

Modeling Electric Vehicle Car-Following Behavior: Classical vs Machine Learning Approach

Md. Shihab Uddin*, Md Nazmus Shakib*, and Rahul Bhadani*

*Electrical and Computer Engineering, The University of Alabama in Huntsville,
Huntsville, AL, USA

Emails: mu0018@uah.edu, nazmus.shakib@uah.edu, rahul.bhadani@uah.edu

Abstract

The increasing adoption of electric vehicles (EVs) necessitates an understanding of their driving behavior to enhance traffic safety and develop smart driving systems. This study compares classical and machine learning models for EV car-following behavior. Classical models include the Intelligent Driver Model (IDM), Optimum Velocity Model (OVM), Optimal Velocity Relative Velocity (OVRV), and a simplified CACC model, while the machine learning approach employs a Random Forest Regressor. Using a real-world dataset of an EV following an internal combustion engine (ICE) vehicle under varied driving conditions, we calibrated classical model parameters by minimizing the RMSE between predictions and real data. The Random Forest model predicts acceleration using spacing, speed, and gap type as inputs. Results demonstrate the Random Forest's superior accuracy, achieving RMSEs of 0.0046 (medium gap), 0.0016 (long gap), and 0.0025 (extra-long gap). Among physics-based models, CACC performed best, with an RMSE of 2.67 for long gaps. These findings highlight the machine learning model's performance across all scenarios. Such models are valuable for simulating EV behavior and analyzing mixed-autonomy traffic dynamics in EV-integrated environments.

Index Terms

Electric vehicle, Car-following behavior, Random Forest, IDM, OVRV, CACC, OVM

I. INTRODUCTION

Electric vehicles (EVs) are becoming increasingly common on roads around the world. As EV adoption grows, it becomes essential to understand how these vehicles behave in mixed traffic environments, particularly in car-following situations [1]. Car-following behavior plays a key role in road safety, traffic flow, and the development of advanced driver assistance systems [2]. Over the years, many car-following models have been developed to simulate and analyze driver behavior. Traditional physics-based models such as the Intelligent Driver Model (IDM) [3], Optimal Velocity Model (OVM) [4], and their extensions [5–7] offer interpretable frameworks based on physical and behavioral principles. These models allow for analytical insights into vehicle interactions and are often used for ensuring safety and stability in simulations [8].

The IDM describes acceleration using factors like desired velocity, spacing, and time headway. Similarly, the OVM and its relative velocity variant (OVRV) compute target speeds based on headway and speed differences [9, 10]. While these models are simple and interpretable, they may not fully capture the complexity and variability found in real-world driving scenarios. To address these limitations, machine learning (ML) models have gained popularity due to their ability to learn directly from data. Supervised learning techniques such as decision trees, random forests, and neural networks can predict acceleration or spacing based on features like speed, relative position, and gap type [11, 12]. ML models are particularly effective at modeling diverse driving behaviors, but they often lack physical interpretability and require large datasets for training [1] [8].

Recent research has explored hybrid approaches that integrate classical models with ML-based corrections. For example, residual learning frameworks use classical model predictions as a baseline and refine

them using machine learning techniques [13]. These approaches aim to combine the strengths of both interpretability and data-driven flexibility. In the context of connected and automated vehicles, Cooperative Adaptive Cruise Control (CACC) models have drawn attention for their ability to enhance platoon stability using vehicle-to-vehicle (V2V) communication [14]. Simplified versions of CACC can still be used in the absence of connectivity, relying on local measurements such as spacing and relative velocity. Empirical studies have shown that CACC can reduce traffic disturbances and improve flow [15].

Efforts like FollowNet have introduced benchmarking datasets that compare classical, ML-based, and hybrid car-following models under realistic driving conditions [16]. These studies help guide researchers in selecting models based on performance, interpretability, and adaptability to real-world variability. The field of car-following modeling has witnessed significant progress with the integration of advanced machine-learning techniques and hybrid approaches. For instance, the study in [17] introduced EnsembleFollower, a hierarchical framework that leverages reinforcement learning to dynamically select among multiple low-level car-following models, enhancing adaptability across diverse driving scenarios. In the realm of hybrid modeling, a recent study [18] introduced a stochastic hybrid car-following model utilizing approximate Bayesian computation. This approach probabilistically integrates multiple car-following models, offering a flexible framework that captures the stochastic nature of driver behavior.

