Noise-Adaption Extended Kalman Filter Based on Deep Deterministic Policy Gradient for Maneuvering Targets

Jiali Li, Shengjing Tang and Jie Guo *

School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China; 3120185066@bit.edu.cn (J.L.); tangsj@bit.edu.cn (S.T.)
* Correspondence: guojie1981@bit.edu.cn

Abstract: Although there have been numerous studies on maneuvering target tracking, few studies have focused on the distinction between unknown maneuvers and inaccurate measurements, leading to low accuracy, poor robustness, or even divergence. To this end, a noise-adaption extended Kalman filter is proposed to track maneuvering targets with multiple synchronous sensors. This filter avoids the simultaneous adjustment of the process model and measurement model without distinction. Instead, the maneuver detection based on the Dempster-Shafer evidence theory is constructed to achieve the reliable distinction between unknown maneuvers and inaccurate measurements by fusing multi-sensor information, which effectively improves the robustness of the filter. Moreover, the adaptive estimation of the process noise covariance is modeled by a Markovian decision process with a proper reward function. Deep deterministic policy gradient is designed to obtain the optimal process noise covariance by taking the innovation as the state and the compensation factor as the action. Furthermore, the recursive estimation of the measurement noise covariance is applied to modify a priori measurement noise covariance of the corresponding sensor. Finally, the fusion algorithm is developed for the global estimation. Simulation experiments are carried out in two scenarios, and simulation results illustrate the feasibility and superiority of the proposed algorithm.

Keywords: maneuvering target; noise adaption; maneuver detection; Dempster-Shafer evidence theory; deep deterministic policy gradient

1. Introduction

With the development of high maneuvering targets, such as hypersonic aircrafts and missiles, maneuvering target tracking has recently drawn a lot of attention [1,2]. During the past few decades, lots of tracking methods have been developed and investigated for maneuvering targets [3–5]. Among the existing target tracking approaches, designing an appropriate model for the target maneuvering motion is regarded as an important part of decreasing the tracking errors caused by the model mismatch. Therefore, many maneuver models were proposed, such as constant acceleration (CA) [6], Singer model [7], current statistical model (CS) [8]. In contrast to single model methods, the interactive multiple model (IMM) algorithm [9,10] has been widely applied because of its adaptive capability for maneuvering target tracking problems. By mapping the motion mode to a model set, IMM performs the parallel filter for each model. Furthermore, based on the residual information and a priori information, the state estimation results of each model are weighted and synthesized to accurately achieve the maneuvering target tracking. Considering the limited performance of IMM based on the single platform, Wang et al. [11] proposed a multi-platform maneuvering target tracking algorithm based on IMM and the best linear unbiased estimate filter. Moreover, an adaptive interacting multiple model algorithm based on information-weighted consensus was proposed [12], which presented better adaptability and accuracy than the classical IMM. To cope with the miss detections and false alarms in a low-observable environment, a state-dependent IMM based on the
particle filter was presented [13]. However, the above IMM algorithms usually use a fixed set of models, so a large number of model sets is required when solving the problem of maneuvering target tracking. To alleviate this problem, the variable structure multiple model (VSMM) method has been proposed [14]. Furthermore, a VSMM version of the consensus filter was presented [15], where the expected mode augmentation was introduced to guarantee the model set adaption. In addition, the particle swarm optimization technique was utilized to improve VSMM [16]. However, for maneuvering target tracking in practice, multiple models cannot fully cover the modes of the target’s motion, which may degrade the tracking performance. Additionally, excessive numbers of models also generate large calculations in many practical applications.

On the other hand, the filtering algorithm is regarded as an important issue to cope with maneuvering target tracking problems. Practically, extended Kalman filter (EKF) [17] has been widely applied for many nonlinear systems during the past few years. EKF is developed based on the Kalman filter and introduces the first-order Taylor expansion to approximate the nonlinear function. However, the first-order Taylor expansion of EKF cannot, at times, satisfy the tracking accuracy. To this end, other nonlinear filter algorithms including unscented Kalman filter (UKF) [18,19] and cubature Kalman filter (CKF) [20] were proposed. By adopting the unscented transformation and spherical-radial principle, UKF and CKF can achieve second-order accuracy without calculating the Jacobian matrix. In order to further improve the robustness of the conventional filters, the strong tracking filter (STF) was proposed by Zhou et al. [21,22]. This algorithm introduces the orthogonal principle of residual sequences into the classical EKF to modify the prediction of the covariance matrix with a time-varying fading factor. When the state estimation of EKF deviates from the actual state, the effect of the old observations on the current filter estimation is reduced using the time-variant fading factor whereas the effect of the new observation is increased [23]. Accordingly, the robustness and accuracy of the classical EKF were improved. In order to improve the performance of STF, UKF and CKF were introduced to replace EKF [24,25]. An adaptive strong tracking particle filter algorithm was established by combing STF and the particle filter (PF), which demonstrated that the proposed method could provide better tracking precision than the classical methods [26]. Since the traditional strong tracking filter only considers the first-order Taylor expansion, a fast-strong tracking CKF was proposed [27], which expanded the number of fading factors from one to two. Sequentially, it extracted the third-order term information and achieved second-order accuracy. To implement higher tracking accuracy for the maneuvering target tracking, many improved strong tracking cubature Kalman filter (STCKF), such as strong tracking spherical simplex-radial CKF [28], fifth-degree STCKF [29], Bayesian-based strong tracking interpolatory CKF [30], model-based strong tracking square-root CKF [31], have recently been proposed. In addition, Ma et al. [32] added STF into the sub-filter of IMM to overcome the low tracking accuracy when dealing with maneuvering situations. However, in practical applications, it is found that the detection and tracking ability of the STF method will be reduced when the maneuver of the target is small. As is known, the residuals of range measurements are usually much bigger than the residuals of azimuth angle and elevation angle measurements, which leads STF to different sensitivities of different residual components. To this end, Jiang et al. [33] proposed a residual-normalized strong tracking filter (RNSTF) by designing the weight of the residual components. However, the process noise is not considered.

The aforementioned algorithms adopt the innovation, also known as the predictive residual, to realize maneuver detection and compensation, because the innovation will increase significantly when the model mismatch is caused by the unknown maneuver. However, the sensor signal may be disturbed or blocked due to the influence of a harsh environment, making the statistical characteristics of measurement noise uncertain and time varying. Inaccurate a priori measurement noise covariance can also lead to increased innovation. In that case, the aforementioned algorithms are too sensitive to inaccurate measurements, leading to filter divergence. To solve the problem of inaccurate a priori
knowledge, several adaptive Kalman filters (AKF) with noise sensing capability were established [34,35]. Unfortunately, the existing methods only can achieve the second-order estimation accuracy [36,37], and the adaptive KF cannot distinguish between model mismatch and inaccurate a priori measurement noise. In addition, adaptive KFs use the innovation to adjust the process noise and measurement noise simultaneously, which leads to poor stability.

Inspired by the strong representation capability of neural networks [38,39], several filters based on reinforcement learning have been studied. Hu et al. [40] designed an attitude estimator by combining Lyapunov’s method and the deep reinforcement learning algorithm. Tang et al. [41] combined the classic EKF with the deep reinforcement learning algorithm to realize the attitude estimation of the navigation system, which introduced a gain matrix of the residual and took it as the action to learn. Gao et al. [42] proposed an adaptive Kalman filter based on the deep deterministic policy gradient (DDPG) algorithm for ground vehicles, which took the integrated navigation system as the environment to obtain the process noise covariance matrix estimation. However, this adaptive filter takes the change in noise vector as the action to learn, resulting in poor filter stability. Additionally, the process noise covariance matrix of Inertial Navigation System (INS) error model will not experience a great change because it represents the inherent performance of the inertial sensor [42], which is different from the target tracking problem.

Motivated by the above investigation, a noise-adaption EKF is proposed in this paper to address the problem of maneuvering target tracking. Based on multiple synchronous sensors, the maneuver detection is constructed by utilizing Dempster-Shafer (D-S) evidence theory [43] to achieve the fused detection of multiple sensors. A Markovian decision process with a proper reward function is modeled for the adaptive estimation of the process noise covariance, and DDPG is designed to learn the compensation factor and feed it into EKF so that the improved filter can adaptively cope with the unknown maneuver. If the detection declares the occurrence of inaccurate measurement noise, the recursive measurement noise estimation is applied to the corresponding sensor. Finally, the local estimations are fused to obtain the target’s global estimation. When the unknown maneuver and mismatched measurement noise emerge, the proposed filter can correct noise covariance adaptively. Distinct from the aforementioned algorithms, the proposed filter avoids the simultaneous adjustment of process model and measurement model without distinction, which effectively improves the robustness of the filter. Moreover, the application the deep deterministic policy gradient method in solving maneuver target tracking problems is explored in this paper.

The remainder of this paper is organized as follows: Section 2 introduces mathematical models and the formula of EKF. Section 3 provides the maneuver detection based on D-S evidence theory in detail, and the framework of the noise-adaption EKF is presented. In Section 4, the process noise adaption based on DDPG is designed. Moreover, the recursive measurement noise estimation and the fusion algorithm are completed. Finally, simulation results and conclusions are shown in Sections 5 and 6, respectively.

