Multi-sensor weighted fusion algorithm based on improved TOPSIS

Sheng Tao¹* and Xia Hai-bao¹

1 Air Force Engineering University, Xi’an, Shanxi, 710038, China
*Corresponding author’s e-mail: 499399538@qq.com

Abstract. In the process of track fusion of multi-sensor target tracking, multi-sensor weighted fusion is used to overcome the shortcoming of incomplete single information source. In order to change the weight of each sensor participating in fusion in real time according to the merits and demerits of sensor information adaptively, a method of sensor weight calculation based on improved TOPSIS is proposed. Firstly, the local estimation and global prediction residuals of each sensor are composed into residuals matrix, from which the ideal solution is found, and then the relative closeness degree is calculated according to the grey correlation distance between the local estimation residuals and the ideal solution, so as to screen the sensors with high weight to participate in the fusion. The simulation results show that the tracking error of the algorithm after fusion is significantly reduced compared with the local estimation error of each sensor, and the tracking error of multiple sensors for the target can be realized.

1. Introduction

With the enrichment of observation means and the improvement of processing ability, the multi-sensor fusion method has been widely used in target tracking problem [1].

Weighted fusion algorithm is a common method in the field of multi-sensor track fusion. For a moving target, there are often multiple sensors observing its motion state at the same time. To make full use of the information provided by each sensor and estimate the moving state of the target more accurately, a weighted average fusion method is often used, which will get the multi-sensor track measurement information. The weighted average produces the fusion value containing the measurement information of each sensor.

In the distributed fusion algorithm, the fusion centre fuses the local track estimated by the local sensor filter with the global track. In order to obtain an optimal weighted global track, the weights of the sensors must accurately reflect the estimated error of the sensors at the current time. Literature [3] divides the weighting coefficients into predictive fusion weights and estimated fusion weights, weighting twice in the prediction and estimation process not only improves the tracking accuracy but also reduces the amount of calculation. Literature [4] divides the weights of each sensor into two factors: static and dynamic. Static is determined by the performance of the sensor, and dynamic is determined by the mathematical statistics of the fusion state to adjust the weights in real time. Based on the analysis of the defects of self-learning least squares weighted data fusion algorithm, the sensor measurement variance is divided into real-time variance and historic variance to make the weights of the sensor more accurate. According to the idea of weighted fusion algorithm and the track membership matrix of each observation time, the weights of the sensor are dynamically allocated in real time to improve the accuracy of fusion estimation.
In this paper, a new method for calculating sensor weights is presented. TOPSIS algorithm is applied to multi-sensor fusion. Ideal solutions are selected based on the estimation of sensor states and global state prediction residuals. Grey correlation distances and relative proximity between sensor residuals and ideal solutions are calculated. Sensors participating in fusion are filtered based on relative proximity, and sensor weights are assigned. The simulation results show that the algorithm can improve the accuracy of multi-sensor fusion estimation, and can effectively suppress the adverse effects caused by interference.

2. Multi-sensor fusion based on TOPSIS

Taking single target tracking as an example, suppose that there are \(N\) sensors to observe the target simultaneously, and the target's motion state is \(m\)-dimensional. At moment \(k\), sensor \(j\) estimates the state of the target as \(\mathbf{x}_j(k | k) = [x_{1j}, x_{2j}, \ldots, x_{mj}]^T\) \((j = 1, 2, \ldots, N)\). The main steps of multi-sensor fusion based on TOPSIS algorithm are as follows:

Step 1: Compute the residual of the state component between the target global predictions and the local estimates of each sensor to form a residual matrix \(\mathbf{X} = [\mathbf{x}_j(k)]_{m \times N} \).

\[
\mathbf{X} = \{ \mathbf{x}_j(k) \}_{m \times N} \quad (j = 1, 2, \ldots, N)
\]

Step 2: Compute the normalized residuals matrix \(\mathbf{R} = [r_{ij}]_{m \times N}\).

\[
r_{ij} = \frac{\bar{x}_{ij}}{\sum_{j=1}^{N} (\bar{x}_{ij})^2}
\]

