


Article

Estimation of Vehicle State Based on IMM-AUKF

Ying Xu , Wenjie Zhang, Wentao Tang, Chengxiang Liu *, Rong Yang *, Li He * and Yun Wang *

College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen 518060, China; 2016110087@email.szu.edu.cn (W.Z.); 1900292008@email.szu.edu.cn (W.T.)

* Correspondence: yxu@szu.edu.cn (Y.X.); chxliu@szu.edu.cn (C.L.); ryang@szu.edu.cn (R.Y.); heli@szu.edu.cn (L.H.); wangyun@szu.edu.cn (Y.W.)

Abstract: Establishing a symmetrical model of surrounding vehicles and accurately obtaining the driving state of the surrounding vehicles in the driving environment can improve the safety of driving, which is an important issue that needs to be considered in the automatic driving system or auxiliary driving system. Therefore, we propose an adaptive unscented Kalman filter algorithm based on Interacting Multiple Model (IMM) theory to estimate the state of target vehicle in the high-speed driving environment. To be specific, we use the Constant Turn Rate and Acceleration (CTRA) theory to establish the target vehicle kinematics model, simultaneously, in order to overcome the problem of estimator failure when the yaw rate is close to zero, a simplified version of the CTRA model is also introduced into the estimation process. In addition, the parameter adaptation strategy is added, so the proposed estimator can overcome the uncertainty of the noise model and improve its accuracy. Finally, the effectiveness of proposed state estimation algorithm is verified on the Carsim and Simulink co-simulation platform. The results of simulations and experiments show that the accuracy and stability of IMM-based algorithm is better than the single-model algorithm in different scenarios, and the parameter adaptation strategy brings performance improvement.

Keywords: state estimation; Interacting Multiple Model; parameter adaptation; symmetrical model of surrounding vehicles



Citation: Xu, Y.; Zhang, W.; Tang, W.; Liu, C.; Yang, R.; He, L.; Wang, Y. Estimation of Vehicle State Based on IMM-AUKF. *Symmetry* **2022**, *14*, 222. <https://doi.org/10.3390/sym14020222>

Academic Editor: Jan Awrejcewicz

Received: 22 December 2021

Accepted: 21 January 2022

Published: 24 January 2022

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



Copyright: © 2022 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

Recently, owing to increasing demand for driving stability and safety, Advanced Driver Assistant Systems (ADAS) have developed rapidly in our daily life, which have significantly improved the safety, performance and efficiency of vehicles in various complex driving environments [1]. Additionally, accurate and stable estimation of vehicle driving state information such as the heading, yaw rate, absolute speed and acceleration of the target vehicle, as well as the relative position and relative speed of the target vehicle relative to the host vehicle, is essential for ADAS [2].

However, in practical applications, real-time acquisition of vehicle state usually requires expensive equipment, existing professional vehicle signal acquisition systems are very expensive and their installation is complicated, and some of the key states are hard to measure directly, making it difficult to popularize [3]. In addition, under different driving conditions such as straights and curves, if the state estimation methods cannot identify the real-time state of the target vehicle, the whole control process will be unstable [4,5]. Therefore, it is necessary to develop a low-cost and high precision online vehicle state estimation system that is applicable to real driving condition [6]. Current state estimation algorithm is mainly divided into two major categories: artificial intelligence (AI) and model-based.

Due to the development of high-performance computing hardware and the emergence of a large number of labeled datasets, the state estimation method based on AI has achieved great success. Ref. [7] used neural network method to detect and annotate vehicle for streaming video data with complex scenes, aiming to infer each vehicle's pose, color and type. Lin et al. [8] presents an auto-masking neural network for vehicle detection and pose

estimation. As proposed in [9], unsupervised convolutional neural network is adopted for vehicle type classification from frontal view images. Refs. [10–13] use Convolutional Neural Network (CNN) to estimate the actual state of the detected vehicles. Although the data-driven artificial intelligence method has achieved unprecedented success in state estimation, it requires a large amount of labeled data for training to prevent neural network parameters from overfitting.

The main research activities in the field of model-based state estimation concentrate on the choice of vehicle dynamics model and the application of filter theory. Ref. [14] presents a non-linear model-based observer which relies on an augmented Extended Kalman filter for combined estimation of motion states. Ref. [15] applies a particle filter (PF) and an unscented Kalman filter (UKF) to estimate the headway and velocity of a six-vehicle platoon system. However, due to the existence of various road conditions such as straights and curves in reality, the use of a single motion model-based approach often cannot accurately fit the state of the target vehicle. Therefore, the multiple-model structure is adopted. Ref. [16] proposes an interacting multiple model algorithm, using improved Markov process to describe the switching probability among the models. And a number of multiple-model techniques have been proposed [17–23]. Additionally, in order to improve the performance of the filter, the parameter adaptive process is introduced. Ref. [24] describes an adaptive interacting multiple-model algorithm that does not need predefined models for use in manoeuvring target tracking. To improve the robustness of the unscented Kalman filter when applied to highly nonlinear models, ref. [25] proposes a new method by modifying the main scaling parameter at every step using a self-adaptive algorithm.

In this paper, we propose an adaptive parameter Interacting Multiple Model unscented Kalman filter algorithm (IMM-AUKF) to achieve accurate real-time target vehicle state estimation performance. The remainder of the paper is organized as follows: In Section 2, vehicle coordinate, dynamics models and UKF algorithm procedure will be introduced. The structure and detailed design about IMM-AUKF algorithm is covered in Section 3. In Section 4, we use Carsim and Simulink co-simulation to verify the effectiveness of IMM-AUKF, in order to verify the algorithm we proposed, the experimental scene adopts flat road, only one and the same host car and one and the same target car, and set up two control experiments. Finally, in Section 5, we offer our conclusions.

2. Problem Formulation

2.1. Vehicle Coordinate

As shown in Figure 1, the research object of this paper is the vehicle in front of the host vehicle, and the target vehicle generally has no vertical movement, or the vertical movement speed is very small compared to the horizontal movement speed. Therefore, ignoring the movement of the vehicle in the vertical direction, the state estimation of the target vehicle can be simplified as the state estimation of the object on the plane in the radar coordinate system [26,27]. For the purpose of simplifying the problem, we set the on-board sensor as the origin of the coordinate system, the x-axis is parallel to the ground and points to the forward direction of the vehicle, the y-axis is parallel to the ground and points to the left of the driver. The output signal of the millimeter wave radar is given in the form of polar coordinates. In order to unify the radar coordinate system with the coordinate system defined above, we set the radar coordinate system as Figure 1: the center of the radar installation position is the pole, the polar axis coincides with the x-axis of the coordinate system and is positive in the counterclockwise direction.

