A Data Fusion Algorithm and Simulation Based on TQMM

Ke Zhang[1,2](✉), Zeyang Wang[3], and Huiling Li[4]

[1] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
kezhang@uestc.edu.cn

[2] Science and Technology on Electronic Information Control Laboratory, Chengdu 610000, China

[3] Sichuan Jiuzhou Electronic Technology Co., Ltd., Mianyang 621000, China

[4] School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Abstract. Asynchronous data fusion is more practical than synchronous data fusion, the model of track-to-track fusion in this case has been established and the concept of Track Quality with Multiple Model (TQMM) was put forward, furthermore a data fusion algorithm is proposed, in which the TQMM is used to assign weights, to improve tracking precision in asynchronous multi-sensor data fusion system. The simulation results show that the algorithm has a better tracking performance compared with original algorithms.

Keywords: Data fusion · Multi-sensor · Track fusion · TQMM

1 Introduction

Generally, asynchronous track fusion is mainly divided into two categories. One is that different kinds of sensors have different and fixed sampling periods and the other one is the time interval of target information provided by sensors has no rule, meaning sensors have no fixed sampling interval. Due to the limitation of the sensor itself, the first category can also be divided into two parts according to the starting time of different sampling periods. In both cases, sensor information can be synchronized through track pretreatment, and then be tracked by synchronous track fusion algorithm. However, the pretreatment process will cause errors increasing and reduce the fusion data reliability. Therefore, researchers put forward a series of asynchronous track fusion algorithm [1–10]. Some asynchronous fusion algorithms introduce data registration method to the fusion algorithm for realizing the synchronization of asynchronous data before fusion, such as the least squares method, interpolation, extrapolation and so on. Besides, some algorithms deal with asynchronous data on the basis of receiving time, and select the proper fusion approach for asynchronous data fusion, such as fusion algorithm under the principle of minimum error covariance matrix trace [1, 2], asynchronous track fusion algorithm based on information matrix [3–5], distributed weighted fusion

H. Song et al. (Eds.): SIMUtools 2019, LNICST 295, pp. 205–217, 2019.
https://doi.org/10.1007/978-3-030-32216-8_20

estimators with random delays [6], time-varying bias estimation for asynchronous multi-sensor multi-target tracking systems [7] and Step by Step Prediction Fusion based on Asynchronous Multi-sensor System (SSPEA) [8, 9], etc. These algorithms also can be used in other research domain, such as traffic analysis [11, 12], big data analysis [13] and smart transportation [14, 15], etc. But, with these algorithms, the first kind of asynchronous problem could be basically solved; while, the second problem could not be solved well.

The SSPFA algorithm mainly uses the multi-sensor's measurement information in a fusion cycle to get the filtering estimation, in order to obtain the local state estimation and the corresponding error covariance of each sensor at the last moment of fusion cycle. Then after the state prediction of fusion time, the algorithm operates the order weighting of the sensor prediction information based on the obtaining order of sensor predictive values and the principle of minimum error covariance matrix. And finally, the multi-sensor asynchronous fusion is achieved. According to the filtering predictive thought of SSPFA, [9] proposes a Track - to - Track Fusion for Asynchronous Multi-sensor based on Step by Step Prediction (TFASP) algorithm. By the local state estimation of multi-sensor fusion, the algorithm predicts sampling values in the fusion cycle. Then, after the weight fusion of same sensor's predictive value at fusion moment, this algorithm regards the fusion value as sensor's equivalent observation information at fusion moment and finally achieves the global fusion of asynchronous multi-sensor by the step-by-step filter fusion. As the input value of the step-by-step filter fusion, local sensor's weight fusion decides the tracking performance of the algorithm. However, determined by the observation precision and sensors' prediction error, the weight of local sensor's weight fusion has no direct relation with the time tag between the sampling time and fusion time. Therefore, the large error of local sensor's state estimation will reduce the tracking accuracy of the whole system. Besides, there is no feedback mechanism in the entire system. These problems cause some shortcoming in TFASP algorithm. Therefore, based on Track Quality of Kalman filtering [16, 17], with the combination of the Track Quality with Multiple Model(TQMM) [18] and introduction of feedback mechanism into the system [19], this paper presents an asynchronous multi-sensor track fusion algorithm with information feedback, that is Asynchronous Fusion based on Track Quality with Multiple Model(AFTQMM) algorithm. AFTQMM feeds back the one-step prediction of global state estimation to local sensors. And then after getting TQMM of all sampling points based on this feedback, local sensors assign the weight according to TQMM of each point, which improves the accuracy of equivalent observations of local sensors at fusion moment as well as the performance of global state estimation.