Step 3: Determine positive and negative ideal solutions. Note that positive and negative ideal solutions are \(r^+ = (r_{11}^+, r_{21}^+, \ldots, r_{m1}^+)\) and \(r^- = (r_{11}^-, r_{21}^-, \ldots, r_{m1}^-)\), respectively. The size of \(r_{ij}\) represents the normalized difference between each component of the local estimates of each sensor and the global predictions of the target, so it is a cost attribute [7] in the TOPSIS algorithm, so the positive and negative ideal solutions are represented as:

\[
r_{ij}^+ = \min \{r_{ij} | j = 1, 2, \ldots, N\}
\]

\[
r_{ij}^- = \max \{r_{ij} | j = 1, 2, \ldots, N\}
\]

Step 4: Calculate the difference between each sensor and the positive and negative ideal solution. Typically, the TOPSIS algorithm chooses the Euclidean distance to represent the difference between the sensor and the ideal solution.

Step 5: Calculate the relative proximity of each sensor to a positive ideal solution.

\[
C_j = \frac{D_j^+}{D_j^+ + D_j^-}
\]

Obviously, \(C_j \in [0, 1]\). The larger the \(C_j\), the greater the weight the estimated sensor \(j\) takes in participating in the fusion.

Step 6: Calculate target state estimates for sensor fusion

Normalize the relative closeness calculated by Formula (5) as the fusion weight \(\omega_j\) for each sensor:

\[
\omega_j = \frac{C_j}{\sum_{j=1}^{N} C_j}
\]
Based on the weight $\omega_j$ calculated by formula (6), each sensor is weighted and the fused output is obtained:

$$x(k | k) = \sum_{j=1}^{N} \omega_j x_j(k | k)$$  \hspace{1cm} (7)

$$P(k | k) = \sum_{j=1}^{N} \omega_j [P_j(k | k) + (x(k | k) - x_j(k | k))(x(k | k) - x_j(k | k)^T]$$ \hspace{1cm} (8)

3. Improved TOPSIS algorithm

3.1. Improved ideal solution distance based on grey correlation

In this paper, the distance between the sensor and the positive and negative ideal solution is improved. The gray correlation distance between the sensor and the positive and negative ideal solution is used instead of the Euclidean distance in the traditional TOPSIS algorithm.

First, the grey correlation between each sensor and the positive and negative ideal solution is calculated.

$$d(r^x_j, \bar{x}_y) = \frac{\min_{i \neq j} \min_{i \neq j} | r^x_i - \bar{x}_y | + \xi \max_{j \neq i} \max_{j \neq i} | r^x_i - \bar{x}_y |}{\max_{j \neq i} \max_{j \neq i} | r^x_i - \bar{x}_y |}$$ \hspace{1cm} (9)

$$d(r^-_j, \bar{x}_y) = \frac{\min_{i \neq j} \min_{i \neq j} | r^-_i - \bar{x}_y | + \xi \max_{j \neq i} \max_{j \neq i} | r^-_i - \bar{x}_y |}{\max_{j \neq i} \max_{j \neq i} | r^-_i - \bar{x}_y |}$$ \hspace{1cm} (10)

In the formula (9) and (10), $\xi$ is the resolution factor, generally between $[0, 1]$, usually 0.5. The greater the degree of grey correlation $d$, the more accurate the data is, the smaller the contribution to the distance from the ideal solution; conversely, the greater the contribution. Therefore, $D$ and $d$ should be inversely proportional, and the grey correlation distance between each sensor and the ideal solution should be defined as:

$$D^+_j = \sum_{i=1}^{m} 1 / d(r^x_i, \bar{x}_y)$$ \hspace{1cm} (11)

$$D^-_j = \sum_{i=1}^{m} 1 / d(r^-_i, \bar{x}_y)$$ \hspace{1cm} (12)

3.2. High Weight Sensor Filtering

Considering that the measurement accuracy of the sensor may change at different times, if the relative closeness of sensor $j$ is too small at a certain time, it should not participate in the fusion estimation of state.