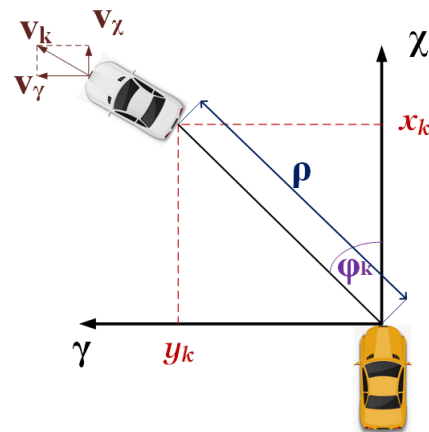


Figure 1. Vehicle Coordinate and Measurement model.

The state vector of the target vehicle at time k is as follows:

$$X(k) = [x_k, y_k, v_k, a_k, \varphi_k, \omega_k]^T \quad (1)$$

where x and y are the components of polar coordinate distance between target vehicle and host vehicle ρ on the x -axis and y -axis respectively. v and a are the velocity and acceleration of the target vehicle respectively. φ is the angle between the target vehicle and the x -axis in the vehicle coordinate system, and ω is yaw rate.

So, the conversion between the millimeter wave radar coordinate system and the vehicle body coordinate system is:

$$\begin{bmatrix} \rho \\ \phi \end{bmatrix} = \begin{bmatrix} \sqrt{x_k^2 + y_k^2} \\ \text{atan2}(y_k, x_k) \end{bmatrix} \quad (2)$$

2.2. Construction of CTRA Model

CTRA model assumes that the tracked object is moving with a constant rate and a constant acceleration [28] from time k to time $k + 1$. Thus, the displacement of the object in the two axes is computed using the following equation:

$$X(k+1) = \begin{pmatrix} x_k + \frac{v_k \cdot (\sin(\varphi_{k+1}) - \sin(\varphi_k))}{\omega_k} + \frac{a_k \cdot (\cos(\varphi_{k+1}) - \cos(\varphi_k)) + a_k \cdot \omega_k \cdot T \cdot \sin(\varphi_{k+1})}{\omega_k^2} \\ y_k - \frac{v_k \cdot (\cos(\varphi_{k+1}) - \cos(\varphi_k))}{\omega_k} + \frac{a_k \cdot (\sin(\varphi_{k+1}) - \sin(\varphi_k)) - a_k \cdot \omega_k \cdot T \cdot \cos(\varphi_{k+1})}{\omega_k^2} \\ v_k + a_k \cdot T \\ a_k \\ \varphi_k + \omega_k \cdot T \\ \omega_k \end{pmatrix}, \omega_k \neq 0 \quad (3)$$

where T is the time interval between two sequential scans.

This model can efficiently describe the true motion of a vehicle in the road, since the true trajectory can be considered that is consisted of segments where the turn rate and the acceleration remain constant. However, note this expression can be degenerated when the ω_k -component of Equation (3) is close to zero, while this issue can be easily resolved

by assuming that the vehicle keeps driving in a straight line, and then Equation (3) is simplified to Equation (4).

$$X(k+1) = \begin{pmatrix} x_k + (v_k \cdot T + \frac{1}{2} \cdot a_k \cdot T^2) \cos(\varphi) \\ y_k + (v_k \cdot T + \frac{1}{2} \cdot a_k \cdot T^2) \sin(\varphi) \\ v_k + a_k \cdot T \\ a_k \\ \varphi_k \\ 0 \end{pmatrix}, \omega_k = 0 \quad (4)$$

2.3. Unscented Kalman Filter (UKF)

For nonlinear motion, we need to consider the interference of noise. For different time k , assume that nonlinear system contains the current state is $X(k)$ with white Gaussian noise $W(k)$ and measurements Z with white Gaussian noise $V(k)$, integrates the interactive input result obtained in Equation (1), the system can be described by Equation (5).

$$\begin{cases} X(k+1) = f(X(k), W(k)) \\ Z(k) = h(X(k), V(k)) \end{cases} \quad (5)$$

where $f(\cdot)$ is the nonlinear state equation function, determined by the motion model Equation (3) or Equation (4), and h is the nonlinear observation equation function, determined by the observation model Equation (2).

Suppose $W(k)$ has a covariance matrix Q , and $V(k)$ has a covariance matrix R . According [29], the steps of the UKF algorithm for the state vector X at different time k are as follows:

(1) Use unscented transform to calculate $2n + 1$ sigma points and corresponding weights, the superscript denotes i -th sigma point.

$$X^{(i)}(k|k) = \left[\hat{X}(k|k) \hat{X}(k|k) + (\sqrt{(n+\lambda)P(k|k)}) \hat{X}(k|k) - (\sqrt{(n+\lambda)P(k|k)}) \right] \quad (6)$$

where $\hat{\cdot}$ and P denote the mean and variance of the current state of the target vehicle respectively. Then use the nonlinear function Equation (3) to predict the $2n + 1$ sigma point set obtained from Equation (6):

$$X^{(i)}(k+1|k) = f[k, X^{(i)}(k|k)], i = 1, 2, \dots, 2n+1 \quad (7)$$

And acquire the mean $\hat{X}(k+1|k)$ and covariance $P(k+1|k)$ of the new distribution based on the weight ω_m corresponding to each sigma point:

$$\begin{cases} \hat{X}(k+1|k) = \sum_{i=0}^{2n} \omega_m^{(i)} X^{(i)}(k+1|k) \\ P(k+1|k) = \sum_{i=0}^{2n} \omega_c^{(i)} ([\hat{X}(k+1|k) - X^{(i)}(k+1|k)] \cdot [\hat{X}(k+1|k) - X^{(i)}(k+1|k)]^T) + Q \end{cases} \quad (8)$$

