

Article

# Adaptive Unscented Kalman Filter for Target Tacking with Time-Varying Noise Covariance Based on Multi-Sensor Information Fusion

Dapeng Wang <sup>1</sup> , Hai Zhang <sup>1,2,\*</sup> and Baoshuang Ge <sup>3</sup> 

<sup>1</sup> School of Automation Science and Electrical Engineering, Beihang University, No. 37 Xueyuan Road, Haidian Distirct, Beijing 100083, China; wangdapeng@buaa.edu.cn

<sup>2</sup> Science and Technology on Aircraft Control Laboratory, Beihang University, No. 37 Xueyuan Road, Haidian Distirct, Beijing 100083, China

<sup>3</sup> Yancheng State-Owned Assets Investment Group Co., Ltd., No. 669 Century Avenue, Yandu District, Yancheng 224000, China; gebaoshuang@buaa.edu.cn

\* Correspondence: zhanghai@buaa.edu.cn; Tel.: +86-10-8233-9366

**Abstract:** In this paper, an innovative optimal information fusion methodology based on adaptive and robust unscented Kalman filter (UKF) for multi-sensor nonlinear stochastic systems is proposed. Based on the linear minimum variance criterion, this multi-sensor information fusion method has a two-layer architecture: at the first layer, a new adaptive UKF scheme for the time-varying noise covariance is developed and serves as a local filter to improve the adaptability together with the estimated measurement noise covariance by applying the redundant measurement noise covariance estimation, which is isolated from the state estimation; the second layer is the fusion structure to calculate the optimal matrix weights and gives the final optimal state estimations. Based on the hypothesis testing theory with the Mahalanobis distance, the new adaptive UKF scheme utilizes both the innovation and the residual sequences to adapt the process noise covariance timely. The results of the target tracking simulations indicate that the proposed method is effective under the condition of time-varying process-error and measurement noise covariance.

**Keywords:** multi-sensor information fusion; process-error estimation; adaptive and robust unscented Kalman filter; target tracking



**Citation:** Wang, D.; Zhang, H.; Ge, B. Adaptive Unscented Kalman Filter for Target Tacking with Time-Varying Noise Covariance Based on Multi-Sensor Information Fusion. *Sensors* **2021**, *21*, 5808. <https://doi.org/10.3390/s21175808>

Academic Editors: Andrzej Staczny and Paolo Mercorelli

Received: 29 June 2021

Accepted: 24 August 2021

Published: 29 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

There have been increasing demands for developing robust, adaptive, and accurate multi-sensor information filters (MSIF), which have been widely applied to many fields such as navigation systems, modern industries, military threat detection, target tracking, and remote sensing [1,2]. Especially in recent years, there have been many state estimation problems in which the processes were often non-linear and uncertain for tracking and navigation, for example [3]. Hence, the theory has been researched and broadly applied to many realistic systems. A method with both adaptability and robustness cannot be realized in real time for a single sensor/observer system [4]. By using a multi-sensor structure, an information fusion algorithm can obtain much more accurate estimations than a single one [5,6]. However, to the best of our knowledge, few methods focus on both adaptability and robustness in MSIF. From a system point of view, there are mainly two different methods to process the data from a multi-sensor [7]. The first method is the centralized filter where all raw sensor data are fed to a central site for processing [1]. The second one is the decentralized or distributed filter where the process is divided between some local filter concurrently to obtain individual raw data-based estimates and one master/center filter to fuse those local estimates to provide a much accurate global optimal estimate [6,8].

The centralized filter is also called measurement fusion, because all observations are directly fused to obtain a final estimate. The main advantage is high accuracy due to the

use of lossless information. However, in practice, this architecture is heavily challenged by a large complexity of computation while the number of sensors increases. Another drawback of this method is the lack of robustness when one or more sensors fail.

In the decentralized structure, the complexity of computation can be reduced in the center filter because part of the computation is taking place at the local filters; furthermore, fault detection and isolation are easier to be implemented. For the reasons given above, parallel structure, which can provide improved reliability and fault tolerance of sensors, has been paid more attention to and implemented in many aspects [1,8].

Frank et al. proposed estimators for multi-sensors with different failure rates [9]. They took the stochastic perturbations and the probability of sensors' failure into account, so the robustness could be improved. However, the matrices which signify the direction of perturbations are assumed known, which may not be available in practice for many systems. An excellent piece of literature also focused on the problem, where the Bernoulli random variables were used to predict the phenomenon of missing observations [10]. Unfortunately, the assumption that any failed sensor may recover after specific  $m$  instances of time cannot be held in many harsh environments.

Qiu et al. proposed a diagonal weighting matrix method for the fusion of local estimates [7]. However, this algorithm gains computational efficiency at the expense of a loss in accuracy [11]. Zhang et al. put forward a method to fuse the multi-sensor measurements in a sequential way [12]. In [13], under the energy harvesting constraints, a robust fusion filtering over a multi-sensor system is proposed. By using the covariance intersection fusion strategy, this theoretical framework for discrete time-varying stochastic multi-sensor systems is established. Based on the fact that so many scholars successfully use the decentralized structure to improve the performance of multi-sensor information fusion, this paper adopts the decentralized structure [8].

For a local filter in the MSIF, there are several options. Kalman filter is a promising method for linear problems [14]. When problems become nonlinear, a set of improved forms could be adopted, of which the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) are used widely. Using Taylor series expansions, the EKF linearizes the non-linear models to make them convenient for the standard Kalman filter procedure. The core drawback of the EKF makes it unable to achieve sufficient estimation accuracy for a maneuvering target, because the first-order approximate error could be huge under strong nonlinear conditions. Although a lot of efforts were made to estimate the states by the adaptive algorithms based on the covariance estimation, as argued by Ge et al. [15], the EKF and its adaptive forms are still not optimal options for the key reason above. The generic particle filter (PF) and the cubature Kalman filter (CKF) are also well-known methods for nonlinear problems, as mentioned by [16]; the high computation cost might not be affordable in many applications.

The posterior mean and covariance of any Gaussian random variable in third-order accuracy could be approximated by UKF based on the unscented transformation (UT) [17]. To apply the UKF, process-error and measurement noise should be taken into consideration, so plenty of adaptive UKF methods have been proposed. Soken et al. corrected the mismatches of the process noise covariance ( $\mathbf{Q}$ -adaptation) based on an adaptive UKF algorithm, and they applied the method to accomplish the picosatellite attitude estimation under the condition that the process noise covariance may vary [18]. However, by using one scalar parameter to correct the  $\mathbf{Q}$ , the accuracy is limited to a certain extent. Similar to [18], Chang proposed a method with both adaptivity and robustness [4]. The method can deal with the condition that both the process and measurement noise covariance change at the same time, but not in real time. Meanwhile, the drawback of the heuristic method is the same as [4], that only one scalar was calculated for adapting the  $\mathbf{Q}$  or  $\mathbf{R}$ , which stand for the measurement noise covariance.

As argued in [10], if measurements only contain noise, they can be seen as outliers [6]. To make a more accurate estimation of the noise covariance, plenty of scholars have turned to adaptive UKF methods. Mohamed et al. developed an adaptive Kalman filter, based on

the maximum likelihood criterion for the proper choice of filtering weight. They argued that the method is efficient by adapting the matrices  $\mathbf{R}$  and/or  $\mathbf{Q}$  [19]. The basic idea of [19] is to adapt the  $\mathbf{R}$  by innovation sequences and the  $\mathbf{Q}$  by residual ones. However, the innovation and residual sequences obtained from the filter are not independent [15].

In [20], based on an adaptively robust EKF, Yuan et al. proposed a PDR/UWB (Pedestrian Dead Reckoning and Ultra-Wide Band) integrated navigation algorithm. To obtain the adaptability, the algorithm takes the positioning scene and the heading as constraints. The robustness of the algorithm is achieved by adopting the idea similar to [6]. However, in many other applications such as tracking and remote sensing, the constraints are not always available or difficult to implement because of the great complexity and the high computational cost [21]. In [22], to handle measurement outliers, the robust estimation reduces the weight of the observation; to handle the model error, an adaptive factor is introduced to balance the adverse effect. This algorithm has inspired many scholars. In [23], two novel quantitative nonlinear observability measures are proposed to get an optimal filter design. However, both [22] and [23] inevitably use the state-dependent calculations to adjust the measured values or the weight matrix, so the same problem as in [20] mentioned above exists. In [24], an adaptive filtering method was proposed. The authors used the residual error to construct a low-pass filter together with the process noise covariance. They argued that the high process noise could be effectively suppressed. However, if measurement noise covariance is not estimated accurately in a timely manner, the residual error will be inaccurate; then, in the next iteration, the residual error may not be adjusted only by the algorithm. So, it is necessary to estimate the measurement noise covariance as accurately as possible. Moreover, it is better that the estimation of the measurement noise covariance is independent of the process noise or estimation of the state [15].

If the estimation of the noise covariance matrix  $\mathbf{R}$  could be accurately made, it would be a solid foundation to give the matrix  $\mathbf{Q}$  a relatively accurate estimation. Zhang et al. developed a measurement-based adaptive Kalman filtering algorithm (MAKF) that overcame the instability drawback of improved Sage–Husa adaptive filter for the integrated navigation system [25,26]. Realizing the limitation of MAKF is that the following assumption could not always be held—one of the measurement noise covariances is relatively smaller than the other—the group of Zhang afterwards developed an improved method named redundant measurement noise covariance estimation (RMNCE) [27], which can estimate the noise variance of the measurement and is not affected by the process state estimation error. So, in this work, we utilize the RMNCE, which can deal with the unknown noise covariance in real time, to calculate the measurement noise covariance  $\hat{\mathbf{R}}_k$  for each sensor.

In our proposed method, the matrix  $\hat{\mathbf{R}}_k$  of each sensor is also calculated through innovation sequences, denoted as  $\mathbf{R}_{in}$ , as mentioned in [28]. We denote the ratio between  $\mathbf{R}_{in}$  and  $\hat{\mathbf{R}}_k$  as the indicator to reflect whether there would be non-ignorable process error or not. Furthermore, if the indicator or the hypothesis test theory based on chi-square suggests the existence of process error, the trigger for adaptation is on. To the best of our knowledge, the combination of the two criteria is firstly introduced by this paper.

For a given MSIF problem, without loss of generality, the statistical properties of measurement noise are not reliable, though they could be obtained in advance. So, we adopt the RMNCE method to estimate the variance of measurement noise in multi-sensor system. Additionally, taking all the above adaptive  $\mathbf{Q}$  estimation algorithms into consideration, a new  $\mathbf{Q}$  estimation algorithm based on both innovation and residual sequences is given, which is inspired by [16]. Finally, the decentralized architecture is used to fuse the estimations from local sensors.

The contributions of the paper are twofold.

Firstly, an efficient algorithm is proposed for the MSIF to detect the process-error based on the indicator, which is combined by hypothesis test theory with the Mahalanobis distance of innovation sequences and the RMNCE. Then, an innovative  $\mathbf{Q}$  estimation algorithm is proposed by using both innovation and residual sequences.

Secondly, to the best of our knowledge, the RMNCE algorithm together with a weighted factor is first introduced into MSIF. To begin with, the covariance of measurement noise obtained by RMNCE is not only used as the measurement noise covariance estimation of each sensor but also as the element to calculate the weighted factor. Then, a novel method is proposed to simplify the calculation of the weighted factor as an alternative to optimal matrix weights in [1]. Moreover, an indicator is also proposed based on the RMNCE to detect whether the process-error exists or not. At last, the simulation results demonstrate that the proposed scheme can increase the tracking precision with both adaptivity and robustness.

The remainder of this paper is organized as follows: in the next section, we describe the standard UKF, the RMNCE, an innovative adaptive UKF (AUKF) proposed by this paper and the decentralized MSIF. Section 3 introduces the adaptive multi-sensor information fusion algorithm (RAUKF-MSIF). Section 4 provides the simulation and discussions. Section 5 finally draws the conclusions.

