A PHYSICS ENHANCED RESIDUAL LEARNING (PERL) FRAMEWORK FOR AUTONOMOUS VEHICLE

by

Keke Long

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Civil and Environmental Engineering)

at the

UNIVERSITY OF WISCONSIN-MADISON

2024

Date of final oral examination: 11/20/2024

The dissertation is approved by the following members of the Final Examination Committee:

Xiaopeng (Shaw) Li, Professor, Civil and Environmental Engineering, UW-Madison
Sikai Chen, Assistant Professor, Civil and Environmental Engineering, UW-Madison
Soyoung (Sue) Ahn, Professor, Civil and Environmental Engineering, UW-Madison
Jinlong Wu, Assistant Professor, Mechanical Engineering, UW-Madison
Hongkai Yu, Assistant Professor, Electrical and Computer Engineering, Cleveland State University

DEDICATION

I dedicate this dissertation to my beloved grandparents and parents.

ACKNOWLEDGMENTS

I am deeply grateful to my advisor, Prof. Xiaopeng Li, whose rigorous academic standards and passionate commitment to research have profoundly influenced me. Prof. Li's insightful guidance, coupled with the steadfast support of the CATS Lab students, has cultivated an inspiring and collaborative environment for growth. His mentorship has transformed my Ph.D. experience into one of profound discovery and personal fulfillment.

I would also like to express my thanks to my co-advisor, Prof. Sikai Chen, whose vision and creativity have been invaluable. His foresight and encouragement to embrace cutting-edge technologies have greatly enriched my academic experience.

My sincere appreciation also goes to my committee members. I am grateful to Prof. Soyoung Ahn for her invaluable help with my paper and dissertation, enhancing its quality and depth. I am also thankful to Prof. Jinlong Wu for his generous assistance and interdisciplinary expertise, which has been crucial in aligning my research interests with my academic goals. Additionally, I would like to express my gratitude to Prof. Hongkai Yu for his valuable insights from the field of Computer Science, which have significantly strengthened my research.

I am thankful for the collaborative academic atmosphere provided by the University of Wisconsin and the University of South Florida. Thank everyone in the CATS Lab. I cherish the time we spent together in the past years.

ABSTRACT

Autonomous vehicles (AV) driving in mixed traffic, comprising human-driven vehicles (HVs) and AVs, is notoriously challenging, underscored by complex driving environments and a multitude of uncertainties. Further, the AV system involves complex interactions cascading across the interconnected modules for perception, planning, and control. An error in the higher chain of these interconnected modules can propagate downstream, inducing instability in vehicle control. Such errors can also cascade through following vehicles and impact traffic dynamics at a broader level. To overcome these challenges, we propose the novel Physics-Enhanced Residual Learning (PERL) framework for AV operations in mixed traffic. PERL comprises two components: a physics-based model and residual learning. The physics model provides primary outputs, including prediction outputs or control outputs, and then its residuals are learned by a learning-based approach as corrections to the physics model to enhance the results.

This dissertation comprises three major research thrusts: (1) Development of PERL frameworks for vehicle trajectory prediction (2) Validate the contribution of PERL-based prediction in AV control, and (3) Development of Physics-Enhanced Residual Policy Learning (PERPL) framework for vehicle control.

The first part proposed the PERL framework and applied it to a vehicle trajectory prediction problem with real-world trajectory data of both HV and AV, using an adapted Newell car-following model as the physics model, and four kinds of neural networks (GRU, Convolution Long Short-Term Memory (CLSTM), VAE and Informer model) as the residual learning model. We compare this PERL model with pure physics models, NN models, and other physics-informed neural network (PINN) models. The result reveals that the PERL model yields the best prediction with limited training data and it has fast convergence during training. Moreover, the PERL model requires fewer parameters to achieve similar predictive performance compared to NN and PINN models.

The second part proposes a PERL-based vehicle control method to mitigate traffic oscillation in the mixed traffic environment of connected and autonomous vehicles (CAVs) and HVs. This model includes the PERL prediction model and a controller. The PERL-based prediction model precisely predicts the behavior of the preceding vehicle, especially downstream speed fluctuations, to allow sufficient time

for the driver to respond to these speed fluctuations. For the controller, we employ a Model Predictive Control (MPC) model that considers the dynamics of the CAV and its following vehicles, improving safety and comfort for the platoon formed including the CAV and following vehicles. The proposed model is validated through a Vehicle-in-the-loop (ViL) field test. Results validate the proposed method in damping traffic oscillation and enhancing the safety and fuel efficiency of the CAV and the following vehicles in mixed traffic with the presence of uncertain vehicle dynamics and actuator lag.

The third part proposes the Physics-Enhanced Residual Policy Learning (PERPL) framework for vehicle control, leveraging the advantages of both physics-based models (data-efficient and interpretable) and RL methods (flexible to multiple objectives and fast computing). The physics component provides model interpretability and stability and the learning-based Residual Policy adjusts the physics-based policy to adapt to the changing environment, thereby refining the decisions of the physics model. This model is applied in decentralized control of a mixed traffic platoon of CAVs and HVs using a constant time gap (CTG) strategy, with actuator lag and communication delays. Experimental results demonstrate that this model has high extrapolation ability, achieving smaller headway errors and better oscillation dampening than the linear control model and reinforcement learning (RL) model in artificial extreme scenarios. At the macroscopic level, overall traffic oscillations are also reduced as the penetration rate of CAVs employing the PERPL-based controller increases.

KEYWORDS

Automated Vehicles,

Connected Vehicles,

Vehicle Trajectory Prediction,

Vehicle Trajectory Planning,

Vehicle Control,

Physics-informed Neural Network,

Deep Reinforcement Learning.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Background

Autonomous driving systems are technologies that enable vehicles to operate independently, performing driving tasks with varying levels of human intervention, from partial assistance to full automation. Autonomous driving systems are typically realized through a sequential integration paradigm, where modules (e.g., prediction, planning, control, etc.) are connected in a chain-like manner, with the output of one module serving as the input for the next. The complex dynamics of autonomous driving systems and the uncertainties therein pose significant challenges in achieving robust and safe autonomous vehicle (AV) operations (Le Mero et al., 2022). Uncertainties arise from stochastic human behaviors (Pan et al., 2021) and control delays attributed to processing time, signal transmission quality, and inherent latency in control signal execution (Chen et al., 2018). These factors contribute to residuals – the discrepancies between the predicted and actual vehicle states, which can accumulate through the system, impacting an AV's ability to execute precise and timely control actions.

The risk of incorrect predictions is exemplified in the world's first fatal accident involving an autonomous vehicle. In this incident, an Uber test vehicle detected a pedestrian crossing the street but failed to accurately predict her path, resulting in a fatal collision (National Transportation Safety Board, n.d.). This tragic case highlights the critical importance of accurate and timely predictions in AV systems. Control errors are also critical in AV safety. In an incident involving a Tesla vehicle operating in Autopilot mode, the vehicle drifted out of the lane, ultimately leading to the driver's death (KATU Staff, 2023). According to the National Highway Traffic Safety Administration (NHTSA) (National Highway Traffic Safety Administration, 2024), 145 out of 467 Tesla Autopilot crashes were caused by roadway departures in low-traction conditions. Moreover, control response delays, such as actuator lag, further complicate safety.

AV control has traditionally relied on physics model-based prediction models and classic control approaches (Güvenç et al., 2012; Milanés and Shladover, 2014). Despite their interpretability and solid theoretical foundation, these methods often oversimplify complex, non-linear dynamics of real-world

driving, particularly in mixed-traffic environments. This can lead to large prediction errors that can amplify through the downstream modules (e.g., lower-level control) (McAllister et al., 2022). Consequently, the final system output may significantly deviate from the ideal scenario of perfect control with perfect prediction. A car-following example in Figure 1-1 (a) demonstrates this phenomenon, where physics models, Intelligent Driver Model (IDM) prediction (Treiber et al., 2000), and Proportional-Integral-Derivative (PID) control (Ang et al., 2005), lead to significant position errors compared to the perfect prediction and control. Data-driven machine learning methods, such as deep learning models, have emerged as powerful alternatives to improve prediction accuracy and control performance, by leveraging their ability to learn from data (Di and Shi, 2021). However, their lack of interpretability and reliance on vast amounts of high-quality training data (Karniadakis et al., 2021) pose challenges, especially in safety-critical applications and rare driving scenarios with limited data.

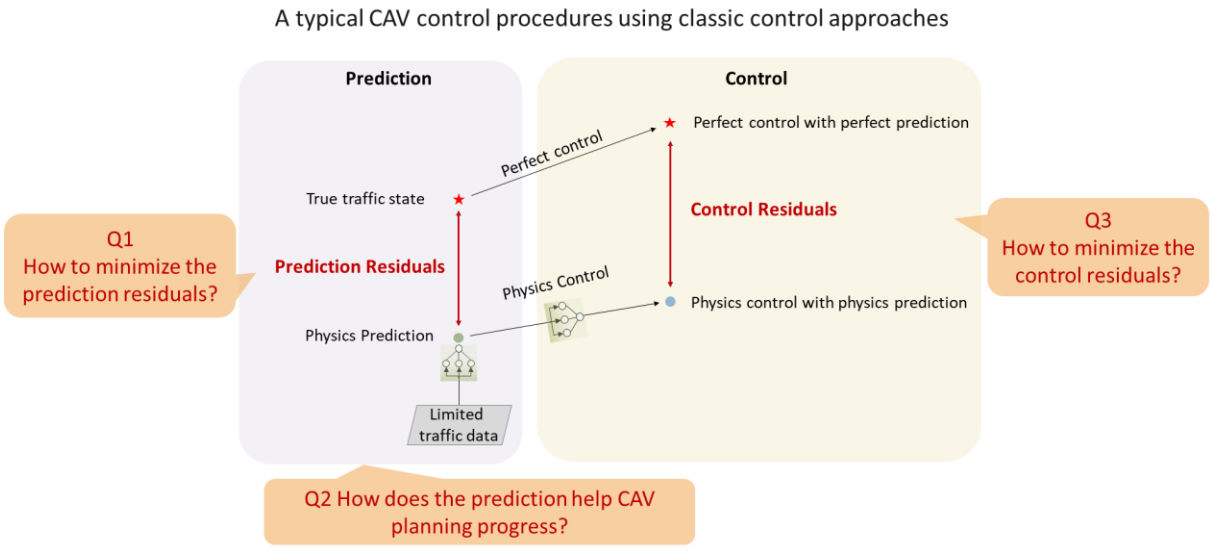


Figure 1-1 Error and trajectory comparison: physics/PIML/PERL-based prediction and control.

Recognizing the shortcomings of both approaches, physics-informed machine learning (PIML) attempts to integrate these approaches by leveraging physics models to guide the learning process of data-driven methods (Karpatne et al., 2017; Raissi et al., 2019); for example, using physics models to generate

additional training data (Han et al., 2022; Naing et al., 2022) or as constraints in the loss function (Mo et al., 2021; Shi et al., 2022). This PIML approach, however, may introduce biases from the physics models and thus limit its adaptability and flexibility (Long et al., 2024b). A PIML method tends to shift the neural network (NN) predictions towards the often less accurate physics model when data is limited Figure 1-1 (b). Moreover, current PIML literature primarily focuses on prediction, leaving the control aspect largely unaddressed, as indicated by the question mark in the control module of Figure 1-1 (b).

To address these shortcomings, we propose a new paradigm that uniquely integrates physics models and data-driven approaches. Specifically, the primary objective of the proposed research is to develop a Physics-Enhanced Residual Learning (PERL) framework for AV systems operating in mixed traffic (see Figure 1-1 (c)). PERL is fundamentally different than PIML in principle. Rather than guiding the learning process, the physics model remains a fundamental component for output in PERL. The physics model infuses PERL with prior knowledge, and the residual learning model focuses on predicting the residual of the physics model as a correction to it. Figure 1-1 (b) and (c) demonstrate the superiority of PERL in prediction performance against PIML and control performance against PID. PERL adopts a more intuitive and rational approach that minimizes the gap between the limited input data and the ideal state of perfect prediction and control, and offers distinctive advantages: (1) lower risk of bias caused by the physics model; (2) efficient learning thanks to reduced dimensionality with its focus on residuals; (3) lower data requirement; and (4) better interpretability and transparency – important qualities for the autonomous driving system.

1.2 Research Objectives and Scope of Work

Considering the residuals present in current vehicle trajectory prediction and autonomous vehicle control, as well as the gaps identified in existing research, this study aims to enhance the effectiveness of prediction and control by proposing a framework that integrates physical models with neural networks:

(i) Propose a framework that combines physical models and neural networks and apply it to the trajectory prediction of both human-driven and autonomous vehicles.

(ii) Integrate the PERL framework with model predictive-based control to plan trajectories for autonomous vehicles in mixed traffic flows, enhancing overall and fuel efficiency.

(iii) Utilize vehicle in the loop to validate the proposed solutions in real-world scenarios.

(iv) Apply the PERL framework to vehicle control to improve model safety and generalizability.

1.3 Contributions

Through this study, the following contributions are expected:

(i) A physics-enhanced residual learning structure is proposed for the prediction problem.

(ii) The PERL framework is applied in vehicle trajectory prediction, and its predictability and interpretability are validated using real-world vehicle trajectory data.

(iii) The PERL-based trajectory prediction model is integrated into a model predictive control for mixed platoons and validated through field tests.

(iv) A Physics-Enhanced Residual Policy Learning (PERPL) framework is proposed and applied to decentralized control of mixed traffic platoons while incorporating actuator and communication delays.

1.4 Dissertation Overview and Contributions

To answer the above questions, this dissertation is organized as shown in Figure 1-2: Chapter 2 reviews the relevant studies on autonomous vehicles (AVs). Chapter 3 presents a PERL-based method for vehicle trajectory prediction. Chapter 4 applies this trajectory prediction method to predictive vehicle control and validates its performance through field testing. Chapter 5 proposes a PERPL-based vehicle control strategy for mixed traffic scenarios. Chapter 6 offers concluding remarks and suggests directions for future research.

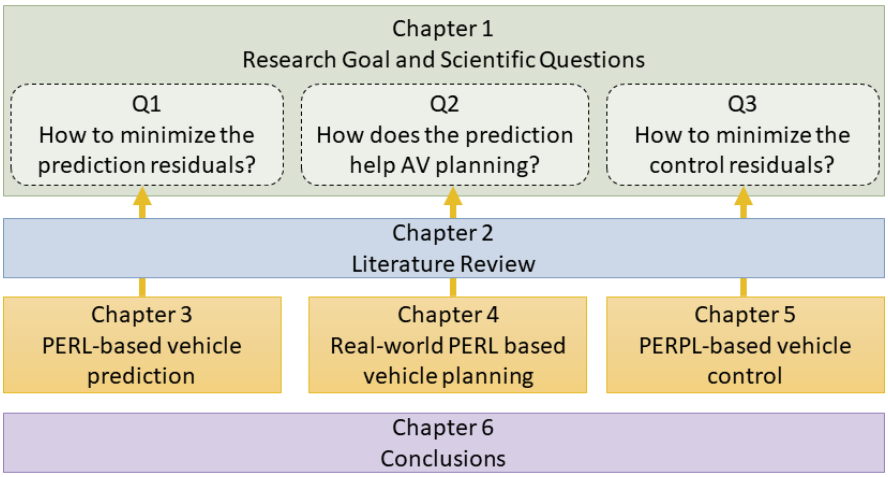


Figure 1-2 Dissertation structure

2 LITERATURE REVIEW

2.1 Prediction model

The parsimony of physics models was sometimes a compromise when only limited data and computational resources were available (Figure 2-1 A). The advent of data science leads to a prolific accumulation of data across various fields. Concurrently, neural network (NN) models, along with numerous data-driven machine learning methods, have emerged to enhance predictability, as it is shown that NN models are capable of approximating any continuous function (Figure 2-1 B) (Hornik et al., 1989). NN models have been extensively utilized in domains such as medical forecasting (Gulshan et al., 2016), weather forecasting (Salman et al., 2015), and road traffic prediction (Bates et al., 2022; Zhou et al., 2017), the third being the focused domain of this study.

Despite the superior predictability of NN models, they suffer from several limitations in real-world applications, including a lack of interpretability, lack of robustness, and dependence on vast data (Karniadakis et al., 2021; Li et al., 2022). To overcome these shortcomings, the physics-informed neural network (PINN) model (Figure 2-1 C) has emerged (Karniadakis et al., 2021, 2021; Zhang et al., 2024). To enhance the interpretability and ensure the stability of models beyond the domains of their training data, the PINN model encodes physics prior to NN. The widely adopted PINN for road traffic prediction, such as vehicle trajectory prediction, integrates physics priors into the NN's loss function (Long et al., 2024e; Mo et al., 2021; Naing et al., 2022; Yao et al., 2023). This approach strikes a balance between the physics model and NN by modulating the weight of the deviation from the physics model prediction and that of NN's loss function. Despite salient advantages, the deviation correction towards the often less accurate physics model prediction may somehow compromise PINN's predictability. Another concern arises from unstable training due to different scales and convergence rates between the NN loss and the physics model deviation. Thus, tuning the weights of the physics model within the loss function is proven to be challenging (Krishnapriyan et al., 2021; Wang et al., 2022, 2021). Furthermore, without sufficient data, PINNs may

underperform since the efficacy of PINNs heavily leans toward the relatively inferior predictability of the physics model.

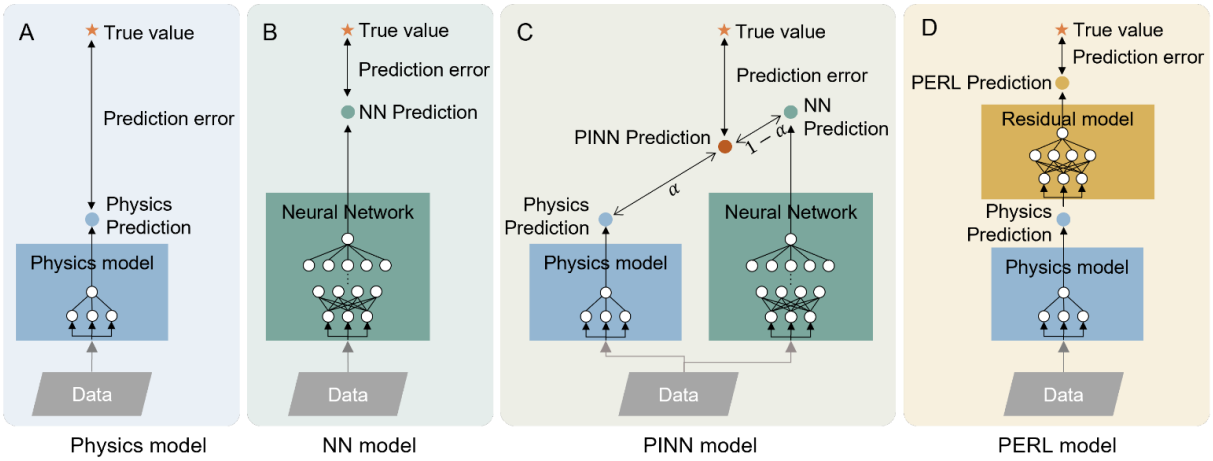


Figure 2-1 General structure of PERL model and three existing models: Physics model, NN model, PINN model.

To harvest the advantages of all these models while circumventing their limitations, we propose a novel Physics-Enhanced Residual Learning (PERL) framework (Figure 2-1 D). Instead of a compromise between the two models, PERL further improves the prediction of the physics model with a residual term learned from the NN model. The physics model component alone can yield a reasonably accurate (albeit not the best) prediction with only a parsimonious structure specified by a few parameters. Then, as opposed to a pure NN model starting from scratch, the residual learning model component only needs to improve an already reasonably accurate prediction to a higher accuracy comparable to the pure NN model, which likely takes much less training data and computational resources and thus only needs fewer parameters and less training data. Meanwhile, unlike PINN, which may draw the prediction toward relatively less accurate physics model prediction, PERL is pushing the prediction away from the physics model prediction to the most accurate value that a data-driven model can ever reach. With this, PERL better integrates the merits of both physics and NN models to achieve the best and uncompromised predictability and the fastest

convergence rate with the fewest parameters and inherited interpretability. While the concept of a residual term has been explored in various fields, such as physics systems modeling (Brunton et al., 2016; Kaheman et al., 2019), robotic manipulation tasks (Möllerstedt et al., 2022; Silver et al., 2018), and electromagnetic modeling (Shan et al., 2023), to the best of the authors' knowledge, residual learning has rarely been explored for system state prediction, which motivated us to develop the PERL model.

2.2 Planning strategy

Traffic oscillations, commonly referred to as 'stop-and-go' traffic, epitomize the fluctuation between slow-moving and fast-moving states in congested traffic, deviating from a steady flow (Li et al., 2014, 2010). This widespread phenomenon in human driving scenarios precipitates several critical issues. Firstly, it exacerbates the risk of accidents by complicating the maintenance of safe vehicle distances, thus heightening collision probabilities (Li et al., 2012; Yao et al., 2020). Secondly, it diminishes traffic efficiency, inducing congestion and a ripple effect that disrupts numerous vehicles. Moreover, the frequent acceleration and deceleration cycles escalate fuel consumption and emissions (Stern et al., 2019), detrimentally impacting the environment.

The evolution of connected and automated vehicles (CAVs) technologies offers a promising solution to mitigate traffic oscillations (Ghiasi et al., 2019; He et al., 2024; Larsson et al., 2021; Wang et al., 2023). CAVs leverage advanced perception systems and trajectory planning. At the perception part, CAVs can perceive information about preceding vehicles, anticipating the traffic oscillations ahead (Zhou et al., 2017). In a connected scenario, lead vehicles harness Vehicle-to-Everything (V2X) technology to collect and analyze traffic data. Conversely, in non-connected settings, roadside units and monitoring systems could gather traffic information for real-time predictions and advisories and send the information to the target vehicle through dynamic message signs. This gathered data enables the identification of traffic oscillation patterns, facilitating accurate predictions and allowing the vehicle to determine when it might be impacted (Yao et al., 2023). Downstream oscillation patterns not only facilitate accurate predictions but also inform the strategic planning and control phases of CAV operation (Fang et al., 2024). Researchers have captured the formation and ensuing propagation of stop-and-go waves and predicted traffic oscillation

using the behavioral car-following model (Chen et al., 2012) and neural network-based models (Zhou et al., 2017). By understanding the likely traffic conditions in advance, CAVs can optimize their trajectory planning to either avoid or mitigate potential impacts from identified oscillations. This proactive integration of prediction and planning enables a more coherent approach to traffic management, ensuring that CAVs dynamically adjust their behavior to maintain optimal flow and enhance overall traffic safety.

For the perception part, most existing CAV controllers mitigate traffic oscillation based on the predicted behavior of the preceding vehicle (Hu et al., 2021; Wang et al., 2023) or aggregated information, e.g., average vehicle speed (Stern et al., 2019). Relying solely on the predicted behavior of the preceding vehicle is insufficient for predictive safety measures. It may raise two main issues. First, traffic oscillation originates downstream (Zheng et al., 2022). The wave's fluctuations upstream are hard to perceive if only one preceding vehicle is considered. Second, when the fluctuations of the preceding vehicle are detected, it's usually too late for the following CAV to respond appropriately, with the consideration of communication delay and actuator lag. Therefore, to effectively predict the state of preceding vehicles, information from multiple vehicles in front is required as input.

In the realm of predicting downstream multi-vehicle trajectories, most studies have adopted physics-based models to model and predict vehicle trajectories. In this research, "physics model" or "physics rules" refers to theoretical or empirical formulations that describe the behavior of physical systems. These models are often based on fundamental principles such as conservation laws, equations of motion, or thermodynamics, structured to explain and predict the dynamics of systems under various conditions. In the context of shockwave modeling. The physics of shockwaves is a fundamental traffic flow characteristic that was first studied by the Lighthill-WhithamRichards (LWR) model (Lighthill and Whitham, 1997). Shockwaves in congested traffic usually follow “stop-and-go” patterns that could cause adverse consequences. These models typically utilize the historical data of the subject vehicle and its immediate predecessor to predict future trajectories for a limited number of time steps, which may not suffice for the CAV controller. Moreover, physics models may struggle to capture the complex interactions and nonlinear

behaviors prevalent in dense traffic conditions (Durrani et al., 2016; Punzo and Montanino, 2020). Recent researchers turned to data-driven methods due to their proficiency in detecting intricate patterns and adapting to diverse datasets (Yao et al., 2022). However, these learning-based methods often require substantial training data (Karniadakis et al., 2021; Li et al., 2022) and lack interpretability. In response to these challenges, the Physics-Informed Neural Networks (PINN) method has been utilized to enhance the applicability of NN models in scenarios with limited data samples (Mo et al., 2021; She and Ouyang, 2024). This is achieved by incorporating physical models to augment data and regularize the models to prevent overfitting, although this method is contingent on the accuracy of the physical models and can lead to unstable training processes (Yao et al., 2023).

Recognizing the complementary strengths and weaknesses of these methodologies, the Physics-Enhanced Residual Learning (PERL) framework was proposed to combine the robust interpretability of physics models with the adaptive precision of learning-based models (Long et al., 2024b). By focusing on the physics model residuals — the differences between the predictions of the physics model and observed data — PERL leverages a neural network to refine these predictions. This method enhances the accuracy of trajectory forecasts, maintains the model's explainability, and reduces its dependency on extensive datasets. Consequently, PERL stands out for its ability to deliver high-precision and stable predictions of future vehicle behaviors, offering a balanced synthesis of the theoretical and empirical realms. When applied to downstream multi-vehicle scenarios, the PERL method effectively extracts the characteristics of downstream oscillation propagation, significantly improving the long-term predictive performance for preceding vehicles. This methodology offers a balanced synthesis of theoretical and empirical insights, ensuring high-precision and stable predictions of future vehicle behaviors.

Based on the predictions of preceding vehicles with traffic oscillation, the CAV controller could mitigate oscillation amplification and backward propagation, thus enhancing overall traffic stability. Meanwhile, it is also crucial to account for the behavior of the following vehicles (Gao et al., 2022). Considering the actions of these following vehicles, particularly in mixed traffic scenarios, is essential for

designing safer and more efficient trajectories (Mohammadian et al., 2023). This anticipation helps minimize the occurrence of sudden driving maneuvers, thereby enhancing the safety and fluidity of traffic overall. Given that traffic will likely comprise a mix of autonomous and human-driven vehicles for the foreseeable means, this presents distinct challenges and opportunities for trajectory optimization. Human-driven Vehicles (HVs) tend to display less predictable behavior compared to CAVs. This unpredictability increases the complexity of trajectory planning for CAVs, requiring more sophisticated prediction algorithms and adaptive control strategies, particularly when a CAV is followed by an HV.

Reflecting on the identified research gaps, this study aims to introduce a physics-enhanced CAV controller. Our approach is developing control strategies specifically tailored for mixed platoons comprising both CAVs and HVs based on predicted information of the downstream traffic from the PERL model. In particular, we focus on scenarios where a CAV is followed by an HV, effectively addressing the dynamics of mixed traffic flows. This strategy incorporates the formulation of a Model Predictive Control (MPC) system that considers the formation of both CAVs and HVs within the control objectives (Chen et al., 2018). While direct control is exerted only over the CAVs, the states of the HVs are also integrated into the optimization objectives, ensuring that the overall strategy accounts for the behavioral patterns of both autonomous and human drivers. By employing the advanced predictive PERL model and adaptive control strategy, our MPC framework optimizes traffic behavior across different vehicle types, considering the interactive dynamics inherent in mixed vehicle streams.

2.3 Control model

With the development of Connected and Autonomous Vehicles (CAVs), there arises the necessity for effective controllers to navigate CAVs in mixed traffic scenarios of CAVs and Human-driven Vehicles (HVs) (Ghiasi et al., 2017; Yao and Li, 2020). Such mixed traffic scenarios are inherently complex and dynamic, primarily due to the unpredictable behaviors of HVs and other road users (K. Yang et al., 2023). This complexity presents a significant challenge for CAV controllers. Inadequate control strategies for CAVs in these environments can result in increased traffic congestion and heightened safety risks.