2 Track Quality with Multiple Model and Local Tracking

2.1 Track Quality with Multiple Model

Assuming that the dynamic equation and measurement equation of multi-sensor system are:

$$X^l(k+1) = F^l(k)X^l(k) + w^l(k) \tag{1}$$

$$Z^l(k) = H^l(k)X^l(k) + V^l(k), \quad l{=}1, 2, \cdots, \ NUM \tag{2}$$

In equations, NUM is the amount of the filter models, and $X^j(k+1)$ stands for the state vector of model l in $k+1$ moment. $F^l(k)$ represents the one-step state transition matrix from moment k to moment $k+1$ under model l, and the system process noise $w^l(k)$ is gaussian white noise sequence. Besides,

$$E\left[w^l(k)\right] = 0 \tag{3}$$

$$Cov\left(w^l(k), w^l(\tau)\right) = E\left[w^l(k)w^{l^T}(\tau)\right] = Q^l(k)\delta_{k\tau} \tag{4}$$

In equations, $Q^l(k)$ is nonnegative definite matrix, and $Z^l(k)$ represents the sensor's observed values of target state under model l. $H^l(k)$ is measurement matrix, and measurement noise $V^l(k)$ stands for gaussian white noise sequence. Besides,

$$E\left[v^l(k)\right] = 0 \tag{5}$$

$$Cov\left(v^l(k), v^l(\tau)\right) = E\left[v^l(k)v^{l^T}(\tau)\right] = R^l(k)\delta_{k\tau} \tag{6}$$

In equations, $R^l(k)$ is the positive definite matrix. System process noise and measurement noise are independent of each other, that is, to meet

$$Cov\left(w^l(k), v^l(\tau)\right) = 0 \quad \tau = 1, 2, \cdots, k, \cdots \tag{7}$$

Local track quality determines the track quality of system, which means the track quality of system after fusion will not be too high [11] if local track quality is poor. Assuming that the one-step prediction and its covariance of the state of model l $(l = 1,\ 2,\ \cdots,\ NUM)$ in time k are $\hat{X}^l(k+1|k)$ and $P^l(k+1|k)$ respectively, then the state's one-step prediction and covariance of model l $(l = 1,\ 2,\ \cdots,\ NUM)$ of sensor i $(i = 1,\ 2,\ \cdots,\ N)$ in $k+1$ time based on l $(l = 1,\ 2,\ \cdots,\ NUM)$ model state in k time are

$$v^l(k+1) = Z(k+1) - H^l(k+1)\hat{X}^l(k+1|k) \tag{8}$$

$$S^l(k+1) = H^l(k+1)P^l(k+1|k)H^l(k+1)^T + R^l(k) \tag{9}$$

In order to describe the track quality, a standardized distance equation [12] could be defined

$$d^l(k+1) = v^l(k+1)^T S^l(k+1)^{-1} v^l(k+1) \tag{10}$$

Then, the track quality of model l in time $k+1$ is

$$U^l(k+1) = \alpha U^l(k) + (1-\alpha) d^l(k+1) \tag{11}$$

The value of U represents the track quality. Obviously, the smaller the U is, the better the track quality is. Here, α is the historical power factor with the range from 0 to 1, and $\alpha = 1/5$ in the simulation. When $k+1 = 4$, the track quality of sensor i in model l is

$$U^l(4) = d^l(4) \tag{12}$$

Therefore, TQMM of sensor i in time $k+1$ is

$$U(k+1) = \sum_{j=1}^{N} U^l(k+1) u_{k+1}(l) \tag{13}$$

2.2 Local Tracking

In order to adapt the target mobility and get precise local estimate information, Interacting Multiple Model (IMM) filtering algorithm is adopted for the local track of sensors. Besides, three kinds of IMM filtering algorithm are applied to reduce the computational complexity and improve the realtime performance of information processing. Among them, the system state vector is $X = [x\ \dot{x}\ \ddot{x}\ y\ \dot{y}\ \ddot{y}\ z\ \dot{z}\ \ddot{z}]^T$, and the model prior probability is $U = [1\ \ 0\ \ 0]$. The total output of IMM filters is the weighted average of multi-filters' filtering results, and the weight is the model probability. If one model plays a dominant role, then it will enjoy higher probability (between 0.9 and 1), and the others only obtain lower probabilities (between 0 and 0.1). Besides, the transition probability of Markov model is