## 2. The Decentralized MSIF and the Proposed Adaptive UKF

Considering a discrete time nonlinear stochastic system with  $l$  sensors, the process and measurement models can be described as

$$\begin{cases} \mathbf{X}_k = f(\mathbf{X}_{k-1}) + \Gamma_{k-1} \mathbf{W}_{k-1} \\ \mathbf{Z}_k^i = h^i(\mathbf{X}_k) + \mathbf{V}_k^i, i = 1, 2, \dots, l \end{cases} \quad (1)$$

where  $\mathbf{X}_k \in \mathbb{R}^{n \times 1}$  denotes the state vector,  $\mathbf{Z}_k^i \in \mathbb{R}^{m_i \times 1}$  is the measurement collected by sensor  $i$  at sampling time instant  $k$ ,  $\mathbf{W}_{k-1}$  and  $\mathbf{V}_k^i$  are uncorrelated zero-mean Gaussian white noise with compatible dimension [4,16,25,26,28,29],  $f(\bullet)$  and  $h(\bullet)$  are the known time-varying nonlinear state transition and measurement function, respectively.  $\Gamma_{k-1}$  is the system noise-driven matrix with compatible dimension.

The statistical properties assumed about noise processes can be summarized as [4]

$$\begin{cases} E[\mathbf{W}_k \mathbf{W}_j^T] = \mathbf{Q}_k \delta_{k,j} \\ E[\mathbf{V}_k^i \mathbf{V}_j^{iT}] = \mathbf{R}_k^i \delta_{k,j} \\ E[\mathbf{W}_k \mathbf{V}_j^{iT}] = 0 \end{cases} \quad (2)$$

where  $\delta_{k,j}$  denotes the Kronecker delta function.

The following assumptions are also made as initial value

$$E(X_0) = \mu_0, E[(X_0 - \mu_0)(X_0 - \mu_0)] = \mathbf{P}_0 \quad (3)$$

where the initial state  $X_0$  is independent of  $\mathbf{W}_{k-1}$  and  $\mathbf{V}_k^i$ ,  $\mathbf{P}_0$  is the initial estimation error covariance matrix.

In the following Section 2.1, the procedure of the standard UKF is briefly reviewed. In Section 2.2, the RMNCE is introduced. To improve the adaptivity of the estimation, the process-error should be detected timely, so an innovative adaptive UKF is proposed in Section 2.3. In Section 2.4, the structure of MSIF adopted by our work is given.

### 2.1. The Standard UKF

The UKF uses the fact that it should be easier to estimate a nonlinear distribution than to give an approximation of a nonlinear system [29]. In the standard UKF, to generate the sigma points to undergo the nonlinear transformation and calculate the first two moments of the transformed set, the UT, a deterministic sampling technique, is implemented. For the sake of simplicity, only one sensor of (1) is taken into consideration, and the general procedures are as follows:

Step 1: Initialization.

$$\begin{cases} \hat{\mathbf{X}}_0 = E[\mathbf{X}_0] \\ \mathbf{P}_0 = E[(\mathbf{X}_0 - \hat{\mathbf{X}}_0)(\mathbf{X}_0 - \hat{\mathbf{X}}_0)^T] \end{cases} \quad (4)$$

where  $\hat{\mathbf{X}}_0$  is the initial state,  $\mathbf{P}_0$  is the initial estimation error covariance.

Step 2: Sigma points generation.

$$\begin{cases} \chi_{k-1}^{(0)} = \hat{\mathbf{X}}_{k-1} \\ \chi_{k-1}^{(i)} = \hat{\mathbf{X}}_{k-1} + \sqrt{(n+\lambda)P_{k-1}}, i = 1, 2, \dots, n \\ \chi_{k-1}^{(i)} = \hat{\mathbf{X}}_{k-1} - \sqrt{(n+\lambda)P_{k-1}}, i = n+1, n+2, \dots, 2n \end{cases} \quad (5)$$

where  $n$  denotes the dimension of the state;  $\lambda = \alpha^2(n + \kappa) - n$  is the composite scaling parameter that is used for fine tuning,  $\alpha$  is set to  $0 \leq \alpha \leq 1$  and a good default setting on  $\kappa$  is  $\kappa = 0$  [15].

Step 3: State prediction.

$$\begin{cases} \chi_{k/k-1}^{(i)} = f(\chi_{k-1}^{(i)}, k-1), i = 0, 1, 2, \dots, 2n \\ \hat{\mathbf{X}}_{k/k-1} = \sum_{i=0}^{2n} \omega_i^{(m)} \chi_{k/k-1}^{(i)} \\ \mathbf{P}_{\mathbf{X}\mathbf{X}} = \sum_{i=0}^{2n} \omega_i^{(c)} (\chi_{k/k-1}^{(i)} - \hat{\mathbf{X}}_{k/k-1})(\chi_{k/k-1}^{(i)} - \hat{\mathbf{X}}_{k/k-1})^T \\ \mathbf{P}_{k/k-1} = \mathbf{P}_{\mathbf{X}\mathbf{X}} + \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^T \end{cases} \quad (6)$$

where  $\hat{\mathbf{X}}_{k/k-1}$  is the predicted state mean; and  $\mathbf{P}_{\mathbf{X}\mathbf{X}}$  is the predicted state covariance.  $\omega_i^{(m)}$  and  $\omega_i^{(c)}$  are weights, which are defined as

$$\begin{aligned} \omega_0^{(m)} &= \frac{\lambda}{n+\lambda} \\ \omega_0^{(c)} &= \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \\ \omega_i^{(m)} &= \omega_i^{(c)} = \frac{\lambda}{2(n+\lambda)}, i = 1, 2, \dots, 2n \end{aligned} \quad (7)$$

where  $\beta \geq 0$  is used to incorporate the higher order information of the distribution, according to [15], for Gaussian distribution  $\beta = 2$  [29].

Step 4: Observation prediction.

$$\begin{aligned} \gamma_{k/k-1}^{(i)} &= h(\chi_{k/k-1}^{(i)}, k), i = 0, 1, 2, \dots, 2n \\ \hat{\mathbf{Z}}_{k/k-1} &= \sum_{i=0}^{2n} \omega_i^{(m)} \gamma_{k/k-1}^{(i)}. \end{aligned} \quad (8)$$

Step 5: Kalman gain calculation.

$$\begin{aligned} \mathbf{P}_{\mathbf{X}\mathbf{Z}} &= \sum_{i=0}^{2n} \omega_i^{(c)} (\chi_{k/k-1}^{(i)} - \hat{\mathbf{X}}_{k/k-1})(\gamma_{k/k-1}^{(i)} - \hat{\mathbf{Z}}_{k/k-1})^T \\ \mathbf{P}_{\mathbf{Z}\mathbf{Z}} &= \sum_{i=0}^{2n} \omega_i^{(c)} (\gamma_{k/k-1}^{(i)} - \hat{\mathbf{Z}}_{k/k-1})(\gamma_{k/k-1}^{(i)} - \hat{\mathbf{Z}}_{k/k-1})^T + \mathbf{R}_k \\ \mathbf{K}_k &= \mathbf{P}_{\mathbf{X}\mathbf{Z}} \mathbf{P}_{\mathbf{Z}\mathbf{Z}}^{-1}. \end{aligned} \quad (9)$$

Step 6: State estimation and error covariance matrix update.

$$\begin{aligned} \hat{\mathbf{X}}_k &= \hat{\mathbf{X}}_{k/k-1} + \mathbf{K}_k (\mathbf{Z}_k - \hat{\mathbf{Z}}_{k/k-1}) \\ \mathbf{P}_k &= \mathbf{P}_{k/k-1} - \mathbf{K}_k \mathbf{P}_{\mathbf{Z}\mathbf{Z}} \mathbf{K}_k^T. \end{aligned} \quad (10)$$

Step 7: Iterate from steps 2 to 6 until all samples are completed.

Under the condition of time-varying process-error and measurement noise covariance, we can infer that if  $Q_{k-1}$  in (6) and  $\mathbf{R}_k$  in (9) could not be estimated timely, then inaccurate estimates would be made because, in (10),  $\hat{\mathbf{X}}_k$  is influenced by  $\mathbf{K}_k$  which is related to  $\mathbf{R}_k$  and  $\hat{\mathbf{X}}_{k/k-1}$ .

## 2.2. Adaptive R Estimation

As proved by Li et al. in [27], for simplicity, assuming  $\mathbf{Z}_1(k)$  and  $\mathbf{Z}_2(k)$  are independent redundant measurements of a signal  $\mathbf{Z}(k)$  from two sensors, they can be modeled as

$$\begin{cases} \mathbf{Z}_1(k) = \mathbf{Z}_T(k) + S_1(k) + \mathbf{V}_1(k) \\ \mathbf{Z}_2(k) = \mathbf{Z}_T(k) + S_2(k) + \mathbf{V}_2(k) \end{cases} \quad (11)$$

where  $\mathbf{Z}_T(k)$  denotes the true value of  $\mathbf{Z}(k)$ ,  $S_1(k)$  and  $S_2(k)$  are steady items of the measurement errors,  $\mathbf{V}_1(k)$  and  $\mathbf{V}_2(k)$  are uncorrelated zero-mean random white noise.

The first-order-self-difference (FOSD)  $\Delta\mathbf{Z}_1$ ,  $\Delta\mathbf{Z}_2$  and the second-order-mutual-difference (SOMD)  $\Delta\mathbf{Z}_{12}$  are defined as

$$\begin{cases} \Delta\mathbf{Z}_1(k) = \mathbf{Z}_1(k) - \mathbf{Z}_1(k-1) \\ \Delta\mathbf{Z}_2(k) = \mathbf{Z}_2(k) - \mathbf{Z}_2(k-1) \\ \Delta\mathbf{Z}_{12}(k) = \Delta\mathbf{Z}_1(k) - \Delta\mathbf{Z}_2(k) \end{cases} \quad (12)$$

Under the condition that the sampling interval is short enough,  $S_i(k) - S_i(k-1) \approx 0$ ,  $i = 1, 2$ . The covariance of the random noise for measurement  $\mathbf{Z}_1(k)$  and  $\mathbf{Z}_2(k)$  can be estimated as

$$\begin{cases} \mathbf{R}_1 = \frac{E[\Delta\mathbf{Z}_{12}(k)\Delta\mathbf{Z}_{12}(k)^T] + E[\Delta\mathbf{Z}_1(k)\Delta\mathbf{Z}_1(k)^T] - E[\Delta\mathbf{Z}_2(k)\Delta\mathbf{Z}_2(k)^T]}{4} \\ \mathbf{R}_2 = \frac{E[\Delta\mathbf{Z}_{12}(k)\Delta\mathbf{Z}_{12}(k)^T] - E[\Delta\mathbf{Z}_1(k)\Delta\mathbf{Z}_1(k)^T] + E[\Delta\mathbf{Z}_2(k)\Delta\mathbf{Z}_2(k)^T]}{4} \end{cases} \quad (13)$$

The mathematical expectations in (13) are calculated as follows, because the statistical characteristics are stable over a relatively short period.

$$\begin{cases} E[\Delta\mathbf{Z}_1(k)\Delta\mathbf{Z}_1(k)^T] \\ = \frac{1}{M} \sum_{m=0}^{M-1} \{\Delta\mathbf{Z}_1(k-m) - E[\Delta\mathbf{Z}_1(k)]\} \{\Delta\mathbf{Z}_1(k-m) - E[\Delta\mathbf{Z}_1(k)]\}^T \\ E[\Delta\mathbf{Z}_2(k)\Delta\mathbf{Z}_2(k)^T] \\ = \frac{1}{M} \sum_{m=0}^{M-1} \{\Delta\mathbf{Z}_2(k-m) - E[\Delta\mathbf{Z}_2(k)]\} \{\Delta\mathbf{Z}_2(k-m) - E[\Delta\mathbf{Z}_2(k)]\}^T \\ E[\Delta\mathbf{Z}_{12}(k)\Delta\mathbf{Z}_{12}(k)^T] \\ = \frac{1}{M} \sum_{m=0}^{M-1} \{\Delta\mathbf{Z}_{12}(k-m) - E[\Delta\mathbf{Z}_{12}(k)]\} \{\Delta\mathbf{Z}_{12}(k-m) - E[\Delta\mathbf{Z}_{12}(k)]\}^T \end{cases} \quad (14)$$

where  $E[\Delta\mathbf{Z}_i(k)]$ ,  $i = 1, 2$  and  $E[\Delta\mathbf{Z}_{12}(k)]$  can be calculated by

$$E[\Delta\mathbf{Z}_i(k)] = \frac{1}{M} \sum_{m=0}^{M-1} \Delta\mathbf{Z}_i(k-m) \quad (15)$$

$$E[\Delta\mathbf{Z}_{12}(k)] = \frac{1}{M} \sum_{m=0}^{M-1} \Delta\mathbf{Z}_{12}(k-m) \quad (16)$$

where  $M$  is the sliding window width which can be empirically set to 30~60 [25].