2. Problem Formulation

The process model of the target is given by

\[ x_{k+1} = f(x_k) + w_k \]  

where \( k \) denotes the index of discrete-time. \( x_k \) represents the state vector, and \( f(\cdot) \) is the state transition function. The process noise \( w_k \) is assumed to be zero-mean Gaussian white noise with the covariance matrix \( Q_k \).

The measurement model of sensors can be represented as

\[ z_{k+1} = h(x_{k+1}) + v_{k+1} \]
where $z_{k+1}$ is the measurement vector, and $h(\cdot)$ is the measurement mapping function. $v_{k+1}$ is the measurement noise, whose covariance matrix is $R_{k+1}$.

Since EKF has the advantages of simple algorithm and fast convergence, it has been widely applied in nonlinear systems [17]. Specifically, EKF can be divided into two stages. The first stage is the one-step prediction based on the process model

$$ \hat{x}_{k+1|k} = f(\hat{x}_k) $$

$$ P_{k+1|k} = F_{k+1|k}P_{k|k}F_{k+1|k}^T + Q_k $$

(3) (4)

where $\hat{x}_{k|k}$ is the state estimation and $P_{k|k}$ is the covariance matrix, $\hat{x}_{k+1|k}$ and $P_{k+1|k}$ are the corresponding predictions. The state transition matrix $F_{k+1|k}$ is defined as the first-order Taylor expansion of the state transition function.

$$ F_{k+1|k} = \frac{\partial f(x)}{\partial x} \bigg|_{x=\hat{x}_k} $$

(5)

The second stage is the one-step update based on the measurement. The state vector and its covariance matrix are updated as

$$ K_{k+1} = P_{k+1|k}H_{k+1}^T \left( H_{k+1}P_{k+1|k}H_{k+1}^T + R_{k+1} \right)^{-1} $$

$$ \hat{x}_{k+1} = \hat{x}_{k+1|k} + K_{k+1} \left[ z_{k+1} - h(\hat{x}_{k+1|k}) \right] $$

$$ P_{k+1|k+1} = (I - K_{k+1}H_{k+1})P_{k+1|k} $$

where $K_{k+1}$ is the gain matrix, and $H_{k+1}$ is defined as

$$ H_{k+1} = \frac{\partial h(x)}{\partial x} \bigg|_{x=\hat{x}_{k+1|k}} $$

(6) (7) (8) (9)

As the main prior information of the system, the process noise $Q$ and the measurement noise $R$ are of great significance to the estimation performance and stability of the filter. $Q$ and $R$ represent the confidence degree in models and measurements, respectively. If $Q$ and $R$ are smaller than the true noise distribution, the uncertainty range of the true state is too narrow, resulting in estimation bias. Conversely, if $Q$ and $R$ are larger than the true noise distribution, it may lead to filter divergence. Additionally, inaccurate $Q$ and $R$ can impair the estimation accuracy.

3. Maneuver Detection and the Framework
3.1. Maneuver Detection Based on D-S Evidence Theory

The key to maneuver detection is to design an appropriate detection strategy to detect the target’s maneuver precisely and timely. As stated in the introduction, the detection depending on the innovation from single-source information is not reliable. To improve the accuracy of the maneuver detection, the maneuver detection with a sliding-window structure based on D-S evidence theory is proposed in this section by introducing multiple synchronous sensors’ information.

D-S evidence theory is a fusion rule established on a nonempty finite space, which includes a limited number of subsets [43]. Through independent observations by sensors, it can fuse observation results and give a joint judgment to improve the confidence and accuracy of events. D-S evidence theory can combine the evidence more intuitively and easily, and it has the expression ability of unknown and uncertain situations by calculating the probability of the set of multiple events.
Consider all the subsets of the finite space $\Theta$ as $2^\Theta$, including $\Theta$ and an empty set $\Phi$, and define the map $m:2^\Theta \to [0,1]$, for $A \subset \Theta$, it satisfies

\[
\sum_{A \subset \Theta} m(A) = 1
\]

where the map $m(\cdot)$ is called the basic probability assignment function. $m(A)$ is called the mass function of $A$, which indicates the degree of confidence for $A$ according to the current observation.

For the problem of maneuvering target tracking, the target has two movement modes: maneuver or not, so there are two events in the maneuver detection problem: “Maneuver” and “Normal”, denoted as $A_1$, $A_2$, respectively. However, when the degree of confidence tends to zero, unreasonable fusion results will be obtained, which is always called as Zadeh paradox [43]. To avoid the Zadeh paradox, one more event “uncertainty” is added, denoted as $A_3 = \{A_1, A_2\}$, and its confidence degree is set to 0.01.

Suppose there are $N$ sensors applied in the target’s measurement mission, denoted as $S_i (i = 1, 2, \ldots, N)$, and the measurement of $S_i$ at time $k$ is $z_{i,k}$. The innovation is defined as the difference between actual measurement value and predictive measurement value.

\[
\gamma_{i,k+1} = z_{i,k+1} - z_{i,k+1|k}
\]

where $z_{i,k+1|k} = \hat{h}_i(x_{i,k+1|k})$ is the predictive measurement. The innovation satisfies

\[
E[\gamma_{i,k+1}] = 0
\]

\[
E[\gamma_{i,k+1}\gamma_{i,k+1}^T] = H_{i,k+1}P_{i,k+1|k}H_{i,k+1}^T + R_{i,k+1}
\]

Following this, the detection variable is constructed based on the innovation.

\[
\eta_{i,k+1} = \gamma_{i,k+1}^T \left( H_{i,k+1}P_{i,k+1|k}H_{i,k+1}^T + R_{i,k+1} \right)^{-1} \gamma_{i,k+1}
\]

Since the innovation is subject to Gaussian distribution, the detection variable is supposed to be subject to $\chi^2$ distribution with $n$ degree of freedom, where $n$ is the dimension of the innovation vector. Therefore, the maneuver probability is defined as

\[
P_{i,k+1}(\text{Maneuver}) = \int_0^{h_{i,k+1}} \chi^2_n(x)dx
\]

It can be seen from Equation (16) that, the innovation is considered to be entirely caused by the unknown maneuver, hypothetically. The increase in the innovation indicates bigger maneuver probability.

Based on the definition of the maneuver probability, the mass function of $S_i$ is given as

\[
m_{i,k+1}(A_1) = P_{i,k+1}(\text{Maneuver})
\]

\[
m_{i,k+1}(A_2) = 1 - P_{i,k+1}(\text{Maneuver}) - 0.01
\]

\[
m_{i,k+1}(A_3) = 0.01
\]

As mentioned above, the mismatch between the real measurement noise and a priori measurement noise can also lead to the innovation’s increase. As a result, the maneuver probability determined by the innovation is unreliable. A joint detection mechanism is developed by fusing maneuver detection results of multiple synchronous sensors to improve the accuracy and reliability of the maneuver detection. For $N$ independent pieces
of evidence with \( N_e \) events, the fused mass function of \( A_1 \) can be merged by D-S evidence theory as follows

\[
m_{k+1}(A_1) = \begin{cases} 
\sum_{A_1 \cap \cdots \cap A_{N_e} = A_1} m_{k+1}(A_1) m_{k+1}(A_2) \cdots m_{k+1}(A_{N_e}) & k + 1 \geq N_{sw} \\
\sum_{A_1 \cap \cdots \cap A_{N_e} \neq \Phi} \left( \sum_{A_1 \cap \cdots \cap A_{N_e} = A_1} m_{k+1}(A_1) m_{k+1}(A_2) \cdots m_{k+1}(A_{N_e}) \right) m_{k+1}(A_1) & k + 1 < N_{sw}
\end{cases}
\]

where \( N_e = 3 \) in this paper.

Similarly, the fused mass function of \( A_2 \) and \( A_3 \) can be acquired, denoted as \( m_{k+1}^f(A_2) \) and \( m_{k+1}^f(A_3) \), respectively. If \( m_{k+1}^f(A_1) \geq m_{k+1}^f(A_2) \), it means that the target is detected to be maneuvering according to the fusion of multiple sensors at time \( k \). Conversely, the target is not maneuvering when \( m_{k+1}^f(A_1) < m_{k+1}^f(A_2) \), and the mismatch of the a priori measurement noise is the main reason for the increased innovation.

However, the maneuver detection based on the current time is unstable. Furthermore, the sliding window structure is introduced into the maneuver detection to improve the stability of detection, where the detection depends not only on the current state but also on the neighboring states.

