

Research Article

A Car-Following Model considering the Effect of Following Vehicles under the Framework of Physics-Informed Deep Learning

Le Xu ¹, Jianxiao Ma ¹ and Yuchen Wang ²

¹College of Automobile and Traffic Engineering, Nanjing Forestry University, Nanjing 210037, China

²School of Transportation, Southeast University, Nanjing 211189, China

Correspondence should be addressed to Jianxiao Ma; majx@njfu.edu.cn

Received 19 May 2022; Accepted 5 August 2022; Published 31 August 2022

Academic Editor: Yang Yang

Copyright © 2022 Le Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Car-following models have been studied for a long time, and many traffic engineers and researchers have devoted attention to them. With the increase in machine learning, this paper proposes a fusion model based on the physics-informed deep learning framework. The purpose of this paper is to inherit the predecessors' ideas, transform them to fit a new framework, and improve the framework's accuracy. The IDM-D (intelligent driver model development) involves reenabling the effect of the following vehicle to form a complementary model (not car-following model) with the IDM (intelligent driver model). The pretreated NGSIM data are used for calibration and validation. The IDM and the IDM-D are combined with the LSTM under the framework of physics-informed deep learning, and the results are mixed in a ratio to form the final result. Using test data for simulation, the results reveal that the IDM-informed LSTM shows better performance than the LSTM and that the fusion model further improves the MSE (mean square error) of the IDM-informed LSTM. The fusion increases the accuracy during the deceleration process, which is better than just a single IDM-informed LSTM. The fusion model further explains drivers' deceleration behaviors.

1. Introduction

As a typical traffic phenomenon, car-following (CF) behaviors have been studied since Greenshields' pioneering paper in 1935 [1], and other predecessors have also made many achievements. Because there are many predecessors' works, questions, such as why car-following behaviors should be studied all the time and what motivates researchers to proceed with the study, are asked. In the past, predecessors wanted to explain the reason for CF behaviors, but for now, the background of the connected and automated vehicles (CAVs) has an outcome and cannot occupy the market in a short time. This means that man-driven vehicles (MDVs) have to coexist with CAVs for a long time. The MDVs cannot communicate with the CAVS, leading to the uncertainty of man-driven vehicles. This is a risk to traffic safety and a cause for making traffic inefficient. Using car-following models to predict CF behaviors

is a possible solution. Researchers focus more on accuracy rather than on the reasons. If the CAVS can learn the CF behaviors of the MDVs, then the CAVS can also learn to guide man-driven vehicles initiatively. Therefore, it is necessary to keep the research about the CF models. As an interdisciplinary subject, methods for studying CF behaviors are abundant. Merging technology, such as deep learning, can handle big data and mine it. Now is the right time for researchers to make a difference with a novel method to overcome the new challenge. Simultaneously for the multitransport modes, highway transport plays an important part, but its efficiency is easily affected by the traffic state, whereas the traffic state on the highway is significantly affected by the trip demand and microscopic traffic phenomenon [2].

As indicated in the literature, the CF phenomena are categorized by three modes: models based on physical theories [3, 4], models driven by data [5, 6], and models that

combine physics and are data driven [7, 8]. Traditionally, physics models take the basic inputs from reality, such as velocity, space headway, and the difference, between the preceding vehicle and objective vehicle. They are combined with complicated mathematics computations to develop formulas that are available for people with input data to calculate the results. They always have strong theories to support them in describing traffic phenomena [9, 10].

The data-driven model outcomes with the rise of machine learning also need inputs that are the same as the inputs of the physics model. Instead of using physics theories, data-driven models focus on hidden information in the data and pursue the logic between inputs and outputs. The physics model with the actual formula can be the white box. With limited time training the models, data-driven models can provide some outputs through the black box process. The black box process mines the relationship between the inputs and outputs [9, 10].

There are two types of fusion models. The first kind of fusion model combines the outputs of the physics models and the outputs of the data-driven models in different ratios [7]. This concept is similar to bagging in ensemble learning. The second kind of fusion model absorbs physics theories into machine learning, which means that the machine learns the laws of physics [8].

But the works above are not enough for the new context which pursues higher accuracy in simulating the car-following behaviors. To make sure the accuracy of the CAV prediction to car-following behaviors, and making safe and efficient decision for velocity change, a higher-accuracy car-following model is necessary. It is also the reason for the researchers in autonomous driving domains to continue to study the car-following behaviors. The physics-informed deep learning framework is proposed to make a deep learning model to learn physics rules, and Mo et al. take an implementation on studying CF behaviors. To further study the CF behaviors, and continue Mo's work to explain it with more detail, this paper reintroduces the following vehicle-related inputs to improve the accuracy based on the acknowledged model by using the thought of the fusion model and the framework of physics-informed deep learning to form the new model with higher accuracy. Therefore, the preceding vehicle in the previous research is still the preceding vehicle, the following vehicle in the previous research is the objective vehicle in this article, as well as the follower of the following vehicle in the previous research is the following vehicle in this article. To inherit the predecessors' ideas and fit the new framework, this paper uses the model to describe the reason why drivers continue to accelerate even if they have reached the ideal velocity by considering the effect of the following vehicle. This passage evaluates the different results for using the following vehicle-related inputs or not.

The rest of the paper is organized as follows: Section 2 introduces the related works, and Section 3 demonstrates the framework used in this paper, as well as the related method. Section 4 presents all steps of the experiment and analyzes the results. Finally, Section 5 presents the conclusion and discussion.

2. Related Works

2.1. Standard Inputs Model. The car-following model has been studied for a long time. As the pioneer in exploring car-following behaviors, Reuschel [11] and Pipes [12] proposed the stimulus-response car-following model. The safe-distance theory was proposed in 1958 [13]. It claimed that drivers kept a fine distance from the rear collision, and there were some models inheriting this concept. A new car-following model was proposed considering the limitation of drivers' acceleration and deceleration rate [14], and the predicted value was changed into velocity. As a famous model, the optimal velocity model (OVM) was proposed by considering the safe velocity of the drivers [3]. The OVM had a fatal error in the simulation, which resulted in unrealistic deceleration and excessively high acceleration. The shortcomings of the OVM have led more researchers to study them and propose many improvements to the OVM. The generalized force model (GFM) [15] is proposed to solve the problem brought about by the OVM. The GFM set a term on the right-hand side (RHS) to consider the effect of deceleration. To develop the GFM, the full velocity difference model (FVDM) was proposed [16], and the FVDM considered both acceleration and deceleration effects. Another well-known model, the intelligent driver model (IDM) [4], was proposed, which set the upper limit of acceleration and absorbed part of the safe-distance theory. The IDM was a comprehensive model that was widely used.

The rise of machine learning also includes its application to car-following situations resulting in data-driven car-following models as documented in Table 1.