$$P_{ij} = \begin{bmatrix} 0.95 & 0.025 & 0.025 \\ 0.025 & 0.95 & 0.025 \\ 0.025 & 0.025 & 0.95 \end{bmatrix} \tag{14}$$

3 Asynchronous Fusion Algorithm Based on Track Quality with Multiple Model

3.1 Basic Flow

The main idea of this algorithm includes: the observed filter prediction at fusion moment is gotten based on the sensors' local estimated information; then, using the weight fusion of same sensor's prediction to obtain sensors' observed information at fusion moment; finally, global fusion of asynchronous multi-sensor is achieved based on step by step filtering fusion process. The process is shown in Fig. 1.

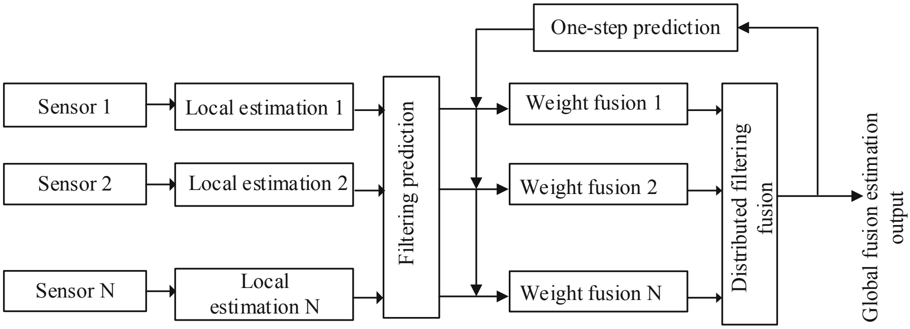

Fig. 1. Algorithm flow chart of step by step asynchronous track fusion based on track quality

3.2 Fusion Model

According to Fig. 1, assuming that the fusion period is T, and the algorithm's basic fusion model is given when the amount of sensors is N in a fusion cycle, as shown in Fig. 2.

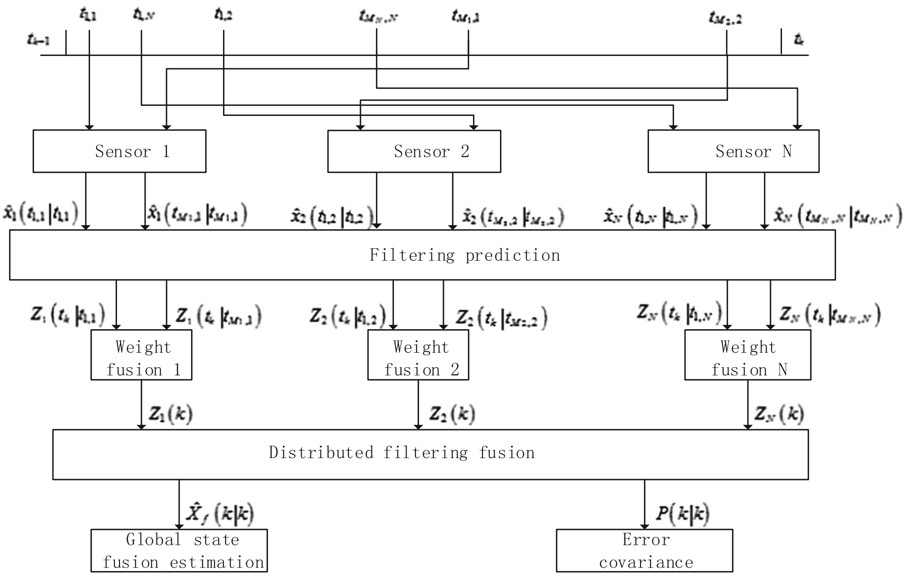

Fig. 2. Asynchronous fusion models

3.3 Steps of Algorithm

As shown in Figs. 1 and 2, the algorithm includes four parts: multi-model prediction, feedback element, local sensors' weight fusion and distributed filtering fusion. The process is shown below.

