



Combined recurrent neural networks and particle-swarm optimization for sideslip-angle estimation based on a vehicle multibody dynamics model

Yu Sun¹ · Yongjun Pan¹ · Ibna Kawsar¹ · Gengxiang Wang² · Liang Hou³

Received: 2 August 2023 / Accepted: 6 February 2024

© The Author(s), under exclusive licence to Springer Nature B.V. 2024

Abstract

The active safety system of a vehicle typically relies on real-time monitoring of the sideslip angle and other critical signals, such as the yaw rate. The vehicle sideslip angle cannot be measured directly due to the high cost and impracticality of sensor networks. The vehicle sideslip can be estimated using kinematic, dynamic, or machine-learning models and available vehicle states. This paper combines recurrent neural networks and the particle-swarm optimization (PSO) algorithm to estimate the vehicle sideslip angle accurately. First, a vehicle-dynamics model is constructed to conduct dynamics simulations of vehicles under various driving conditions and road environments for data collection. Secondly, the obtained vehicle states, including velocity, acceleration, yaw rate, and steering, are used to develop machine-learning models that estimate the vehicle sideslip angle. Two machine-learning models are proposed using the long short-term memory neural network (LSTM) and the bidirectional long short-term memory neural network (BiLSTM). Thirdly, the PSO algorithm is employed to optimize the hyperparameters of the LSTM and BiLSTM models for enhanced estimation precision. The Gaussian noise is added to the datasets to evaluate the robustness of the estimation models. The results indicate that the estimation models are capable of accurately predicting the vehicle's sideslip angle. The R^2 values of the results are mostly greater than 0.96. The PSO algorithm can improve estimation precision, and the PSO-LSTM model performs the best.

Keywords Sideslip-angle estimation · Vehicle-dynamics model · Long short temporal neural network · Particle-swarm optimization · Vehicle simulation

✉ Y. Pan
yongjun.pan@cqu.edu.cn

¹ College of Mechanical and Vehicle Engineering, Chongqing University, Chongqing, 400044, China

² Exeter Small-Scale Robotics Laboratory, Engineering Department, University of Exeter, Exeter, EX4 4QF, UK

³ Pen-Tung Sah Institute of Micro-Nano Science and Technology, Xiamen University, Xiamen, 361102, China

1 Introduction

Active safety systems such as electronic stability programs and traction-control systems are increasingly used in passenger cars. The operation of the active safety system depends heavily on the relevant states and parameters of the vehicle. The sideslip angle, which represents the angle between the vehicle's longitudinal axis and the direction of travel, is particularly important for handling and stability control (Zhang et al. [52]). The sideslip angle can be measured directly using optical or GPS sensors. Due to the high price of sensor networks and the inaccuracy of GPS sensors, it is impractical to implement these sensors in economical vehicles (Rajamani et al. [42]). Therefore, low-cost alternative methods are necessary for estimating the vehicle's sideslip angle for stability control (Li et al. [25]).

Methods based on models and machine learning are widely used to estimate the sideslip angle of vehicles (Guo et al. [16]; Chindamo et al. [12]). Kinematic and dynamic model-based methods can be roughly divided into two distinct categories. The kinematic model-based method considers only the vehicle's kinematics. This type of method is unsuitable for high-speed vehicles because dynamic characteristics need to be addressed in kinematic models (Park [40]). The state observer or Kalman filter is primarily utilized by the sideslip-angle estimation strategy based on the dynamic-based model (Boada et al. [6]; Liao and Borrelli [27]). If the vehicle model parameters are provided, this strategy yields satisfactory estimates. In addition, the dynamic tire model plays a crucial role in sideslip-angle estimation. Chen et al. designed a novel longitudinal-force observer based on strong-tracking filters and a robust sideslip-angle estimator based on Kalman filters for an electric vehicle. The authors utilized an enhanced tire model to improve the sideslip-angle estimation (Chen et al. [10]). Wang et al. developed a vehicle sideslip-angle observer based on the square-root cubature Kalman filter to estimate the sideslip angle during driving (Wang et al. [47]). Zhang et al. proposed a Takagi–Sugeno observer based on a piecewise affine lateral dynamics model to estimate the vehicle sideslip angle (Zhang et al. [53]). Xia et al. proposed a model that integrates vehicle kinematics and dynamics to estimate vehicle sideslip angle (Xia et al. [49, 50]). Chen et al. developed an adaptive square-root volume Kalman filter to enhance the estimation precision of the vehicle's sideslip angle (Chen et al. [11]). The literature review indicates that estimation precision requires a precise vehicle-dynamics model. The complexity of the vehicle model and driving environment have a significant impact on the estimated results.

To eliminate the need for any type of vehicle model and its associated complex parameter sets, approaches based on machine learning were proposed to estimate the angle of the sideslip of vehicles. Melzi et al. developed a hierarchical neural-network learning method that uses sensor-obtained vehicle parameters to train the mapping between the parameters and the sideslip angle (Melzi and Sabbioni [29]). Bonfitto et al. proposed an algorithm based on artificial neural networks to estimate the sideslip angle of vehicles in various road conditions (Bonfitto et al. [8]). The machine-learning-based methods can significantly reduce the impact of complex models and parameters on estimation precision, allowing for more precise estimated results. Furthermore, with the development of high-speed computing techniques and equipment, machine-learning-based methods have more obvious advantages in estimation and prediction. Methods based on machine learning have become more prevalent in recent years for estimating the angle of the sideslip of vehicles (Guo et al. [17]; Boada et al. [7]; Sieberg et al. [44]; Pan et al. [38]). The literature review demonstrates that machine-learning-based methods are complicated to implement due to the uncertainties in complex driving conditions. Combining a comprehensive vehicle-dynamics model with machine-learning algorithms is an appealing alternative for accurately estimating vehicle sideslip angles in complex environments.

This study employs a vehicle multibody dynamics model and recurrent neural-network (RNN) algorithms to estimate vehicle sideslip angles accurately. The lateral and longitudinal velocities, acceleration, yaw rate, and steering are used to train an estimation model. The output of the estimation model is the vehicle sideslip angle. The model's inputs and outputs are time-series data containing noise signals. We implement a unique form of the RNN algorithm, namely LSTM, to solve this time-series problem (Jalali et al. [22]). The LSTM algorithm can precisely characterize the complex nonlinear relationship between input parameters and output response. Recently, it has been widely utilized in parameter estimation and prediction. In addition, the PSO algorithm is used to optimize the hyperparameters of the LSTM algorithm. It improves the generalization ability and learning performance of LSTM models (Rahman and Venayagamoorthy [41]; Wrobel et al. [48]). The LSTM model's training data is obtained using a semirecursive vehicle multibody model. Contrary to other vehicle models, such as surrogate and nonlinear dynamics models, the multibody model considers the dynamics of all vehicle-system components. The classic Pacejka tire model is utilized in vehicle modeling, allowing accurate tire-force calculations and precise vehicle states to be obtained.

At present, many scholars devote themselves to studying the application of artificial intelligence in engineering (Hashemi et al. [19]; Li et al. [26]; Blanco et al. [5]; Nie et al. [33]). Many methods, such as machine learning, have begun to be combined with engineering applications to solve problems in engineering. For example, some optimization algorithms combined with LSTM network models have been applied to wind-power forecasting, which has played a specific role in promoting the wind-power generation project (Tuerxun et al. [46]; Ewees et al. [14]; Al-qaness et al. [2]). At the same time, some heuristic optimization algorithms, together with LSTM and improved LSTM, have been applied to predict CO₂ capture in deep saline aquifers and crude-oil production (Al-qaness et al. [1, 3]). The LSTM-based vehicle sideslip-angle estimation models are created by using a large number of vehicle states obtained from an accurate vehicle multibody model. Vehicle states related to the vehicle sideslip angle can be easily measured via some cheap sensors, such as speed and acceleration sensors, steering-wheel angle sensors, and so on. In addition, with the development of autonomous vehicles, the Advanced Driving Assistance System (ADAS) can also output these vehicle states. An increase in the number of vehicle states can improve the model estimation accuracy. The optimization algorithm is used to optimize the model. The estimation result of sideslip angle can be used in vehicle fields, such as active safety control. The highlights of this work are demonstrated in four aspects:

- The vehicle states are obtained in real time to estimate the sideslip angle due to the high cost of direct measurement via sensor networks.
- A LSTM model incorporating vehicle states as inputs are created to estimate vehicle sideslip angle to reduce the effects of uncertainty and the complexity of vehicle parameters.
- The PSO algorithm is used to optimize the LSTM model to improve the sideslip-angle estimation precision.
- A BiLSTM model is employed to compare the performance of the LSTM model and the PSO algorithm.

The remainder of the article is organized as follows: Section 2 introduces the semirecursive vehicle multibody model, LSTM algorithm, and PSO algorithm used to create the sideslip-angle estimation model. Section 3 illustrates the acquisition of data (vehicle states) and LSTM modeling. Section 4 presents the estimated LSTM model results, which are presented and analyzed in depth. In addition, the validity and precision of the LSTM model

following the introduction of Gaussian noise are discussed. Finally, we conclude our work in Sect. 5.

