machines

MDPI

Article

Advanced State Estimation for Multi-Articulated Virtual Track Trains: A Fusion Approach

Zhenggang Lu [1], Zehan Wang [1,*] and Xianguang Luo [2]

[1] Institute of Rail Transit, Tongji University, Shanghai 201804, China
[2] CRRC Zhuzhou Locomotive Co., Ltd., Zhuzhou 412001, China
* Correspondence: wangzehan@tongji.edu.cn

Abstract: The Virtual Track Train (VTT) represents an innovative urban public transportation system that combines tire-based running gears with rail transit management. Effective control of such a system necessitates precise state estimation, a task rendered complex by the multi-articulated nature of the vehicles. This study addresses the challenge by focusing on state estimation for the first unit under significant interference, introducing a fusion state estimation strategy utilizing Gaussian Process Regression (GPR) and Interacting Multiple Model (IMM) techniques. First, a joint model for the first unit is established, comprising the dynamics model as the main model and a residual model constructed based on GPR to accommodate the main model's error. The proposed fusion strategy comprises two components: a kinematic model-based method for handling transient and high-acceleration phases, and a joint-model-based method suitable for near-steady-state and low-acceleration conditions. The IMM method is employed to integrate these two approaches. Subsequent units' states are computed from the first unit's state, articulation angles, and yaw rates' filtered data. Validation through hardware-in-the-loop (HIL) simulation demonstrates the strategy's efficacy, achieving high accuracy with an average lateral speed estimation error below 0.02 m/s and a maximum error not exceeding 0.22 m/s. Additionally, the impact on VTT control performance after incorporating state estimation is minimal, with a reduction of only 3–6%.

Keywords: virtual track train; fusion state estimation; interacting multiple model; joint model; Gaussian process regression; kinematic; particle filtering

check for updates

Citation: Lu, Z.; Wang, Z.; Luo, X. Advanced State Estimation for Multi-Articulated Virtual Track Trains: A Fusion Approach. *Machines* **2024**, *12*, 565. https://doi.org/10.3390/machines12080565

Academic Editors: Jing Zhao, Zhongchao Liang and Zhongguo Li

Received: 12 July 2024
Revised: 13 August 2024
Accepted: 15 August 2024
Published: 17 August 2024

Copyright: © 2024 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

The virtual track train (VTT) is an innovative urban public transport system that employs a rail transit management mode and rubber-tire running gears. Compared to rail systems, the construction and maintenance costs of VTT can be significantly reduced [1,2]. In comparison to bus rapid transit (BRT), the longer body of VTT can accommodate a larger number of passengers with a capacity comparable to that of trams. The implementation of all-axle-steering and all-wheel-driving technologies enables the vehicle to exhibit greater flexibility and a smaller swept width. Nevertheless, active traction and path-tracking control are essential. In recent years, there has been a notable increase in the attention paid to VTT, which has already established many business lines [3]. To control the vehicle, it is essential to have accurate information about the vehicle status, including the longitudinal and lateral velocities of each unit and the yaw rate. However, lateral velocity cannot be directly measured by sensors on production vehicles. Consequently, the estimation of this parameter is a key issue, with a significant impact on the effectiveness of the control strategy. In addition, the information obtained from sensors is subject to noise and requires filtering.

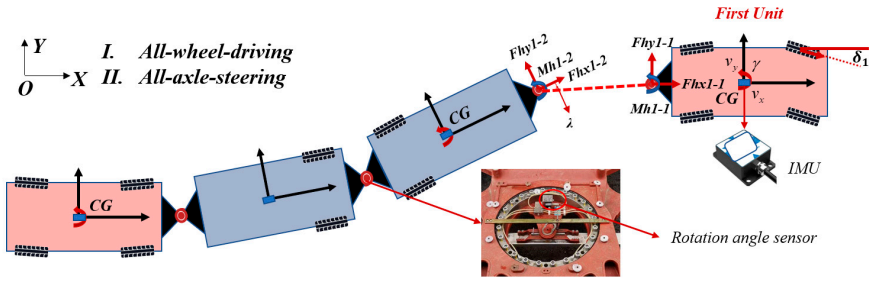
Current research on the estimation of multi-articulated vehicles, such as VTT, is limited. Most of the existing research focuses on articulated four-wheel vehicles or tractor-trailer combinations. The typical approach is to construct a model for the whole vehicle and integrate all sensors to achieve full state estimation [4,5]. However, this method is complex,

computationally inefficient, and lacks robustness. Furthermore, when the number of sensors is large, it becomes impractical to obtain the covariance matrix of the signal errors. Consequently, this method may be employed for structurally simple articulated vehicles but is not suitable for complex systems, such as VTT.

This paper proposes an alternative approach: the different units of the VTT are constrained by the articulation, as illustrated in Figure 1. The rotation angles of each articulation plate are accurately measured by sensors, and the acceleration information of each unit can be obtained through the Inertial Measurement Unit (IMU). Consequently, once the velocity information of the first unit has been obtained, the states of the remaining units can be determined by simple calculations. Therefore, the key to state estimation of the VTT is the first unit. However, the first unit is subject to the articulation force, which cannot be accurately obtained and represents a significant disturbance. Therefore, the state estimation of the VTT can be transformed into that of the first unit under strong interference or significant modelling errors.



(**a**)



(**b**)

Figure 1. 4-units 6-axles VTT and the problem of state estimation: (**a**) 4-units 6-axles VTT; (**b**) Transform the full-state estimation of VTT into the problem of the first unit under strong interference.

A multitude of studies have been conducted on the estimation of the state of a single vehicle. These include direct integration methods, state estimation methods based on kinematic models [6], dynamics methods [7], neural network methods [8], and methods based on GPS/INS [9]. Among these, those based on the kinematic model and the dynamics model are particularly widespread due to their simple observer construction and low cost. The kinematic estimation method is independent of tire forces and articulation forces, resulting in more effective estimation during transient phases. However, the input signals are obtained from the communication network and on-board sensors, which contain noise. This leads to steady-state errors that are difficult to eliminate.

The dynamics method, which is based on a vehicle dynamics model and modern observation technology, is an effective approach. In a previous study, Zhang et al. [10] considered the potential impact of measurement noise on the performance of the extended Kalman filter (EKF) and proposed a sliding window adaptive adjustment strategy. In a

further contribution to this field, Li et al. [11] devised a state estimation strategy based on the double EKF, which enables the real-time estimation of the vehicle side slip angle and yaw angle. Rodríguez et al. [12] employed the EKF to estimate the vehicle's parameters. Boada et al. [13] proposed a state estimation method based on the EKF with a pseudo-side slip angle. In consideration of the impact of coloured measurement noise, Cheng et al. [14] proposed an adaptive square-root curvature Kalman filter (SCKF) with an integral correction term. Cui et al. [15] conducted a study on the robust SCKF method. In addition to the Kalman filter (KF), the particle filter (PF) does not require the process noise of the system, and the observation results are more accurate when the system exhibits strong nonlinearity [16–18]. The dynamic method exhibits superior steady-state performance, yet its efficacy is contingent upon the precision of the underlying model. During the transient phase, however, the method is susceptible to certain inaccuracies.

To overcome the limitations of a single observer, numerous studies have proposed fusion methods based on multiple observations. Cheli [19] proposed a fuzzy logic fusion method to fuse the vehicle side slip angle obtained by direct integration and the results of the dynamic observer. Chen [20] put forth a fusion method for the kinematic approach and the dynamic method, to estimate the vehicle side slip angle. In addition to the kinematic and dynamic observers, Li [11] incorporated GPS data to enhance the accuracy of the estimation. Piyabongkarn [21] and Han [22] also proposed various fusion methods for kinematic and dynamic observer results. However, the observation results are significantly influenced by the accuracy of the tire and vehicle models. The Interactive Multiple Model (IMM) is also an effective fusion method. The core idea of IMM is to use different estimation models and, based on the current state of the system and measurement data, switch and fuse between these observers interactively to improve the accuracy and robustness of state estimation. This approach enhances the system's adaptability to dynamic changes in the target [23–26]. In particular, for the first unit of the VTT, which exhibits significant inter-unit articulation force interference, the establishment of an accurate model and the fusion of multiple observation methods are of paramount importance for state estimation.

The objective of this paper is to develop a reliable state estimation strategy for the first unit of the VTT, particularly in the context of strong inter-unit articulation forces. Consequently, a fusion state estimation strategy based on the Interacting Multiple Model (IMM) is proposed. Firstly, a kinematic approach is established based on the vehicle's kinematic model and the PF for the transient state estimation. In light of the significant steady-state error associated with the kinematic approach, a joint-model approach integrating the vehicle dynamics model and residual model is proposed. The dynamics model is unable to account for the inter-unit articulation force, which results in significant modelling error, particularly when traversing curves. Therefore, a residual model is trained utilising the Gaussian process regression (GPR) method to fit the error [27,28]. The joint-model observer is also constructed based on PF. Given the characteristics of both approaches in terms of dynamic and steady-state estimation accuracy, the IMM method is employed to integrate the two estimation results, thereby enhancing the overall accuracy across the full state range. Subsequently, the effectiveness of the proposed strategy was evaluated under different conditions on the hardware-in-the-loop (HIL) platform. The results demonstrate that the proposed state estimation strategy is capable of achieving high accuracy and it will only slightly reduce the effectiveness of the control strategy.

Section 2 establishes a joint model for the first unit of the VTT, including the dynamics model as the main model and the residual model based on GPR. Section 3 proposes a fusion state estimation strategy, including the kinematic model approach, the joint-model approach, the IMM-based fusion method and the state calculation of the subsequent units. Section 4 verifies the effect based on the HIL platform. Conclusions are given in Section 5.

2. Vehicle Joint Model

The joint model of the first unit consists of two parts: the dynamics model, which serves as the main model, and a residual model built using GPR to fit the errors of the dynamics model.