After the $C_j$ is calculated from Formula (5), the relative proximity of each sensor is rearranged according to its size:

$$C_{(1)} > C_{(2)} > \cdots C_{(N)}$$ \hspace{1cm} (13)

The criterion $\gamma_n$ is defined as:

$$\gamma_n = \sum_{j=1}^{N} C_{(j)} / \sum_{j=1}^{N} C_{(j)}$$ \hspace{1cm} (14)

Set $n_{\min}$ to the maximum $n$ that satisfies gamma $\gamma_n < \rho$, i.e.

$$n_{\min} = \min_{n} \left\{ \arg(\gamma_n \geq \rho) \right\}$$ \hspace{1cm} (15)

Normalize the relative proximity required by Formula (15) as the sensor weight used to fuse the estimates:
\[ \omega_{(j)} = \frac{C_{(j)}}{\sum_{j=1}^{n_{max}} C_{(j)}} \]  

**4. Simulation analysis**

To verify the effectiveness of this algorithm, a three-dimensional space tracking simulation is chosen for the maneuvering target. Set the state vector of target \( t \) at \( k \)-time as \( \mathbf{x}(k) = (x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z})^T \), where \( x, \dot{x}, \ddot{x} \), \( y, \dot{y}, \ddot{y} \), \( z, \dot{z}, \ddot{z} \) are the positions, speeds, and accelerations in the X direction, Y and Z represent the same X, and the transport state equation and observation equation of target are:

\[
\mathbf{x}(k) = F \mathbf{x}(k-1) + Gw(k) 
\]

\[
z(k) = Hx(k) + v(k) 
\]

In the formula, \( w(k) \), \( v(k) \) means process noise and measurement noise with a mean value of 0 and a variance of \( Q(k) \), \( R(k) \); \( F \) is the state transition matrix; \( G \) is the perturbation matrix; \( H \) is the measurement matrix.

Assume that there are five sensors with different measurement accuracy in the airspace, and the ranging error of each sensor on the X, Y, Z equations is \( 15+2j \) m (\( j = 1, 2, 3, 4, 5 \)), and the initial motion state of the target is \([150, 10, 0, 100, 10, 0, 30, 10, 0]^T \), threshold \( \rho = 0.75 \); sampling interval is 1 s; Monte Carlo simulation is 50 times. The movement process of the target is: 1~10 s, the target makes a uniform straight line motion; 11~30 s, the target makes a uniform circular motion with angular velocity\( \omega = 0.1571 \) rad/s; 31~40 s, the target makes a uniform straight line motion; 41~50 s, the target makes a uniform acceleration movement, the acceleration in XYZ direction is \(-3 \text{ m/s}^2, 1 \text{ m/s}^2, -0.5 \text{ m/s}^2\); 51~60 s, the target makes a uniform acceleration movement. The sensor is estimated using an IMMKF filter algorithm, and the target's motion trajectory and the fused estimated trajectory are shown in Figure 1.

![Figure 1. Target trajectory and fusion estimation trajectory](image-url)

To verify the performance of the algorithm, the estimation accuracy of the algorithm is measured by the root mean square error of the location. Define the root mean square error of the location as follows:

\[
\text{RMSE} = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \left( (x_m - \hat{x}_m)^2 + (y_m - \hat{y}_m)^2 + (z_m - \hat{z}_m)^2 \right)}
\]
Figure 2 shows the RMSE diagram of each sensor and fusion estimation. It is obvious from Figure 2 that the tracking error of a maneuvering target is significantly reduced after the fusion of the algorithm in this paper.

![Figure 2. RMSE for each sensor and sensor fusion](image)

To better verify the effectiveness of the algorithm, the effect of sensor interference on target tracking accuracy is considered. Random sensors are interfered at 20 random moments in the target motion time, resulting in serious deviations in the measurement difference of the sensors.

Figure 3 illustrates the RMSE comparison of the sensor observation system after interference. The estimation error of each sensor increases significantly, but the RMSE of the fused estimation does not change significantly, because when the difference between the sensor's estimate and the system's prediction is large, the relative closeness of the sensor becomes smaller, which reduces its fusion weight, and if the relative closeness is too small, it is eliminated. In addition, it does not participate in the fusion estimation and will not have a significant impact on the fusion results.

The effects of threshold $\rho$ and number of sensors $N$ on fusion estimation are analyzed below. Figure 4 is an RMSE graph with different $\rho$ values. As shown in Figure 4, when $\rho=0.75$, the RMSE amplitude estimated by fusion is the smallest, and too large or too small $\rho$ will cause a decrease in tracking accuracy. This is because if the $\rho$ value is too large, a sensor with a large error will be introduced to participate in the fusion; if the $\rho$ value is too small, the number of sensors participating in the fusion will be reduced. Therefore, setting the appropriate $\rho$ value can effectively suppress the interference.