2 Methodology

2.1 Vehicle multibody dynamics model

The semirecursive multibody dynamics modeling technique is a well-established method for accurate and efficient data acquisition (vehicle states) (García de Jalón et al. [15]; Pan et al. [39]). In this study, we create a vehicle-dynamics model using this semirecursive multibody modeling technique. This method uses the double-velocity transformation technique. The first velocity transformation employs relative coordinates \mathbf{z} to describe Cartesian coordinates and dynamic equations that are easy to implement for open-loop multibody systems. Using the virtual power method and the first velocity transformation matrix, the open-loop vehicle system's equations of motion can be expressed as (Schwerin [43]):

$$\mathbf{R}_d^T \bar{\mathbf{M}}^\Sigma \mathbf{R}_d \ddot{\mathbf{z}} = \mathbf{R}_d^T (\bar{\mathbf{Q}}^\Sigma - \bar{\mathbf{P}}^\Sigma), \quad (1)$$

where $\ddot{\mathbf{z}}$ represents the relative accelerations of the system, and \mathbf{R}_d represents the first velocity transformation matrix. $\bar{\mathbf{M}}^\Sigma$, $\bar{\mathbf{Q}}^\Sigma$, and $\bar{\mathbf{P}}^\Sigma$ represent the accumulated mass matrix, external forces, and velocity-dependent inertial forces, respectively.

Considering a closed-loop vehicle system, a series of kinematic joints or slender rods must be eliminated to take full advantage of the system tree-topology (Hidalgo and García de Jalón [21]; Pan et al. [36]; He et al. [20]; Min et al. [30]). The rod-removal technique necessitates the calculation and accumulation of rod-related inertial forces and external forces such as damping forces, spring forces, and gravity into a recursive system (Pan et al. [37]). Consequently, the loop-closure constraint equations must be considered based on Eq. (1). The objective of the second velocity transformation is to represent relative coordinates in terms of independent relative coordinates using constraint equations and their Jacobian. Considering the constraint equations and applying the second velocity transformation matrix, the equations of motion for a closed-loop vehicle system are as follows (Pan et al. [34]):

$$\mathbf{R}_z^T \mathbf{R}_d^T \bar{\mathbf{M}}^\Sigma \mathbf{R}_d \mathbf{R}_z \ddot{\mathbf{z}}^i = \mathbf{R}_z^T \mathbf{R}_d^T \left[\bar{\mathbf{Q}}^\Sigma - \mathbf{T}^T \bar{\mathbf{M}} \frac{d(\mathbf{T} \mathbf{R}_d \mathbf{R}_z)}{dt} \dot{\mathbf{z}}^i \right], \quad (2)$$

where $\dot{\mathbf{z}}^i$ and $\ddot{\mathbf{z}}^i$ represent the independent relative velocities and accelerations of the closed-loop vehicle system, respectively. \mathbf{T} represents the path matrix, illustrating the system connectivity and tree-topology. $\bar{\mathbf{M}}$ represents the composite mass matrix, and \mathbf{R}_z denotes the second velocity transformation matrix.

Although Eq. (2) is significantly more complex than other multibody formulations, it only contains a small set of independent relative accelerations. Thus, computational efficiency is enhanced compared to other commercial software packages such as RecurDyn or ADAMS. Real-time and faster-than-real-time dynamic simulation can also be achieved (Callejo et al. [9]). Moreover, the ordinary differential form of Eq. (2) enables stable simulation of closed-loop systems using high-order numerical integration schemes. The fourth-order Runge–Kutta and Adams–Bashforth–Moulton methods are widely used for real-time simulations with long simulation times (Pan et al. [35]). This method of vehicle modeling considers and models the dynamics of all vehicle system components, including the chassis, suspensions, and tires. It is appropriate for precise vehicle dynamics calculations in complex driving conditions and road environments.

2.2 Particle-swarm optimization

The PSO optimization algorithm considers both individual local information and group global information to converge to the optimal solution faster. The PSO algorithm begins by initializing a set of particles in the search-space (Coronel-Escamilla et al. [13]). The particles update themselves at each iteration by monitoring two extreme values. One is the optimal solution discovered by the particle itself, known as the individual extrema. The other is the optimal solution currently discovered by the entire population, which is known as the global extrema. Individual and global extrema are continuously updated during the iterative process, and the final global extrema is the optimal solution obtained by the algorithm (Khan and Ling [23]; Zeng et al. [51]; Nie et al. [32]).

Assuming that the population is composed of n particles is $\mathbf{X} = [X_1, X_2, \dots, X_n]$, each particle is regarded as a D -dimensional point. Therefore, $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ represents the position of the i th particle in the search space. The best previous position of the i th particle (the position of the best fitness value) is defined as $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. The velocity of the i th particle is defined as $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. The global optimal position of the population is $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. At each iteration, the particle's velocity and position are updated using individual and global extrema. The mathematical expression for updating solution can be expressed as:

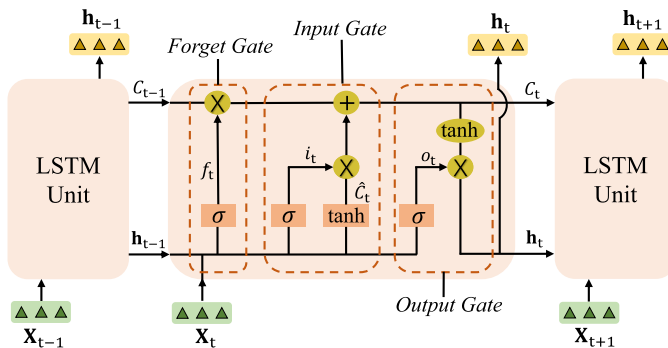
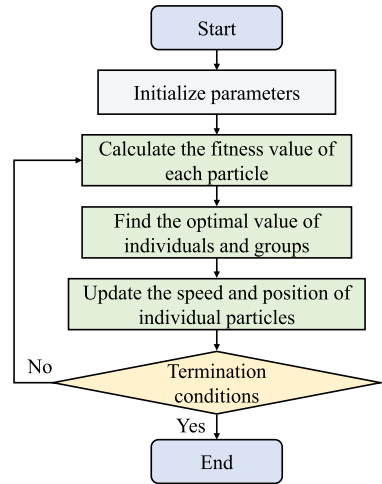
$$V_{id}^{m+1} = \omega V_{id}^m + c_1 r_1 (P_{id}^m - X_{id}^m) + c_2 r_2 (P_{gd}^m - X_{id}^m), \quad (3)$$

$$X_{id}^{m+1} = X_{id}^m + V_{id}^{m+1}, \quad (4)$$

where m represents the current number of iterations and d denotes the d th dimension of the variable, $d = 1, 2, \dots, D$. i represents the i th particle, $i = 1, 2, \dots, n$. V_{id}^m denotes the velocity vector of the d th dimension of particle i in the m th iteration, and X_{id}^m represents the position vector of the d th dimension of particle i in the m th iteration. P_{id}^m represents the historical optimal position of particle i in the d th dimension and m th iteration, that is, the optimal solution of the i th particle after the m th iteration. P_{gd}^m denotes the historical optimal position of the group in the d th dimension and m th iteration, that is, the optimal solution of the whole particle group after the m th iteration. ω is the inertia factor, which is used to balance particles' global and local search ability. The social and cognitive acceleration coefficients are represented by c_1 and c_2 , respectively. These coefficients are used to adjust the maximum time step of the individual and global optimal positions. r_1 and r_2 are random numbers between 0 and 1. They are used to increase the randomness of the search. The position and velocity are usually limited to a reasonable range between $[-X_{\max}, X_{\max}]$ and $[-V_{\max}, V_{\max}]$ to avoid a blind random search of particles. In addition, the fitness function of the PSO optimization procedure is determined by the mean absolute error between the reference result and the estimated result of each particle during each iteration. Individual and global extrema are continuously updated by continuously comparing the fitness values of every particle (Azimi and Moradi [4]; Han et al. [18]; Nguyen et al. [31]). The design step diagram of PSO is presented in Fig. 1.

2.3 Long short-term memory network

RNNs are neural networks that are used to process sequence data. LSTM is a specialized RNN that is primarily employed to solve the gradient disappearance and gradient explosion issues encountered during long sequence training. LSTM is more commonly used than conventional RNN to solve problems involving long-term sequences. The key to the LSTM is

Fig. 1 Design-step diagram of PSO**Fig. 2** Schematic diagram of LSTM structure

the cell state, which runs through the entire network and plays a critical role in transmitting network information. Three gates control the state of an LSTM cell: the forget gate, the input gate, and the output gate. The forget gate is used to selectively forget the input passed in from the previous node. The input gate controls the effect of the memory cell's input on the memory cell's state. The output gate controls the effect of the memory cell's state on the memory cell's output. The three-gate structure of the LSTM network enables the removal or addition of information to the cell state. The structure diagram of LSTM is presented in Fig. 2.

