

A Method Based on the Kalman Filter to Solve the Multi-sensor Data Loss Problem

Yiang Lei

*School of Control Engineering, Northeastern University, Shenyang, China
1759328280@qq.com*

Abstract: Multi-sensor systems are widely deployed in autonomous driving, industrial control, and environmental monitoring, yet data loss due to sensor failures remains a critical challenge. To address this issue, this paper proposes a novel Kalman filter-based framework for multi-sensor data loss recovery. In this framework, the system model is dynamically integrated with the observed data from related sensors. The missing information is then reconstructed through the recursive calculation and data fusion capabilities of the extended Kalman filter. Specifically, the paper derive a minimum variance coefficient to optimize fusion synchronous sensor measurements such as radar position and velocity data through coordinate transformations and the inner relation of the measured object. The paper apply the theoretical analysis to derive the coefficients that minimize the estimated variance under known conditions, which show that this method can effectively reduce the estimation error caused by data loss while maintaining the computational efficiency. Compared to traditional interpolation or neural network-based approaches, the framework provides a lightweight and precise solution for real-time applications in resource-constrained environments.

Keywords: Kalman filter, Multi-sensor systems, Data loss recovery, minimum-variance estimation, sensor fusion

1. Introduction

The Kalman filter is an widely used algorithm in some sensor data fusion tasks, which use a series of data from the sensors and the previous status to estimate the future status [1, 2]. With the application in the diverse realm, the Kalman filter has been perfected and updated since it birth. And the various updated algorithm can be Applicable to various scenarios and situations [3, 4].

The multi-sensor systems are ubiquitous, and obviously, the data loss is normal because of the breakdown in both the hardware and software[3-5]. The antecedent study always focus on the methods of interpolation and some statistical approaches [6]. However, traditional interpolation methods often produce inaccurate results, while neural network (NN)-based approaches tend to be computationally intensive. In recent years, with the advancement of neural networks, some studies have focused on developing algorithms that leverage NNs for data completion in multi-sensor systems [7]. Nevertheless, these algorithms remain complex and resource-demanding, as the implementation of neural networks typically relies on substantial computational power. The Kalman filter is benefit by its efficiency through its recursive computation and its accuracy through the data fusion between the system model and observation data [8]. This paper will focus on some algorithms based on the Kalman filter in the multi-sensor system data loss situations, which try to make it possess

the advantages of the Kalman filter. Specifically, the synchronized sensors always share the data in the same format. If one of them lose the data, the others and the estimate results can jointly complement the data by some specific transform. Giving extended application, some properties of the observed object have specific relations, which can complement the datas mutually. This algorithm can help the system have more data source from the different sensors to get the states of the observed objects, especially when the some sensors lose data, which Increased robustness of the system.

2. The data complementation by the sensors which share the highly correlated and synchronized sensor data

In certain scenarios, homologous sensors which array in some specific pattern are always used to get some data can't given by solo sensor. These sensors share the highly correlated and synchronized sensor data in the same format. If one of them breakdown, the others can send and share the data to complement and get the better results. By integrating this approach with traditional methods such as the Kalman filter, even better outcomes can be obtained through data fusion.

2.1. The overview of the Kalman filter

The Kalman filter is widely employed for object prediction in various applications. As a recursive estimator, it operates by predicting the current state of a system and then updating this prediction based on new measurements:

$$\hat{x}_k^- = F_k \hat{x}_{k-1} + B_k \vec{u}_k \quad (1)$$

And the measurement data can be got from the sensor and deal with it by:

$$\hat{x}_{km} = Z_k H^{-1} \quad (2)$$

Then final result can represented as:

$$\hat{x}_k = \hat{x}_k^- + K_k (Z_k - H_k \hat{x}_k^-) \quad (3)$$

Where, the coefficient K_k can ensured by:

$$K_k = \frac{P_k^- H^T}{H P_k^- H^T + R} \quad (4)$$

Through the mathematical proof, the coefficient K_k can provides minimum-variance state estimates by optimally combining system models and observation data.