2.1. Main Model

First, a 3-degrees-of-freedom (DoF) dynamics model of the first unit is established as the main model to describe the main dynamic characteristics, shown in Figure 2. Without considering the influence of inter-unit articulation, the longitudinal, lateral and yaw equations of the first unit can be described as:

$$\dot{v}_x = \gamma v_y + \frac{F_x}{m}, \tag{1}$$

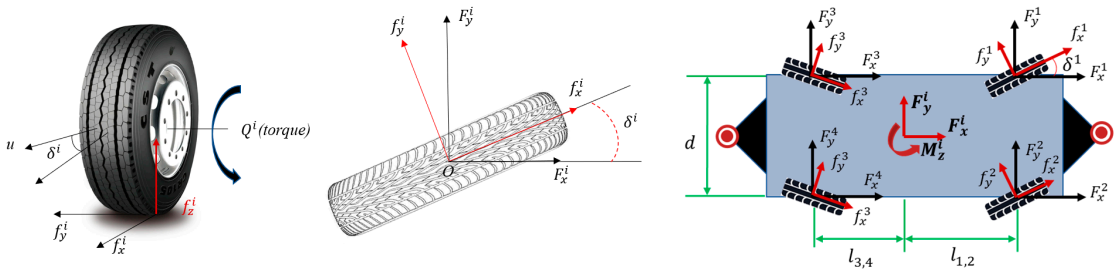
$$\dot{v}_y = -\gamma v_x + \frac{F_y}{m}, \tag{2}$$

$$\dot{\gamma} = \frac{M_z}{I_z}, \tag{3}$$

where v_x is the longitudinal velocity, v_y is the lateral velocity, γ is the yaw rate, m is the mass, and I_z is the yaw moment of inertia. F_x, F_y and M_z are the generalised longitudinal force, lateral force, and yaw moment acting at the centre of gravity (CG). The above equations can be expressed as:

$$\dot{x} = f_{dyn}(x, u), \tag{4}$$

$$y_{dyn} = h_{dyn}(x, u). \tag{5}$$



(**a**) (**b**)

Figure 2. Schematic diagram of tire forces and equivalent CG generalised forces: (**a**) Schematic diagram of tire forces; (**b**) Tire forces and equivalent CG generalised forces.

The state vector, measurement vector, and input vector are:

$$x = \begin{bmatrix} v_x & v_y & \gamma \end{bmatrix}^T, \tag{6}$$

$$y_{dyn} = \begin{bmatrix} a_x & a_y & v_x & \gamma \end{bmatrix}^T, \tag{7}$$

$$u = \begin{bmatrix} F_x & F_y & M_z \end{bmatrix}^T. \tag{8}$$

The state equation and measurement equation of the system are:

$$f_{dyn}(x, u) = \begin{bmatrix} \gamma v_y + \frac{F_x}{m} & -\gamma v_x + \frac{F_y}{m} & \frac{M_z}{I_z} \end{bmatrix}^T, \tag{9}$$

$$h_{dyn}(x, u) = \begin{bmatrix} a_x & a_y & v_x & \gamma \end{bmatrix}^T = \begin{bmatrix} \frac{F_x}{m} & \frac{F_y}{m} & v_x & \gamma \end{bmatrix}^T. \tag{10}$$

2.2. Generalised CG Forces and Tire Vertical Load Transfer

This section calculates the input to the vehicle model in Equation (8), which shows the generalised CG forces. By using the generalised CG forces as input, rather than specific

wheel steering angles and torques, the model can be applied to VTTs with different actuator configurations, such as all-wheel drive or central drive mode. Each wheel has at most two control inputs, namely wheel torque and steering angle, ignoring the influence of camber. As shown in Figure 2, the longitudinal force of the tire is generated by the wheel torque and the lateral force is generated by the steering angle. Therefore, the longitudinal and lateral forces of the tire can be expressed in the local coordinate system (CS) of each wheel:

$$f^i = \begin{bmatrix} f_x^i & f_y^i \end{bmatrix}^T, \tag{11}$$

where f_x^i is the longitudinal tire force in the tire CS, and f_y^i is the lateral tire force.

According to the Magic Formula (MF) tire model, the lateral and longitudinal tire forces in Equation (11) can be calculated:

$$f^i = -\mu \cdot D \cdot \sin\{C \cdot \arctan[B \cdot \alpha + E(B \cdot \alpha - \arctan(B \cdot \alpha))]\}, \tag{12}$$

where, f^i represents longitudinal force or lateral force. α represents longitudinal slip rate or sideslip angle, B is the stiffness factor, C is the shape factor, D is the peak factor, and E is the curvature factor. The above parameters are fitted from the measured tire force data [29,30]. The peak factor D is greatly affected by the tire vertical load. When the vehicle has longitudinal and lateral acceleration and yaw rate, the vertical load of the four tires will change, and the vertical load transfer can be expressed as follows:

$$f_z^1 = \left(\frac{l_{3,4}}{l_{1,2}+l_{3,4}} - \frac{ha_x}{g(l_{1,2}+l_{3,4})} - \frac{ha_y}{gd} - \frac{2hl_{3,4}v_x\gamma}{gd(l_{1,2}+l_{3,4})}\right)\frac{mg}{2}, \tag{13}$$

$$f_z^2 = \left(\frac{l_{3,4}}{l_{1,2}+l_{3,4}} - \frac{ha_x}{g(l_{1,2}+l_{3,4})} + \frac{ha_y}{gd} + \frac{2hl_{3,4}v_x\gamma}{gd(l_{1,2}+l_{3,4})}\right)\frac{mg}{2}, \tag{14}$$

$$f_z^3 = \left(\frac{l_{1,2}}{l_{1,2}+l_{3,4}} + \frac{ha_x}{g(l_{1,2}+l_{3,4})} - \frac{ha_y}{gd} + \frac{2hl_{1,2}v_x\gamma}{gd(l_{1,2}+l_{3,4})}\right)\frac{mg}{2}, \tag{15}$$

$$f_z^4 = \left(\frac{l_{1,2}}{l_{1,2}+l_{3,4}} + \frac{ha_x}{g(l_{1,2}+l_{3,4})} + \frac{ha_y}{gd} - \frac{2hl_{1,2}v_x\gamma}{gd(l_{1,2}+l_{3,4})}\right)\frac{mg}{2}. \tag{16}$$

By substituting the tire vertical load with the wheel control input and the vehicle state, the tire force can be obtained. Considering that the first unit of a VTT usually has four wheels, the (11) is extended:

$$f = \begin{bmatrix} (f^1)^T & (f^2)^T & (f^3)^T & (f^4)^T \end{bmatrix}^T, f \in R^{8\times 1}. \tag{17}$$

For the ith wheel, the tire force in the vehicle CS can be obtained through the cosine matrix:

$$\begin{bmatrix} F_x^i & F_y^i \end{bmatrix}^T = DCM^i \cdot f^i, \tag{18}$$

where F_x^i is the longitudinal tire force in the vehicle CS, F_y^i is the lateral force, and DCM^i is the directional cosine matrix of the ith wheel:

$$DCM^i = \begin{bmatrix} \cos\delta^i & -\sin\delta^i \\ \sin\delta^i & \cos\delta^i \end{bmatrix} \tag{19}$$

The expanded cosine matrix is represented as:

$$DCM = diag\left(\begin{bmatrix} DCM^1 & DCM^2 & DCM^2 & DCM^4 \end{bmatrix}\right), DCM \in R^{8\times 8}. \tag{20}$$

The tire force in the vehicle CS is given by:

$$\begin{bmatrix} F_x^1 & F_y^1 & F_x^2 & F_y^2 & F_x^3 & F_y^3 & F_x^4 & F_y^4 \end{bmatrix}^T = DCM \cdot f. \tag{21}$$

Defining L as a mapping matrix considering vehicle parameters,

$$L = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ -d/2 & l_{1,2} & d/2 & l_{1,2} & -d/2 & l_{3,4} & d/2 & l_{3,4} \end{bmatrix}, L \in R^{3\times 8}, \tag{22}$$

where, as shown in Figure 2, d is the track, $l_{1,2}$ is the longitudinal distance from the front axle to the CG, and $l_{3,4}$ is the longitudinal distance from the rear axle to the CG.

Define B_m as the Boolean matrix for actuator configuration,

$$B_m = diag\left(\begin{bmatrix} b_m^{1,x} & b_m^{1,y} & b_m^{2,x} & b_m^{2,y} & b_m^{3,x} & b_m^{3,y} & b_m^{4,x} & b_m^{4,y} \end{bmatrix}\right), \tag{23}$$

where $b_m^{i,x}$ and $b_m^{i,y}$ values of 0 or 1 represent whether the wheel has driving ability and whether the wheel exists, respectively. Therefore, the CG generalised forces model can be applied to VTT with different actuator configurations. The CG generalised force (8) for the first unit can be expressed as:

$$u = L \cdot B_m \cdot DCM \cdot f. \tag{24}$$

2.3. Residual Model

The above dynamics model has modelling errors and cannot take into account the effect of the articulation force, which even leads to model mismatch when small radius curves and large lateral accelerations occur. To solve this problem and improve the accuracy of the model in predicting the state of the first unit, a residual model based on GPR is established in this section, which is used to adjust the error of the dynamics model. The dynamic model serves as the main model and together with the residual model forms the joint model. The workflow of the joint model is shown in Figure 3.

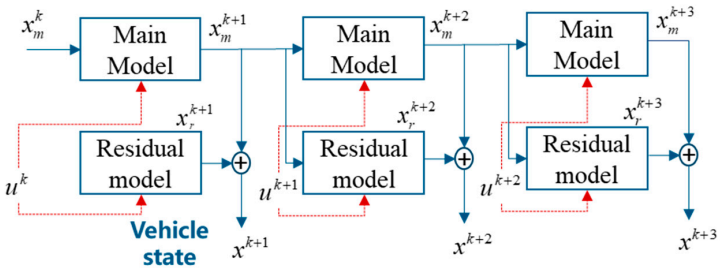


Figure 3. The workflow of the joint model.

The joint model of the first unit can be represented as:

$$x^{k+1} = F_{\text{Joint}}(x_m^k, u^k) = F_{dyn}(x_m^k, u^k) + \mathcal{L}(x_m^k, u^k). \tag{25}$$

Assume that both the state and the input vary within a bounded range, i.e., $x \in \mathcal{X}$, $u \in \mathcal{U}$, where $\mathcal{X}$ and $\mathcal{U}$ both represent spatial polyhedrons. Since the observed system is a real physical system, it is assumed that the unmodeled part $\mathcal{L}(x_m^k, u^k)$ is also bounded, and therefore always exists in a spatial polyhedron $\mathcal{W}$, i.e., $\mathcal{L}(x_m^k, u^k) \in \mathcal{W}$. By using a relatively stable main model state x_m^k as the prediction base point, the consistency between the main model and the residual model prediction base point is maintained, while avoiding the influence of the residual model on it.