Assuming that the global state fusion estimation and the corresponding error covariance of system are $\hat{X}_f(k-1|k-1)$ and $P(k-1|k-1)$ in fusion time t_{k-1}. There are N $(N \geq 0)$ sensors with observed information in fusion period, and the observation value of sensor i $(i = 1, 2, \cdots, N)$ is M_i $(M_i \geq 0)$. The fusion model in Fig. 2 is built on the basis of each sensor with more than 2 sampling points, which means $M_i \geq 2$. However, due to the randomness of the data provided by the sensors, there are two cases in fusion period. One is $N = 0$, which means the fusion center cannot obtain the continuous target information in a certain interval. Another one is $N > 0$. In this case, based on the observation number, M_i can be divided into two parts: $M_i = 1$ and $M_i \geq 2$. For different situations, different approaches are adopted to optimize the fusion process.

When $N = 0$, the state estimation of current fusion moment is achieved based on the state estimation of precious fusion moment. However, when $N = 0$ continuously exists, using this method to get information will reduce the fusion algorithm's effectiveness because of the accumulation of prediction error. With the improvement of their performance, sensors will be chosen to detect tracking object in all directions, to avoid $N = 0$ in the fusion center. When $N > 0$, if $M_i = 1$, the prediction and the step-by-step filtering fusion could be taken directly without weighted fusion process. While, if $M_i \geq 2$, the algorithm can be operated following the asynchronous fusion model in Fig. 2. Besides, if the observed information of some sensors at fusion moment exists, the information could be applied directly to participate the step-by-step filtering fusion.

Now, there are two known issue. Firstly, the number of sensors and the observed number in fusion period $(t_{k-1}, t_k]$ are N and M_i $(i = 1, 2, \cdots, N)$. Besides, the target state estimation and the corresponding covariance error of sensor i in observed time $t_{j,i}(j = 1, 2, \cdots, M_i)$ are $\hat{x}_i(t_{j,i}|t_{j,i})$ and $p_i(t_{j,i}|t_{j,i})$. The process of getting the state estimation $\hat{X}_f(k|k)$ and the covariance error $P(k|k)$ of system track in fusion center at time t_k will be introduced.

3.3.1 Multi-model Prediction

Judge the value of N. When $N \neq 0$, search all sampling points $[t_{1,i}, t_{2,i}, \cdots, t_{M_i,i}]$ of sensor i in fusion circle, and then operate one step test based on 3 models, and predict all sampling pinots to fusion moment t_k. The process is as follow:

$$\Delta t_{j,i} = t_k - t_{j,i}, \quad j = 1, 2, \cdots M_i \tag{15}$$

In the equation, the local state estimation and the error covariance of sensors in time $t_{j,i}$ are $\hat{x}_i(t_{j,i}|t_{j,i})$ and $p_i(t_{j,i}|t_{j,i})$. Based on time difference, the corresponding state transition matrix $F^l_{j,i}(t_{j,i})$ $(l = 1,\ 2,\ 3)$ could be gotten by IMM filtering idea, and then the observation predictive value could be worked out.

$$Z_i^l(t_k|t_{j,i}) = H_i^l(k) \cdot F_{j,i}^l\left(t_{j,i}\right) \cdot \hat{x}_i(t_{j,i}|t_{j,i}), \quad l = 1,\ 2,\ 3 \tag{16}$$

In the equation, $H_i^l(k)$ is the observation matrix of sensor i's model l. The multi-model prediction could be obtained based on the observed prediction $Z_i^l(t_k|t_{j,i})$ of model l

$$Z_i(t_k|t_{j,i}) = \sum_{l=1}^{3} Z_i^l(t_k|t_{j,i}) \cdot u_l^i(k) \tag{17}$$

In the equation, $u_l^i(k)$ is the probability of sensor i's model l at time t_k.