2.2. The data processing

If the homologous sensors were used in a system, the sensors share the same state matrix. And the sensor whose data lost in the given time can be complemented from the other sensors' data by the linear transform.

2.2.1. The radar example

The system with the radar located in the different position can serve as a typical case. The radars can measure the location and the velocity in their reference system, which get the different data. If one of them lost the data in a certain time, its data can be complemented by the others through the geometric relation.[9]

To transform the measurement results from two different radars into each other's coordinate systems, it is necessary to determine the transformation relationship between the coordinate systems and apply translation and rotation operations.

Specifically, the first step is to clearly define the origin and orientation of each radar's coordinate system need to be clearly defined, and the translation vector and rotation matrix must be determined. The translation step involves moving the origin of one radar's coordinate system to the other radar's coordinate system. The rotation step then aligns the coordinate axes using the rotation matrix to ensure that the measurement results are accurately transformed and consistent between the two radar systems.

Mathematically, this can be represented using a translation vector $T = [t_x \ t_y \ t_z]^T$ and a rotation matrix R . The transformation formula is:

$$P_k^- = RP_A + T \quad (5)$$

Where,

P_A is the location of the measured points in Radar A's coordinate system

P_k^- is the estimated position of the transformed points in Radar B's coordinate system.

In practical applications, it is important to ensure that the data from both radars are time-synchronized and to consider measurement and transformation errors. So it always represented as:

$$P_k = RP_A + T + w_p \quad (6)$$

Where,

w_p is the errors produced in the measurement and transformation process.

P_k is the exact position of the transformed points in Radar B's coordinate system.

Measured velocities can also be transformed through coordinate conversion. Similar to position conversion, velocity conversion also requires considering the rotation of the coordinate system, but translation does not affect velocity because velocity is a vector and depends only on direction.

The basic principle of velocity conversion is to use the same rotation matrix as in position conversion. The formula for velocity conversion is:

$$v_k^- = Rv_A \quad (7)$$

$$v_k = Rv_A + w_v \quad (8)$$

Where

v_A is the velocity vector in Radar A's coordinate system,

v_k^- is the transformed velocity vector estimation in Radar B's coordinate system.

v_k is the exact transformed velocity vector in Radar B's coordinate system.

Evidently, the transformations for position and velocity can be integrated into a unified mathematical framework, typically using a state vector and a transformation matrix. In the domains of target tracking and sensor fusion, position and velocity are often combined into a state vector, which is then processed using a unified transformation matrix.

The States Vector can be defined as:

$$x = \begin{bmatrix} p \\ v \end{bmatrix} \quad (9)$$

Where,

x can include both position vector p and velocity vector v ,

p is the position vector,

v is the velocity vector.

The transformation matrix T_R can handle both position and velocity transformations.

For coordinate system rotation and translation, the transformation matrix takes the form:

$$T_R = \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} \quad (10)$$

Where,

R is the rotation matrix,

0 is a zero matrix.

Then, the transformation of position and velocity can be integrated as:

$$x_k^- = T_R x_A + T_t \quad (11)$$

Where:

x_A is the state vector in Radar A's coordinate system.

x_k^- is the transformed state vector in Radar B's coordinate system.

T_t is the correction vector and can be represented as

$$T_t = \begin{bmatrix} T \\ 0 \end{bmatrix} \quad (12)$$

Where,

T is the translation vector, only applied to the position component.

0 is a zero vector, indicating that velocity is unaffected by translation.

Because it is necessary to ensure that the data from both radars are time-synchronized and to consider measurement and transformation errors In practical applications, the final result can be represented as:

$$x_k = T_R x_A + T_t + w \quad (13)$$

Where,

w is the errors produced in the measurement and transformation process.