$h_{obj}(t)$ represents the space headway of the objective vehicle, $v_{obj}(t)$ represents the velocity of the objective vehicle, $\Delta v(t)$ represents the velocity difference between the objective vehicle and the preceding vehicle, $\Delta v(t + \Delta t)$ represents the velocity changed in the next moment, and $\alpha(t + \Delta t)$ represents the acceleration in the next moment. There is a specification that some data-driven models use time sequence data as the inputs and this article uses a time series of something in Table 1 to express it.

The fusion models are divided into two types. The first kind of fusion model is decision fusion, meaning that using a method to combine the outputs of the physics model and the outputs of the data-driven model in a different ratio. Yang et al. [7] proposed the Gipps-RF and Gipps-BPNN models. The Gipps-RF model combined the Gipps models with random forest in different ratios, and the Gipps-BPNN model was used as the same method when combined. Both models improved their safe level and robustness. Li et al. [29] used the improved Kalman filter to combine the IDM and LSTM, and the result obtained field data, which reflected the trajectory of vehicles. The second kind of fusion model embeds physics knowledge into machine learning. Yun et al. [30] proposed a method to turn the physics model into physics with regularization and embed it into a Gaussian process (GP) to improve the accuracy of modeling. Then, Mo et al. [8] proposed a physics-informed deep learning framework (PIDL) based on a physics-informed neural network (PINN). They applied it to model car-following

TABLE 1: The review of the data-driven model.

Method	Inputs	Outputs	Study
Neutral network	$h_{obj}(t), v_{obj}(t)$	$\Delta v(t + \Delta t)$	Kehtarnavaz et al. [17]
	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$\alpha(t + \Delta t)$	Ma [18]
	$h_{obj}(t), v_{pre}(t)$	$v(t + \Delta t)$	Panwai et al. [19]
SVM	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$v(t + \Delta t)$	Wei et al. [20]; Zhang et al. [21]
RNN	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$\alpha(t + \Delta t)$	Zhou et al. [6]
LSTM	A time series of $h_{obj}(t), v_{obj}(t), \Delta v(t)$	$v(t + \Delta t)$	Wang et al. [22]; Gu et al. [23]; Lin et al. [24]
	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$v(t + \Delta t)$	Huang et al. [25]
	A time series of $h_{obj}(t), v_{obj}(t), \Delta v(t)$	A time series of $\alpha(t + \Delta t)$	Ma et al. [26]
DRL	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$\alpha(t + \Delta t)$	Zhu et al. [27]
GAIL	$h_{obj}(t), v_{obj}(t), \Delta v(t)$	$\alpha(t + \Delta t)$	Kuefler et al. [28]

behaviors. The method enabled the deep learning models to learn physics theory. This is also the point of this paper.

The model is talked about above usually uses the velocity, velocity difference, and time gap as input; we call it standard inputs model. There are also abundant inputs being not mentioned like leader acceleration (on visual like braking light), followers' velocity, followers' horn, and wipers considering the limitation of the length and focal point of this paper.

2.2. The Following Vehicle Domain-Related Inputs Model.

Scholars are not always set on traditional thought, and some specific inputs have been proposed. This part of the paper focuses on the specific inputs proposed by other researchers. Hayakawa and Nakanishi [31] proposed an improved OV model that considered the space between the following vehicle and the objective vehicle. Hasebe et al. [32] proposed an improved OV model that considered the k preceding vehicles and the k following vehicles. Ge et al. [33] continued Hasebe's research and set a ratio for the preceding vehicle and the following vehicle. They noticed that the driver looked forward differently than when looking backward. They also inspired this paper.

The above model had specific inputs based on the preceding and following two vehicles, which also had a shortcoming to a certain extent, but it still inspired the following study. Returning to 1950, Reuschel's paper described the platoon operation, most car-following models avoided the effect of following vehicles and set the assumption. However, vehicles always operate in platoons in the real world, and the effect of the third vehicle should be discussed. Some scholars have explored the effect of the third vehicle. The vehicles were affected by the following vehicle. Zeng et al. [34] used an OVM to conduct a simulation to explore the effect of the vehicle and found that the driver of the preceding vehicle focused more on the information of the following vehicle with less instability in the traffic flow. The predecessors had done much research on it, and this paper describes continuing the works of the predecessors to apply them to the new framework.

In summary, it is clear that predecessors have done great work with different models. Physics models express drivers' behaviors, data-driven models show great accuracy for human-driven vehicle trajectory data, and fusion models

show outstanding performance when fusing the two types of models. In addition, other inputs have been conducted as the new inputs, which made some difference in the past. However, the new method in recent years seems to default the inputs and does not consider more conditions for several reasons. Therefore, this paper uses the following vehicle effect as extra inputs and transforms it to fit the new method to propose a model with higher accuracy.

3. Methodology

In this section, we consider the reality road state and propose a complementary model to append the traditional IDM model which is called IDM-D. The IDM-D model is set as the key to connecting the new methodology which is presented in Section 2.1 and the past idea which is presented in Section 2.2. The LSTM and the framework of physically informed deep learning (PIDL) are introduced briefly.

With the key applied in the new method, a new model called IDM-D-informed LSTM is obtained. In the way of linear combination with fixed constant coefficient, the IDM-informed LSTM and IDM-D-informed LSTM are formed into the fully expressed IDM-informed LSTM hybrid model (FEHM).

3.1. IDM Model and IDM-D Model.

Before we talked about the model, the car-following events in this paper should be declared. The preceding vehicle in the traditional car-following events is still the preceding vehicle, the following vehicle in traditional car-following events is defined as objective vehicle in this paper, while the following vehicle refers to the follower of the following vehicle in traditional car-following events.

The IDM model has been widely applied for simulating human driving behaviors and expressing car-following behaviors in past studies [4, 11, 30, 35]. Compared to other behavioral models, the IDM is beneficial for input normalization with data-driven models, and the IDM fits the field data better in Mo's experiment [4]. The IDM model is shown as follows.

$\alpha(t + \Delta t)$ represents the acceleration at $t + \Delta t$, α_{\max} represents the max acceleration, v_{obj} represents the objective vehicle velocity, v_i *de al* represents the ideal velocity of the objective vehicle, s^* represents the objective vehicle

safe space headway to the preceding vehicle, $sobj$ represents the objective vehicle space headway, s_0 represents static safety distance, T represents safe time headway, Δv represents the velocity difference between the objective vehicle and the preceding vehicle, and b_{\max} represents the max deceleration.

$$\alpha(t + \Delta t) = \alpha_{\max} \left[1 - \left(\frac{vobj(t)}{videal} \right)^\delta - \left(\frac{s * (vobj(t), \Delta v(t))}{sobj(t)} \right)^2 \right],$$

$$s * (vobj(t), \Delta v(t)) = s_0 + T vobj(t) + \frac{vobj(t) \Delta v(t)}{2 \sqrt{a_{\max} b_{\max}}}. \quad (1)$$

By disassembling the IDM formula in two parts, $\alpha_{\max} (1 - (vobj(t)/videal)^\delta)$ is obtained as Part 1, which represents the willingness of the driver, and $\alpha_{\max} (-s * (vobj(t), \Delta v(t))/sobj(t))^2$ is obtained as Part 2, which represents the constraint from the preceding vehicle.