Mainstream methodologies for controlling CAVs can be classified into three primary types: closed-form controllers, controllers based on Model Predictive Control (MPC), and those utilizing Reinforcement Learning (RL). The closed-form linear (Li, 2022) state-feedback CAV controller has been favored due to its rapid and straightforward implementation facilitated by its analytical representation. However, despite their simplicity, these controllers struggle with constrained optimization frameworks involving multiple explicit objectives and constraints, making them less adaptable to the dynamic driving environments found in mixed traffic. The MPC-based controller, supported by a flexible framework, addresses this limitation by optimizing multiple objectives with constraints in a rolling horizon (Zhou et al., 2019). Nonetheless, MPC often necessitates convex problem formulations and imposes a relatively high computational burden, hindering high-resolution real-time implementation. Some learning-based MPCs employ neural networks to solve problems, which mitigates slow-solving issues. However, these systems still face inherent challenges within the MPC framework, as their optimizer is trained through imitation learning based on datasets of MPC solutions. This training approach inherently limits their performance to not exceed that of the quality of the underlying MPC controller (Sacks et al., 2023).

In contrast to MPC, RL-based control facilitates fast computing for real-time implementation (He et al., 2024; Qu et al., 2020). Furthermore, RL-based controllers can capture the nonlinear and stochastic characteristics of complex systems due to the capabilities of neural networks. However, as summarized from previous literature, RL encounters several challenges regarding applicability in mixed traffic scenarios. In environments where autonomous and human-driven vehicles interact, the unpredictability of human behavior adds complexity to the driving context. RL's inherent issues with generalization mean it might not reliably interpret or react to the diverse behaviors encountered among different drivers, which could lead to safety risks or inefficient traffic flow. Moreover, RL's challenges with safe exploration become particularly critical in mixed traffic, as the system should navigate safely without extensive prior exposure to every possible driving scenario, potentially leading to unsafe actions in unanticipated situations. These limitations underscore the need for robust testing and the integration of safety-oriented strategies within RL frameworks before deployment in dynamic, real-world environments.

Safe exploration stands out as a significant challenge for RL applications in the domain of autonomous vehicle control. The lack of safety assurance poses a major obstacle to RL application in real-world scenarios, manifesting in two key aspects. First, the generalization ability of RL has long remained an unsolved issue. Generalization requires RL models to be robust to variations in their environments and able to transfer and adapt to unseen (but similar) environments during deployment. However, current RL methods often evaluate the policy in the same environment it was trained in (Kirk et al., 2023). Over-reliance on data-driven approaches exposes the system to data biases, leading to overfitting of the training data. When deployed in the real world, RL and other deep learning models often encounter previously unseen categories of samples—out-of-distribution (OOD) data—posing significant challenges. This limitation hinders existing autonomous systems from effectively addressing long-tail and cross-domain issues, restricting their safety and adaptability in new environments (Li et al., 2023). Second, RL may learn behaviors that violate physical laws because it relies on function approximation and representation learning (Cao et al., 2024), making it heavily dependent on data quality. This is a common issue for learning-based models (Long et al., 2024b). Thus, RL may not learn skills not provided or rare in the demonstration data. Moreover, many RL-based vehicle control approaches, such as Adaptive Cruise Control (ACC) and Cooperative Adaptive Cruise Control (CACC), address safety by incorporating it into the reward function, aka Lagrangian relaxation (Shi et al., 2023; Tang et al., 2024; Chen et al., 2022), converting hard constraints into penalty terms within the objective function. Consequently, until the policy converges, we cannot expect constraints to be fully considered and met (Chen et al., 2022; Y. Yang et al., 2023; Zhao et al., 2023). Ensuring vehicles explore new behaviors under strict safety constraints necessitates strict safety measures from the outset of training, which is challenging to implement and requires prior knowledge about the environment or task (Yue et al., 2024).

3 PERL-BASED VEHICLE TRAJECTORY PREDICTION

This chapter proposes the PERL framework that adjusts a physics model prediction with a corrective residual predicted from a residual learning NN model. The integration of the physics model preserves interpretability and tremendously reduces the amount of training data compared to pure NN models. We apply PERL to a vehicle trajectory prediction problem with real-world trajectory data of both HV and AV, using an adapted Newell car-following model as the physics model, and four kinds of neural networks (GRU, CLSTM, VAE, and Informer model) as the residual learning model. We compare this PERL model with pure physics models, NN models, and other PINN models. The result reveals that PERL yields the best prediction when the training data is small. The PERL model has fast convergence during training. Moreover, the PERL model requires fewer parameters to achieve similar predictive performance compared to NN and PINN models. Sensitivity analysis shows the PERL model consistently outperforms other models using different physics and residual learning models. Among these, the PERL model based on CLSTM achieved the most accurate predictions.

This chapter is organized as follows. Section 3.1 describes the prediction problem and the methodology employed in the study, including the PERL model framework and the baseline models. Section 3.2 presents the use case of vehicle trajectory prediction for the validation of the PERL model. Section 3.3 provides an example of the application of the PERL model, including details of the physics and residual learning components and the baseline models. Section 3.4 presents the results of the comparative analysis, including multi-step prediction comparisons, convergence comparisons, and sensitivity analysis. Finally, Section 3.5 concludes the paper with a summary of findings and future research directions.

3.1 Methodology

This section describes the prediction problem and the PERL model. Denote the input state (e.g., traffic states) space and the output response (e.g., set of longitudinal accelerations of the subject vehicle) space of a physics system by $\mathcal{S}$ and $\mathcal{Y}$, respectively, where M and N denote the numbers of dimensions of

these two spaces, and $\mathbb{R}$ is the set of real numbers, respectively. Let $g(\cdot)\colon \mathcal{S} \to \mathcal{Y}$ denote the ground-truth system that has response $g(s)$ for each input state $s \in \mathcal{S}$.

The prediction problem is to construct a prediction function $f(\cdot\,|\theta)\colon \mathcal{S} \to \mathcal{Y}$ such that $f(s|\theta)$ is the prediction of $g(s)$, $\forall s \in \mathcal{S}$, where $\theta \in \Theta$ represents a tunable parameter vector in a parameter space Θ. The predictability of function $f(\cdot\,|\theta)$ is evaluated by the prediction error between $f(s|\theta)$ and $g(s), \forall s \in \mathcal{S}$, e.g., the mean squared error (MSE). Solving a prediction problem is usually to find the optimal θ to minimize the prediction error or maximize its predictability. The interpretability of $f(\cdot\,|\theta)$ is characterized by the ability of this function to reflect comprehensible physics rules and relationships, e.g., how elements in θ can be interpreted as certain physics rules and relationships.

3.1.1 PERL

Built upon the defined prediction problem, the PERL model can be described as follows. The PERL prediction is the combination of a physics model component and a residual learning component, formulated as $f^{\mathrm{PERL}}\left(\cdot\,\middle|\theta^{\mathrm{PERL}}\right) \coloneqq f^{\mathrm{Phy}}\left(\mathcal{S}^{\mathrm{Phy}}(\cdot)\middle|\theta^{\mathrm{Phy}}\right) + f^{\mathrm{RL}}\left(\cdot\,\middle|\theta^{\mathrm{RL}}\right)$. The components of this formulation are specified as follows. $f^{\mathrm{Phy}}\left(\mathcal{S}^{\mathrm{Phy}}(s)\middle|\theta^{\mathrm{Phy}}\right)\colon \mathcal{S}^{\mathrm{Phy}} \to \mathcal{Y}^{\mathrm{Phy}}$ is the physics model component that makes an initial prediction of an input state $s \in \mathcal{S}$. Here, $\mathcal{S}^{\mathrm{Phy}}(s) \in \mathcal{S}^{\mathrm{Phy}}$ denote the projection of $s \in \mathcal{S}$ to $\mathcal{S}^{\mathrm{Phy}}$ as a subspace of $\mathcal{S}$. Note that due to parsimony, the input state to the physics model component f^{Phy} may not include all dimensions in $\mathcal{S}$ and can be a projection to its subspace $\mathcal{S}^{\mathrm{Phy}}$. θ^{Phy} is the parameter vector for f^{Phy} in parameter space Θ^{Phy}, which is often a low-dimensional space due to the parsimony of the physics model. $\mathcal{Y}^{\mathrm{Phy}}$ is the range of component f^{Phy} with domain $\mathcal{S}^{\mathrm{Phy}}$ and shall have the same cardinality as $\mathcal{Y}$ since f^{Phy} also predicts g. $f^{\mathrm{RL}}\left(\cdot\,\middle|\theta^{\mathrm{RL}}\right)\colon \mathcal{S} \to \mathcal{Y}^{\mathrm{RL}} \coloneqq \mathcal{Y} - \mathcal{Y}^{\mathrm{Phy}}$ is the residual learning component such that $f^{\mathrm{RL}}\left(s\middle|\theta^{\mathrm{RL}}\right)$ predicts the residual from the physical model prediction to the ground truth, i.e., $g(s) - f^{\mathrm{Phy}}\left(\mathcal{S}^{\mathrm{Phy}}(s)\middle|\theta^{\mathrm{Phy}}\right), \forall s \in \mathcal{S}$. θ^{RL} is the parameter vector for the residual learning component f^{RL} in parameter space Θ^{RL}. With this, the PERL model prediction $f^{\mathrm{PERL}}\left(s\middle|\theta^{\mathrm{PERL}}\right) \in \mathcal{Y}$ is the summation of the physics model prediction $f^{\mathrm{Phy}}\left(\mathcal{S}^{\mathrm{Phy}}(s)\middle|\theta^{\mathrm{Phy}}\right)$ and the residual prediction $f^{\mathrm{RL}}\left(s\middle|\theta^{\mathrm{RL}}\right), \forall s \in \mathcal{S}$, where the

parameter vector θ^{PERL} is a simple concatenation of those for the physics and residual learning components, i.e., $\theta^{\mathrm{PERL}} \coloneqq [\theta^{\mathrm{Phy}}, \theta^{\mathrm{RL}}] \in \Theta^{\mathrm{Phy}} \times \Theta^{\mathrm{RL}}$.

In a real-world application, let $\mathcal{I} \coloneqq \{1, \cdots, I\}$ denote the index set of observed samples, where $I \in \mathbb{R}^{+}$ is the total number of samples. $\mathcal{I}$ is randomly divided into three sets, i.e., $\mathcal{I}^{\mathrm{Train}} \cup \mathcal{I}^{\mathrm{Val}} \cup \mathcal{I}^{\mathrm{Test}} = \mathcal{I}$, $\mathcal{I}^{\mathrm{Train}} \cap \mathcal{I}^{\mathrm{Val}} = \emptyset$; $\mathcal{I}^{\mathrm{Train}} \cap \mathcal{I}^{\mathrm{Test}} = \emptyset$; $\mathcal{I}^{\mathrm{Val}} \cap \mathcal{I}^{\mathrm{Test}} = \emptyset$. Define ω^{Train} and ω^{Val} as the proportion of the total dataset allocated for training and validation purposes: $\left|\mathcal{I}^{\mathrm{Train}}\right| = \left\lfloor \omega^{\mathrm{Train}} \times \mathcal{I} \right\rfloor$ and $\left|\mathcal{I}^{\mathrm{Val}}\right| = \left\lfloor \omega^{\mathrm{Val}} \times \mathcal{I} \right\rfloor$ where $\lfloor \cdot \rfloor$ denotes the floor function. The testing set index $\mathcal{I}^{\mathrm{Test}}$ consists of the remaining indexes, calculated as $\left|\mathcal{I}^{\mathrm{Test}}\right| = |\mathcal{I}| - \left|\mathcal{I}^{\mathrm{Train}}\right| - \left|\mathcal{I}^{\mathrm{Val}}\right|$. θ^{Phy} is obtained by calibration using a set of observed states $\mathcal{S}^{\mathrm{Train}} \coloneqq \{s_i \in \mathcal{S}\}_{i \in \mathcal{I}^{\mathrm{Train}}}$:

$$\theta^{*\mathrm{Phy}} \coloneqq \underset{\theta^{\mathrm{Phy}} \in \Theta^{\mathrm{Phy}}}{\operatorname{argmin}} \sum_{s \in \mathcal{S}^{\mathrm{Train}}} \left(f^{\mathrm{Phy}}\left(S^{\mathrm{Phy}}(s) \middle| \theta^{\mathrm{Phy}}\right) - g(s) \right)^2 \tag{3-1}$$

θ^{RL} is obtained by training with the training dataset:

$$\theta^{*\mathrm{RL}} = \underset{\theta^{\mathrm{RL}} \in \Theta^{\mathrm{RL}}}{\operatorname{argmin}} \sum_{s \in \mathcal{S}^{\mathrm{Train}}} \left(f^{\mathrm{RL}}\left(s \middle| \theta^{\mathrm{RL}}\right) - \left(g(s) - f^{\mathrm{Phy}}\left(S^{\mathrm{Phy}}(s) \middle| \theta^{*\mathrm{Phy}}\right) \right) \right)^2 \tag{3-2}$$

3.1.2 Baseline models

For comparative analysis, we also evaluate the performance of a pure neural network (NN) prediction model, a pure physics-based prediction model, and a physics-informed neural network (PINN) model. The model descriptions are given as follows.

3.1.2.1 Physics model

A physics model (e.g., a car following model) is denoted as $f^{\mathrm{Phy}}\left(S^{\mathrm{Phy}}(s) \middle| \theta^{\mathrm{Phy}}\right): \mathcal{S}^{\mathrm{Phy}} \to \mathcal{Y}^{\mathrm{Phy}}$. Here $S^{\mathrm{Phy}}(s) \in \mathcal{S}^{\mathrm{Phy}}$ denote the projection of $s \in \mathcal{S}$ to $\mathcal{S}^{\mathrm{Phy}}$ as a subspace of $\mathcal{S}$. $\mathcal{Y}^{\mathrm{Phy}}$ is the range of component f^{Phy} with domain $\mathcal{S}^{\mathrm{Phy}}$ and have the same cardinality as $\mathcal{Y}$. θ^{Phy} is the parameter vector for f^{Phy} in a low-dimensional parameter space Θ^{Phy} and is determined through calibration as specified in Eq. (3-1).

3.1.2.2 NN model

An NN model is denoted as $f^{\mathrm{NN}}(s|\theta^{\mathrm{NN}}): \mathcal{S} \to \mathcal{Y}^{\mathrm{NN}}$ that makes a prediction of an input state $s \in \mathcal{S}$. $\mathcal{Y}^{\mathrm{NN}}$ is the range of component f^{NN} with domain $\mathcal{S}$ and have the same cardinality as $\mathcal{Y}$. θ^{NN} is the parameter vector for f^{NN} in parameter space Θ^{NN} and obtained by training using the training dataset:

$$\theta^{*\mathrm{NN}} = \underset{\theta^{\mathrm{NN}} \in \Theta^{\mathrm{NN}}}{\operatorname{argmin}} \sum_{s \in \mathcal{S}^{\mathrm{Train}}} \left(f^{\mathrm{NN}}(s|\theta^{\mathrm{NN}}) - g(s) \right)^2 \tag{3-3}$$

3.1.2.3 PINN model

A PINN model denoted as $f^{\mathrm{PINN}}(s|\theta^{\mathrm{PINN}}): \mathcal{S} \to \mathcal{Y}^{\mathrm{PINN}}$ that predicts an input state $s \in \mathcal{S}$. $\mathcal{Y}^{\mathrm{PINN}}$ is the range of component f^{PINN} with domain $\mathcal{S}$ and have the same cardinality as $\mathcal{Y}$. In the PINN model, the physics model component $f^{\mathrm{Phy}}(S^{\mathrm{Phy}}(s)|\theta^{\mathrm{Phy}})$ supports the training of the NN component $f^{\mathrm{NN}}(s|\theta^{\mathrm{NN}})$. During training, the physics model enhances the loss function of the NN component, aiming to align the NN predictions closely with both the physics model predictions and the ground truth. Thus, the loss function of the NN component contains two parts LOSS_1 and LOSS_2:

$$\mathrm{LOSS}_1 \coloneqq \left(f^{\mathrm{NN}}(s|\theta^{\mathrm{NN}}) - g(s) \right)^2 \tag{3-4}$$

$$\mathrm{LOSS}_2 \coloneqq \left(f^{\mathrm{NN}}(s|\theta^{\mathrm{NN}}) - f^{\mathrm{Phy}}(S^{\mathrm{Phy}}(s)|\theta^{\mathrm{Phy}}) \right)^2 \tag{3-5}$$

$$\theta^{\mathrm{PINN}} = \underset{\theta^{\mathrm{PINN}} \in \Theta^{\mathrm{PINN}}}{\operatorname{argmin}} \sum_{s \in \mathcal{S}^{\mathrm{Train}}} \mu \cdot \mathrm{LOSS}_1 + (1-\mu) \cdot \mathrm{LOSS}_2 \tag{3-6}$$

where μ is the weight hyperparameters to be tuned. The parameter vector $\theta^{\mathrm{PINN}} \in \Theta^{\mathrm{PINN}}$ is a concatenation of the physics component and NN component, i.e., $\theta^{\mathrm{PINN}} \coloneqq [\theta^{\mathrm{Phy}}, \theta^{\mathrm{NN}}] \in \Theta^{\mathrm{Phy}} \times \Theta^{\mathrm{NN}}$.

3.2 PERL Use case: vehicle trajectory prediction

Section 2 introduced the PERL model methodology. In this section, we apply the PERL model to a specific prediction problem: vehicle trajectory prediction, as described below. As shown in Figure 3-1, an input state stems from a set of K consecutive vehicle trajectories, indexed by $k \in \mathcal{K} \coloneqq \{1,2,\cdots,K\}$ from the upstream lead vehicle indexed by 1 through the downstream ego vehicle by K, at T^{b} historical time points $\mathcal{T}^{\mathrm{b}} \coloneqq \{t_0 - (T^{\mathrm{b}}-1)\delta, \cdots, t_0 - \delta, t_0\}$ separated by time interval δ. The input state is defined as $s =$

$[a_{kt}, v_{kt}, \Delta d_{kt}]_{\forall k \in \mathcal{K}, t \in \mathcal{T}^{\mathrm{b}}}$ where a_{kt}, v_{kt}, and Δd_{kt} are the acceleration, speed, and spacing (from the preceding vehicle) of vehicle k at time t, respectively. With this, $|\mathcal{S}| = 3|\mathcal{K}|T^{\mathrm{b}}$. The corresponding output response $g(s)$ is the ground-truth trajectory of ego vehicle K at future time points $\mathcal{T}^{\mathrm{f}} \coloneqq \{t_0 + \delta, t_0 + 2\delta, \cdots, t_0 + T^{\mathrm{f}}\delta\}$, i.e., $g(s) = [a_{Kt}]_{t \in \mathcal{T}^{\mathrm{f}}}$ where a_{Kt} is the ground-truth acceleration of ego vehicle K at a future time point t. With this, $|\mathcal{Y}| = T^{\mathrm{f}}$.

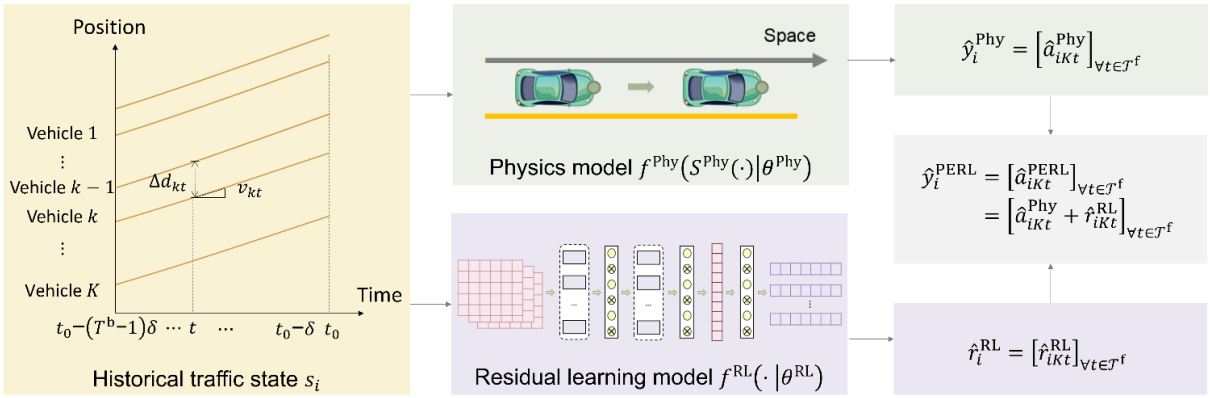


Figure 3-1 Workflow of the PERL model (example in vehicle trajectory prediction).

In the physics component, $S^{\mathrm{Phy}}(\cdot)$ projects s to a much lower-dimensional space, i.e., $S^{\mathrm{Phy}}(s) = [a_{kt_k}, v_{kt_k}, \Delta d_{kt_k}]_{k \in \mathcal{K}, t_k \in [t_0 - \tau_k, \cdots, t_0]}$ or its subset, where τ_k is a short reaction time depending on the specific physics model $t_0 - \tau_k \in \mathcal{T}^{\mathrm{b}}$. With this, $|\Theta^{\mathrm{Phy}}|$ is on the order of $|S^{\mathrm{Phy}}(s)|$, i.e., $O(|S^{\mathrm{Phy}}(s)|)$, and thus much smaller than $|\mathcal{S}|$. $\mathcal{Y}^{\mathrm{Phy}}$ is the predicted acceleration of vehicle K: $\left[\hat{a}_{Kt}^{\mathrm{Phy}}(\theta^{\mathrm{Phy}})\right]_{\forall t \in \mathcal{T}^{\mathrm{f}}} = f^{\mathrm{Phy}}\left(S^{\mathrm{Phy}}(s)\middle|\theta^{\mathrm{Phy}}\right)$. Therefore, we get the residual acceleration predicted by the physics model $r_{Kt}(\theta^{\mathrm{Phy}}) \coloneqq g(s) - \hat{a}_{Kt}^{\mathrm{Phy}}(\theta^{\mathrm{Phy}}), \forall t \in \mathcal{T}^{\mathrm{f}}$.

In the residual learning component, $|\Theta^{\mathrm{RL}}|$ is generally polynomial in relation to $|s|$ and the architecture of the model, thus $|\Theta^{\mathrm{RL}}|$ is likely much greater than $|\Theta^{\mathrm{Phy}}|$. $\mathcal{Y}^{\mathrm{RL}}$ is the predicted residual of

vehicle K: r_{Kt}, i.e., $\left[\hat{r}_{Kt}^{\mathrm{RL}}(\theta^{\mathrm{RL}})\right]_{\forall t\in\mathcal{T}^{\mathrm{f}}} = f^{\mathrm{RL}}\left(s\middle|\theta^{\mathrm{RL}}\right)$. Therefore, the output of the PERL model is the $f^{\mathrm{PERL}}\left(s\middle|\theta^{\mathrm{PERL}}\right) = \left[\hat{a}_{Kt}^{\mathrm{Phy}}(\theta^{\mathrm{Phy}}) + \hat{r}_{Kt}^{\mathrm{RL}}(\theta^{\mathrm{RL}})\right]_{\forall t\in\mathcal{T}^{\mathrm{f}}}$.

3.3 PERL APPLICATION example

Based on the PERL vehicle trajectory prediction use case, this section gives a detailed example that demonstrates the application of the PERL model to vehicle trajectory prediction, including data preparation, model components adoption for the physics component and residual learning component, and baseline model adoption.

3.3.1 Data and metrics

We utilized two types of data in our study. For the validation of the PERL model, we adopted the NGSIM US101 data (Dong et al., 2021), focusing on human-driven vehicles (HV). In the sensitivity analysis, we employed data on purely autonomous vehicle (AV) platoons from OpenACC (Makridis et al., 2021) to validate the model's generalizability across different vehicle trajectory types.

To ensure sufficient training and comprehensive evaluation, we thoroughly prepared the data and evaluated methods based on different scenarios. A total of I =20,000 samples, $\delta = 0.1s$, $\omega^{\mathrm{Train}} = 0.6$, $\omega^{\mathrm{Val}} = 0.2$. Each sample comprises $K = 4$ consecutive vehicles. We evaluated two typical prediction scenarios: one-step and multi-step predictions in this study. For one-step prediction: $T^{\mathrm{b}} = 50$, $T^{\mathrm{f}} = 1$; for multi-step prediction, $T^{\mathrm{b}} = T^{\mathrm{f}} = 50$. The predicted speed of vehicle K at time t in sample $i \in \mathcal{I}$ is calculated as $\hat{v}_{iKt} = v_{iKt_0} + \sum_{t'=t_0}^{t} \hat{a}_{iKt'}\delta, \forall t' \in \mathcal{T}^{\mathrm{f}}$. The evaluation for one-step and multi-step prediction on the test set $\mathcal{I}^{\mathrm{test}}$ includes two metrics: the MSE of acceleration and speed prediction: $\mathrm{MSE}_{\mathrm{test}}^{a} = \frac{1}{|\mathcal{I}^{\mathrm{Test}}|T^{\mathrm{f}}}\sum_{i\in\mathcal{I}^{\mathrm{Test}},t\in\mathcal{T}^{\mathrm{f}}}(a_{iKt} - \hat{a}_{iKt})^2$ and $\mathrm{MSE}_{\mathrm{test}}^{v} = \frac{1}{|\mathcal{I}^{\mathrm{Test}}|T^{\mathrm{f}}}\sum_{i\in\mathcal{I}^{\mathrm{Test}},t\in\mathcal{T}^{\mathrm{f}}}(v_{iKt} - \hat{v}_{iKt})^2$. The convergence of the training process is measured by the change of the MSE of acceleration and speed prediction on the validation set: $\mathrm{MSE}_{\mathrm{val}}^{a} = \frac{1}{|\mathcal{I}^{\mathrm{Val}}|T^{\mathrm{f}}}\sum_{i\in\mathcal{I}^{\mathrm{Val}},t\in\mathcal{T}^{\mathrm{f}}}(a_{iKt} - \hat{a}_{iKt})^2$ and $\mathrm{MSE}_{\mathrm{val}}^{v} = \frac{1}{|\mathcal{I}^{\mathrm{Val}}|T^{\mathrm{f}}}\sum_{i\in\mathcal{I}^{\mathrm{Val}},t\in\mathcal{T}^{\mathrm{f}}}(v_{iKt} - \hat{v}_{iKt})^2$.

3.3.2 PERL physics component

A model adapted from Newell's car-following model is adopted as the physics component of the PERL model. Additionally, for sensitivity analysis, IDM and FVD models serve as alternative physics components for validation.

Initially, we utilized an adapted Newell model characterized by a single parameter w denoting the wave speed, shown in Figure 3-2. The physics model output $\mathcal{Y}^{\text{Phy}} = \left[\hat{a}_{iKt}^{\text{Phy}}(\theta^{\text{Phy}})\right]_{\forall t\in\mathcal{T}^{\text{f}},i\in\mathcal{I}}$ is given by:

$$\hat{a}_{iKt}^{\text{Phy}} = a_{ik't-\frac{D_{iKk't}}{w}}, \forall k' \in \{1,2,\cdots,K-1\}, t \in \mathcal{T}^{\text{f}}, i \in \mathcal{I} \tag{3-7}$$

where $D_{iKk't}$ represents the position distance between vehicle K and vehicle k' at time $t \in \mathcal{T}^{\text{f}}$ in sample $i \in \mathcal{I}$.

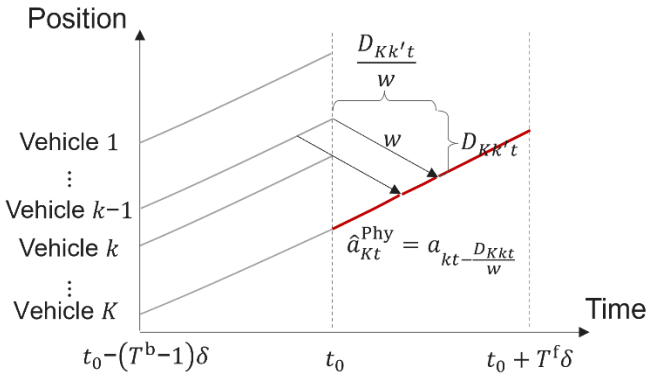


Figure 3-2 Adapted Newell model for vehicle trajectory prediction.