The size of the sliding window is set as \( N_{sw} \), and the final mass function \( m_{k+1}^f(\cdot) \) can be calculated by

\[
m_{k+1}^f(A_1) = \begin{cases} 
\sum_{A_1 \cap \cdots \cap A_{N_e} = A_1} m_{k+1\cdots N_{sw}}(A_1) \cdots m_k(A_{N_e}) m_{k+1}(A_{N_e}) & k + 1 \geq N_{sw} \\
\sum_{A_1 \cap \cdots \cap A_{N_e} \neq \Phi} \left( \sum_{A_1 \cap \cdots \cap A_{N_e} = A_1} m_{k+1\cdots N_{sw}}(A_1) \cdots m_k(A_{N_e}) m_{k+1}(A_{N_e}) \right) m_{k+1}(A_1) & k + 1 < N_{sw}
\end{cases}
\]

Similarly, the final mass function \( m_{k+1}^f(A_2) \) and \( m_{k+1}^f(A_3) \) can be obtained. According to the fused detection result, there are three conditions of the tracking problem. (a) When \( m_{k+1}^f(A_1) \geq m_{k+1}^f(A_2) \), the target is detected to be maneuvering, and the adaptive process noise estimation needs to be introduced to deal with unknown maneuvers. (b) If \( m_{k+1}^f(A_1) < m_{k+1}^f(A_2) \) and \( m_{k+1}(A_1) \geq m_{k+1}(A_2) \), the target is not maneuvering, and the mismatch of measurement noise emerges, so the measurement noise adaption of \( S_j \) is necessary. (c) There are no above two conditions when \( m_{k+1}^f(A_1) < m_{k+1}^f(A_2) \) and \( m_{k+1}(A_1) < m_{k+1}(A_2) \), so EKF can cope with the target tracking task.

### 3.2. The Framework of the Noise-Adaption EKF

Based on the above maneuver detection, a noise-adaption EKF is proposed to adaptively cope with the unknown maneuver and inaccurate measurement noise. The framework of the proposed filter is shown in Figure 1.

**Figure 1.** The framework of the noise-adaption EKF.
To achieve the accurate compensation for unknown maneuvers and the inaccurate measurement noise covariance, multi-sensor information undergoes the maneuver detection. Whether or not the process noise adaption or the measurement noise adaption proceeds is determined by the maneuver detection. If the target is detected to be maneuvering, the adaptive process noise is calculated by DDPG and introduced into the filter to cope with the model mismatch caused by the unknown maneuver. If the inaccurate measurement noise covariance is detected to be the main reason for the innovation’s increase, the recursive measurement noise estimation is applied to the corresponding sensor. Sequentially, the local estimations are fused to obtain the global estimation of the target. When the unknown maneuver or mismatched measurement noise emerge, the proposed filter can estimate the maneuvering target’s state with high performance.

4. Noise-Adaption EKF for Maneuvering Targets
4.1. Process Noise Adaption Based on DDPG

For the problem of maneuvering target tracking, the main task is to find a tracking strategy to estimate the target’s station rapidly and accurately. However, the accurate mathematical models of non-cooperative targets cannot be obtained. When the target is maneuvering, the estimation error increases due to the model mismatch. To cope with the unknown maneuver, an effective method is to adaptively estimate the process noise covariance online. To this end, a process noise adaption method based on DDPG is proposed in this section. Through its self-learning and optimization capabilities, DDPG algorithm can adaptively cope with the unknown maneuver.

DDPG is known as one of the reinforcement learning algorithms for the continuous action strategy. The interactive learning process of DDPG is formalized as the Markov decision process (MDP), which can be defined by four elements: the state $S = \{s_1, s_2, \ldots\}$, the action $A = \{a_1, a_2, \ldots\}$, the reward $R = \{r_1, r_2, \ldots\}$, and the state transition probability $P(s_{k+1}|s_k, a_k)$. The framework of DDPG is shown in Figure 2. The agent generates an action to interact with the environment at first, and then under the joint effects of action and environment, a new state is generated, and the environment gives a reward to the critic. Afterwards, the critic calculates the action-value function to evaluate the given policy, then the action policy is further optimized according to the evaluation results.

The adaptive estimation of the process noise covariance is defined as

$$\hat{Q}_k = \lambda_k \hat{Q}_{k-1}$$  \hspace{1cm} (22)

where $\hat{Q}_k$ is the adaptive process noise covariance, and its initial value is the a priori process noise covariance matrix. $\lambda_k > 1$ is the compensation factor. This adaptive estimation form is more robust due to fewer parameters.

Correspondingly, the prediction of the covariance matrix in Equation (4) is updated as

$$P_{k+1|k} = F_{k+1|k} P_{k|k} F_{k+1}^T + \hat{Q}_k$$  \hspace{1cm} (23)
Following by this, the problem of designing the maneuvering tracking strategy is transformed into the determination of the compensation factor, which can be described as an MDP. DPPG algorithm is designed to learn the compensation factor, so the action is defined as

\[ a_k = \lambda_k \]  

(24)

The innovation reflects the degree of model mismatch. The larger the innovation, the larger the model error, and the compensation factor needs to be increased accordingly. Therefore, the agent state is defined as the filter’s innovation in Equation (12), which contains three-dimensional innovation: range, azimuth angle and elevation angle, i.e.,

\[ s_k = \gamma_k^T = [\gamma_{k,r}, \gamma_{k,\beta}, \gamma_{k,\epsilon}]^T \]  

(25)

When the environment, i.e., the filter, receives an action, a sufficient reward mechanism is required to evaluate the filter performance under the current action. Therefore, the filter residual is taken as the evaluation index, which is expressed as

\[ \varphi_k = z_k - \hat{z}_k = z_k - h(\hat{x}_k) \]  

(26)

In practice, the residuals of range measurements are usually much bigger than the residuals of azimuth angle and elevation angle measurements in practice, which leads to different sensitivity to different residual components. To make the filter equally sensitive to residual elements, a weighting matrix \( W \) is introduced for normalizing the residuals.

\[
W = \begin{bmatrix}
\frac{1}{\sigma_r} & 0 & 0 \\
0 & \frac{1}{\sigma_\beta} & 0 \\
0 & 0 & \frac{1}{\sigma_\epsilon}
\end{bmatrix}
\]  

(27)

where \( \sigma_r, \sigma_\beta, \) and \( \sigma_\epsilon \) represents the measurement errors’ standard deviation of range, azimuth angle, and elevation angle, respectively.

Therefore, the reward function is established as follows

\[ r_k = tr[(W\varphi_k)(W\varphi_k)^T] \]  

(28)

where \( tr[\cdot] \) is the trace operation.

In DDPG algorithm, two neural networks including the actor network and the critic network, are utilized to approximate the action function and the action-value function. Denote \( A^\mu(s_k) \) as the action function, which is parameterized by \( \mu \). It maps from state to action, i.e., \( a_k = A^\mu(s_k) \). Moreover, the action-value function is denoted as \( Q^\omega(s_k, a_k) \), which is parameterized by \( \omega \). Additionally, to enhance the convergence of training, one target actor network and one target critic network are introduced into DDPG, denoted as \( A^\mu'(s_k) \) and \( Q^{\omega'}(s_k, a_k) \), respectively.

The optimal action policy is realized by maximizing the expected total reward

\[ \mu = \arg\max_{\mu} J(A^\mu) \]  

(29)

The expected total reward is defined as

\[ J(A^\mu) = E\left[ Q^{\omega'}(s,a)\big|_{s_k,a_k = A^\mu(s_k)} \right] \]  

(30)

Therefore, the actor function is optimized toward the gradient of the expected total reward

\[ \nabla_\mu J(A^\mu) = \nabla_\mu A^\mu(s_k) \nabla_{a_k} Q^{\omega'}(s_k, a_k) \]  

(31)
Similarly, the action-value function is optimized by minimizing the loss function
\[ \omega = \arg\min_{\omega} L(\omega) \]  

(32)

The temporal difference error \( \delta_k \) is utilized to establish the loss function
\[ L(\omega) = \delta_k^2 = [r_k + \xi Q^\omega(s_{k+1}, A^\mu(s_{k+1})) - Q^\omega(s_k, A^\mu(s_k))]^2 \]  

(33)

where \( r_k \) is the reward, and \( \xi \) is the discounting factor.

The actor-value function is optimized toward the gradient of the loss function
\[ \nabla_{\omega} L(\omega) = -2\delta_k \nabla_{\omega} Q^\omega(s_k, a_k) \]  

(34)

To improve the stability in the DDPG training process, there is a replay memory buffer \( R \) to store the training samples. At every training step, training samples with a size \( N_{\text{sample}} \) are utilized to train the actor and critic networks. The current transition experience \( (s_k, a_k, r_k, s_{k+1}) \) will be added into the buffer to replace the oldest one.

The actor network is updated by the Adam optimizer
\[ \mu_{k+1} = f_{\text{Adam}}(\mu_k, \nabla_{\mu} J(A^\mu)) \]  

(35)

\[ \nabla_{\mu} J(A^\mu) = \frac{1}{N_{\text{sample}}} \sum_{i=1}^{N_{\text{sample}}} \nabla_{\mu} A^\mu(s_i) \nabla a_i Q^\omega(s_i, a_i) \]  

(36)

Similarly, the critic network is updated using the negative gradient of the loss function
\[ \omega_{k+1} = f_{\text{Adam}}(\omega_k, \nabla_{\omega} L(\omega)) \]  

(37)

\[ \nabla_{\omega} L(\omega) = \frac{1}{N_{\text{sample}}} \sum_{i=1}^{N_{\text{sample}}} \delta_i \nabla_{\omega} Q^\omega(s_i, a_i) \]  

(38)

At the end of each sample training, the two target networks are soft updated as
\[ \mu_{k+1}' = \tau \mu_{k+1} + (1 - \tau) \mu_k \]  

(39)

\[ \omega_{k+1}' = \tau \omega_{k+1} + (1 - \tau) \omega_k \]  

(40)

where \( \tau \) is the updating rate.

