more accurate nonlinear filtering named the L-M method based iteration square root cubature Kalman filter (ISRCKFLM). The measurement update of the ISRCKFLM algorithm is transformed to nonlinear least square problem using predicted state estimation and covariance as initial value, and then the optimal state estimation is obtained using the L-M method. The ISRCKFLM algorithm has the advantages of global convergence, fast covergence and numerical stability. The ISRCKFLM algorithm is applied to state estimation for re-entry ballistic target. Simulation results demonstrate that the performance of ISRCKFLM algorithm is superior to SRCKF and ISRCKF algorithms. So the ISRCKFLM algorithm is much more effective and improves the performance of state estimation to a marked degree.

Acknowledgments

The authors would like to thank the support of the State and Local Joint Laboratory of Advanced Network and Monitoring Control Engineering (No. GSYSJ201603).

References

[1] Bar-Shalom, Y.; Li, X.R. Kirubarajan, T. Estimation with Applications to Tracking and Navigation. New York: John Wiley &Son, 2001.

[2] Grewal, M.S.; Andrews, A.P. Kalman filtering: theory and practice using Matlab. New York: John Wiley & Sons, 2008.

[3] Julier, S. J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. Proceedings of IEEE. 2004, 92(12): 401-422.

[4] Arasaratnam, I.; Haykin, S.; Hurd, T.R. Cubature Kalman filtering for continuous-discrete systems: theory and simulations. IEEE Transaction on Signal Processing. 2010, 58(10): 4977-4993.

[5] Arasaratnam, I.; Haykin, S. Cubature Kalman filters. IEEE Transactions on. Automatic Control. 2009, 54 (6): 1254-1269.

[6] Zhang, Y.G.; Huang, Y. L.; Li, N.; Zhao, L. Interpolatory cubature Kalman filters. IET Control Theory & Applicaitons. 2015, 9 (11): 1731 – 1739.

[7] Jafar, Z.; Ehsan, S. Convergence analysis of non-linear filtering based on cubature Kalman filter. IET Science Measurement & Technology. 2015, 9 (3):294 – 305.

[8] Mu Jing, Cai Yuanli, Wang Changyuan. L-M Method Based Iteration Cubature Kalman Filter and Its Applications.Journal of Xi’an Technoligical University, 2013, 33(1):1-6.

[9] Chandra, K.P.B.; Da-Wei, G.; Postlethwaite, I. Square root cubature information filter, IEEE Sensors Journal. 2013, 13 (2):750 – 758.

[10] Mu. J.; Cai. Y. Likelihood-based iteration square-root cubature Kalman filter with applicaitons to state estimation of re-entry ballistic target. Transactions of the Institute of Measurement and Control. 2013, 35(7): 949-958.