(2) According to the predicted value, use Equation (6) again to generate a new sigma point set

$$X^{(i)}(k+1|k) = [\hat{X}(k+1|k) \hat{X}(k+1|k) + (\sqrt{(n+\lambda)P(k+1|k)}) \cdot [\hat{X}(k+1|k) - (\sqrt{(n+\lambda)P(k+1|k)})] \quad (9)$$

Then substitute the new sigma point set into the observation equation Equation (2) to get the predicted measurement:

$$Z^{(i)}(k+1|k) = h[X^{(i)}(k+1|k)] \quad (10)$$

Therefore, the mean and covariance matrix of the system prediction can be obtained by weighted summation:

$$\begin{cases} \bar{Z}(k+1|k) = \sum_{i=0}^{2n} \omega_m^{(i)} Z^{(i)}(k+1|k) \\ P_{ZZ,k} = \sum_{i=0}^{2n} \omega_c^{(i)} [Z^{(i)}(k+1|k) - \bar{Z}(k+1|k)][Z^{(i)}(k+1|k) - \bar{Z}(k+1|k)]^T + R \\ P_{XZ,k} = \sum_{i=0}^{2n} \omega_c^{(i)} [X^{(i)}(k+1|k) - \hat{X}(k+1|k)][Z^{(i)}(k+1|k) - \bar{Z}(k+1|k)]^T \end{cases} \quad (11)$$

(3) Finally, calculate the Kalman gain, update the state and covariance matrix of the system.

$$\begin{cases} K(k+1) = P_{XZ,k} P_{ZZ,k}^{-1} \\ \hat{X}(k+1|k+1) = \hat{X}(k+1|k) + K(k+1)[Z(k+1) - \bar{Z}(k+1|k)] \\ P(k+1|k+1) = P(k+1|k) - K(k+1) \cdot P_{ZZ,k} \cdot K^T(k+1) \end{cases} \quad (12)$$

Traditional KF and EKF perform Taylor series expansion at the estimated point, and then perform n-order approximation. Different from them, UKF performs UT near the estimated point to match the mean and variance of the obtained sigma point set with the original statistical characteristics, and then directly performs nonlinear mapping on these sigma-point sets to obtain the state probability density by using statistical approximation methods [30].

3. Design of State Estimation Basing on IMM-AUKF

The main idea of the Interacting Multiple Model (IMM) algorithm is to realize automatic recognition and switching of multiple models based on Bayesian theory [31]. At any tracking moment, set multiple model filters for real-time motion model detection, and set the weight coefficient as well as model update probability for each filter, finally obtain the current optimal estimation state through weighted calculation, so as to achieve model adaptation track. The following Figure 2 is the flow chart of the IMM-AUKF algorithm:

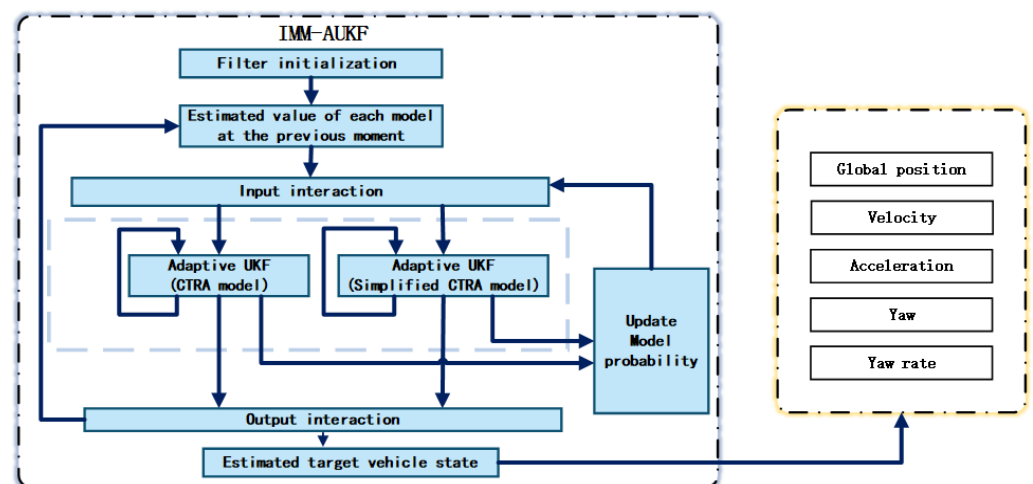


Figure 2. Block diagram of IMM-AUKF algorithm.

It is assumed that the target has r motion states, corresponding to r motion models, and the transition between the models is determined by the Markov probability transition

matrix of Equation (13), where p_{ij} indicates that the target is transferred from the i -th motion model to the probability of the j -th motion model [32].

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1r} \\ \vdots & \ddots & \vdots \\ p_{r1} & \cdots & p_{rr} \end{bmatrix} \quad (13)$$

IMM-AUKF algorithm are used to achieve recursive estimation, each recursion is mainly divided into the following four steps.

3.1. Input Interaction

Assuming that the current moment is k , take model j as an example. According to the optimal state estimation $\hat{X}_j(k-1|k-1)$, the covariance matrix $P_i(k-1|k-1)$ and the probability model of each filter $\mu_i(k-1)$ obtained by using the IMM-UKF algorithm at the last state estimation process, the hybrid state estimation and covariance are obtained. And the mixed estimation is used as the initial state of the current cycle, the parameter calculation process is as follows:

The predicted probability of model j (normalized constant) is

$$\bar{c}_j = \sum_{i=1}^r p_{ij} \mu_i(k-1) \quad (14)$$

Then we propagate the hybrid probability from model i to model j based on output hybrid probability from last scan:

$$\mu_{ij}(k-1|k-1) = \sum_{i=1}^r p_{ij} \mu_i(k-1) / \bar{c}_j \quad (15)$$

Finally, the output estimated state and covariance of model j is

$$\left\{ \begin{array}{l} \hat{X}_{0j}(k-1|k-1) = \sum_{i=1}^r \hat{X}_i(k-1|k-1) \mu_{ij}(k-1|k-1) \\ P_{0j}(k-1|k-1) = \sum_{i=1}^r \mu_{ij}(k-1|k-1) \{ P_i(k-1|k-1) \\ + [\hat{X}_i(k-1|k-1) - \hat{X}_{0j}(k-1|k-1)] \\ \cdot [\hat{X}_i(k-1|k-1) - \hat{X}_{0j}(k-1|k-1)]^T \} \end{array} \right. \quad (16)$$