The implementation framework of the adaptive estimation of process noise based on DDPG is shown in Figure 3.

---

Figure 3. The implementation framework of the process noise adaption based on DDPG.
The process noise adaption based on DDPG is summarized in Algorithm 1.

**Algorithm 1** The process noise adaption based on DDPG

1. Initialize the parameters of the actor network and critic network
2. Initialize target networks by copying the actor and critic network
3. Initialize the replay memory buffer $R$

**For each episode, perform the following steps**

4. Initialize the estimation state and its covariance matrix

**For each timestep, perform the following steps**

5. Generate an action based on the actor network and the current state
   
   \[
   a_k = A\mu(s_k) + N_k, \quad \text{where the random noise is generated by Ornstein-Uhlenbeck process}
   \]

6. Execute the action, i.e., the compensation factor in the filter to obtain a new state $s_{k+1}$ and a new reward $r_k$

7. Store the sample $(s_k, a_k, s_{k+1}, r_k)$ in the buffer $R$

8. Randomly select $N_{\text{Sample}}$ samples from the buffer

9. Calculate the temporal difference error $\delta_p$
   
   \[
   \delta_p = r_p + \xi Q^\omega(s_{p+1}, A^\mu(s_{p+1})) - Q^\omega(s_p, A^\mu(s_p))
   \]

10. Calculate the policy gradient

11. Update the actor network by Adam optimizer:
   
   \[
   \mu_{k+1} = f_{\text{Adam}}(\nabla \mu J(A^\mu))
   \]

12. Update the critic network

13. Update the two target networks by soft update

End timestep

End episode

4.2. Recursive Measurement Noise Estimation

When the sensor’s maneuver detection result is inconsistent with the fused maneuver detection result, the corresponding sensor’s measurement model is considered not to match a priori knowledge. In this case, the adaptive estimation method based on DDPG is no longer applicable to deal with it. Since the target is non-cooperative, the deviation between the process model and the actual model exists, so it is difficult to realize the optimal estimation of measurement noise in a similar way to the adaptive estimation of process noise. To modify the measurement noise adaptively, the recursive estimation of measurement noise is introduced [34].

The estimation of $\hat{R}_{k+1}$ can be obtained by maximizing the a posteriori density function, which is given by

\[
\hat{R}_{k+1} = \frac{1}{k+1} \sum_{j=1}^{k+1} \left( z_j - h(\hat{x}_j) \right) \left( z_j - h(\hat{x}_j) \right)^T
\]

The residual can be represented as

\[
\begin{align*}
\hat{z}_j - h(\hat{x}_j) & = z_j - h(\hat{x}_{ji-1} + K_j \gamma_j) \\
& = z_j - h(\hat{x}_{ji-1}) - H_j K_j \gamma_j + o(K_j \gamma_j) \\
& \approx (I - H_j K_j) \gamma_j
\end{align*}
\]
Substitute Equation (42) into Equation (41), yields

\[ R_{k+1} = \frac{1}{k+1} \sum_{j=1}^{k+1} \left( (I - H_j K_j) \gamma_j \gamma_j^T (I - H_j K_j)^T \right) \]  

(43)

The expectation of the measurement noise can be expressed as (the detailed derivation can be found in the appendix of [34].)

\[ E \left[ R_{k+1} \right] = E \left\{ \frac{1}{k+1} \sum_{j=1}^{k+1} \left( (I - H_j K_j) \gamma_j \gamma_j^T (I - H_j K_j)^T \right) \right\} = R - \frac{1}{k+1} \sum_{j=1}^{k+1} \left( (I - H_j K_j) H_j P_{j|j-1} H_j^T \right) \]  

(44)

The statistical residual covariance is calculated as the variance of historical residual sequence

\[ R_{k+1} = (I - H_{k+1} K_{k+1}) \left( \gamma_{k+1} \gamma_{k+1}^T (I - H_{k+1} K_{k+1})^T + H_{k+1} P_{k+1|k} H_{k+1}^T \right) \]  

(45)

Accordingly, to approximate the real measurement covariance, the modified measurement covariance is updated as the following recursive form

\[ \hat{R}_{k+1} = (1 - \rho_{k+1}) R_k + \rho_{k+1} \hat{R}_{k+1} \]  

(46)

\[ \rho_{k+1} = \frac{1 - b}{1 - b^{k+1}} \]  

(47)

where \( \rho_{k+1} \) is the weighting factor, and \( b \) is a fading factor. The initial value of \( \hat{R}_k \) is the a priori measurement noise covariance matrix.

Therefore, the gain matrix in Equation (6) is replaced as

\[ K_{k+1} = P_{k+1|k} H_{k+1}^T (H_{k+1} P_{k+1|k} H_{k+1}^T + \hat{R}_{k+1})^{-1} \]  

(48)

By modifying the measurement covariance, the gain matrix is adjusted, then the impact of inaccurate a priori measurement noise is reduced.

4.3. Fusion Algorithm

The local estimations from multiple sensors are fused in this section. Since the cross-covariance matrices of different sensors are unknown, the fault-tolerant generalized convex combination algorithm (FGCC) [44] based on information theory is introduced in this section.

For two local estimations \( \hat{x}_1 \) and \( \hat{x}_2 \), the fusion rule of FGCC is

\[ \hat{x}_f = P_f \left( v_1 P_1^{-1} \hat{x}_1 + v_2 P_2^{-1} \hat{x}_2 \right) \]  

(49)

\[ P_f = (v_1 P_1^{-1} + v_2 P_2^{-1})^{-1} \]  

(50)

\[ v_1 + v_2 = \delta \]  

(51)

where \( P_1 \) and \( P_2 \) are corresponding covariance matrices, \( v_1 \) and \( v_2 \) are the weighting parameters, and the \( \delta \) is an adaptive parameter constructed as
\[ \delta = \frac{\mathcal{H}(P_1) + \mathcal{H}(P_2)}{\mathcal{H}(P_1) + \mathcal{H}(P_2) + \mathcal{I}(P_1, P_2)} \]  

(52)

In the above equation, \( \mathcal{H} \) is the Shannon entropy, and \( \mathcal{I} \) is the symmetrized Kullback-Leibler distance between two distributions known as J-divergence.

\[ \mathcal{H}(P_i) = \frac{1}{2} \log[(2\pi)^n|P_i|] + \frac{n}{2} \]  

(53)

\[ \mathcal{I}(P_i, P_j) = \mathcal{D}(P_i, P_j) + \mathcal{D}(P_j, P_i) \]  

(54)

where \( n \) is the number of state vector dimensions, and \( \mathcal{D} \) is the Kullback-Leibler divergence.

\[ \mathcal{D}(P_i, P_j) = \frac{1}{2} \left[ \ln \frac{|P_j|}{|P_i|} + ||dx_{ij}||_{P_i^{-1}} + \text{tr} \left( P_iP_j^{-1} \right) - n \right] \]  

(55)

where \( d_{x_{ij}} = \hat{x}_i - \hat{x}_j \), and \( i, j \in \{1, 2\}, i \neq j \).

The weights can be determined by the following equation

\[ v_i = \frac{\delta \mathcal{D}(P_i, P_f)}{\mathcal{D}(P_i, P_f) + \mathcal{D}(P_f, P_i)} \]  

(56)

Obviously, the above algorithm can only be applied to the fusion of two local estimations. In order to improve the applicability of the fusion algorithm, an extended fault-tolerant generalized convex combination algorithm (EFGCC) is developed, which provides the general fusion rule for \( N > 2 \).

Suppose there are \( N \) local estimations \( \hat{x}_i (i = 1, \ldots, N) \) and corresponding covariance matrices \( P_i \). The fusion form of EFGCC is given as

\[ \hat{x}_f = P_f \sum_{i=1}^{N} v_i P_i^{-1} \hat{x}_i \]  

(57)

\[ P_f = \left( \sum_{i=1}^{N} v_i P_i^{-1} \right)^{-1} \]  

(58)

\[ \sum_{i=1}^{N} v_i = \delta \]  

(59)

where the adaptive parameter \( \delta \) is updated as

\[ \delta = \frac{\sum_{i=1}^{N} \mathcal{H}(P_i)}{\sum_{i=1}^{N} \mathcal{H}(P_i) + \sum_{i=1}^{N} \sum_{j=i+1}^{N} \mathcal{I}(P_i, P_j)} \]  

(60)

The definitions of the Shannon entropy and the symmetrized Kullback-Leibler distance are the same as above. The weighting parameter \( v_i \) can be determined by the following equation

\[ v_i = \frac{\delta f_i(D)}{\sum_{j=1}^{N} f_j(D)} \]  

(61)

where
white noise. The acceleration is assumed to be a first-order time correlation. The model can be given by

\[ x_{k+1} = F_{k+1|x_k}x_k + w_k \] (63)

The state vector \( x_k = [x_k, v_{x,k}, a_{x,k}, y_k, v_{y,k}, a_{y,k}, z_k, v_{z,k}, a_{z,k}]^T \), which includes the position, velocity, and acceleration in three directions.