By considering the effect of the following vehicle, the origin preceding vehicle is turned into an objective vehicle, and the origin objective vehicle is turned into the following vehicle. We assume that the objective vehicle has reached the ideal velocity without the preceding vehicle constraint, and the following vehicle that continues accelerating decreases the distance between the two vehicles. This situation makes the objective driver feel uncomfortable, but he cannot accelerate because each part of the IDM model cannot express this phenomenon. Therefore, we follow the idea of the IDM and propose a complementary model and call it IDM-D.

$$\alpha(t + \Delta t) = \alpha_{\max} \left[1 - \left(\frac{vobj(t)}{videal} \right)^\delta + \left(\frac{s * (vfol(t), \Delta v(t))}{sfol(t)} \right)^2 \right],$$

$$s * (vfol(t), \Delta v(t)) = s_0 + T vfol(t) + \frac{vfol(t) \Delta v(t)}{2 \sqrt{a_{\max} b_{\max}}}, \quad (2)$$

where $s *$ represents the objective vehicle safe space headway to the following vehicle and Δv represents the velocity difference between the objective vehicle and the following vehicle.

IDM-D is also divided into two parts as follows: $\alpha_{\max} (1 - (vobj(t)/videal)^\delta)$ is the same as the IDM in Part 1. Part 2 is changed as follows: $\alpha_{\max} (s * (vfol(t), \Delta v(t))/sfol(t))^2$, which represents the driving power of the following vehicle.

It needs to be declared that the IDM-D is not a car-following model but more like a chasing-effect model to reveal some human psychology.

Vehicles always operate in the form of platoons when the traffic flow is tremendous. The state of traffic can be shown as in Figure 1. Traditionally, the IDM can explain the behaviors in the platoon in most situations, but the assumption proposed in this paper cannot be expressed. The IDM-D is proposed to express the behaviors of the leader when the leader is willing to consider its follower, that is the IDM cannot be expressed in this paper. The leader and the tail are special because the leader has no preceding vehicle and the tail has no following vehicle. Traditional CF models assume no effect of the following vehicle, but the effect of the

following vehicle is considered and is taken as the assumption in this paper. Therefore, members in the platoon are affected by both the preceding vehicle and the following vehicle. By combining the IDM and the IDM-D in ratio, the willingness of the driver is retained, and the effect of both the preceding vehicle and the following vehicle can be controlled by the ratio. In this way, the behaviors of the members in the platoon can be expressed under the assumption in this paper.

3.2. LSTM. The physics model can explain the CF behaviors, but it has a poor performance in fitting the field data. Therefore, data-driven models are needed. Long short-term memory is an improved recurrent neural network (RNN) that is proposed to solve the vanishing gradient problem. In contrast, from the structure of the RNN, the LSTM has two lines for working. The main line recording the long-term memory is called the cell state and the subline recording the working memory is called the hidden state. The cell state keeps the long-term information from avoiding the vanishing gradient, and the hidden state works the same as hidden state of the RNN. The output gate lets the result from the short-term analysis to output. The procedure is shown as follows and the structure is shown in Figure 2.

The X_t denotes the input vectors, C_t denotes the cell memory, and H_t denotes the output LSTM units at a time t . F_t , I_t , and O_t denotes the gating vectors, respectively, for different gates, σ and \tanh denote formula (3) and (4), respectively. The W_{fx} , W_{fh} , W_{tx} , W_{th} , W_{cx} , W_{ch} , W_{ox} , and W_{oh} denote different weights, respectively, and the b_f , b_i , b_c , and b_o denote different biases, respectively.

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-x}}, \quad (3)$$

$$\text{tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \quad (4)$$

The forget gate is set to determine whether the cell state forgets the information from the last cell state by using a step signal, which is the parameter F_t calculated by the formula as follows:

$$F_t = \text{tanh}(W_{fx} X_t + W_{fh} H_{t-1} + b_f). \quad (5)$$

The input gate decides whether the features extracted from new inputs are used to update the cell state by using a step signal, which is the parameter I_t calculated by the formula as follows:

$$I_t = \text{sigmoid}(W_{ix} X_t + W_{ih} H_{t-1} + b_i). \quad (6)$$

The extracted features come from formula (7).

$$\tilde{C}_t = \text{tanh}(W_{cx} X_t + W_{ch} H_{t-1} + b_c). \quad (7)$$

The cell state is updated in formula (8) as follows:

$$C_t = F_t \odot C_{t-1} + I_t \odot \tilde{C}_t. \quad (8)$$



FIGURE 1: Vehicles operating in the platoon.

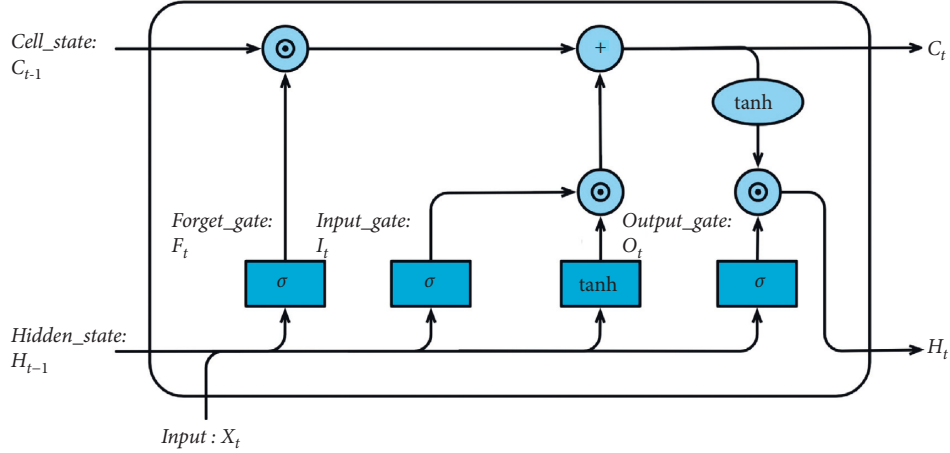


FIGURE 2: The details of the LSTM cell.

The output gate lets the result from the short-term analysis output. The result is combined with the cell state's result to form the new hidden state.

$$\begin{aligned} O_t &= \text{sigmoid}(W_{ox}X_t + W_{oh}H_{t-1} + b_o), \\ H_t &= O_t \odot \tanh(C_t). \end{aligned} \quad (9)$$

3.3. Physics-Informed Deep Learning Framework. A physics-informed neural network (PINN) is proposed to make the neural network learn the physics theory and solve the formula to obtain the result by adding a penalty function to the loss function of the neural network based on the physics theories [36]. In order for the PINN to be trained, the penalty function is taken as a part of the loss function. When the training is finished, a neural network (NN) that contains the physics theories is proposed. PINN has been used in many domains, which proves its feasibility.