![Figure 4. RMSE of sensor fusion under different $\rho$](image)

![Figure 5. RMSE of sensor fusion under different $N$](image)
Figure 5 shows the effect of the number of sensors on tracking accuracy. As shown in Fig. 5, the number of sensors increases in the presence of interference, which also improves the tracking accuracy of the target and reduces the impact of interference.

5. Conclusion

To solve the problem of multi-sensor weights, a multi-sensor track weighted fusion algorithm based on TOPSIS algorithm is presented. According to the residual between local estimation and global prediction of the sensor at the current time, the weights of the sensor are calculated, and the traditional TOPSIS algorithm is improved by using gray correlation algorithm. A method to filter the high weight sensor is presented, which enhances the anti-interference ability of the algorithm. The simulation results verify that the algorithm can achieve weighted fusion of multiple sensors, reduce the error of target tracking, and provide a new method for weight calculation of multi-sensor fusion.

Acknowledgments

Thanks to my classmates and friends, they gave me a lot of useful materials in the process of completing my thesis, and also provided warm help in the process of paper layout and writing. At the same time, I also want to thank all the scholars cited in this paper for their monographs. Without the inspiration and help of these scholars' research results, I will not be able to complete the final writing of this paper.

References

[1] Xue-Bo J , Shuli S , Hong W , et al. Advances in Multi-Sensor Information Fusion: Theory and Applications 2017[J]. Sensors, 2018, 18(4):1162-.
[2] Wu S, Mclean S. Data Fusion with Correlation Weights[M], Advances in Information Retrieval. 2005. Mettam, G.R., Adams, L.B. (2009) How to prepare an electronic version of your article. In: Jones, B.S., Smith, R.Z. (Eds.), Introduction to the Electronic Age. E-Publishing Inc., New York. pp. 281-304.
[3] SI Yingli, SU Xiaolei, WANG Yidong, et al. An Improved Track Fusion Model with Fused Prediction [J]. Infrared Technology, 2019, 41(05):90-95.
[4] YANG Xiao-Dan, WANG Yun-Feng, ZHANG Xiao-Qin. Dynamic Weighting fusion on Kalman filter[J]. Journal of Sichuan University(Natural Science Edition), 2017, 54(5): 947-952.
[5] SI Ying-li, YANG Xin-yu, CHEN Yong, et al. Multi-sensor Weighted Data Fusion Algorithm Based on Global State Estimation[J]. Infrared Technology, 2014, 36(05):20-24.
[6] RAN Jin-he, ZHANG Yu. Distributed System Data Fusion Based on Track Fuzzy Membership[J]. Signal Processing, 2011, 27(02):68-71.
[7] Ching-Lai Hwang, Abu Syed Md. Masud. Multiple Objective Decision Making—Methods and Applications: A State-of-the-Art Survey[M]. 1979.
[8] Julier S J, Uhlmann J K. A Non-Divergent Estimation Algorithm in the Presence of Unknown Correlations[C]// Proceedings of the 1997 American Control Conference. Albuquerque, NM: IEEE, 1997: 2369-2373.
[9] Li T , Zhao Z , Sun C , et al. Adaptive Channel Weighted CNN with Multi-Sensor Fusion for Condition Monitoring of Helicopter Transmission System[J]. IEEE Sensors Journal, 2020, PP(99):1-1.
[10] Yang C , Yang Z , Deng Z . Robust weighted state fusion Kalman estimators for networked systems with mixed uncertainties[J]. Information Fusion, 2019, 45:246-265.
[11] Eui-Hyuk Lee, Taek Lyul Song. Multi-sensor track-to-track fusion with target existence in cluttered environments[J]. Iet Radar Sonar Navigation, 2017, 11(7):1108-1115.
[12] Jiff Ajgl, Ondrej Straka. Covariance intersection in track-to-track fusion with memory[C]/ IEEE International Conference on Multi-sensor Fusion & Integration for Intelligent Systems. IEEE, 2016.
[13] Deng Z., Wang J. Multi-sensor Data Fusion Based on Improved Analytic Hierarchy Process[J]. IEEE Access, 2020, PP(99):1-1.