The state transition matrix is defined as

\[ F_{k+1|x_k} = \text{diag}(F_x, F_y, F_z) \] (64)

where

\[ F_x = F_y = F_z = \begin{bmatrix} 1 & T & (e^{-\alpha T} + \alpha T - 1) / \alpha^2 \\ 0 & 1 & (1 - e^{-\alpha T}) / \alpha \\ 0 & 0 & e^{-\alpha T} \end{bmatrix} \] (65)

\( \alpha \) is the target’s maneuvering frequency, and \( T \) is the sampling period.

The covariance matrix is calculated as

\[ Q_k = \text{diag}(Q_x, Q_y, Q_z) \] (66)

where

\[ Q_x = Q_y = Q_z = 2\alpha \sigma_a^2 [q_{ij}]_{3 \times 3} \] (67)

\[
\begin{align*}
q_{11} &= \frac{1}{2\alpha^3} \left[ 1 - e^{-2\alpha T} + 2\alpha T + 2\alpha^3 T^3 / 3 - 2\alpha^2 T^2 - 4\alpha T e^{-\alpha T} \right] \\
q_{12} &= q_{21} = \frac{1}{2\alpha^3} \left[ e^{-2\alpha T} + 1 - 2e^{-\alpha T} + 2\alpha T e^{-\alpha T} - 2\alpha T + \alpha^2 T^2 \right] \\
q_{13} &= q_{31} = \frac{1}{2\alpha^3} \left[ 1 - e^{-2\alpha T} - 2\alpha T e^{-\alpha T} \right] \\
q_{22} &= \frac{1}{2\alpha^3} \left[ 4e^{-\alpha T} - 3 - e^{-2\alpha T} + 2\alpha T \right] \\
q_{23} &= q_{32} = \frac{1}{2\alpha^3} \left[ e^{-2\alpha T} + 1 - 2e^{-\alpha T} \right] \\
q_{33} &= \frac{1}{2\alpha^3} \left[ 1 - e^{-2\alpha T} \right]
\end{align*}
\] (68)

The measurement vector is defined as \( z_k = [r_k, \beta_k, \epsilon_k]^T \), where \( r_k \) is the range, \( \beta_k \) is the azimuth angle, and \( \epsilon_k \) is the elevation angle. The measurement mapping function is

\[ h(x_{k+1}) = \begin{bmatrix} \sqrt{x_{k+1}^2 + y_{k+1}^2 + z_{k+1}^2} \\
\arctan(z_{k+1} / x_{k+1})/\sqrt{x_{k+1}^2 + z_{k+1}^2} \\
\arctan(y_{k+1} / \sqrt{x_{k+1}^2 + z_{k+1}^2}) \end{bmatrix} \] (69)

By the fusion rule of Equation (57), the global estimation for maneuvering targets is realized.

5. Simulation

In this section, the effectiveness and superiority of the proposed tracking method are demonstrated through numerical simulations.

5.1. Target Model

In the simulation experiments, the Singer model [7] is adopted as the process model in this paper, which describes the target acceleration as the colored noise rather than the white noise. The acceleration is assumed to be a first-order time correlation. The model can be given by
The initial covariance matrix of the measurement noise is

\[
R = \begin{bmatrix}
    \sigma_r^2 & 0 & 0 \\
    0 & \sigma_p^2 & 0 \\
    0 & 0 & \sigma_e^2
\end{bmatrix}
\]  

(70)

5.2. Construction of Neural Networks

In the proposed process noise estimation based on DDPG, the actor network is used to learn the action policy from the input of the innovation \( \gamma_k, r_k, \beta_k, \epsilon_k \) to the output of the compensation factor \( \lambda_k \), and the critic network is used to obtain the action-value function based on the innovation and the compensation factor. Inspired by \([45,46]\), four-layer fully connected networks are commonly utilized in deep reinforcement learning applications. Hence the structure of the actor network is designed as shown in Figure 4, and the critic network is represented by the same network structure. Too few neurons will lead to underfitting. Conversely, too many neurons can lead to overfitting and increased training time. By referring to \([42,45,46]\) and simulation experiments, the size of neural networks are set as in Table 1.

![Figure 4. The structure of actor network.](image)

Table 1. Neural network settings.

| Layers         | Actor Network | Critic Network |
|----------------|---------------|----------------|
| Input layer    | 3             | 4              |
| Hidden layer 1 | 128           | 128            |
| Hidden layer 2 | 128           | 128            |
| Output layer   | 1             | 1              |

The activation function is tan h function, which is given by

\[
\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}
\]  

(71)

Simulation parameters are summarized in Table 2.

Table 2. Simulation parameters of DDPG.

| Parameters            | Value   |
|-----------------------|---------|
| The discounting factor \( \bar{\epsilon} \) | 0.99    |
| The updating rate \( \tau \)                | 0.001   |
| The number of samples \( N_{\text{sample}} \) | 64      |
| The number of episodes | 500     |
| The number of timesteps | 1000   |

The simulation environment is defined as 6000 km × 100 km × 6000 km, in which the target tracking task is carried out. The training task includes 500 random episodes, and
each episode runs 1000 timesteps. The maximum acceleration of the target is 30 m/s², and the minimum acceleration is −30 m/s². For each episode, the target randomly maneuvers with values uniformly distributed between the minimum and the maximum accelerations. According to the definition of the training environment and the maneuver space, training scenarios are constructed.

5.3. Simulation Results

To demonstrate the superiority of the proposed filter, IMM [9], STF [22], RNSTF [33], and AKF [34] are simulated for comparison. Two simulation scenarios are conducted in this section to illustrate the feasibility of the proposed noise-adaption EKF. In the first scenario, the target makes continuous time-varying maneuvers. In the second scenario, the target is set to make abrupt maneuvers to verify the performance against strong maneuvers.

In the simulation, the fading factor in the recursive measurement noise estimation and AKF $b = 0.9$, and the maneuvering frequency in the Singer model $\alpha = 1/20$. The forgetting factor in STF and RNSTF is 0.95. CA model, Singer model, and CS model are adopted in IMM, and the maneuvering frequency in the CS model $\alpha = 1/20$. The initial probability of each sub model is $[0.4 \ 0.4 \ 0.2]$, and the Markov transition probability matrix is set as

$$P = \begin{bmatrix} 0.95 & 0.025 & 0.025 \\ 0.025 & 0.95 & 0.025 \\ 0.025 & 0.025 & 0.95 \end{bmatrix}$$  \hspace{1cm} (72)

The root mean square error (RMSE) is generally used to estimate the tracking accuracy, given as

$$RMSE = \sqrt{\frac{\sum_{j=1}^{N_{MC}} (\hat{x}_k^j - x_k^j)^2}{N_{MC}}}$$  \hspace{1cm} (73)

where $\hat{x}_k^j$ is the estimation value, and $x_k$ is the real value. $N_{MC}$ is the number of Monte Carlo experiments, which is 100 in the following simulations.

5.3.1. Continuous Time Varying Maneuver

In this scenario, the target’s acceleration is continuous time-varying in sinusoidal form, which can be described as

$$a = A \sin(\omega t + \varphi_0)$$  \hspace{1cm} (74)

where $A$ is the amplitude, $\omega$ is the frequency, and $\varphi_0$ is the initial phase. The entire flight process lasts 1000 s. The initial position is [800 km, 80 km, 800 km], and the initial velocity is [−300 m/s, 10 m/s, −300 m/s]. The sinusoidal parameters are set as shown in Table 3.

| Directions | $x$ | $y$ | $z$ |
|------------|----|----|----|
| $A$        | 1  | 2  | 1  |
| $\omega$   | 0.005 | 0.01 | 0.005 |
| $\varphi_0$ | 30 | 30 | 30 |

Assuming that there are three sensors at the origin to track the maneuvering target with a sampling period of 1 s, and a priori measurement errors are set as shown in Table 4. Furthermore, to verify the performance of the proposed noise-adaption EKF under the condition of inaccurate measurement noise, the magnitude of the range measurement noise of $S_1$ enlarged five times during 500–600 s, which is unknown to the filter.
Table 4. Measurement accuracy of multiple sensors.

| Sensors | \(\sigma_r\) (m) | \(\sigma_\beta\) (\(^\circ\)) | \(\sigma_\varepsilon\) (\(^\circ\)) |
|---------|-----------------|-----------------|-----------------|
| \(S_1\) | 10              | 0.01            | 0.01            |
| \(S_2\) | 5               | 0.02            | 0.02            |
| \(S_3\) | 10              | 0.02            | 0.02            |

The algorithms mentioned above are utilized for tracking solution, and the same fusion method, i.e., EFGCC, is adopted to obtain fusion trajectories. For convenience, the proposed noise-adaption EKF is denoted as NAEKF in the following simulation results. The estimated trajectories are shown in Figure 5.