IDM and FVD models are adopted for sensitivity analysis as alternative physics models. IDM model provides a model acceleration function as a continuous function of speed, gap, and speed difference and is expressed as follows:

$$\hat{a}_{iK(t_0+1)}^{\text{Phy}} = \bar{a}\left[1-\left(\frac{v_{iKt_0}}{v^{\text{f}}}\right)^4-\left(\frac{S\left(v_{iKt_0},\Delta v_{iKt_0}\right)}{\Delta x_{Kt_0}}\right)^2\right], \forall i \in \mathcal{I} \tag{3-8}$$

$$S\left(v_{iKt_0},\Delta v_{iKt_0}\right) = S_0 + T^{\text{g}}v_{iKt_0} - \frac{v_{iKt_0}\cdot\Delta v_{iKt_0}}{2\sqrt{\bar{a}\bar{b}}}, \forall i \in \mathcal{I} \tag{3-9}$$

where $\Delta d_{i,n,t}$ is the relative distance between two adjacent vehicles, $S\left(v_{iKt_0}, \Delta v_{iKt_0}\right)$ is the desired space headway function and is calculated from the speed $v_{n,t}$ and the relative speed $\Delta v_{iKt_0} \coloneqq v_{iKt_0} - v_{i(K-1)t_0}, \forall i \in \mathcal{I}$, v^{f} is the free flow speed, $\bar{a}$ is the maximum acceleration, $\bar{b}$ is the comfortable deceleration, S_0 is the minimum space. T^{g} is the desired time gap.

The FVD model focuses on the speed difference between vehicles, considering both relative speeds and spatial gaps. This model comprehensively represents car-following dynamics, especially in high-density traffic scenarios where speed differences play a pivotal role.

$$\hat{a}^{\mathrm{Phy}}_{iK(t_0+1)} = \kappa\left[V\left(\Delta d_{iKt_0}\right) - v_{iKt_0}\right] + \lambda \Delta v_{iKt_0}, \forall i \in \mathcal{I} \tag{3-10}$$

$$V\left(\Delta d_{iKt_0}\right) \coloneqq V_1 + V_2 \tanh\left[C_1\left(\Delta d_{iKt_0} - l_c\right) - C_2\right], \forall i \in \mathcal{I} \tag{3-11}$$

where κ and λ are sensitivity parameters and $V\left(\Delta d_{iKt_0}\right)$ is the optimal speed that the drivers prefer. We adopted the calibrated parameters for $V\left(\Delta d_{iKt_0}\right)$ as $V_1 = 6.75m/s, V_2 = 7.91m/s, C_1 = 0.13m^{-1}$, and $C_2 = 1.54$.

We employed the Monte Carlo method for physics model calibration, which randomly selected a certain number of samples from the dataset. The calibrated parameters of physics models are shown in TABLE 3-1. As the training data size increased, the calibration results tended to stabilize, and the calibrated parameter was near the value in the literature reference (Duret et al., 2011; Laval and Leclercq, 2010). The calibrated parameters of the physics model will be used as initial values in the PINN model and PERL model.

TABLE 3-1 Calibrated parameters of physics models with different training sample sizes.

Model	Parameters	Training sample size						
		300	500	1000	2000	5000	10000	12000
Adapted	$w\ (m/s)$	4.044	4.15	4.122	4.027	4.03	4.01	4.01
Newell model	(variance)	0.088	0.017	0.016	0.018	0.001	0	0
IDM	v^{f} (m/s)	22.821	22.512	22.084	22.377	22.125	22.125	22.495
	(variance)	0.035	0.095	0.013	0.129	0.003	0.003	0.217
	$\bar{a}\ (m/s^2)$	0.682	0.475	0.626	0.54	0.613	0.613	0.911
	(variance)	0.095	0.09	0.175	0.099	0.192	0.192	0
	$\bar{b}\ (m/s^2)$	2.562	2.271	2.473	2.262	2.478	2.478	2.859
	(variance)	0.205	0.141	0.246	0.087	0.237	0.237	0.003
	S_0 (m)	1.712	1.703	1.627	1.608	1.615	1.615	1.627
	(variance)	0.01	0.021	0.013	0.006	0.01	0.01	0.013
	T^{g} (s)	1.2	1.216	1.23	1.162	1.266	1.266	1.132
	(variance)	0.005	0.002	0	0.02	0.003	0.003	0.017
FVD	k	0.108	0.073	0.088	0.044	0.108	0.076	0.007
	(variance)	0.003	0.003	0.002	0.003	0.005	0.006	0
	λ	0.498	0.397	0.478	0.452	0.445	0.284	0.137
	(variance)	0.019	0.041	0.041	0.076	0.029	0.037	0
	V_0 (m/s)	19.779	18.814	19.647	23.034	21.542	22.093	24.158
	(variance)	15.982	10.059	19.536	26.672	17.456	14.257	5
	$b\ (m/s^2)$	10.084	10.746	9.637	8.406	10.327	11.243	7.954
	(variance)	4.935	14.088	5.636	10.33	8.447	11.241	4.34
	β	2.561	3.011	2.824	2.449	2.383	2.133	1.724

(variance)	1.338	1.6	1.585	0.733	2.388	3.52	0.079

3.3.3 PERL residual learning component

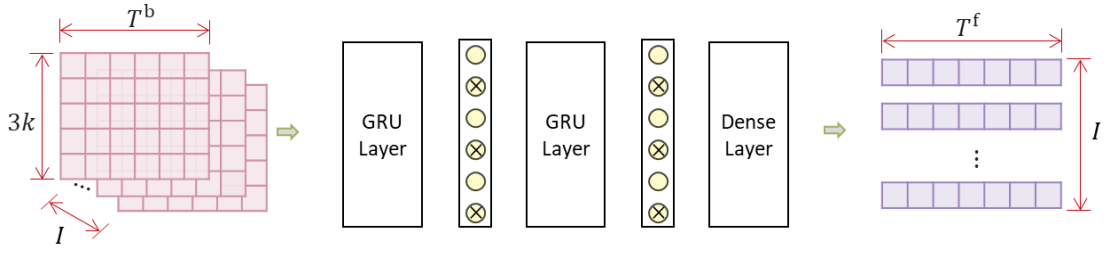
For the residual learning component, we compared four groups of learning-based models: the RNN model, convolution model, generation model, and attention model, to validate the robustness and generalization of the PERL model.

For the RNN model, we chose the Gated Recurrent Unit (GRU) model (Figure 3-3 (a)). GRU models are a kind of recurrent NN that allows connections between neurons to form a directed cycle, thus making it possible to exhibit dynamic temporal behavior. GRU has also been applied in car-following behavior modeling (Wang et al., 2018).

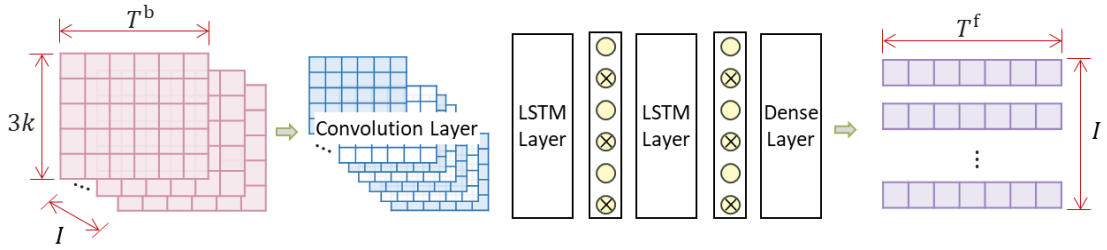
For the convolution model, we selected CLSTM, a model that combines convolutional layers with Long Short-Term Memory (LSTM) (Figure 3-3 (b)) (Lee et al., 2017; Pereira et al., 2022), suitable for spatiotemporal data.

For the generation model, we chose VAE (Variational Autoencoder) (Figure 3-3 (c)), a generative model used to learn the latent distribution of data (Ivanovic et al., 2021; Miguel et al., 2022). VAE offers two advantages: firstly, it exhibits stable training dynamics, and secondly, its interpretable latent space facilitates understanding of the learned representations by the model.

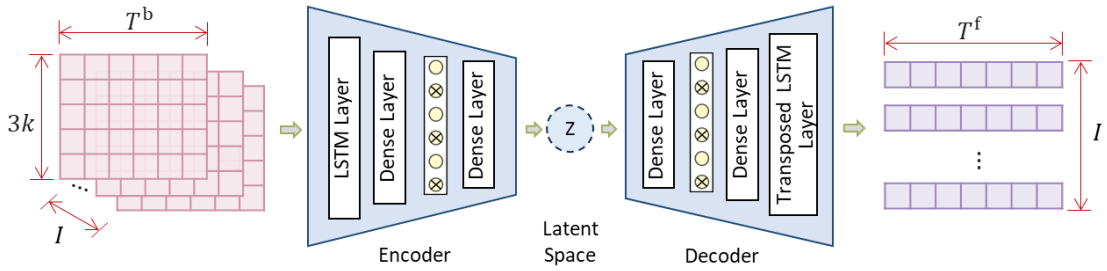
For the attention model, we opted for the Informer model (Figure 3-3 (d)), which is a transformer-based architecture (Zhou et al., 2021). It introduces a masking mechanism in the self-attention mechanism to ensure that the model only attends to information preceding the current time step, preventing leakage of future information when processing sequential data. Additionally, it employs a multi-step input representation method to better capture long-term dependencies in time series data. Figure 3-3 illustrates the structures of four models, focusing only on the primary model layers; activation layers are omitted for clarity. Detailed model configurations and parameters are listed in TABLE 3-2, to TABLE 3-5. The input to each model includes T^{b} time steps of acceleration, speed, and spacing data from K consecutive vehicles.



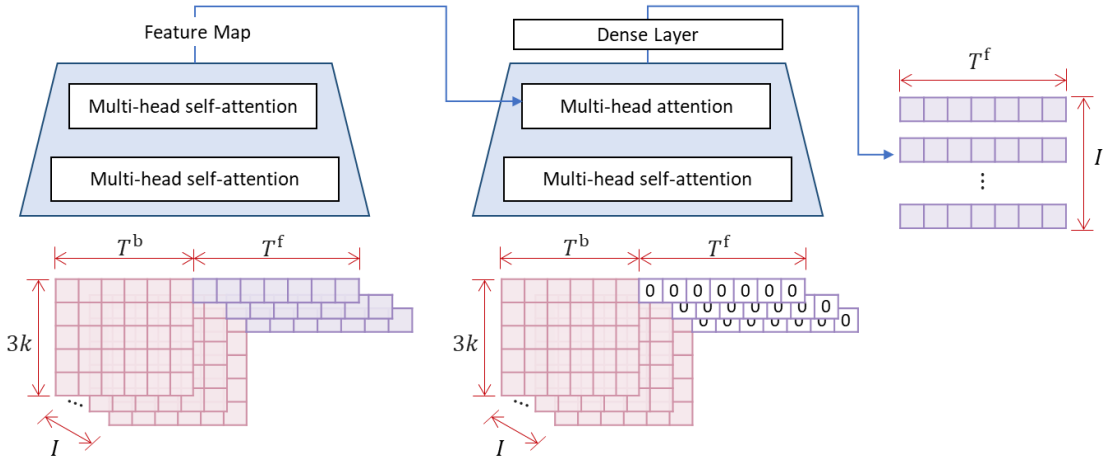
(a) GRU model



(b) CLSTM model



(c) VAE model



(d) Informer model

Figure 3-3 Model structures

TABLE 3-2 Model structure and parameter setting in GRU-based models.

	NN	PINN	PERL
GRU layer	units=128	units=128	units=96
Dropout layer	0.3	0.3	0.3
GRU layer	units=128	units=128	units=64
Dropout layer	0.3	0.3	0.3
Activation layer	relu	relu	relu

TABLE 3-3 Model structure and parameter setting in CLSTM-based models.

	NN	PINN	PERL
Convolution layer			
LSTM layer	units=128	units=128	units=96
Dropout layer	0.3	0.3	0.3
LSTM layer	units=128	units=128	units=64
Dropout layer	0.3	0.3	0.3
Activation layer	relu	relu	relu

TABLE 3-4 Model structure and parameter setting in VAE-based models.

		NN	PINN	PERL
Encoder	Convolution layer			
	LSTM layer	units=128	units=128	units=96
	Dropout layer	0.3	0.3	0.3
	LSTM layer	units=128	units=128	units=64
Latent Space		Demention=20	Demention=20	Demention=20
Decoder	Activation layer	relu	relu	relu

TABLE 3-5 Model structure and parameter setting in VAE-based models.

		NN	PINN	PERL
Encoder	Self-Attention Layer	n_heads=12	n_heads=8	n_heads=8
	Convolutional Layer	Down Convolution layer kernel_size=3 activation=ELU	Down Convolution layer kernel_size=3 activation=ELU	Down Convolution layer kernel_size=3 activation=ELU
	Feed-Forward Network	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1
Decoder	Self-Attention Layer	n_heads=12	n_heads=8	n_heads=8
	Cross-Attention Layer	n_heads=12	n_heads=8	n_heads=8
	Feed-Forward Network	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1	Convolutional Layer1 kernel_size=1 Convolutional Layer2 kernel_size=1

3.3.4 Baseline models adoption

Regarding the three baseline models, the physics model employs the adapted Newell model, aligning with the physics component of PERL. The NN model structure aligns with the residual learning component of PERL but with more parameters. Its prediction target is not the residuals but the future

acceleration of the subject vehicle over T^{f} steps. The PINN model combines the adapted Newell model for its physics component and mirrors the NN model structures in its NN component.

3.4 Results

Based on the methodology and application example introduced in the previous sections, this section gives a detailed comparative analysis of the multi-step prediction performance using various learning-based models, including PERL, NN, PINN, and baseline models, under different training data sizes and conditions.

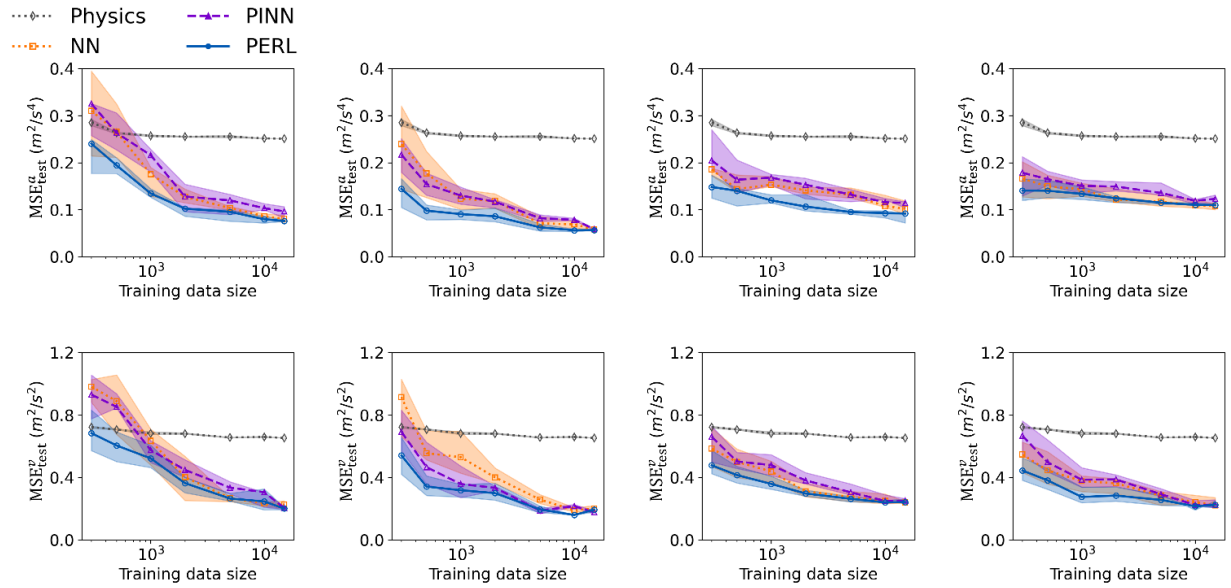
3.4.1 Multi-step prediction comparison

Figure 3-4 compares the prediction performance on acceleration and speed of PERL and baseline using different learning-based models in multi-step prediction. For all four groups of learning methods, the results show that the PERL model exhibits superior predictability with less training data. In scenarios characterized by very limited data availability (e.g., 300 training data size), both the PERL and PINN models outperform the NN model in terms of predictability. As the dataset expands beyond 1000 training samples, the performance differential between the PERL model and the baseline models, including the NN model, becomes markedly less pronounced (as depicted in Figure 2-1 A and B). In this context, all models demonstrate remarkable predictability, achieving acceleration prediction errors as low as $\mathrm{MSE}^{a}_{\mathrm{Test}} = 0.07\ m^2/s^4$.

When comparing the four groups of learning methods with a small training dataset size of 300, models based on the Informer architecture (NN, PINN, PERL) performed notably well. This can be attributed primarily to two advantages of the Informer model: Firstly, the Informer's outputs are relatively stable, ensuring reliable predictions even with limited data. Secondly, unlike traditional recurrent neural networks such as CLSTM and GRU, which depend on the output and hidden states of previous time steps, the Informer model processes data in a non-sequential manner, reducing the accumulation of prediction errors.

With larger datasets, the models based on CLSTM exhibited the best performance, achieving acceleration prediction errors as low as $\mathrm{MSE}^{a}_{\mathrm{Test}} = 0.056\ m^2/s^4$ at a dataset size of 15,000. The

convolutional layers preprocess the data, and the LSTM is sufficient for managing the relatively straightforward car-following dynamics in our study, whereas the attention mechanism of the Informer is more challenging to train and offers limited benefits for this specific scenario. Meanwhile, the GRU-based NN model performed worse than the LSTM-based NN model, primarily due to the simpler structure of GRU, which struggles with issues like vanishing or exploding gradients, particularly in longer sequences. This simplicity may prevent GRU from capturing complex sequence patterns or long-term dependencies as effectively as LSTM. Additionally, the GRU-based PERL model (GRU+Adapted Newell) also underperformed compared to the LSTM-based PERL (LSTM+Adapted Newell). The VAE model was less suited to this prediction task due to the assumption that data is generated from latent variables following a Gaussian distribution, which may restrict the model's ability to fit the trajectory data.



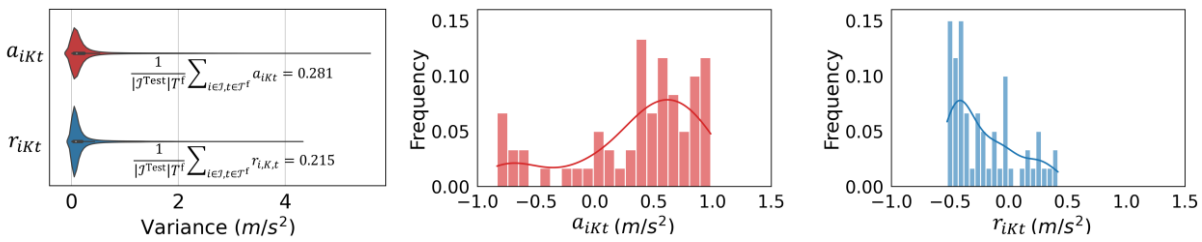
(a) GRU model (b) CLSTM (c) VAE (d) Informer

Figure 3-4 PERL and baseline prediction performance using different learning-based models.

One contributing factor to PERL's superior performance in predicting acceleration residuals, as opposed to the NN model's prediction of acceleration, is the different distribution patterns of acceleration

residuals and acceleration. The variance in acceleration is greater than that of acceleration residuals across all samples, as depicted in Figure 3-4. This difference is further highlighted by comparing the distributions of acceleration and acceleration residuals in a single sample, illustrated in Figure 3-5 (a) and (b). The observed lesser variance in acceleration residual suggests that in scenarios with limited data, the distribution of acceleration residual may more accurately mirror the actual distribution, facilitating better prediction outcomes. On the other hand, the higher variance in acceleration might lead to a less accurate representation of the true distribution in small sample scenarios, thereby potentially diminishing the prediction accuracy. These findings align with the operational principles of the PERL model, which chiefly engages in predicting the residual of the physics model, a task seemingly well-suited for situations with restricted data availability. This stands in contrast to the PINN model, which aims at the prediction of acceleration, a process that could be adversely affected by the noted higher variance, especially in data-scarce conditions. Hence, this visual representation further underscores the sample efficiency and effectiveness of the PERL model in acceleration prediction, particularly when data resources are limited.

Figure 3-4 shows that when the training data size is 12000, the prediction accuracy of the PERL, PINN, and NN models is comparable, and all of them are superior to the physics model. This is validated by the six examples in Figure 3-6. Results indicate that both PINN and NN models can predict the overall trend of acceleration changes, but they fail to fit the local acceleration oscillation. The physics model can account for local acceleration oscillation based on physical principles, but as the physics model is calibrated using multiple vehicle trajectories, its results deviate significantly from real-world values. Only the PERL model can predict the overall trend of acceleration change and keep the local acceleration oscillation results from its physics model component.



(a) (b) (c)

Figure 3-5 Differences in the distribution of acceleration $\boldsymbol{a_{iKt}}$ and acceleration residual $\boldsymbol{r_{iKt}}$. (a) Variance distribution comparison of acceleration $\boldsymbol{a_{iKt}}$ and acceleration residual $\boldsymbol{r_{iKt}}$ of all samples. (b) Distribution of acceleration $\boldsymbol{a_{iKt}}$ in one sample. (c) Distribution of acceleration residual $\boldsymbol{r_{iKt}}$ in one sample.

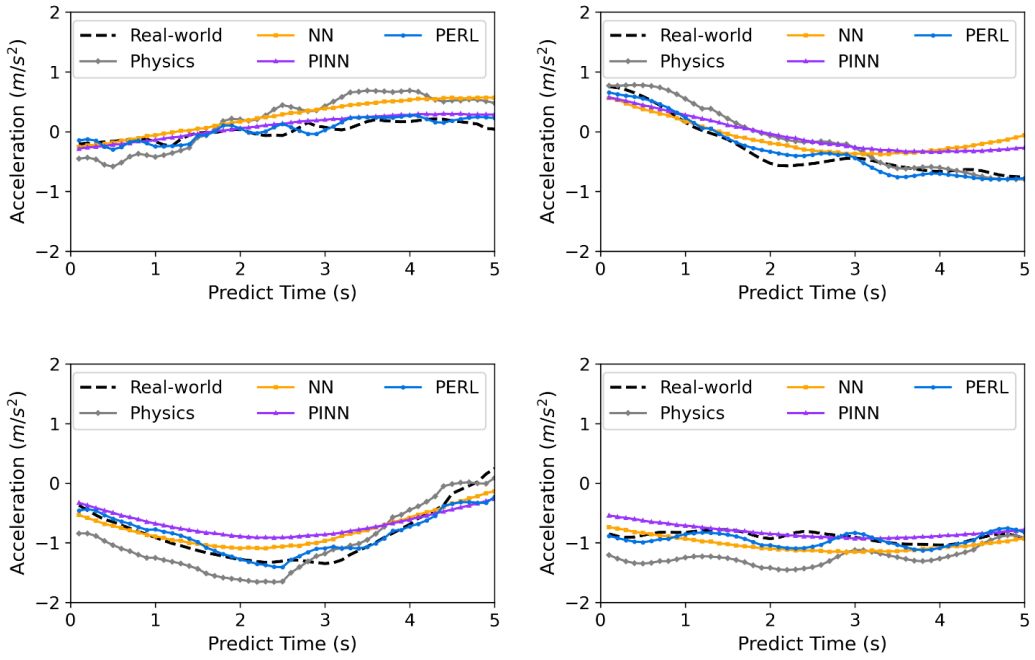
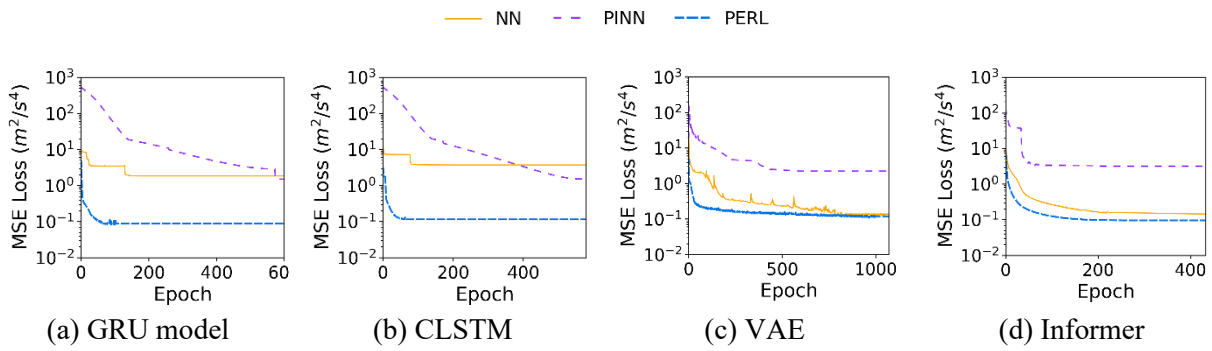


Figure 3-6 Comparison of real-word acceleration and predicted result of PERL (Newell+CLSTM) model and baseline models using CLSTM.

3.4.2 Convergence comparison

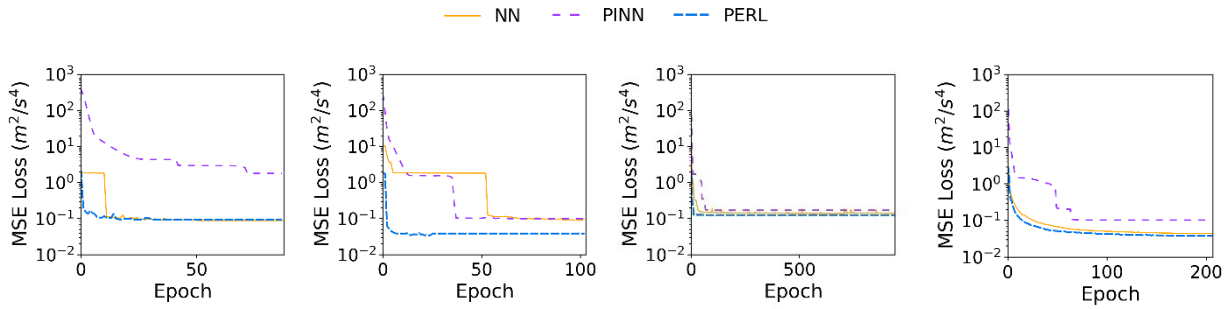
To ensure a fair comparison of convergence among different models, we aligned the number of parameters in the NN, PINN, and PERL models. Specifically, we standardized the LSTM architecture

within the NN component of the NN and PINN models and the residual learning component of the PERL model, setting both LSTM layers to have 128 and 64 units, respectively. The result shows that PERL converges more rapidly than both the NN and PINN models with various training data sizes (Figure 3-7 and Figure 3-8). In scenarios with limited data (600 training data size), the PINN model converges at approximately 80 epochs, and the NN model converges at approximately 140 epochs, whereas the PINN model requires a substantially shorter time, around 1000 epochs (Figure 3-7). This unstable training behavior of PINN arises because different parts of the loss function dominate the decrease at various times: sometimes, it's influenced by the neural network, and other times by the physics rules. This unbalanced back-propagated gradient results in an unstable learning process (Wang et al., 2022). The NN model also converges slower than PERL despite having a near-identical number of parameters (Figure 3-8). This is because the NN model needs to learn all the kinematic rules, while PERL only has to learn the features of the residuals. The convergence speed of PINN is also slower than that of the NN model because its objective function includes terms from both the neural network and the physics model, making the gradient search process more challenging. This result highlights the advantage of the PERL model, which employs the same LSTM architecture and nearly identical number of parameters as the NN and PINN models but uses a more focused prediction approach that enables faster convergence and more efficient computation using fewer parameters.



(a) GRU model (b) CLSTM (c) VAE (d) Informer

Figure 3-7 Training processes of the NN, PINN, and PERL models with a sample size of 1000, utilizing various learning-based models.



(a) GRU model (b) CLSTM (c) VAE (d) Informer

Figure 3-8 Training processes of the NN, PINN, and PERL models with a sample size of 15000, utilizing various learning-based models.