In Fig. 2, C_{t-1} , \hat{C}_t , and C_t represent the previous cell state, the temporary state, and the current cell state, respectively. LSTM controls them through the forget gate and the input gate. h_{t-1} and h_t denote the previous and current hidden states, respectively. X_t represents the current input. Therefore, the forget gate, input gate, and output gate can be mathematically expressed as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f), \quad (5)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i), \quad (6)$$

$$o_t = \sigma(W_o \cdot [\mathbf{h}_{t-1}, \mathbf{X}_t] + b_o), \quad (7)$$

where $W_{(\cdot)}$ and $b_{(\cdot)}$ represent the weight matrix and bias vector of the forget, input, and output gates, respectively. $\sigma(\cdot)$ represents the sigmoid activation function.

The cell state exists in the whole chain system of LSTM, which can be mathematically expressed as:

$$C_t = f_t \times C_{t-1} + i_t \times \hat{C}_t, \quad (8)$$

$$\hat{C}_t = \tanh(W_c \cdot [\mathbf{h}_{t-1}, \mathbf{X}_t] + b_c), \quad (9)$$

where W_c and b_c represent the weight matrix and bias vector of the temporary cell state, respectively. $\tanh(\cdot)$ is the activation function. The calculation formulas of the two activation functions are described as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \quad (10)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \quad (11)$$

2.4 Bidirectional long short-term memory network

LSTM is a one-way RNN that learns previous data to drive subsequent data without considering subsequent data. Moreover, there is a BiLSTM. It is an extension of the LSTM model and shares the same structure. Based on the LSTM calculation process, a reverse operation is introduced to consider the influence of several subsequent inputs. The structure diagram of BiLSTM is presented in Fig. 3.

The single-layer BiLSTM is composed of two LSTMs. One is to process the input sequence in the forward direction, and the other is to process the sequence in the reverse direction. After processing, the outputs of the two LSTMs are spliced. As shown in the Fig. 3, the final output of BiLSTM can only be obtained after all time steps are calculated. The forward LSTM receives a result vector through a fixed time step. The reverse LSTM also receives

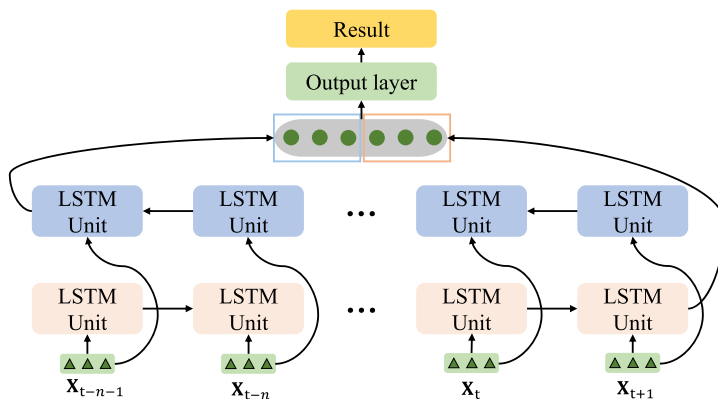


Fig. 3 Schematic diagram of BiLSTM structure

another result after the same time step. The two result vectors are spliced to obtain the final BiLSTM output result.

Similar to other neural-network models, initial LSTM parameters are randomly generated. The gradient is then solved using the backpropagation algorithm using the training data to calculate the loss function. The model parameters are adjusted and updated in response to the gradient until convergence is reached. The most common mean squared error function is used as the loss function:

$$L = \frac{\|\mathbf{Y} - \hat{\mathbf{Y}}\|^2}{2n}, \quad (12)$$

where L represents the loss of the LSTM model, \mathbf{Y} denotes the reference data, $\hat{\mathbf{Y}}$ represents the results of the LSTM model, and n is the number of datasets. In addition, the Adam algorithm is used to train LSTM. In the Adam algorithm, the index of convergence is the statistics $R(T)$:

$$R(T) \leq \sum_{i=1}^d \sum_{t=1}^T g_{t,i} \left(\theta_i^{(t)} - \theta_i^* \right), \quad (13)$$

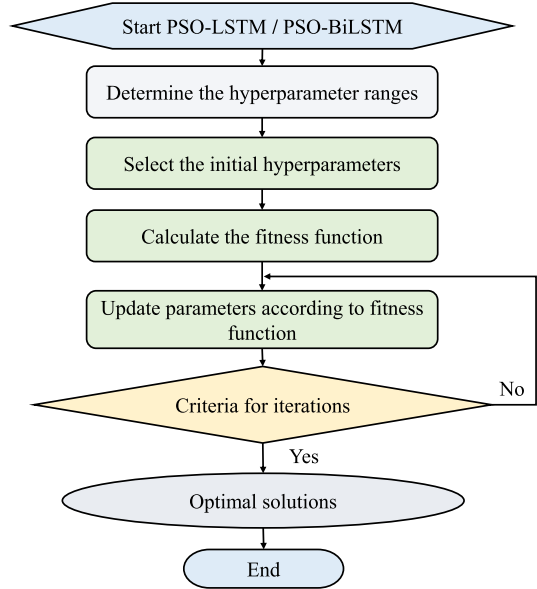
where d represents the number of parameter variables. θ represents a parameter variable. θ^* represents the final solution. $g(\cdot)$ represents the gradient of the objective function. When the loss function is a convex function at any time, the order of the upper-bound optimal number of $R(T)$ is $O(T^{1/2})$, and the algorithm converges.

The performance of the LSTM model is highly dependent on the network training parameters. It is critical to constantly tweak the model's hyperparameters, such as the learning rate and the number of neurons. The objective of neural-network hyperparameters adjustment is to obtain a set of optimal hyperparameters that yield the most accurate predictions. Inappropriate hyperparameters will reduce the prediction or estimation ability. This study optimizes the learning rate and the number of neurons in the hidden layer of the LSTM model using the PSO algorithm. The procedure of the PSO-optimized LSTM and BiLSTM modeling is presented in Fig. 4. This method significantly reduces the need for human intervention during model training and ensures that the final model has the highest prediction performance.

3 Datasets and LSTM modeling

3.1 Acquisition of datasets

Once the desired network type is selected, the network must be trained using datasets to establish a mapping between inputs and the output. Thus, a vehicle's sideslip angle (output) can be estimated based on its current state (inputs). We consider a sedan vehicle that consists of a MacPherson strut suspension in the front axle and a multilink suspension in the rear axle. The main parameters of the vehicle and the necessary parameters of the vehicle components are listed in Table 1 and Table 2, respectively. The vehicle's multibody model is developed using the semirecursive multibody modeling method. By performing vehicle simulations, we obtain the lateral and longitudinal velocity and acceleration (Y_v , Y_a , X_v , X_a), yaw rate ($\dot{\psi}$), and steering-bar travel length (STL) (l) at every time step. On the other hand, the sideslip angle (β) of the vehicle is also obtained.

Fig. 4 PSO-LSTM/BiLSTM procedure**Table 1** Main parameters of the vehicle system

Parameter	Value	Parameter	Value
Vehicle mass	1155 kg	Tire-rolling radius	0.4673 m
Distance from centroid to front axle	0.7209 m	Centroid height	0.5373 m
Distance from centroid to rear axle	2.0791 m	Wheelbase	2.8 m
Stiffness of front suspensions	40,000 N/m	Damping of front suspensions	1800 N s/m
Stiffness of rear suspensions	35,000 N/m	Damping of rear suspensions	1800 N s/m
Sprung mass	1068 kg	Unsprung mass	87 kg

Table 2 Parameters of some pivotal components of the vehicle model

Component	Mass (kg)	Centroid Coordinates (m)	Moment of inertia (kg m^2)
Steering knuckle	Left 10	(2.1286, 1.1518, 0.1808)	diag([0.01, 0.01, 0.0,])
	Right 10	(2.1286, -1.1518, 0.4808)	diag([0.01, 0.01, 0.01])
Suspension	Dampener_L 2	(2.1066, 1.1008, 1.0257)	diag([0.0001, 0.0001, 0.0001])
	Dampener_R 2	(2.1066, -1.1008, 1.0257)	diag([0.0001, 0.0001, 0.0001])
	Swing arm_L 2.5	(2.1648, 0.9299, 0.3373)	diag([0.02, 0.02, 0.02])
	Swing arm_R 2.5	(2.1648, -0.9299, 0.3373)	diag([0.02, 0.02, 0.02])
Front wheel	Left 23	(2.0791, 1.1967, 0.4673)	diag([0.4, 0.8, 0.4])
	Right 23	(2.0791, -1.1967, 0.4673)	diag([0.4, 0.8, 0.4])
Rear wheel	Left 10	(-2.0791, 1.1967, 0.4673)	diag([0.74, 1.46, 0.74])
	Right 10	(-2.0791, -1.1967, 0.4673)	diag([0.74, 1.46, 0.74])