Figure 5. Estimated trajectories.

The position estimation errors are presented in Figure 6. In order to correspond to the measurements, the position estimation errors are given in terms of range, azimuth angle and elevation angle. The position errors of NAEKF are smaller than that of other filters, especially in the stage of inaccurate measurement noise. Correspondingly, Figure 7 shows the \(\sigma\)-boundaries of position estimation errors. At the beginning of the simulation, position errors are generally large due to initialization errors, but the estimation accuracy gradually improves over time. After 100 s, the range error of the proposed filter converges to 3.27 m. The azimuth angle error reaches 0.0033\(^\circ\), and the elevation angle error is stable around 0.0029\(^\circ\). Due to the effect of inaccurate measurement noise between 500–600 s, the stable value of range errors reaches 3.54 m. Similarly, the azimuth angle error increases slightly and finally stabilizes at 0.0043\(^\circ\), and the elevation angle error stabilizes at 0.0043\(^\circ\). To compare local estimation accuracy of different algorithms, the detailed estimation errors are shown in Tables 5–7. Taking the range estimation of \(S_1\) as an example, the estimation accuracy of NAEKF is 8.95\%, 21.28\% and 20.90\% higher than that of IMM, STF and RNSTF, respectively, and the estimation error of AKF is almost seven times that of NAEKF. Obviously, the proposed filter has better tracking accuracy than other filters.
Figure 6. Position estimations errors. (a) Estimation errors of the range; (b) Estimation errors of the azimuth angle; (c) Estimation errors of the elevation angle.

Figure 7. $\sigma$-Boundary of position estimation errors. (a) $\sigma$-Boundary of the range; (b) $\sigma$-Boundary of the azimuth angle; (c) $\sigma$-Boundary of the elevation angle.

Table 5. Estimation errors of the range (m).

| Sensors | IMM   | STF   | RNSTF | AKF   | NAEKF |
|---------|-------|-------|-------|-------|-------|
| S₁      | 15.7454 | 17.5285 | 17.4727 | 96.1808 | 14.4525 |
| S₂      | 4.8372  | 3.9745 | 3.9816 | 3.8560 | 3.8291 |
| S₃      | 8.9003  | 8.1024 | 8.1208 | 7.604  | 7.3634 |
| Fusion  | 4.7536  | 8.1988 | 8.4076 | 129.2355 | 3.5371 |

Table 6. Estimation errors of the azimuth angle (°).

| Sensors | IMM   | STF   | RNSTF | AKF   | NAEKF |
|---------|-------|-------|-------|-------|-------|
| S₁      | 0.0073 | 0.0098 | 0.0098 | 0.0116 | 0.0057 |
| S₂      | 0.0110 | 0.0130 | 0.0133 | 0.0088 | 0.0090 |
| S₃      | 0.0101 | 0.0135 | 0.0139 | 0.0099 | 0.0087 |
| Fusion  | 0.0052 | 0.0076 | 0.0077 | 0.0108 | 0.0043 |
Table 7. Estimation errors of the elevation angle (°).

| Sensors | IMM  | STF  | RNSTF | AKF  | NAEKF |
|---------|------|------|-------|------|-------|
| $S_1$   | 0.0076 | 0.0100 | 0.0098 | 0.4293 | 0.0060 |
| $S_2$   | 0.0108 | 0.0140 | 0.0136 | 0.0097 | 0.0097 |
| $S_3$   | 0.0102 | 0.0145 | 0.0139 | 0.0328 | 0.0094 |
| Fusion  | 0.0051 | 0.0076 | 0.0074 | 0.1790 | 0.0043 |

Figures 8 and 9 show the velocity estimation errors and the corresponding $\sigma$-Boundary in three directions, respectively. As shown in simulation results, the velocity estimation accuracy of IMM, STF, RNSTF and AKF are worse than that of NAEKF. It can be seen from Figure 9 that the velocity estimation errors of STF and RNSTF increase significantly when inaccurate measurement noise occurs, while the result of IMM and NAEKF do not. Taking the velocity estimation in x direction as an example, the estimation error of STF increases from 24.85 m/s to 35.20 m/s, and that of RNSTF increases from 24.87 m/s to 35.27 m/s. The estimation error of IMM is stable around 18.20 m/s, while the estimation error of AKF reaches 98.65 m/s. In contrast, the velocity estimation error in x direction of NAEKF reaches 13.83 m/s. The tracking results in y and z directions are similar as that in x direction, which will not be repeated here. In the end, the velocity estimation accuracy of NAEKF reaches 14.21 m/s, which is the average of three directions.

Figure 8. Velocity estimation errors. (a) Estimation errors of $v_x$; (b) Estimation errors of $v_y$; (c) Estimation errors of $v_z$.

Figure 9. $\sigma$-Boundary of velocity estimation errors. (a) $\sigma$-Boundary of $v_x$; (b) $\sigma$-Boundary of $v_y$; (c) $\sigma$-Boundary of $v_z$. 
The acceleration estimation errors and the corresponding $\sigma$-boundaries of acceleration estimation errors are presented in Figures 10 and 11, separately. The average acceleration estimation accuracy of NAEKF is 1.27 m/s$^2$, whereas that of IMM, STF, RNSTF and AKF is around 1.86 m/s$^2$, 4.20 m/s$^2$, 4.11 m/s$^2$ and 28.01 m/s$^2$, respectively. It can be confirmed that the proposed NAEKF can effectively deal with unknown sinusoidal maneuvers and achieves better tracking performance than IMM, STF, RNSTF as well as AKF. Table 8 provides computing times of five algorithms. Since NAEKF adds steps such as maneuver detection, the computing time is slightly longer than that of STF and RNSTF. However, NAEKF takes less computing time than IMM and AKF, because the parallel calculation of sub filters in IMM and the simultaneous noise estimation in AKF are time-consuming.

![Figure 10](image1.png)

**Figure 10.** Acceleration estimation errors. (a) Estimation errors of $a_x$; (b) Estimation errors of $a_y$; (c) Estimation errors of $a_z$.

![Figure 11](image2.png)

**Figure 11.** $\sigma$-Boundary of acceleration estimation errors. (a) $\sigma$-Boundary of $a_x$; (b) $\sigma$-Boundary of $a_y$; (c) $\sigma$-Boundary of $a_z$.

|          | IMM   | STF   | RNSTF | AKF   | NAEKF |
|----------|-------|-------|-------|-------|-------|
| Computing time (s) | 1.9645 | 1.7343 | 1.8173 | 3.3055 | 1.8572 |

On the basis of the above experiment settings, the magnitude of the angle measurement noises of $S_1$ also enlarged five times during 500–600 s. Simulation results show that the matrix singularity problem of AKF occurs in this simulation experiment. As described in the introduction, AKF shows poor stability, because it updates the process noise and measurement noise simultaneously without distinction. Therefore, IMM, STF, RNSTF and NAEKF are presented in this experiment.
Figures 12 and 13 show the position estimation errors and the corresponding $\sigma$-Boundary. Although the inaccurate measurement noises lead to certain increases in estimation errors, NAEKF shows better estimation accuracy and robustness compared to other three algorithms. Taking the azimuth angle as an example, the estimation error of NAEKF is $0.0068^\circ$, while that of IMM reaches $0.0098^\circ$. Additionally, the estimation error of STF and RNSTF is $0.0159^\circ$ and $0.0158^\circ$, respectively, which is more than twice the estimation error of NAEKF.

![Figure 12](image1.png)

Figure 12. Position estimations errors. (a) Estimation errors of the range; (b) Estimation errors of the azimuth angle; (c) Estimation errors of the elevation angle.

![Figure 13](image2.png)

Figure 13. $\sigma$-Boundary of position estimation errors. (a) $\sigma$-Boundary of the range; (b) $\sigma$-Boundary of the azimuth angle; (c) $\sigma$-Boundary of the elevation angle.

The estimation results of the velocity and acceleration are shown in Figures 14–17. It is obvious that the proposed NAEKF can effectively deal with inaccurate measurement noises in this experiment and achieves the best estimation accuracy compared to other algorithms. The velocity estimation errors of IMM, STF and RNSTF are several times that of NAEKF, and the same conclusion can be drawn from the acceleration estimation results.
Figure 14. Velocity estimation errors. (a) Estimation errors of $v_x$; (b) Estimation errors of $v_y$; (c) Estimation errors of $v_z$.

Figure 15. $\sigma$-Boundary of velocity estimation errors. (a) $\sigma$-Boundary of $v_x$; (b) $\sigma$-Boundary of $v_y$; (c) $\sigma$-Boundary of $v_z$.

Figure 16. Acceleration estimation errors. (a) Estimation errors of $a_x$; (b) Estimation errors of $a_y$; (c) Estimation errors of $a_z$. 