In practice, in order to capture the transient behavior of the variation of  $\mathbf{R}_k$  in time while considering the smoothness, a fading memory calculation is implemented as [30]

$$\begin{cases} \hat{\mathbf{R}}_k = (1 - d_k)\hat{\mathbf{R}}_{k-1} + d_k\mathbf{R}_k \\ d_k = \frac{1-b}{1-b^{k+1}}, (0 < b < 1) \end{cases} \quad (17)$$

where  $b$  is the fading factor and in our work is set to 0.980,  $\mathbf{R}_k$  represents the direct output in (13) at time  $k$ ,  $\hat{\mathbf{R}}_k$  is the final covariance matrix.

### 2.3. The Proposed Adaptive UKF Based on the Mahalanobis Distance of the Innovation Vector

As emphasized by many researchers [15,17,31], the parameters of the system model and the distribution of the measurement noise and the process noise could not be maintained as constants in practice all the time. In order to prevent the estimation from deteriorating or even diverging, caused by the model error, it is vital to determine and correct the mismatch between the real process error and the parameters' preset. Herein, in our proposed method, based on hypothesis testing theory [14], the Mahalanobis distance of innovation vector is employed as a criterion to identify whether the system modeling error exists or not [17].

For the sake of simplicity, only one sensor of (1) is taken into consideration, thus, the superscript  $i$  is dropped. This paper develops a new method to improve the adaptability of the classical UKF against process model error based on the Mahalanobis distance. However, if the measurement noise matrix  $\mathbf{R}_k$  is also contaminated, then one single sensor cannot cope with this case. Thanks to the RMNCE, in MSIF, the  $\mathbf{R}_k$  can be estimated and is immune to the state estimation.

Define the innovation sequence  $\varepsilon_k$  according to [15] as

$$\varepsilon_k = \mathbf{Z}_k - h(\hat{\mathbf{X}}_{k/k-1}) \quad (18)$$

Then  $\varepsilon_k$  should be zero-mean Gaussian-distributed with covariance  $E[\varepsilon_k\varepsilon_k^T] = \mathbf{H}_{k/k-1}\mathbf{P}_{k/k-1}\mathbf{H}_{k/k-1}^T + \hat{\mathbf{R}}_k$  [19],  $\mathbf{H}_{k/k-1} = \frac{\partial h}{\partial \mathbf{X}} \Big|_{\mathbf{X}=\hat{\mathbf{X}}_{k/k-1}}$ ; the square of the Mahalanobis distance of the innovation should be  $\chi^2$  distributed [4],

$$M^2 = \varepsilon_k^T \left\{ E[\varepsilon_k\varepsilon_k^T]^{-1} \right\} \varepsilon_k \sim \chi_m^2 \quad (19)$$

where  $m$  is the degree of freedom.

Based on the hypothesis testing theory, let  $\alpha$  be the given significance level, and then we have

$$\Pr(M^2 < \chi_{m,\alpha}^2) = 1 - \alpha \quad (0 \leq \alpha \leq 1) \quad (20)$$

where  $\Pr(\bullet)$  stands for the probability of a random event,  $\chi_{m,\alpha}^2$  denotes the  $\alpha$ -quantile of the distribution  $\chi_m^2$ .

If (20) does not hold, it can be deduced with high probability  $(1 - \alpha)$  that there exists process-error in the system (1), assuming that the observations are within reasonable bounds. Because there is more than one sensor in the system, (20) should be calculated for each sensor.

Different from [4,17], instead of a single parameter that acts on  $\mathbf{P}_{k/k-1}$

$$\mathbf{P}_{k/k-1}^* = \lambda_k \mathbf{P}_{k/k-1} = \lambda_k \left( \mathbf{P}_{\mathbf{X}\mathbf{X}} + \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^T \right) \quad (21)$$

We present a new algorithm that takes the advantage of sequences of both innovation and residual to correct the matrix  $\mathbf{Q}$  directly by a diagonal matrix, and the procedure is as follows.

According to [15], the residual  $\eta_k$  can be defined as

$$\eta_k = \mathbf{Z}_k - h(\hat{\mathbf{X}}_k) \quad (22)$$

According to [18], based on the principle of orthogonality [12], the residual sequences are uncorrelated from the measurements. So, the following equations should hold:

$$E(\eta_k \eta_k^T) = \hat{\mathbf{R}}_k - \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T. \quad (23)$$

If  $\mathbf{P}_k$  is replaced by its definition in (10), then

$$\mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T = \mathbf{H}_k (\mathbf{P}_{k/k-1} - \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T) \mathbf{H}_k^T. \quad (24)$$

The traces of both sides should be equal:

$$\begin{aligned} \text{tr}(\eta_k \eta_k^T) &= \text{tr}(\hat{\mathbf{R}}_k) - \text{tr}(\mathbf{H}_k (\mathbf{P}_{k/k-1} - \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T) \mathbf{H}_k^T) \\ &= \text{tr}(\hat{\mathbf{R}}_k) - \text{tr}(\mathbf{H}_k (\mathbf{P}_{k/k-1}^* + \Gamma_{k-1} \mathbf{Q} \Gamma_{k-1}^T) \mathbf{H}_k^T) + \text{tr}(\mathbf{H}_k \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T \mathbf{H}_k^T) \end{aligned} \quad (25)$$

where  $\mathbf{H}_k = \left. \frac{\partial h}{\partial \mathbf{X}} \right|_{\mathbf{X}=\hat{\mathbf{X}}_k}$ , and  $\mathbf{P}_{k/k-1}^*$  is the predicted covariance without the additive process noise. If (23) does not hold, there will be some change in matrix  $\mathbf{Q}$ . So, one scalar called the adaptive fading factor is generated based on the Equation (23) [16]:

$$\begin{aligned} \mathbf{P}_{k/k-1} &= \mathbf{P}_{k/k-1}^* + \Lambda_k \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^T \\ \Lambda_k &= \frac{\text{tr}(\hat{\mathbf{R}}_k) - \text{tr}(\mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T) + \text{tr}(\mathbf{H}_k \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T \mathbf{H}_k^T) - \text{tr}(\eta_k \eta_k^T)}{\text{tr}(\mathbf{H}_k \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^T \mathbf{H}_k^T)}. \end{aligned} \quad (26)$$

Instead of one scalar being calculated to tune the matrix  $\mathbf{Q}$ , much higher accuracy could be obtained by using a matrix  $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_m\}$ , in which  $m$  is the dimension of the  $\mathbf{Q}$ .

So, an innovative adaptive method is proposed as

$$\mathbf{P}_k = \mathbf{P}_{k/k-1}^* + \Gamma_{k-1} \Lambda_k \mathbf{Q}_{k-1} \Gamma_{k-1}^T, \Lambda_k = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_m\} \quad (27)$$

$$\Lambda_k = \frac{1}{2} (\mathbf{H}_k \Gamma_{k-1})^{-1} \left\{ E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - E[\eta_k \eta_k^T] + E[(\boldsymbol{\varepsilon}_k - \eta_k)(\boldsymbol{\varepsilon}_k - \eta_k)^T] - 2\mathbf{H}_k \mathbf{P}_{k/k-1}^* \mathbf{H}_k^T \right\} (\Gamma_{k-1}^T \mathbf{H}_k^T)^{-1} \mathbf{Q}_{k-1}^{-1}. \quad (28)$$

To obtain (28), firstly, the relationship of  $E[(\boldsymbol{\varepsilon}_k - \eta_k)(\boldsymbol{\varepsilon}_k - \eta_k)^T]$ ,  $E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T]$  and  $E[\eta_k \eta_k^T]$  are given; then the matrix  $\Lambda_k$  is solved but not in the form of the trace of a matrix.

The covariance of innovation is:

$$E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] = \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \hat{\mathbf{R}}_k. \quad (29)$$

The covariance of difference sequences between innovation and residual is

$$E[(\boldsymbol{\varepsilon}_k - \eta_k)(\boldsymbol{\varepsilon}_k - \eta_k)^T] = E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] + E[\eta_k \eta_k^T] - E[\boldsymbol{\varepsilon}_k \eta_k^T] - E[\eta_k \boldsymbol{\varepsilon}_k^T]. \quad (30)$$

A deep look should be taken at  $\boldsymbol{\varepsilon}_k$  and  $\eta_k$  before calculating (30)

$$\begin{aligned} \boldsymbol{\varepsilon}_k &= \mathbf{Z}_k - h(\hat{\mathbf{X}}_{k/k-1}) = h(\mathbf{X}_k) + \mathbf{V}_k - h(\hat{\mathbf{X}}_{k/k-1}) \\ &\approx \mathbf{H}_{k/k-1} (\mathbf{X}_k - \hat{\mathbf{X}}_{k/k-1}) + \mathbf{V}_k \\ &= \mathbf{H}_{k/k-1} \tilde{\mathbf{X}}_{k/k-1} + \mathbf{V}_k \\ \eta_k &= \mathbf{Z}_k - h(\hat{\mathbf{X}}_k) = h(\mathbf{X}_k) + \mathbf{V}_k - h(\hat{\mathbf{X}}_k) \\ &\approx \mathbf{H}_k (\mathbf{X}_k - \hat{\mathbf{X}}_k) + \mathbf{V}_k \\ &= \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \end{aligned} \quad (31)$$



where  $\tilde{\mathbf{X}}_{k/k-1}$  and  $\tilde{\mathbf{X}}_k$  represent the prediction error and the estimation error, respectively. The first two items in the right side of (30) can be derived based on (31) as

$$\begin{aligned} E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] &= E\left[\left(\mathbf{H}_{k/k-1} \tilde{\mathbf{X}}_{k/k-1} + \mathbf{V}_k\right)\left(\mathbf{H}_{k/k-1} \tilde{\mathbf{X}}_{k/k-1} + \mathbf{V}_k\right)^T\right] \\ &= \mathbf{H}_{k/k-1} \mathbf{P}_{k/k-1} \mathbf{H}_{k/k-1}^T + \hat{\mathbf{R}}_k \\ E[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T] &= E\left[\left(\mathbf{H}_k (\mathbf{X}_k - \hat{\mathbf{X}}_k) + \mathbf{V}_k\right)\left(\mathbf{H}_k (\mathbf{X}_k - \hat{\mathbf{X}}_k) + \mathbf{V}_k\right)^T\right] \\ &= \hat{\mathbf{R}}_k - \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T. \end{aligned} \quad (32)$$

Because the sample frequency is high, the Jacobian matrices  $\mathbf{H}_{k/k-1} \approx \mathbf{H}_k$ , and  $E[\boldsymbol{\varepsilon}_k \boldsymbol{\eta}_k^T] \approx E[\boldsymbol{\eta}_k \boldsymbol{\varepsilon}_k^T]$ , Ge et al. gave the proof in [15]

$$\begin{aligned} E[\boldsymbol{\varepsilon}_k \boldsymbol{\eta}_k^T] &= E\left[\left(\mathbf{H}_{k/k-1} (\tilde{\mathbf{X}}_{k/k-1}) + \mathbf{V}_k\right)\left(\mathbf{H}_{k/k-1} (\tilde{\mathbf{X}}_k) + \mathbf{V}_k\right)^T\right] \\ &= \mathbf{H}_{k/k-1} \mathbf{P}_k \mathbf{H}_k^T + \hat{\mathbf{R}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k^T). \end{aligned} \quad (33)$$