F_{dyn} represents the modifiable part of the system, i.e., the main model, which is the discrete form of the dynamics model (9). For the unmodeled and model mismatched parts, the residual model is used:

$$x_r^{k+1} = \mathcal{L}(x_m^k, u^k), \tag{26}$$

where, $\mathcal{L}$ represents the residual model, which is a continuous function over the state vector and the input vector. The full system model is the sum of the main model and the residual

model. Using GPR to build the residual model, the input z of the model is the merged state and input vector.

$$z = \begin{bmatrix} (x_m)^T & (u)^T \end{bmatrix}^T, \ z \in R^{n_x+n_u}. \tag{27}$$

The Gaussian process can be determined by the mean and variance functions:

$$\mu(z) = E[\mathcal{L}(z)], \tag{28}$$

$$C(z, z') = E\big[(\mathcal{L}(z) - \mu(z))(\mathcal{L}(z') - \mu(z'))\big]. \tag{29}$$

z and z' represent two samples, and the Gaussian process can be represented as:

$$\mathcal{L}(z) \sim N(\mu(z), C(z, z')). \tag{30}$$

The Squared Exponential (SE) kernel function is selected:

$$C(z, z') = \sigma_f^2 \exp(-\frac{\|z - z'\|^2}{2l^2}), \tag{31}$$

where σ_f and l are the hyperparameters. Given a training set, $D = \{(z^i, Y^i) | \ i = 1, \ldots, n\} = \{Z, Y\}$, z^i is a given sample, and Z is the input matrix composed of all input variables. Y^i is the output variable corresponding to each input variable, and Y is the output vector composed of all output variables. Build a GPR model based on the training set:

$$Y = \mathcal{L}(Z) + \varepsilon, \varepsilon \sim N(0, \sigma_n^2). \tag{32}$$

Assuming ε is white noise with variance σ_n^2 and zero mean error, the prior distribution of the output vector Y can be obtained as:

$$Y \sim N(0, C(Z, Z) + \sigma_n^2 I), \tag{33}$$

where $C(Z, Z)$ is the covariance matrix, also known as the kernel matrix, and the elements in the matrix are calculated using the kernel function (31):

$$C(Z, Z) = \begin{bmatrix} C(z^1, z^1) & C(z^1, z^2) & \cdots & C(z^1, z^n) \\ C(z^2, z^1) & C(z^2, z^2) & \cdots & C(z^2, z^n) \\ \vdots & \vdots & \ddots & \vdots \\ C(z^n, z^1) & C(z^n, z^2) & \cdots & C(z^n, z^n) \end{bmatrix} \tag{34}$$

Assuming that the test sample, i.e., the input vector at a given moment, is z^k, and its corresponding random variable, i.e., the state of the first unit, is Y^k, it can be represented as a Gaussian distribution as follows:

$$z^k = \begin{bmatrix} (x_m^k)' & (u^k)' \end{bmatrix}', \tag{35}$$

$$Y^k \sim N(0, C(z^k, z^k)). \tag{36}$$

Thus, in GPR, the output values in the training set and the predicted values in the test samples follow a joint Gaussian distribution:

$$\begin{bmatrix} Y \\ Y^k \end{bmatrix} \sim N\left(0, \begin{bmatrix} C(Z, Z) + \sigma_n^2 I & C(Z, z^k) \\ C(z^k, Z) & C(z^k, z^k) \end{bmatrix}\right). \tag{37}$$

Therefore, the posterior distribution of the predicted values Y^k is:

$$Y^k | Y \sim N\left(\overline{Y}^k, \mathrm{cov}(Y^k)\right). \tag{38}$$

The mean can be written as:

$$\overline{Y}^k \triangleq C(z^k, Z)\left[C(Z,Z) + \sigma_n^2 I\right]^{-1} Y. \tag{39}$$

The variance is:

$$\mathrm{cov}(Y^k) = C\left(z^k, z^k\right) - C\left(z^k, Z\right)\left[C(Z,Z) + \sigma_n^2 I\right]^{-1} C\left(Z, z^k\right) \tag{40}$$

Then, the mean (39) is the output of the residual model:

$$\mathcal{L}(x_m^k, u^k) = C(z^k, Z)\left[C(Z,Z) + \sigma_n^2 I\right]^{-1} Y. \tag{41}$$

Figure 4 shows the vehicle state prediction using the joint model and the main model, including the longitudinal acceleration, lateral acceleration and yaw acceleration of the first unit. It can be seen that the prediction performance of the first unit using a joint model is significantly improved compared to a single main model, which effectively improves the accuracy of the system description and can be used for state estimation.

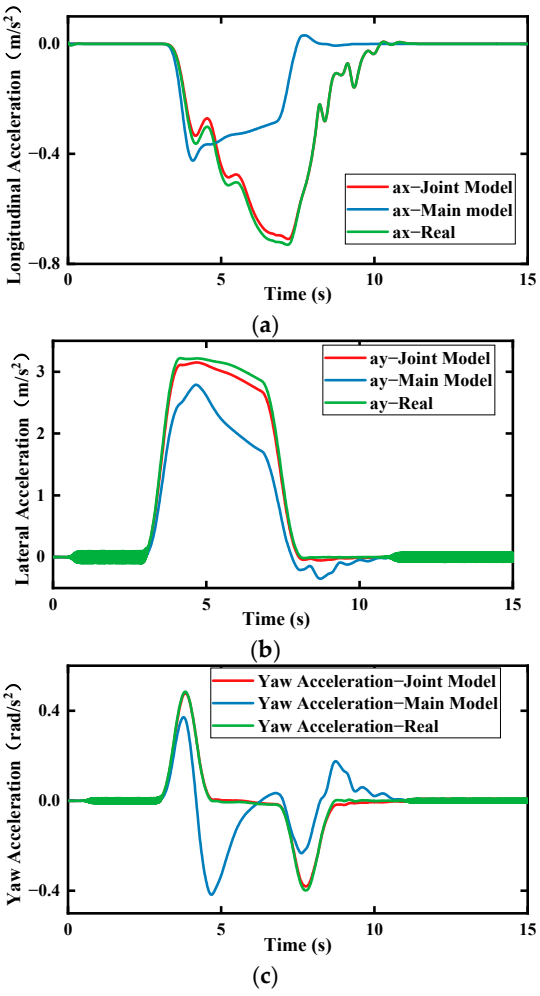


Figure 4. The effect of the first unit's state prediction using the joint model (R30m circular curve, v = 10 m/s): (**a**) Longitudinal acceleration; (**b**) Lateral acceleration; (**c**) Yaw acceleration.

3. The Fusion State Estimation Strategy

Figure 5 shows the structure of the proposed state estimation fusion strategy. The input signal is the sensor information of the first unit, including longitudinal acceleration a_x, lateral acceleration a_y, yaw rate γ, estimated vehicle speed v_x provided by the ECU,

wheel steering angle δ^i, and wheel torque Q^i. This strategy is mainly divided into three parts: kinematic approach, joint-model approach, and fusion method based on IMM.

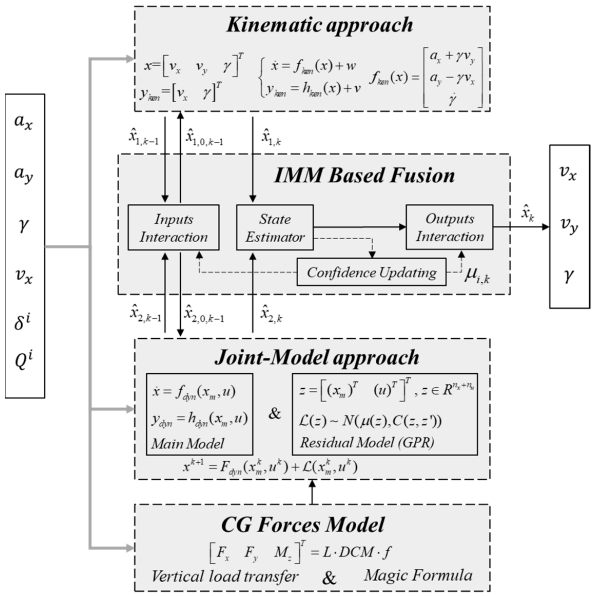


Figure 5. The structure of the proposed state estimation strategy.

The kinematic approach uses the kinematic model and PF to construct an observer for transient state changes and large magnitudes. The joint-model approach adopts the joint model and is also based on PF, which has a good steady-state effect. Based on the IMM method, the fusion of two estimation results can achieve high accuracy under different operating conditions.

3.1. Kinematic Approach

As mentioned above, kinematic methods have high robustness to vehicle parameters and tire forces and thus have good transient state tracking performance. PF is a widely used and effective filter for nonlinear systems. The 3-DoF kinematic model of the first unit can be constructed as follows:

$$\begin{cases} \dot{x} = f_{ken}(x) + W \\ y_{ken} = h_{ken}(x) + V \end{cases} \tag{42}$$

The measurement vector is:

$$y_{ken} = \begin{bmatrix} v_x & \gamma \end{bmatrix}^T \tag{43}$$

The nonlinear state equation $f_{ken}(x)$ is:

$$f_{ken}(x) = \begin{bmatrix} a_x + \gamma v_y \\ a_y - \gamma v_x \\ \dot{\gamma} \end{bmatrix} \tag{44}$$

The measurement equation is:

$$h_{ken}(x) = \begin{bmatrix} v_x & \gamma \end{bmatrix}^T \tag{45}$$

where the process noise W and the measurement noise V are represented by their error covariance matrices, represented by Q and R, respectively. The system state v_x, v_y and γ are updated at each step, and the results of the observer are corrected based on a posterior

value. The discretisation form of (42) can be obtained by applying the zero-order-hold method as follows:

$$\begin{cases} x^{k+1} = F_{ken}\left(x^{k+1}\right) + W^{k+1} \\ y_{ken}^{k+1} = H_{ken}(x^{k+1}) + V^{k+1} \end{cases} \tag{46}$$

where

$$F_{ken} = \begin{bmatrix} v_x^k + \left(a_x^k + \gamma^k v_y^k\right) T_s \\ v_y^k + \left(a_y^k - \gamma^k v_x^k\right) T_s \\ \gamma^k \end{bmatrix} \tag{47}$$

$$H_{ken} = \begin{bmatrix} v_x^k & \gamma^k \end{bmatrix}^T \tag{48}$$

T_s is the sampling time.