5.3.2. Abrupt Maneuver

In this scenario, the target’s acceleration is set to be abrupt, and the maneuver parameters are shown in Table 9. The initial position is \([6000 \, \text{km}, 15 \, \text{km}, 6000 \, \text{km}]\), and the initial velocity is \([-1000 \, \text{m/s}, 0 \, \text{m/s}, -1000 \, \text{m/s}]\).

### Table 9. Maneuver parameters of the target.

| Time (s) | Acceleration (m/s²) | Time (s) | Acceleration (m/s²) |
|----------|----------------------|----------|----------------------|
|          | x                    | y        | z                    |                      |
| 0–60     | 0                    | 0        | 0                    | 480–540              |
| 60–120   | -22.5                | 18       | -22.5                | 540–600              |
| 120–180  | 2.5                  | -10      | 2.5                  | 600–660              |
| 180–240  | -15                  | -15      | -15                  | 660–720              |
| 240–300  | 4.5                  | -10      | 4.5                  | 720–780              |
| 300–360  | -20                  | 29       | -20                  | 780–840              |
| 360–420  | -10                  | -8       | -10                  | 840–900              |
| 420–480  | -12.5                | -10      | -12.5                | 900–1000             |

The measurement errors are the same as in Table 4, and the magnitude of the range measurement noise of S₁ enlarged five times during 700–800 s. It should be noted that the matrix singularity problem of AKF occurs in this simulation scenario. Therefore, IMM, STF, RNSTF and NAEKF are presented in this section. The tracking results are shown in Figure 18.

The position estimation errors and the corresponding \(\sigma\)-boundaries are shown in Figures 19 and 20, respectively. Simulation results show that the proposed filter can track the abrupt maneuver target effectively. Owing to the D-S maneuver detection, the proposed noise-adaption EKF can detect unknown maneuvers and inaccurate measurement noise simultaneously. As a result, the proposed filter can utilize the recursive measurement noise adaption to weaken its influence. Certain increases in estimation errors are caused by inaccurate measurement noise during 700–800 s. The estimation errors of STF and RNSTF increase remarkably, especially when the inaccurate measurement noise appears. The estimation errors of different algorithms are shown in Tables 10–12. It is evident that the proposed filter achieves higher estimation accuracy than IMM, STF and RNSTF in both local estimation and fusion estimation.
Figure 18. Estimated trajectories.

Figure 19. Position estimation errors. (a) Estimation errors of the range; (b) Estimation errors of the azimuth angle; (c) Estimation errors of the elevation angle.

Figure 20. $\sigma$-Boundary of position estimation errors. (a) $\sigma$-Boundary of the range; (b) $\sigma$-Boundary of the azimuth angle; (c) $\sigma$-Boundary of the elevation angle.
Table 10. Estimation errors of the range (m).

| Sensors | IMM   | STF   | RNSTF | NAEKF |
|---------|-------|-------|-------|-------|
| S₁      | 18.2491 | 18.0910 | 18.2003 | 17.4784 |
| S₂      | 5.0014  | 5.2004  | 5.2474  | 4.6219  |
| S₃      | 9.4525  | 9.2029  | 9.3168  | 8.6291  |
| Fusion  | 4.9532  | 11.8334 | 12.0310 | 4.8331  |

Table 11. Estimation errors of the azimuth angle (°).

| Sensors | IMM   | STF   | RNSTF | NAEKF |
|---------|-------|-------|-------|-------|
| S₁      | 0.0062 | 0.0098 | 0.0099 | 0.0064 |
| S₂      | 0.0110 | 0.0137 | 0.0143 | 0.0104 |
| S₃      | 0.0117 | 0.0133 | 0.0136 | 0.0096 |
| Fusion  | 0.0048 | 0.0079 | 0.0080 | 0.0047 |

Table 12. Estimation errors of the elevation angle (°).

| Sensors | IMM   | STF   | RNSTF | NAEKF |
|---------|-------|-------|-------|-------|
| S₁      | 0.0062 | 0.0093 | 0.0092 | 0.0057 |
| S₂      | 0.0102 | 0.0141 | 0.0137 | 0.0107 |
| S₃      | 0.0112 | 0.0134 | 0.0131 | 0.0098 |
| Fusion  | 0.0050 | 0.0076 | 0.0074 | 0.0045 |

Figures 21 and 22 present velocity estimation errors and corresponding σ-boundaries. Velocity estimation errors in x and z directions of NAEKF are 77.32 m/s and 77.86 m/s, respectively. However, the estimation error in y direction reaches 146.70 m/s, because the maneuver in y direction is much bigger than other directions. Estimation errors of STF and RNSTF are significantly bigger than that of NAEKF. In contrast, IMM achieves better estimation accuracy than STF and RNSTF, but still worse than NAEKF.

According to acceleration estimation results in Figure 23, there are several peaks of acceleration tracking errors during abrupt maneuvers, but they decrease rapidly. The σ-boundaries of acceleration estimation errors are shown in Figure 24. The acceleration estimation accuracy in x direction of NAEKF is 4.97 m/s², whereas that of IMM, STF and RNSTF is finally stabilized at 10.23 m/s², 12.31 m/s² and 12.49 m/s², respectively. Similarly, the acceleration estimation accuracy in z direction of NAEKF is 5.06 m/s², and the estimation accuracy of IMM, STF and RNSTF is around 10.38 m/s², 12.02 m/s² and 12.24 m/s², respectively. Due to the greater maneuver, the acceleration estimation accuracy
in y direction is worse than that in other directions. Specifically, the acceleration estimation error of NAEKF reaches 10.27 m/s². In contrast, the acceleration estimation error of IMM, STF and RNSTF is around 14.66 m/s², 18.93 m/s², and 18.32 m/s², respectively. Table 13 provides computing times of four algorithms. Similarly to the first scenario, the computing time of NAEKF is shorter than that of IMM, but slightly longer than that of STF and RNSTF.

**Figure 22.** $\sigma$-Boundary of velocity estimation errors. (a) $\sigma$-Boundary of $v_x$; (b) $\sigma$-Boundary of $v_y$; (c) $\sigma$-Boundary of $v_z$.

**Figure 23.** Acceleration estimation errors. (a) Estimation errors of $a_x$; (b) Estimation errors of $a_y$; (c) Estimation errors of $a_z$.

**Figure 24.** $\sigma$-Boundary of acceleration estimation errors. (a) $\sigma$-Boundary of $a_x$; (b) $\sigma$-Boundary of $a_y$; (c) $\sigma$-Boundary of $a_z$. 
Table 13. Computing time (s).

|        | IMM   | STF   | RNSTF | NAEKF |
|--------|-------|-------|-------|-------|
| Time (s)| 1.9920| 1.6958| 1.8096| 1.8723|

6. Conclusions

To address the problem of maneuvering target tracking with inaccurate measurements, the noise-adaption EKF based on DDPG is proposed in this paper. The proposed filter avoids the simultaneous adjustment of the process model and the measurement model without distinction, which effectively improves the robustness of the filter. Within the framework of the noise-adaption EKF, the maneuver detection based on D-S evidence theory is constructed to distinguish between the unknown maneuver and inaccurate measurement noise simultaneously by fusing multi-sensor information. Moreover, a Markovian decision process of maneuver tracking is established to cope with unknown maneuvers. DDPG is developed to learn the adaptive estimation of the compensation factor and feed it to EKF. In addition, the recursive measurement noise estimation is applied to estimate a priori measurement noise covariance online. The local estimations are fused at last, achieving the global estimation of multiple sensors. The simulation results indicate that the proposed noise-adaption EKF is effective in both scenarios of continuous time-varying maneuver and the abrupt maneuver. As shown in simulation results, the proposed tracking method has a better tracking performance compared to IMM, STF, RNSTF and AKF.

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

Funding: This research was funded by National Natural Science Foundation of China, grant number 11572036.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors give their sincere thanks to the editors and the anonymous reviewers for their constructive comments of the manuscript.