3.4.3 Sensitivity analysis

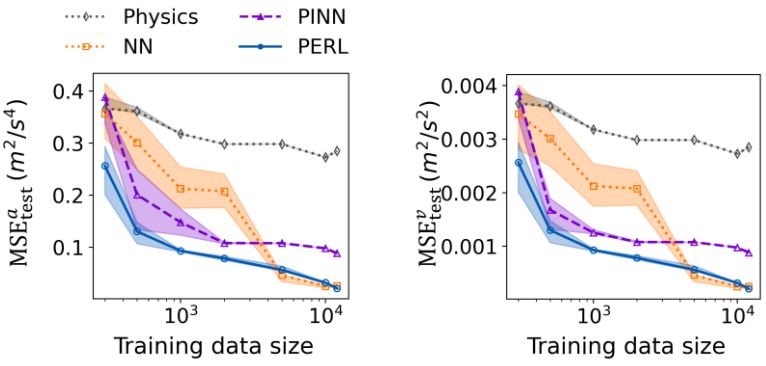
In the sensitivity analysis, we first examined the impact of different physics models on the results. Subsequently, we validated the model's generalization capabilities across various vehicle types.

3.4.3.1 Impacts of Physics car-following models

In single-step prediction tasks, a relatively straightforward task, it's feasible to employ car-following models that only consider the information of the preceding vehicle, such as the Intelligent Driver Model (IDM) (Treiber et al., 2000) and the Full Velocity Difference (FVD) model (Yao et al., 2019). We integrated the PERL model with the IDM and FVD models, utilizing them as the physics model, and paired them with an LSTM for residual learning.

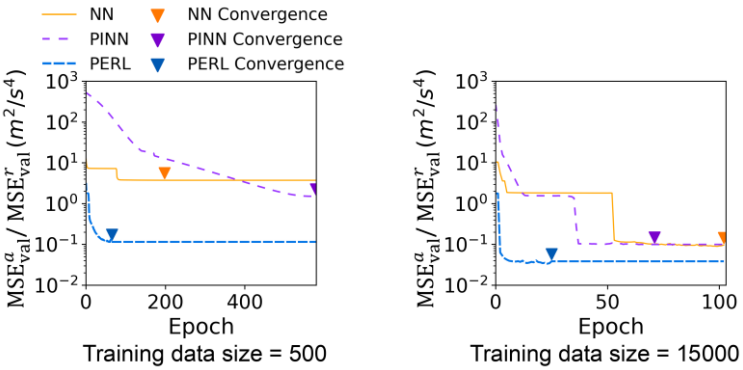
PERL (LSTM+IDM) outperforms the physics model, the NN model (LSTM), and the PINN (LSTM+IDM) when the training data size is smaller than 10000, as shown in Figure 3-9. This suggests the versatility of the PERL model, highlighting its compatibility with the IDM physics model. The IDM model effectively captures certain physics laws, and the residuals in acceleration prediction from the IDM model can be proficiently learned by the LSTM. The convergence of the IDM-based PERL (LSTM+IDM) is faster than both NN (LSTM) and PINN (LSTM+IDM) models, as shown in Figure 3-10. This emphasizes the capability of the IDM model to delineate a significant portion of the physics laws. Moreover, the IDM

model can learn the acceleration residuals with less data, and its training convergence is faster than directly learning the acceleration using the NN model and PINN model.



(a) (b)

Figure 3-9 (LSTM+IDM) and baseline prediction performance.

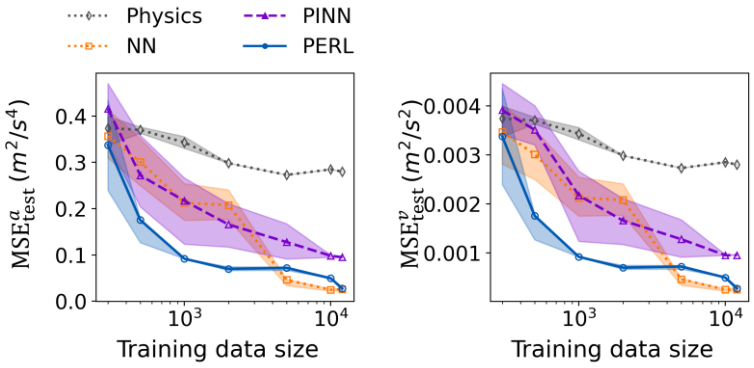


(a) (b)

Figure 3-10 PERL (LSTM+IDM) and baseline prediction performance.

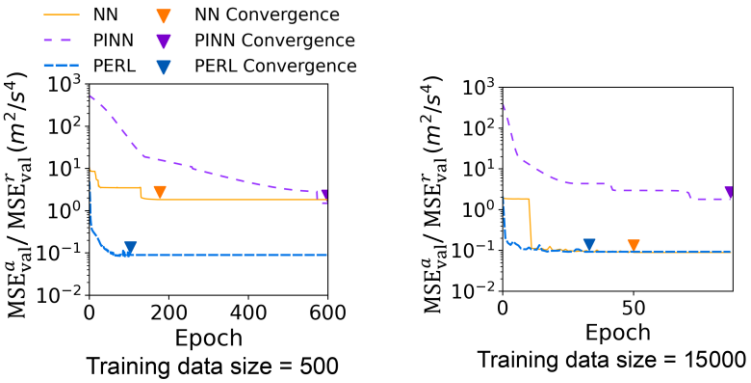
The performance of PERL (LSTM+FVD) is less effective than PERL (LSTM+IDM) with small amounts of data, as shown in Figure 3-11. This mainly results from the poor performance of the FVD model as a physical model when calibrated using small amounts of data. However, even though the physical model is not predictive, PERL (LSTM+FVD) still has a comparative prediction with the NN model and is better

than the PINN model. Results also show the PERL model exhibits quick convergence rates for both small and large training data sizes, as shown in Figure 3-12.



(a) (b)

Figure 3-11 PERL (LSTM+IDM) and baseline prediction performance.



(a) (b)

Figure 3-12 PERL (LSTM+IDM) and baseline prediction performance.

3.4.3.2 Prediction results on AV data

The proposed method is tested on AV data to validate the model's generalization ability. TABLE 3-6 compares the prediction results for acceleration and speed across different training data sizes using the physics model, LSTM-based NN, PINN, and PERL models. We observed that the predictive performance

on AV data is generally better than on HV data. One reason for this is that AV behaviors are governed by specific control algorithms, making their actions more rational and predictable, whereas HV trajectories exhibit many unstable and random behaviors. Another reason is that the range of acceleration distribution in AVs is much tighter than in HVs. Figure 3-13 compares the distribution of acceleration in the AV and HV trajectory datasets, showing that 50% of accelerations are contained within the range of -0.17 to 0.17. This narrower variance in the prediction target makes the prediction task easier.

TABLE 3-6 Comparison prediction performance on HV and AV trajectory using the CLSTM model

Training data size	Trajectory Data	$\text{MSE}^{a}_{\text{test}}$ (m^2/s^4)				$\text{MSE}^{v}_{\text{test}}$ (m^2/s^2)			
		Physics	NN	PINN	PERL	Physics	NN	PINN	PERL
1000	HV data	0.256	0.123	0.130	0.091	0.682	0.530	0.358	0.319
	CAV data	0.195	0.088	0.089	0.086	0.516	0.191	0.217	0.190
15000	HV data	0.251	0.058	0.059	0.056	0.653	0.201	0.177	0.194
	CAV data	0.194	0.047	0.046	0.047	0.512	0.176	0.176	0.176

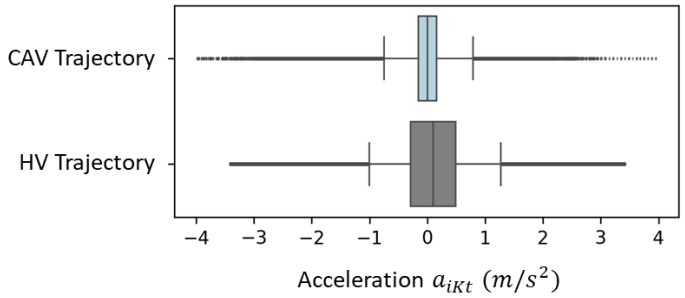


Figure 3-13 Comparison of the distribution of acceleration in CAV and HV trajectory dataset

3.5 CONCLUSION AND FUTURE WORKS

This study presented the novel PERL model as a potent tool for trajectory prediction, aiming to address inherent limitations found in existing physics models and NN models. The PERL model leverages

the power of residual learning, a concept relatively unexplored in the traffic domain, providing the research community with a new perspective on traffic modeling and prediction.

We compared the predictive performance of the PERL model with traditional physics and NN models, using both HV and AV trajectory data. The results demonstrated the superiority of the PERL model in both one-step and multi-step acceleration prediction tasks. Notably, the PERL model consistently outperformed all other tested models across different data sources and vehicle types. PERL also has faster initial convergence during the training process than the traditional NN model and PINN. For the sensitivity analysis, we assessed PERL's performance using a diverse array of residual learning models: GRU, CLSTM, VAE, and Informer models. Each model represents a different neural network approach—GRU and CLSTM for temporal dynamics, VAE for generative capabilities, and Informer for attention mechanisms—highlighting the versatility of PERL across various data processing contexts. Additionally, we integrated various physics-based car-following models, including IDM and FVD, to further validate the adaptability and robustness of the PERL framework.

4 PERL-BASED VEHICLE PREDICTIVE CONTROL

This chapter proposes a PERL-based predictive control method to mitigate traffic oscillation in the mixed traffic environment of CAVs and HVs. The model introduced includes a prediction model and a CAV controller. The PERL-based prediction model precisely predicts the behavior of the preceding vehicle, especially speed fluctuations, to allow sufficient time for the vehicle/driver to respond to these speed fluctuations. For the CAV controller, we employ a Model Predictive Control (MPC) model that considers the dynamics of the CAV and its following vehicles, improving safety and comfort for the entire platoon. The proposed model is applied to an autonomous driving vehicle through Vehicle-in-the-loop (ViL) and compared with real driving data and three benchmark models. Experimental results validate the proposed method in damping traffic oscillation and enhancing the safety and fuel efficiency of the CAV and the following vehicles in mixed traffic with the presence of uncertain human-driven vehicle dynamics and actuator lag (Long et al., 2024a).

This chapter is organized as follows. Section 4.1 presents the investigated problem. Section 4.2 proposes the PERL-based predictive control model. Section 4.3 conducts a ViL experiment to compare the proposed model with the existing traffic dataset benchmark. Section 4.4 concludes this chapter.

4.1 Problem statement

This study investigates a CAV controller using multi-step real-time downstream vehicle trajectory data to assist roadway safety and dampen traffic oscillation in mixed traffic. As shown in Figure 4-1, a stream of vehicles operates in a single-lane roadway segment. The subject CAV is indexed by N. The preceding vehicles (indexed by 1 to $N-1$ from downstream to upstream). At time t, the future longitudinal behavior of the preceding vehicles is predicted from t_0 to t_0+T, based on the historical trajectory from t_0-T to t_0, where T is the observation period. Time is discretized with a small interval δ in this study because vehicle trajectory data is discretized in time.

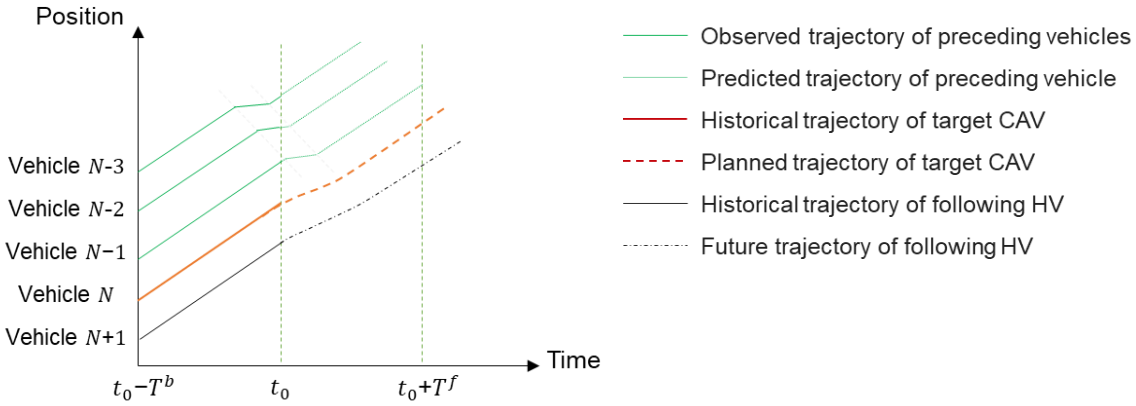


Figure 4-1 CAV trajectory planning based on multi-preceding-vehicle prediction.

The model first predicts the future behavior of vehicle $N-1$, and then the planning model output is the future planned trajectory of vehicle N. The main objective here is to leverage the predictive capabilities of PERL to better anticipate the situational behavior of vehicles in its vicinity and plan AV trajectory accordingly.

Notice that this research is applicable to mixed traffic. The type of preceding vehicles and the following vehicles are not pre-assumed since the heterogeneity of vehicles is considered. We assume that the target vehicle could have the information of its preceding $N-1$ vehicles and the following vehicle. The heterogeneity is captured by their trajectory during the observation period.

The objective is to construct a control model for the target CAV to yield trajectory based on the prediction of the preceding vehicle. Detailed methodology is presented in Section 3.

4.2 Methodology

Based on the problem statement, this section presents the proposed PERL-based predictive control model that aims to mitigate traffic oscillation. As shown in Figure 4-2.

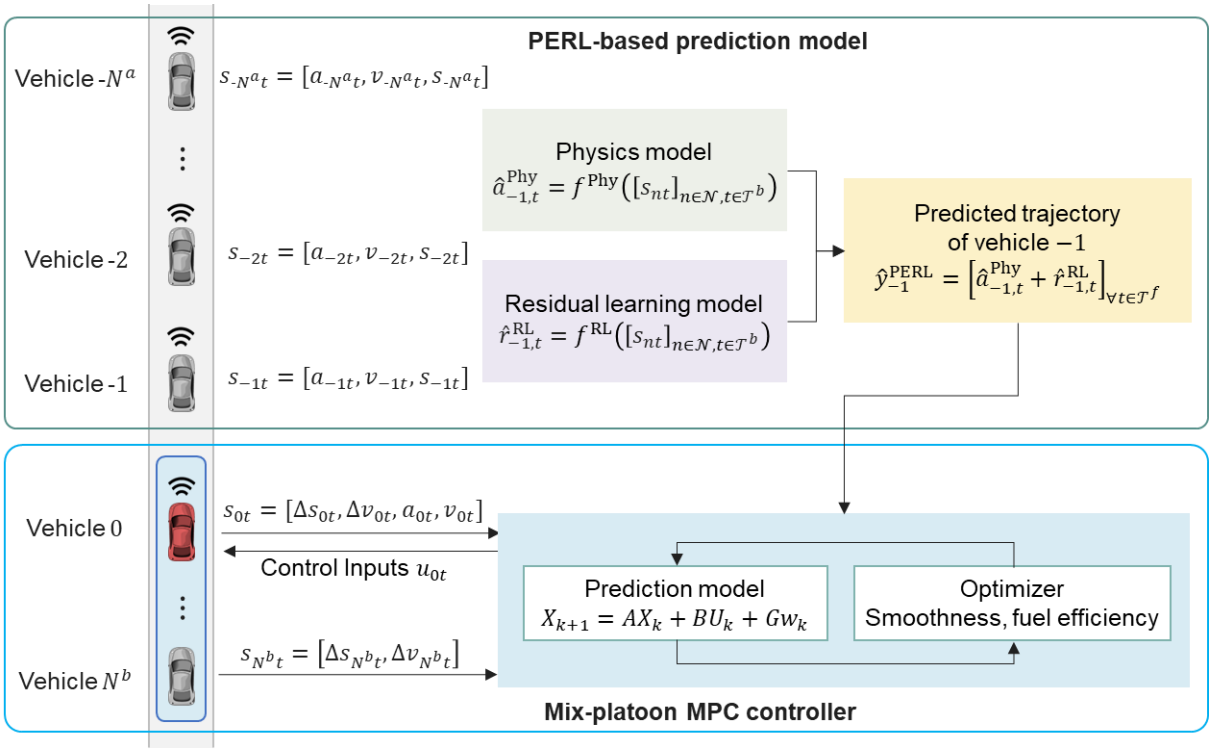


Figure 4-2 PERL-based Predictive Control (PERL-MPC) model.

4.2.1 PERL-based Prediction Model

This component is designed to forecast the future states of the preceding vehicle, incorporating physical laws and residual learning mechanisms to enhance prediction accuracy and stability. By integrating physical models with deep learning techniques, the PERL model effectively anticipates traffic oscillations, providing a robust foundation for downstream control decisions (Long et al., 2024b).

As shown in Figure 4-2, an input state stems from a set of K consecutive vehicle upstream of the subject CAV, indexed by $k \in \mathcal{K} \coloneqq \{-1, -2, \cdots, -K\}$ from the upstream lead vehicle indexed by -1 through the upstream vehicle by $-K$, at T^{b} historical observation points $\mathcal{T}^{\text{b}} \coloneqq \{t_0 - (T^{\text{b}} - 1)\delta, \cdots, t_0 - \delta, t_0\}$. The input state is defined as $s = [a_{nt}, v_{nt}, \Delta d_{nt}]_{\forall n\in\mathcal{N},t\in\mathcal{T}^{\text{b}}}$ where a_{kt}, v_{kt}, and Δd_{kt} are the acceleration, speed, and spacing (from the preceding vehicle) of vehicle k at time t in sample i, respectively. The prediction output is the trajectory of preceding vehicle -1 at future time points $\mathcal{T}^{\text{f}} \coloneqq$

$\{t_0+\delta, t_0+2\delta, \cdots, t_0+T^{\mathrm{f}}\delta\}$, i.e., $\hat{y}_{-1}^{\mathrm{PERL}} = [\hat{a}_{-1t}]_{t\in\mathcal{T}^{\mathrm{f}}}$ where $\hat{a}_{-1t}$ is the predicted acceleration of vehicle -1 at a future time point t.

In the physics component, $S^{\mathrm{Phy}}(\cdot)$ projects s to a much lower-dimensional space, i.e., $S^{\mathrm{Phy}}(s) = \left[a_{kt_k}, v_{kt_k}, \Delta d_{kt_k}\right]_{k\in\mathcal{K}, t_k\in[t_0-\tau_k,\cdots,t_0]}$ or its subset where τ_k is a short reaction time depending on the specific physics model $t_0-\tau_k \in \mathcal{T}^{\mathrm{b}}$. With this, $\left|\Theta^{\mathrm{Phy}}\right|$ is on the order of $|S^{\mathrm{Phy}}(s)|$, i.e., $O\left(|S^{\mathrm{Phy}}(s)|\right)$, and thus much smaller than $|\mathcal{S}|$. $\mathcal{Y}^{\mathrm{Phy}}$ is the predicted acceleration of vehicle K: $\left[\hat{a}_{Kt}^{\mathrm{Phy}}\left(\theta^{\mathrm{Phy}}\right)\right]_{\forall t\in\mathcal{T}^{\mathrm{f}}} = f^{\mathrm{Phy}}\left(S^{\mathrm{Phy}}(s)\middle|\theta^{\mathrm{Phy}}\right)$. Therefore, we get the residual acceleration predicted by the physics model $r_{Kt}\left(\theta^{\mathrm{Phy}}\right) \coloneqq g(s) - \hat{a}_{Kt}^{\mathrm{Phy}}\left(\theta^{\mathrm{Phy}}\right), \forall t \in \mathcal{T}^{\mathrm{f}}$.

In the residual learning component, $\left|\Theta^{\mathrm{RL}}\right|$ is generally polynomial in relation to $|s|$ and the architecture of the model, thus $\left|\Theta^{\mathrm{RL}}\right|$ is likely much greater than $\left|\Theta^{\mathrm{Phy}}\right|$. $\mathcal{Y}^{\mathrm{RL}}$ is the predicted residual of vehicle K: r_{Kt}, i.e., $\left[\hat{r}_{Kt}^{\mathrm{RL}}\left(\theta^{\mathrm{RL}}\right)\right]_{\forall t\in\mathcal{T}^{\mathrm{f}}} = f^{\mathrm{RL}}\left(s\middle|\theta^{\mathrm{RL}}\right)$. Therefore, the output of the PERL model is the $f^{\mathrm{PERL}}\left(s\middle|\theta^{\mathrm{PERL}}\right) = \left[\hat{a}_{Kt}^{\mathrm{Phy}}\left(\theta^{\mathrm{Phy}}\right) + \hat{r}_{Kt}^{\mathrm{RL}}\left(\theta^{\mathrm{RL}}\right)\right]_{\forall t\in\mathcal{T}^{\mathrm{f}}}$.

For the physics component, we employ a shockwave-based car-following model. This model utilizes shockwave dynamics to predict vehicle behavior based on the movement of surrounding vehicles. The predicted future speeds are calculated based on the relative positions and the defined wave speed. This model notably expands the scope of traditional car-following models by incorporating the effects of multiple vehicles in proximity, not just the one directly ahead, providing a more comprehensive analysis of traffic dynamics.

For the residual learning component, we utilized a Convolution Long Short-Term Memory (CLSTM) model to effectively capture the dynamic and sequential nature of vehicle driving behaviors. Integrating convolutional layers in the CLSTM model aims to abstract and understand the overall trends and variations within the input data (Yao et al., 2023). These convolutional layers are adept at handling spatial dependencies and mitigating the impact of minor inaccuracies in the data, which can be crucial for maintaining robustness in predictions. The LSTM component of the model aims to capture temporal

dependencies and the sequence of events in the vehicle states over time, aligning with the fundamental car-following rule where vehicles follow each other in sequence from downstream to upstream. This sequence-sensitive processing allows the model to anticipate future vehicle behaviors based on past and current observations, providing a predictive insight that is more aligned with the actual driving behaviors than traditional models.

The proposed CLSTM model structure begins with an input layer that takes in the state of vehicles. This data passes through a convolutional layer where initial feature extraction occurs. Subsequent LSTM layers delve deeper into these features, analyzing the time-related dependencies and evolving conditions in the traffic environment. Dropout layers interspersed among the LSTM layers help prevent overfitting by randomly omitting subsets of features during training, which enhances the model's generalizability. This sequence concludes with dense layers that consolidate the learned features into outputs that predict vehicle behaviors, followed by a final dropout layer to further refine the output by minimizing overfitting. Therefore, the CLSTM architecture combines the strengths of convolutional neural networks in feature extraction and LSTM networks in sequence modeling, making it particularly suitable for complex, dynamic traffic scenarios.

4.2.2 Mix-platoon MPC Model

Building on the predictions generated by the PERL model, the MPC component optimizes the trajectory of the controlled vehicle in real-time. It considers the dynamic constraints of the traffic environment and the vehicle's operational limitations, aiming to minimize the impact of traffic oscillations on the controlled vehicle and, by extension, the surrounding traffic flow. The MPC model's primary objective is to improve traffic efficiency and safety by adjusting the vehicle's speed and position in a predictive manner.

4.2.2.1 Single CAV dynamic model

We introduce a longitudinal dynamics model for a single CAV first. Its nonlinear longitudinal dynamic model can be described as:

$$\begin{cases} \dfrac{ds}{dt} = v \\ \dfrac{dv}{dt} = a \\ \dfrac{da}{dt} = f(v,a) + g(v)\eta \end{cases} \tag{4-1}$$

where s, v, a are the position, velocity, and acceleration of the vehicle respectively, and η is the engine input. Functions f and g are given by

$$f(v,a) = -\frac{2K_d}{m}va - \frac{1}{\tau_n^A}\left[a + \frac{K_d}{m}v^2 + \frac{d_m}{m}\right] \tag{4-2}$$

$$g(v) = \frac{1}{m\tau_n^A} \tag{4-3}$$

where K_d represents the aerodynamic drag coefficient, m the vehicle mass, τ_n^A is the actuator lag, and d_m is the mechanical drag. In this paper, we focus on the longitudinal kinematics of vehicles. Assuming the parameters in (4) (5) are priori known, we adopt the following control law structure to implement feedback linearization:

$$\eta = mu + K_d v^2 + d_m + 2\tau_n^A K_d va \tag{4-4}$$

where u is the desired acceleration, determined by the upper controller.

Substitute equations (4-2)-(4-4) into (4-1), the differential equation of acceleration can be rewritten as

$$\dot{a_n} = \frac{u_n - a_n}{\tau_n^A} \tag{4-5}$$

The objectives of CAV planning are following its preceding vehicle with a desired spacing distance and ensuring safety. Therefore, a constant time headway (CTH) spacing strategy was applied. The desired spacing distance of vehicle n is $s_n^d = hv_n + d$, h and d are the desired constant headway and space at a standstill. Based on the CTH rule, the position error Δs with respect to a desired distance from the preceding vehicle $\Delta s = s_{n-1} - s_n - l_{n-1} - s_n^d$ and l_{n-1} is the length of the preceding vehicle.

Given the system state $\mathbf{x} = [\Delta s, \Delta v, a]^T$, Δv is the velocity error between the ego and the preceding vehicle: $\Delta v = v_{n-1} - v_n$. The control variable $\mathbf{u} = u_n$, where u_n is the desired acceleration, the outside

disturbance is the acceleration of the preceding vehicle $\mathbf{d} = a_{n-1}$, the longitudinal dynamics state-space model of CAV is:

$$\frac{d}{dt}\mathbf{x} = \frac{d}{dt}\begin{pmatrix}\Delta s\\ \Delta v\\ a\end{pmatrix} = \frac{d}{dt}\begin{pmatrix}s_{n-1} - s_n - l_n - s_n^d\\ v_{n-1} - v_n\\ a_n\end{pmatrix} = \begin{pmatrix}v_{n-1} - v_n - l_n - s_n^d\\ a_{n-1} - a_n\\ (u_n - a_n)/\tau_n^A\end{pmatrix} = A\mathbf{x} + \mathrm{B}\mathbf{u} + \mathrm{C}\mathbf{d} \quad (4\text{-}6)$$

where

$$A = \begin{bmatrix}0 & 1 & -h\\ 0 & 0 & -1\\ 0 & 0 & -\frac{1}{\tau_n^A}\end{bmatrix}, B = \begin{bmatrix}0\\ 0\\ \frac{1}{\tau_n^A}\end{bmatrix}, C = \begin{bmatrix}0\\ 1\\ 0\end{bmatrix} \quad (4\text{-}7)$$

4.2.2.2 Car-following behavior of HV

In this paper, we use the IDM as the CF model. The IDM is widely used and studied in the literature as it can successfully produce stop-and-go oscillations in congested traffic. IDM model provides a model acceleration function as a continuous function of velocity, gap, and velocity difference and is expressed as follows:

$$\begin{cases}\dfrac{ds}{dt} = v\\ \dfrac{dv}{dt} = \bar{a}\left[1 - \left(\dfrac{v_t^n}{v^f}\right)^4 - \left(\dfrac{S^*(v_t^n, \Delta v_t^n)}{\Delta x_t^n}\right)^2\right]\end{cases} \quad (4\text{-}8)$$

$$S^*(v_t^n, \Delta v_t^n) = S_0 + t_0 v_t^n - \frac{v_t^n \cdot \Delta v_t^n}{2\sqrt{\bar{a}\bar{b}}} \quad (4\text{-}9)$$

where $\bar{a}$ is the maximum acceleration of the vehicle, $\bar{b}$ is the comfortable deceleration, S_0 and t_0 are parameters representing the minimum desired distance to the car in front and the time headway, respectively. v^f is the desired speed.

4.2.2.3 Heterogeneous Platoon Dynamics Model

Consider one CAV with several HVs driving following it; here, we consider the most simple situation, with only one HV following it, as shown in Figure 4-3. For this platoon, the state and control variables can be defined as $\mathbf{x}^{\mathrm{H}} = (\Delta s_0, \Delta v_0, a_0, \Delta s_1, \Delta s_1)^T$ and $\mathbf{u}^{\mathrm{H}} = (u_0)^T$, and the exogeneous disturbance is $\mathbf{w}^{\mathrm{H}} = (a_{-1}, a_1)$.