3.2. Joint-Model Approach

In the joint model of Section 2, the vehicle's main model is continuous and also needs to be discretised. The discrete state space obtained by using the zero-order-hold method is given by:

$$x_m^{k+1} = F_{dyn}(x_m^k, u^k) \tag{49}$$

$$y_{dyn}^k = H_{dyn}(x_m^k, u^k) \tag{50}$$

The state equation and measurement equations of the system can be expressed as:

$$F_{dyn}(x_m^k, u^k) = \begin{bmatrix} v_x^k + \frac{T_s \cdot F_x^k}{m} + T_s \cdot v_y^k \cdot \gamma^k \\ v_y^k + \frac{T_s \cdot F_y^k}{m} - T_s \cdot v_x^k \cdot \gamma^k \\ \gamma^k + \frac{T_s \cdot M_z^k}{I_z} \end{bmatrix} \tag{51}$$

$$H_{dyn}(x_m^k, u^k) = \begin{bmatrix} \frac{F_x^k}{m} & \frac{F_y^k}{m} & v_x^k & \gamma^k \end{bmatrix}^T \tag{52}$$

The residual model is already discrete, and the discretised main model and the residual model form the joint model. Based on the joint model in Section 2.3 and the vehicle kinematic model in Section 3.1, the PF method is used to construct the observer [16–18]. The main steps of PF are summarised in Algorithm 1, and the number of particles used in this paper is 300.

Algorithm 1: Particle Filter

Step 1: Initialise the filter
$k = 0$, extract N particles from a prior distribution $p(x_0)$, $\{x_i^0\}_1^N$, $\overline{x}_i^0 = E\left[x_i^0\right]$,
$P_i^0 = E\left[(x_i^0 - \overline{x}_i^0)(x_i^0 - \overline{x}_i^0)^T\right]$
Step 2: Sampling
$k = k + 1$, generate new particles: $x_i^k \sim p(x^k |\, x_i^{k-1})$
Step 3: Particle weight calculation
Particle weight: $w_i^k \propto w_i^{k-1} p(y_k |\, \hat{x}_i^k)$
Particle weight normalisation: $\widetilde{w}_i^k = \frac{w_i^k}{\sum_{j=1}^{N} w_i^k}$
Step 4: Resampling
The weights of each particle after resampling are the same: $w_i^k = \widetilde{w}_i^k = \frac{1}{N}$
Step 5: Output Calculation
State estimation result: $\hat{x}^k = \overline{x}^k = E(x^k |\, y^k) = \sum_{i=1}^{N} w_i^k \cdot x_i^k$
The state covariance matrix: $P^k = E((x^k - \overline{x}^k)(x^k - \overline{x}^k)^T) = \sum_{i=1}^{N} w_i^k (x_i^k - \overline{x}^k)(x_i^k - \overline{x}^k)^T$
Repeat steps 2 to 5.

3.3. IMM Based Fusion Method

To improve the accuracy of the first unit state estimation and to effectively combine the advantages of the methods proposed in the previous two sections, a fusion strategy based on the IMM method is proposed. There are two models in the IMM algorithm in this paper, which can be represented by the following state equation:

$$x_i^{k+1} = F_i(x_i^k) + W_i^k, \ \ i = 1, 2 \tag{53}$$

where x_i^k represents the unit state, $F_i(x_i^k)$ representing the kinematic model F_{ken} and the joint model F_{Joint}.

The measurement equation is:

$$y_i^k = H_i(x_i^k) + V_i^k, \ \ i = 1, 2 \tag{54}$$

The Markov probability transition matrix of the models can be expressed as:

$$P = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \tag{55}$$

where p_{ij} represents the transition probability from the i model to the j model. The flowchart of the IMM method is shown in Figure 6.

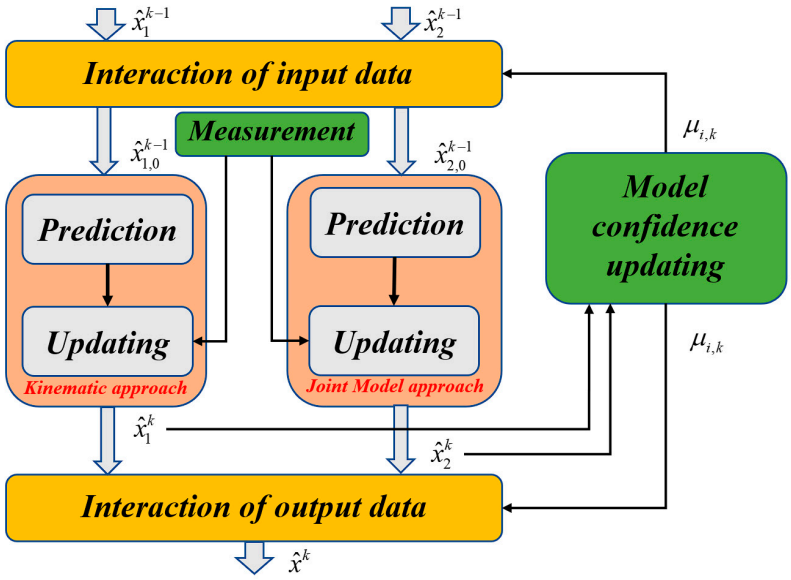


Figure 6. Flowchart of the IMM algorithm.

Step 1: Input interaction

First, assuming that at step $k-1$, each model obtains an optimal estimate of the unit state $\hat{x}_i^{k-1}$ and the estimated covariance matrix $\hat{P}_i^{k-1}$, the estimation results and covariance matrix of all models are fused as the inputs of the step k:

$$\hat{x}_{j,0}^{k-1} = \sum_{i=1}^{n} \lambda_{ij}^{k-1} \cdot \hat{x}_i^{k-1} \tag{56}$$

$$\hat{P}_{j,0}^{k-1} = \sum_{i=1}^{n} \lambda_{ij}^{k-1} \left[\hat{P}_i^{k-1} + (\hat{x}_{j,0}^{k-1} - \hat{x}_i^{k-1})(\hat{x}_{j,0}^{k-1} - \hat{x}_i^{k-1})^T \right] \tag{57}$$

where λ_{ij}^{k-1} is the fusion state transition coefficient. According to the defined probability transition matrix, the ith row in the matrix (55) represents the correlation between all

models and the ith model. After multiplying the values in the transition matrix by the credibility μ_i^{k-1}, normalisation is performed to obtain the fused state transition coefficient:

$$\lambda_{ij}^{k-1} = \frac{p_{ij}\mu_i^{k-1}}{\sum_{i=1}^{r} p_{ij}\mu_i^{k-1}} \tag{58}$$

Step 2: Filter-based estimation

Based on the aforementioned kinematic approach and joint-model approach, unit state estimation results $\hat{x}_i^k$ based on different models, as well as the covariance matrix of the states $\hat{P}_i^k$ are obtained.

Step 3: Model confidence update

In the IMM algorithm, the maximum likelihood estimation method is used for model confidence updating, which determines the proportion of the estimation results of the current approach by calculating the similarity between the estimation results and the measurement results. The maximum likelihood function based on the ith model in step k is:

$$\Lambda_i^k = \frac{1}{\sqrt{(2\pi)^N |S_i^k|}} \exp\left(-\frac{1}{2}\left(r_i^k\right)^T \left(S_i^k\right)^{-1} \left(r_i^k\right)\right) \tag{59}$$

where r_i^k and S_i^k are measurement errors and covariance matrices of the errors, respectively.

$$r_i^k = y^k - H_i(\hat{x}_i^k) \tag{60}$$

$$S_i^k = (H_i^k)\hat{P}_i^k (H_i^k)^T + R_i^k \tag{61}$$

Based on the likelihood value of each approach in step k, the estimation accuracy of each method can be obtained, thereby achieving the update of the credibility:

$$\mu_i^k = \frac{\left(\mu_i^{k-1}\right)^- \cdot \Lambda_i^k}{c} \tag{62}$$

where $\left(\mu_i^{k-1}\right)^-$ is the fused credibility value and c is the normalisation coefficient.

$$\left(\mu_i^{k-1}\right)^- = \sum_{j=1}^{2} p_{i,j}\mu_i^{k-1} \tag{63}$$

$$c = \sum_{i=1}^{n} \left(\mu_i^{k-1}\right)^- \Lambda_i^k \tag{64}$$

Step 4: Output interaction

Finally, based on the updated credibility, the fusion of the results of all the approaches is performed. The output of the IMM method, i.e., the state and covariance matrix of the first unit, can be obtained:

$$\hat{x}^k = \sum_{i=1}^{n} \mu_i^k \hat{x}_i^k \tag{65}$$

$$\hat{P}^k = \sum_{i=1}^{n} \mu_i^k \left[\hat{P}_i^k + (\hat{x}^k - \hat{x}_i^k)(\hat{x}^k - \hat{x}_i^k)^T\right] \tag{66}$$

The main steps of IMM are summarised in Algorithm 2:

Algorithm 2: IMM

Step 1: Input interaction

$$\lambda_{ij}^{k-1} = \frac{p_{ij}\mu_i^{k-1}}{\sum_{i=1}^{r} p_{ij}\mu_i^{k-1}},\ \ i,j = 1,2$$

$$\hat{x}_{j,0}^{k-1} = \sum_{i=1}^{n} \lambda_{ij}^{k-1} \cdot \hat{x}_i^{k-1},\ \ i,j = 1,2$$

$$\hat{P}_{j,0}^{k-1} = \sum_{i=1}^{n} \lambda_{ij}^{k-1} \left[\hat{P}_i^{k-1} + (\hat{x}_{j,0}^{k-1} - \hat{x}_i^{k-1})(\hat{x}_{j,0}^{k-1} - \hat{x}_i^{k-1})^T\right],\ \ i,j = 1,2$$