3.2. Adaptive Parameter UKF

When the vehicle-mounted sensor tracks the state of the preceding vehicle, due to the interference of various factors, the data filtered by the UKF may contain other noise values, which will inevitably increase the error of the filtering and noise reduction. In order to solve this problem, the AUKF algorithm is implemented by combining the UKF and Sage-Husa filtering algorithms [33]. The Sage-Husa adaptive filter enables colleagues who use the original data to filter and reduce noise, and correct the system noise variance in real time, which greatly improves the filtering. Combining Equation (17), the adaptive parameter process is as follows:

$$\left\{ \begin{array}{l} \varepsilon_{k+1} = Z(k+1) - Z(k+1|k) \\ d_{k+1} = \frac{1-b}{1-b^{k+1}} \\ R_{k+1} = (1-d_{k+1})R_k + d_{k+1} \cdot \varepsilon_{k+1} \cdot \varepsilon_{k+1}^T \\ Q_{k+1} = (1-d_{k+1})Q_k + d_{k+1}(\varepsilon_{k+1} \cdot \varepsilon_{k+1}^T \cdot k_{k+1}^T + P_{k+1}) \end{array} \right. \quad (17)$$

where b is the forgetting factor, which is generally $0.95 \leq b \leq 0.99$.

Use Equations (16) and (17) and millimeter-wave radar measurement values $Z(k)$ as input to the UKF mentioned in Section 2.3, then use the adaptive parameters mentioned above to update the prediction state $\hat{X}_j(k|k)$ and filter covariance $P_j(k|k)$.

3.3. Update Model Probability

Use the maximum likelihood function to update the model probability, and the likelihood function of model j is:

$$\Lambda_j(k) = \frac{1}{(2\pi)^{n/2} |S_j(k)|^{1/2}} \exp \left\{ -\frac{1}{2} v_j^T S_j^{-1}(k) v_j \right\} \quad (18)$$

where:

$$\begin{cases} v_j(k) = Z(k) - H(k) \hat{X}_j(k|k-1) \\ S_j(k) = H(k) P_j(k|k-1) H(k)^T + R(k) \end{cases} \quad (19)$$

Now, the probability of model j can be parameterized as:

$$\mu_j(k) = \Lambda_j(k) \cdot \bar{c}_j / c \quad (20)$$

where c is the normalization constant:

$$c = \sum_{j=1}^r \Lambda_j(k) \cdot \bar{c}_j \quad (21)$$

3.4. Output Interaction

Based on Equations (14)–(21), the estimation results of each filter are weighted and combined to obtain a comprehensive state estimation and covariance estimation. Total estimated state and covariance is as Equation (22) shown

$$\begin{cases} \hat{X}(k|k) = \sum_{j=1}^r \hat{X}_j(k|k) \mu_j(k) \\ P(k|k) = \sum_{j=1}^r \mu_j(k) \{ P_j(k|k) + [\hat{X}_j(k|k) - \hat{X}(k|k)] \cdot [\hat{X}_j(k|k) - \hat{X}(k|k)]^T \} \end{cases} \quad (22)$$

The above is a filtering derivation process of the IMM-UKF tracking algorithm. Each time, the previous interactive output is used as the next interactive input value, thereby completing the entire filter tracking process loop.

4. Testing and Analyzing

4.1. Experiment Introduction and Evaluation Indicators

In order to test the effectiveness of proposed IMM-AUKF algorithm, we use Carsim to set up a road as shown in Figure 3. The road contains straights, left turns, and right turns and other elements, which are used to simulate the four driving environments in our daily life: the target vehicle (t) and host vehicle (h) are all in a straight road, t enters a turn and h is still in the straight road, both cars are in a curve, and t leaves a turn and h is still in the curve.

In the four scenarios, the virtual radar obtains the relative motion information of the target vehicle, and transmits it to the IMM-AUKF estimator established in the Simulink environment in real time through the interface to realize joint simulation. The criteria that are used to evaluate the performance of each filter are based in the calculation of the estimation errors of the actual distance. The best filter is the one that minimize the mean of filtering error $\bar{e}_x(k)$ and the standard deviation $\sigma_{\hat{x}}$.

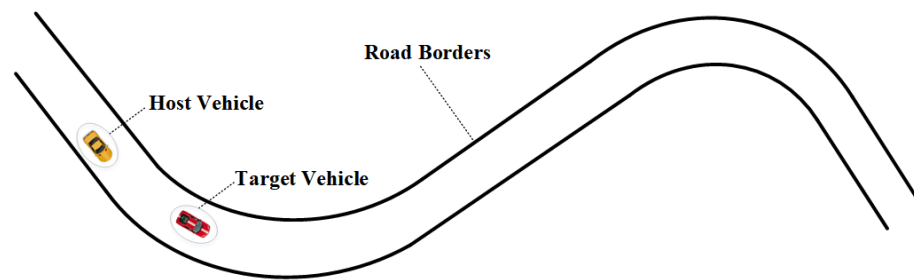


Figure 3. Schematic diagram of the experimental scene.

$$\begin{cases} \overline{e_x(k)} = \frac{1}{M} \sum_{i=1}^M [x_i(k) - \hat{x}_i(k|k)] \\ \sigma_{\hat{x}} = \sqrt{\frac{1}{M} \sum_{i=1}^M [x_i(k) - \hat{x}_i(k|k)]^2 - [\overline{e_x(k)}]^2} \end{cases} \quad (23)$$

where M is the number of MonteCarlo simulations, it is $k = 1, 2, \dots, n$, n is sampling times, we take $n = 50$.

4.2. Results Analysis

Figure 4 is the trajectory estimated by IMM-AUKF, and Figure 5 is the speed, acceleration, yaw and the yaw rate estimated by IMM-AUKF. It can be seen from the simulation results that whether the target vehicle is driving in a straight line or a curve, the filtering outputs are all small fluctuations near the actual value.

Figures 6 and 7 shows the estimation error and standard deviation estimation error for each state respectively. It can be seen from the curve that because there are two turns, there are four large fluctuations in the error before and after the turn. When the trajectory of the target becomes a uniform motion, the error curve returns to zero with small fluctuations.