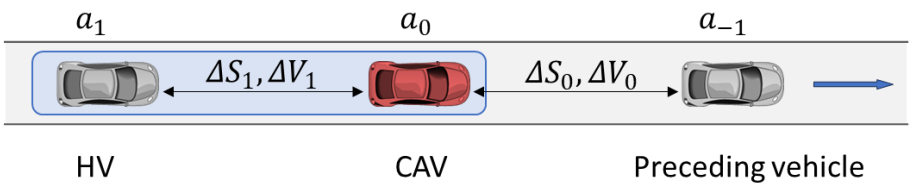


Figure 4-3 Heterogeneous Platoon of a CAV followed by an HV.

Here is the longitudinal dynamics model for the platoon formed by a leading CAV and the following HV:

$$\frac{d}{dt}\mathbf{x}^H = \frac{d}{dt}(\Delta s_0, \Delta v_0, a_0, \Delta s_1, \Delta v_1)^T = \mathrm{A}^{\mathrm{H}}\mathbf{x}^{\mathbf{H}} + \mathrm{B}^{\mathrm{H}}\mathbf{u}^{\mathbf{H}} + \mathrm{C}^{\mathrm{H}}\mathbf{w}^{\mathbf{H}} \tag{4-10}$$

where

$$A^H = \begin{bmatrix} A_{n=1} & 0^{3\times 2} \\ 0^{2\times 2} & D^{2\times 3} \end{bmatrix}, B^H = \begin{bmatrix} 0 \\ 0 \\ 1 \\ \frac{}{\tau_n^A} \end{bmatrix}, C^H = \begin{bmatrix} C & 0^{3\times 1} \\ 0^{2\times 1} & E \end{bmatrix}, D^{2\times 3} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, E = \begin{bmatrix} -T \\ -1 \end{bmatrix} \tag{4-11}$$

4.2.2.4 Model Predictive Control (MPC) Algorithm

Consider a linear discrete-time state-space model given in the following form:

$$\mathrm{x}_{\mathrm{t+1}} = A\mathrm{x}_{\mathrm{t}} + B_1 u_t + B_2 w_t \tag{4-12}$$

$$\mathrm{y}_{\mathrm{t+1}} = C\mathrm{x}_{\mathrm{t}} \tag{4-13}$$

where $\mathrm{x}_{\mathrm{t+1}} \in R^n$ is the system state; $\mathrm{y}_{\mathrm{t+1}} \in R^m$ is the measured output; $u_t \in R^p$ is the control input, and w_t is the disturbance input. We assume that the state and the disturbance vectors can be measured in every sampling period. The designed controller regulates platoon desired accelerations over a time horizon $[t_0, t_0 + N_c]$ to minimize a cost function J representing driving safety, efficiency, and ride comfort. Considering that the strict constraints of MPC may make the optimization problem infeasible, a slack variable is introduced here to soften the constraints. We add the slack variable into the optimization problem:

$$\min_{\Delta \boldsymbol{u}[t_0, t_0+N_c]} J = \sum_{j=0}^{N-1} \left\| \Delta s_{t+j|t} \right\|_Q^2 + \sum_{j=0}^{N_c-1} \left\| \Delta u_{k+j|k} \right\|_R^2 + \varepsilon^T \rho \varepsilon \tag{4-14}$$

where N the predictive horizon length, and N_c the control horizon length. Q, R are the weight matrix of error and input, respectively. ε is the slack vector, and ρ its weight. The optimization objective is subjected to the following constraints, with $\sigma_{\min}^{y}, \sigma_{\max}^{y}$ are specific scaling factors applied to constraints:

$$\Delta s_{\min} + \varepsilon\sigma_{\min}^{y} \leq s_{t+j|t} \leq \Delta s_{\max} + \varepsilon\sigma_{\max}^{y} \tag{4-15}$$

$$v_{\min} + \varepsilon\sigma_{\min}^{v} \leq v_{t+j|t} \leq v_{\max} + \varepsilon\sigma_{\max}^{v} \tag{4-16}$$

$$u_{\min} + \varepsilon\sigma_{\min}^{u} \leq u_{t+j|t} \leq u_{\max} + \varepsilon\sigma_{\max}^{u} \tag{4-17}$$

4.3 Experiments

4.3.1 Experiment settings

We conducted a series of experiments to validate the above-proposed methodology. For the experiment, this section introduced the setting of the ViL experiment, including the data preparation, the researched CAV, the ViL testing environment, and the baseline controllers.

4.3.1.1 Data preparation

In the field experiment, the proposed PERL-based controller is validated using the widely applied Next-Generation Simulation (NGSIM) dataset (NGSIM, 2007). The trajectory of consecutive 6 vehicles is extracted as the result of Baseline 1; For other baseline situations and the proposed situation, the preceding 4 vehicles of the 6 vehicles are used as the trajectory of Vehicle -1 to Vehicle $-K$, $K = 4$.

Six sample trips were chosen for the experiments, which included two sets from acceleration trips, two from deceleration trips, and two from cruising trips. These trips are labeled as Trip 1 through Trip 5, each with a duration of 20 seconds. Considering the heterogeneity of the traffic flow, the headway h^d for each vehicle was calibrated based on their trajectories (Long et al., 2024c).

4.3.1.2 ViL environment

This study employs a ViL approach with the experimental setup illustrated in Figure 4-4 and experimental parameters presented in TABLE 4-1. In this setting, our algorithm sets a maximum speed limit of 15 m/s to ensure safety, considering the constraints of the test track environment. The experimental environment comprises two parts: simulation and field experiment. Six vehicles are simulated in the

simulation, where the trajectories of the first four vehicles are extracted from the total NGSIM dataset. The fifth vehicle is a controlled CAV, and the sixth vehicle is an HV. During the experiment, the trajectories of the preceding vehicles are transmitted to the physical CAV as inputs to the control model. Subsequently, the onboard computer applies PERL-based predictive control to derive the longitudinal target speed, relayed to the vehicle's lower-level actuation system. The vehicle's actual position on the test track is fed back into the simulation environment for updates.

In the field experiment segment, the physical CAV is Lincoln MKZs 2016. The vehicle's lower-level control contains longitudinal and lateral dynamics, where the longitudinal behavior is governed by piece-wise PID control, and an MPC controller manages the lateral behavior.

The field experiments are conducted at the test track in Madison, WI, USA. Figure 4-4 shows the satellite picture of the test track. We conduct experiments on an approximate 300-meter straight road segment. There are two lanes on the road; each lane is 3.5 meters in width, with shoulders on the boundary of the road segment. Moreover, the road segment is level without super-elevation or grades. Figure 4-5 shows the interior view of the experimental vehicle during the experiment. A safety operator is responsible for driving the vehicle to the starting point of the test track and then initiating the program that allows the vehicle's computer to control the vehicle. The safety operator does not operate the accelerator pedal or the steering wheel. In case of emergency situations (e.g., unexpected obstacles), the safety operator will take over the vehicle control to ensure safety. Another individual, the data collector, is responsible for operations related to data collection.

TABLE 4-1 Experiment Parameters

Parameters	Value
Number of Vehicles	6
Time interval	0.1 s
Vehicle length L	4.5 m
Safety distance S_0	1.5 m
Actuation time lag τ	0.1
Communication time lag τ^h	0.2
Acceleration boundary	$[-4\ m/s^2, 4\ m/s^2]$
Speed boundary	$[0\ m/s, 15\ m/s]$

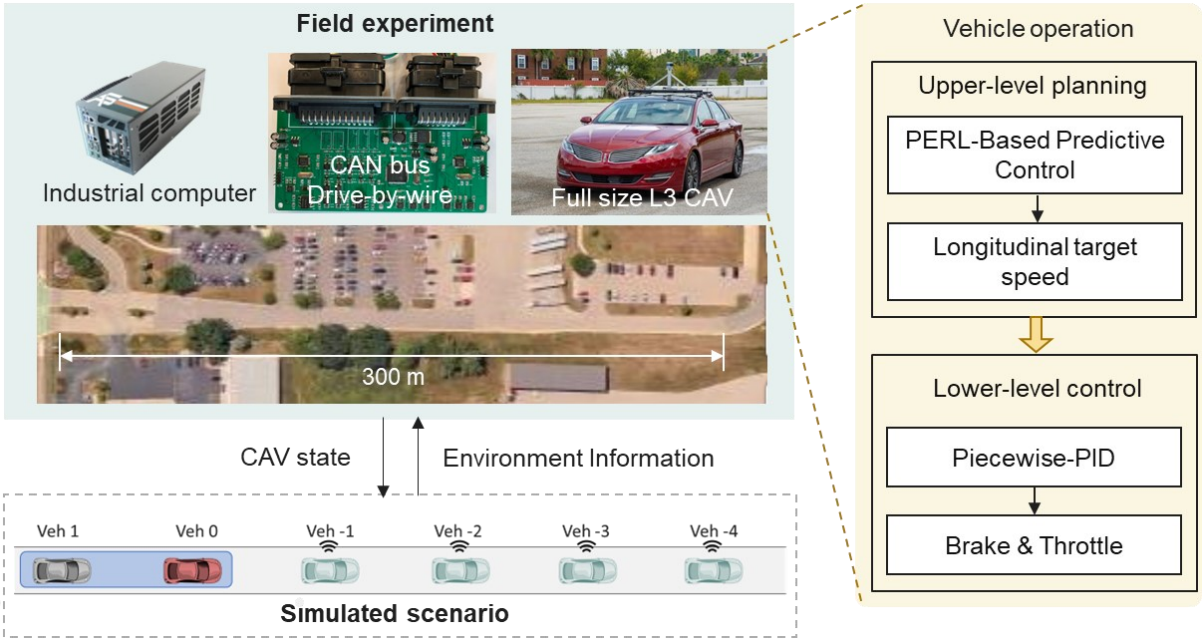


Figure 4-4 Vehicle-in-the-loop (ViL) environment.

Figure 4-5 Interior view of the experimental vehicle.

4.3.1.3 Baseline methods

To validate the effectiveness of the two components in our proposed method in mitigating traffic oscillation, we compared it with different model components. We set up three baseline methods compared to the proposed control method:

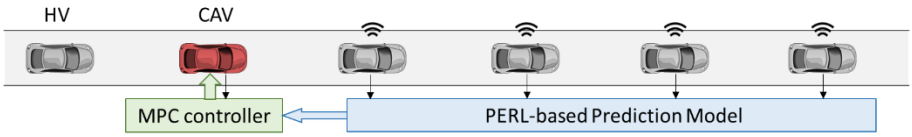
Baseline 1: No prediction and control. All the vehicles are HVs without control. This baseline is extracted from the original NGSIM dataset.

Baseline 2: MPC applied to CAV with PERL-based prediction. The following HV is not considered in the MPC control model.

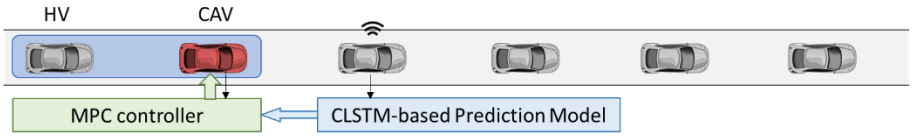
Baseline 3: MPC applied to CAV with CLSTM-based prediction using preceding vehicle information. The following HV is not considered in the MPC control model.

Proposed controller: MPC applied to the mixed platoon of CAV and HV with PERL-based prediction. The first four vehicles are called preceding vehicles with a designed trajectory from the dataset; the fifth vehicle is a CAV, and the sixth vehicle is an HV. The physics model utilized the shockwave-based Newell's car-following model, augmented with the terminal state connection process in section 3.2. For the NN model, we chose a CLSTM model, the same as the NN component in the PINN model.

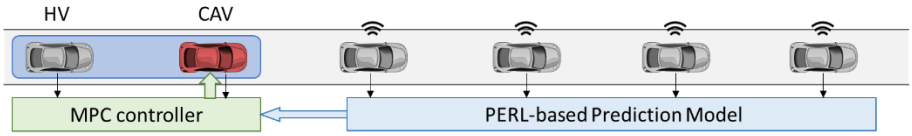
(a) Baseline 1: No prediction and control



(b) Baseline 2: MPC applied to CAV with PERL-based prediction



(c) Baseline 3: MPC applied to CAV with NN-based prediction



(d) Proposed: MPC applied to the mixed platoon of CAV and HV with PERL-based prediction

Figure 4-6 Researched scenarios.

4.3.1.4 Measurements

To comprehensively evaluate the proposed method, our measurements encompass three aspects: safety, oscillation propagation, and fuel consumption. Safety is the foundational premise of the proposed approach, while oscillation mitigation and fuel efficiency are the primary objectives of our optimization efforts.

Safety metric: This study chooses a widely employed safety evaluation metric in traffic: Time-to-Collision (TTC) to evaluate safety (Kiefer et al., 2005).

$$\mathrm{TTC_t} = \frac{\Delta d_n}{v_n - v_{n-1}} \tag{4-18}$$

with v_n and v_{n-1} denoting the instantaneous velocities of the following and leading vehicle respectively, and Δd_n representing the spacing between them. A decreased TTC value signifies a higher risk of collision, correlating to scenarios where the following vehicle is approaching the leading vehicle at a faster rate relative to their spacing.

Traffic oscillation metric: Oscillation is evaluated at the platoon level using the damping ratio, denoted asd_i, is used to evaluate the oscillation-dampening performance of a controller over a finite horizon, adapting the concept of acceleration L2 norm string stability typically assessed over an infinite horizon. This ratio is calculated as the square root of the sum of squared accelerations for vehicle n over T time steps, normalized by the square root of the sum of squared accelerations for the lead vehicle at time zero, a smaller damping ratio indicates a greater reduction in oscillation magnitude, signifying enhanced controller performance (Ploeg et al., 2014a).

$$d_i = \sqrt{\sum_{t=0}^{T} \|a_{nt}\|^2} \Big/ \sqrt{\sum_{t=0}^{T} \|a_{0t}\|^2} \tag{4-19}$$

Fuel consumption metric: The extensively applied VT-Micro model (Ahn et al., 2002) is chosen for fuel consumption evaluation. This methodology allows for modifications to the running cost function to accommodate alternative methods of estimating instantaneous fuel consumption.

$$MOE_e\left(v_{n,t}^{sim}, a_{n,t}^{sim}\right) = \begin{cases} e^{\sum_{m=0}^{3}\sum_{p=0}^{3}\left(L_{m,p}^{e} \cdot {v_{n,t}^{sim}}^{m} \cdot {a_{n,t}^{sim}}^{p}\right)}, a_{n,t}^{sim} \geq 0 \\ e^{\sum_{m=3}^{3}\sum_{p=0}^{3}\left(M_{m,p}^{e} \cdot {v_{n,t}^{sim}}^{m} \cdot {a_{n,t}^{sim}}^{p}\right)}, a_{n,t}^{sim} < 0 \end{cases} \tag{4-20}$$

$$e_n^{sim} = \left(\frac{\sum_{t\in\mathcal{T}} MOE_e\left(v_{n,t}^{sim}, a_{n,t}^{sim}\right) \times 3600}{\sum_{t\in\mathcal{T}} v_{n,t}^{sim} \times 1000}\right) \div 0.75 \tag{4-21}$$

where $MOE_e\left(v_{n,t}^{sim}, a_{n,t}^{sim}\right)$ is the instantaneous fuel consumption or emission rate (mg/s). $L_{m,p}^{e}$ and $M_{m,p}^{e}$ represent the model regression coefficients. m and p are power degrees. e_n^{sim} is the fuel consumption (L/100km).

4.3.2 Control performance and data process

After the experiments concluded, we processed the data collected from the vehicles, primarily focusing on speed and positional information for the analysis phase. The vehicle's speed and position data are sourced from two systems: one is the vehicle-mounted LiDAR positioning data, which is recorded via the CAN bus. This data is derived from the vehicle's location and tends to be unstable with significant fluctuations, affected by the vehicle's lateral movements. The other source is the On-Board Diagnostics (OBD), which measures the fuel injector's operation frequency and throttle position. From this, the Engine Control Unit (ECU) can estimate the vehicle's speed. This recorded speed is relatively stable but exhibits larger errors at lower speeds.

Figure 4-7 compares the expected trajectories with the raw trajectories recorded by OBD and LiDAR across three experiments. Trajectories 1 and 3 represent vehicle movements from south to north, while trajectory 2 depicts movement from north to south. The results demonstrate that the speed measurements from both OBD and LiDAR sources are very similar, with discrepancies primarily due to data noise and the precision limitations of the vehicles.

When comparing the actual vehicle trajectories with the expected ones, it is observed that during the acceleration and cruising phases, the vehicles closely follow the desired speeds. However, performance is slightly less effective during the deceleration phase, exhibiting considerable delays. Vehicles tend to decelerate abruptly to match the reduced target speeds only after a delay. To effectively integrate the speed data from both sources, we employ a Kalman filter for smoothing, followed by taking a weighted average of the filtered results.

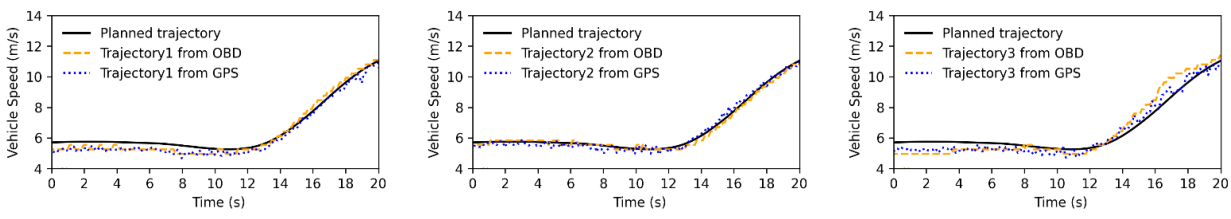


Figure 4-7 The expected trajectories with the raw trajectories recorded by OBD and LiDAR across three experiments.

4.3.3 Field Experiment Results

This section first verifies the predictive capability of the proposed PERL-based prediction model for the leading vehicles of CAVs across five selected trips. It then compares the performance of the actual vehicle trajectories following the implementation of the proposed Mixed-platoon controller with other control strategies over the same five trips.

4.3.3.1 Prediction results

In Section 4.3, four scenarios are discussed: Baseline 1 represents a purely HV environment without predictive capabilities for the preceding vehicle. Both Baseline2 and the Proposed method use the same prediction strategy, leveraging information from the four preceding vehicles fed into the PERL model for prediction. Conversely, Baseline3 utilizes data from only the leading vehicle input into a CLSTM model for prediction. TABLE 4-2 shows that the PERL-based prediction model has a significantly lower average error than the CLSTM model, demonstrating the necessity of the PERL model considering multiple preceding vehicles to capture the downstream oscillation characters.

TABLE 4-2 Preceding vehicle speed prediction RMSE (m/s) of CLSTM-based and PERL-based prediction model

Trips	CLSTM-based prediction model (applied in Baseline3)	PERL-based prediction model (applied in Baseline2, Proposed)
1	0.157	0.096
2	0.423	0.277
3	0.653	0.408
4	0.461	0.311
5	0.921	0.459
Avg	0.523	0.310

4.3.3.2 Safety results

TABLE 4-3 compares the minimum TTC during the trip under the proposed method and baseline methods. The TTC remains in a safe range (TTC>3 seconds) (Das and Maurya, 2020) under the Baseline2, Baseline3, and proposed method, indicating that all these three methods could guarantee safety. It is important to note that the goal of this research is not to increase or decrease TTC, but rather to reduce oscillation while ensuring safety. Figure 4-8, Figure 4-9, and Figure 4-10 show representative trajectories of acceleration, speed, and position for three scenarios: acceleration, cruising, and deceleration. It is evident that in all these scenarios, the CAV maintains a stable trajectory and ensures a safe distance from the preceding vehicle.

A noteworthy observation is that in case 5, an unsafe scenario occurred: TTC dropped to 2.29 seconds due to the leading vehicle's sudden deceleration and the following vehicle's delayed response, as shown in Figure 4-10. Under Baseline2, Baseline3, and the proposed method, such unsafe situations are effectively avoided, ensuring that the TTC remains above 2.5 seconds.

TABLE 4-3 Minimum TTC (s) result of three baseline methods and the proposed method.

Trips	Vehicle 0				Vehicle 1			
	Baseline1	Baseline2	Baseline3	Proposed	Baseline1	Baseline2	Baseline3	Proposed
1	13.98	8.96	9.69	9.12	7.96	9.21	8.28	8.65
2	6.75	9.74	8.42	9.76	8.54	11.37	16.23	16.34
3	9.05	72.98	141.71	87.09	11.57	71.58	147.68	88.64
4	17.54	56.31	21.32	31.26	23.43	62.01	21.81	35.33
5	2.29	5.39	4.37	5.42	6.20	7.73	10.83	7.95

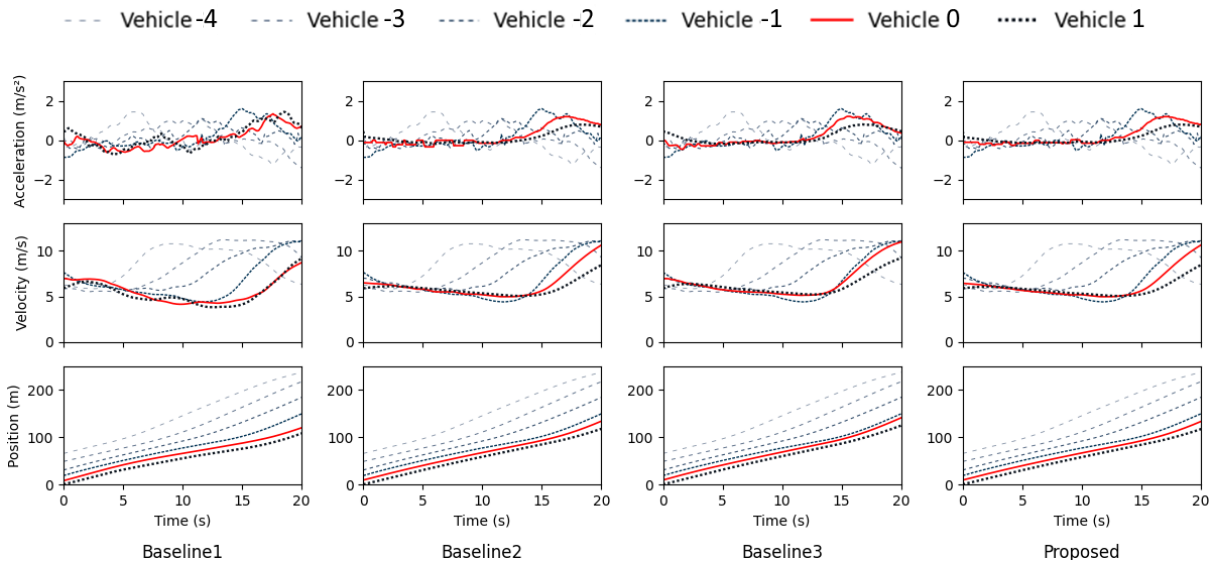


Figure 4-8 Vehicle trajectory of Trip 1 (acceleration scenario) under three baseline methods and the proposed method.

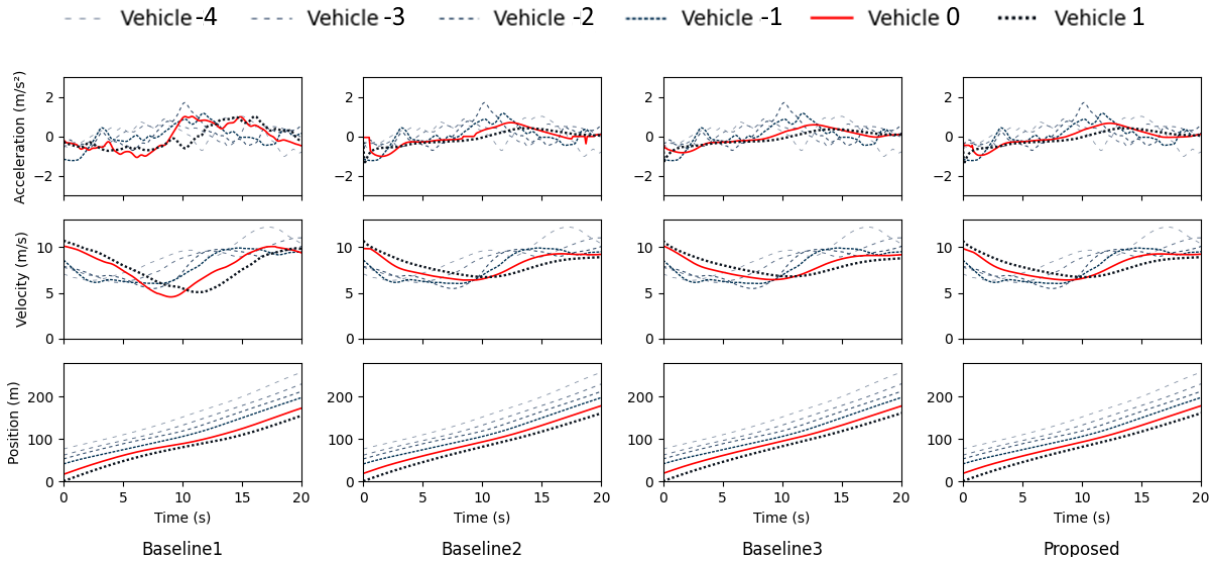


Figure 4-9 Vehicle trajectory of Trip 2 (cruising scenario) under three baseline methods and the proposed method.

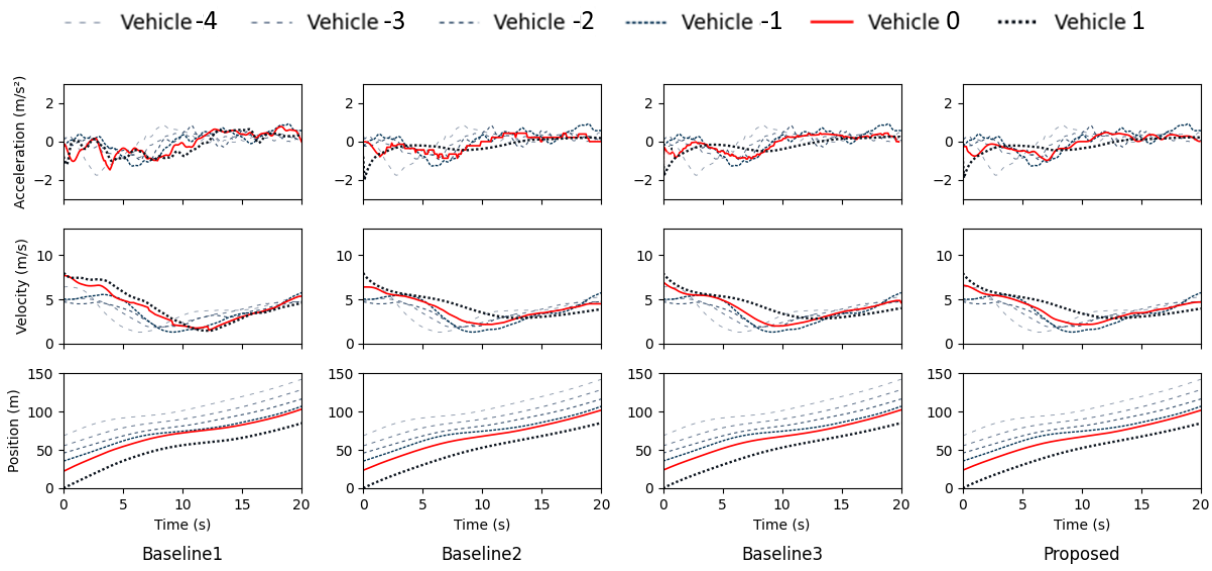


Figure 4-10 Vehicle trajectory of Trip 5 (deceleration scenario) under three baseline methods and the proposed method.