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

References
1. Yin, J.; Yang, Z.; Luo, Y. Adaptive tracking method for non-cooperative continuously thrusting spacecraft. Aerospace 2021, 8, 244. [CrossRef]
2. Wang, S.; Bi, D.; Ruan, H.; Du, M. Radar maneuvering target tracking algorithm based on human cognition mechanism. Chin. J. Aeronaut. 2019, 32, 1695–1704. [CrossRef]
3. Lim, J.; Kim, H.; Park, H. Minimax particle filtering for tracking a highly maneuvering target. Int. J. Robust Nonlinear Control 2020, 30, 636–651. [CrossRef]
4. Cheng, Y.; Yan, X.; Tang, S.; Wu, M.; Li, C. An adaptive non-zero mean damping model for trajectory tracking of hypersonic glide vehicles. Aerosp. Sci. Technol. 2021, 111, 106529. [CrossRef]
5. Liu, J.; Wang, Z.; Xu, M. DeepMTT: A deep learning maneuvering target-tracking algorithm based on bidirectional LSTM network. Inf. Fusion 2020, 53, 289–304. [CrossRef]
6. Zhu, W.; Xu, Z.; Li, B.; Wu, Z. Research on the observability of bearings-only target tracking based on multiple sonar sensors. In Proceedings of the 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control (IMCCC), Harbin, China, 8–10 December 2012.
7. Jia, S.; Zhang, Y.; Wang, G. Highly maneuvering target tracking using multi-parameter fusion Singer model. J. Syst. Eng. Electron. 2017, 28, 841–850. [CrossRef]
8. Jiang, B.; Sheng, W.; Zhang, R.; Han, Y.; Ma, X. Range tracking method based on adaptive “current” statistical model with velocity prediction. Signal Process. 2017, 131, 261–270. [CrossRef]
9. Blom, H.; Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Trans. Autom. Control* **1988**, *33*, 780–783. [CrossRef]

10. Lim, J.; Kim, H.; Park, H. Interactive-multiple-model algorithm based on minimax particle filtering. *IEEE Signal Process. Lett.* **2020**, *27*, 36–40. [CrossRef]

11. Wang, M.; Bai, Y. Multi-platform maneuvering target tracking based on BLUE-AIMM-CI algorithm. In Proceedings of the 34th Chinese Control Conference, Hangzhou, China, 28–30 July 2015.

12. Ding, Z.; Liu, Y.; Liu, J.; Yu, K.; You, Y.; Jing, P.; He, Y. Adaptive interacting multiple model algorithm based on information-weighted consensus for maneuvering target tracking. *Sensors* **2018**, *18*, 2012. [CrossRef]

13. Yu, M.; Chen, W.; Chambers, J. State dependent multiple model-based particle filtering for ballistic missile tracking in a low-observable environment. *Aerosp. Sci. Technol.* **2017**, *67*, 144–154. [CrossRef]

14. Li, X.; Jilkov, V.; Ru, J. Multiple-model estimation with variable structure part VI: Expected-mode augmentation. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 853–867.

15. Wang, J.; Dong, P.; Jing, Z.; Cheng, J. Consensus variable structure multiple model filtering for distributed maneuvering target tracking in sensor networks. *Trans. Inst. Meas. Control* **2020**, *42*, 3052–3062. [CrossRef]

16. Yu, M.; Oh, H.; Chen, W. An improved multiple model particle filtering approach for maneuvering target tracking using airborne GMTO with geographic information. *Aerosp. Sci. Technol.* **2016**, *52*, 62–69. [CrossRef]

17. Gelb, A. *Applied Optimal Estimation*; MIT Press: Cambridge MA, USA, 1974.

18. Kim, J.; Lee, K. Unscented Kalman filter-aided long short-term memory approach for wind nowcasting. *Aerospace* **2021**, *8*, 236. [CrossRef]

19. Wan, E.A.; van der Merwe, R. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium* (Cat. No.00EX373), Lake Louise, AB, Canada, 1–4 October 2000.

20. Zhong, Z.; Zhao, E.; Zheng, X.; Zhao, X. A consensus-based square-root cubature Kalman filter for maneuvering target tracking in sensor networks. *Trans. Inst. Meas. Control* **2020**, *42*, 3052–3062. [CrossRef]

21. Zhou, D.; Xi, Y.; Zhang, Z. Suboptimal fading extended Kalman filter for nonlinear systems. *Control Decis.* **1990**, *5*, 1–6.

22. Zhou, D.; Xi, Y.; Zhang, Z. A suboptimal multiple fading extended Kalman filter. *Acta Autom. Sin.* **1991**, *17*, 689–695.

23. Liu, Q.; Huang, C.; Li, P. Distributed consensus strong tracking filter for wireless sensor networks with model mismatches. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 1–10. [CrossRef]

24. Hu, G.; Gao, D.; Zhong, Y.; Subic, A. Modified strong tracking unscented Kalman filter for nonlinear state estimation with process model uncertainty. *Int. J. Adapt. Control Signal Process.* **2015**, *29*, 1561–1577. [CrossRef]

25. Xia, B.; Wang, H.; Wang, M.; Sun, W.; Xu, Z.; Lai, Y. A new method for state of charge estimation of lithium-ion battery based on strong tracking cubature Kalman filter. *Energies* **2015**, *8*, 13458–13472. [CrossRef]

26. Li, J.; Zhao, R.; Chen, J.; Zhao, C.; Zhu, Y. Target tracking algorithm based on adaptive strong tracking particle filter. *IET Sci. Meas. Technol.* **2016**, *10*, 704–710.

27. Zhang, A.; Bao, S.; Gao, F.; Bi, W. A novel strong tracking cubature Kalman filter and its application in maneuvering target tracking. *Chin. J. Aeronaut.* **2019**, *32*, 2489–2502. [CrossRef]

28. Liu, H.; Wu, W. Strong tracking spherical simplex-radial cubature Kalman filter for maneuvering target tracking. *Sensors* **2017**, *17*, 741. [CrossRef]

29. Li, Z.; Yang, W.; Ding, D. A novel fifth-degree strongly tracking cubature Kalman filter for two-dimensional maneuvering target tracking. *Math. Probl. Eng.* **2018**, *2018*, 5918456. [CrossRef]

30. Wang, J.; Zhang, T.; Xu, X.; Li, Y. A variational Bayesian based strong tracking interpolatory cubature Kalman filter for maneuvering target tracking. *IEEE Access* **2018**, *6*, 52544–52560. [CrossRef]

31. Zhang, H.; Xie, J.; Ge, J.; Lu, W.; Liu, B. Strong tracking SCKF based on adaptive CV model for maneuvering aircraft tracking. *IET Radar Sonar Navig.* **2018**, *12*, 742–749. [CrossRef]

32. Ma, J.; Guo, X. Combination of IMM algorithm and ASTRWCKF for maneuvering target tracking. *IEEE Access* **2020**, *8*, 143095–143103. [CrossRef]

33. Jiang, Y.; Ma, P.; Baoyin, H. Residual-normalized strong tracking filter for tracking a noncooperative maneuvering spacecraft. *J. Guid. Control Dyn.* **2019**, *42*, 2304–2309. [CrossRef]

34. Gao, W.; Li, J.; Zhou, G.; Li, Q. Adaptive Kalman filtering with recursive noise estimator for integrated SINS/DVL systems. *J. Navig.* **2015**, *68*, 142–161. [CrossRef]

35. Huang, Y.; Zhang, Y.; Wu, Z.; Li, N.; Chambers, J. A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices. *IEEE Trans. Autom. Control* **2018**, *63*, 594–601. [CrossRef]

36. Ge, B.; Zhang, H.; Jiang, L.; Li, Z.; Butt, M. Adaptive unscented Kalman filter for target tracking with unknown time-varying noise covariance. *Sensors* **2019**, *19*, 1371. [CrossRef] [PubMed]

37. Fang, X.; Chen, L. Noise-aware maneuvering target tracking algorithm in wireless sensor networks by a novel adaptive cubature Kalman filter. *IET Radar Sonar Navig.* **2020**, *14*, 1795–1802. [CrossRef]

38. Fang, W.; Jiang, J.; Lu, S.; Gong, Y.; Tao, Y.; Tang, Y.; Yan, P.; Luo, H.; Liu, J. A LSTM algorithm estimating pseudo measurements for aiding INS during GNSS signal outages. *Remote Sens.* **2020**, *12*, 256. [CrossRef]
39. Wu, F.; Luo, H.; Jia, H.; Zhao, F.; Xiao, Y.; Gao, X. Predicting the noise covariance with a multitask learning model for Kalman filter-based GNSS/INS integrated navigation. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 8500613. [CrossRef]

40. Hu, L.; Tang, Y.; Zhou, Z.; Pan, W. Reinforcement learning for orientation estimation using inertial sensors with performance guarantee. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi’an, China, 30 May–5 June 2021.

41. Tang, Y.; Hu, L.; Zhang, Q.; Pan, W. Reinforcement learning compensated extended Kalman filter for attitude estimation. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS), New York, NY, USA, 27 September–1 October 2021.

42. Gao, X.; Luo, H.; Ning, B.; Zhao, F.; Bao, L.; Gong, Y.; Xiao, Y.; Jiang, J. RL-AKF: An adaptive Kalman filter navigation algorithm based on reinforcement learning for ground vehicles. *Remote Sens.* **2020**, *12*, 1704. [CrossRef]

43. Liu, Y.; Wang, X.; Liu, K. Network anomaly detection system with optimized DS Evidence Theory. *Sci. World J.* **2014**, *753659*, 1–14. [CrossRef]

44. Wang, Y.; Li, X. Distributed estimation fusion with unavailable cross-correlation. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 259–278. [CrossRef]

45. Henderson, P.; Islam, R.; Bachman, P.; Pineau, J.; Precup, D.; Meger, D. Deep reinforcement learning that matters. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.

46. Li, B.; Gan, Z.; Chen, D.; Sergey Aleksandrovich, D. UAV maneuvering target tracking in uncertain environments based on deep reinforcement learning and meta-learning. *Remote Sens.* **2020**, *12*, 3789. [CrossRef]