3.3.2 Feedback Element

Operate one step prediction for system state estimation in time t_{k-1}. The state vector and the covariance of that are

$$\hat{X}_f(k|k-1) = \sum_{l=1}^{3} \hat{X}_f^l(k|k-1) \cdot u^l(k) \tag{18}$$

$$P(k|k-1) = \sum_{l=1}^{3} u^l(k) \cdot \left\{P^l(k|k-1) + \left[\hat{X}_f^l(k|k-1) - \hat{X}_f(k|k-1)\right] \cdot \left[\hat{X}_f^l(k|k-1) - \hat{X}_f(k|k-1)\right]'\right\} \tag{19}$$

In equations, $u_l(k)$ is the probability of system's model at time t_k. $\hat{X}_f^l(k|k-1)$ and $P^l(k|k-1)$ are the one step prediction and error covariance of system track at time t_{k-1} based on model l. The expressions are as follows.

$$\hat{X}_f^l(k|k-1) = F^l(k-1)\hat{X}_f^l(k-1|k-1) \tag{20}$$

$$P^l(k|k-1) = F^l(k-1)P^l(k-1|k-1)F^l(k-1)^T + Q^l(k-1) \tag{21}$$

$F^l(k-1)$, $\hat{X}_f^l(k-1|k-1)$ and $P^l(k-1|k-1)$ are the state transition matrix, state estimation and error covariance of system track's model l at time t_{k-1}.

3.3.3 The Weight Fusion of Local Sensors

The feedback element will get one step prediction of the previous state of fusion center, and feeds back to local sensors. Then, after getting the TQMM of local sensors' states prediction, the feedback could further determine the observation state's weight and realize the weight fusion. The deterministic process of the weight factor is shown in Fig. 3.

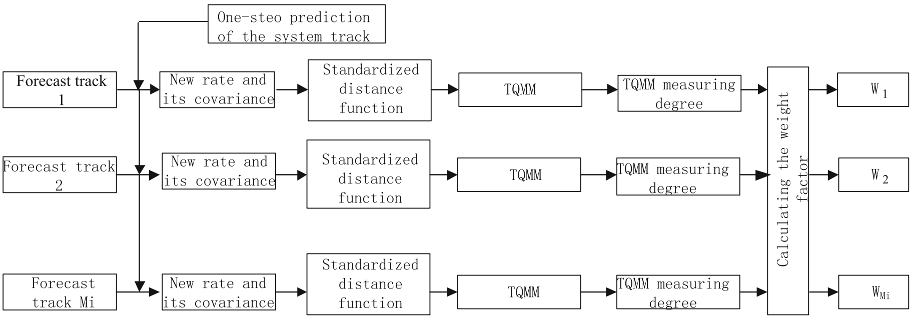

Fig. 3. The flow chart of the distribution of the weight factor

In fusion period $(t_{k-1}, t_k]$, the innovation and covariance of sensor $i(i = 1,\ 2,\ \cdots,\ N)$'s observation from time $t_{j,i}(j = 1,\ 2,\ \cdots,\ M_i)$ to fusion time t_k are

$$v^l_{i,j}(k) = z_i(t_k|t_{j,i}) - H^l_i(k)\hat{X}_f(k|k-1) \tag{22}$$

$$S^l_{i,j}(k) = H^l(k)P(k|k-1)H^l_i(k)^T + R^l(k-1) \tag{23}$$

Based on Sect. 2.1, TQMM of the sensor i's sampling point j is $U_{i,j}(k)$.Then the measuring degree of TQMM of sensor i's sampling point j could be calculated.

$$h^j_i(k) = \exp\{-U_{i,j}(k)\} \tag{24}$$

The corresponding weight is

$$\omega^i_j(k) = h^j_i(k) / \sum_j^{M_i} h^j_i(k) \tag{25}$$

Finally, by the weight fusion, the equivalent observation data of sensor i in time t_k is obtained.

$$Z_i(k) = \sum_{j=1}^{M_i} w^j_i(k) Z_i(t_k|t_{j,i}) \tag{26}$$

3.3.4 Distributed Filtering Fusion

From steps above, we can get the observation information $Z_1(k), Z_2(k), \cdots Z_N(k)$ of N sensors at fusion time t_k. Then, the global state fusion estimation and the corresponding error covariance at fusion moment could be worked out with the step by step fusion thought [16].

$$\begin{cases} \hat{X}_f(k|k) = \hat{X}_N(k|k) \\ P(k|k) = P_N(k|k) \end{cases} \tag{27}$$