Table 3 Maneuvers for data acquisition

Maneuver	STL (m)	Time (s)	Initial speed (km/h)	Driving torque (N m)
Straight	(-)	5	50, 90	50, 500
Double-lane	$\pm 0.06, \pm 0.08, \pm 0.10$	5	60, 70, 90	50, 300, 500
Sine-steer	$\pm 0.06, \pm 0.08, \pm 0.10$	8	70, 80, 90	400, 500, 600
Sweep-steer	$\pm 0.10, \pm 0.11$	9	60, 80, 90	400, 500, 600
Cornering	(-)	3, 4, 5, 6	10, 30, 50, 90	500
Random-steer	(-)	15, 17, 18	40, 50, 60, 70, 80, 90, 100	300, 500, 550, 600

The dataset must include as many driving conditions as possible to ensure the effectiveness of the LSTM model. The training data set obtained includes nearly 100,000 sets of data under various driving conditions. In this study, we acquire data under straight, double-lane, sine-steering, sweep-steering, cornering, and random-steering conditions, which are common driving movements on urban roads. Vehicle simulations employ a variety of initial vehicle speeds and driving torques. The details are presented in Table 3. This table includes the following maneuvers: (1) straight on the road without lateral drift and sideslip; (2) double lane changes with different initial speeds, driving torques, and steering behavior; (3) sinusoidal steering; (4) sweep steering; (5) right-angle turning maneuvers with different speeds, driving torques, and steering bar displacements; and (6) random steering with different speeds, driving torques, and steering behavior.

The tire model used in the vehicle multibody dynamics modeling is the Pacejka tire model, where tire deformation and lateral forces can be considered. This tire model can be used to precisely calculate the forces and moments caused by the tire's contact with the ground. During all simulations, the time step is 2.5 ms. As shown in Table 3, different simulation times are used for different maneuvers. By simulating vehicles, precise vehicle states and slip angle can be obtained at each time step. Next, LSTM/BiLSTM models are developed to estimate the vehicle's sideslip angle.

3.2 Sideslip-angle estimation

The LSTM model is able to prevent gradient explosions during training. However, the normalization of datasets is necessary. Effective normalization can improve the accuracy and convergence speed of the LSTM model. We use the Z-score normalization method in this study. The processed data has a mean of 0 and a standard deviation of 1. The Z-score normalization method is mathematically expressed as follows:

$$x_i^* = \frac{x_i - \mu(x_i)}{\sigma(x_i)}, \quad (14)$$

where x_i^* represents normalized data, x_i denotes the i th actual value of the array, $\mu(\cdot)$ represents the mean value of data, and $\sigma(\cdot)$ is the standard deviation of data. The LSTM modeling is based on the principle that the vehicle sideslip angle can be expressed as a function of lateral and longitudinal velocity and acceleration (Y_v , Y_a , X_v , X_a), yaw rate ($\dot{\psi}$), and steering bar travel length (l):

$$\beta(k) = f[X_v(k-1), \dots, X_v(k-n), \dots, l(k-1), \dots, l(k-n)]. \quad (15)$$

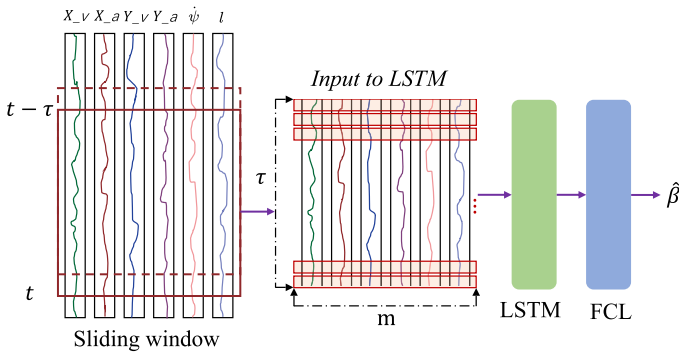


Fig. 5 Network for sideslip-angle estimation

The vehicle's sideslip angle (β) at time k can be described as a time-series function composed of the previous n time steps' vehicle states. This study combines the LSTM and full connection layer (FCL) to estimate the vehicle's sideslip angle. The LSTM inputs include six vehicle states obtained from the vehicle multibody dynamics model. They are the lateral and longitudinal velocity and acceleration (Y_v , Y_a , X_v , X_a) of the vehicle, yaw rate ($\dot{\psi}$), and steering-bar travel length (l). Considering the functional relationship between the LSTM input parameters and the vehicle sideslip angle, a sliding window of size $m \cdot \tau$ is employed to represent the network inputs, where m is the number of input parameters and τ is the time-span of the input datasets. They are illustrated in Fig. 5.

The setting τ ($= 250$ ms) in Fig. 5 consists of n ($= 100$) steps with a time-step of 2.5 ms. The setting of a time step of 2.5 ms takes both computational costs and solution accuracy into account during the vehicle dynamics simulations. Each sliding window moves at a one-step time interval. The purpose of these parameters is to consider Eq. (15), where we estimate the output of the current time-step using the inputs from the previous 100 time steps. In other words, the vehicle states of the preceding 250 ms are used to estimate the current sideslip angle. Therefore, the input values of each network are (m, τ) units formed by the sliding window. The LSTM network processes the relationship between the time-series input data and the output target. Finally, the network estimation is obtained by passing the result at the end of LSTM through a fully connected layer. After many attempts, the maximum epoch of LSTM was set to 100. Furthermore, the PSO algorithm is used to optimize the LSTM model to estimate the vehicle's sideslip angle accurately. The number of iterations in PSO is set to 10 due to the large amount of training data. As the number of iteration steps is small, the stopping criterion is set according to the maximum number of iteration steps. The c_1 and c_2 values are empirically set to 2.0 and 2.0, respectively. The procedure for estimating the vehicle sideslip angle is presented in Fig. 6.

4 Results and discussion

4.1 Results

The LSTM, PSO-LSTM, BiLSTM, and PSO-BiLSTM models are developed by training the datasets obtained from various maneuvers. As a result, four complex scenarios are chosen to estimate the sideslip angle. The four complex scenarios are presented in Table 4. The