PINN is adopted to fit the car-following issue, and a paradigm was proposed by Mo et al. [8]. The structure of the paradigm is shown in Figure 3, and it is called physically informed deep learning (PIDL). α_{observed} indicates the change in the observed velocity in 0.1 s. α_{output1} indicates the predicted change in the observed velocity in 0.1 s. α_{output2} indicates the predicted change in the physics velocity in 0.1 s. α_{physics} indicates the change in the physics velocity in 0.1 s.

PIDL provides a new method for fusing physics theory and data-driven machine learning, and it is applied in car-following behavior research, which provides a new direction for researchers to explore. With a brilliant paradigm, we develop it with a new physics formula and consider more conditions.

3.4. FEIDM-LSTM Hybrid Model (FEHM). The FEIDM-LSTM hybrid model is a fully expressed IDM-informed LSTM hybrid model. The FEHM is formed by the IDM, IDM-D, and LSTM under the framework of PIDL. The details of the FEHM are shown in Figure 4. The final $a(t + \Delta t)$ is obtained by combining the results of the IDM-informed LSTM and the IDM-D-informed LSTM in different ratios. β is the gain from the result of the IDM-informed LSTM.

In this passage, the PIDL framework is used twice. Inspired by [34], the first framework combines the IDM and the LSTM, which is called the IDM-informed LSTM. The second framework combines the IDM-D and the LSTM, which is called IDM-D-informed LSTM. Finally, the results of the two frameworks are combined with a ratio to forecast the acceleration of the objective vehicle as

$$\begin{aligned} a_{fin}(t + \Delta t) &= \beta * a_{IDM-LSTM}(t + \Delta t) \\ &+ (1 - \beta) * a_{IDM-D-LSTM}(t + \Delta t), \end{aligned} \quad (10)$$

where $a_{fin}(t + \Delta t)$ represents the final acceleration in the next moment, $a_{IDM-LSTM}(t + \Delta t)$ represents the acceleration in the next moment predicted by IDM-informed LSTM, and $a_{IDM-D-LSTM}(t + \Delta t)$ represents the acceleration in the next moment predicted by IDM-D-informed LSTM.

The physical model in this paper like IDM or IDM-D, plays an important role in training the LSTM. In other words, LSTM needs a penalty to mimic human driving, this penalty is a driver (not the man who drives the vehicle, but the result in section 4.3 shows it is not good enough for only IDM as penalty in deceleration, therefore we push out a IDM-D to boost the LSTM to learn to act more likely as

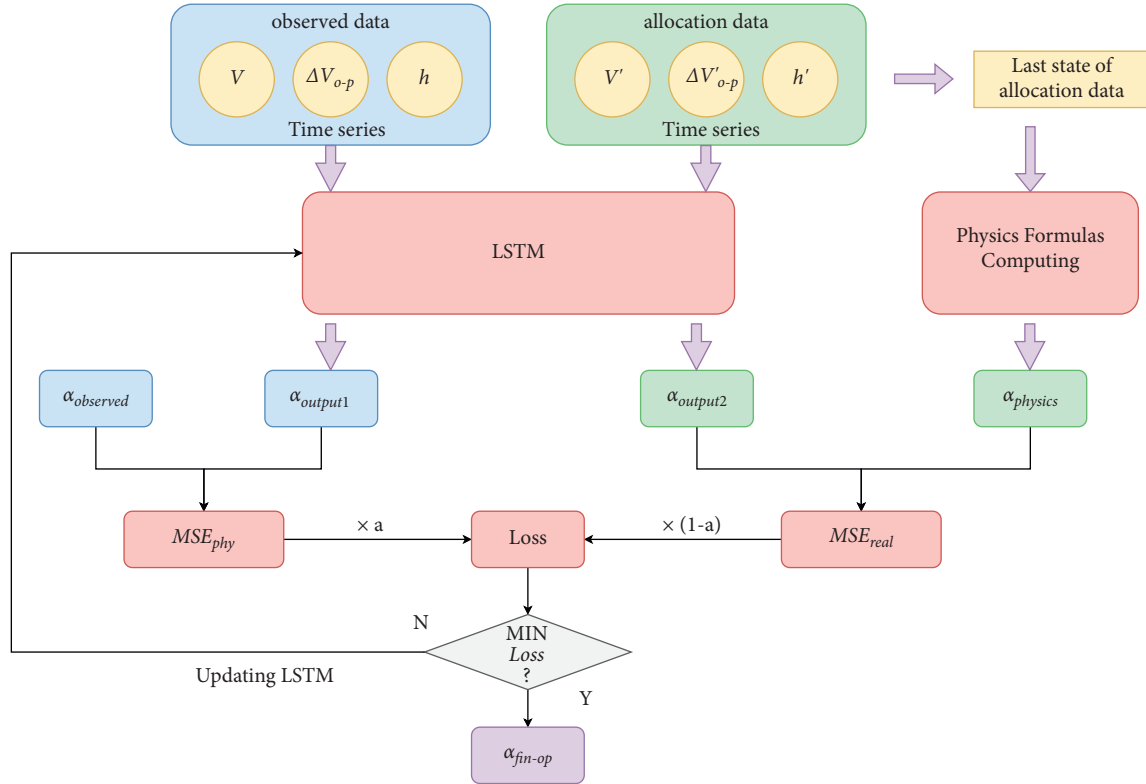


FIGURE 3: The details of the physics-informed deep learning framework.

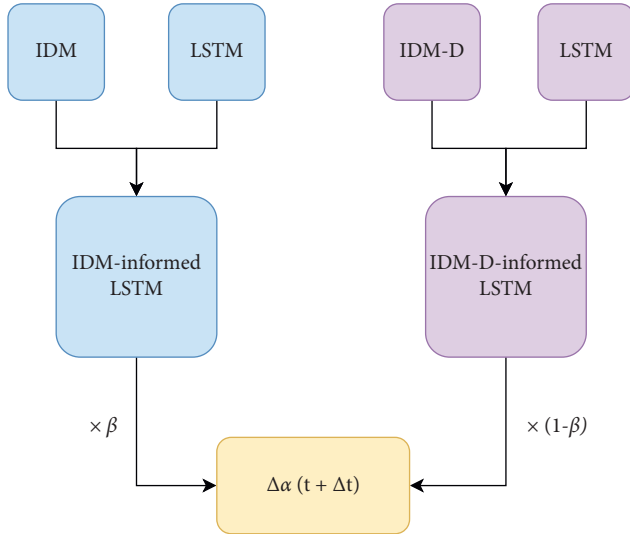


FIGURE 4: The structure of the FEHM.

human; we admit IDM-D got its shortcoming but it can make LSTM learning to perform better on mimicking human driving in the deceleration).

4. Numerical Experiment

In this section, we use the NGSIM dataset to calibrate the IDM parameters and train our IDM-informed deep learning model. The results are then compared with the LSTM model and some machine learning methods.