With the known information of $\hat{X}_f(k|k-1) = F(k-1)\hat{X}_f(k-1|k-1)$ and $P(k|k-1) = F(k-1)P(k-1|k-1)F(k-1)^{\mathrm{T}} + GQG^{\mathrm{T}}$, the actual expressions of Eq. (27) are as follows. When $N = 1$, assuming $\hat{X}_1(k|k-1) = \hat{X}_f(k|k-1)$ and $P_1(k|k-1) = P(k|k-1)$, then

$$\hat{X}_1(k|k) = F(k-1)\hat{X}_f(k-1|k-1) + K_1(k)\left[Z_1(k) - H\hat{X}_1(k|k-1)\right] \tag{28}$$

$$P_1(k|k) = [I - K_1(k)H(k)]P_1(k|k-1) \tag{29}$$

When $N \geq 2$,

$$\hat{X}_N(k|k) = F(k-1)\hat{X}_f(k-1|k-1) + \sum_{i=1}^{N}\left\{K_i(k)\left[Z_i(k) - H(k)\hat{X}_i(k|k-1)\right]\right\} \tag{30}$$

$$P_N(k|k) = \left\{\prod_{i=1}^{N}[I - K_i(k)H(k)]\right\}P_1(k|k-1) \tag{31}$$

From Eq. (28) to (31), $K_i(k)$ is the filtering gain matrix of sensor $i(i = 1, 2, \cdots, N)$, and its calculating formula is shown as the following

$$K_i(k) = P_i(k|k-1)H(k)^{\mathrm{T}}\left[H(k)P_i(k|k-1)H(k)^{\mathrm{T}} + R_i(k)\right]^{-1} \tag{32}$$

When $i = 2, \cdots, N$, we can get $\hat{X}_i(k|k-1) = \hat{X}_{i-1}(k|k)$ and $P_i(k|k-1) = P_{i-1}(k|k)$.

4 Simulation Analysis

4.1 Simulation Environment

For comparative analysis, Root Mean Square Error (RMSE) and Trace of Error Covariance Matrix (TECM) are chosen as the target tracking performance index.

Assuming that six radars on the same platform observe the same target asynchronously, the sampling time of track data got by fusion center may deviate from the fixed sampling period, due to the sensor limitation and the communication time-delay from local node to fusion center. So, we should pay attention to the offset Δt of sensor from the actual sampling period to the fixed sampling period. Besides, there is no sampling information at some sampling moments because of the target escaping from the tracking area of the corresponding radar. The tracking fusion problem in this situation is the typical second kind of asynchronous fusion problem. Each sensor correlates the observed data to form a target track and reports the track and the data to the fusion center. However, due to the disunity of measurement error and observation coordinates of each sensor, the data from each sensor needs preprocessing before fusion, generally including data space alignment, gross error rejection and so on.

The target track time is 100 s, and the monte carlo simulation will be done for 600 times ($M = 600$). Then the expression of RMSE and TECM are

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{n=1}^{M}\left((x-\hat{x}^n)^2+(y-\hat{y}^n)^2+(z-\hat{z}^n)^2\right)}{M}} \tag{33}$$

$$\mathrm{TECM} = \left(\sum_{n=1}^{M} trace(P^n)\right) \Big/ M \tag{34}$$

In the equation, $\hat{x}^n$, $\hat{y}^n$, $\hat{z}^n$ are the location information of the n th simulation fusion track, and P^n is the error covariant matrix of the n th simulation track.

4.2 Results and Analysis

The sampling periods of six radars are 0.2 s, 0.5 s, 0.8 s, 1.0 s, 1.2 s and 1.5 s. The observation precisions of radars on x, y, z directions are 50.23 m, 51.15 m, 55.57 m, 50.28 m, 57.69 m, 51.59 m and their locations are (2800 m, 0 m, 0 m), (0 m, 500 m, 0 m), (0 m, 0 m, 1800 m), (50 m, 100 m, 500 m), (50 m, 100 m, 2800 m), (100 m, 500 m, 800 m). The initial position of target is (−3000 m,1000 m,−4000 m), with the initial speed of 100 m/s. The target flies at a constant speed in 0–20 s, carry turning maneuver at a speed of 0.157 rad/s in 20–40 s, then flies at a constant speed in 40–60 s, and does turning maneuver at a speed of −0.157 m/s in 60–80 s. Finally, it flies at a constant speed in 80–120 s. The total flying time of the target is 120 s, and the flight path is shown in Fig. 4.