Then, based on (32) and (33), namely,

$$\begin{aligned} E\left[(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)^T\right] &= E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] + E[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T] - E[\boldsymbol{\varepsilon}_k \boldsymbol{\eta}_k^T] - E[\boldsymbol{\eta}_k \boldsymbol{\varepsilon}_k^T] \\ &\approx \mathbf{H}_{k/k-1} \mathbf{P}_{k/k-1} \mathbf{H}_{k/k-1}^T + \hat{\mathbf{R}}_k + \hat{\mathbf{R}}_k - \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T \\ &\quad - 2(\mathbf{H}_{k/k-1} \mathbf{P}_k \mathbf{H}_k^T - \hat{\mathbf{R}}_k \mathbf{K}_k \mathbf{H}_k^T + \hat{\mathbf{R}}_k) \\ &= \mathbf{H}_{k/k-1} \mathbf{P}_{k/k-1} \mathbf{H}_{k/k-1}^T - \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T \\ &\approx \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T - \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T \\ &= \mathbf{H}_k (\mathbf{P}_{k/k-1} - \mathbf{P}_k) \mathbf{H}_k^T \\ &= \mathbf{H}_k \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T \mathbf{H}_k^T \end{aligned} \quad (34)$$

According to (29), replacing  $\hat{\mathbf{R}}_k$  by  $E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T$  in (25) can be rewritten without trace calculation based on (32), (33), and  $\mathbf{P}_{k/k-1} = \mathbf{P}_{k/k-1}^* + \boldsymbol{\Gamma}_{k-1} \boldsymbol{\Lambda}_k \mathbf{Q}_{k-1} \boldsymbol{\Gamma}_{k-1}^T$

$$\begin{aligned} E[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T] &= E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T - \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{H}_k \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T \mathbf{H}_k^T \\ &= E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - 2\mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{H}_k \mathbf{K}_k \mathbf{P}_{ZZ} \mathbf{K}_k^T \mathbf{H}_k^T \\ &= E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - 2\mathbf{H}_k (\mathbf{P}_{k/k-1}^* + \boldsymbol{\Lambda}_k \boldsymbol{\Gamma}_{k-1} \mathbf{Q}_{k-1} \boldsymbol{\Gamma}_{k-1}^T) \mathbf{H}_k^T + E\left[(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)^T\right]. \end{aligned} \quad (35)$$

Hence, (28) is obtained by rearranging (35). Normally, the covariance equations stacked above are calculated as

$$\begin{aligned} E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] &= \frac{1}{M} \sum_{m=0}^{M-1} (\boldsymbol{\varepsilon}_{k-m} - \bar{\boldsymbol{\varepsilon}})(\boldsymbol{\varepsilon}_{k-m} - \bar{\boldsymbol{\varepsilon}})^T \\ E[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T] &= \frac{1}{M} \sum_{m=0}^{M-1} (\boldsymbol{\eta}_{k-m} - \bar{\boldsymbol{\eta}})(\boldsymbol{\eta}_{k-m} - \bar{\boldsymbol{\eta}})^T \\ E\left[(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)^T\right] &= \frac{1}{M} \sum_{m=0}^{M-1} (\boldsymbol{\varepsilon}_{k-m} - \boldsymbol{\eta}_{k-m})(\boldsymbol{\varepsilon}_{k-m} - \boldsymbol{\eta}_{k-m})^T \end{aligned} \quad (36)$$

in which  $M$  is the windows size,  $\bar{\boldsymbol{\varepsilon}} = \frac{1}{M} \sum_{m=0}^{M-1} \boldsymbol{\varepsilon}_{k-m}$ , and  $\bar{\boldsymbol{\eta}} = \frac{1}{M} \sum_{m=0}^{M-1} \boldsymbol{\eta}_{k-m}$ .

**Remark 1.** In Equation (28), the inverse matrices of  $(\mathbf{H}_k \boldsymbol{\Gamma}_{k-1})^{-1}$  and  $\mathbf{Q}_{k-1}^{-1}$  are required to be calculated. In practice,  $\mathbf{Q}_{k-1}$  is normally a positive definite diagonal matrix, but  $\mathbf{H}_k \boldsymbol{\Gamma}_{k-1}$  may be not an invertible matrix, or not even a square matrix. Considering that  $\boldsymbol{\Lambda}_k$  is a diagonal matrix, we can obtain the elements by solving the following equation:

$$\mathbf{H}_k \boldsymbol{\Gamma}_{k-1} \boldsymbol{\Lambda}_k \mathbf{Q}_{k-1} \boldsymbol{\Gamma}_{k-1}^T \mathbf{H}_k^T = \frac{1}{2} \left\{ E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - E[\boldsymbol{\eta}_k \boldsymbol{\eta}_k^T] + E\left[(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)(\boldsymbol{\varepsilon}_k - \boldsymbol{\eta}_k)^T\right] - 2\mathbf{H}_k \mathbf{P}_{k/k-1}^* \mathbf{H}_k^T \right\}. \quad (37)$$

**Remark 2.** In Equation (37),  $\mathbf{H}_k$  and  $\mathbf{H}_k^T$  are the Jacobian matrices. They normally can be considered as the slope at  $\hat{\mathbf{X}}_k$ . During the course of numerical calculation,  $\mathbf{H}_k$  and  $\mathbf{H}_k^T$  may be very small, so one or more elements of  $\mathbf{\Lambda}_k$  will be large enough to cause  $\mathbf{P}_k$  to be a negative definite in the next step. To solve this problem, a threshold value is used to limit the element of  $\mathbf{\Lambda}_k$ . Meanwhile, if  $\lambda_i$  is not positive in numerical applications, it is always reset to the absolute value of its estimate [28].

**Remark 3.** In (28) or (37), the result of the first three items in the right side of equations may be much greater than the last item, because the assumption that the innovation and residual sequences are orthogonal is statistically ideal. In other words, for a limited set of the sample, (36) is commonly biased [28]. However, in many real-time or online systems, the sliding window could not be set very large to meet the assumption. So, the tradeoff should be made between the numerical stability and real-time performance.

#### 2.4. The Decentralized MSIF

Due to the expensive computational cost for high-dimension matrices and low stability when some measurements are abnormal [10], i.e., one or more sensors provide information with large noises, or just noises, the centralized architecture is not adopted widely. This work adopts the decentralized MSIF structure similar to [1], which is derived from [32]. For simplicity, the time instant  $k$  in  $\mathbf{X}_k$  is dropped in this section [1].

Let  $\hat{\mathbf{X}}^i, i = 1, 2, \dots, l$  be unbiased estimators of  $\mathbf{X}$  in (1) and the estimation errors be  $\tilde{\mathbf{X}}^i = \mathbf{X} - \hat{\mathbf{X}}^i, i = 1, 2, \dots, l$ . Assume that  $\mathbf{X}^i$  and  $\mathbf{X}^j$  are correlated, the covariance and cross covariance are  $\mathbf{P}^{ii}$  and  $\mathbf{P}^{ij}$ , respectively. The optimal fusion estimator  $\hat{\mathbf{X}}_{op}$  with matrix weights can be described as

$$\hat{\mathbf{X}}_{op} = \sum_{i=1}^l \bar{\mathbf{A}}^i \hat{\mathbf{X}}^i = \bar{\mathbf{A}}^1 \hat{\mathbf{X}}^1 + \bar{\mathbf{A}}^2 \hat{\mathbf{X}}^2 + \dots + \bar{\mathbf{A}}^l \hat{\mathbf{X}}^l \quad (38)$$

where the optimal matrix weights  $\bar{\mathbf{A}}^i, i = 1, 2, \dots, l$  are to be determined.

The globally optimal information fusion Kalman filter  $\hat{\mathbf{X}}_{op}$  of the state  $\mathbf{X}$ , based on the principle of linear minimum variance, will satisfy the following conditions [1,17]:

**Condition 1.**  $\hat{\mathbf{X}}_{op}$  must be the unbiased estimation of  $\mathbf{X}$ , namely,  $E[\hat{\mathbf{X}}_{op}] = E[\mathbf{X}]$ .

**Condition 2.**  $\hat{\mathbf{X}}_{op}$  makes  $\text{tr}\{\mathbf{P}\}$  minimum, in which  $\mathbf{P}$  is the error covariance matrix of  $\hat{\mathbf{X}}_{op}$ .

For simplicity, we denote  $\mathbf{A} = [\mathbf{A}^1, \mathbf{A}^2, \dots, \mathbf{A}^l]^T$ . Our aim is to find  $\mathbf{A}$  to construct the unbiased estimator

$$\hat{\mathbf{X}} = \sum_{i=1}^l \mathbf{A}^i \hat{\mathbf{X}}^i = \mathbf{A}^1 \hat{\mathbf{X}}^1 + \mathbf{A}^2 \hat{\mathbf{X}}^2 + \dots + \mathbf{A}^l \hat{\mathbf{X}}^l \quad (39)$$

where  $\mathbf{A}^i, i = 1, 2, \dots, l$  are arbitrary matrices.

If  $\hat{\mathbf{X}}_{op}$  is an unbiased estimator for  $\mathbf{X}$ , the following condition should be fulfilled,

$$E[\hat{\mathbf{X}}_{op}] = E[\mathbf{X}] \quad (40)$$

Taking the expectation of both sides of (38) yields,

$$E[\hat{\mathbf{X}}_{op}] = E\left[\sum_{i=1}^l \mathbf{A}^i \hat{\mathbf{X}}^i\right] = \sum_{i=1}^l \mathbf{A}^i E[\hat{\mathbf{X}}^i]. \quad (41)$$

So (43) is obtained

$$\mathbf{I}_n = \sum_{i=1}^l \mathbf{A}^i = \mathbf{A}^1 + \mathbf{A}^2 + \dots + \mathbf{A}^l. \quad (42)$$

Based on (38) and (42), we get the fusion estimation error

$$\mathbf{X} = \mathbf{X} - \sum_{i=1}^l \mathbf{A}^i \hat{\mathbf{X}}^i = \sum_{i=1}^l \mathbf{A}^i (\mathbf{X} - \hat{\mathbf{X}}^i) = \sum_{i=1}^l \mathbf{A}^i \mathbf{X}^i. \quad (43)$$

The error variance matrix of the fusion estimator is

$$\mathbf{P} = E[\mathbf{X}\mathbf{X}^T] = \mathbf{A}^T \mathbf{\Sigma} \mathbf{A} \quad (44)$$

where  $\mathbf{\Sigma} = (\mathbf{P}_{ij})$   $i, j = 1, 2, \dots, l$  is a symmetric positive definite matrix.

Now the problem is converted to a classic one: under the constraint of (42), to solve the minimum of  $tr\{\mathbf{P}\} = tr\{\mathbf{A}^T \mathbf{\Sigma} \mathbf{A}\}$  by applying the Lagrange multiplier

$$F = tr\{\mathbf{P}\} + tr\{\mathbf{\Lambda}[\mathbf{A}^T \mathbf{E} - \mathbf{I}_n]\} \quad (45)$$

where  $\mathbf{\Lambda} \in \mathbb{R}^{n \times n}$  is the Lagrange multiplier,  $\mathbf{E} = [\mathbf{I}_n, \mathbf{I}_n, \dots, \mathbf{I}_n]^T \in \mathbb{R}^{nN \times n}$ ; and  $\mathbf{I}_n$  represents an  $n$ -dimensional identity matrix.