Step 2: Filter-based estimation

Based on $\hat{x}_{j,0}^{k-1}$, different approaches are used to obtain the unit state $\hat{x}_i^k$ and the covariance matrix $\hat{P}_i^k$

Step 3: Model confidence update

Maximum likelihood function: $\Lambda_i^k = \frac{1}{\sqrt{(2\pi)^N |S_i^k|}} \exp\left(-\frac{1}{2}\left(r_i^k\right)^T \left(S_i^k\right)^{-1} \left(r_i^k\right)\right),\ \ i = 1,2$

$$r_i^k = y^k - H_i(\hat{x}_i^k),\ \ i = 1,2$$

$$S_i^k = (H_i^k)\hat{P}_i^k (H_i^k)^T + R_i^k,\ \ i = 1,2$$

Credibility update:

$$c = \sum_{i=1}^{n} \left(\mu_i^{k-1}\right)^{-} \Lambda_i^k,\ \ i = 1,2$$

$$\left(\mu_i^{k-1}\right)^{-} = \sum_{j=1}^{2} p_{i,j}\mu_i^{k-1},\ \ i,j = 1,2$$

$$\mu_i^k = \frac{\left(\mu_i^{k-1}\right)^{-} \cdot \Lambda_i^k}{c},\ \ i = 1,2$$

Step 4: Output interaction

First unit's state and covariance matrix:

$$\hat{x}^k = \sum_{i=1}^{n} \mu_i^k \hat{x}_i^k$$

$$\hat{P}^k = \sum_{i=1}^{n} \mu_i^k \left[\hat{P}_i^k + (\hat{x}^k - \hat{x}_i^k)(\hat{x}^k - \hat{x}_i^k)^T\right]$$

$k = k + 1$, repeat steps 1 to 4.

3.4. State Calculation of Subsequent Units

After obtaining the longitudinal and lateral velocities and the yaw rate of the first unit, the states of the subsequent units are calculated using the angle sensor signals of the articulation plates and the yaw rate signals of every unit. These results provide the required inputs for the control strategy [31]. The adjacent units of the VTT are constrained by articulation plates, as shown in Figure 7. The velocities of the subsequent vehicles can be calculated using the following formula [31]:

$$v_x^{i+1} = v_{x,h}^{i+1} = v_x^i \cos\lambda^i - (v_y^i - l_{h,r}^i \gamma^i) \sin\lambda^i \tag{67}$$

$$v_y^{i+1} = v_{y,h}^{i+1} - l_{h,f}^{i+1}\gamma^{i+1} = v_x^i \sin\lambda^i + (v_y^i - l_{h,r}^i \gamma^i)\cos\lambda^i - l_{h,f}^{i+1}\gamma^{i+1} \tag{68}$$

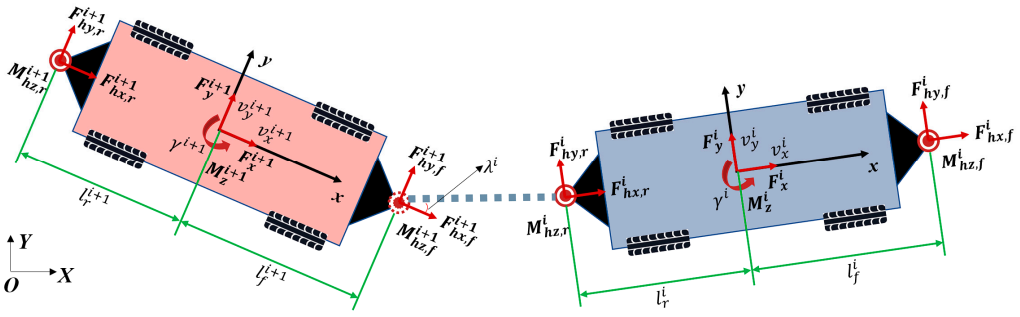


Figure 7. Inter-unit articulation constraints.

Due to the signal noise in the yaw rates and the articulation angles, this paper employs a one-dimensional Kalman Filter (KF) to filter these signals. The KF method has been widely applied and will not be elaborated upon here.

4. HIL Real-Time Simulation Result

4.1. HIL Platform

The proposed state estimation strategy is verified using a HIL simulation platform. The platform mainly includes five parts: real-time simulation host PC, controller, analogue sensor signal generation, driver control input, and simulation scenario, as shown in Figure 8. In the host PC, the dynamics model of a 4-unit 6-axle VTT is established using SIMPACK Linux. The model consists of four modules, with two independent steering running gears for the first and fourth modules and one independent steering running gear for the middle two modules. The first and sixth axles can provide driving torque. The running gear comprises an axle, two wheels, and a steering mechanism. The steering mechanism can be simplified as a four-bar linkage. Furthermore, the tire adopts a magic formula empirical model. The real-time module of SIMPACK Linux can realise the synchronisation of simulation time and real-time, to achieve the effect of simulating the actual operation of the VTT and to ensure the real-time performance of the platform.

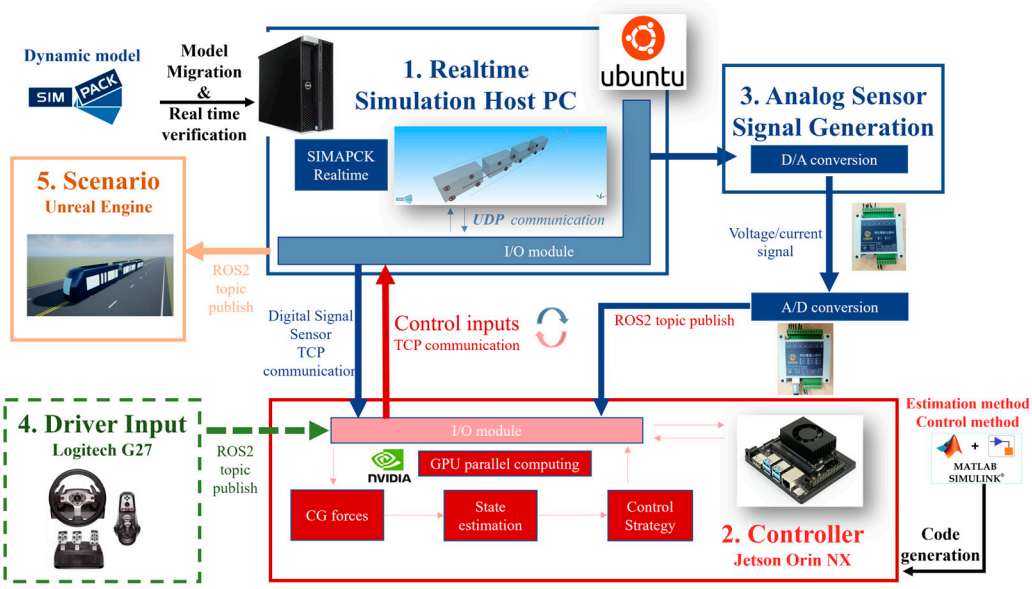


Figure 8. The HIL simulation platform architecture.

For VTT, active control is required at all times, otherwise, there may be rearward amplification, potentially leading to instability. Therefore, the effectiveness of its state estimation strategy cannot be independently verified and must be evaluated in conjunction with the control strategy. Consequently, all simulation analyses in this section are based on the observed VTT state under the control strategy proposed in reference [32]. The state estimation strategy and the control strategy are implemented on the NVIDIA Jetson Orin NX module. Communication between the controller and the real-time simulation PC is via TCP/IP, retaining Controller Area Network (CAN) functionality to accommodate different network formats. Upon receiving output information from the dynamics model and sensor module, the controller first performs state estimation and then calculates wheel steering angles and torque based on the estimated state, which is then output to the real-time simulation model. The frequency for state estimation and vehicle control are both 100 Hz.

As the on-board articulation angle sensor generates an analogue signal, the HIL platform also includes a module for the sensor voltage signals. The remaining sensor signals are digital and communicate via TCP/IP. The HIL platform also includes the driver

control input and simulation scenario modules, which communicate between the real-time simulation PC and the ECU via the ROS2 system.

4.2. Simulation Conditions

Lane changing is a typical scenario for verifying VTT performance and is also the most common vehicle manoeuvre in the real world. The typical lane change is shown in Figure 9, where X represents the direction of travel of the VTT and Y is the direction of the adjacent lane. If the leading vehicle is travelling at too low a speed, or if the current vehicle needs to change lanes due to route design, it should be done by traversing two curves with opposite curvature.



Figure 9. Lane change of VTT.

In addition, circular curves are also common in operation. Currently, the speed range for VTTs is 0–70 km/h, with a minimum turning radius of 15 m. Therefore, to effectively simulate real VTT operation, the following three conditions are selected:

Scenario 1: Minimum radius curve. The VTT travels through a circular curve with a radius of 15 m at a speed of 10 km/h. There are no transition curves between straight and curved sections and the path has no superelevation. This condition effectively reflects the state of the VTT when navigating extreme radius curves under approximate steady-state conditions.

Scenario 2: Large radius curve. The VTT travels at a speed of 40 km/h through a circular curve with a radius of 50 m. There are no transition curves between straight and curved sections and the track has no superelevation.

Scenario 3: Continuous curve. The VTT travels along a continuous curve that includes a lane change, a large radius curve, a small radius curve and a straight line. This trajectory includes different operating conditions encountered during operation and effectively reflects the overall condition of the VTT. Furthermore, the variation of curve radii reflects the adaptability of the state estimation strategy as the VTT's state changes continuously.

4.3. Minimum Radius Curve Simulation

The VTT is travelling on a dry asphalt road surface with a coefficient of friction of 0.85. This speed and curve radius represent the VTT navigating extreme radius curves under approximate steady-state conditions. The reference path is shown in Figure 10 and the input signals for state estimation and the estimation results are shown in Figures 10 and 11, respectively.