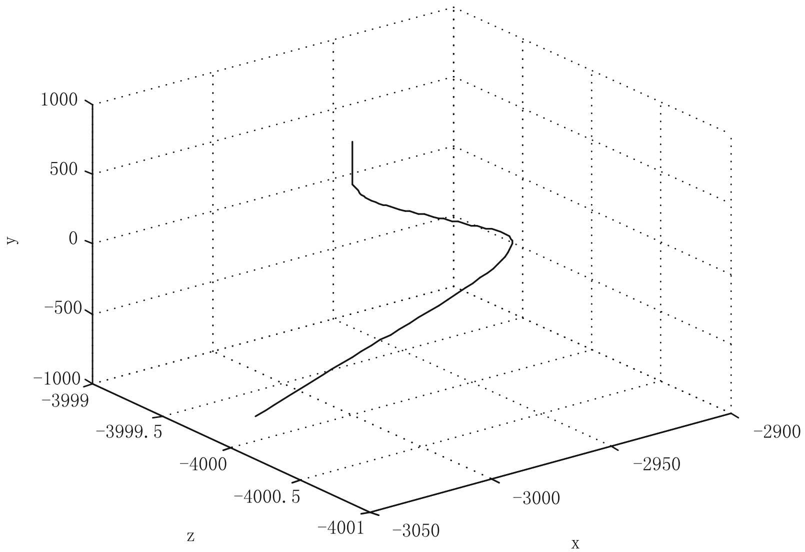

Fig. 4. The flight path of the maneuvering target

Systems with the number of sensors from 3 to 6 track the maneuvering target simultaneously (the fusion period of four systems is 1.0 s) to test the influence of the sensor's number on the track performance of AFTQMM (Fig. 5 and Chart 1).

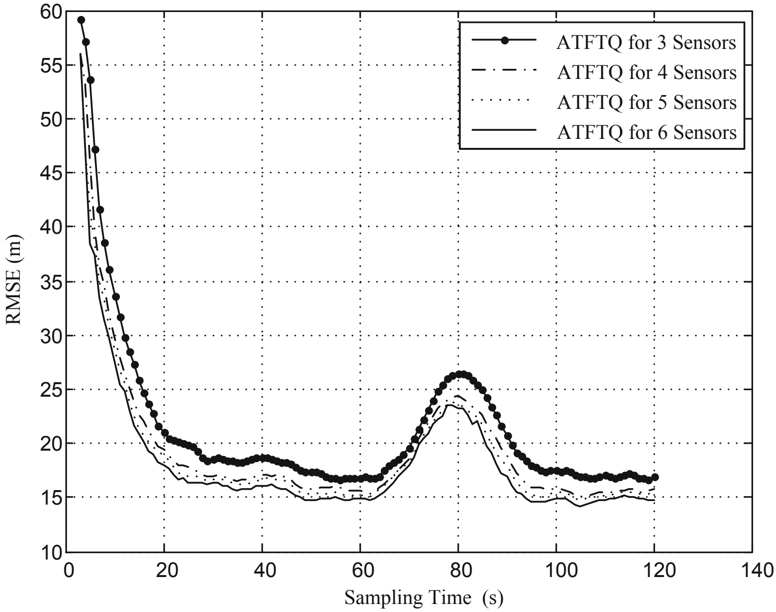

Fig. 5. The relationship of RMSE and the number of sensors in AFTMM algorithm

Track method	Local track	AFTQMM (3 Sensors)	AFTQMM (4 Sensors)	AFTQMM (5 Sensors)	AFTQMM (6 Sensors)
TECM average value (m^2)	4631	691	535	478	436

Chart 1. The relationship of TECM average value and the number of sensors in AFTQMM algorithm

The figures and the charts above proves that with the increasing of the number of sensors, the RMSE and TECH curves of AFTQMM decline, and the system track performance improve gradually. However, after the number is greater than 5, the system fusion accuracy has not been significantly improved. In engineering application, based on the relationship of track performance and systematic complexity, proper number of sensors can achieve the higher tracking accuracy, real-time processing and the project cost reducing as much as possible.