2.2.2. The data fusion between the estimated data and complemented data

The estimated states always represented as:

$$x_k = F_k \hat{x}_{k-1} + B_k \vec{u}_k + w \quad (14)$$

However, the noise always can't be modeled, so previous studys always estimate the final results by the formula (1) to get an approximate result. To get the better result, previous studys always use the formula (3) to fuse the sensor measurement data and estimated data and the coefficient can be obtained by the formula (4). However, according to the scenario presented in today's paper, the corresponding sensor malfunctioned and was unable to provide data. Consequently, the paper aims to achieve more optimal results by effectively fusing the measurement data from the remaining sensors. Therefore, an operation similar to formula (3) is performed on equation (10):

$$\hat{x}_k = \hat{x}_k^- + K_c (T_R x_A + T_t - \hat{x}_k^-) \quad (15)$$

Where,

2.2.3. The coefficient K_k which provides minimum-variance state estimates

The coefficient K_k can ensured by the principle that the the variance of the final states matrix $\hat{x}_k$ need to be the least.

Firstly, the estimate error can be represented as:

$$e = x_k - \hat{x}_k \quad (16)$$

Then, substitute the $\hat{x}_k^-$ with the formula (14):

$$e = x_k - \hat{x}_k^- - K_c(T_R x_A + T_t) + k\hat{x}_k^- \quad (17)$$

Because of (13):

$$e = x_k - \hat{x}_k^- - K_c(x_k - v) + k\hat{x}_k^- \quad (18)$$

$$e = x_k - \hat{x}_k^- - K_c x_k + k\hat{x}_k^- + kv \quad (19)$$

Then,

$$e = (I - K_c)(x_k - \hat{x}_k^-) + kv \quad (20)$$

Where,

I is the unit matrix.

Because the covariance matrix of the final estimate matrix can be represented as:

$$P = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \quad (21)$$

Then the covariance matrix of the final estimate matrix can be represented as:

$$P = E[((I - K_c)e^- + kv)((I - K_c)e^- + kv)^T] \quad (22)$$

By the principle of the matrix multiplication:

$$P = E[((I - K_c)e^- + kv)(e^{-T}(I - K_c)^T + v^T k^T)] \quad (23)$$

$$P = E[(I - K_c)e^- e^{-T}(I - K_c)^T] + E[(I - K_c)e^- v^T k^T] + E[kve^{-T}(I - K_c)^T] + E[kvv^T k^T] \quad (24)$$

Because the two principle:

$$E[AB] = E[A]E[B] \quad (25)$$

$$E[cA] = cE[A] \quad (26)$$

Where,

A and B are two random variable which are independent,

c is the constant.

Then,

$$E[(I - K_c)e^- v^T k^T] = (I - K_c)k^T E[e^- v^T] \quad (27)$$

$$E[(I - K_c)e^- v^T k^T] = (I - K_c)k^T E[e^-] E[v^T] \quad (28)$$

In this condition the paper suppose that the noise obey normal distribution. Therefore, the paper definite that:

$$P(w) \sim N(0, Q) \quad (29)$$

$$P(v) \sim N(0, R) \quad (30)$$

Therefore,

$$E[e^-] = 0 \quad (31)$$

$$E[v^T] = 0 \quad (32)$$

Therefore,

$$E[(I - K_c)e^- v^T k^T] = 0 \quad (33)$$

Then,

$$P = E[(I - K_c)e^- e^{-T}(I - K_c)^T] + E[kvv^T k^T] \quad (34)$$

$$P = (I - K_c)E[e^-e^{-T}](I - K_c)^T + kE[vv^T]k^T \quad (35)$$

Because of the principle of (29) and the formula(34),

$$P = (I - K_c)P(I - K_c)^T + K_cRK_c^T \quad (36)$$

$$P = P^- - K_cP^- - P^-K_c^T + K_cP^-K_c^T + K_cRK_c^T \quad (37)$$

Variance can be expressed as the trace of the covariance matrix, so the variance of the final estimate states can be:

$$\text{tr}(P) = \text{tr}(P^-) - 2\text{tr}(K_cP^-) + \text{tr}(K_cP^-K_c^T) + \text{tr}(K_cRK_c^T) \quad (38)$$