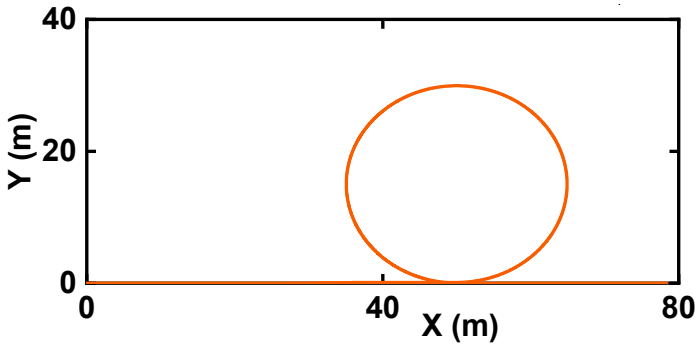


Figure 10. Reference path of minimum radius curve operating.

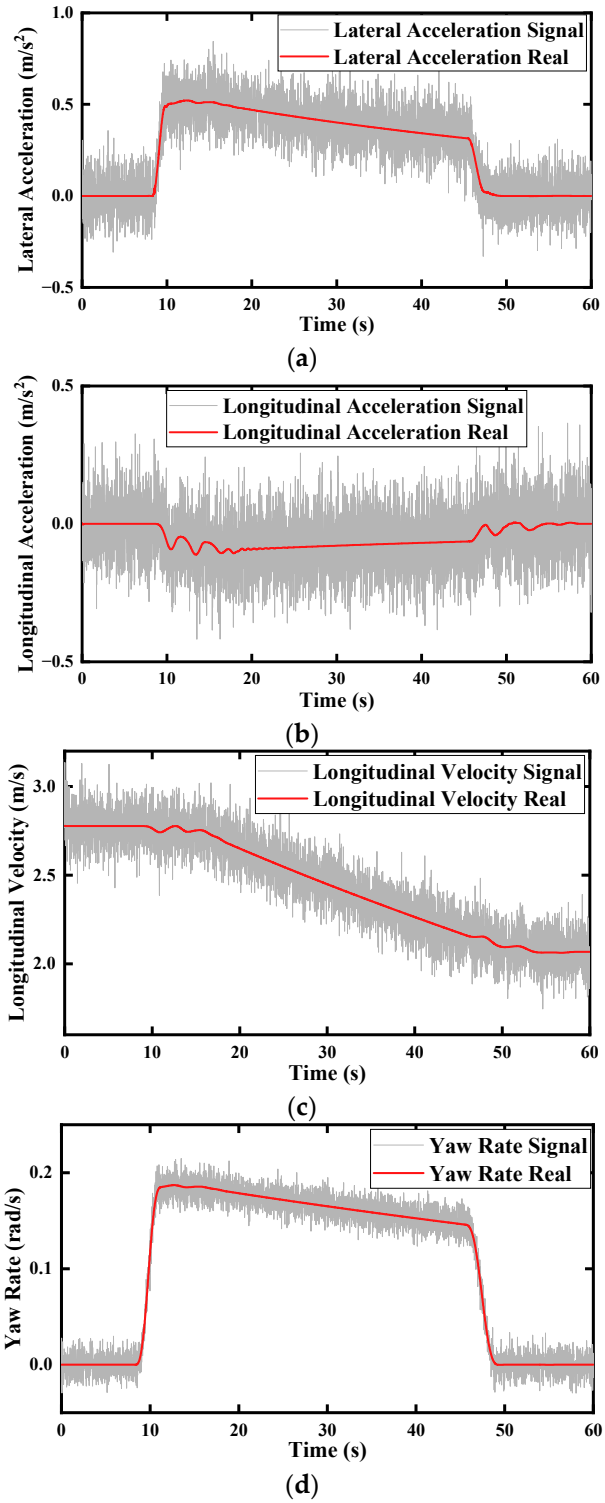


Figure 11. Input signals of the estimator: (**a**) Lateral acceleration; (**b**) Longitudinal acceleration; (**c**) Vehicle speed provided by the ECU; (**d**) Yaw rate.

The comparison between the actual lateral acceleration of the first unit and the observer's input signal is shown in Figure 11, and the peak lateral acceleration is around 0.5 m/s^2. The lateral acceleration first increases, then slowly decreases and finally returns to zero. The slow decrease in lateral acceleration is due to the slow decrease in speed caused by the driving resistance after the initial speed of the VTT.

The result of the lateral velocity and longitudinal velocity estimation under this condition is shown in Figure 12a,b. It can be observed that, compared with the actual values, the kinematic approach has smaller errors in the curve segment, but there is a larger error in the straight line, and the maximum error in lateral velocity is 0.05 m/s, with an average of 0.02 m/s. The joint-model approach has higher accuracy in the straight line, with a relatively larger error in the curved section, and the maximum error in lateral velocity is 0.02 m/s, with an average of 0.01 m/s. However, due to the low lateral acceleration, the vehicle is in an approximate steady state and the overall error of the joint-model method is not significant. The final IMM fusion method also combines the advantages of both, with higher accuracy over the whole range, the average lateral velocity error is 0.01 m/s, and the average longitudinal velocity error is 0.003 m/s. The statistical data of the lateral velocity estimation results are shown in Table 1. As shown in Figure 12c, it can be seen that the joint-model approach produces certain errors when entering the curve in the estimation of yaw rate, while the method based on the kinematic model always performs well, and the final result is also close to the results of the kinematic model method, with a maximum yaw rate estimation error of 0.01 rad/s.

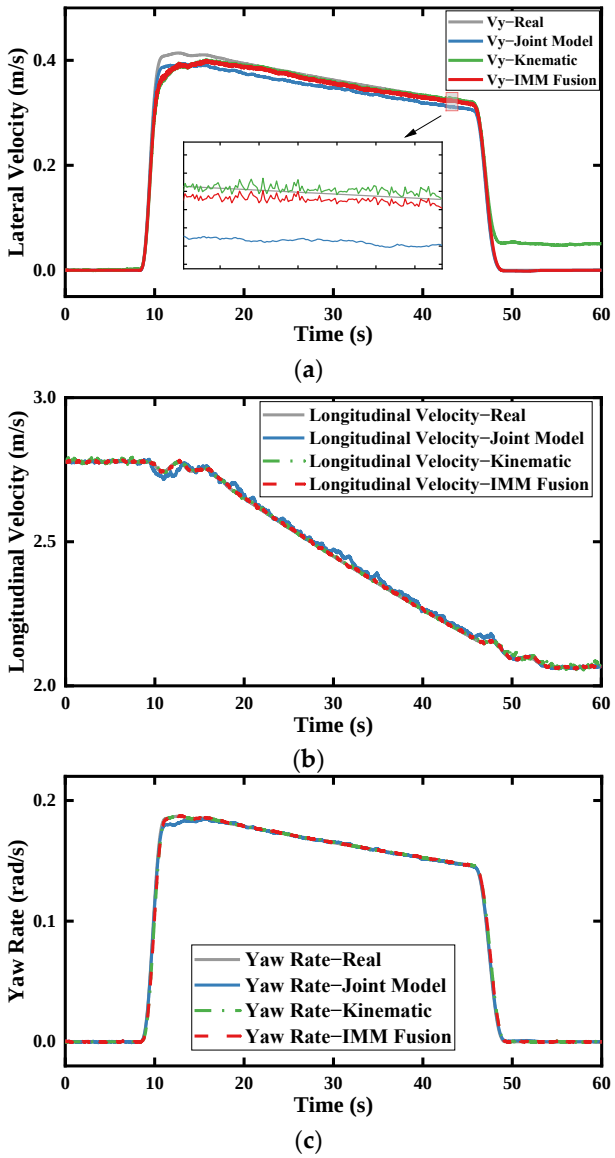


Figure 12. State estimation results of passing the minimum radius curve at a speed of 10 km/h: (**a**) Lateral speed; (**b**) Longitudinal speed; (**c**) Yaw rate.

Table 1. Statistical results of estimation error of lateral speed through R15m curve at 10 km/h.

Approach	The Absolute Error of Lateral Speed Estimation (m/s)		
	Average	Standard Deviation	Maximum
Kinematic approach	0.02	0.02	0.05
Joint-model approach	0.01	0.01	0.02
IMM fusion	0.01	0.01	0.05

As shown in Figure 13, the calculated longitudinal and lateral velocities of the second unit are compared with the true values. It can be observed that the calculated longitudinal velocity closely matches the true value, while the calculated lateral velocity shows a slight deviation, with a maximum error of only 0.03 m/s.

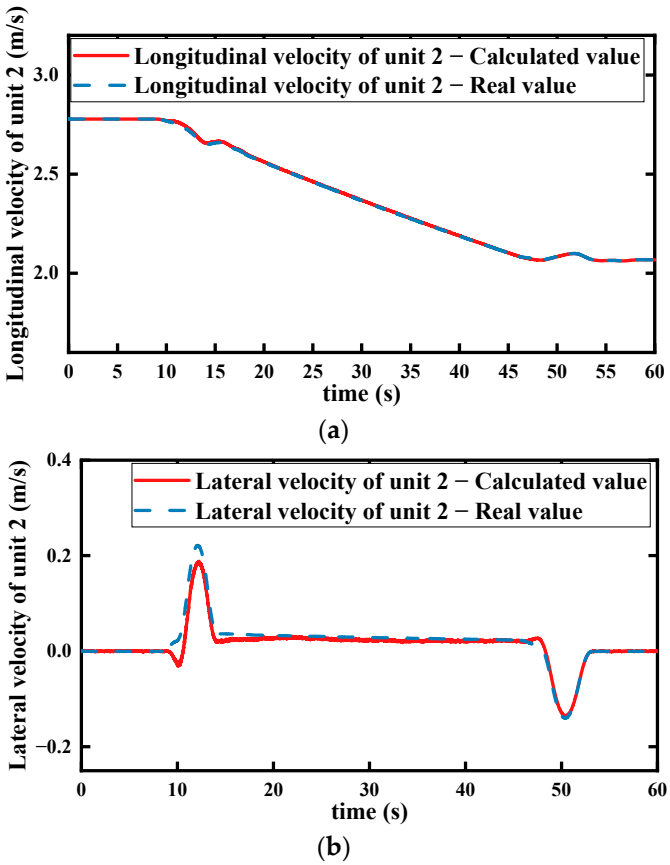


Figure 13. Calculated longitudinal and lateral velocities of the second unit: (**a**) Longitudinal velocity of the second unit; (**b**) Lateral velocity of the second unit.