Acknowledgement. The paper is partially supported by the National Natural Science Foundation of China (Nos. 61571104), the 6th Innovation and Entrepreneurship Leading Talents Project of Dongguan, the General Project of Scientific Research of the Education Department of Liaoning Province (No. L20150174), and the Program for New Century Excellent Talents in University (No. NCET-11-0075), and Project of Science and Technology on Electronic Information Control Laboratory.

References

1. Liu, Q., Wang, X., Rao, N.S.V.: Information feedback for estimation and fusion in long-haul sensor networks. In: IEEE 2014 17th International Conference on Information Fusion (2014)
2. Lin, X.: One fusion-algorithm of asynchronous multi-sensor integrated navigation system. Geomat. Inf. Sci. Wuhan Univ. **37**(1), 54–57 (2012)
3. Aeberhard, M., Schlichtharle, S., Kaempchen, N., et al.: Track-to-track fusion with asynchronous sensors using information matrix fusion for surround environment perception. IEEE Trans. Intell. Transp. Syst. **PP**, 1–10 (2012)
4. Aeberhard, M., Rauch, A., Rabiega, M., et al.: Track-to-track fusion with asynchronous sensors and out-of-sequence tracks using information matrix fusion for advanced driver assistance systems. In: 2012 IEEE Intelligent Vehicles Symposium (IV), pp. 1–6 (2012)
5. Aeberhard, M., Schlichtharle, S., Kaempchen, N., et al.: Track-to-track fusion with asynchronous sensors using information matrix for surround environment perception. IEEE Trans. Intell. Transp. Syst. **PP**(99), 1–10 (2011)
6. Sun, S., Xiao, W.D.: Distributed weighted fusion estimators with random delays and packet dropping. Circ. Syst. Signal Process. **26**(4), 591–605 (2007)
7. Hu, Y., Zhou, D.: Time-varying bias estimation for asynchronous multi-sensor multi-target tracking systems using STF. Chin. J. Electron. **22**(3), 525–529 (2013)
8. Wen, C., Ge, Q.: The step by step predictive fusion of asynchronous multi-sensor system. J. Central South Univ. (Nat. Sci. Ed.) **32**(1), 652–653 (2005). (in Chinese with English Abstract)
9. Li, Q.: The research and implementation of airborne multi-sensor data fusion target's tracking technology. Mater's thesis. University of Electronic Science and Technology of China (2012). (in Chinese with English abstract)
10. Lim, S., Lee, C.: Data fusion algorithm improves travel time predictions. IET Intel. Transport Syst. **5**(4), 302–309 (2011)
11. Jiang, D., Wang, W., Shi, L., Song, H.: A compressive sensing-based approach to end-to-end network traffic reconstruction. IEEE Trans. Netw. Sci. Eng. (2018). https://doi.org/10.1109/tnse.2018.2877597
12. Jiang, D., Huo, L., Li, Y.: Fine-granularity inference and estimations to network traffic for SDN. PLoS ONE **13**(5), 1–23 (2018)
13. Jiang, D., Huo, L., Song, H.: Rethinking behaviors and activities of base stations in mobile cellular networks based on big data analysis. IEEE Trans. Netw. Sci. Eng. **1**(1), 1–12 (2018)
14. Jiang, D., Huo, L., Lv, Z., et al.: A joint multi-criteria utility-based network selection approach for vehicle-to-infrastructure networking. IEEE Trans. Intell. Transp. Syst. **PP**(99), 1–15 (2018)
15. Jiang, D., Zhang, P., Lv, Z., et al.: Energy-efficient multi-constraint routing algorithm with load balancing for smart city applications. IEEE Internet of Things J. **3**(6), 1437–1447 (2016)

16. Jeffery, T.: Track quality estimation for multiple-target tracking radars. In: IEEE Radar Conference, vol. PP, pp. 76–79 (1989)
17. Tafti, D., Sadati, N.: Novel adaptive Kalman filtering and fuzzy track fusion approach for real time applications. In: 3rd IEEE Conference on Industrial Electronics and Application, pp. 120–125 (2008)
18. Zhang, W., Wang, Z., Zhang, K.: Fusion algorithm based on multi-model track quality. Comput. Sci. **40**(2), 65–70 (2013). (in Chinese with English abstract)
19. Zhu, Y., You, Z., Li, X.R., et al.: The optimality for the distributed Kalman filtering fusion with feedback. Automatica (37), 1489–1493 (2001)