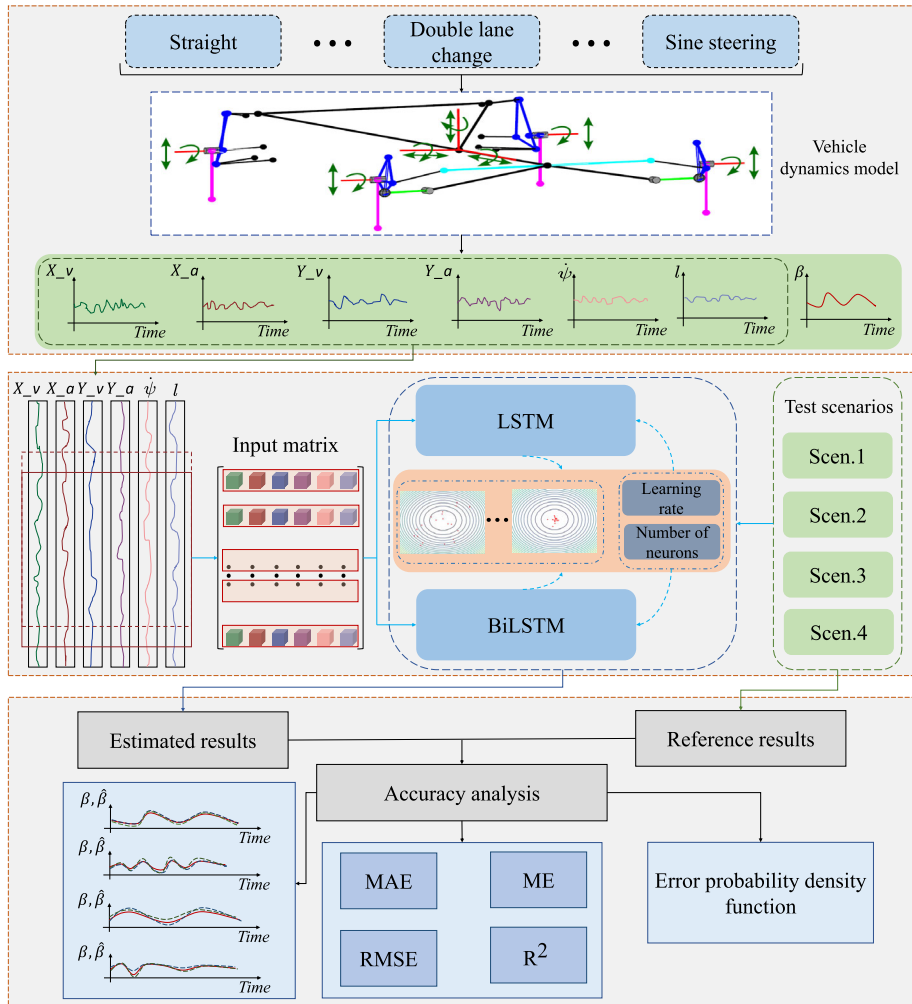


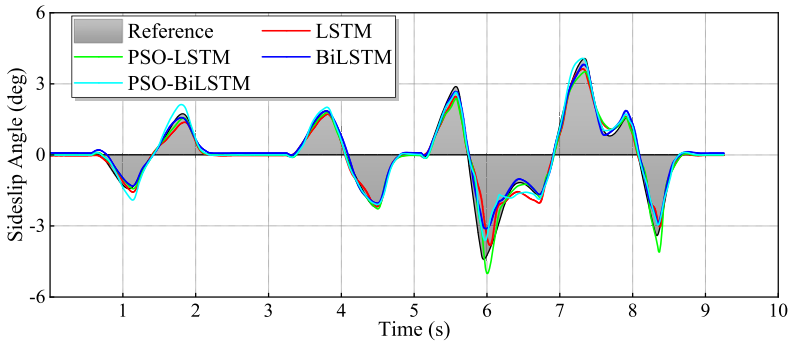
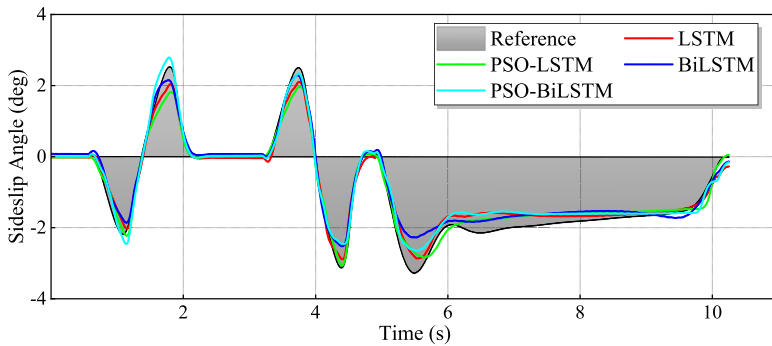
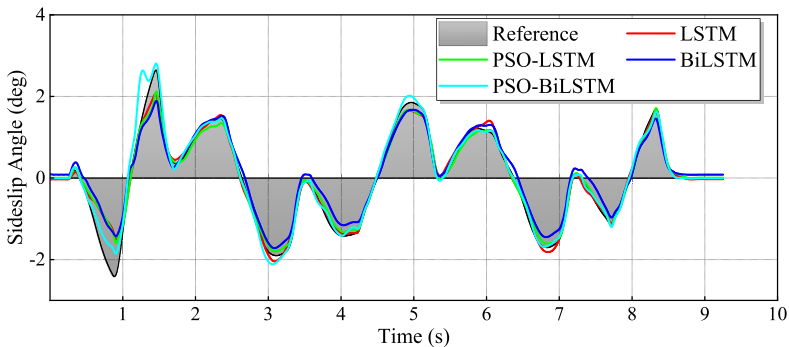
Fig. 6 Procedure of vehicle sideslip-angle estimation

first test scenario combines a double lane-change maneuver and a complete sine-steering maneuver, and the initial vehicle speed is 80 km/h. The second scenario combines a double lane-change maneuver with a right-angle turn, with an initial vehicle speed of 90 km/h. The third scenario is a continuous-sine maneuver with an initial speed of 75 km/h. The fourth scenario is a random-steering maneuver with an initial speed of 75 km/h. The driving torque in each of the four test scenarios is 500 N m. The performance of the four models is investigated and validated by comparing the output of the individual models to the reference sideslip angle during a series of maneuvers.

Figures 7 through 10 depict the estimated sideslip angle for each of the four test scenarios. In addition, the reference of the sideslip angle is provided for comparison. It is evident from the figures that each model can provide a smooth prediction. In addition, the trend of vehicle sideslip-angle variation can be accurately estimated. However, the maximum esti-

Table 4 Scenarios for testing and validation

Test scenario	Description	Initial speed (km/h)	Torque (N m)	Time (s)
Scen.1	DLC + Sinesteer	80	500	9.5
Scen.2	DLC + Cornering	90	500	10.5
Scen.3	Continuous sinesteer	75	500	9.5
Scen.4	Randomsteer	75	500	8

**Fig. 7** Estimated sideslip angle of scenario 1**Fig. 8** Estimated sideslip angle of scenario 2**Fig. 9** Estimated sideslip angle of scenario 3

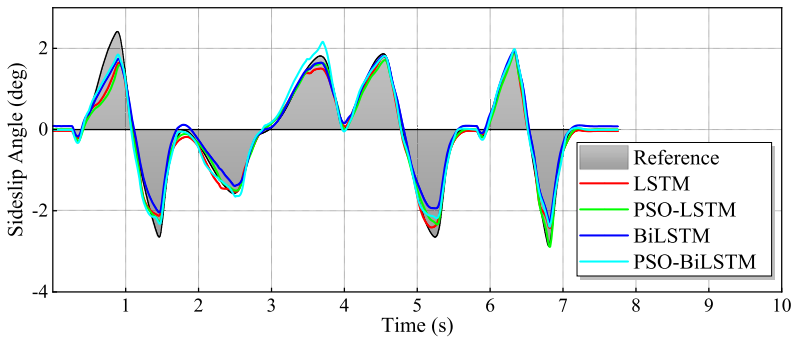


Fig. 10 Estimated sideslip angle of scenario 4

mated sideslip angle varies between the LSTM, PSO-LSTM, BiLSTM, and PSO-BiLSTM models. In the first test scenario, the maximum error occurs when the steering wheel is turned by sine. In the second scenario, the maximum error occurs during cornering because cornering has a significant effect on the vehicle's sideslip angle. In the third scenario, the error at the beginning of sine steering is apparent. The sideslip angle of the vehicle can be accurately estimated in the later stage of driving. In the later stages of driving, it is possible to estimate the vehicle's sideslip angle accurately. In the fourth scenario, the maximum error occurs at the first peak, and the estimated sideslip angle is accurate in the later stage. A relatively large error between the estimated and the reference results is observed. The error mostly occurs at the peak position, which is the position where the rotation of the steering-wheel changes. The testing scenarios have more steering-wheel rotation behaviors than the training scenarios, resulting in some estimation errors.

We employ a probability density function (PDF) to investigate the estimation errors in greater detail. Figures 11 through 14 depict the error distributions for the four test scenarios. The solid ball in these figures represents the mean, while the horizontal line represents the standard deviation. They fit the Gaussian output of the estimated errors. The error distributions for test scenarios 1 and 2 are described in Figs. 11 and 12, respectively. The PSO-LSTM model has the smallest mean and standard deviation, as evidenced by the graphs. The standard deviation of the PSO-BiLSTM model described in Fig. 13 is the smallest, but its mean is greater than that of the LSTM and PSO-LSTM models. The mean and standard deviation of the PSO-BiLSTM model described in Fig. 14 compared to other models are smaller. Overall, the estimation performance can be enhanced when the PSO algorithm is combined with LSTM or BiLSTM.

Four common evaluation indicators, including mean error (ME), mean absolute error (MAE), root mean square error (RMSE), and mean coefficient of determination (R^2), are used to evaluate the performance of each model. The results are depicted in Table 5. The R^2 of each model in each test scenario is greater than 0.95, as shown in the table, indicating that the estimated results fit the reference data. The MAE, ME, and RMSE for the four models in the four test scenarios are also very small. In scenarios 1 and 2, the MAE values of the PSO-LSTM model are lower than those of other models. In scenarios 3 and 4, the MAE values of PSO-LSTM and PSO-BiLSTM are very similar and less than those of the other two models. This indicates that the PSO-optimized LSTM/BiLSTM model outperforms the independent LSTM and BiLSTM model when estimating the sideslip angle of a vehicle. In addition, the results of the Gaussian distribution of the estimation errors and the model-evaluation metrics