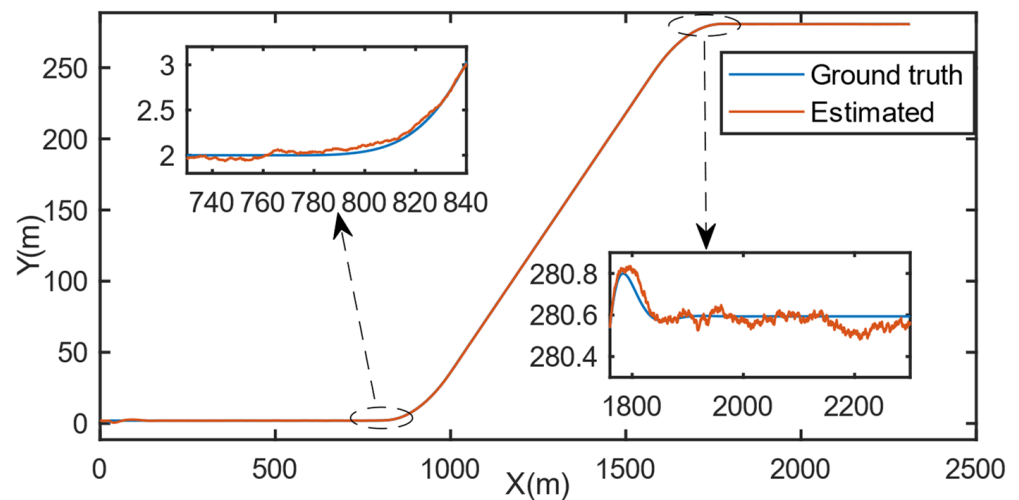


Figure 4. Global position estimated by IMM-AUKF.

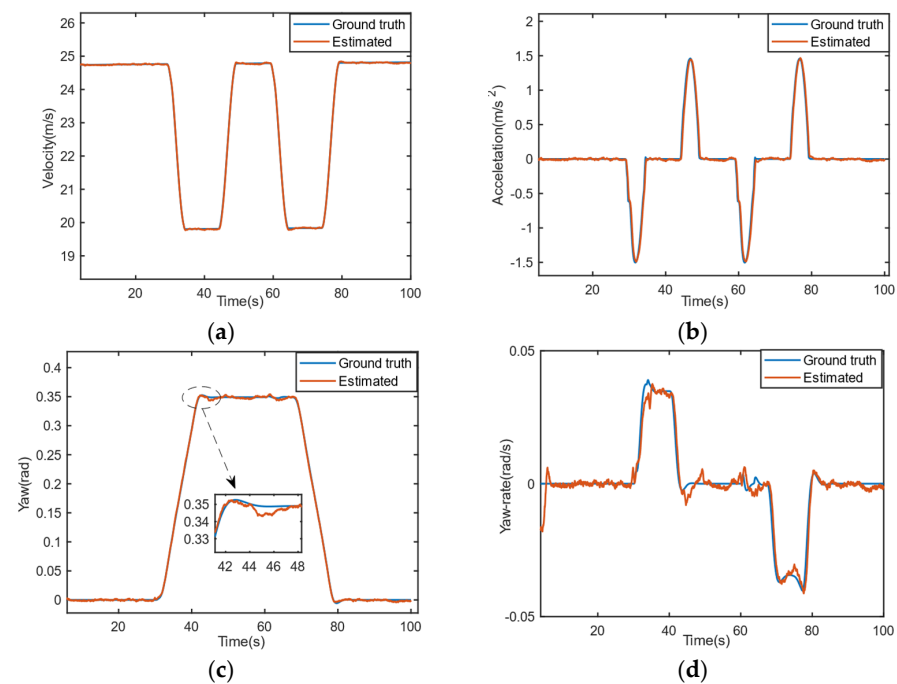


Figure 5. Target vehicle state estimated by IMM-AUKF. (a) velocity, (b) acceleration, (c) yaw, (d) yaw rate.

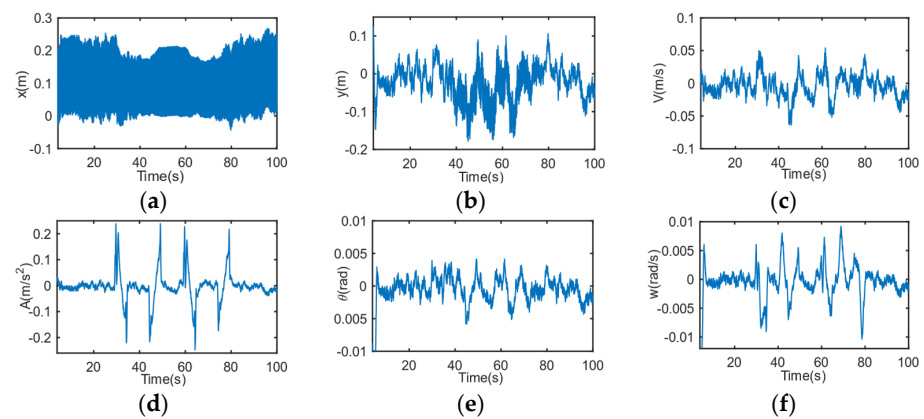


Figure 6. Estimation error of IMM-AUKF for multi-states. (a) x-direction, (b) y-direction, (c) velocity, (d) acceleration, (e) yaw, (f) yaw rate.

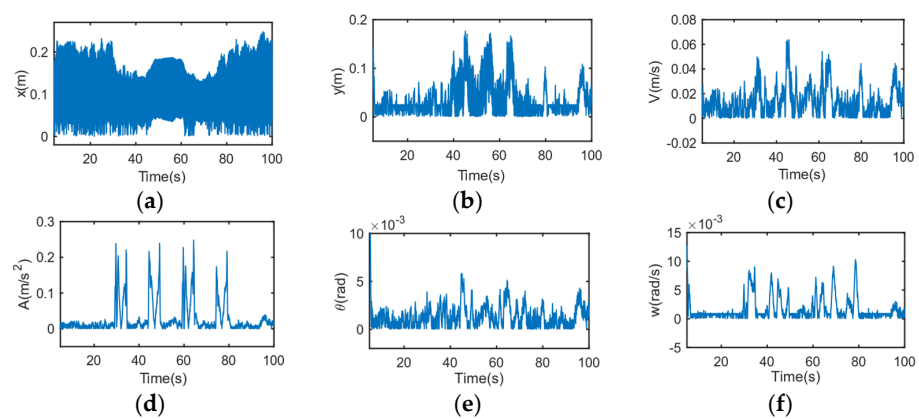


Figure 7. Standard deviation estimation error of IMM-AUKF for multi-states. (a) x-direction, (b) y-direction, (c) velocity, (d) acceleration, (e) yaw, (f) yaw rate.