By setting  $\partial F / \partial \mathbf{A}|_{\mathbf{A}=\bar{\mathbf{A}}} = 0$ , and noting that  $\mathbf{\Sigma}^T = \mathbf{\Sigma}$ , we have

$$\mathbf{\Sigma} \bar{\mathbf{A}} + \mathbf{E} \mathbf{\Lambda} = 0 \quad (46)$$

By substituting (42) into (46)

$$\begin{pmatrix} \mathbf{\Sigma} & \mathbf{E} \\ \mathbf{E}^T & 0 \end{pmatrix} \begin{pmatrix} \bar{\mathbf{A}} \\ \mathbf{\Lambda} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{I}_n \end{pmatrix} \quad (47)$$

Because  $\mathbf{\Sigma}$  is a symmetric positive definite matrix,  $\mathbf{E}^T \mathbf{\Sigma}^{-1} \mathbf{E}$  is nonsingular. Based on the matrix theory, (47) can be solved

$$\begin{pmatrix} \bar{\mathbf{A}} \\ \mathbf{\Lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{\Sigma} & \mathbf{E} \\ \mathbf{E}^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \mathbf{I}_n \end{pmatrix} = \begin{pmatrix} \mathbf{\Sigma}^{-1} \mathbf{E} (\mathbf{E}^T \mathbf{\Sigma}^{-1} \mathbf{E})^{-1} \\ -(\mathbf{E}^T \mathbf{\Sigma}^{-1} \mathbf{E})^{-1} \end{pmatrix} \quad (48)$$

From (48), the matrix  $\mathbf{\Sigma}$  should be calculated to obtain the global optimal state estimation. The diagonal elements  $\mathbf{P}^{ii}$ ,  $i = 1, 2, \dots, l$  in the matrix  $\mathbf{\Sigma}$  can be directly calculated by the error covariance matrix of the state estimation in the  $i$ th local filter. However, the cross-covariance matrix is difficult to get. In our work, we use the result given by Gao et al. base on UT [17].

$$\begin{aligned} \mathbf{P}_k^{ij} &= \sum_{s=0}^{2n} \omega_s^{(c)} (\chi_{s,k/k-1}^{(i)} - \hat{\mathbf{X}}_{s,k/k-1}^{(i)}) (\chi_{s,k/k-1}^{(j)} - \hat{\mathbf{X}}_{s,k/k-1}^{(j)})^T \\ &\quad - \left[ \sum_{s=0}^{2n} \omega_s^{(c)} (\chi_{s,k/k-1}^{(i)} - \hat{\mathbf{X}}_{s,k/k-1}^{(i)}) (\gamma_{s,k/k-1}^{(j)} - \hat{\mathbf{Z}}_{s,k/k-1}^{(j)})^T \right] (\mathbf{K}^{(j)})^T \\ &\quad - \mathbf{K}^{(i)} \left[ \sum_{s=0}^{2n} \omega_s^{(c)} (\gamma_{s,k/k-1}^{(i)} - \hat{\mathbf{Z}}_{s,k/k-1}^{(i)}) (\chi_{s,k/k-1}^{(j)} - \hat{\mathbf{X}}_{s,k/k-1}^{(j)})^T \right] \\ &\quad + \mathbf{K}^{(i)} \left[ \sum_{s=0}^{2n} \omega_s^{(c)} (\gamma_{s,k/k-1}^{(i)} - \hat{\mathbf{Z}}_{s,k/k-1}^{(i)}) (\gamma_{s,k/k-1}^{(j)} - \hat{\mathbf{Z}}_{s,k/k-1}^{(j)})^T \right] (\mathbf{K}^{(j)})^T \end{aligned} \quad (49)$$

where  $\chi_{s,k/k-1}^{(i)}$  is the sigma point transformed by the nonlinear function  $f(\bullet)$  in (6) for the  $i$ th sensor;  $\gamma_{s,k/k-1}^{(j)}$  represents the sigma points transformed by the nonlinear function  $h(\bullet)$  in (8); and  $s = 0, 1, \dots, 2n$  denotes the order of the transformed sigma points.

**Remark 4.** In order to maintain  $\mathbf{P}_k^{ij}$  as a positive definite, some exceptions should be taken into consideration. Firstly, Equation (42) should be examined immediately when  $\Sigma$  is fulfilled by  $\mathbf{P}_k^{ij}$ . This step is necessary because there may be truncation errors during calculation. If the constraint defined by (42) is not satisfied, set  $\mathbf{P}_k^{ij} = \mathbf{P}_{k-1}^{ij}$ .

Secondly, in MSIF, the optimal estimation  $\hat{\mathbf{X}}_{op}$  theoretically lies in the closed interval:

$$\min(\text{norm2}(\hat{\mathbf{X}}^i)) \leq \text{norm2}(\hat{\mathbf{X}}_{op}) \leq \max(\text{norm2}(\hat{\mathbf{X}}^i)), \quad i = 1, 2, \dots, l \quad (50)$$

where  $\text{norm2}(\bullet)$  denotes the 2-norm. If (50) is not maintained, the following degenerative methods (52) or (53) can be used:

$$\mathbf{P}_k^{ij}(\text{row}, \text{col}) = 0. \text{ when row} \neq \text{col} \quad (51)$$

$$\hat{\mathbf{X}}_{op} = \frac{1}{l} \sum_{i=1}^l \hat{\mathbf{X}}^i. \quad (52)$$

### 3. The Proposed Method with Both Adaptivity and Robustness

In Section 2, we proposed the innovative algorithm to adapt  $\mathbf{Q}$  when process-error exists. In this section, based on the hypothesis theory, the adopted judging criterion on process-error detection is given [4,17]. However, the drawbacks of the methods proposed by [4] and [17] are that the  $\mathbf{Q}$  and  $\mathbf{R}$  could not be estimated at the same time. Severe problems would be caused under the dilemma to decide which one should be adapted:  $\mathbf{Q}$ ,  $\mathbf{R}$ , or both? To conquer this challenge, the RMNCE is employed to estimate the noise covariance of each sensor, then a decision is made whether the matrix  $\mathbf{R}$  is suffering gross errors or not. Based on RMNCE, the decision is made easier by MSIF because the estimations of  $\mathbf{R}$  are obtained with relatively high accuracy.

Briefly, in our proposed MSIF architecture, the matrix  $\mathbf{R}$  of each sensor is estimated by RMNCE, and is denoted as  $\hat{\mathbf{R}}_k^i$ ; meanwhile, the process-error can be corrected by adapting  $\mathbf{Q}$  if necessary. For the sake of simplicity, only one sensor of (1) is taken into consideration, so the superscript  $i$  in  $\hat{\mathbf{R}}_k^i$  could be dropped.

#### 3.1. Robust $\mathbf{R}$ Estimation Based on RMNCE

As described in Section 2.2, the main advantage of RMNCE is that the estimate of variance is based only on measurements and hence can be immune to the state estimation error [27]. So,  $\hat{\mathbf{R}}_k$  calculated by (17) can be considered as the benchmark to test whether the process error exists or not. Let  $\mathbf{R}_{in}$  be calculated by the following equation [28]

$$\mathbf{R}_{in} = E[\boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T] - \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T. \quad (53)$$

The difference between  $\mathbf{R}_{in}$  and  $\hat{\mathbf{R}}_k$  could be large if process error exists. The quotient is used as an indicator to detect the process error as

$$\text{indicator} = \frac{\text{tr}(\hat{\mathbf{R}}_k)}{\text{tr}(\mathbf{R}_{in})} = \begin{cases} [0.90, 1.10] : \text{normal} \\ \text{else} : \text{process error may exist} \end{cases} \quad (54)$$

where  $\text{tr}(\hat{\mathbf{R}}_k)$  and  $\text{tr}(\mathbf{R}_{in})$  denote the trace of  $\hat{\mathbf{R}}_k$  and  $\mathbf{R}_{in}$ , respectively.

When the quotient is around 1.0, it could be deduced with great probability that there is no process error. In this paper, a closed interval from 0.90 to 1.10 is used as the normal range. Otherwise, if the indicator is larger than 1.10 or smaller than 0.90, there is process error with great probability.

**Remark 5.** If the diagonal elements differ by more than one order of magnitude, it is better to calculate the indicator separately. For example, the matrices  $\hat{\mathbf{R}}_k$  and  $\mathbf{R}_{in}$  are as

$$\hat{\mathbf{R}}_k = \begin{bmatrix} 10 & 0 \\ 0 & 0.1 \end{bmatrix}, \mathbf{R}_{in} = \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}. \quad (55)$$

Although the indicator calculated by (54) belongs to the closed interval,  $\frac{\text{tr}(\hat{\mathbf{R}}_k)}{\text{tr}(\mathbf{R}_{in})} = \frac{10.01}{11} = 0.918 \in [0.90, 1.10]$ , it is obvious that the second element in the diagonal differs 10 times in  $\mathbf{R}_{in}$  than  $\hat{\mathbf{R}}_k$ .

**Remark 6.** The sliding window width for calculating  $\mathbf{R}_{in}$  and  $\hat{\mathbf{R}}_k$  should be the same. In this paper, it is set to 50.

In our MSIF architecture, we also proposed the following matrix weights  $\bar{\mathbf{A}}_{R^i}$  as an alternative to (48).

Consider a MSIF system with three sensors, the noise covariance matrices  $\hat{\mathbf{R}}_i$ ,  $i = 1, 2, 3$  could be obtained by (13). Because the noise covariance matrices are usually diagonal, the following equation is employed to calculate the  $\bar{\mathbf{A}}_{R^i}$

$$\bar{\mathbf{A}}_{R^i}(j, j) = \text{diag} \left\{ \frac{1}{\hat{R}^i(j, j)} \times \left( \sum_{l=1}^3 \frac{1}{\hat{R}^l(j, j)} \right) \right\}, j = 1, 2, \dots, m. \quad (56)$$

where  $m$  is the dimension of  $\mathbf{Z}^i$ .

**Remark 7.** In our proposed MSIF structure, all the sensors have the same measurement model, so all the  $\bar{\mathbf{A}}_{R^i}$  have the same dimension. If a MSIF system has different sensors, the variation of (56) should be derived specifically. However, the core idea is that the smaller  $\hat{R}^i(j, j)$  is, the greater weight  $\bar{\mathbf{A}}_{R^i}(j, j)$  is. A degenerated case is that if one or more elements of  $\mathbf{X}$  are observed only by one sensor, assuming sensor  $i$ , then

$$\bar{\mathbf{A}}_{R^i} = \begin{bmatrix} \mathbf{I}^r & \mathbf{0}^r \\ \mathbf{0}^{m-r} & \bar{\mathbf{A}}_{R^i}^{m-r} \end{bmatrix}, \bar{\mathbf{A}}_{R^j} = \begin{bmatrix} \mathbf{0}^r & \mathbf{0}^r \\ \mathbf{0}^{m-r} & \bar{\mathbf{A}}_{R^j} \end{bmatrix}, j \neq i \quad (57)$$

where  $r$  denotes the number of elements in  $\mathbf{X}$  that are only observed by sensor  $i$ . Hence, this is not a MSIF anymore, and this paper would not discuss this case any further.

**Remark 8.** When there are  $l \geq 2$  sensors in a MSIF system, there are basically two options to utilize (13):

**Option 1.** To calculate two matrices  $\hat{\mathbf{R}}_k$  one time, so (13) and the relative equations will be run  $l - 2$  times at least.

**Option 2.** To calculate all matrices  $\hat{\mathbf{R}}_k$  one time, then the classical least square method is used to solve the overdetermined equations, which are derived by the variation of (13). In this paper, Option 1 is adopted.

### 3.2. The Adaptive and Robust UKF Algorithm for MSIF (ARUKF-MSIF)

In this subsection, the complete scheme is given. The proposed Algorithm 1 ARUKF-MSIF aimed at target tracking can be implemented as follows.

**Algorithm 1.** ARUKF-MSIF algorithm

Initiation: initiate the  $l$  sensors' filters with  $\hat{\mathbf{X}}_0^i$ ,  $\mathbf{P}_0^{ij}$ ,  $\mathbf{P}_0^{ij}$ ,  $\alpha$ , and  $\chi_{m,\alpha}^2$ ;

Step 1: State prediction through (5)–(7) for each sensor.

Step 2: Observation prediction through (7) and (8) for each sensor.

Step 3: Estimate matrix  $\mathbf{R}^i$  for each sensor through (12) to (17).

Step 4: Process-error judgment through (20) and (54).

Step 5: Abnormal innovation distinguishing.

5.1 If (20) and (54) holds:

5.1.1 Go to step 6.

5.2 Else:

5.2.1 Adapt  $\mathbf{Q}$  through (28).

Step 6: Calculate Kalman gain and filtering through (9) and (10).