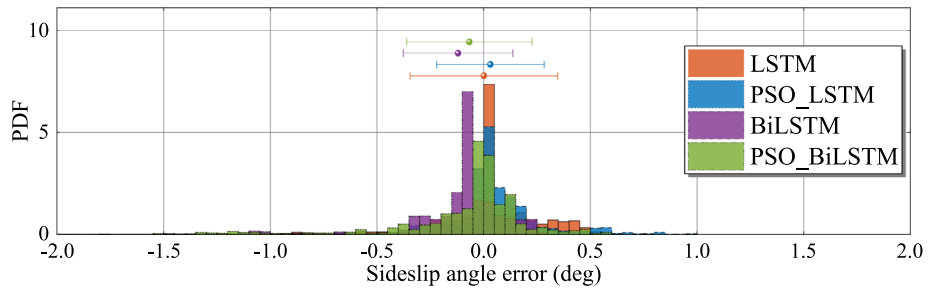


Fig. 11 PDF of the estimation error: scenario 1

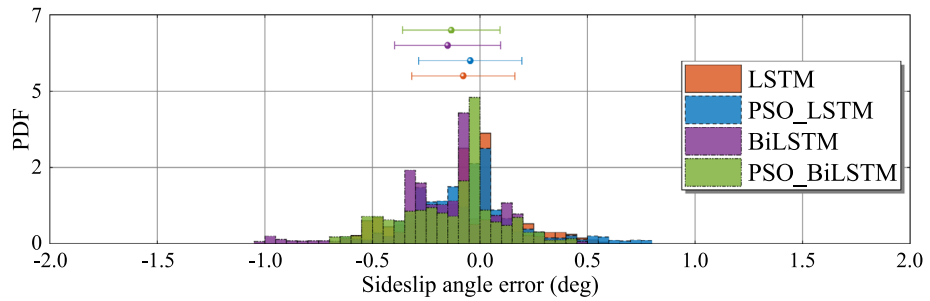


Fig. 12 PDF of the estimation error: scenario 2

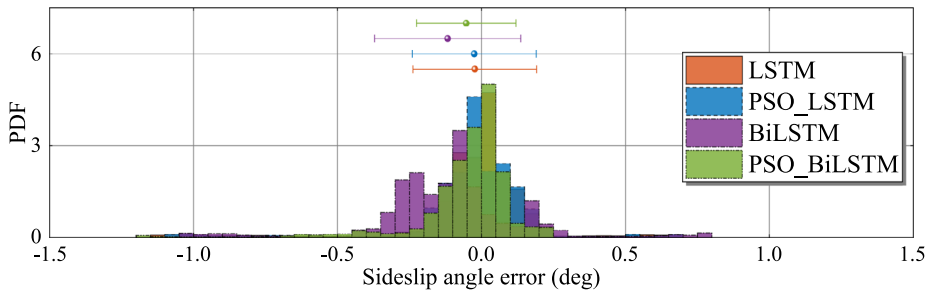


Fig. 13 PDF of the estimation error: scenario 3

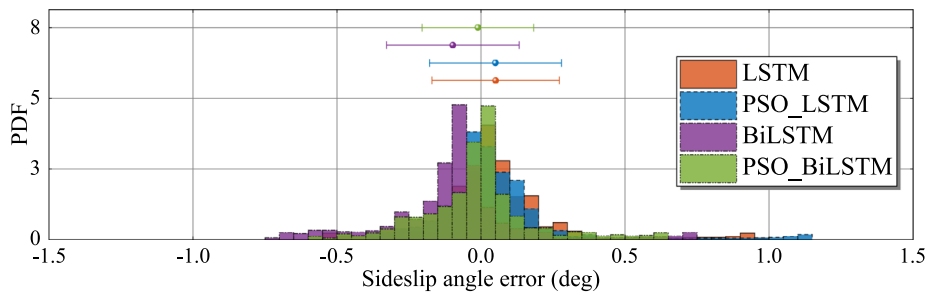


Fig. 14 PDF of the estimation error: scenario 4

Table 5 Accuracy analysis of the models

Maneuvers	Models	MAE	ME	RMSE	R^2
Scne.1	LSTM	0.188	−0.001	0.346	0.954
	PSO-LSTM	0.149	−0.032	0.254	0.972
	BiLSTM	0.169	0.119	0.283	0.981
	PSO-BiLSTM	0.166	0.067	0.299	0.962
Scne.2	LSTM	0.195	0.077	0.252	0.986
	PSO-LSTM	0.180	0.045	0.244	0.983
	BiLSTM	0.218	0.150	0.288	0.978
	PSO-BiLSTM	0.196	0.133	0.263	0.979
Scne.3	LSTM	0.119	0.023	0.216	0.966
	PSO-LSTM	0.112	0.025	0.216	0.975
	BiLSTM	0.205	0.117	0.280	0.971
	PSO-BiLSTM	0.099	0.053	0.181	0.976
Scne.4	LSTM	0.143	−0.051	0.227	0.975
	PSO-LSTM	0.133	−0.050	0.234	0.976
	BiLSTM	0.177	0.098	0.250	0.986
	PSO-BiLSTM	0.129	0.011	0.194	0.975

indicate that the PSO-LSTM can better extract and process the information of the vehicle states and accurately estimate the vehicle sideslip angle.

4.2 Discussion

As stated previously, the training datasets generated by the vehicle multibody model are simulated data. Gaussian noise is introduced to the datasets to imitate the actual data collected via sensor networks (Strano and Terzo [45]; Liu et al. [28]). The noise includes lateral and longitudinal velocities with a mean of 0 and a variance of 1 km/h, lateral and longitudinal accelerations with a variance of 0.1 m/s², yaw rate with a variance of 0.2 deg/s, and steering-bar travel length with a variance of 1 mm (Kim et al. [24]). The LSTM, PSO-LSTM, BiLSTM, and PSO-BiLSTM models are trained with Gaussian noise-contaminated training datasets. In addition, Gaussian noise is added to the four test scenarios to estimate the sideslip angle. Figures 15 through 18 describe the estimated sideslip in the four test scenarios. Figure 15 shows the estimated results for the double lane-change and single sine-steering scenario. As can be seen, the maximum error appears at the peak of the sideslip angle when the sine steering occurs at about 6 s. This result is similar to the estimated results without Gaussian noise. In the second test scenario depicted in Fig. 16, the maximum error occurs during the initial part of the vehicle's turn. In the cases of continuous sine steering and random steering, as shown in Figs. 17 and 18, the maximum error occurs during the steering's initial time period.

Even though Gaussian noise is introduced, the results of the four test scenarios indicate that each model can estimate the variation trend of the sideslip angle. Additionally, it can be seen that the PSO-LSTM model's resulting curve fits the reference better than the other three models. This conclusion is consistent with the previous conclusion. Furthermore, the same evaluation indicators are used to quantitatively assess the accuracy of each model

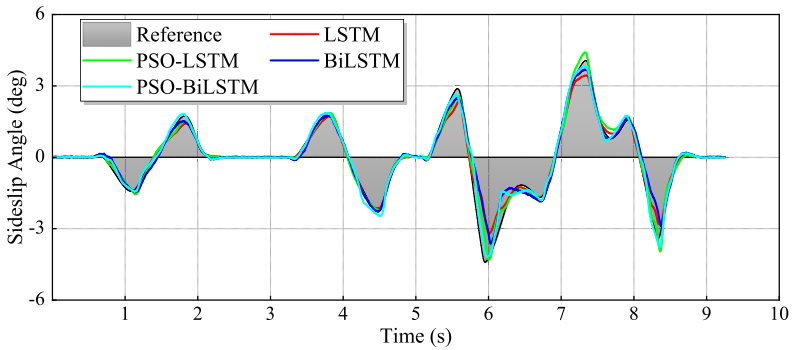


Fig. 15 Estimated sideslip angle of scenario 1 with Gaussian noise

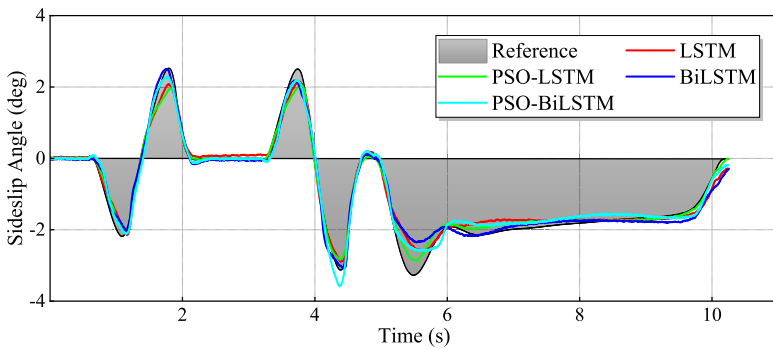


Fig. 16 Estimated sideslip angle of scenario 2 with Gaussian noise

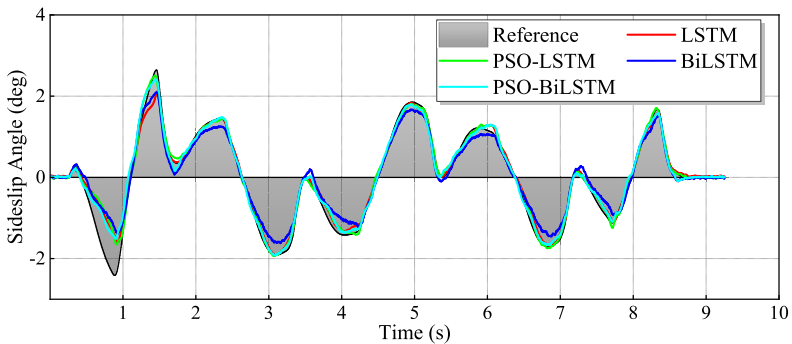


Fig. 17 Estimated sideslip angle of scenario 3 with Gaussian noise

subjected to Gaussian noise. The results are listed in Table 6. The R^2 of each model is greater than 0.96, and the R^2 of the PSO-LSTM model in different scenarios is greater than 0.97. The results in Figs. 15 to 18 and Table 6 demonstrate that the PSO-LSTM model can robustly and accurately estimate the vehicle's sideslip angle. In addition, considering the training time cost of the model, due to the introduction of the PSO algorithm and the huge