4.3. Ablation Experiment

4.3.1. Comparative Analysis of Multi-model and Single-Model Estimation

It can be seen from Figures 8 and 9 that the IMM-UKF algorithm performs better than the CTRA-UKF/simplified CTRA-UKF algorithm when the target vehicle is traveling in a straight line; when the target vehicle is traveling on a curve, the IMM-UKF model performs better than the Simplified CTRA-UKF/simplified CTRA-UKF algorithm. This is because the IMM algorithm in this paper uses two models to describe the possible state of the target vehicle during driving, and then uses effective weighted fusion to estimate the system state, which overcomes the single-model state estimation in different motion modes (straight roads, Corners) poor performance when switching.

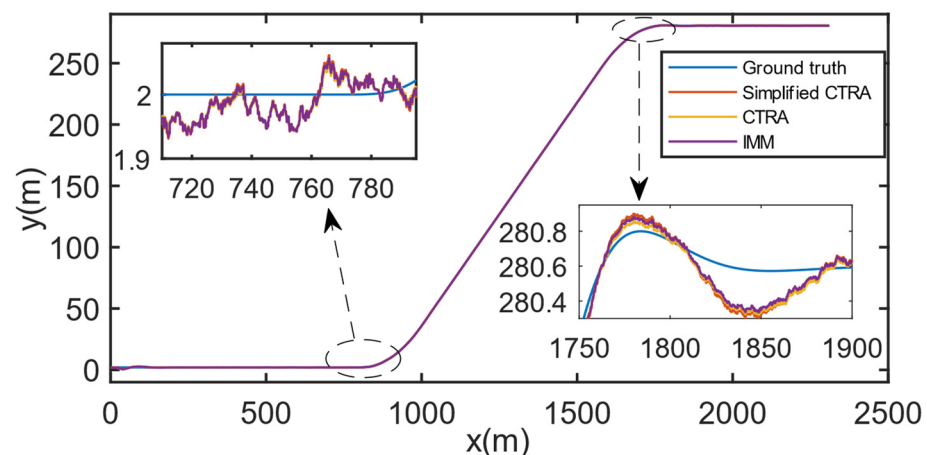


Figure 8. The performance of different estimation methods on global position.

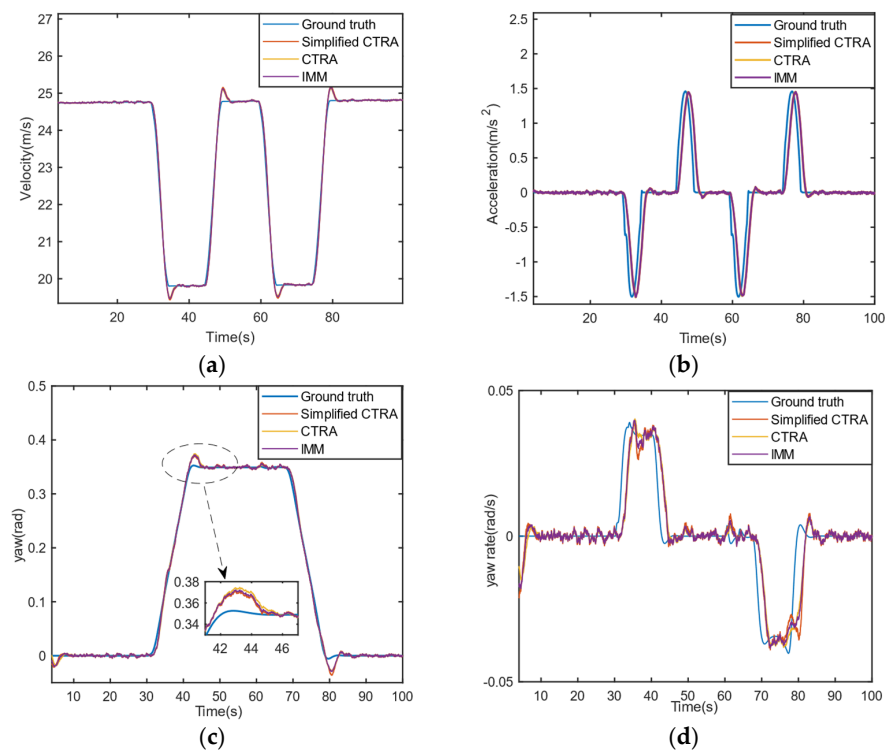


Figure 9. The performance of different estimation methods on multi-states. (a) velocity, (b) acceleration, (c) yaw, (d) yaw rate.

4.3.2. Comparison of AUKF and UKF Based on Multi-Model

Figures 10 and 11 are comparisons results of IMM-UKF with IMM-AUKF. It can be seen that after the step of parameter adaptation is added, the filtering result is closer to the real value, and the problem of large delay in state estimation in filtering is eliminated.

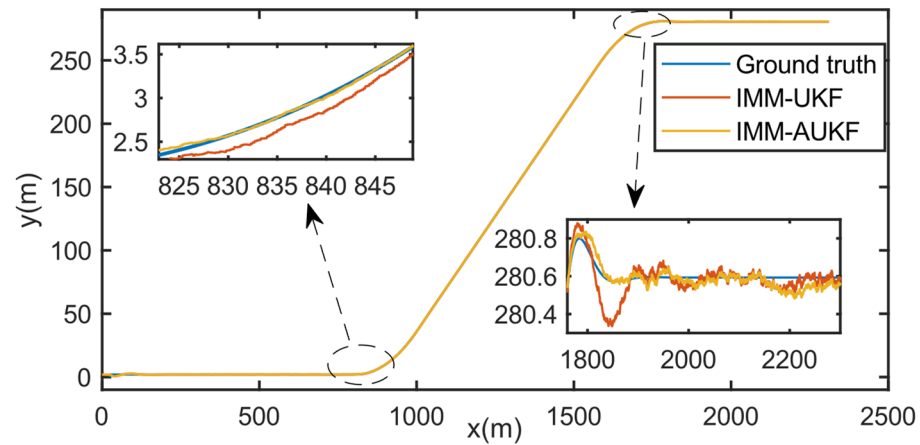


Figure 10. The performance of different estimation methods on global position.