4.4. Large Radius Curve Simulation

To further verify the effectiveness of the proposed strategy in large radius curves, the VTT was allowed to negotiate a circular curve with a radius of 50 m at a speed of 40 km/h. The reference path is shown in Figure 14 and the vehicle was also driven on a dry asphalt road with a road friction coefficient of 0.85. Compare and analyse the results of the real values with the kinematic method, the joint-model approach and the IMM-based fusion method.

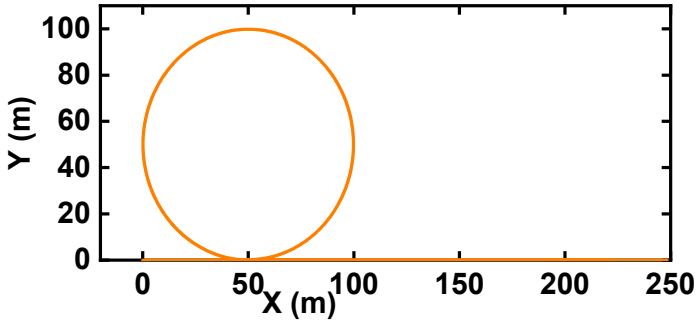


Figure 14. Reference path for large radius (R50m) curve operating.

The lateral velocity estimation results are shown in Figure 15, and Table 2 shows the statistical data of the estimation errors. It can be seen that, compared with the real value, the lateral acceleration also increases as the VTT speed increases. Under high lateral acceleration conditions, the effect of the joint-model method always has a large error in the curve section, while the error of the kinematic method is small, but there is always a certain error in the straight line that cannot be eliminated. The final IMM fusion combines the advantages of both and maintains a high estimation accuracy, and the average lateral velocity estimation error is 0.01 m/s.

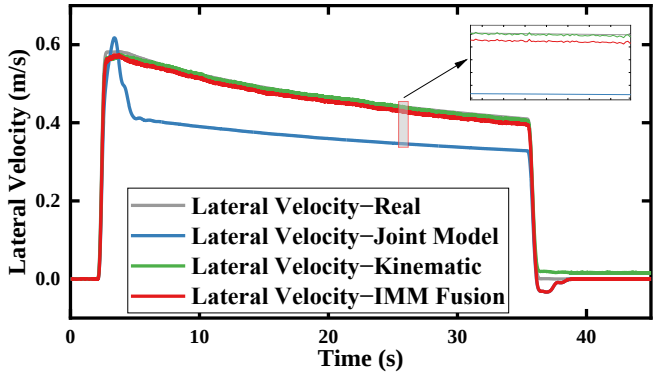


Figure 15. Lateral velocity estimation results through the R50m curve at 40 km/h.

Table 2. Statistical results of estimation error of lateral speed through R50m curve at 40 km/h.

Approach	The Absolute Error of Lateral Speed Estimation (m/s)		
	Average	Standard Deviation	Maximum
Kinematic approach	0.01	0.01	0.03
Joint-model approach	0.06	0.05	0.16
IMM fusion	0.01	0.01	0.05

4.5. Continuous Curve Simulation

To verify the effect of estimation on continuous curve operation, the VTT is controlled to pass the designed curve at a speed of 30 km/h. The curve includes a lane change, a large radius curve, a small radius curve and a straight section, which can fully reflect the frequent operating conditions of the VTT in real operation. The designed continuous curve is shown in Figure 16.

The results of the vehicle lateral speed are shown in Figure 17, and Table 3 shows the statistical data of the estimation errors. Compared with the real value, as the curvature of the reference path changes, the lateral acceleration of the VTT also changes. Under this dynamic condition, the estimation effect of the kinematic method is closer to the real value. The error is small in the curved section, but there is always some error in the straight section

that cannot be eliminated, as mentioned above. The joint-model method, on the other hand, has certain errors in several situations, such as the input of curves and the curve segments, but always performs well in a straight line. The final IMM fusion combines the advantages of both, and the average lateral velocity estimation error is 0.02 m/s.

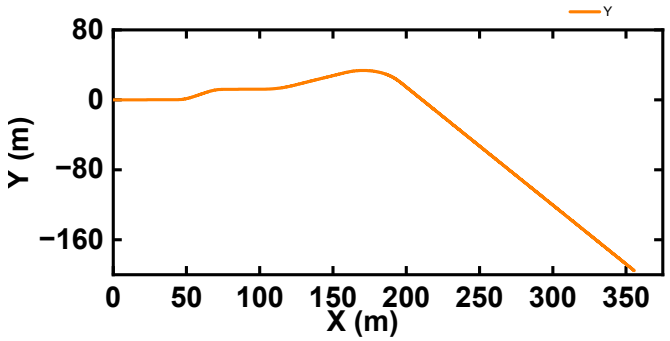


Figure 16. Reference path of the continuous curve passing.

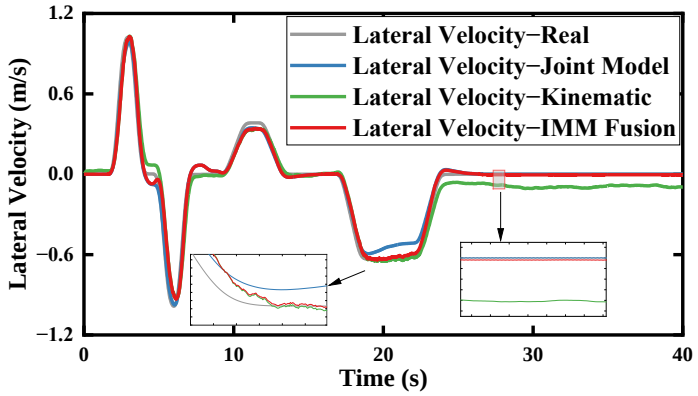


Figure 17. Lateral velocity estimation results through the continuous curve at 30 km/h.

Table 3. Statistical results of the estimation error of lateral speed through continuous curve at 30 km/h.

Approach	The Absolute Error of Lateral Speed Estimation (m/s)		
	Average	Standard Deviation	Maximum
Kinematic approach	0.07	0.04	0.23
Joint-model approach	0.02	0.03	0.10
IMM fusion	0.02	0.03	0.22

4.6. The Impact of State Estimation Errors on Control Performance

By integrating the proposed state estimation strategy with the control strategy from previous research [32], a complete closed-loop system is formed. The state estimation results are used as inputs for the control strategy to analyze the impact of the estimation error on VTT control performance.

The VTT is controlled to travel at a speed of 40 km/h (11.11 m/s) through a circular curve with a radius of 50 m. Figure 18 illustrates the lateral tracking error. The results indicate that the addition of the estimation strategy leads to a certain increase in the lateral tracking error of each unit. The first unit shows the most significant increase, with the maximum tracking error increasing by 6.2% compared to using the actual VTT state. The maximum tracking errors for the second, third, and fourth units increased by 3.1%, 5.2%, and 4.1%, respectively. Although the errors for each unit increased by 3–6%, the absolute increase is only 13 mm, which is also in the millimeter range. This validates the effectiveness of the combined condition monitoring and control strategy.

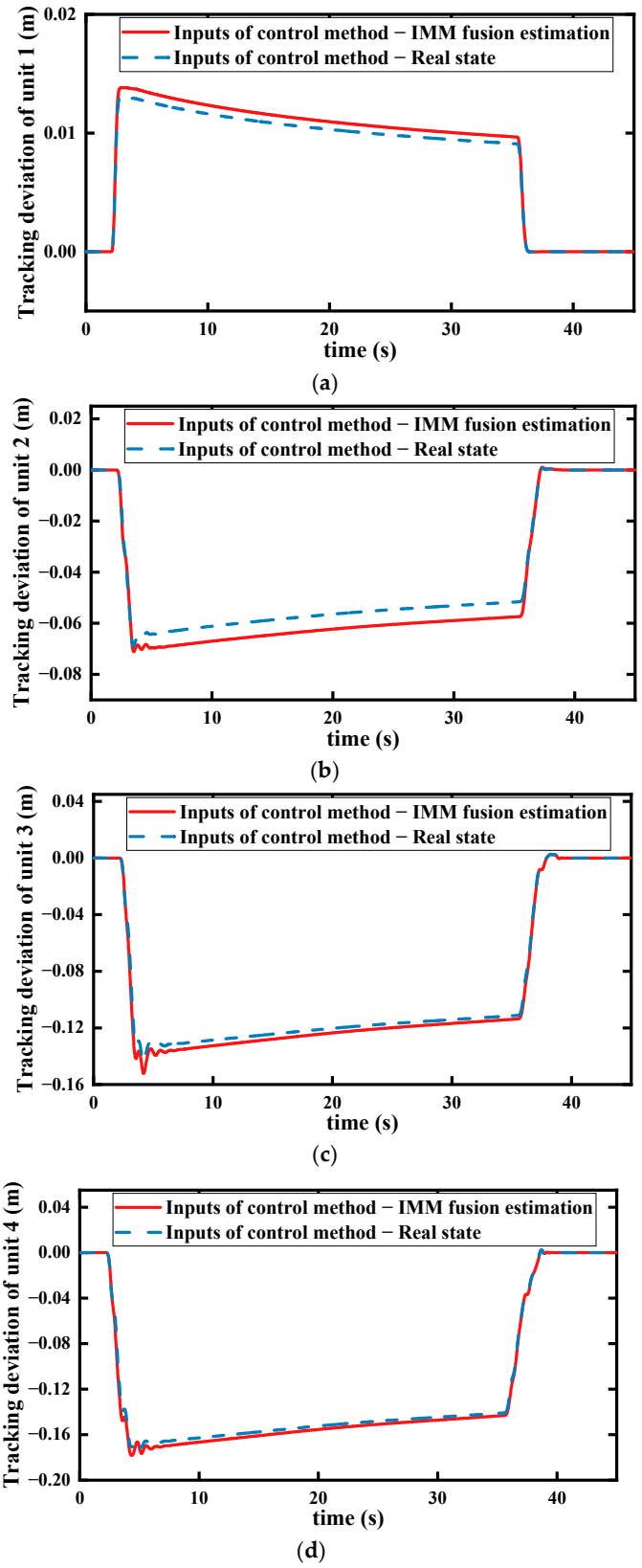


Figure 18. The impact of estimation error on the control performance: (**a**) Lateral tracking deviation of the 1st unit; (**b**) Lateral tracking deviation of the 2nd unit; (**c**) Lateral tracking deviation of the 3rd unit; (**d**) Lateral tracking deviation of the 4th unit.