To get the least variance, take the derivative of formula (37):

$$\frac{d[\text{tr}(P)]}{dK_c} = \frac{d[\text{tr}(P^-)]}{dK_c} - 2\frac{d[\text{tr}(K_cP^-)]}{dK_c} + \frac{d[\text{tr}(K_cP^-K_c^T)]}{dK_c} + \frac{d[\text{tr}(K_cRK_c^T)]}{dK_c} \quad (39)$$

Given this two principle about the Matrix derivative rule [10]:

$$\frac{d[\text{tr}(AB)]}{dA} = B^T \quad (40)$$

$$\frac{d[\text{tr}(ABA^T)]}{dA} = 2AB \quad (41)$$

Then,

$$\frac{d[\text{tr}(P)]}{dK_c} = -2P^{-T} + 2K_cP + 2K_cR \quad (42)$$

Let,

$$\frac{d[\text{tr}(P)]}{dK_c} = 0 \quad (43)$$

Then the final coefficient which makes the least variance of the estimate state can be got:

$$K_c = \frac{P}{P+R} \quad (44)$$

3. The data complementation by the inner relation of the measured object

If certain characteristics characters of the measured object have some specific linear relation, these relationships can be used to make up the data loss. If the states whose corresponding sensor's data has lost, and the matrix can be represented by the other states:

$$x_t = A_ty_t + C_t + u_t \quad (45)$$

Where,

x_t is the states measured by the sensor whose data lost

y_t is the states which has some specific linear relation and can be measured by the other sensors

A_t and C_t is the coefficient of the linear relation between the states matrix x_t and y_t

u_t is the noise when the x_t represented by the y_t

Because the sensor's measurement data matrix can be represented by:

$$Z_y = H_yy_t + v_t \quad (46)$$

y_t is the states which has some specific linear relation and can be measured by the other sensors,

Z_y is the sensor's measurement data matrix,

H_y is the measurement matrix which describe the linear relation between the y_t and Z_y ,

v_t is the noise produced in the measurement process.

Hence, the states matrix y_t can be represented by the sensor's measurement matrix:

$$y_t = Z_y H_y^{-1} + v_t \quad (47)$$

Where,

v_t is the noise in the inverse transform process.

Then the result through the indirect measurement can be got by the formula (44) and (46):

$$x_t = A_t(Z_y H_y^{-1} + v_t) + C_t + u_t \quad (48)$$

However, the noise v_t and u_t always can't be modeled, so the following formula always be used to get the approximate results:

$$\hat{x}_{tm} = TZ_y H_y^{-1} + C_t \quad (49)$$

The states can be predicted by the last states matrix by the formula (3). Then, The final predictions can be obtained through this formula:

$$\hat{x}_t = \hat{x}_t^- + K_t(TZ_t H_t^{-1} + C_t - \hat{x}_k^-) \quad (50)$$

However, in some condition, the measurement matrix may be non-invertible, hence it can be transformed as:

$$\hat{x}_t = \hat{x}_t^- + K_t(TZ_y + H_t C_t - H_t \hat{x}_k^-) \quad (51)$$

The coefficient K_t can ensured by the principle that the variance of the final states matrix $\hat{x}_k'$ need to be the least. Because of the similar format with the formula (14), the formula (50) can be done the same operation like the formula (15)~(43), and the final coefficient K can be represented as:

$$K_t = \frac{PH^T}{HPH^T + R} \quad (52)$$

4. The data complementation by the sensors which share part of the states with the loss data

In most of the multi-sensors systems, the different type sensors always used to finish the given task. If one of them loss some data at a moment, the other sensors' data can be used to complement it. Because the other sensors only share part of the states with the temporarily faulty sensor, the final states matrix measured by the temporarily faulty sensor can be partly represented as:

$$\hat{x}_k = \hat{x}_k^- + K_{cA}(T_{RA}x_A + T_{tA} - \hat{x}_k^-) + K_{cB}(T_{RB}x_B + T_{tB} - \hat{x}_k^-) + \dots \quad (53)$$