4.3.3.3 Oscillation results

TABLE 4-4 compares the damping ratios of Vehicle 0 and Vehicle 1 relative to Vehicle -1 under both proposed and baseline methods. In Baseline1, the average damping ratio for Vehicle 0 across five trajectories is 1.062, indicating that in a 100% HV scenario, the speed fluctuations of Vehicle 0 and Vehicle 1 are greater than those of Vehicle -1, leading to the amplification and backward propagation of oscillations. In contrast, the average damping ratios are reduced with the control strategies applied in Baseline 2, Baseline 3, and the proposed method. Compared to the baseline controllers, the proposed method achieves the lowest average damping ratios for Vehicles 0 and 1 for two main reasons. First, the use of downstream multi-vehicle information enables the PERL-based prediction model to make more accurate predictions of the preceding vehicle, particularly in capturing oscillation characteristics. This enhances the prediction of oscillation behaviors, facilitating the trajectory planning process of target CAV. Second, incorporating the behavior of the following vehicle into the trajectory planning optimization contributes to stable driving in mixed traffic conditions and helps prevent the propagation of oscillations.

TABLE 4-4 Damping ratio result of three baseline methods and the proposed method.

Trip	Vehicle 0				Vehicle 1			
	Baseline1	Baseline2	Baseline3	Proposed	Baseline1	Baseline2	Baseline3	Proposed
1	0.96	0.83	0.83	0.81	1.01	1.02	0.88	0.86
2	1.16	0.81	0.77	0.77	1.27	1.04	0.77	0.79
3	1.23	0.70	0.74	0.71	1.33	0.90	0.73	0.78
4	0.95	0.79	0.69	0.72	1.03	0.88	0.73	0.79
5	1.01	0.87	0.98	0.89	1.09	0.91	1.23	1.04
Avg	1.062	0.8	0.80	0.78	1.14	0.95	0.87	0.85

4.3.3.4 Fuel consumption results

TABLE 4-5 compares the fuel consumption throughout the entire trip using the proposed method and baseline methods. In four out of the five scenarios, the proposed method resulted in lower fuel consumption. The improved efficiency can be attributed primarily to the reduced damping ratios achieved under the proposed method, which signify less oscillation and, thus, smoother vehicle dynamics. In trip 5, there was an observed increase in fuel consumption. This anomaly was due to the vehicle needing to decelerate earlier than usual to preemptively address potential safety concerns. The early deceleration led to increased speed variations, resulting in greater fuel usage. This scenario highlights a crucial aspect of vehicle dynamics where safety measures, although necessary, might lead to less efficient fuel usage due to the required changes in driving patterns.

TABLE 4-5 Fuel consumption (L/100km) results of three baseline methods and the proposed method.

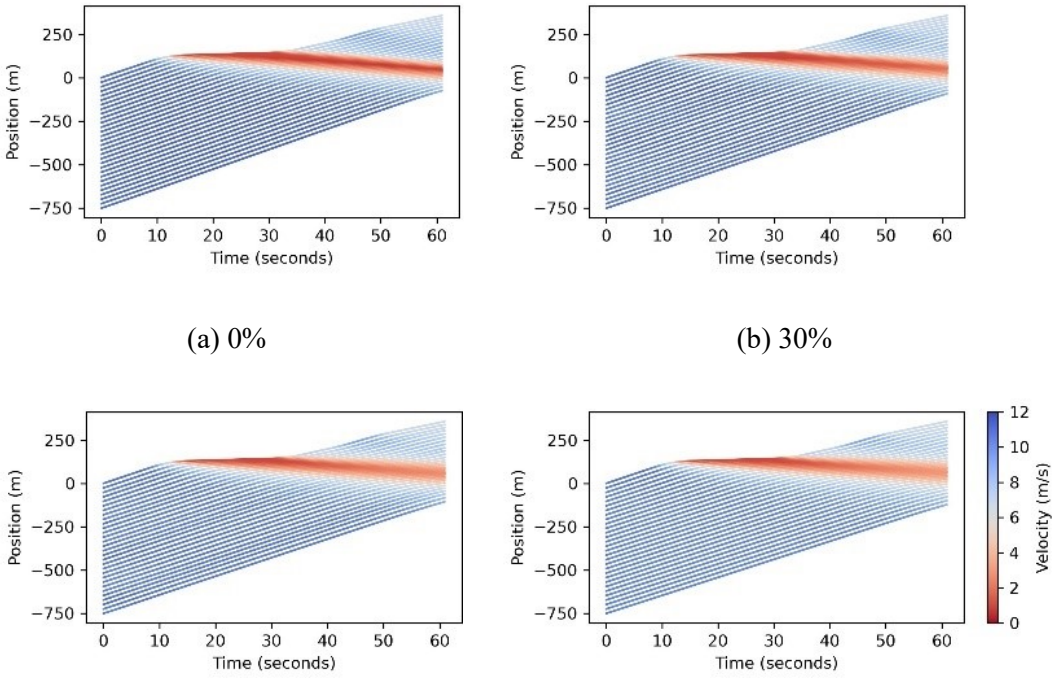
Trips	Vehicle 0				Vehicle 1			
	Baseline1	Baseline2	Baseline3	Proposed	Baseline1	Baseline2	Baseline3	Proposed
1	6.779	6.174	6.005	6.025	7.097	7.572	6.31	6.366
2	5.398	5.14	5.13	5.086	5.827	6.557	5.404	4.929
3	7.28	7.127	7.017	7.129	7.86	9.125	6.963	7.927
4	6.664	6.789	6.75	6.387	7.194	7.472	7.083	6.78
5	8.781	8.791	8.926	8.993	9.389	9.172	11.13	10.224
Avg	6.98	6.804	6.766	6.724	7.473	7.98	7.378	7.285

4.3.4 Sensitivity analysis

Besides the field test, this study utilizes simulation analysis to apply the proposed framework on a larger scale platoon. We consider a mixed platoon consisting of 30 vehicles, with the lead vehicle's trajectory derived from the NGSIM dataset. This simulation investigates the impact of penetration rates on the framework's effectiveness. The trajectories obtained from the simulation were analyzed, focusing on oscillation and fuel consumption characteristics. Simulation results indicate that with an increase in the penetration rate of proposed method-equipped CAVs, the lead vehicle's speed fluctuations dissipate more readily. Figure 4-11 compares vehicle trajectories under different CAV penetration rates, revealing that at a 0% penetration rate, shockwaves continuously propagate backward without mitigation. However, as CAV penetration increases, upstream vehicles no longer experience a complete stop, and the magnitude of speed oscillation is reduced. TABLE 4-6 contrasts the average damping ratio and fuel consumption across 30 vehicles under varying CAV penetration rates, showing that higher CAV penetration leads to lower average damping ratios—indicating smoother traffic flow and improved fuel efficiency.

TABLE 4-6 Average damping ratio and fuel consumption of the platoon of 30 vehicles.

CAV penetration rate (%)	Average damping ratio	Average fuel consumption (L/100km)
0	0.581	7.882
30	0.498	6.534
60	0.455	5.890
90	0.415	5.414



(a) 0% (b) 30%

(c) 60% (d) 90%

Figure 4-11 Trajectory of 30 vehicles using the proposed method under different CAV penetration rates.

4.4 Conclusion

This study proposes a PERL-based predictive control model for CAVs to mitigate traffic oscillation. The model introduced includes two parts: a PERL-based prediction model and an MPC-based mixed-platoon controller. The prediction model forecasts the future behavior of the preceding vehicle by

combining physical shockwave information with neural network techniques. This approach enables precise predictions of speed fluctuations, providing sufficient time for the vehicle or driver to respond effectively. For the PERL-based predictive control model, the dynamics of CAV and its following vehicles are platooned, achieving an improvement in safety and comfort for the entire platoon. In this paper, the proposed mix-platoon MPC controller is applied to a mix platoon of CAV and HV through ViL and compared with real trajectory data and three benchmark models. Experimental results validate the proposed method in damping traffic oscillation and enhancing the safety and fuel efficiency of the CAV and the following HV in mixed traffic.

5 PERPL-BASED VEHICLE CONTROL

proposes the Physics-Enhanced Residual Policy Learning (PERPL) framework for vehicle control, leveraging the advantages of both physics-based models (data-efficient and interpretable) and RL methods (flexible to multiple objectives and fast computing). The physics component provides model interpretability and stability and the learning-based Residual Policy adjusts the physics-based policy to adapt to the changing environment, thereby refining the decisions of the physics model. This model is applied in the decentralized control of a mixed traffic platoon of CAVs and HVs using a constant time gap (CTG) strategy, with actuator lag and communication delays. Experimental results demonstrate that this model has high extrapolation ability, achieving smaller headway errors and better oscillation dampening than the linear control model and reinforcement learning (RL) model in artificial extreme scenarios. At the macroscopic level, overall traffic oscillations are also reduced as the penetration rate of CAVs employing the PERPL-based controller increases (Long et al., 2024d).

This Section is organized as follows. Section 5.1 outlines the investigated problem. Section 5.2 introduces the proposed PERPL controller. Section 5.3 presents the experiments conducted to compare the proposed model with baseline models. Finally, Section 5.4 provides the conclusion and discusses directions for future research.

5.1 Problem Statement

5.1.1 Environment Setting

This research focuses on the longitudinal control of CAVs in mixed traffic of CAVs and HVs. We consider the car following process without lateral movement on the highway. We denote $\mathcal{N}$ as the set of vehicle index, $\mathcal{N} = \{\mathcal{N}^{\mathrm{CAV}}, \mathcal{N}^{\mathrm{HV}}\}$, where $\mathcal{N}^{\mathrm{CAV}}$ is the set of CAVs and $\mathcal{N}^{\mathrm{HV}}$ is the set of HVs. CAVs broadcast their state information (e.g., speed, position) to other vehicles in the platoon via Vehicle-to-Vehicle (V2V) communications, subject to a constant communication delay. They can also access real-time data about their own state. HVs lack autonomous driving capability and do not receive digital information from other vehicles, but HVs could broadcast information to CAVs within communication range.

This research considers a communication delay τ^{c} during receiving preceding vehicle data, which includes both the signal propagation time and the processing time once the data is received. Moreover, when executing decisions derived from the PERPL framework, the vehicles experience actuator delay τ^{A}, which refers to the time lag between the issuance of commands by the control system and the actual response by the vehicle's actuators. The related notations are defined in TABLE 5-1.

5.1.2 Distributed Platoon Control Scheme

In this section, we proposed a distributed CAV longitudinal control for distributed platooning, whose framework is presented in Figure 5-1. Any CAV, $n \in \mathcal{N}^{\mathrm{CAV}}$, can obtain its own state at time t: $\boldsymbol{s}_{nt}, n \in \mathcal{N}^{\mathrm{CAV}}, t \in \mathcal{T}$, where $\mathcal{T}$ denotes the set of all time stamps. each CAV can access the states of up to three preceding vehicles within the communication range. The states of these vehicles are represented as $\left\{\boldsymbol{s}_{(n-k)(t-\tau^{\mathrm{c}})}\right\}_{k\in[1,2,3]}, n \in \mathcal{N}^{\mathrm{CAV}}$, where τ^{c} is the communication time delay. If fewer than three vehicles precede any CAV or if there are no preceding vehicles at all, the number of vehicles considered adjusts accordingly to zero, one, or two based on availability. This approach ensures that each vehicle can dynamically adjust its behavior based on the immediate traffic conditions, enhancing both the responsiveness and safety of the platoon.

Vehicle state $\boldsymbol{s}_{nt}$ including three parts: position error Δd_{nt}, speed difference Δv_{nt} and acceleration a_{nt}: $\boldsymbol{s}_{nt} = [\Delta d_{nt}, \Delta v_{nt}, a_{nt}]^{T}, n \in \mathcal{N}, t \in \mathcal{T}$. Definitions for these components are as follows:

CAVs apply a constant time headway (CTH) spacing strategy, which follows its preceding vehicle with a desired spacing distance and ensures safety. Thus, the desired spacing distance of vehicle n at time t is $h^{d}v_{nt} + d_{0}$, where h^{d} and d_{0} are the desired constant headway and standstill space, respectively. Based on the CTH rule, the position error of Vehicle n with respect to a desired distance from the preceding vehicle $(n-1)$ was denoted by Δd_{nt}:

$$\Delta d_{nt} = d_{(n-1)(t-\tau^{\mathrm{c}})} - d_{nt} - d_{0} - h^{d}v_{n}, n \in \mathcal{N}, t \in \mathcal{T} \tag{5-1}$$

The speed difference between the ego and the preceding vehicle is:

$$\Delta v_{nt} = v_{(n-1)(t-\tau^{\mathrm{c}})} - v_{nt}, n \in \mathcal{N}, t \in \mathcal{T} \tag{5-2}$$

The control variable is u_{nt}. Given the assumptions and communication environment, the vehicle dynamics are modeled by linearized dynamics with the consideration of time delay τ^A:

$$\dot{a_{nt}} = \frac{u_{nt} - a_{nt}}{\tau^A}, n \in \mathcal{N}, t \in \mathcal{T} \tag{5-3}$$

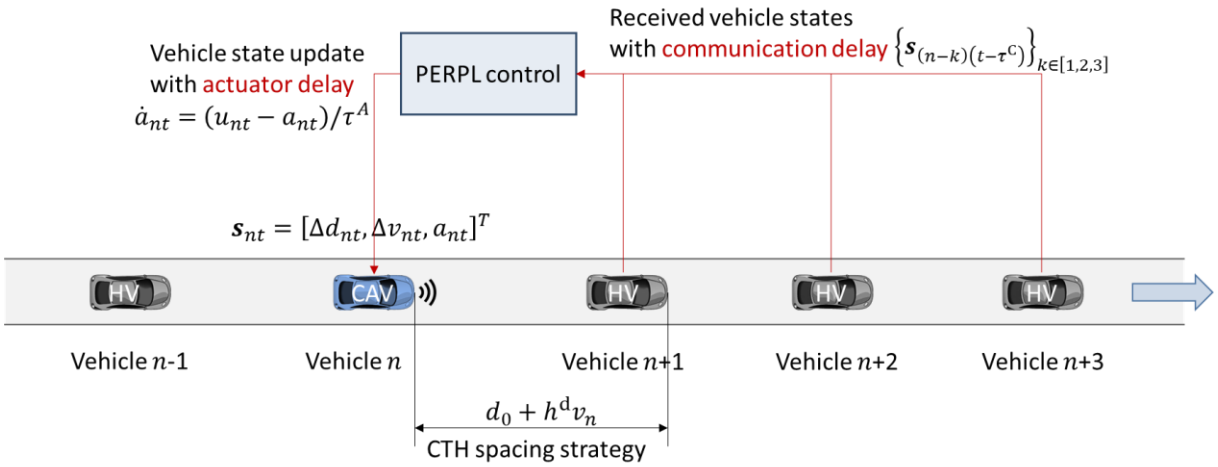


Figure 5-1 Distributed control scheme for vehicular platoon.

TABLE 5-1 Key notation

Notation	Description
$\mathcal{N}$	Set of vehicle trajectories
n	Index of trajectory. $n \in \mathcal{N}$
$\mathcal{T}$	Set of all time stamps
t	Index of time $t \in \mathcal{T}$
d_{nt}	Longitudinal position of Vehicle n at time t (m)
v_{nt}	speed of Vehicle n at time t (m/s)
a_{nt}	Acceleration of Vehicle n at time t (m/s^2)
h^d	Desired headway (s)
d_0	desired space at a standstill (m)

Δd_{nt}	Distance error of vehicle n under CTH spacing strategy at time t (m)
Δv_{nt}	speed difference of Vehicle n at time t (m/s) $\Delta v \coloneqq v_{n-1} - v_n$
$\boldsymbol{s}_{nt}$	state of Vehicle n at time t, $\boldsymbol{s}_{nt} \coloneqq [\Delta d_{nt}, \Delta v_{nt}, a_{nt}]^T$
τ^{A}	Actuator time delay (s)
τ^{C}	Communication time delay (s)
$\boldsymbol{\mathcal{A}}$	Feasible set of a_{nt}.
$\boldsymbol{a}_{nt}^{\mathrm{phy}}$	Action given by physics model-based action policy
$\boldsymbol{a}_{nt}^{\mathrm{RL}}$	Action given by RL-based residual action policy
f^{SAG}	Physics-based safety action barrier
d_n	Damping ratio of vehicle n

5.2 PERPL controller

5.2.1 Framework

Our proposed control framework encompasses two parallel controllers: a linear controller that focuses on ensuring local and string stability using a non-linear programming formulation and a DRL controller that specifically targets handling traffic disturbances and time delays. By integrating these two controllers, our framework aims to effectively address the challenges associated with car-following control in CAV environments. A safety barrier is added to guarantee safety. As shown in Figure 5-2, the control action $\boldsymbol{a}_{nt}^{\mathrm{PERPL}}$ from PERPL framework is:

$$\boldsymbol{a}_{nt}^{\mathrm{PERPL}} = f^{\mathrm{SG}}\left(\boldsymbol{a}_{nt}^{\mathrm{phy}} + \boldsymbol{a}_{nt}^{\mathrm{RL}}\right) \tag{5-4}$$

where $\boldsymbol{a}_{nt}^{\mathrm{phy}}$ is the output action of physics-based policy, $\boldsymbol{a}_{nt}^{\mathrm{RL}}$ is the output action of residual policy. f^{SG} is the safety barrier that projects the combined output $\boldsymbol{a}_{nt}^{\mathrm{phy}} + \boldsymbol{a}_{nt}^{\mathrm{RL}}$ to a safety range to guarantee safety (Ames et al., 2017). The details of each component are given in the following subsections.

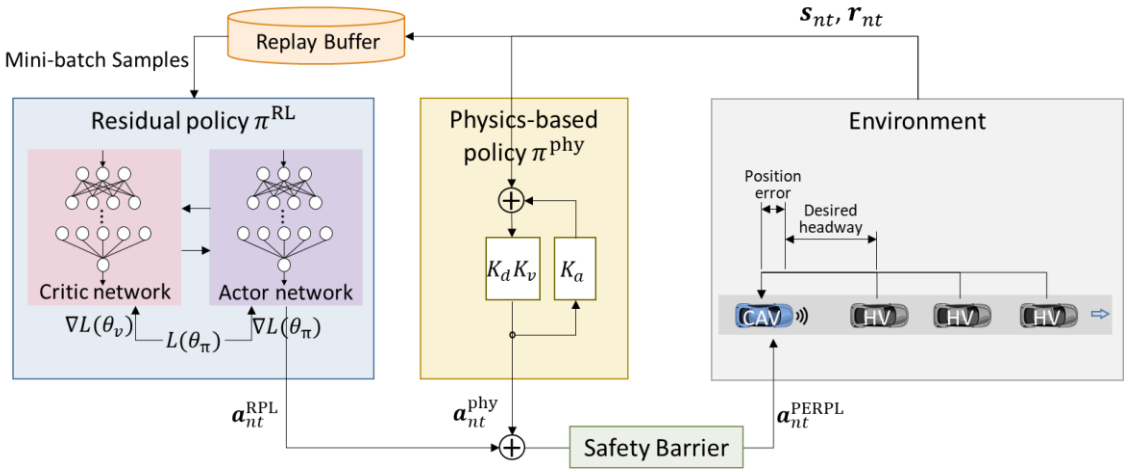


Figure 5-2 PERPL Framework.

5.2.2 Model-based Policy

A linear control model is employed to manage the behavior of a following vehicle within a platoon, emphasizing maintaining constant headway and minimizing the deviation from the lead vehicle's trajectory. The model is based on principles of classical control theory. The control input from the linear controller is:

$$\boldsymbol{a}_{nt}^{\text{phy}} = K \cdot \boldsymbol{s}_{nt} \tag{5-5}$$

where $K = [K_d, K_v, 0]$ is a vector of feedback gained forming a closed loop of the controller, where K_d, K_v are the feedback coefficients for Δd_{nt} and Δv_{nt}.

5.2.3 Residual Policy

The residual action policy is constructed using the Proximal Policy Optimization (PPO) method (Schulman et al., 2017). Unlike other reinforcement learning methods that use the sum of discounted rewards to estimate future returns, the PPO method employs a policy gradient method that focuses on optimizing an objective function. This function is not solely based on the sum of returns but also involves the concept of probability ratios of policies and advantage functions, which facilitates learning and exploration on top of the existing policy. Therefore, PPO attempts to maintain training stability by limiting the extent of policy updates, thereby preventing excessively large updates during training.

The PPO is based on the Actor-Critic structure, with an actor (policy) and critic (value) network. This dual network structure facilitates efficient exploration of the action space while stabilizing the learning updates through the critic's value estimates.

5.2.3.1 Actor Network

The Actor network is responsible for defining the policy π with parameter θ. It takes the DRL state as input and outputs a probability distribution over actions. The control signal $\boldsymbol{a}$ is then sampled from this distribution. The Actor network is updated by maximizing the objective function $L(\boldsymbol{s}, \boldsymbol{a}, \theta_t, \theta)$, which is defined as:

$$L(\boldsymbol{s}, \boldsymbol{a}, \theta_t, \theta) = \min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t\right) \tag{5-6}$$

where $r_t(\theta)$ is the probability ratio $r_t(\theta) = \pi_\theta(a|s)/\pi_{\theta_t}(a|s)$, clip$(\cdot)$ is a clipping function to remove incentives for the new policy to get far from the old policy, which prevents large updates that could destabilize the training process. ϵ is a small hyperparameter, which determines how much the ratio can differ from 1 before it is clipped. $\hat{A}_t$ is the advantage estimate at time t, calculated by:

$$\hat{A}_t = R_t - V(s_t) \tag{5-7}$$

where $V(s_t)$ is the value estimated by the Critic network; R_t denotes the discounted sum of rewards over T steps at state s_t:

$$R_t = r_t + \gamma r_{t+1} + \cdots + \gamma^{T-t+1} r_{T-1} + V^{T-t}(s_T) \tag{5-8}$$

Therefore, the parameter θ of the Actor network is updated based on the gradient of $L(\boldsymbol{s}, \boldsymbol{a}, \theta_t, \theta)$ with a learning rate α^θ. This update method effectively balances policy improvement with training stability, using a clipping mechanism to prevent excessively aggressive policy updates in certain situations, which could otherwise lead to performance deterioration.

$$\theta = \theta - \alpha^\theta \cdot \nabla L(\boldsymbol{s}, \boldsymbol{a}, \theta_t, \theta) \tag{5-9}$$

$$\theta_{t+1} = \arg\max_\theta \mathbb{E}_{s,a\sim\pi_{\theta_t}}[L(\boldsymbol{s}, \boldsymbol{a}, \theta_t, \theta)] \tag{5-10}$$

5.2.3.2 Critic Network

The Critic network evaluates the decision output by the Actor network. The Critic network receives the DRL state $\boldsymbol{s}$ as input and outputs the estimated state value $V(s_t)$. The network structure also includes one hidden layer with 100 neurons, and the ReLU function is used as the activation function for the output. The Critic network is updated by minimizing the critic loss function:

$$L_c(\Phi) = \hat{E}_t[(V(s_t) - R_t)^2] \tag{5-11}$$

The parameter Φ is iteratively optimized based on the gradient $L_c(\Phi)$with learning rate α^{Φ}:

$$\Phi = \Phi - \alpha^{\Phi} \cdot \nabla L_c(\Phi) \tag{5-12}$$

The residual action policy pseudocode is shown in TABLE 5-2.

TABLE 5-2 Pseudocode

Algorithm 1: Training process of Residual action policy algorithm for Longitudinal cruising control

Inputs: state information, action information

Initialize agent network with critic and actor networks, environment, actor policy μ: S → R^m+1 and σ:

S → diag(σ1, σ2, ..., σm+1)

for iteration=1,2,… **do**

for actor=1,2,…N **do**

Run policy π_{θ_t} in the environment for T timesteps and collect (s_t, a_t, r_t)

Compute advantages estimates $\hat{A}_1, \ldots, \hat{A}_T$ with discount along the time

end for

Optimize surrogate objective L with respect to θ, with K epochs and minibatch size $M \leq NT$

Update critic $\theta_t \leftarrow \theta_{t+1}$

end for

5.2.4 Physics-based Safety Barrier

The safety barrier projects the action into the safety region based on the safety requirement on headway. The safety barrier adjusts the control action by taking the combined output of the physics-based and RL policies, $\boldsymbol{a}_{nt}^{\mathrm{phy}} + \boldsymbol{a}_{nt}^{\mathrm{RPL}}$, which may not inherently satisfy safety requirements. It then computes the adjusted action $\boldsymbol{a}_{nt}^{\mathrm{PERPL}}$ that minimizes the deviation from this combined action while ensuring it adheres to safety constraints, formalized as:

$$\boldsymbol{a}_{nt}^{\mathrm{PERPL}} = f^{\mathrm{SG}}\left(\boldsymbol{a}_{nt}^{\mathrm{phy}} + \boldsymbol{a}_{nt}^{\mathrm{RPL}}\right) = \arg\min_{\boldsymbol{a}_{nt} \in \boldsymbol{\mathcal{A}}} \left\| \boldsymbol{a}_{nt} - \left(\boldsymbol{a}_{nt}^{\mathrm{phy}} + \boldsymbol{a}_{nt}^{\mathrm{RPL}}\right) \right\|^2 \tag{5-13}$$

subject to the constraint ensures that the resulting state transition remains within a safe state $\boldsymbol{s}_{n(t+1)}^{\mathrm{safe}}$:

$$A\boldsymbol{s}_{nt} + B\boldsymbol{a}_{nt} + C\mathbf{w}_{nt} \in \boldsymbol{s}_{n(t+1)}^{\mathrm{safe}} \tag{5-14}$$

where $\boldsymbol{s}_{n(t+1)}^{\mathrm{safe}}$ is the set of safety states where the headway between Vehicle n and Vehicle $(n-1)$ is kept within safe limits, typically between 1s and 3s. The lower limit is to guarantee safety (Vogel, 2003), and the upper limit is to avoid overlarge headway causing unstable inside the platoon. $\boldsymbol{\mathcal{A}}$ is the feasible set of $\boldsymbol{a}_{nt}$.

Importantly, the safety barrier is not activated under normal safe control scenarios and does not compromise the control effectiveness of the model. While the safety barrier is not inherently a part of the PERPL framework, it is included for two main reasons: firstly, safety barriers are a common mechanism currently implemented in RL-based models in practical applications. Secondly, it helps account for physical constraints in real-world scenarios. In reality, a collision will occur if the following vehicle is too close to the leading vehicle. However, in simulation environments, the following vehicle may even pass the leading vehicle without any constraint. The safety barrier has been implemented to prevent such unrealistic scenarios. Although actual accidents might be mitigated, the frequency of activation of the safety barrier still serves as an important safety metric.

5.3 Experiments

5.3.1 Experiment settings

This section details the design philosophy of the experiment, evaluation metrics, simulation methods for HVs in mixed traffic, and baseline models.

5.3.1.1 Experiment Overview

Our analysis spans two dimensions: 1) **Single Vehicle Cruising**: We conducted training and testing using both artificially designed and real-world leading vehicle trajectories. The artificially designed trajectories were designed to create extreme danger scenarios (rapid acceleration and deceleration) to evaluate the model's generalization capabilities. Real-world trajectories were used to align more closely with practical conditions. 2) **Mixed Traffic Platooning**: We assessed the performance of multiple vehicles and conducted a macro-level analysis.

Communication delay τ_n^C was set at 0.3 seconds, based on empirical results, which lends a reasonable degree of reliability (Liang et al., 2024). The safety headway interval is defined between 1 and 3 seconds. The lower limit of headway prevents collisions, while the upper limit ensures vehicles do not lag too far behind the lead vehicle, thereby affecting subsequent vehicles. The safety barrier utilizes this range; if a pending action is detected that would cause the headway to exceed these bounds, the safety barrier projects the action to bring the headway back within a safe range. The desired constant headway t^d and the safety headway set can be adjusted based on practical applications. Other parameters are listed in TABLE 5-3.

TABLE 5-3 Experiment parameters.

	Parameters	Value
Experiment parameters	Vehicle length	4 m
	Update interval Δt	0.1s
	Desired constant headway h^d	2s
	The range of the safety headway	[1,3] s
	Desired standstill space d_0	4 m
	Actuator delay τ^A	0.2 s
	Communication delay τ^C	0.3 s
HV modeling parameters	Desired velocity V_0	20.3 m/s
	Safe time headway T	1.2 s
	Maximum acceleration a	1.9 m/s²
	Comfortable Deceleration b	3.9 m/s²
	Acceleration exponent σ	4
	Minimum distance S_0	2.0 m

5.3.1.2 Evaluation metrics

To systematically evaluate the proposed framework, three performance indicators to quantitatively assess the control performance: driving comfort.