An adaptive Kalman-based hybrid strategy was proposed [19], combining Twin Delayed Deep Deterministic Policy Gradient (TD3) with Cooperative Adaptive Cruise Control (CACC). This approach uses an adaptive coefficient derived from multi-timestep predictions to improve safety and ride comfort in mixed-traffic environments. Another recent study [1] introduced a hybrid car-following model that integrates kinetic dynamics with deep learning networks. By combining traditional mathematical frameworks with Long Short-Term Memory (LSTM) networks, the model effectively captures both general traffic patterns and individual driver characteristics, enhancing prediction accuracy and interpretability. These recent contributions underscore the ongoing evolution of car-following models, emphasizing the importance of hybrid approaches that blend physics-based insights with data-driven adaptability. Incorporating such models can lead to more robust and accurate representations of vehicle-following behavior, particularly in the context of electric vehicles operating in mixed-traffic environments.

This paper builds on these foundations by presenting a comparative analysis of classical and data-driven models for EV car-following behavior. We use a real-world dataset where an EV follows a conventional ICE vehicle under various gap settings. Classical models are calibrated by minimizing RMSE to observed data, while a Random Forest model is trained to predict acceleration based on spacing, speed, and gap type. The main objectives of this study are as follows:

- We use real-world EV experiment data to calibrate classical car-following models and find the best parameters for each model.
- We use these best parameters to simulate the classical models, and we also train a machine learning model to compare its performance with the classical models.
- Our study provides a comprehensive set of calibrated parameters offering valuable insights for future traffic simulation studies.

II. DATASET

To model the dynamic traffic behavior of EVs in a microscopic environment, we utilized real-world experimental trajectory data, which is commercially available for EV-ACC vehicles. The dataset was collected by Kan [20]. The experiment was conducted in a controlled environment, where the ICE vehicle acted as the lead vehicle (Toyota Camry, 2021), and it followed a predefined speed profile, and an EV equipped with ACC (Hyundai IONIQ 5, 2022) followed the leader's trajectory. The positions of both the leader and follower vehicles were recorded using a Racebox GPS logger, which captures data at a frequency of 25 Hz and achieves a horizontal accuracy of approximately 10 cm. Vehicle spacing was then derived by applying the Haversine distance formula to the recorded positional data.

Both the leader and follower started with predefined initial spacing and acceleration. After reaching the free-flow speed, the subject vehicle activated the ACC and occasionally adjusted the acceleration above the designated point to maintain the spacing equilibrium. Once the leader vehicle reached the equilibrium, it decelerated to a congested speed and maintained that speed for the next 10 seconds, after that it returned to the previous free-flow speed. The leader vehicle's free-flow speed was set at four different levels for each gap configuration: 60 mph, 55 mph, 45 mph, and 35 mph. This sequence of speed fluctuation was repeated 8 times, with two repetitions for each of four gap settings provided by the vehicle's ACC system: short, medium, long, and extra-long, a total of 136 speed fluctuations in the final dataset.

III. CLASSICAL MODELS

This section illustrates physics-based models used in the calibration and simulation using EV data.

A. Intelligent Driver Model (IDM)

The IDM is a widely adopted car-following model that computes acceleration based on spacing, speed, and desired time gap. The acceleration is given by:

$$a(t) = a_{\max} \left[1 - \left(\frac{v(t)}{v_0} \right)^\delta - \left(\frac{s^*(t)}{s(t)} \right)^2 \right] \quad (1)$$

where $v(t)$ is the current speed, v_0 is the desired speed, δ is the acceleration exponent and $s(t)$ is the actual spacing. The desired spacing $s^*(t)$ is defined as:

$$s^*(t) = s_0 + T \cdot v(t) + \frac{v(t) \cdot \Delta v(t)}{2\sqrt{a_{\max}b}} \quad (2)$$

where v_0 is the desired speed, T is the time headway, s_0 is the minimum spacing, $a_{\max}$ is the maximum acceleration, and b is the comfortable deceleration.