4.1. Data Preparation. In contrast to the common machine learning methods, deep learning is more complicated and confused with many parameters to calibrate. Thus, massive data are needed. The NGSIM datasets meet the above conditions and are available for us. Therefore, we decide to use it for observed data.

The NGSIM datasets were collected from a segment of Highway I-80 in the USA by a camera placed on the top of a high building. The recorded segment contained five lanes in the main lane, along with an auxiliary lane between an on-ramp and off-ramp. The datasets recorded the traffic flow and its time interval was 0.1 s. To leverage the effect of lane changing and capture as many car-following behaviors as possible, the median lane (lanes 1, 2, 3, and 4) vehicles' trajectories without lane changing were used.

After investigating the relevant research [35, 36], we decide to use the reconstructed I-80 dataset provided by Montanino and Punzo because their methods improve the accuracy of the NGSIM dataset and make it available for researchers to study the traffic phenomenons.

A total of 357650 samples (lanes 1, 2, and 4) are used for training, and 175658 samples (lane3) are used for testing. We show the details of the training data and test data in Tables 2–4.

The allocation data are also an important part of the PINN and represent unobserved data calculated by physical formulas. The difference between allocation data and observed data as the inputting part is shown in Figure 5. The orange points represent allocation points, which are also called unobserved points, and the blue points represent observed points.

TABLE 2: Detailed information on the velocity of objective vehicles in different lanes.

Velocity of objective vehicle	Lane 1	Lane 2	Lane 3	Lane 4
Max (m/s)	27.094	17.736	16.320	15.197
Min (m/s)	4.622	0.000	0.000	0.000
Mean (m/s)	16.178	7.1040	7.1062	6.387
Median (m/s)	16.238	7.3423	7.3529	6.432
STD variance	3.534	2.5318	2.3170	2.598

TABLE 3: Detailed information on the velocity of the preceding vehicle in different lanes.

Velocity of preceding vehicle	Lane 1	Lane 2	Lane 3	Lane 4
Max (m/s)	26.997	15.751	15.205	14.456
Min (m/s)	4.622	0.102	0.100	0.100
Mean (m/s)	16.527	7.144	7.0643	6.314
Median (m/s)	16.547	7.417	7.280	6.332
STD variance	3.445	2.545	2.323	2.542

TABLE 4: Detailed information on the gap of the objective vehicle.

Space headway of objective vehicle	Lane 1	Lane 2	Lane 3	Lane 4
Max (m)	303.525	80.221	76.769	73.578
Min (m)	4.098	0.121	0.071	0.278
Mean (m)	33.673	13.372	13.676	12.477
Median (m)	27.731	11.817	12.195	10.880
STD variance	22.765	7.337	7.420	7.323

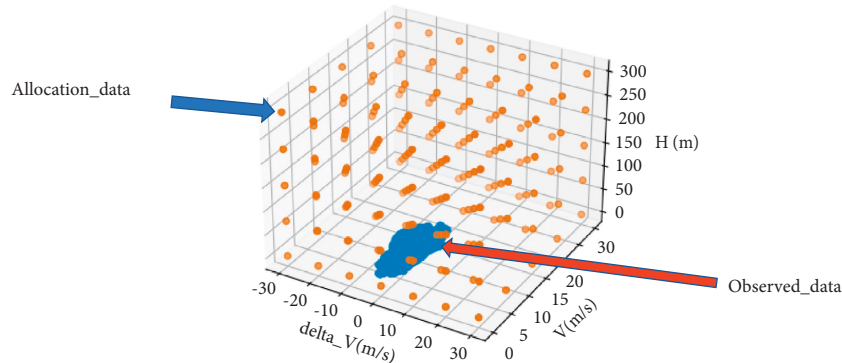


FIGURE 5: The descriptions of the allocation data and the observed data.

4.2. Model Training and Evaluation Method. The inputs of the model IDM are the objective vehicle velocity, the relative velocity (the objective vehicle velocity minus the preceding vehicle velocity), the space headway, and the allocation data.

The inputs of the model-IDM development are the objective vehicle velocity, the relative velocity (the objective vehicle velocity minus the following vehicle velocity), the space headway, and the allocation data. The time steps are set to 1 s (10 intervals of 0.1) by considering the time steps used in other papers [29]. To avoid overfitting, we set the learning rate to 0.001 and divide the training data into three parts. For part 1, we let each time step overlap 70% of the information and then overlap 30% in part 2. Finally, we set time steps without overlapping. Each part of the information overlapping uses one dimension as input, for example, and is shown in Figure 6.

The format of the training data has been changed to be more flexible. In a traditional case, like Figure 7, the format of training data makes less utilization of data (each vehicle is used for once). We transform the objective vehicle to the following vehicle and transform the preceding vehicle to the objective vehicle, which means we double the data for training. It is shown in Figure 8.

The whole framework was structured by Keras with TensorFlow as the backend. The CPU is i7-10700K. The structure of the LSTM is considered by reference [29] and it contains five hidden layers, which we set as 60, 100, 200, 300, and 100 neurons for each layer. Behind each layer, we set the dropout layer (0.2) to avoid overfitting. By considering the operation time and the accuracy, the epochs of training for each model are set to 25, and the mean training time for each epoch is 61 s.

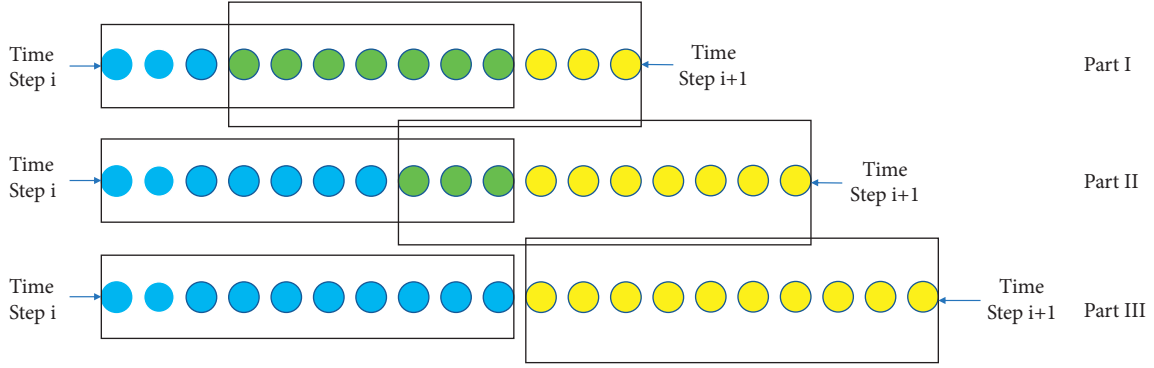


FIGURE 6: The details of the training process (taking one-dimension inputs as an example).



FIGURE 7: Traditional training data considering the effect of the following vehicle.