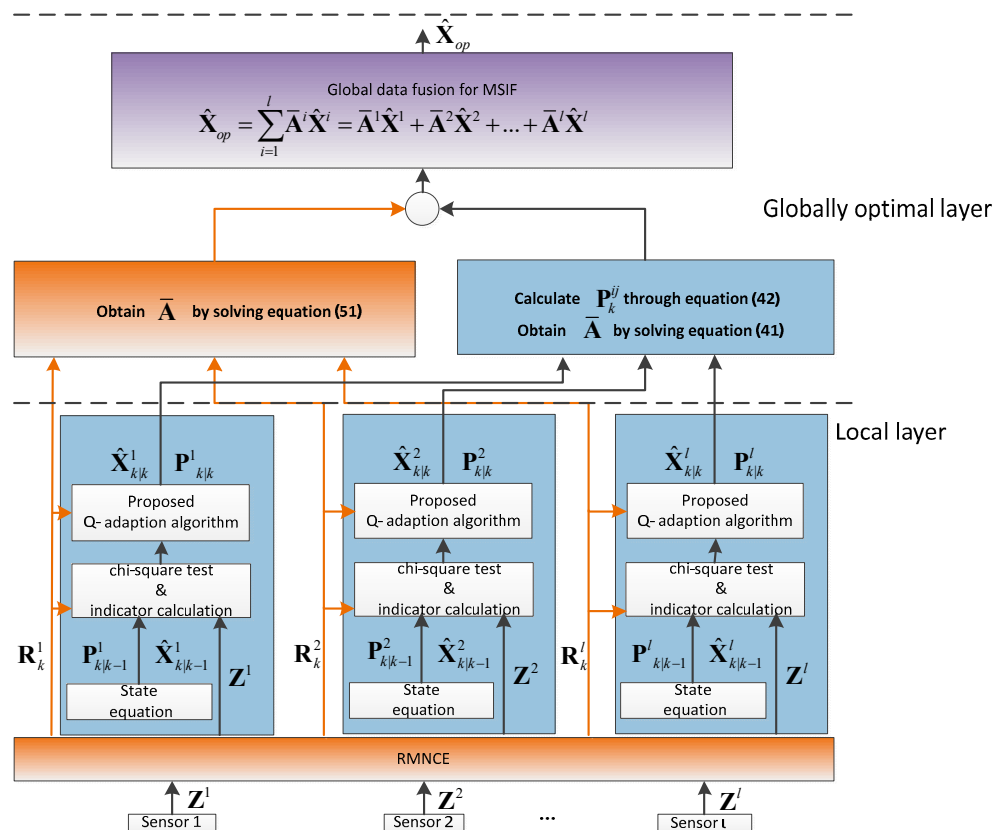
Step 7: MSIF implementation.

7.1 Calculate  $\mathbf{P}^{ij}$  through (49).

7.2 Calculate matrix weights  $\bar{\mathbf{A}}$  through (48) or (56), generate the optimal estimation  $\hat{\mathbf{X}}_{op}$ .

Step 8: For the next iteration, repeat steps from 1 to 7.

The framework of the proposed ARUKF-MSIF methodology is shown in Figure 1. It has a two-layer fusion structure, the local layer and the globally optimal layer.



**Figure 1.** The framework of the proposed ARUKF-MSIF method.

Based on RMNCE, the measurement noise covariance of each sensor is estimated. Every sensor estimates the states independently in the local layer. If any sensor subsystem detects the process-error by the chi-square test or the indicator proposed by this paper suggests the existence of the process-error, then the proposed  $\mathbf{Q}$ -adaption algorithm is employed to correct this mismatch. The globally optimal layer is the final fusion center, where the optimal matrix weights are determined.

#### 4. Simulations and Discussion

In this section, a set of numerical simulations of the radar tracking problems will be presented to illustrate the effectiveness of the proposed ARUKF-MSIF.

##### 4.1. Process Model and Measurement Model

Similar to [1], consider the radar tracking system with three sensors. The differences between our model and [1] are threefold.

Firstly, our model is more complicated and challenging: a two-dimension trajectory with time-varying process-error and/or measurement noise. In order to illustrate the effectiveness of our proposed method, the uncertainty of the system contains not only the measurement outliers/failures argued in [1] but also the process-error as in [15], which is usually the case in practice.

Secondly, all the sensors have the same measurement model in our simulation. With the development of hardware technology, more and more systems can afford redundant equipment to apply the multi-sensor information fusion into practical projects because the price of hardware is getting cheaper and cheaper. As proved by [6], if the sensors in decentralized structure have identical measurement matrices as in the centralized one, the two fusion methods are functionally equivalent. By taking advantage of the much smaller computational burden of the decentralized structure and the same accuracy as the centralized one, the simulation is arranged with three identical sensors whose noise characteristics are different.

Thirdly, the 2-D motion is taken into consideration as with [15,31]. The function of the radar can be demonstrated well enough; meanwhile, the model is not so complicated as the 3-D one. The simulated target trajectory is depicted in (Figure 2), and the true acceleration is shown in (Figure 3).

Consider that a target trajectory is in  $x$ - $y$  plane. The position, velocity, and acceleration of the objective at time  $k$  are represented by the vector  $X_k = [x_k, y_k, \dot{x}_k, \dot{y}_k, \ddot{x}_k, \ddot{y}_k]$  in the Cartesian coordinates. The dynamic equation for the target movement is as [15,31],

$$X_k = \begin{bmatrix} 1 & 0 & T & 0 & T^2/2 & 0 \\ 0 & 1 & 0 & T & 0 & T^2/2 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} X_{k-1} + \Gamma_{k-1} W_{k-1}, \quad \Gamma_{k-1} = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (58)$$

where  $T$  denotes the sampling period.

The initial state is  $X_0 = [1000 \text{ m}, 5000 \text{ m}, 10 \text{ m/s}, 50 \text{ m/s}, 2 \text{ m/s}^2, -4 \text{ m/s}^2]$ , the process noise covariance matrix is

$$Q_{k-1} = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix} \quad (59)$$

The multi-sensor observation systems are composed by three radars with the same measurement model described as

$$Z_k^i = \begin{bmatrix} r_k^i \\ \phi_k^i \end{bmatrix} = \begin{bmatrix} \sqrt{x_k^2 + y_k^2} \\ \arctan\left(\frac{y_k}{x_k}\right) \end{bmatrix} + V_k^i, \quad i = 1, 2, 3 \quad (60)$$

where  $r_k^i$  and  $\phi_k^i$  denote the slant range and azimuth angle in polar coordinates measured by the  $i$ th radar, respectively. The variance matrices of  $V_k^i$ ,  $i = 1, 2, 3$  are different from each other.

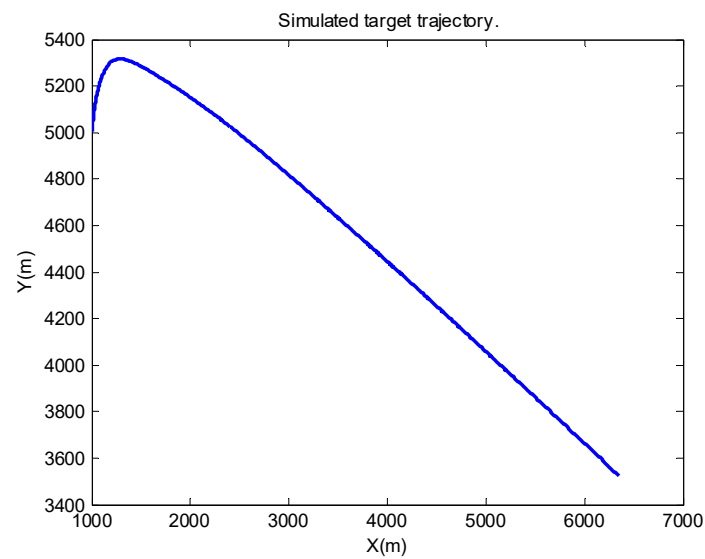


Figure 2. Flight trajectory of the target.

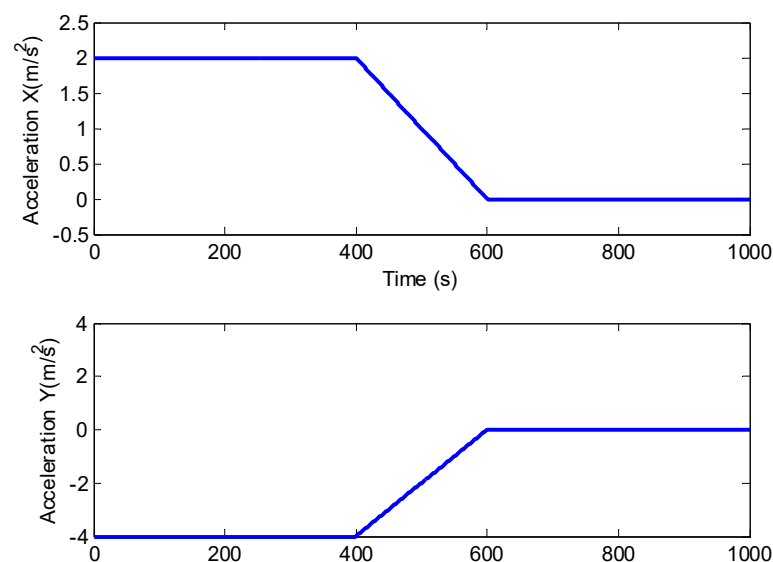


Figure 3. True acceleration during the trajectory.

#### 4.2. Simulations

To demonstrate the effectiveness of our proposed method, the following three cases are designed specifically.

**Case 1:** The noise covariance matrices  $\mathbf{R}^i = \text{diag}[i^2 \times 100, i^2 \times 0.01]$ ,  $i = 1, 2, 3$  are known and the process noise covariance matrix  $\mathbf{Q}$  varies over time. The process noise covariance matrix is assigned to be  $\mathbf{Q} = \text{diag}[0.015, 0.015]$  during the epochs [200, 400].

**Case 2:** The measurement noise covariance matrices  $\mathbf{R}^i$  are uncertain, and the process noise covariance matrix  $\mathbf{Q} = \text{diag}[0.001, 0.001]$  is known. The measurement noise covariance matrix  $\mathbf{R}^1$  is assigned to be  $\mathbf{R}^1 = 10 \times \text{diag}[1^2 \times 100, 1^2 \times 0.01]$  during the epochs [200, 400], and it is set to the same value as in Case 1 for the remaining periods.

**Case 3:** Not only the measurement noise covariance matrices  $\mathbf{R}^1$  but also the process noise covariance matrix  $\mathbf{Q}$  are uncertain. The changes in Case 1 and Case 2 are implemented simultaneously in this case.



Different from [15], in which the dynamic maneuver is simulated during a period when both the  $\mathbf{R}^i$  and the  $\mathbf{Q}$  stay unchanged, in our simulations, all cases are designed to set the interval of the maneuver to the same one when  $\mathbf{R}^1$  and/or  $\mathbf{Q}$  varies. So, the situations in our simulations are much harsher than the ones in [15]. The interacting multiple model (IMM) is employed by Ge et al. [15] to compare the algorithms; however, in this paper, we do not take the IMM method into consideration because its computational load is higher than the algorithm proposed by [15], and the complexity of our algorithm to adapt the  $\mathbf{Q}$  is similar to [15]. Moreover, the Markov transition matrix used in IMM is obtained on the basis of statistics on the system evolution [33], so in order to employ the IMM, the transition matrix should be available. However, it could not be obtained precisely in practice.

For Case 1, the reason why only the matrix  $\mathbf{Q}$  is uncertain is that we want to illustrate the effectiveness of our proposed method contrast to several algorithms which focus on the  $\mathbf{Q}$  adaption. Except for the standard UKF, the AUKF algorithm proposed by [19], the adaptive fading UKF (AFUKF) method developed by [18], and the N-UKF given by [32] are taken into consideration together with ours to run the simulation.

For Case 2, to test the validity of our proposed MSIF structure with RMNCE, only  $\mathbf{R}^1$  changes during the specified intervals. The algorithms contained in the simulation are the improved Sage–Husa adaptive method in [26], the N-UKF, the standard UKF, and ours.

It is natural for Case 3 that the algorithms with both adaptability and robustness claimed by their inventors are taken into consideration except for the standard UKF and the improved Sage–Husa adaptive method. So, the N-UKF, the robust adaptive UKF proposed by [34], and our proposed method are applied to track the target.