B. Optimal Velocity Model (OVM)

The OVM assumes that a driver adapts their acceleration based on the difference between the current speed and an optimal velocity determined by spacing:

$$a(t) = \alpha (v_{\text{opt}}(s) - v(t)) \quad (3)$$

where $v_{\text{opt}}(s)$ is typically modeled as a sigmoid function:

$$v_{\text{opt}}(s) = v_{\max} \cdot \tanh \left(\frac{s}{s_{st}} - 2 \right) \quad (4)$$

where α is sensitivity parameter, $v_{\max}$ is the maximum velocity, and s_{st} is the spacing scale factor.

C. Optimal Velocity Relative Velocity Model (OVRV)

The OVRV model extends OVM by incorporating the effect of the relative velocity between the lead and the following vehicles:

$$a(t) = k_1 (s(t) - \eta - \tau \cdot v(t)) + k_2 (v_l(t) - v(t)) \quad (5)$$

Here, k_1 and k_2 are sensitivity parameters, η is the spacing offset, τ is a delay-like factor, and $v_l(t)$ is the leader's speed.

D. Cooperative Adaptive Cruise Control (CACC)

The CACC model incorporates cooperative behavior using a spacing control law and relative velocity. We use a simplified version that excludes leader acceleration:

$$a(t) = k_1 (v_l(t) - v(t)) + k_2 (s(t) - (s_0 + T \cdot v(t))) \quad (6)$$

Here, k_1 controls the response to speed difference, and k_2 scales the correction to spacing error. The desired spacing is modeled as $s_0 + T \cdot v(t)$, where T is the time gap and s_0 is the minimum spacing. Their performance is evaluated in Section VI.

IV. MODEL CALIBRATION

In this section, we used real-world trajectory data of EVs to ensure the fair evaluation of classical car-following models.

TABLE I: Best-calibrated parameters and RMSE across Gap Settings

Models	Medium Gap		Long Gap		XLong Gap	
	Parameters	RMSE	Parameters	RMSE	Parameters	RMSE
IDM	[22.12, 1.29, 1.73, 2.00, 2.00]	93.79	[20.00, 1.29, 3.81, 2.00, 2.00]	207.90	[20.15, 1.41, 4.71, 2.00, 1.11]	237.81
OVRV	[0.34, 0.39, 9.92, 0.99]	5.13	[0.60, 0.00, 8.40, 1.36]	4.06	[0.69, 0.00, 12.54, 1.69]	4.18
OVM	[21.88, 0.82, 9.43]	53.69	[21.69, 1.49, 10.63]	63.02	[21.78, 0.54, 12.73]	68.47
CACC	[0.10, 0.39, 4.96, 1.47]	3.06	[0.31, 0.28, 8.89, 1.44]	2.67	[0.19, 0.28, 7.43, 2.00]	2.88
RF Regressor	—	0.0046	—	0.0016	—	0.0025

A. Calibration Objective

We calculated the Root Mean Squared Error (RMSE) between the simulated and experimental spacing as the calibration objective function using the following equation:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (s_{\text{sim}}(i) - s_{\text{exp}}(i))^2} \quad (7)$$

where s_{sim} and s_{exp} represent the simulated and experimentally observed spacing at the i -th time step, respectively, and N is the total number of observations.

B. Optimization Strategy

Each model was calibrated using the L-BFGS-B algorithm provided by `scipy.optimize.minimize`, which supports parameter box constraints. The dataset was sliced into six subsets to train the model, and calibration was performed separately on these slices (200s subsets) for each gap, and the best parameter set was selected based on overall RMSE performance. The subset with the lowest RMSE was selected for the final simulation. This ensured that the models were tested across a range of traffic conditions.

V. MACHINE LEARNING MODELS

In addition to classical models, we trained data-driven ML models to predict vehicle behavior. Two separate models were developed: one for acceleration prediction and another for spacing prediction. Both were implemented using Random Forest Regressors. We selected the Random Forest regressor as our sole ML model. This choice stems from our aim to benchmark ML against classical methods, rather than exploring inter-ML comparisons. RF offers a robust, fast, and interpretable model that has proven superior to both traditional and neural methods in prior car-following studies [21–23].