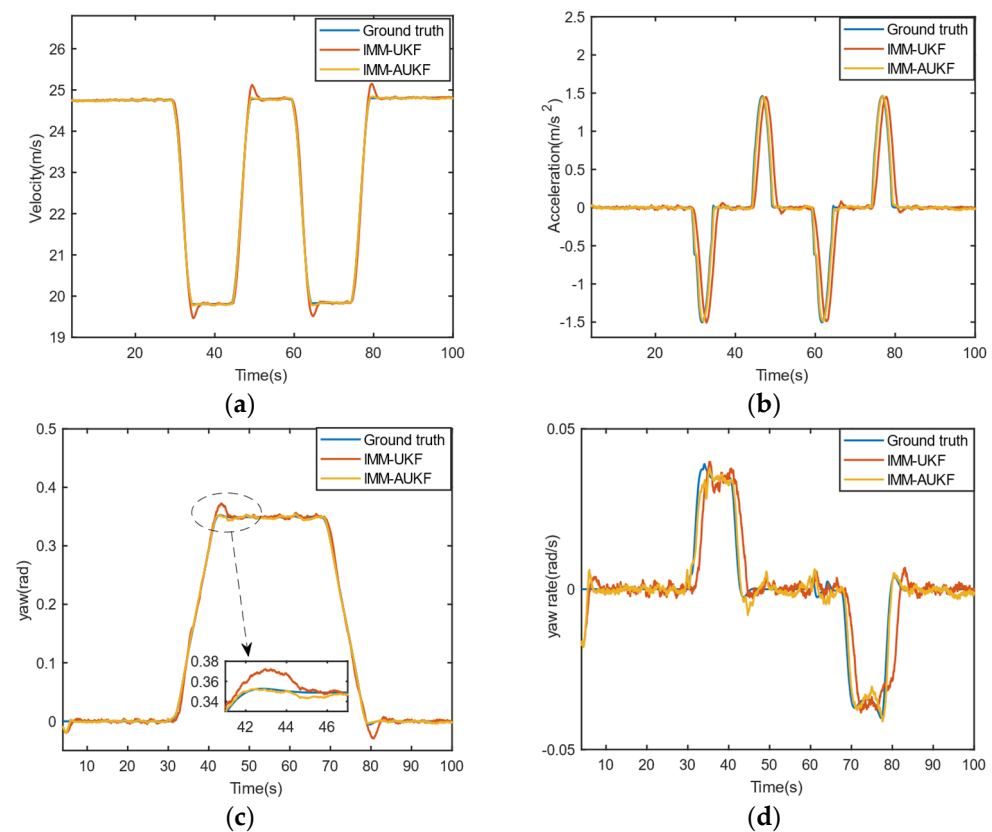


Figure 11. The performance of IMM-UKF and IMM-AUKF on multi-states. (a) velocity, (b) acceleration, (c) yaw, (d) yaw rate.

Table 1 shows the mean value of the standard deviation of 4 models. It can be seen that the state estimation filtered by IMM-AUKF is closer to the actual states of the tracked object, and its performance is more stable when switching between different states.

Table 1. The STD error of different estimation methods on different driving environment.

Motion	CTRA	Simplified CTRA	IMM-UKF	IMM-AUKF
Drive straight	0.0461	0.0464	0.0459	0.0379
Cornering	0.1480	0.1325	0.1387	0.0411

5. Conclusions and Future Work

In this paper, based on Interacting Multiple Model theory and parameter adaptation strategy, a real-time target vehicle state estimation algorithm is proposed. We use the Constant Turn Rate and Acceleration (CTRA) theory to establish the target vehicle kinematics model, and add its simplified version combine with IMM algorithm to improve the estimator performance in dealing with different driving environments. Additionally, in order to enhance the stability of the proposed algorithm, we add parameter adaptation process to the UKF algorithm to overcome the uncertainty of the noise model. Carsim and Simulink co-simulation results show that our proposed algorithm performs better than the single-model state estimation algorithm on accuracy and stability.

In the future, we will use public datasets to validate our proposed algorithm, and consider applying multi-sensor fusion algorithm to enhance the robustness of the algorithm.

Author Contributions: Conceptualization, Y.X.; methodology, W.T.; software, W.T.; validation, W.Z. and Y.X.; formal analysis, W.T.; investigation, C.L. and R.Y.; resources, L.H. and Y.W.; data curation, W.T.; writing—original draft preparation, Y.X. and W.Z.; writing—review and editing, C.L. and R.Y.; visualization, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61403259, 61773266, 2019YFB2102703; and the Science and Technology Research and Development Foundation of Shenzhen, grant number 20200813140339001, 20200809215801001, JCYJ20210324120209027.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- Kastner, R.; Michalke, T.; Adamy, J.; Fritsch, J.; Goerick, C. Task-Based Environment Interpretation and System Architecture for Next Generation ADAS. *IEEE Intell. Transp. Syst. Mag.* **2011**, *3*, 20–33. [\[CrossRef\]](#)
- Geng, G.; Wei, B.; Jiang, H.; Hua, Y.; Wu, Z. A Research on Driving State Estimation for Distributed Drive Electric Vehicle Based on NA-EKF. *Qiche Gongcheng/Automot. Eng.* **2018**, *40*, 770–776. [\[CrossRef\]](#)
- Li, X.; Chan, C.Y.; Yu, W. A Reliable Fusion Methodology for Simultaneous Estimation of Vehicle Sideslip and Yaw Angles. *IEEE Trans. Veh. Technol.* **2016**, *65*, 4440–4458. [\[CrossRef\]](#)
- Li, L.; Jia, G.; Chen, J.; Zhu, H.; Cao, D.; Song, J. A novel vehicle dynamics stability control algorithm based on the hierarchical strategy with constrain of nonlinear tyre forces. *Veh. Syst. Dyn.* **2015**, *53*, 1093–1116. [\[CrossRef\]](#)
- Wang, J.; Wu, J.; Li, Y. The Driving Safety Field Based on Driver-Vehicle-Road Interactions. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2203–2214. [\[CrossRef\]](#)
- Ying, X.; Biyun, C.; Cheng, C. Estimation of Road Friction Coefficient and Vehicle States by 3-DOF Dynamic Model and HSRI Model Based on Information Fusion. *Asian J. Control* **2018**, *20*, 1067–1076.
- Yi, Z.; Li, L.; Ling, S.; Matt, M. DAVE: A Unified Framework for Fast Vehicle Detection and Annotation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016.
- Yang, L.; Liu, J.; Tang, X. Object Detection and Viewpoint Estimation with Auto-masking Neural Network. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014.
- Dong, Z.; Wu, Y.; Pei, M.; Jia, Y. Vehicle Type Classification Using a Semisupervised Convolutional Neural Network. *Intelligent Transportation Systems. IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2247–2256. [\[CrossRef\]](#)
- Pino, I.D.; Vaquero, V.; Masini, B.; Solà, J.; Moreno-Noguer, F.; Sanfeliu, A.; Andrade-Cetto, J. Low resolution lidar-based multi object tracking for driving applications. In Proceedings of the Iberian Robotics Conference, Seville, Spain, 22–24 November 2017.
- Li, B. 3D fully convolutional network for vehicle detection in point cloud. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 1513–1518.
- Bo, L.; Zhang, T.; Tian, X. Vehicle Detection from 3D Lidar Using Fully Convolutional Network. *arXiv* **2016**, arXiv:1608.07916.

13. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-View 3D Object Detection Network for Autonomous Driving. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
14. Reina, G.; Messina, A. Vehicle Dynamics Estimation via Augmented Extended Kalman Filtering. *Measurement* **2018**, *133*, 383–395. [[CrossRef](#)]
15. Suzuki, H. Dynamic State Estimation in Vehicle Platoon System by Applying Particle Filter and Unscented Kalman Filter. *Procedia Comput. Sci.* **2013**, *24*, 30–41. [[CrossRef](#)]
16. Chen, B.; Tugnait, J.K. An interacting multiple model fixed-lag smoothing algorithm for Markovian switching systems. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 243–250. [[CrossRef](#)]
17. Cho, T.; Lee, C. Multi-Sensor Fusion with Interacting Multiple Model Filter for Improved Aircraft Position Accuracy. *Sensors* **2013**, *13*, 4122–4137. [[CrossRef](#)] [[PubMed](#)]
18. Toledo-Moreo, R.; Zamora-Izquierdo, M.A.; Ubeda-Miarro, B.; Gomez-Skarmeta, A.F. High-Integrity IMM-EKF-Based Road Vehicle Navigation with Low-Cost GPS/SBAS/INS. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 491–511. [[CrossRef](#)]
19. Qu, H.; Pang, L.; Li, S. A novel interacting multiple model algorithm. *Signal Process.* **2009**, *89*, 2171–2177. [[CrossRef](#)]
20. Li, X.R.; Jilkov, V.P. Survey of maneuvering target tracking. Part V. Multiple-model methods. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1255–1321.
21. Lee, B.J.; Park, J.B.; Lee, H.J.; Joo, Y.H. Fuzzy-logic-based IMM algorithm for tracking a manoeuvring target. *IEEE Proc. Radar Sonar Navig.* **2005**, *152*, 16–22. [[CrossRef](#)]
22. Bar-Shalom, Y.; Challa, S.; Subhash, B.; Henk, A.P. IMM estimator versus optimal estimator for hybrid systems. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 986–991. [[CrossRef](#)]
23. Kim, B.; Yi, K.; Yoo, H.-J.; Chong, H.-J.; Ko, B. An IMM/EKF Approach for Enhanced Multitarget State Estimation for Application to Integrated Risk Management System. *IEEE Trans. Veh. Technol.* **2015**, *64*, 876–889. [[CrossRef](#)]
24. Munir, A.; Atherton, D.P. Adaptive interacting multiple model algorithm for tracking a manoeuvring target. *Radar Sonar Navig. IEEE Proc.* **1995**, *142*, 11–17. [[CrossRef](#)]
25. Nie, Y.; Tao, Z. A self-adaptive scaling parameter selection algorithm for the Unscented Kalman Filter. In Proceedings of the Chinese Automation Congress, Wuhan, China, 27–29 November 2015.
26. Jo, K.; Chu, K.; Kim, J.; Sunwoo, M. Distributed vehicle state estimation system using information fusion of GPS and in-vehicle sensors for vehicle localization. In Proceedings of the 2011 14th International IEEE Conference on Intelligent Transportation Systems, Washington, DC, USA, 5–7 October 2011.
27. Huang, J.; Yan, B.; Hu, S. Centralized Fusion of Unscented Kalman Filter Based on Huber Robust Method for Nonlinear Moving Target Tracking. *Math. Probl. Eng.* **2015**, *2015*, 291913. [[CrossRef](#)]
28. Tsogas, M.; Polychronopoulos, A.; Amditis, A. Unscented Kalman filter design for curvilinear motion models suitable for automotive safety applications. In Proceedings of the 2005 7th International Conference on Information Fusion, Philadelphia, PA, USA, 25–28 July 2006.
29. 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, 4 October 2000; pp. 153–158.
30. He, H.; Qin, H.; Sun, X.; Shui, Y. Comparison Study on the Battery SoC Estimation with EKF and UKF Algorithms. *Energies* **2013**, *6*, 5088–5100. [[CrossRef](#)]
31. Kong, S.-P.; Zhang, H.-R.; Liao, X.-P.; Hong, D.-P. A Hierarchical Track Fusion Algorithm Based on IMM-UKF. *Fire Control Command Control* **2018**, *2*, 122–126.
32. Huang, D.; Leung, H. EM-IMM based land-vehicle navigation with GPS/INS. In Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No.04TH8749), Washington, WA, USA, 3–6 October 2004.
33. Zheng, Z.; Liu, S.; Zhang, B. An improved Sage-Husa adaptive filtering algorithm. In Proceedings of the 31st Chinese Control Conference, Hefei, China, 25–27 July 2012.