For each case,  $\mathbf{R}^1$  of the sensor 1 is used by every single method. From 50 times of the Monte Carlo simulation in all cases, the root mean square error (RMSE) of each moment is obtained as

$$RMSE_k = \sqrt{\frac{1}{N} \sum_{i=1}^N [(\hat{x}_k^i - x_k)^2 + (\hat{y}_k^i - y_k)^2]} \quad (61)$$

where  $N$  is the total times of simulation,  $(\hat{x}_k^i, \hat{y}_k^i)$  represents the filtering position of the target at time instant  $k$  in the  $i$ th simulation.

For the first case, the position tracking errors of the standard UKF, adaptive fading UKF, N-UKF, and our proposed method are shown in (Figure 4). To illustrate the performance of these algorithms clearly, we list the means and variance of the position tracking errors during the epochs [200, 600] and [601, 1000].

The epochs of Case 1 can be divided into three intervals: [0, 199], [200, 600], and [601, 1000]. The positioning errors during the first interval are close to each other because there is not process-modeling error. In the second interval, the maneuver of the target deteriorates the performance of the standard UKF. For the robust adaptive UKF, a modulation is made to prevent the algorithm divergence, i.e., a limitation is used for the adaptive factor. It can be seen clearly that all the methods except for the standard UKF can maintain their accuracy and robustness due to the  $\mathbf{Q}$  adaption strategy used. From Table 1, the conclusion can be made that our proposed algorithm can resist the influence from both the time-varying process and the maneuvering motion models. The positioning errors estimated by our method remained the lowest among all the algorithms.

For Case 2, the improved Sage–Husa adaptive algorithm proposed by Zhang et al. is introduced to verify the performance in  $\mathbf{R}$  estimation [26]. The position errors estimated by the algorithms mentioned above are shown in (Figure 5), the means and variances of the position tracking errors are listed in Table 2. The measurement noise variances estimated in the algorithms are shown in Figure 6.

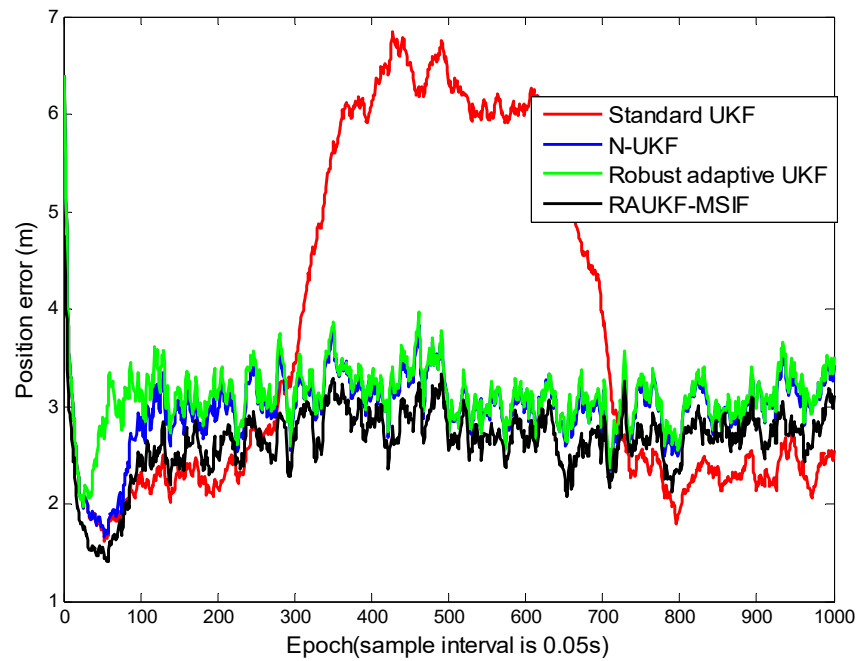


Figure 4. Position tracking errors for Case 1.

Table 1. Mean RMSEs of the overall estimation errors for the simulation Case 1. The bold numbers stand for the minimum.

Algorithm	200–600 Epochs		601–1000 Epochs	
	Mean (m)	Variance (m <sup>2</sup> )	Mean (m)	Variance (m <sup>2</sup> )
Standard UKF	4.9253	2.7772	2.7306	0.7438
Robust adaptive UKF	3.1699	0.0620	3.0537	0.0555
N-UKF	3.1010	0.0584	2.9953	0.0552
Our proposed method	<b>2.7508</b>	<b>0.0494</b>	<b>2.6785</b>	<b>0.0448</b>

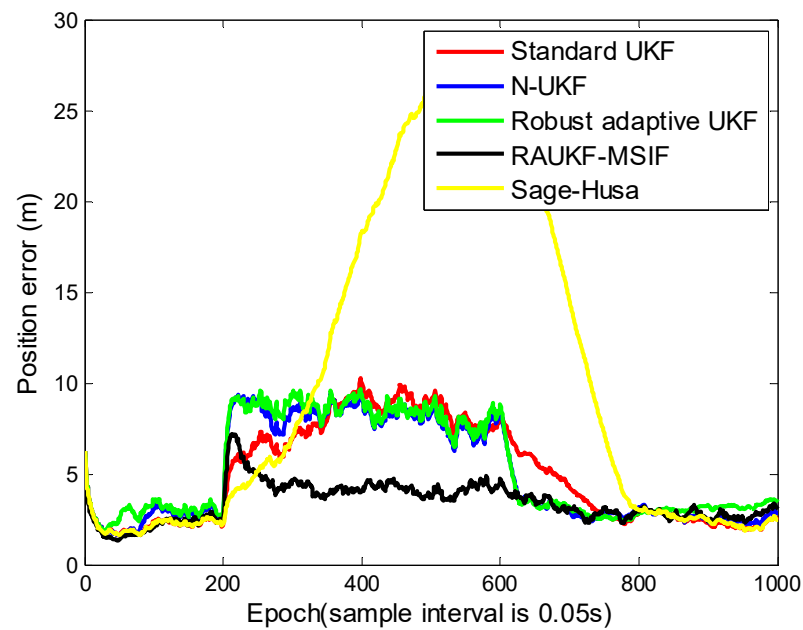
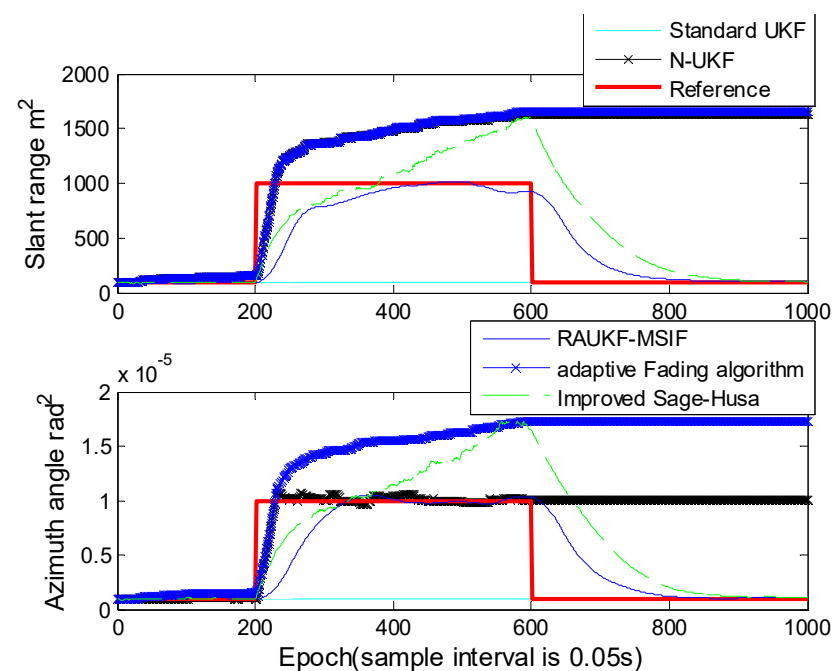


Figure 5. Position tracking errors for Case 2.

**Table 2.** Position tracking errors of the different algorithms for Case 2. The bold numbers stand for the minimum.

Algorithm	200–600 Epochs		601–1000 Epochs	
	Mean (m)	Variance (m <sup>2</sup> )	Mean (m)	Variance (m <sup>2</sup> )
Standard UKF	7.183457	4.194677	2.963046	0.946813
Robust adaptive UKF	7.516162	4.186171	3.026402	<b>0.083734</b>
N-UKF	7.297261	3.885044	2.731655	0.091129
Improved Sage–Husa UKF	15.99712	90.22968	6.232764	34.17588
Our proposed method	<b>4.160627</b>	<b>0.738772</b>	<b>2.848997</b>	0.095412



**Figure 6.** Estimated measurement noise by the algorithms in Case 2.

It can be seen from Figure 5 and Table 2 that the error of the improved Sage–Husa is much greater than other algorithms. This is because the innovation sequences used by this algorithm are contaminated by the maneuvering motion. The improved Sage–Husa method cannot maintain its accuracy or even diverges when process-error becomes non-negligible, although it can overcome the time-varying noise covariance of the measurement. Similar to N-UKF, the robust adaptive UKF can recognize the mismatch between the theoretical values and the calculated ones of  $\mathbf{Q}$  and/or  $\mathbf{R}$ , but it cannot provide the same accuracy as our proposed method because it cannot estimate the  $\mathbf{R}$  independently.

From Figure 6, we can see that the improved Sage–Husa algorithm could estimate the measurement noise variance. However, this method requires a much longer time to catch up with the reference values; moreover, this method would give a much greater overestimation than ours. The N-UKF and the adaptive fading algorithm can estimate the measurement noise variance only when the variance becomes greater, and they fail when the variance becomes smaller. After the epoch 600, the estimated values cannot be restored to the reference values. The values estimated by the standard UKF are unchanged, because the standard UKF does not have the ability to estimate the measurement noise variance.

For Case 3, from Figure 7 and Table 3, we can see that the standard UKF takes about 200 epochs (from epochs [601, 800]) longer to obtain the accuracy than the N-UKF, the adaptive fading UKF, and our proposed method. The reason is that in order to maintain its accuracy, the standard UKF could only rely on the accurate measurement under the condition in Case 3, this correction procedure has a relative long lag. As to the improved Sage–Husa method, it could be influenced relatively significantly by the process-error

more than other algorithms. The simulation result is similar as with Case 2, that the positioning errors estimated by the improved Sage–Husa algorithm are the greatest among the candidates. The N-UKF and the adaptive fading UKF have a similar accuracy because they can both resist the process-error and the time-varying measurement noise to some extent, but they cannot estimate the measurement noise variance as accurate as ours, as mentioned above in Case 2.

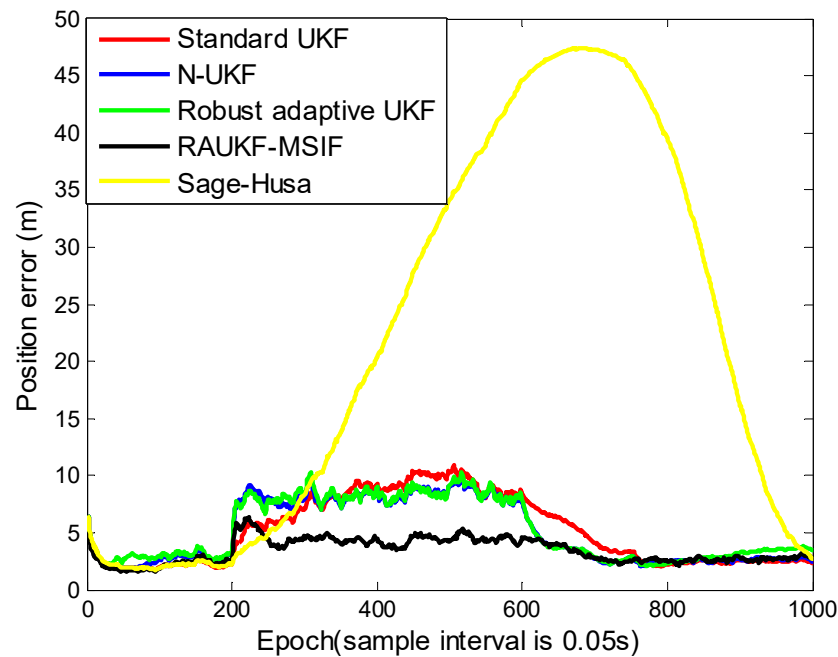


Figure 7. Position tracking errors for Case 3.