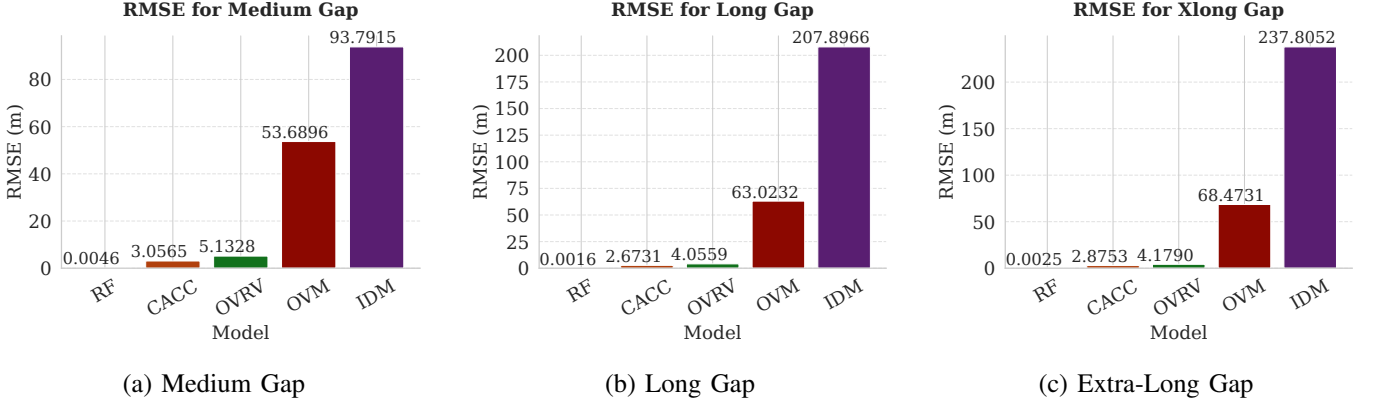


Fig. 1: RMSE comparison of RF, CACC, OVRV, OVM, and IDM models across three gap settings: (a) Medium Gap — RF, CACC and OVRV show low errors, while OVM and IDM have higher RMSE; (b) Long Gap — similar trend, with IDM showing substantial error; (c) Extra-Long Gap — IDM reaches the highest RMSE.

A. Acceleration Prediction Model

The goal of the first model is to predict the acceleration of the follower vehicle directly from input features such as spacing and speed. We consider follower speed, gap settings, speed fluctuation, and the speed difference Δv between the follower and the lead vehicle. We computed the target acceleration $a(t)$ using equation 8,

$$a(t) = \frac{v(t) - v(t - \Delta t)}{\Delta t} \quad (8)$$

It achieved high precision with RMSE around 0.68 m/s² and R^2 close to 0.996 on the entire dataset.

B. Space Prediction Model

The second ML model predicts the spacing between vehicles. The model was trained using the same dataset and evaluated against actual spacing measurements. The results showed a strong alignment across various time windows.

VI. EVALUATION AND COMPARISON

In this section, we evaluate the performance of both classical and machine learning models by comparing their ability to reproduce real-world EV car-following behavior. The primary metric used to assess accuracy is the Root Mean Squared Error (RMSE) as shown in Equation (7). For machine learning models, we also report the coefficient of determination (R^2), which measures how well the predictions explain the variance in the data.

A. Residual Analysis of Machine Learning Models

Beyond numerical evaluation metrics, we also analyze the residuals (the difference between actual and predicted values) for both the acceleration and spacing prediction models. A residual plot is a common diagnostic tool used to assess model performance, revealing potential biases and non-random prediction patterns [24, 25].

Figure 2(a) shows the residuals of the acceleration model plotted against predicted acceleration. The points are tightly clustered around the zero-error line, indicating that the model captures the underlying trend well and does not exhibit systematic errors across different predicted values.