5. Conclusions

This paper deals with the problem of state estimation of VTT with multi-articulation. The estimation methods based on the kinematic model and the joint model are investigated. The joint model includes the dynamics model as the main model and its residual model, which is established based on GPR. Furthermore, an IMM-based fusion method is proposed, which achieves high estimation accuracy under different conditions. The proposed state estimation strategy is suitable for practical application using production VTT's on-board sensors. The specific research and conclusions are as follows:

(1) The full state estimation for VTT is transformed into the estimation of the first unit under strong disturbances. The states of the subsequent units can be calculated based on the state of the first unit, the filtered articulation angles, and the yaw rates, and the kinematic approach based on the kinematic model of the first unit and PF is designed. This method can accurately track the transient state and is insensitive to vehicle structure and parameters, although it has steady-state errors.

(2) The vertical load transfer of the tire is investigated and the dynamics model of the first unit is established. Considering the modelling errors caused by nonlinearity and articulation forces, a residual model based on GPR is proposed. Based on this, a joint-model estimation method is proposed, which significantly improves the steady-state accuracy.

(3) A fusion strategy based on IMM is implemented, which combines the advantages of the above two methods, resulting in higher accuracy over the entire state range. The effect is validated on a HIL simulation platform by combining the state estimation strategy and the developed control strategy. Various simulation conditions, including minimum radius curve, large radius curve and continuous curve, have been set up. The results show that, under small lateral acceleration (approximately steady state), the joint-model approach exhibits higher stability and accuracy, while the kinematic method is superior in tracking the transient states. The fusion-based strategy can achieve accurate estimation results even under variable conditions. The average error in lateral velocity estimation does not exceed 0.02 m/s, and the maximum estimation error does not exceed 0.22 m/s. With the addition of state estimation, the VTT's tracking error increases by 3–6%, with the absolute value of the error increase being at the millimeter level.

In conclusion, the proposed fusion estimation strategy provides a new approach and implementation method for the state estimation of VTT. The future research work of this study will focus on the verification and optimization of the proposed estimation strategy with the laboratory scale model and actual operating vehicle.

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

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the author, Zehan Wang, upon reasonable request.

Acknowledgments: The authors are grateful for helpful comments from referees to improve this manuscript.

Conflicts of Interest: Author Xianguang Luo was employed by the company CRRC Zhuzhou Locomotive Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Wang, K.; Ma, S.; Chen, J.; Ren, F.; Lu, J. Approaches Challenges and Applications for Deep Visual Odometry Toward to Complicated and Emerging Areas. *IEEE Trans. Cogn. Dev. Syst.* **2020**, *14*, 35–49. [CrossRef]
2. Yan, L. Development and Application of the Maglev Transportation System. *IEEE Trans. Appl. Supercond.* **2008**, *18*, 92–99. [CrossRef]
3. Feng, J.; Hu, Y.; Yuan, X.; Huang, R.; Zhang, X.; Xiao, L. Development and validation of an automatic all-wheel steering system for multiple-articulated rubber-tire transit. *IET Electr. Syst. Transp.* **2021**, *11*, 227–240. [CrossRef]
4. Gao, L.; Ma, F.; Jin, C. A Model-Based Method for Estimating the Attitude of Underground Articulated Vehicles. *Sensors* **2019**, *19*, 5245. [CrossRef] [PubMed]
5. Gao, G.; Wang, J.; Ma, T.; Liu, W.; Lei, T. Multistage Estimators for the Distributed Drive Articulated Steering Vehicle. *Math. Probl. Eng.* **2020**, *2020*, e5921285. [CrossRef]
6. Li, X.; Xu, N.; Li, Q.; Guo, K.; Zhou, J. A fusion methodology for sideslip angle estimation on the basis of kinematics-based and model-based approaches. *Proc. Inst. Mech. Eng. Part J. Automob. Eng.* **2020**, *234*, 1930–1943. [CrossRef]
7. Qi, D.; Feng, J.; Li, Y.; Wang, L.; Song, B. A Robust Hierarchical Estimation Scheme for Vehicle State Based on Maximum Correntropy Square-Root Cubature Kalman Filter. *Entropy* **2023**, *25*, 453. [CrossRef]
8. Kullu, O.; Cinar, E. A Deep-Learning-Based Multi-Modal Sensor Fusion Approach for Detection of Equipment Faults. *Machines* **2022**, *10*, 1105. [CrossRef]
9. Song, Y.; Nuske, S.; Scherer, S. A Multi-Sensor Fusion MAV State Estimation from Long-Range Stereo, IMU, GPS and Barometric Sensors. *Sensors* **2016**, *17*, 11. [CrossRef]
10. Fan, R.; Li, G.; Wu, Y. State Estimation of Distributed Drive Electric Vehicle Based on Adaptive Kalman Filter. *Sustainability* **2023**, *15*, 13446. [CrossRef]
11. Li, X.; Chan, C.-Y.; Wang, Y. A Reliable Fusion Methodology for Simultaneous Estimation of Vehicle Sideslip and Yaw Angles. *IEEE Trans. Veh. Technol.* **2016**, *65*, 4440–4458. [CrossRef]
12. Rodríguez, A.J.; Sanjurjo, E.; Pastorino, R.; Naya, M.Á. State, parameter and input observers based on multibody models and Kalman filters for vehicle dynamics. *Mech. Syst. Signal Process.* **2021**, *155*, 107544. [CrossRef]
13. Boada, B.L.; Boada, M.J.L.; Diaz, V. Vehicle sideslip angle measurement based on sensor data fusion using an integrated ANFIS and an Unscented Kalman Filter algorithm. *Mech. Syst. Signal Process.* **2016**, *72*, 832–845. [CrossRef]
14. Cheng, S.; Li, L.; Chen, J. Fusion Algorithm Design Based on Adaptive SCKF and Integral Correction for Side-Slip Angle Observation. *IEEE Trans. Ind. Electron.* **2018**, *65*, 5754–5763. [CrossRef]
15. Cui, B.; Chen, X.; Tang, X.; Huang, H.; Liu, X. Robust cubature Kalman filter for GNSS/INS with missing observations and colored measurement noise. *ISA Trans.* **2018**, *72*, 138–146. [CrossRef]
16. Chu, W.; Luo, Y.; Dai, Y.; Li, K. In-wheel motor electric vehicle state estimation by using unscented particle filter. *Int. J. Veh. Des.* **2015**, *67*, 115. [CrossRef]
17. Julier, S.J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proc. IEEE* **2004**, *92*, 401–422. [CrossRef]
18. van der Merw, R.; Doucet, A.; de Freitas, N.; Wan, E. The Unscented Particle Filter. In Proceedings of the 13th International Conference on Neural Information Processing Systems, Hong Kong, China, 3 October 2006. Available online: https://proceedings.neurips.cc/paper/2000/file/f5c3dd7514bf620a1b85450d2ae374b1-Paper.pdf (accessed on 14 August 2024).
19. Cheli, F.; Sabbioni, E.; Pesce, M.; Melzi, S. A methodology for vehicle sideslip angle identification: Comparison with experimental data. *Veh. Syst. Dyn.* **2007**, *45*, 549–563. [CrossRef]
20. Chen, T.; Chen, L.; Xu, X.; Cai, Y.; Jiang, H.; Sun, X. Reliable Sideslip Angle Estimation of Four-Wheel Independent Drive Electric Vehicle by Information Iteration and Fusion. *Math. Probl. Eng.* **2018**, *2018*, e9075372. [CrossRef]
21. Piyabongkarn, D.; Rajamani, R.; Grogg, J.A.; Lew, J.Y. Development and Experimental Evaluation of a Slip Angle Estimator for Vehicle Stability Control. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 78–88. [CrossRef]
22. You, S.-H.; Hahn, J.-O.; Lee, H. New adaptive approaches to real-time estimation of vehicle sideslip angle. *Control Eng. Pract.* **2009**, *17*, 1367–1379. [CrossRef]
23. Liu, Y.L.; Gu, X.H. Adaptive Interacting Multiple Model Unscented Particle Filter for Dynamic Acoustic Array. *Appl. Mech. Mater.* **2013**, *300*, 407–413. [CrossRef]
24. Munir, A.; Atherton, D.P. Adaptive interacting multiple model algorithm for tracking a manoeuvring target. *IEE Proc.-Radar Sonar Navig.* **1995**, *142*, 11–17. [CrossRef]
25. Rong Li, X.; Jilkov, V.P. Survey of maneuvering target tracking. Part V. Multiple-model methods. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1255–1321. [CrossRef]
26. Blom, H.A.P.; Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Trans. Autom. Control* **1988**, *33*, 780–783. [CrossRef]
27. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; The MIT Press: Cambridge, MA, USA, 2005; ISBN 978-0-262-25683-4.
28. Hewing, L.; Kabzan, J.; Zeilinger, M.N. Cautious Model Predictive Control Using Gaussian Process Regression. *IEEE Trans. Control Syst. Technol.* **2020**, *28*, 2736–2743. [CrossRef]
29. Pacejka, H.B.; Bakker, E. The Magic Formula Tyre Model. *Veh. Syst. Dyn.* **1992**, *21*, 1–18. [CrossRef]

30. Pacejka, H.B. *Tire and Vehicle Dynamics*, 3rd ed.; SAE International and Butterworth Heinemann: Warrendale, PA, USA, 2012. [CrossRef]
31. Wang, Z.; Lu, Z. Research on traction guidance cooperative control of virtual track train based on path-tracking and articulation hinge forces optimisation. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2024**, *238*, 5469–5487. [CrossRef]
32. Wang, Z.; Lu, Z.; Wei, J.; Qiu, X. Research on Virtual Track Train Path-Tracking Control Based on Improved MPC and Hierarchical Framework: A Reconfigurable Approach. *Appl. Sci.* **2023**, *13*, 8443. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.