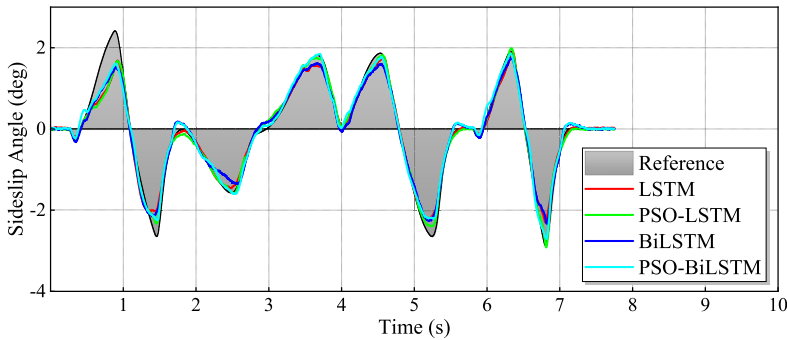


Fig. 18 Estimated sideslip angle of scenario 4 with Gaussian noise

Table 6 Accuracy analysis of the models with Gaussian noise

Maneuvers	Models	MAE	ME	RMSE	R^2
Scne.1	LSTM	0.170	0.024	0.331	0.967
	PSO-LSTM	0.139	0.022	0.024	0.974
	BiLSTM	0.187	0.053	0.358	0.951
	PSO-BiLSTM	0.133	0.015	0.221	0.978
Scne.2	LSTM	0.170	0.056	0.239	0.978
	PSO-LSTM	0.128	0.029	0.187	0.991
	BiLSTM	0.168	0.012	0.252	0.967
	PSO-BiLSTM	0.165	0.037	0.216	0.978
Scne.3	LSTM	0.120	0.063	0.216	0.974
	PSO-LSTM	0.107	0.061	0.209	0.980
	BiLSTM	0.188	0.066	0.277	0.970
	PSO-BiLSTM	0.108	0.057	0.193	0.976
Scne.4	LSTM	0.146	-0.011	0.250	0.972
	PSO-LSTM	0.120	-0.030	0.221	0.972
	BiLSTM	0.165	-0.010	0.246	0.976
	PSO-BiLSTM	0.136	0.008	0.212	0.975

amount of training data, the training time of the PSO-LSTM and PSO-BiLSTM models is longer than that of the other two models, by nearly 20 h. However, in the prediction stage, the time consumed by the PSO-LSTM model is roughly the same as that of other models, which is relatively short. Therefore, the PSO-LSTM model is still superior to other models in estimating vehicle sideslip angle and has better real-time estimation performance.

5 Conclusions

Due to direct measurement's high cost and complexity, this study proposes a new sideslip-angle estimation method based on the PSO-LSTM and PSO-BiLSTM algorithms and the vehicle multibody dynamics model. LSTM and BiLSTM describe the mapping relationship

between vehicle states (inputs) and the sideslip angle (output). The PSO algorithm searches for optimal LSTM and BiLSTM hyperparameters. The vehicle multibody dynamics model is used to acquire training data for modeling PSO-LSTM and PSO-BiLSTM under various maneuvers. Four complex scenarios are selected for experiments and validation to validate the effectiveness of the proposed models for vehicle sideslip-angle estimation. The results demonstrate that the PSO algorithm enhances the estimation performance of the LSTM and BiLSTM. The PSO-LSTM and PSO-BiLSTM models proposed in this paper can estimate the vehicle sideslip angle robustly and precisely. The estimation accuracy of the PSO-LSTM model is superior to other models. Overall, a novel method combining LSTM/BiLSTM and PSO algorithms for vehicle sideslip-angle estimation is presented. In future research, we will explore the performance of the latest deep-learning methods and optimization algorithms in estimating vehicle sideslip angles, and compare them with standard physical models such as Kalman filters. The vehicle states obtained via sensor networks can be used as model inputs to estimate the sideslip on real applications. In order to complete the task of vehicle sideslip-angle estimation as fully as possible, the interval estimation of vehicle sideslip angle will also be the work content of the next stage. In addition, the estimation models will be further validated by physical tests and used in intelligent vehicles for online prediction and early warning control.

Author contributions Yu Sun: Software, Formal analysis, Data curation, Investigation, Writing-Original draft preparation. Yongjun Pan: Conceptualization, Methodology, Writing-Reviewing and Editing, Supervision, Funding acquisition. Ibna Kawsar: Writing-Reviewing and Editing, Validation. Gengxiang Wang: Formal analysis, Data curation, Validation. Liang Hou: Methodology, Writing-Reviewing and Editing.

Funding This work was funded by the National Natural Science Foundation of China (No. 12072050).

Declarations

Competing interests The authors declare no competing interests.

References

1. Al-qaness, M.A.A., Ewees, A., Abualigah, L., Alrassas, A., Elsayed Abd Elaziz, M.: Evaluating the applications of dendritic neuron model with metaheuristic optimization algorithms for crude-oil-production forecasting. *Entropy* **24**, 1674 (2022)
2. Al-qaness, M.A.A., Ewees, A., Elsayed Abd Elaziz, M., Samak, A.: Wind power forecasting using optimized dendritic neural model based on seagull optimization algorithm and aquila optimizer. *Energies* **15**(24), 9261 (2022)
3. Al-qaness, M.A.A., Ewees, A., Alrassas, A., Dahou, A., Elsayed Abd Elaziz, M.: Predicting CO₂ trapping in deep saline aquifers using optimized long short-term memory. *Environ. Sci. Pollut. Res. Int.* **30**(12), 33780–33794 (2023)
4. Azimi, M., Moradi, S.: Robust optimal solution for a smart rigid–flexible system control during multi-mode operational mission via actuators in combination. *Multibody Syst. Dyn.* **52**, 1–25 (2021)
5. Blanco, J.L., Moreno, J.L., Gimenez, A.: Multibody dynamic systems as Bayesian networks: applications to robust state estimation of mechanisms. *Multibody Syst. Dyn.* **34**, 103–128 (2015)
6. Boada, B., Boada, M., Diaz, V.: Vehicle sideslip angle measurement based on sensor data fusion using an integrated AFIS and an unscented Kalman filter algorithm. *Mech. Syst. Signal Process.* **72**(73), 832–845 (2016)
7. Boada, B., Boada, M., Diaz, V.: A robust observer based on energy-to-peak filtering in combination with neural networks for parameter varying systems and its application to vehicle roll angle estimation. *Mechatronics* **50**, 196–204 (2018)
8. Bonfitto, A., Feraco, S., Tonoli, A., Amati, N.: Combined regression and classification artificial neural networks for sideslip angle estimation and road condition identification. *Veh. Syst. Dyn.* **58**, 1–22 (2019)