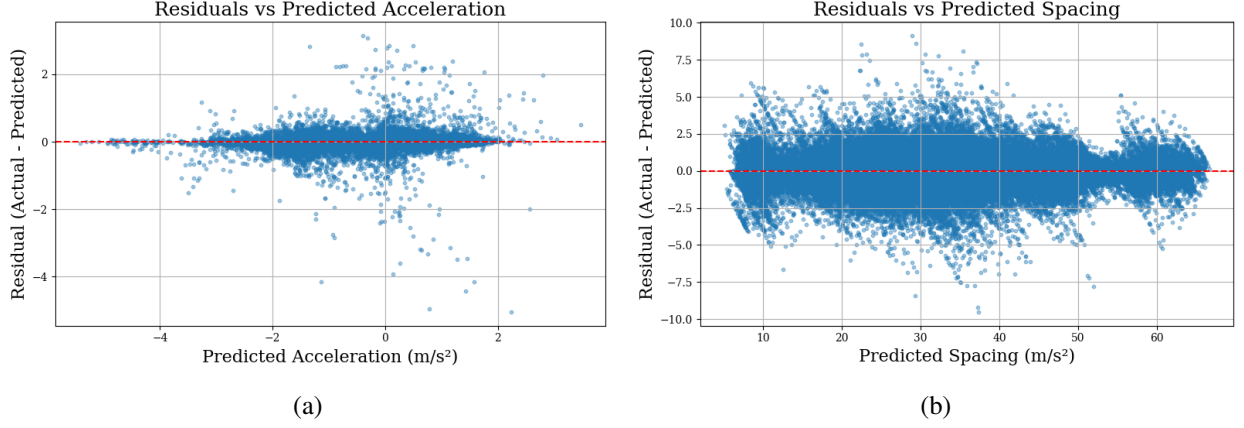


Fig. 2: Most of the residual points are closely centered around $y = 0$ line for both (a) and (b), which suggests that the models are not biased and do not overpredict or underpredict consistently. Both (a) and (b) show slight deviation for very high and low data points, but effectively capture the complex dynamics.

Similarly, Figure 2(b) presents the residuals for the spacing prediction model. Although there is a slightly wider spread compared to the acceleration model, the residuals remain largely centered around zero, suggesting that the model generalizes well over a wide range of spacing conditions.

B. Models Performance Comparison

Table I illustrates the best-calibrated parameters and RMSE for three gap settings. The parameters for each classical model are as follows: IDM - (v_0, τ, s_0, a, b) , OVRV - (k_1, k_2, τ, η) , OVM - $(v_{\max}, \alpha, s_{st})$, CACC - (k_1, k_2, s_0, T) . The capacity of classical physics-based models and ML models to replicate actual spacing and accelerating behavior is used to evaluate their performance.

The CACC consistently has better performance in all three gap settings than OVRV, OVM, and IDM. This is because the improved version has leader acceleration and spacing error feedback, which allows it to adjust the dynamic changes. OVRV also demonstrates commendable performance, especially in medium and long-gap settings. The performance of OVM and IDM was not as accurate as others. IDM overestimates spacing in larger gaps, and OVM much more deviates from the experimental one for spacing prediction.

The Random Forest regressor is trained to predict the acceleration and the RMSE of this model is as low as **0.0016** m for the long gap setting, which is much lower than all classical models in every gap category. This illustrates how, without depending on predetermined physical assumptions, the model can directly extract complex nonlinear connections from the data. The ML model's effectiveness is further supported by residual analysis, which indicates that errors are symmetrically distributed, centered around zero, and not the subject of significant bias or variance drift.

These findings imply that machine learning models, when given enough data, offer better accuracy and generalization than traditional models, which provide interpretation, a theoretical foundation, and more control over the assumptions.

C. Visual Comparison

The Random Forest (RF) regressor demonstrates superior performance in mimicking the leader's speed, particularly under the long gap setting, where it closely follows the actual trajectory. In contrast, for medium and extra long (xlong) gap settings, the RF model tends to be more conservative, slightly underestimating acceleration changes but still maintaining smooth transitions.