Headway Error: To assess the stability and safety of vehicle operation, it is crucial that the headway—the distance between vehicles—remains consistent at the designated desired headway. The headway error reflects the vehicle's adherence to the Constant Time Gap (CTG) rule as well as its safety characteristics. For this purpose, we employ the headway's Root Mean Square Error (RMSE) to quantify deviations from the desired values.

$$\mathrm{RMSE}^h = \sqrt{\sum_{t\in\mathcal{T}} \left(\frac{d_{(n-1)t} - d_{nt} - d_0}{v_{nt}} - h^d\right)^2 / |\mathcal{T}|} \tag{5-15}$$

Damping Ratio: The cumulative damping ratio (*di*) is a measure of the CAV controller's ability to dampen traffic oscillations, which quantifies the empirical string stability (Ploeg et al., 2014b). When the traffic passes through a string-stable CAV, the magnitude of traffic oscillations is either reduced or remains unchanged. The l_2-norm acceleration damping ratio d_n can be formulated as follows:

$$d_n = \frac{\|a_n\|_2}{\|a_0\|_2} = \left(\frac{\sum_{t\in\mathcal{T}}|a_{nt}|^2}{\sum_{t\in\mathcal{T}}|a_{0t}|^2}\right)^{1/2} \tag{5-16}$$

Comfort level: the driving cost function c_{nt} aims to evaluate the eco-driving performance and empirical string stability, which is defined as:

$$c_{nt} = \alpha_i (a_{nt})^2 \tag{5-17}$$

5.3.1.3 HV Modeling Method

To ensure the simulation reflects realistic mixed traffic conditions, this study employs the Intelligent Driver Model (IDM), which is calibrated to depict the string instability characteristic often observed in HVs. This approach allows for a more convincing representation of HV behaviors within our experiments with parameters listed in TABLE 5-3.

5.3.1.4 Baseline models

To analyze the performance and influence of the PERPL framework in a mixed traffic environment, the Linear control and RL models are employed as a baseline for comparison. The Linear control model, similar to a model-based action policy, is calibrated on the training set with parameters set at $K_d = 0.62$ and $K_v = 0.37$. The RL (PPO) model shares a similar structural setup with the residual action policy used in the PERPL framework. These comparisons highlight the distinct advantages of integrating physical principles with RL techniques in managing complex traffic dynamics. Other training settings are listed in TABLE 5-4.

TABLE 5-4 Hyper Parameters of the PPO training

Parameter	Value
Initial learning rate	2e-4
Discount factor γ	0.9
gae_λ	0.95
Update epochs	10
Clip ratio ϵ	0.2
Learning rate for value function optimizer	0.5
max_grad_norm	0.5

5.3.2 Results of Single vehicle cruising

In this section, we compared three different control approaches for a single vehicle following. This setup helped us evaluate the accuracy and generalization capabilities of the PERPL model, compared to two baseline models: RL (PPO), and PERPL (Linear+PPO).

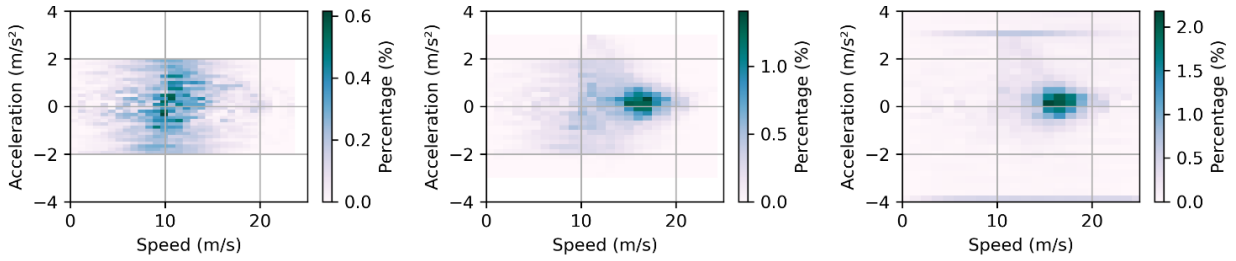
5.3.2.1 Data

In the single-vehicle cruising scenario, the data comprises a mix of real-world trajectories from NGSIM (NGSIM, 2007) and a subset of artificially designed trajectories. Trajectories are divided into three sets: training set, test set, and extrapolation set, with each set containing 100 trajectories, containing 500 timesteps with a $0.1s$ time interval. Each controller is trained on each trajectory for 3000 episodes.

Training set and Test set: These are derived from NGSIM trajectories, with acceleration magnitudes limited to within $\pm 3\ m/s^2$, as shown in Figure 5-3 (a) (b). The datasets are randomly split into training and testing subsets.

Extrapolation set: This set includes trajectories from NGSIM, modified to incorporate extreme acceleration and deceleration events beyond $\pm 3\ m/s^2$. This arrangement allows for a comparative assessment of the generalization capabilities of various methods beyond their training datasets. Notice that

the notably darker lines at $a = -4m/s^2$ and $a = 3m/s^2$, as shown in Figure 5-3 (c). These were intentionally set to simulate extreme scenarios by artificially enhancing the acceleration and deceleration behaviors based on real driving trajectories. This approach is used to test the model's stability under challenging conditions.



(a) Training Set (b) Test Set (c) Extrapolation Set

Figure 5-3 Distribution of acceleration and speed in three preceding vehicle trajectory sets.

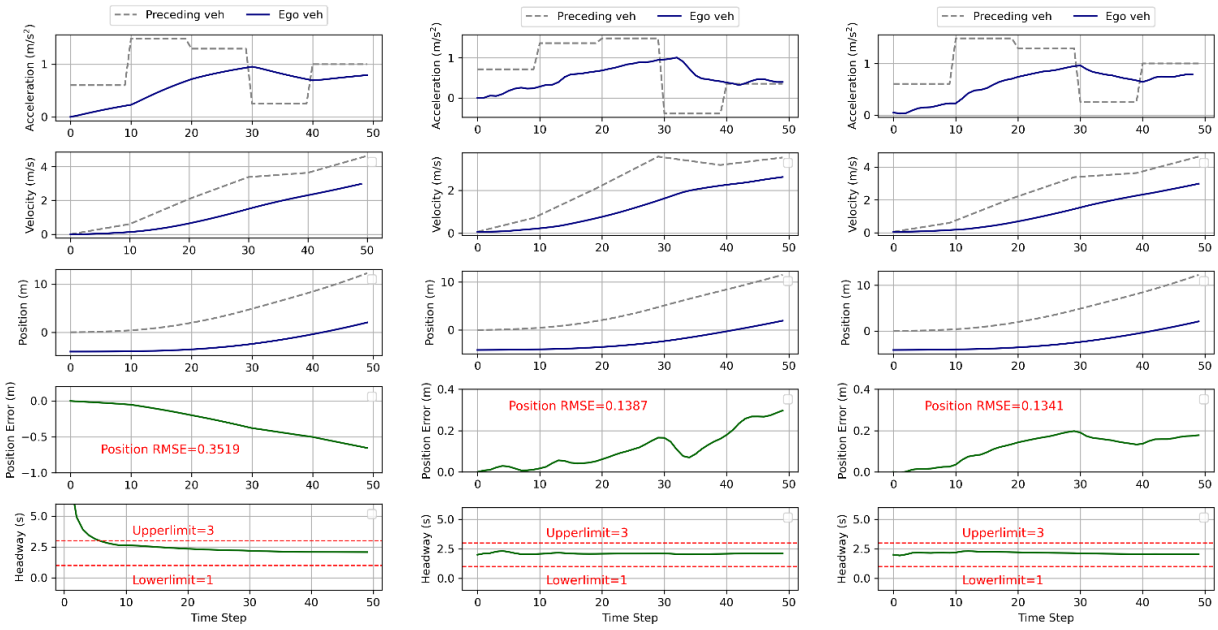
5.3.2.2 Control Performance Evaluation

As shown in TABLE 5-5, both RL and PERPL achieved lower headway errors than the Linear controller in both training and test sets. In the extrapolation set, PERPL significantly outperformed the other methods, with headway errors much smaller than those of RL, which were six times higher than those of PERPL. An analysis of the Safety Barrier activation revealed that both PERPL and the Linear controller did not activate the Safety Barrier in any of the scenarios, demonstrating their robustness and consistent achievement of constant headway across all domains. However, the RL approach showed instability in the extrapolation set, with the Safety Barrier being activated 7.92% of the time, indicating that the RL model's control outputs occasionally pushed the state beyond safe limits. Figure 5-5 illustrates the vehicle following behavior under the three control models for a trajectory from the extrapolation set. Around the 130s mark, when the lead vehicle abruptly decelerated to -2.7, both Linear and PERPL managed to prompt rapid deceleration in the following vehicle, reaching about -3, whereas the RL controller only achieved a deceleration of around -1.7. This lesser response gradually increased headway error until the safety

headway limit of 1s was breached around the 170s mark, triggering the Safety Barrier. This forced the vehicle to adopt an extremely sharp deceleration of -12.1 m/s² to exit the unsafe condition, preventing a potential collision and resulting in a greater average Headway RMSE.

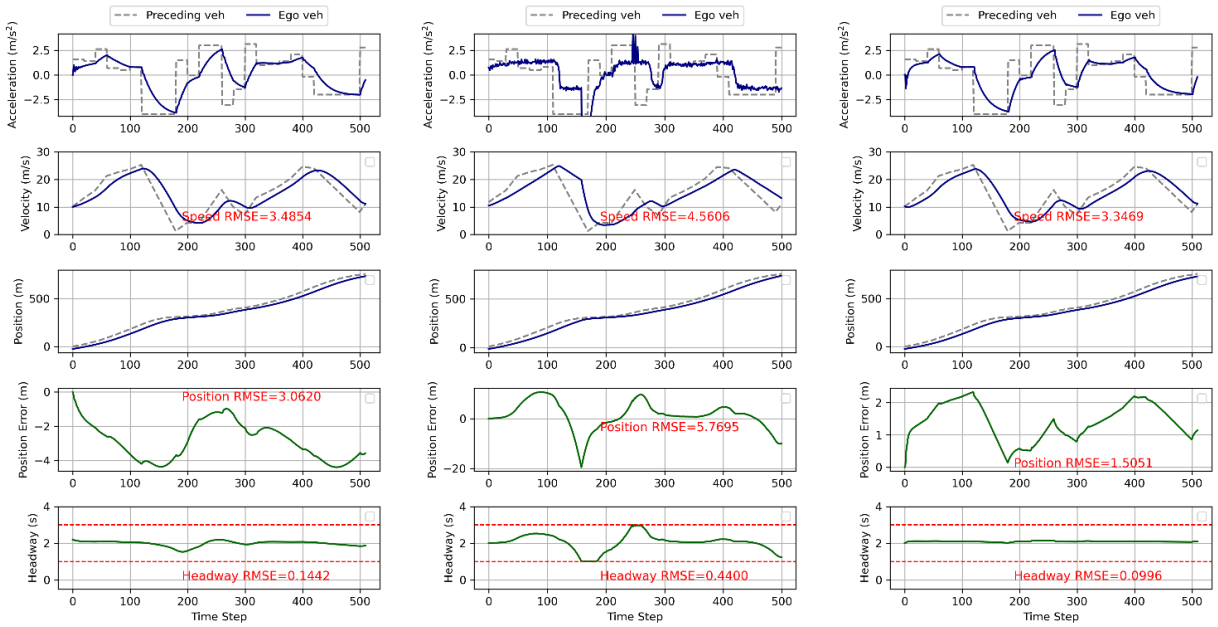
TABLE 5-5 Single-vehicle following performance artificially designed preceding trajectory.

	Average Headway RMSE RMSE^h (s)			Proportion of time Safety barrier is activated (%)		
	Training set	Test set	Extrapolation set	Training set	Test set	Extrapolation set
Linear	0.326	0.372	1.726	0	0	0
RL (PPO)	0.169	0.172	2.429	0	0	7.92
PERPL (Linear+PPO)	0.098	0.149	0.419	0	0	0



(a) Linear (b) RL (PPO) (c) PERPL (Linear+PPO)

Figure 5-4 Single vehicle following result of one example from the test set.

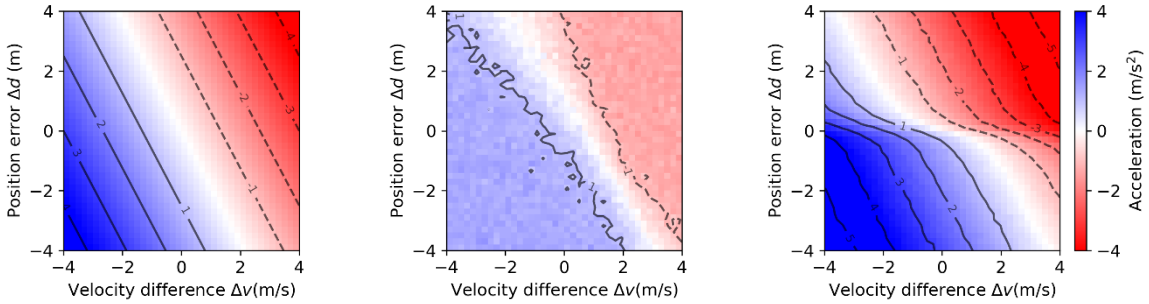


(a) Linear (b) RL (PPO) (c) PERPL (Linear+PPO)

Figure 5-5 Single vehicle following result of one example from the extrapolation set.

To provide a clearer comparison of the behavior of the three actors in a wider range of scenarios, Figure 5-6 displays the action outcomes for each method under assumed conditions of zero acceleration with position error $\Delta \boldsymbol{d_{nt}}$ ranging from -5 to 5 and speed difference $\Delta \boldsymbol{v_{nt}}$ also ranging from -5 to 5. As depicted in Figure 5-6 (a), the Linear model's acceleration responses are proportional to changes in Δd_nt and Δv_nt, indicating predictable behavior even under extreme conditions. Figure 5-6 (b) shows that the RL model's responses are not linear; it behaves similarly to the Linear model when $\Delta \boldsymbol{d_{nt}}$ and $\Delta \boldsymbol{v_{nt}}$ are small but adopts minimal absolute acceleration values under small $\Delta \boldsymbol{d_{nt}}$ and $\Delta \boldsymbol{v_{nt}}$, indicating a lack of aggressive response in critical situations. Conversely, the PERPL model in Figure 5-6 (c) exhibits behavior similar to the Linear model when Δd_{nt} and Δv_{nt} are within [-1,1] and adopts accelerations greater than 4 or decelerations less than -4 when the absolute values of Δd_{nt} and Δv_{nt} exceed 3, demonstrating its ability to respond assertively under extreme conditions.

Consider an extreme scenario where $\Delta d_{nt} = 5$ and $\Delta v_{nt} = 5$, indicating a situation where, if the lead vehicle's speed is $10 m/s$ and the following vehicle's speed is $15 m/s$, to maintain a 2s headway, the required distance would be 30 m. However, the actual distance is only 27 m, resulting in a headway of approximately 1.53 s and a Time-to-Collision (TTC) of 5.4 s, clearly necessitating rapid deceleration. Here, the RL model's deceleration rate is $-2.1\ m/s^2$, whereas the Linear model and PERPL achieve deceleration rates of approximately 4 m/s^2 and 5.3 m/s², respectively, highlighting the enhanced responsiveness of the PERPL model under critical conditions.



(a) Linear (b) RL (PPO) (c) PERPL (Linear+PPO)

Figure 5-6 Comparison of control policy trained on the training set.

To further clarify the differences in policy between RL and PERPL, Figure 5-7 shows how the control policies of these two models evolved during training. The training process of PPO is less stable than PERPL. In the first 3000 epochs, PPO incurred a higher loss (i.e., lower reward) compared to PERPL. From 3000 to 6000 epochs, after changing the leading vehicle data, PPO's performance significantly deteriorated relative to PERPL on the new data.

Figure 5-8 further illustrates the control policies learned by PPO and PERPL across different epochs. PERPL controller starts with a linear pattern and gradually learns a non-linear policy. During training with various trajectories, only the results in the second quadrant ($\Delta v > 0\ and\ \Delta d < 0$) and the

fourth quadrant ($\Delta v < 0\ and\ \Delta d > 0$) change slightly. Meanwhile, results in the first quadrant ($\Delta v > 0\ and\ \Delta d > 0$) and the third quadrant ($\Delta v < 0\ and\ \Delta d < 0$) remain largely consistent.

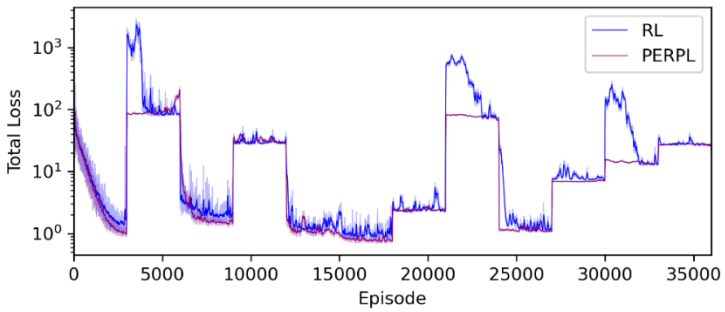
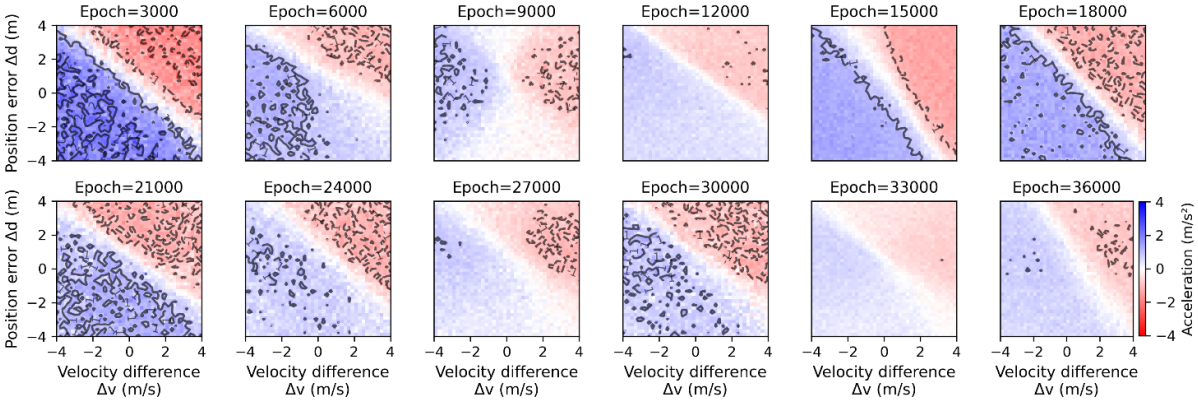
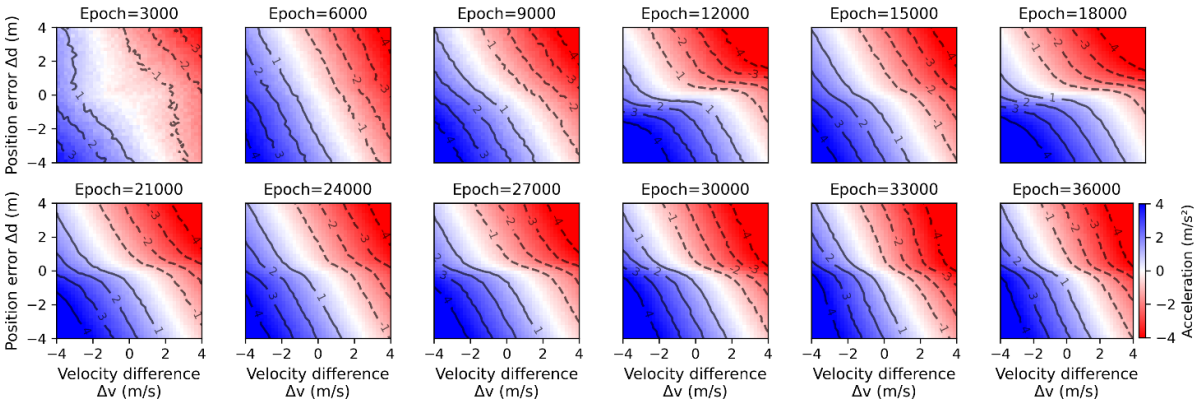


Figure 5-7 Total loss (Negative of rewards) during training.



(a) RL (PPO)



(b) PERPL (Linear+PPO)

Figure 5-8. Control policy of RL and PERPL model during training process

5.3.3 Results of Mixed-traffic platooning

5.3.3.1 Data

Experiments utilizing real-world trajectory datasets are conducted to evaluate the Distributed Reinforcement Learning (DRL)-based control strategy. We constructed a platoon consisting of ten vehicles, labeled from upstream to downstream as 0, 2, 3, 4, 6, 7, and 9 as HVs and the remaining as CAVs. The trajectories for vehicle 0 on I-80 from 4:00 p.m. to 4:15 p.m., a period noted for frequent traffic oscillations, were selected for these experiments. To ensure consistency across trials, each vehicle in the experiment starts from an initial equilibrium state.

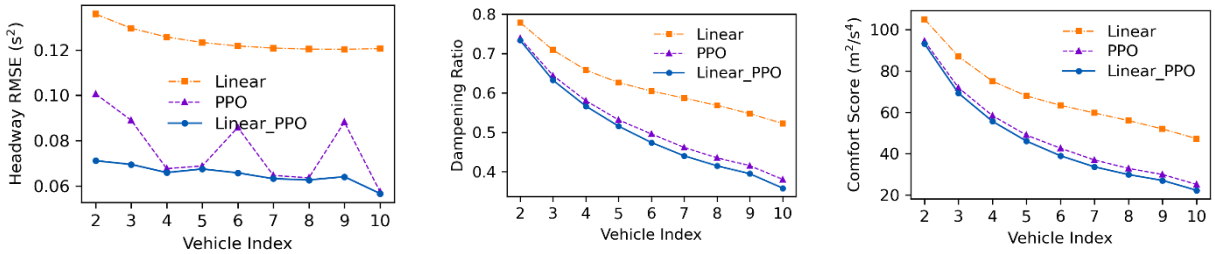
5.3.3.2 Mixed platoon control performance

Results illustrated in TABLE 5-6 and Figure 5-9 of the section reveal that the PERPL model outperformed the other models in terms of headway maintenance, with a lower RMSE, suggesting better precision in following distances. Additionally, the PERPL model showed improved damping ratios and comfort scores, indicating enhanced overall platoon stability and passenger comfort compared to the Linear and standalone RL models. These findings underscore the potential of integrating physics-based control with RL techniques to enhance automated driving systems in complex traffic environments.

As an illustrative example, Figure 5-10 shows the ten-vehicle platoon trajectories of the field data and simulated results using the proposed PERPL and baseline models. It can be seen that under linear control, the following vehicles are most affected by the lead vehicle's stop, with nearly all following vehicles coming to a stop after the lead vehicle stops.

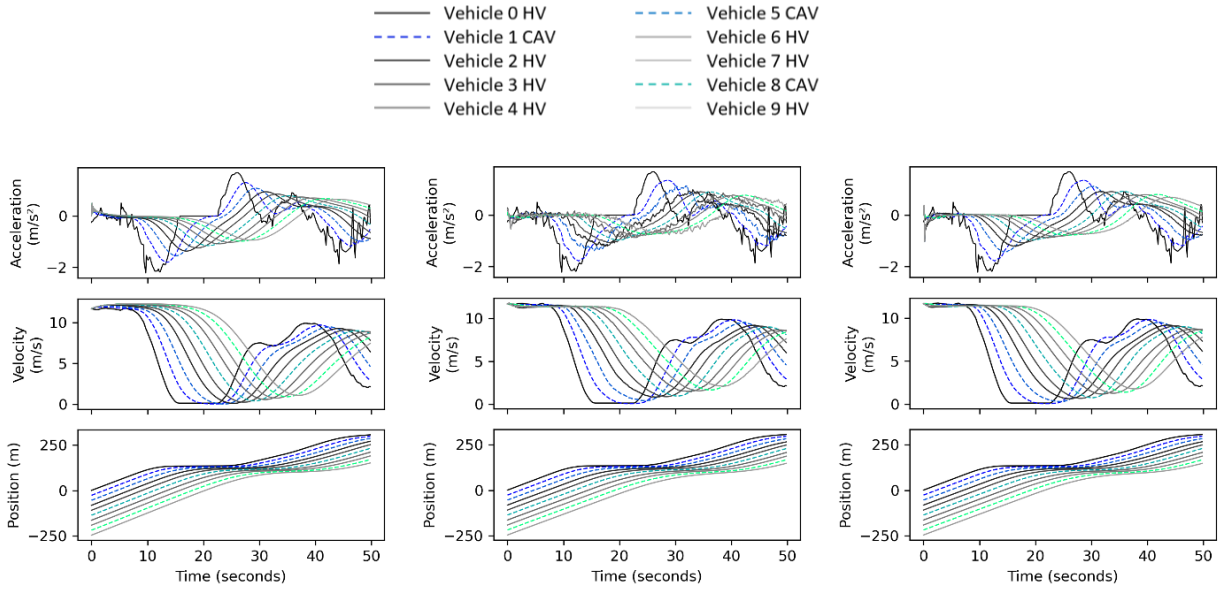
TABLE 5-6 Mix-platoon performance.

	Headway RMSE (s)	Damping Ratio	Comfort Score
Linear	0.439	0.616	0.301
RL (PPO)	0.233	0.575	0.263
PERPL (Linear+PPO)	0.204	0.558	0.249



(a) Headway RMSE (b) Damping Ratio (c) Comfort Score

Figure 5-9 the metric results of each vehicle in the mixed platoon.

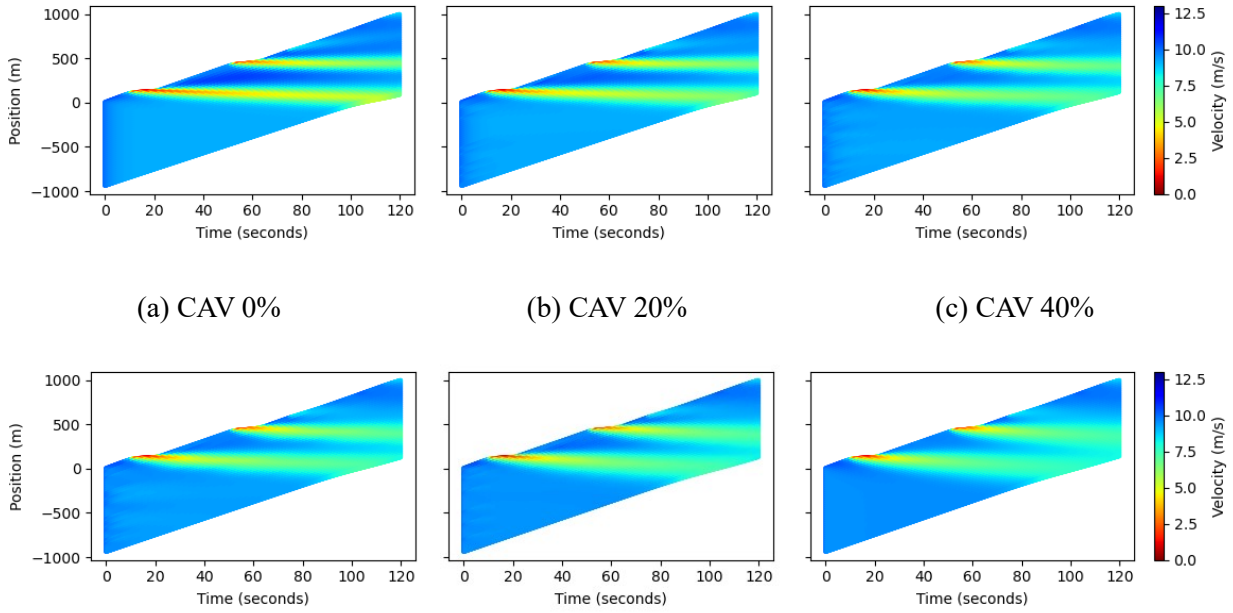


(a) Linear (b) RL (PPO) (c) PERPL (Linear+PPO)

Figure 5-10 The position, velocity, and realized acceleration results of the mixed platoon.