9. Callejo, A., Pan, Y., Ricón, J., Kovács, J., García de Jalón, J.: Comparison of semirecursive and subsystem synthesis algorithms for the efficient simulation of multibody systems. *J. Comput. Nonlinear Dyn.* **12**, 011020 (2017)
10. Chen, L., Chen, T., Xu, X., Cai, Y., Jiang, H., Sun, X.: Sideslip angle estimation of in-wheel motor drive electric vehicles by cascaded multi-Kalman filters and modified tire model. *Metrol. Meas. Syst.* **26**, 185–208 (2019)
11. Chen, X., Li, S., Li, L., Zhao, W., Cheng, S.: Longitudinal-lateral-cooperative estimation algorithm for vehicle dynamics states based on adaptive-square-root-cubature-Kalman-filter and similarity-principle. *Mech. Syst. Signal Process.* **176**, 109162 (2022)
12. Chindamo, D., Lenzo, B., Gadola, M.: On the vehicle sideslip angle estimation: a literature review of methods, models, and innovations. *Appl. Sci.* **8**(3), 355 (2018)
13. Coronel-Escamilla, A., Torres, F., Gómez-Aguilar, J., Escobar Jiménez, R., Guerrero-Ramírez, G.: On the trajectory tracking control for an scara robot manipulator in a fractional model driven by induction motors with pso tuning. *Multibody Syst. Dyn.* **40**, 1–21 (2018)
14. Ewees, A.A., Al-qaness, M.A., Abualigah, L., Elaziz, M.A.: Hbo-Istm: optimized long short term memory with heap-based optimizer for wind power forecasting. *Energy Convers. Manag.* **268**, 116022 (2022)
15. García de Jalón, J., Álvarez, E., de Ribera, F., Rodríguez, I., Funes, F.: A fast and simple semi-recursive formulation for multi-rigid-body systems. In: Ambrósio, J. (ed.) *Advances in Computational Multibody Systems. Computational Methods in Applied Sciences*, vol. 2, Chap. 1, pp. 1–23. Springer, Dordrecht (2005)
16. Guo, H., Cao, D., Chen, H., Lv, C., Wang, H., Yang, S.: Vehicle dynamic state estimation: state of the art schemes and perspectives. *IEEE/CAA J. Autom. Sin.* **5**(2), 418–431 (2018)
17. Guo, J., Luo, Y., Li, K., Dai, Y.: Coordinated path-following and direct yaw-moment control of autonomous electric vehicles with sideslip angle estimation. *Mech. Syst. Signal Process.* **105**, 183–199 (2018)
18. Han, B.L., Zhao, R., Luo, Q.S., Xu, F., Zhao, J.H.: Static gait optimization method for quadruped robot based on particle swarm optimization algorithm. *Beijing Ligong Daxue Xuebao/Trans. Beijing Inst. Technol.* **37**, 461–465 (2017)
19. Hashemi, A., Orzechowski, G., Mikkola, A., McPhee, J.: Multibody dynamics and control using machine learning. *Multibody Syst. Dyn.* **58**, 397–431 (2023)
20. He, L., Pan, Y., He, Y., Li, Z., Krolczyk, G., Du, H.: Control strategy for vibration suppression of a vehicle multibody system on a bumpy road. *Mech. Mach. Theory* **174**, 104891 (2022)
21. Hidalgo, A.F., García de Jalón, J.: Real-time dynamic simulations of large road vehicles using dense, sparse, and parallelization techniques. *J. Comput. Nonlinear Dyn.* **10**(3), 031005 (2015)
22. Jalali, S., Ahmadian, S., Khodayar, M., Khosravi, A., Ghasemi, V., Shafie-khah, M., Nahavandi, S., Catalão, J.: Towards novel deep neuroevolution models: chaotic Levy grasshopper optimization for short-term wind speed forecasting. *Eng. Comput.* **38**, 1787–1811 (2022)
23. Khan, T.A., Ling, S.H.: A novel hybrid gravitational search particle swarm optimization algorithm. *Eng. Appl. Artif. Intell.* **102**, 104263 (2021)
24. Kim, D., Min, K., Kim, H., Huh, K.: Vehicle sideslip angle estimation using deep ensemble-based adaptive Kalman filter. *Mech. Syst. Signal Process.* **144**, 106862 (2020)
25. Li, L., Jia, G., Ran, X., Song, J., Wu, K.: A variable structure extended Kalman filter for vehicle sideslip angle estimation on a low friction road. *Veh. Syst. Dyn.* **52**(2), 280–308 (2014)
26. Li, M., Si, W., Ren, Q., Song, L., Liu, H.: An integrated method for evaluating and predicting long-term operation safety of concrete dams considering lag effect. *Eng. Comput.* **37**, 2505–2519 (2021)
27. Liao, Y.W., Borrelli, F.: An adaptive approach to real-time estimation of vehicle sideslip, road bank angles, and sensor bias. *IEEE Trans. Veh. Technol.* **68**(8), 7443–7454 (2019)
28. Liu, J., Wang, Z., Zhang, L., Walker, P.: Sideslip angle estimation of ground vehicles: a comparative study. *IET Control Theory Appl.* **14**(20), 3490–3505 (2020)
29. Melzi, S., Sabbioni, E.: On the vehicle sideslip angle estimation through neural networks: numerical and experimental results. *Mech. Syst. Signal Process.* **25**(6), 2005–2019 (2011)
30. Min, C., Pan, Y., Dai, W., Kawsar, I., Li, Z., Wang, G.: Trajectory optimization of an electric vehicle with minimum energy consumption using inverse dynamics model and servo constraints. *Mech. Mach. Theory* **181**, 105185 (2023)
31. Nguyen, H., Bui, X.N., Hieu, T., Nguyen, H., Nguyen Dinh, A., Thi Thu Hoa, L., Lê, Q.: Prediction of ground vibration intensity in mine blasting using the novel hybrid MARS-PSO-MLP model. *Eng. Comput.* **38**, 4007–4025 (2022)
32. Nie, X., Min, C., Pan, Y., Li, Z., Krolczyk, G.: An improved deep neural network model of intelligent vehicle dynamics via linear decreasing weight particle swarm and invasive weed optimization algorithms. *Sensors* **22**, 4676 (2022)

33. Nie, X., Min, C., Pan, Y., Li, K., Li, Z.: Deep-neural-network-based modelling of longitudinal-lateral dynamics to predict the vehicle states for autonomous driving. *Sensors* **22**, 2013 (2022)
34. Pan, Y., Callejo, A., Bueno, J.L., Wehage, R.A., García de Jalón, J.: Efficient and accurate modeling of rigid rods. *Multibody Syst. Dyn.* **40**(1), 23–42 (2017)
35. Pan, Y., He, Y., Mikkola, A.: Accurate real-time truck simulation via semirecursive formulation and Adams–Bashforth–Moulton algorithm. *Acta Mech. Sin.* **35**, 641–652 (2019)
36. Pan, Y., Dai, W., Xiong, Y., Xiang, S., Mikkola, A.: Tree-topology-oriented modeling for the real-time simulation of sedan vehicle dynamics using independent coordinates and the rod-removal technique. *Mech. Mach. Theory* **143**, 103626 (2020)
37. Pan, Y., Huang, L., Dai, W., Zhao, J., Yu, X., Mikkola, A.: Rod-removal technique for flexible-rods in the framework of semi-recursive multibody formulation. *Mech. Mach. Theory* **169**, 104625 (2022)
38. Pan, Y., Sun, Y., Min, C., Li, Z., Gardoni, P.: Maneuver-based deep learning parameter identification of vehicle suspensions subjected to performance degradation. *Veh. Syst. Dyn.* **61**, 1–17 (2022)
39. Pan, Y., Sun, Y., Li, Z., Gardoni, P.: Machine learning approaches to estimate suspension parameters for performance degradation assessment using accurate dynamic simulations. *Reliab. Eng. Syst. Saf.* **230**, 108950 (2023)
40. Park, G.: Vehicle sideslip angle estimation based on interacting multiple model Kalman filter using low-cost sensor fusion. *IEEE Trans. Veh. Technol.* **71**(6), 6088–6099 (2022)
41. Rahman, M.A., Venayagamoorthy, G.K.: A hybrid method for power system state estimation using cellular computational network. *Eng. Appl. Artif. Intell.* **64**, 140–151 (2017)
42. Rajamani, R., Phanomchoeng, G., Piyabongkarn, D., Lew, J.Y.: Algorithms for real-time estimation of individual wheel tire-road friction coefficients. *IEEE/ASME Trans. Mechatron.* **17**(6), 1183–1195 (2012)
43. Schwerin, R.V.: *Multibody System Simulation, Numerical Methods, Algorithms and Software*. Springer, Berlin (1999)
44. Sieberg, P.M., Blume, S., Reicherts, S., Maas, N., Schramm, D.: Hybrid state estimation – a contribution towards reliability enhancement of artificial neural network estimators. *IEEE Trans. Intell. Transp. Syst.* **23**(7), 6337–6346 (2022)
45. Strano, S., Terzo, M.: Constrained nonlinear filter for vehicle sideslip angle estimation withno a priori knowledge of tyre characteristics. *Control Eng. Pract.* **71**, 10–17 (2018)
46. Tuerxun, W., Xu, C., Guo, H., Guo, L., Zeng, N., Gao, Y.: A wind power forecasting model using lstm optimized by the modified bald eagle search algorithm. *Energies* **15**, 2031 (2022)
47. Wang, H., Liu, B., Qiao, J.: Advanced high-speed lane keeping system of autonomous vehicle with sideslip angle estimation. *Machines* **10**(4), 257 (2022)
48. Wrobel, K., Doroz, R., Porwik, P., Naruniec, J., Kowalski, M.: Using a probabilistic neural network for lip-based biometric verification. *Eng. Appl. Artif. Intell.* **64**, 112–127 (2017)
49. Xia, X., Xiong, L., Lu, Y., Gao, L., Yu, Z.: Vehicle sideslip angle estimation: fusion of vehicle kinematics and dynamics. *Int. J. Veh. Des.* **87**, 73–94 (2021)
50. Xia, X., Hashemi, E., Xiong, L., Khajepour, A.: Autonomous vehicle kinematics and dynamics synthesis for sideslip angle estimation based on consensus Kalman filter. *IEEE Trans. Control Syst. Technol.* **31**(1), 179–192 (2023)
51. Zeng, J., Roy, B., Kumar, D., Mohammed, A., Jahed Armaghani, D., Zhou, J., Mohamad, E.: Proposing several hybrid pso-extreme learning machine techniques to predict tbm performance. *Eng. Comput.* **38**, 3811–3827 (2022)
52. Zhang, B., Du, H., Lam, J., Zhang, N., Li, W.: A novel observer design for simultaneous estimation of vehicle steering angle and sideslip angle. *IEEE Trans. Ind. Electron.* **63**(7), 4357–4366 (2016)
53. Zhang, Q., Jing, H., Liu, Z., Jiang, Y., Gu, M.: A novel PWA lateral dynamics modeling method and switched T-S observer design for vehicle sideslip angle estimation. *IEEE Trans. Ind. Electron.* **69**(2), 1847–1857 (2022)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.