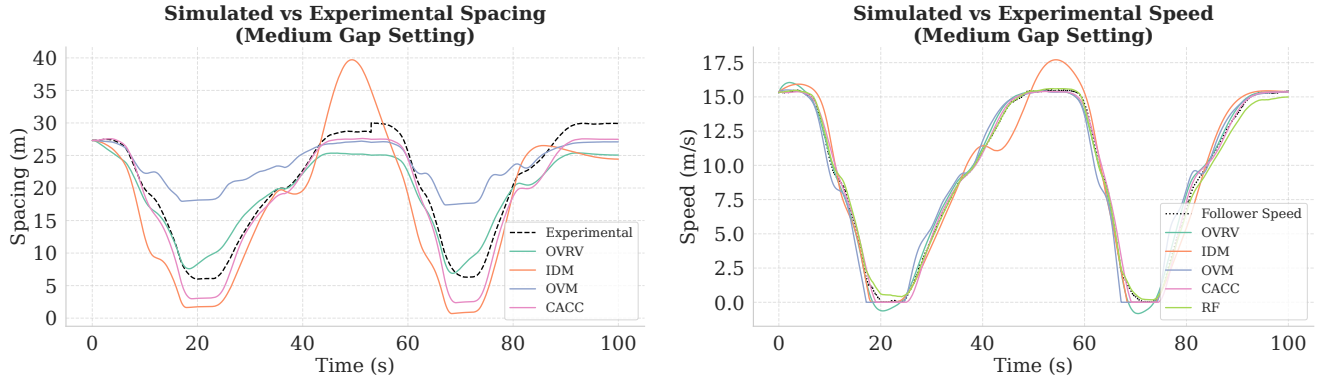


Fig. 3: Simulated vs Experimental Spacing (left) and Speed (right) for Medium Gap