Table 3. Position tracking errors of the different algorithms for Case 3. The bold numbers stand for the minimum.

Algorithm	200–600 Epochs		601–1000 Epochs	
	Mean (m)	Variance (m <sup>2</sup> )	Mean (m)	Variance (m <sup>2</sup> )
Standard UKF	7.477111	5.486144	3.029508	1.137522
Robust adaptive UKF	7.433516	3.756174	2.982606	0.265064
N-UKF	7.350406	3.667636	<b>2.698935</b>	0.141301
Improved Sage–Husa UKF	22.03666	240.3023	29.90538	267.7623
Our proposed method	<b>4.260319</b>	<b>0.56904</b>	2.749053	<b>0.110879</b>

#### 4.3. Discussion

Base on the decentralized MSIF architecture, we proposed the RAUKF-MSIF to tackle the problems for the nonlinear target tracking systems with time-varying noise covariance. From the results simulated in Case 1, we can see that if there are time-varying noise covariances, the algorithms can detect the process-error and adapt the  $\mathbf{Q}$  except for the standard UKF.

In practice, the measurement noise covariance is also time-varying. So, in Case 2, we simulate the condition under which each sensors' measurement noise covariance is set to be 10 times greater during the specified interval than the others. From Figure 6, we can see that the estimation of  $\mathbf{R}$  by our proposed scheme is more accurate than the others due to RMNCE and the fusion strategy described in Section 2.2 and 3.1. The performance of the improved Sage–Husa is not as good as ours because it could not resist the process-error.

The harshest environment of the simulations is presented in Case 3, in which, together with maneuvering, the matrix  $\mathbf{Q}$  and  $\mathbf{R}$  are changed simultaneously in the same interval. As analyzed in Section 4.2 for Case 3, our proposed method can obtain a relatively high ac-

curacy due to both the process-error and the measurement noise variance can be estimated or adapted by our ARUKF-MSIF architecture.

Future work can be divided into two aspects. Firstly, we will focus on improving the numerical stability when adapting the matrix  $\mathbf{Q}$ . Although the algorithms including ours can adapt the matrix  $\mathbf{Q}$ , the adapted  $\mathbf{Q}$  would be a negative definite or the adapted scalar factor would be so much greater than desired that some certain limited values must be set to prevent the algorithms from diverging.

Secondly, in addition to adapting the matrix  $\mathbf{Q}$ , we would introduce the idea of IMM to estimate the dynamics of the target, while reducing the computational load of the IMM. Due to the existence of redundant measurement, we could improve the accuracy of the probability of the target moving from one model to another. Thus, a more precise model together with an adapted  $\mathbf{Q}$  would improve the estimation accuracy further.

## 5. Conclusions

This paper addresses the target tracking problem with both time-varying process-error and time-varying measurement noise covariance by using a multi-sensor information filter structure based on an adaptive UKF and the RMNCE. The proposed ARUKF-MSIF employs the RMNCE to estimate the measurement noise variance of each sensor. Subsequently, it uses the theory of hypothesis testing to identify process model uncertainty; next, an innovative algorithm uses both the innovation sequences and the residual sequences to correct the process-error. Then, the algorithm calculates the weight matrices, which are used to fusion the results obtained by each individual sensor. The simulation results demonstrate that the proposed architecture can be robust against the process model uncertainty and error. Moreover, in the harshest circumstances as simulated in Case 3, the proposed method would still maintain the lowest position error among the algorithms. Further, although this paper uses the UKF as the filter, the core idea of the ARUKF-MSIF can be to broaden the EKF, CKF, and other variants of Kalman filter.

**Author Contributions:** Conceptualization, H.Z., D.W. and B.G.; methodology, D.W.; software, D.W.; validation, D.W. and B.G.; formal analysis, D.W.; investigation, D.W.; resources, D.W.; data curation, D.W.; writing—original draft preparation, D.W.; writing—review and editing, H.Z., D.W. and B.G.; visualization, D.W.; supervision, H.Z.; project administration, H.Z.; funding acquisition, H.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by the National Key Research and Development Program of China (2017YFC0821102, 2016YFB0502004). This research is also funded by the Academic Excellence Foundation of BUAA for PhD Students.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sun, S.L.; Deng, Z.L. Multi-sensor optimal information fusion Kalman filter. *Automatica* **2004**, *40*, 1017–1023. [[CrossRef](#)]
2. Lin, H.L.; Sun, S.L. Globally optimal sequential and distributed fusion state estimation for multi-sensor systems with cross-correlated noises. *Automatica* **2019**, *101*, 128–137. [[CrossRef](#)]
3. Chirs, J.H.; Gan, Q. State estimation and multi-sensor data fusion using data-based neurofuzzy local linearization process models. *Inf. Fusion* **2001**, *2*, 17–29. [[CrossRef](#)]
4. Chang, G.B. Kalman filter with both adaptivity and robustness. *J. Process Control* **2014**, *24*, 81–87. [[CrossRef](#)]
5. Gao, Y.; Krakiwsky, E.J.; Abousalem, M.A.; McLellan, J.F. Comparison and analysis of centralized, decentralized, and federated Filters. *Navigation* **1993**, *40*, 69–86. [[CrossRef](#)]
6. Qiang, G.; Harris, C.J. Comparison of two measurement fusion methods for Kalman-filter-based multisensor data fusion. *IEEE Trans. Aerosp. Electron. Syst.* **2001**, *37*, 273–280. [[CrossRef](#)]
7. Qiu, H.Z.; Zhang, H.Y.; Jin, H. Fusion algorithm of correlated local estimates. *Aerosp. Sci. Technol.* **2004**, *8*, 619–626. [[CrossRef](#)]
8. Mahmoud, M.S.; Khalid, H.M. Distributed Kalman filtering: A bibliographic review. *IET Control Theory Appl.* **2013**, *7*, 483–501. [[CrossRef](#)]

9. Fank, O.H.; Edwin, E.Y. Robust minimum variance linear state estimators for multiple sensors with different failure rates. *Automatica* **2007**, *43*, 1274–1280. [[CrossRef](#)]
10. Caballero-Águila, R.; García-Garrido, I.; Linares-Pérez, J. Information fusion algorithms for state estimation in multi-sensor systems with correlated missing measurements. *Appl. Math. Comput.* **2014**, *226*, 548–563. [[CrossRef](#)]
11. Ren, Z.P.; Wang, H.L. Fusion Estimation Algorithm with Uncertain Noises and Its Application in Navigation System. *Math. Probl. Eng.* **2015**, *2015*. [[CrossRef](#)]
12. Zhang, W.A.; Shi, L. Sequential Fusion Estimation for Clustered Sensor Networks. *Automatica* **2017**, *89*, 358–363. [[CrossRef](#)]
13. Shen, B.; Wang, Z.D.; Tan, H.L.; Chen, H.W. Robust fusion filtering over multisensor systems with energy harvesting constraints. *Automatica* **2021**, *131*, 109782. [[CrossRef](#)]
14. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; John Wiley & Sons: New York, NY, USA, 2001; pp. 6–8.
15. Ge, B.S.; Zhang, H.; Jiang, L.L.; Li, Z.; Butt, M.M. Adaptive Unscented Kalman Filter for Target Tracking with Unknown Time-Varying Noise Covariance. *Sensors* **2019**, *19*, 1371. [[CrossRef](#)] [[PubMed](#)]
16. Salahshoor, K.; Mosallaei, M.; Bayat, M. Centralized and decentralized process and sensor fault monitoring using data fusion based on adaptive extended Kalman filter algorithm. *Measurement* **2008**, *41*, 1059–1076. [[CrossRef](#)]
17. Gao, B.B.; Hu, G.G.; Gao, S.S.; Zhong, Y.M.; Gu, C.F. Multi-Sensor Optimal Data Fusion Based on the Adaptive Fading Unscented Kalman Filter. *Sensors* **2018**, *18*, 488. [[CrossRef](#)]
18. Soken, H.E.; Hajiyev, C. Adaptive Fading UKF with Q-Adaptation: Application to Picosatellite Attitude Estimation. *Aerosp. Eng.* **2013**, *26*, 628–636. [[CrossRef](#)]
19. Mohamed, A.H.; Schwarz, K.P. Adaptive Kalman Filtering for INS/GPS. *J. Geod.* **1999**, *73*, 193–203. [[CrossRef](#)]
20. Yuan, D.; Zhang, J.; Wang, J.; Cui, X.; Liu, F.; Zhang, Y. Robustly Adaptive EKF PDR/UWB Integrated Navigation Based on Additional Heading Constraint. *Sensors* **2021**, *21*, 4390. [[CrossRef](#)]
21. Simon, D. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318. [[CrossRef](#)]
22. Yang, Y.; He, H.; Xu, G. Adaptively robust filtering for kinematic geodetic positioning. *J. Geod.* **2001**, *75*, 109–116. [[CrossRef](#)]
23. Ruggaber, J.; Brembeck, J. A Novel Kalman Filter Design and Analysis Method Considering Observability and Dominance Properties of Measurands Applied to Vehicle State Estimation. *Sensors* **2021**, *21*, 4750. [[CrossRef](#)]
24. Tang, C.; He, C.; Dou, L. An IMU/ODM/UWB Joint Localization System Based on Modified Cubature Kalman Filtering. *Sensors* **2021**, *21*, 4823. [[CrossRef](#)]
25. Zhou, Q.F.; Zhang, H.; Li, Y.; Li, Z. An Adaptive Low-Cost GNSS/MEMS-IMU Tightly-Coupled Integration System with Aiding Measurement in a GNSS Signal-Challenged Environment. *Sensors* **2015**, *15*, 23953–23982. [[CrossRef](#)]
26. Zhang, H.; Chang, Y.H.; Che, H. Measurement-based adaptive Kalman filtering algorithm for GPS/INS integrated navigation system. *Chin. Inert. Technol.* **2010**, *18*, 696–701.
27. Li, Z.; Zhang, H.; Zhou, Q.F.; Che, H. An Adaptive Low-Cost INS/GNSS Tightly-Coupled Integration Architecture Based on Redundant Measurement Noise Covariance Estimation. *Sensors* **2017**, *17*, 2032. [[CrossRef](#)]
28. Meng, Y.; Gao, S.; Zhong, Y.; Hu, G.; Subic, A. Covariance matching based adaptive unscented Kalman filter for direct filtering in INS/GNSS integration. *Acta Astronaut.* **2016**, *120*, 171–181. [[CrossRef](#)]
29. Julier, S.; Uhlmann, J.K. A General Method for Approximating Nonlinear Transformations of Probability Distributions. Available online: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.6718> (accessed on 17 December 2020).
30. Jiang, L.; Zhang, H. Redundant measurement-based second order mutual difference adaptive Kalman filter. *Automatica* **2019**, *100*, 396–402. [[CrossRef](#)]
31. Li, L.; Hua, C.; Yang, H. A new adaptive unscented Kalman filter based on covariance matching technique. In Proceedings of the 2014 International Conference on Mechatronics and Control, Jinzhou, China, 3–5 July 2014; pp. 1308–1313. [[CrossRef](#)]
32. Kim, K.H. Development of track to track fusion algorithm. In Proceeding of the American Control Conference, Baltimore, MD, USA, 29 June–1 July 1994; pp. 1037–1041. [[CrossRef](#)]
33. Ndjeng, A.N.; Gruyer, D.; Glaser, S.; Lambert, A. Low cost IMU-odometer-GPS ego localization for unusual maneuvers. *Inf. Fusion* **2011**, *12*, 264–274. [[CrossRef](#)]
34. Hajiyev, C. Robust adaptive unscented Kalman filter for attitude estimation of picosatellites. *Int. Adapt. Control Signal Process* **2014**, *28*, 107–120. [[CrossRef](#)]