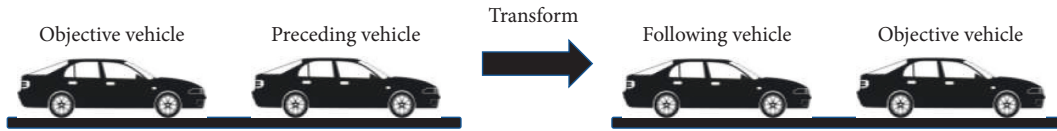


FIGURE 8: The transformation to fit the model in this passage.

The LOSS is adopted as an evaluation indicator for determining the parameter values of the model IDM and the model IDM development. It is formed by indicators MSE_{phy} and MSE_{real} , which illustrate the physical and realistic error between the trajectories and the field data, respectively. The α is set 0.7 following reference [8].

$a_i^{observe}$ indicates the observed acceleration in 0.1 s. $a_i^{sim-real}$ indicates the predicted change in the observed velocity in 0.1 s. $a_i^{sim-phy}$ indicates the predicted physics acceleration in 0.1 s. $a_i^{allocate}$ indicates allocation data's acceleration in 0.1 s.

$$MSE_{phy} = \frac{1}{N} \sum_{i=1}^N (a_i^{allocate} - a_i^{sim-phy})^2,$$

$$MSE_{real} = \frac{1}{N} \sum_{i=1}^N (a_i^{observe} - a_i^{sim-real})^2, \quad (11)$$

$$LOSS = \alpha MSE_{phy} + (1 - \alpha) MSE_{real}.$$

It is needed for declaring that the parameter in IDM and IDM-D is shared in our assumption. The parameters are shown in Table 5.

4.3. Analysis and Improvement. We use some machine learning methods to compare with our model. The ANN was constructed by Keras. It contains one input layer with three-dimensional inputs, one hidden layer with 64 neurons, and

TABLE 5: The ground truth value of each parameter.

Parameters	Value
$\alpha (m/s^2)$	0.73
$T (s)$	1.5
$v_{ideal} (m/s)$	30
$S_0 (m)$	2
$b (m/s^2)$	1.63

one output layer with one dimension. The data are converted by the “Max_Min_scale.” Then, we split it into training and testing data with the “train_test_split” of the “sklearn.” The model is fitted, and test data are evaluated.

The KNN is realized by “KNeighborsRegressor” in python. To consider fairness, the dataset is cut by vehicle IDs. This means that the train data do not have an identical ID vehicle in the test data. The ratio of the training data and the testing data is almost 2 : 1, and the neighbors' number is set to 10 according to [5].

Through the indicator loss, we can determine that the IDM-informed LSTM model is better than the other machine learning algorithm in Table 6.

As we consider the effect of the following vehicle, the IDM-D-informed deep learning is combined with the IDM-informed LSTM to conduct a fusion model. We assume that the IDM-informed LSTM model learns the willingness of the driver and the effect of the preceding vehicle. Additionally, the IDM-D-informed LSTM model learns the same willingness of the driver as the IDM-informed LSTM model and the effect of the following vehicle. Through the simple linear

TABLE 6: The results of different models.

	LSTM	IDM-informed LSTM	ANN	KNN
MSE/LOSS	0.00931	0.00704	0.04924	0.01809

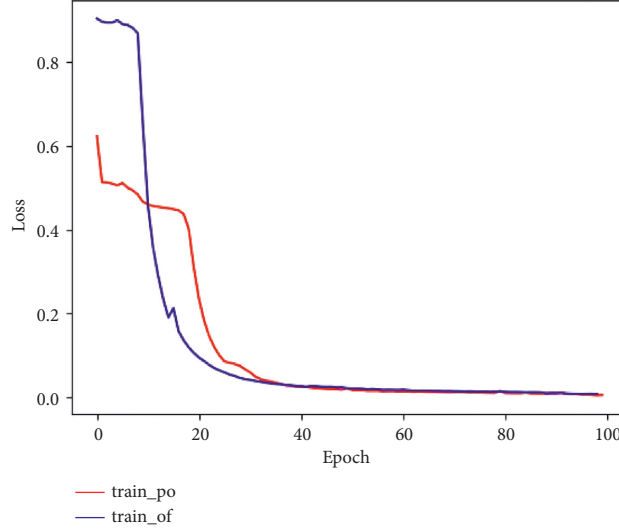


FIGURE 9: The train loss curves.

TABLE 7: Results of the different ratios in the test.

Different ratio β in test	Loss in test
0	0.01048
0.1	0.00933
0.2	0.00847
0.3	0.00772
0.4	0.00715
0.5	0.00673
0.6	0.00648
0.7	0.00638
0.8	0.00644
0.9	0.00666
1	0.00704

relation, we try to find a balance to fit the effect of the following vehicle and the preceding vehicle on the driver. The formula and result are shown as follows:

$$R_{fin} = \beta RI_{DM-LSTM} + (1 - \beta) RI_{DM-D-LSTM}, \quad (12)$$

where R_{fin} represents the final result, $RI_{DM-LSTM}$ represents the result of the IDM-informed LSTM, $RI_{DM-D-LSTM}$ represents the result of the IDM-D-informed LSTM, and β is a coefficient between 0 and 1.

The training loss curves for the two models are shown in Figure 9. When the ratio β is set as 0.7 in Table 7, the test loss is the lowest. When the ratio β reaches 0.7, the loss changes slightly. To improve the generalizability of our model, we decide to use 0.7 as the value of β .

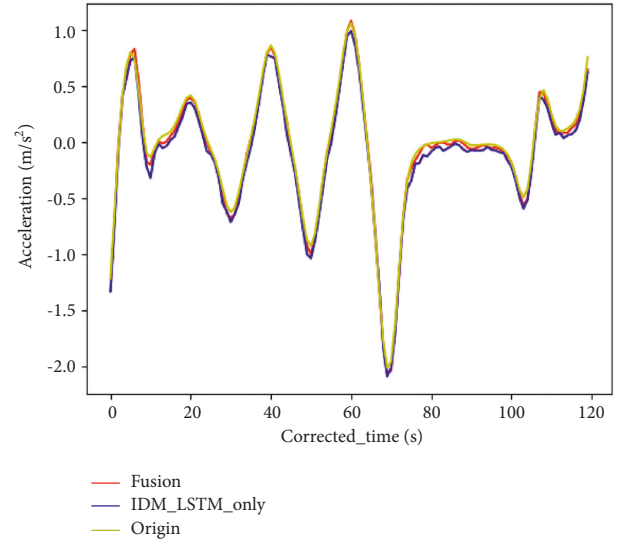


FIGURE 10: Car-following sampled acceleration of ID 121 generated by IDM-informed LSTM and fusion model.

For example, for the ID 21 vehicle in Figure 10, the fusion result shows better performance when the objective vehicle struggles with the deceleration process and better fits the origin velocity difference. The IDM-informed LSTM shows that the vehicles' deceleration is greater than the origin without considering the following vehicle, and the FEHM effect, to some extent, prevents the vehicle from decelerating and fitting the origin deceleration.