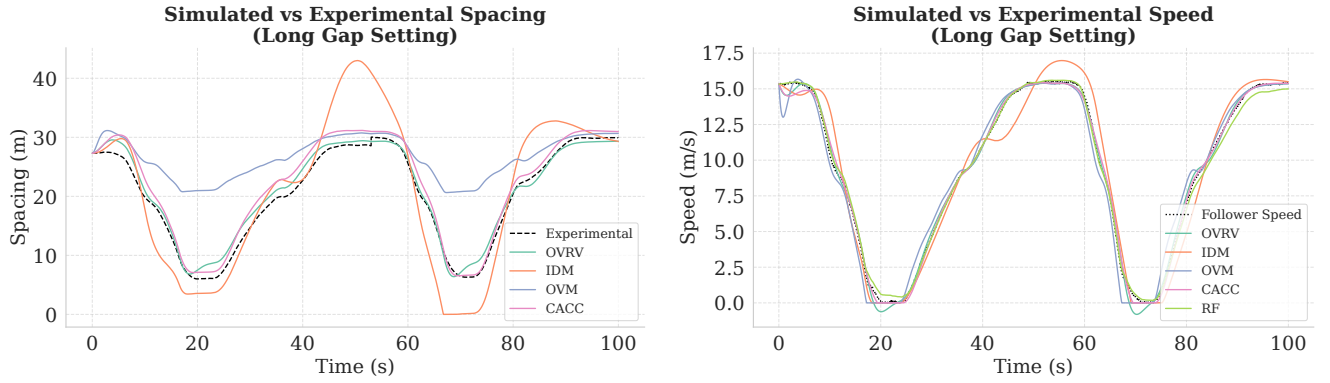


Fig. 4: Simulated vs Experimental Spacing (left) and Speed (right) for Long Gap

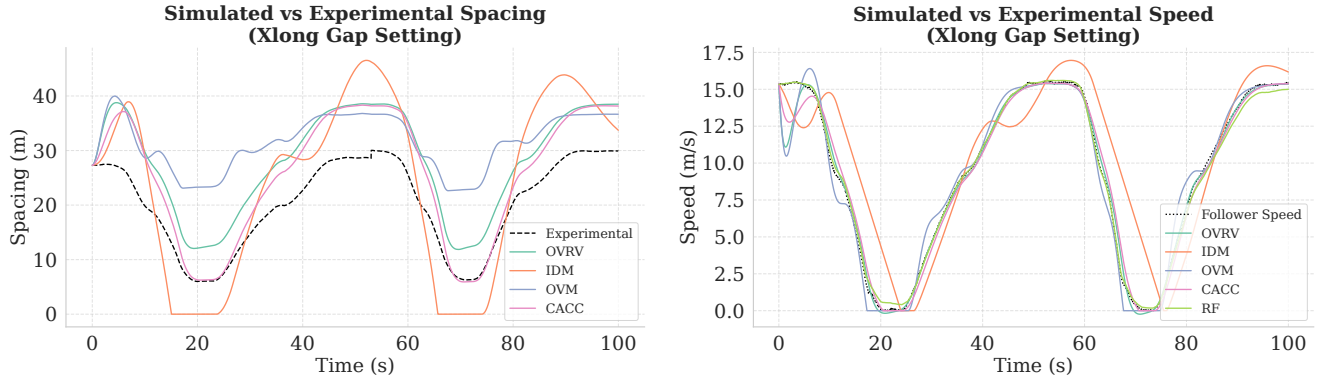


Fig. 5: Simulated vs Experimental Spacing (left) and Speed (right) for XLong Gap

A remarkable pattern is observed during the initial 10 seconds across all gap settings – classical physics-based models (IDM, OVM, OVRV, and CACC) exhibit greater fluctuations and lag in adapting to the leader's speed. On the other hand, the ML-based model shows immediate responsiveness and stability, indicating better generalization and adaptability from the start of the sequence. These results highlight the ML model's advantage in consistency and early accuracy, especially under complex or dynamically changing driving conditions.

Among physics-based models, CACC and OVRV align most closely with the leader's speed. The CACC model notably maintains smooth acceleration profiles despite fluctuations in the leader's velocity—a critical trait for reliable car-following behavior. At 38 and 85 seconds, where other models overreact or destabilize,

CACC achieves smoother transitions without abrupt adjustments.

This stability is further evident in spacing plots (Figure 3, Figure 4 and Figure 5), where CACC consistently preserves uniform inter-vehicle gaps. Its performance improves with larger gap settings which is suitable for real-world cooperative adaptive cruise control systems.

VII. CONCLUSION AND FUTURE WORK

This paper presented a comparative study of classical and machine learning approaches to model the behavior of EV car-following scenarios. We used a real-world dataset in which an electric vehicle followed an ICE vehicle under various driving conditions.

We implemented and calibrated four classical models: the Intelligent Driver Model (IDM), Optimal Velocity Model (OVM), Optimal Velocity Relative Velocity (OVRV), and a simplified Cooperative Adaptive Cruise Control (CACC) model. These models were evaluated using RMSE against observed spacing and speed data. We also developed separate random forest regressors, one to predict acceleration and the other one to predict spacing. These models achieved high accuracy, outperforming classical models in accuracy metrics such as RMSE.

While classical models offer simplicity, interpretability, and robustness, machine learning models provide better accuracy and adaptability to varying driving patterns. However, ML models may suffer from a lack of explainability, which can be critical in safety-sensitive applications.

Future work will focus on assessing the practical applicability of these models in real-world scenarios, we propose evaluating the computational efficiency and inference time of each model. This will provide insights into their feasibility for real-time deployment in electric vehicles for automation. Another important extension is to broaden the analysis to multi-vehicle scenarios or vehicle platoons, which will allow for a deeper understanding of the models' scalability, interaction dynamics, and collective stability in more complex traffic environments.

Code and Data Availability. Code and Data is available on https://github.com/AARC-lab/EV_Carfollowing_Calibration_ITSC25.