Where,

T_{RA} and T_{RB} can be represented as $\begin{bmatrix} 0 & 0 & 0 \\ 0 & T_r & 0 \\ 0 & 0 & 0 \end{bmatrix}$ (some 0 are not necessary), and T_{tA} and T_{tB} can

be represented as $\begin{bmatrix} 0 \\ T_t \\ 0 \end{bmatrix}$ (some 0 are not necessary), which help them get the different state from the different sensor instead of the sensor with all the lost data.

5. Conclusion

This paper expounds a comprehensive and methodical approach to tackling the issue of multi-sensor data loss. It capitalizes on the remarkable capabilities of the Kalman filter in recursive state estimation and optimal data fusion. By skillfully exploiting the correlations between synchronized

sensors, such as those in radar systems a unified mathematical framework is established for transforming and fusing positional and velocity data across coordinate systems, even when partial measurements are missing. The derived minimum-variance coefficient ensures robust recovery of lost data while minimizing estimation errors. Key advantages of the proposed method include computational efficiency, which avoids reliance on resource-intensive techniques like neural networks, making it suitable for real-time embedded systems; accuracy, which integrates system dynamics and sensor observations through rigorous error minimization and generality, which is applicable to diverse multi-sensor configurations with linear state relationships.

This study primarily focuses on the theoretical framework of Kalman filtering for data complementation, with two notable limitations: the absence of a rigorous derivation for optimal covariance matrix updates, and the lack of experimental validation. Future work could extend this framework to nonlinear systems, such as by utilizing extended Kalman filters as suggested in reference and to validate its performance in scenarios with asynchronous or heterogeneous sensor networks. Additionally, the estimate of the covariance matrix, which is also an important work in the Kalman filter, wasn't discussed, so further work can be done in this part.

References

- [1] Kalman, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problems". *Journal of Basic Engineering*. 82 (1): 35–45.
- [2] "IEE Colloquium on 'Kalman Filters: Introduction, Applications and Future Developments' (Digest No.27)," *IEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments*, London, UK, 1989, pp. 0_1-.
- [3] Kibrete, Fasikaw, Dereje Engida Woldemichael, and Hailu Shimels Gebremedhen. "Multi-Sensor data fusion in intelligent fault diagnosis of rotating machines: A comprehensive review." *Measurement* (2024): 114658.
- [4] Xu, Xiaobin, et al. "A review of multi-sensor fusion slam systems based on 3D LIDAR." *Remote Sensing* 14.12 (2022): 2835.
- [5] L. Kong, M. Xia, X. -Y. Liu, M. -Y. Wu and X. Liu, "Data loss and reconstruction in sensor networks," 2013 *Proceedings IEEE INFOCOM*, Turin, Italy, 2013, pp. 1654-1662, doi: 10.1109/INFOCOM.2013.6566962.
- [6] Durrant-Whyte, Hugh F. *Integration, coordination and control of multi-sensor robot systems*. Vol. 36. Springer Science & Business Media, 2012.
- [7] Fan, Gao, Jun Li, and Hong Hao. "Lost data recovery for structural health monitoring based on convolutional neural networks." *Structural Control and Health Monitoring* 26.10 (2019): e2433.
- [8] Morrison, G. Wayne, and David H. Pike. "Kalman filtering applied to statistical forecasting." *Management Science* 23.7 (1977): 768-774.
- [9] Barton, David K. *Radar system analysis and modeling*. Artech House, 2004.
- [10] Little, Roderick JA, and Donald B. Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2019.