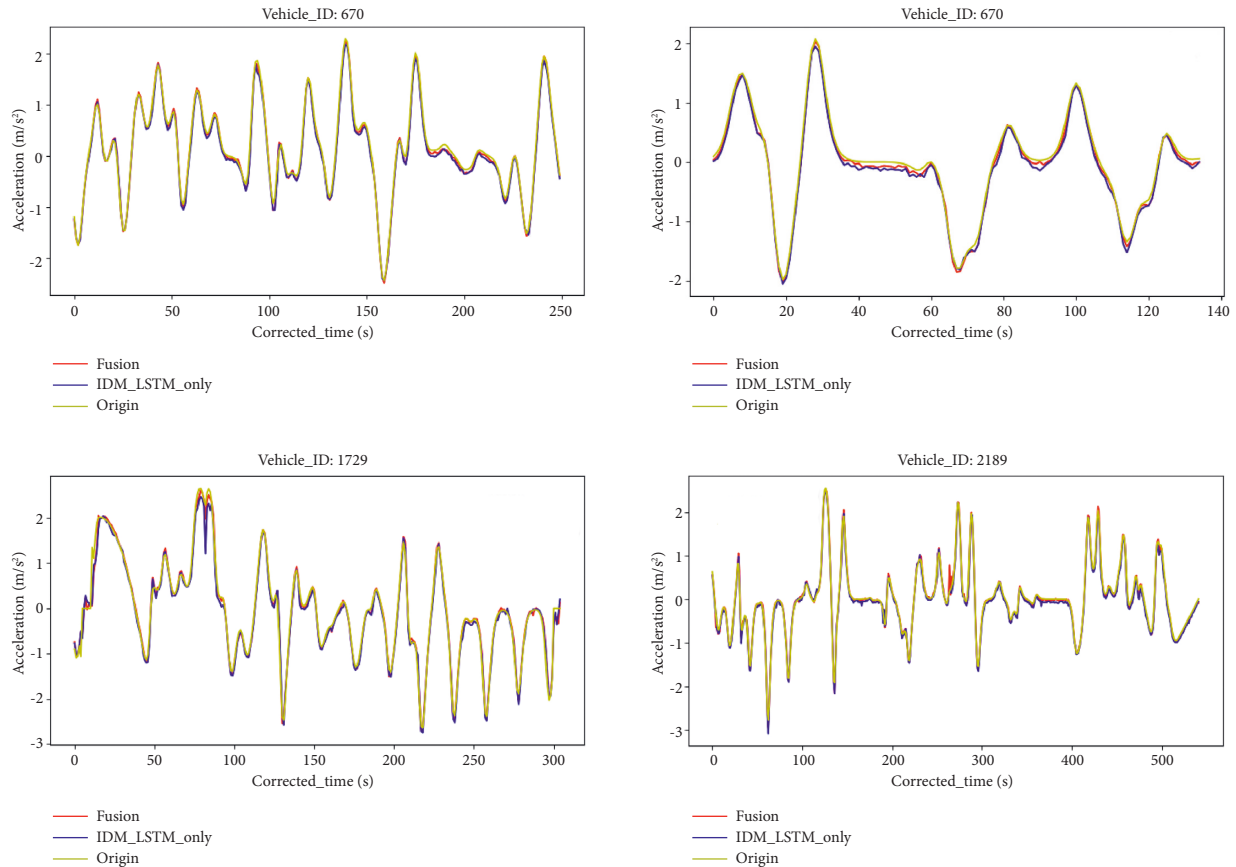


FIGURE 11: More examples of comparisons of the IDM-informed LSTM and the fusion model (vehicle_ID: 670,1073,1729,2189).

5. Conclusions and Discussion

To better capture and mimic human decisions on car following, an IDM-informed LSTM and IDM-D-informed LSTM fusion model (FEHM) is proposed in this paper. The fusion model incorporates the historical driving information powered by the LSTM and temporal driving decisions powered by the physics models. NGSIM I-80-reconstructed dataset is used to validate the fusion model.

Applying the physics-informed deep learning framework proposed by the predecessor, this paper has reproduced the brilliant results of the predecessor. However, this paper is not satisfied with the result, according to the principle of closely simulating the real manual driving data as much as possible, this article further improved the accuracy of the PIDL car following model. Therefore, this paper reenables the effect of the following vehicles and attempts to realize the effect based on the concept of the IDM. By proposing the IDM-D without a traditional validation, this paper expects it to work in expressing the acceleration of the objective vehicle caused by the following vehicle when chasing the objective vehicle. This phenomenon cannot be expressed by the IDM, and thus, physically informed deep learning lost the support of the theory. The IDM-D is applied in the experiment. By combining the results of the IDM-informed LSTM and the results of the IDM-D-informed LSTM in different ratios,

the final results show that the FEHM's result is better than that of the IDM-informed LSTM alone with a lower MSE.

The result of the IDM-informed LSTM shows worse performance in predicting the deceleration process. The solo IDM-informed LSTM has studied the effect of preceding vehicles without considering the following vehicles. Therefore, the prediction of its deceleration is out of the range of observed deceleration without considering the following vehicles, although it did receive a following vehicle. This proves the effect of the following vehicles and the feasibility of the IDM-D, even though it is imperfect. More details can be shown in Figure 11.

Furthermore, the IDM-D will be expected to be improved, and the ratio will be calculated by the reasonable formula instead of the rough estimation. The nonlinear relationship between the 2 models is expected to be studied.

The purpose of this paper is not to propose a new and advanced method; instead, this paper inherits predecessors' ideas and realizes the ideas based on the new technique. It is a great honor for the beginner to help the predecessor to improve and realize their ideas under a new framework.

Data Availability

The data used to support the findings of this study are included within the article. The researcher can use the constructed dataset is website link in the article to get the origin

data; by following the process in the article, the data used in the article will be obtained. The researchers can obtain the dataset in <https://www.dropbox.com/s/hyo0slm816m06hx/Reconstructed%20NGSIM%20I80-1%20data.zip?dl=0>.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by University Philosophy and Social Science Research Project of Jiangsu Province (2020SJA0133), Universities Natural Science Research Project of Jiangsu Province (20KJB580016), and The Second Batch of 2021 MOE of PRC Industry-University Collaborative Education Program (Program No. 202102055014, Kingfar-CES “Human Factors and Ergonomics” Program). The authors thank the generous sharing of Punzo and Montanino.