REFERENCES

- [1] B. N. Matcha, S. N. Namasivayam, M. Hosseini Fouladi, K. Ng, S. Sivanesan, and S. Y. Eh Noum, "Simulation Strategies for Mixed Traffic Conditions: A Review of Car-Following Models and Simulation Frameworks," *Journal of engineering*, vol. 2020, no. 1, p. 8231930, 2020.
- [2] Y. Zhou, R. Fu, C. Wang, and R. Zhang, "Modeling Car-Following Behaviors and Driving Styles with Generative Adversarial Imitation Learning," *Sensors*, vol. 20, no. 18, p. 5034, 2020.
- [3] M. Treiber and A. Kesting, *Traffic Flow Dynamics: Data, Models and Simulation*. Springer-Verlag Berlin Heidelberg, 2013.
- [4] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, "Dynamical Model of Traffic Congestion and Numerical Simulation," *Physical Review E*, vol. 51, no. 2, pp. 1035–1042, 1995.
- [5] M. Sharath and N. R. Velaga, "Enhanced Intelligent Driver Model for Two-Dimensional Motion Planning in Mixed Traffic," *Transportation Research Part C: Emerging Technologies*, vol. 120, p. 102780, 2020.
- [6] Y. Jin and J. Meng, "Dynamical Analysis of an Optimal Velocity Model with Time-Delayed Feedback Control," *Communications in Nonlinear Science and Numerical Simulation*, vol. 90, p. 105333, 2020.
- [7] D. Holley, J. D'sa, H. N. Mahjoub, G. Ali, B. Chalaki, and E. Moradi-Pari, "MR-IDM-Merge Reactive Intelligent Driver Model: Towards Enhancing Laterally Aware Car-Following Models," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 1460–1467.
- [8] X. Chen, M. Zhu, K. Chen, P. Wang, H. Lu, H. Zhong, X. Han, X. Wang, and Y. Wang, "Follownet: A Comprehensive Benchmark for Car-Following Behavior Modeling," *Scientific data*, vol. 10, no. 1, p. 828, 2023.
- [9] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, "Dynamical Model of Traffic Congestion and Numerical Simulation," *Physical Review E*, vol. 51, no. 2, pp. 1035–1042, 1995.
- [10] R. Jiang, Q. S. Wu, and Z. J. Zhu, "Full Velocity Difference Model for a Car-Following Theory," *Physical Review E*, vol. 64, no. 1, p. 017101, 2001.
- [11] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long Short-Term Memory Neural Network for Traffic Speed Prediction Using Remote Microwave Sensor Data," *Transportation Research Part C*, vol. 54, pp. 187–197, 2015.

- [12] X. Liang, X. Du, G. Wang, and Z. Han, "Deep Reinforcement Learning for Autonomous Driving: A Survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 940–960, 2018.
- [13] X. Chen, S. Xia, Y. Liu, Z. He, and X. Jin, "Residual Learning-Based Hybrid Car-Following Model Using Real-World Driving Data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4275–4287, 2021.
- [14] R. Rajamani, *Vehicle Dynamics and Control*. Springer Science & Business Media, 2011.
- [15] V. Milanes and S. E. Shladover, "Modeling Cooperative and Autonomous Adaptive Cruise Control Dynamic Responses Using Experimental Data," *Transportation Research Record*, vol. 2489, no. 1, pp. 124–132, 2014.
- [16] Y. Zhou, Z. He, Z. Pu, J. Sun, and X. Wang, "FollowNet: Benchmarking Car-Following Models in Naturalistic Driving Scenarios," *Scientific Data*, vol. 10, no. 1, pp. 1–13, 2023.
- [17] X. Han, X. Chen, M. Zhu, P. Cai, J. Zhou, and X. Chu, "Ensemblefollower: A Hybrid Car-Following Framework Based on Reinforcement Learning and Hierarchical Planning," *arXiv preprint arXiv:2308.16008*, 2023.
- [18] J. Jiang, Y. Zhou, X. Wang, and S. Ahn, "A Generic Stochastic Hybrid Car-Following Model Based on Approximate Bayesian Computation," *Transportation Research Part C: Emerging Technologies*, vol. 167, p. 104799, 2024.
- [19] Y. Zheng, R. Yan, B. Jia, R. Jiang, A. Tapus, X. Chen, S. Zheng, and Y. Shang, "Adaptive Kalman-Based Hybrid Car Following Strategy Using TD3 and CACC," *arXiv preprint arXiv:2312.15993*, 2023.
- [20] P. Chon Kan-Munoz, S. Lapardhaja, and X. D. Kan, "Field experiments of commercially available automated vehicles on freeways," 2021.
- [21] H. Shi, T. Wang, F. Zhong, H. Wang, J. Han, and X. Wang, "A data-driven car-following model based on the random forest," *World Journal of Engineering and Technology*, vol. 9, no. 3, pp. 503–515, 2021.
- [22] D. Choi and S. Lee, "Comparison of machine learning algorithms for predicting lane changing intent," *International journal of automotive technology*, vol. 22, no. 2, pp. 507–518, 2021.
- [23] D. Rowan, H. He, F. Hui, A. Yasir, and Q. Mohammed, "A systematic review of machine learning-based microscopic traffic flow models and simulations," *Communications in Transportation Research*, vol. 5, p. 100164, 2025.
- [24] S. Chatterjee and A. S. Hadi, *Regression Analysis by Example*. John Wiley & Sons, 2015.
- [25] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. John Wiley & Sons, 2021.