References

- [1] B. D. Greenshields, J. R. Bibbins, W. S. Channing, and H. H. Miller, “A study of traffic capacity,” in *Highway Research Board Proceedings*, vol. 1935, National Research Council (USA), Highway Research Board, USA, 1935.
- [2] W. Tang and D. M. Levinson, “Deviation between actual and shortest travel time paths for commuters,” *Journal of Transportation Engineering, Part A: Systems*, vol. 144, no. 8, Article ID 04018042, 2018.
- [3] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, “Dynamical model of traffic congestion and numerical simulation,” *Physical Review*, vol. 51, no. 2, p. 1035, 1995.
- [4] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical Review*, vol. 62, no. 2, p. 1805, 2000.
- [5] Z. He, L. Zheng, and W. Guan, “A simple nonparametric car-following model driven by field data,” *Transportation Research Part B: Methodological*, vol. 80, pp. 185–201, 2015.
- [6] M. Zhou, X. Qu, and X. Li, “A recurrent neural network based microscopic car following model to predict traffic oscillation,” *Transportation Research Part C: Emerging Technologies*, vol. 84, pp. 245–264, 2017.
- [7] D. Yang, L. Zhu, Y. Liu, D. Wu, and B. Ran, “A novel car-following control model combining machine learning and kinematics models for automated vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 1991–2000, 2018.
- [8] Z. Mo, X. Di, and R. Shi, “A physics-informed deep learning paradigm for car-following models,” 2020, <https://arxiv.org/abs/2012.13376>.
- [9] V. Papathanasopoulou and C. Antoniou, “Towards data-driven car-following models,” *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 496–509, 2015.
- [10] S. Hao, L. Yang, and Y. Shi, “Data-driven car-following model based on rough set theory,” *IET Intelligent Transport Systems*, vol. 12, no. 1, pp. 49–57, 2018.
- [11] A. Reuschel, “Vehicle movements in a platoon,” *Oesterreichisches Ingenieur-Archiv*, vol. 4, pp. 193–215, 1950.
- [12] L. A. Pipes, “An operational analysis of traffic dynamics,” *Journal of Applied Physics*, vol. 24, no. 3, pp. 274–281, 1953.
- [13] E. I. J. I. Kometani and T. S. U. N. A. Sasaki, “On the stability of traffic flow (report-I),” *Journal of the Operations Research Society of Japan*, vol. 2, no. 1, pp. 11–26, 1958, https://www.orsj.or.jp/%7Earchive/pdf/e_mag/Vol.02_01_011.pdf.
- [14] P. G. Gipps, “A behavioural car-following model for computer simulation,” *Transportation Research Part B: Methodological*, vol. 15, no. 2, pp. 105–111, 1981.
- [15] D. Helbing and B. Tilch, “Generalized force model of traffic dynamics,” *Physical Review*, vol. 58, no. 1, p. 133, 1998.
- [16] R. Jiang, Q. Wu, and Z. Zhu, “Full velocity difference model for a car-following theory,” *Physical Review*, vol. 64, no. 1, Article ID 017101, 2001.
- [17] N. Kehtarnavaz, N. Groszold, K. Miller, and P. Lascoe, “A transportable neural-network approach to autonomous vehicle following,” *IEEE Transactions on Vehicular Technology*, vol. 47, no. 2, pp. 694–702, 1998.
- [18] X. Ma, “A neural-fuzzy framework for modeling car-following behavior,” vol. 2, pp. 1178–1183, in *Proceedings of the IEEE international conference on systems, man and cybernetics*, vol. 2, IEEE, Taipei, Taiwan, 2006, October.
- [19] S. Panwai and H. Dia, “Neural agent car-following models,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 60–70, 2007.
- [20] D. Wei and H. Liu, “Analysis of asymmetric driving behavior using a self-learning approach,” *Transportation Research Part B: Methodological*, vol. 47, pp. 1–14, 2013.
- [21] J. Zhang, Y. Liao, S. Wang, and J. Han, “Study on driving decision-making mechanism of autonomous vehicle based on an optimized support vector machine regression,” *Applied Sciences*, vol. 8, no. 1, p. 13, 2018.
- [22] J. Wang, H. A. Rakha, and K. Fadhloun, *Comparison of Car-Following Models: A Vehicle Fuel Consumption and Emissions Estimation Perspective (No. 17-01158)*, 2017.
- [23] Z. Gu, Z. Li, X. Di, and R. Shi, “An lstm-based autonomous driving model using a waymo open dataset,” *Applied Sciences*, vol. 10, no. 6, p. 2046, 2020.
- [24] Y. Lin, P. Wang, Y. Zhou, F. Ding, C. Wang, and H. Tan, “Platoon trajectories generation: a unidirectional interconnected LSTM-based car-following model,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, 2020.
- [25] X. Huang, J. Sun, and J. Sun, “A car-following model considering asymmetric driving behavior based on long short-term memory neural networks,” *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 346–362, 2018.
- [26] L. Ma and S. Qu, “A sequence to sequence learning based car-following model for multi-step predictions considering reaction delay,” *Transportation Research Part C: Emerging Technologies*, vol. 120, Article ID 102785, 2020.
- [27] M. Zhu, X. Wang, and Y. Wang, “Human-like autonomous car-following model with deep reinforcement learning,” *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 348–368, 2018.
- [28] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, “Imitating driver behavior with generative adversarial networks,” in *Proceedings of the IEEE Intelligent Vehicles Symposium*, no. IV, pp. 204–211, IEEE, Los Angeles, CA, USA, 2017, June.
- [29] Y. Li, X. Lu, C. Ren, and H. Zhao, “Fusion modeling method of car-following characteristics,” *IEEE Access*, vol. 7, Article ID 162778, 2019.
- [30] Y. Yun, Q. Wang, and X. T. Yang, “Modeling stochastic microscopic traffic behaviors: a physics regularized Gaussian

- process approach,” 2020, <https://arxiv.org/pdf/2007.10109.pdf>.
- [31] H. Hayakawa and K. Nakanishi, “Universal behavior in granular flows and traffic flows,” *Progress of Theoretical Physics - Supplement*, vol. 130, pp. 57–75, 1998.
 - [32] K. Hasebe, A. Nakayama, and Y. Sugiyama, “Dynamical model of a cooperative driving system for freeway traffic,” *Physical Review*, vol. 68, no. 2, Article ID 026102, 2003.
 - [33] H. X. Ge, H. B. Zhu, and S. Q. Dai, “Effect of looking backward on traffic flow in a cooperative driving car following model,” *European Physical Journal B: Condensed Matter and Complex Systems*, vol. 54, no. 4, pp. 503–507, 2006.
 - [34] Y.-Z. Zeng and Z. Ning, “Effects of comprehensive information of the nearest following vehicle on traffic flow instability,” *Acta Physica Sinica*, vol. 63, no. 21, 2014.
 - [35] V. Milanés and S. E. Shladover, “Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data,” *Transportation Research Part C: Emerging Technologies*, vol. 48, pp. 285–300, 2014.
 - [36] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.
 - [37] M. Treiber and D. Helbing, “Hamilton-like statistics in one-dimensional driven dissipative many-particle systems,” *The European Physical Journal B*, vol. 68, no. 4, pp. 607–618, 2009.
 - [38] M. Montanino and V. Punzo, “Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns,” *Transportation Research Part B: Methodological*, vol. 80, pp. 82–106, 2015.
 - [39] V. Punzo, M. T. Borzacchiello, and B. Ciuffo, “On the assessment of vehicle trajectory data accuracy and application to the Next Generation SIMulation (NGSIM) program data,” *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 6, pp. 1243–1262, 2011.