5.3.3.3 Mixed platoons with different penetration rates

To visually demonstrate the dampening effectiveness of the proposed control strategy, we applied the proposed method in a 40-follower mixed platoon with different penetration rates (0%, 20%, 40%, 60%, 80%, 100%). In these scenarios, the CAVs were randomly distributed throughout the mixed traffic. As shown in Figure 5-11, with higher CAV penetration rates, the fluctuations in traffic flow become progressively smoother. This indicates that as the penetration rate of CAVs increases, their control strategy contributes to stabilizing the traffic flow and reducing velocity fluctuations across the platoon.



(a) CAV 0% (b) CAV 20% (c) CAV 40%

(d) CAV 60% (e) CAV 80% (f) CAV 100%

Figure 5-11 Velocity heatmap of the mixed platoon with various penetration rates.

5.4 Conclusion

This section presents a control framework based on PERPL for decentralized platoon control. It harnesses the interpretability and robustness of linear control policies alongside the flexible, multi-objective learning capabilities of reinforcement control policies. As such, this model exhibits high control precision, achieving stable headway and reduced traffic oscillation, and it demonstrates strong generalization

capabilities, maintaining stability and safety in unseen domains. We applied our proposed model to decentralized control at the mixed traffic platoon level, utilizing CTG for cruising while considering actuator and communication delays. The model's performance was validated through artificially generated extreme scenarios and real-world trajectories. Experimental findings indicate that, both under artificial extreme conditions and with actual vehicle trajectories, our approach yields smaller headway errors and superior oscillation control compared to traditional linear and standalone RL methods. On a macroscopic traffic scale, traffic oscillations diminish as more CAVs adopt the PERPL framework, enhancing overall traffic dynamics.

For future research, we plan to leverage the safety features of the proposed method by testing it on both small-scale laboratory vehicles and larger vehicles. This approach will allow for a more realistic consideration of the cumulative effects of errors in perception, communication, and control systems that are commonly encountered in practical deployments. Moreover, we aim to enhance the capabilities of our framework by integrating it with advanced predictive models. This integration seeks to establish an end-to-end control system that reacts to immediate environmental inputs and anticipates future states. By combining real-time control adjustments with foresighted planning, the system could dynamically adapt to changes in traffic conditions, road layouts, and vehicle behaviors, significantly boosting its effectiveness in complex traffic scenarios.

6 CONCLUSION

6.1 Summary of Chapters

Chapter 1 introduces the background of CAV control in mixed traffic environment. Identifying the research objectives and scope of work.

Chapter 2 reviews the existing literature on CAVs in trajectory prediction, trajectory planning, and vehicle control. It analyzes previous studies that have sought to mitigate residuals in prediction and control, discussing their advantages and limitations, and highlighting the research gaps.

Chapter 3 presents a PERL-based method for vehicle trajectory prediction. It is validated using real-world trajectory data and the result reveals that i) the PERL model yields the best prediction when the training data is small with fewer model parameters; ii) the PERL model has faster convergence during training than NN and PINN models; iii) the PERL model consistently outperforms other models using the different physics and residual learning models. This chapter underscores the advance of the PERL structure for precise trajectory prediction, laying the groundwork for future research in combining physics priors with learning-based methods.

Chapter 4 applies this trajectory prediction method in predictive vehicle control to mitigate traffic oscillations in a mixed traffic environment. The PERL-based trajectory prediction method captures the characteristics of downstream oscillation propagation from the historical trajectories of multiple preceding vehicles. This comprehensive approach enhances the management and performance of mixed platoons. The performance of the proposed method is validated through ViL experiment. Results demonstrate the effectiveness of the proposed method in damping traffic oscillations and improving the safety and fuel efficiency of CAVs and following vehicles in mixed traffic, despite the presence of uncertain human-driven vehicle dynamics and actuator lag.

Chapter 5 proposes a PERPL-based vehicle control strategy for decentralized vehicle control to mixed traffic platoons. Experimental results demonstrate that our method achieves smaller headway errors and better oscillation dampening than linear models and RL alone in scenarios with artificially extreme

conditions and real preceding vehicle trajectories. At the macroscopic level, overall traffic oscillations are also reduced as the penetration rate of CAVs employing the PERPL scheme increases.

6.2 Future research Directions

The proposed control framework can be enhanced in several ways in the future.

First, field tests in real-world environments are essential for demonstrating and validating the effectiveness of the PERL framework. The field tests in this dissertation were limited by the number of experimental vehicles, precluding large-scale platoon experiments. Future research will aim to deploy the framework on a fleet of AVs developed at UW-Madison, which includes both Internal Combustion Engine (ICE)-based passenger cars and an Electric Motor (EM)-based van. We plan to evaluate its performance under various scenarios by first collecting comprehensive data on vehicular and system dynamics, environmental conditions, and other relevant metrics. This data will be used to assess the framework's ability to reduce error propagation and compare it against baseline methods. The field data will also support the fine-tuning of the residual learning component in PERL. This iterative process will bridge the gap between theoretical research and practical applications, ensuring the PERL framework evolves to meet real-world demands.

Second, expanding the application scope of the PERL framework is a crucial direction. This dissertation has applied the framework to vehicle trajectory prediction and control scenarios, demonstrating its effectiveness as a hybrid approach integrating physical priors with machine learning techniques. Future research should explore its performance across a broader range of problems. By comparing PERL with baseline models in diverse tasks, researchers can develop more comprehensive guidelines for determining when to use PERL, PINN, or a combination of both. Such guidelines will provide clarity on the integration of physical priors with machine learning in both prediction and control tasks.

Third, exploring the adaptability of the PERL framework with other AI methods and prior knowledge represents an exciting future avenue. The PERL framework can be extended to other AI techniques, depending on the characteristics of the AI method and the prior knowledge involved. For instance, future research could investigate the application of large language models (LLMs) in autonomous

driving. LLMs, while powerful in reasoning, face challenges such as hallucinations and probabilistic outputs, which hinder their application in safety-critical tasks like autonomous driving. A promising direction could involve using LLMs for high-level planning and reasoning while relying on stable, widely applied physical models for low-level control. This structure could combine the strengths of both LLMs and physics models, leading to safer and more stable control performance.

7 REFERENCES

Ahn, K., Rakha, H., Asce, M., Trani, A., Asce, M., Aerde, M. Van, 2002. Estimating Vehicle Fuel Consumption and Emissions based on Instantaneous Speed and Acceleration Levels 128, 182–190.

Ames, A.D., Xu, X., Grizzle, J.W., Tabuada, P., 2017. Control Barrier Function Based Quadratic Programs for Safety Critical Systems. IEEE Transactions on Automatic Control 62, 3861–3876. https://doi.org/10.1109/TAC.2016.2638961

Ang, K.H., Chong, G., Li, Y., 2005. PID control system analysis, design, and technology. IEEE Transactions on Control Systems Technology 13, 559–576. https://doi.org/10.1109/TCST.2005.847331

Bates, I.W., Karimoddini, A., Karimadini, M., 2022. A Learning-Based Approach for Diagnosis and Diagnosability of Unknown Discrete Event Systems. IEEE Transactions on Neural Networks and Learning Systems 1–14. https://doi.org/10.1109/TNNLS.2022.3204557

Brunton, S.L., Proctor, J.L., Kutz, J.N., 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the National Academy of Sciences of the United States of America 113, 3932–3937. https://doi.org/10.1073/pnas.1517384113

Cao, H., Mao, Y., Sha, L., Caccamo, M., 2024. PHYSICS-REGULATED DEEP REINFORCEMENT LEARN- ING: INVARIANT EMBEDDINGS.

Chen, D., Laval, J., Zheng, Z., Ahn, S., 2012. A behavioral car-following model that captures traffic oscillations. Transportation Research Part B: Methodological 46, 744–761. https://doi.org/10.1016/j.trb.2012.01.009

Chen, J., Li, S.E., Tomizuka, M., 2022. Interpretable End-to-End Urban Autonomous Driving With Latent Deep Reinforcement Learning. IEEE Transactions on Intelligent Transportation Systems 23, 5068–5078. https://doi.org/10.1109/TITS.2020.3046646

Chen, N., Wang, M., Alkim, T., van Arem, B., 2018. A Robust Longitudinal Control Strategy of Platoons under Model Uncertainties and Time Delays. Journal of Advanced Transportation 2018, 9852721. https://doi.org/10.1155/2018/9852721

Das, S., Maurya, A.K., 2020. Defining Time-to-Collision Thresholds by the Type of Lead Vehicle in Non-Lane-Based Traffic Environments. IEEE Transactions on Intelligent Transportation Systems 21, 4972–4982. https://doi.org/10.1109/TITS.2019.2946001

Di, X., Shi, R., 2021. A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to AI-guided driving policy learning. Transportation Research Part C: Emerging Technologies 125, 103008. https://doi.org/10.1016/j.trc.2021.103008

Dong, S., Zhou, Y., Chen, T., Li, S., Gao, Q., Ran, B., 2021. An integrated Empirical Mode Decomposition and Butterworth filter based vehicle trajectory reconstruction method. Physica A: Statistical Mechanics and its Applications 583, 126295. https://doi.org/10.1016/j.physa.2021.126295

Duret, A., Ahn, S., Buisson, C., 2011. Passing rates to measure relaxation and impact of lane-changing in congestion. Computer-Aided Civil and Infrastructure Engineering 26, 285–297.

Durrani, U., Lee, C., Maoh, H., 2016. Calibrating the Wiedemann's vehicle-following model using mixed vehicle-pair interactions. Transportation Research Part C: Emerging Technologies 67, 227–242. https://doi.org/10.1016/j.trc.2016.02.012

Fang, S., Yang, L., Zhao, X., Wang, W., Xu, Z., Wu, G., Liu, Y., Qu, X., 2024. A Dynamic Transformation Car-Following Model for the Prediction of the Traffic Flow Oscillation. IEEE Intelligent Transportation Systems Magazine 16, 174–198. https://doi.org/10.1109/MITS.2023.3317081

Gao, Z., Wu, Z., Hao, W., Long, Keke, Byon, Y.-J., Long, Kejun, 2022. Optimal Trajectory Planning of Connected and Automated Vehicles at On-Ramp Merging Area. IEEE Transactions on Intelligent Transportation Systems 23, 12675–12687. https://doi.org/10.1109/TITS.2021.3116666

Ghiasi, A., Hussain, O., Qian, Z. (Sean), Li, X., 2017. A mixed traffic capacity analysis and lane management model for connected automated vehicles: A Markov chain method. Transportation Research Part B: Methodological 106, 266–292. https://doi.org/10.1016/j.trb.2017.09.022

Ghiasi, A., Li, X., Ma, J., 2019. A mixed traffic speed harmonization model with connected autonomous vehicles. Transportation Research Part C: Emerging Technologies 104, 210–233. https://doi.org/10.1016/j.trc.2019.05.005

Gulshan, V., Peng, L., Coram, M., Stumpe, M.C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., Kim, R., Raman, R., Nelson, P.C., Mega, J.L., Webster, D.R., 2016. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. JAMA 316, 2402–2410. https://doi.org/10.1001/jama.2016.17216

Güvenç, L., Uygan, I.M.C., Kahraman, K., Karaahmetoglu, R., Altay, I., Sentürk, M., Emirler, M.T., Hartavi Karci, A.E., Aksun Guvenc, B., Altug, E., Turan, M.C., Tas, Ö.S., Bozkurt, E., Ozguner, Ü., Redmill, K., Kurt, A., Efendioglu, B., 2012. Cooperative Adaptive Cruise Control Implementation of Team Mekar at the Grand Cooperative Driving Challenge. IEEE Transactions on Intelligent Transportation Systems 13, 1062–1074. https://doi.org/10.1109/TITS.2012.2204053

Han, Y., Wang, M., Li, L., Roncoli, C., Gao, J., Liu, P., 2022. A physics-informed reinforcement learning-based strategy for local and coordinated ramp metering. Transportation Research Part C: Emerging Technologies 137, 103584. https://doi.org/10.1016/j.trc.2022.103584

He, Y., Liu, Y., Yang, L., Qu, X., 2024. Deep Adaptive Control: Deep Reinforcement Learning-Based Adaptive Vehicle Trajectory Control Algorithms for Different Risk Levels. IEEE Transactions on Intelligent Vehicles 9, 1654–1666. https://doi.org/10.1109/TIV.2023.3303408

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Networks 2, 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Hu, J., Zhang, Z., Xiong, L., Wang, H., Wu, G., 2021. Cut through traffic to catch green light: Eco approach with overtaking capability. Transportation Research Part C: Emerging Technologies 123, 102927. https://doi.org/10.1016/j.trc.2020.102927

Ivanovic, B., Leung, K., Schmerling, E., Pavone, M., 2021. Multimodal Deep Generative Models for Trajectory Prediction: A Conditional Variational Autoencoder Approach. IEEE Robotics and Automation Letters 6, 295–302. https://doi.org/10.1109/LRA.2020.3043163

Kaheman, K., Kaiser, E., Strom, B., Kutz, J.N., Brunton, S.L., 2019. Learning Discrepancy Models From Experimental Data.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed machine learning. Nat Rev Phys 3, 422–440. https://doi.org/10.1038/s42254-021-00314-5

Karpatne, A., Atluri, G., Faghmous, J.H., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N., Kumar, V., 2017. Theory-Guided Data Science: A New Paradigm for Scientific Discovery from Data. IEEE Transactions on Knowledge and Data Engineering 29, 2318–2331. https://doi.org/10.1109/TKDE.2017.2720168

KATU Staff, 2023. OSP: Driver dies after Tesla drifts off I-5 and crashes into tree, catches on fire [WWW Document]. KTVL. URL https://ktvl.com/news/local/osp-driver-dies-after-tesla-drifts-off-roadway-and-crashes-into-tree-catches-on-fire (accessed 11.7.24).

Kiefer, R.J., LeBlanc, D.J., Flannagan, C.A., 2005. Developing an inverse time-to-collision crash alert timing approach based on drivers' last-second braking and steering judgments. Accident Analysis & Prevention 37, 295–303. https://doi.org/10.1016/j.aap.2004.09.003

Kirk, R., Zhang, A., Grefenstette, E., Rocktäschel, T., 2023. A Survey of Zero-shot Generalisation in Deep Reinforcement Learning. jair 76, 201–264. https://doi.org/10.1613/jair.1.14174

Krishnapriyan, A.S., Gholami, A., Zhe, S., Kirby, R.M., Mahoney, M.W., 2021. Characterizing possible failure modes in physics-informed neural networks. Advances in Neural Information Processing Systems 32, 26548–26560.

Larsson, J., Keskin, M.F., Peng, B., Kulcsár, B., Wymeersch, H., 2021. Pro-social control of connected automated vehicles in mixed-autonomy multi-lane highway traffic. Communications in Transportation Research 1, 100019. https://doi.org/10.1016/j.commtr.2021.100019

Laval, J.A., Leclercq, L., 2010. A mechanism to describe the formation and propagation of stop-and-go waves in congested freeway traffic. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 368, 4519–4541.

Le Mero, L., Yi, D., Dianati, M., Mouzakitis, A., 2022. A Survey on Imitation Learning Techniques for End-to-End Autonomous Vehicles. IEEE Transactions on Intelligent Transportation Systems 23, 14128–14147. https://doi.org/10.1109/TITS.2022.3144867

Lee, N., Choi, W., Vernaza, P., Choy, C.B., Torr, P.H.S., Chandraker, M., 2017. Desire: Distant future prediction in dynamic scenes with interacting agents, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 336–345.

Li, J., Xie, N., Zhang, K., Guo, F., Hu, S., Chen, X. (Michael), 2022. Network-scale traffic prediction via knowledge transfer and regional MFD analysis. Transportation Research Part C: Emerging Technologies 141, 103719. https://doi.org/10.1016/j.trc.2022.103719

Li, X., 2022. Trade-off between safety, mobility and stability in automated vehicle following control: An analytical method. Transportation Research Part B: Methodological 166, 1–18. https://doi.org/10.1016/j.trb.2022.09.003

Li, X., Bai, Y., Cai, P., Wen, L., Fu, D., Zhang, B., Yang, X., Cai, X., Ma, T., Guo, J., Gao, X., Dou, M., Li, Y., Shi, B., Liu, Y., He, L., Qiao, Y., 2023. Towards Knowledge-driven Autonomous Driving [WWW Document]. arXiv.org. URL https://arxiv.org/abs/2312.04316v3 (accessed 7.25.24).

Li, X., Cui, J., An, S., Parsafard, M., 2014. Stop-and-go traffic analysis: Theoretical properties, environmental impacts and oscillation mitigation. Transportation Research Part B: Methodological 70, 319–339.

Li, X., Peng, F., Ouyang, Y., 2010. Measurement and estimation of traffic oscillation properties. Transportation Research Part B: Methodological 44, 1–14.

Li, X., Wang, X., Ouyang, Y., 2012. Prediction and field validation of traffic oscillation propagation under nonlinear car-following laws. Transportation Research Part B: Methodological 46, 409–423. https://doi.org/10.1016/j.trb.2011.11.003

Liang, Z., Han, J., Li, X., Karbowski, D., Ma, C., Rousseau, A., 2024. Testing Cellular Vehicle-to-Everything Communication Performance and Feasibility in Automated Vehicles *, in: 2024 IEEE Intelligent Vehicles Symposium (IV). Presented at the 2024 IEEE Intelligent Vehicles Symposium (IV), pp. 2917–2922. https://doi.org/10.1109/IV55156.2024.10588381

Lighthill, M.J., Whitham, G.B., 1997. On kinematic waves II. A theory of traffic flow on long crowded roads. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences 229, 317–345. https://doi.org/10.1098/rspa.1955.0089

Long, K., Liang, Z., Shi, H., Shi, L., Chen, S., Li, X., 2024a. Traffic oscillation mitigation with physics-enhanced residual learning (PERL)-based predictive control. Communications in Transportation Research 4, 100154. https://doi.org/10.1016/j.commtr.2024.100154

Long, K., Sheng, Z., Shi, H., Li, X., Chen, S., Ahn, S., 2024b. A Physics Enhanced Residual Learning (PERL) Framework for Vehicle Trajectory Prediction. https://doi.org/10.48550/arXiv.2309.15284

Long, K., Shi, H., Chen, Z., Liang, Z., Li, X., de Souza, F., 2024c. Bi-scale car-following model calibration based on corridor-level trajectory. Transportation Research Part E: Logistics and Transportation Review 186, 103497. https://doi.org/10.1016/j.tre.2024.103497

Long, K., Shi, H., Zhou, Y., Li, X., 2024d. Physics Enhanced Residual Policy Learning (PERPL) for safety cruising in mixed traffic platooning under actuator and communication delay. https://doi.org/10.48550/arXiv.2409.15595

Long, K., Shi, X., Li, X., 2024e. Physics-informed neural network for cross-dynamics vehicle trajectory stitching. Transportation Research Part E: Logistics and Transportation Review 192, 103799. https://doi.org/10.1016/j.tre.2024.103799

Makridis, M., Mattas, K., Anesiadou, A., Ciuffo, B., 2021. OpenACC. An open database of car-following experiments to study the properties of commercial ACC systems. Transportation Research Part C: Emerging Technologies 125, 103047. https://doi.org/10.1016/j.trc.2021.103047

McAllister, R., Wulfe, B., Mercat, J., Ellis, L., Levine, S., Gaidon, A., 2022. Control-Aware Prediction Objectives for Autonomous Driving, in: 2022 International Conference on Robotics and Automation (ICRA). Presented at the 2022 International Conference on Robotics and Automation (ICRA), pp. 01–08. https://doi.org/10.1109/ICRA46639.2022.9811884

Miguel, M.Á.D., Armingol, J.M., García, F., 2022. Vehicles Trajectory Prediction Using Recurrent VAE Network. IEEE Access 10, 32742–32749. https://doi.org/10.1109/ACCESS.2022.3161661

Milanés, V., Shladover, S.E., 2014. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. Transportation Research Part C: Emerging Technologies 48, 285–300. https://doi.org/10.1016/j.trc.2014.09.001

Mo, Z., Shi, R., Di, X., 2021. A physics-informed deep learning paradigm for car-following models. Transportation Research Part C: Emerging Technologies 130, 103240. https://doi.org/10.1016/j.trc.2021.103240

Mohammadian, S., Zheng, Z., Haque, Md.M., Bhaskar, A., 2023. Continuum modeling of freeway traffic flows: State-of-the-art, challenges and future directions in the era of connected and automated vehicles. Communications in Transportation Research 3, 100107. https://doi.org/10.1016/j.commtr.2023.100107

Möllerstedt, V.E., Russo, A., Bouton, M., 2022. Model Based Residual Policy Learning with Applications to Antenna Control XX, 1–21.

Naing, H., Cai, W., Nan, H., Tiantian, W., Liang, Y., 2022. Dynamic Data-driven Microscopic Traffic Simulation using Jointly Trained Physics-guided Long Short-Term Memory. ACM Trans. Model. Comput. Simul. 32, 28:1-28:27. https://doi.org/10.1145/3558555

National Highway Traffic Safety Administration, 2024. ODI Resume: Autopilot System Driver Controls (No. EA22002).

National Transportation Safety Board, n.d. Collision Between Vehicle Controlled by Developmental Automated Driving System and Pedestrian, Tempe, Arizona, March 18, 2018.

NGSIM, 2007. US Department of Transportation, NGSIM-Next generation simulation.

Pan, T., Guo, R., Lam, W.H.K., Zhong, R., Wang, W., He, B., 2021. Integrated optimal control strategies for freeway traffic mixed with connected automated vehicles: A model-based reinforcement learning approach. Transportation Research Part C: Emerging Technologies 123, 102987. https://doi.org/10.1016/j.trc.2021.102987

Pereira, M., Lang, A., Kulcsár, B., 2022. Short-term traffic prediction using physics-aware neural networks. Transportation Research Part C: Emerging Technologies 142, 103772. https://doi.org/10.1016/j.trc.2022.103772

Ploeg, J., van de Wouw, N., Nijmeijer, H., 2014a. Lp String Stability of Cascaded Systems: Application to Vehicle Platooning. IEEE Transactions on Control Systems Technology 22, 786–793. https://doi.org/10.1109/TCST.2013.2258346

Ploeg, J., van de Wouw, N., Nijmeijer, H., 2014b. Lp String Stability of Cascaded Systems: Application to Vehicle Platooning. IEEE Transactions on Control Systems Technology 22, 786–793. https://doi.org/10.1109/TCST.2013.2258346

Punzo, V., Montanino, M., 2020. A two-level probabilistic approach for validation of stochastic traffic simulations: impact of drivers' heterogeneity models. Transportation Research Part C: Emerging Technologies 121, 102843. https://doi.org/10.1016/j.trc.2020.102843

Qu, X., Yu, Y., Zhou, M., Lin, C.-T., Wang, X., 2020. Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: A reinforcement learning based approach. Applied Energy 257, 114030. https://doi.org/10.1016/j.apenergy.2019.114030

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics 378, 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

Sacks, J., Rana, R., Huang, K., Spitzer, A., Shi, G., Boots, B., 2023. Deep Model Predictive Optimization. https://doi.org/10.48550/arXiv.2310.04590

Salman, A.G., Kanigoro, B., Heryadi, Y., 2015. Weather forecasting using deep learning techniques, in: 2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS). pp. 281–285. https://doi.org/10.1109/ICACSIS.2015.7415154

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal Policy Optimization Algorithms. https://doi.org/10.48550/arXiv.1707.06347

Shan, T., Zeng, J., Song, X., Guo, R., Li, M., Yang, F., Xu, S., 2023. Physics-Informed Supervised Residual Learning for Electromagnetic Modeling. IEEE Transactions on Antennas and Propagation 71, 3393–3407. https://doi.org/10.1109/TAP.2023.3245281

She, R., Ouyang, Y., 2024. Hybrid truck–drone delivery under aerial traffic congestion. Transportation Research Part B: Methodological 185, 102970. https://doi.org/10.1016/j.trb.2024.102970

Shi, H., Chen, D., Zheng, N., Wang, X., Zhou, Y., Ran, B., 2023. A deep reinforcement learning based distributed control strategy for connected automated vehicles in mixed traffic platoon. Transportation Research Part C: Emerging Technologies 148, 104019. https://doi.org/10.1016/j.trc.2023.104019

Shi, R., Mo, Z., Huang, K., Di, X., Du, Q., 2022. A Physics-Informed Deep Learning Paradigm for Traffic State and Fundamental Diagram Estimation. IEEE Transactions on Intelligent Transportation Systems 23, 11688–11698. https://doi.org/10.1109/TITS.2021.3106259

Silver, T., Allen, K., Tenenbaum, J., Kaelbling, L., 2018. Residual Policy Learning.

Stern, R.E., Chen, Y., Churchill, M., Wu, F., Delle Monache, M.L., Piccoli, B., Seibold, B., Sprinkle, J., Work, D.B., 2019. Quantifying air quality benefits resulting from few autonomous vehicles stabilizing traffic. Transportation Research Part D: Transport and Environment 67, 351–365. https://doi.org/10.1016/j.trd.2018.12.008

Tang, Z., Chen, X., Li, Y., Chen, J., 2024. Safe and Generalized end-to-end Autonomous Driving System with Reinforcement Learning and Demonstrations.

Treiber, M., Hennecke, A., Helbing, D., 2000. Congested traffic states in empirical observations and microscopic simulations. Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics 62, 1805–1824. https://doi.org/10.1103/PhysRevE.62.1805

Vogel, K., 2003. A comparison of headway and time to collision as safety indicators. Accident Analysis & Prevention 35, 427–433. https://doi.org/10.1016/S0001-4575(02)00022-2

Wang, S., Shang, M., Levin, M.W., Stern, R., 2023. A general approach to smoothing nonlinear mixed traffic via control of autonomous vehicles. Transportation Research Part C: Emerging Technologies 146, 103967. https://doi.org/10.1016/j.trc.2022.103967

Wang, S., Teng, Y., Perdikaris, P., 2021. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM Journal on Scientific Computing 43, 3055–3081. https://doi.org/10.1137/20M1318043

Wang, S., Yu, X., Perdikaris, P., 2022. When and why PINNs fail to train: A neural tangent kernel perspective. Journal of Computational Physics 449, 1–29. https://doi.org/10.1016/j.jcp.2021.110768

Wang, X., Jiang, R., Li, L., Lin, Y., Zheng, X., Wang, F.-Y., 2018. Capturing Car-Following Behaviors by Deep Learning. IEEE Transactions on Intelligent Transportation Systems 19, 910–920. https://doi.org/10.1109/TITS.2017.2706963

Wu, J., Qu, X., 2022. Intersection control with connected and automated vehicles: A review. Journal of Intelligent and Connected Vehicles 5, 260–269. https://doi.org/10.1108/JICV-06-2022-0023

Yang, K., Tang, X., Li, J., Wang, H., Zhong, G., Chen, J., Cao, D., 2023. Uncertainties in Onboard Algorithms for Autonomous Vehicles: Challenges, Mitigation, and Perspectives. IEEE Transactions on Intelligent Transportation Systems 24, 8963–8987. https://doi.org/10.1109/TITS.2023.3270887

Yang, Y., Jiang, Y., Liu, Y., Chen, J., Li, S.E., 2023. Model-Free Safe Reinforcement Learning Through Neural Barrier Certificate. IEEE Robotics and Automation Letters 8, 1295–1302. https://doi.org/10.1109/LRA.2023.3238656

Yao, H., Li, Q., Li, X., 2022. Trajectory prediction dimensionality reduction for low-cost connected automated vehicle systems. Transportation Research Part D: Transport and Environment 111, 103439. https://doi.org/10.1016/j.trd.2022.103439

Yao, H., Li, Q., Li, X., 2020. A study of relationships in traffic oscillation features based on field experiments. Transportation Research Part A: Policy and Practice 141, 339–355. https://doi.org/10.1016/j.tra.2020.09.006

Yao, H., Li, X., 2020. Decentralized control of connected automated vehicle trajectories in mixed traffic at an isolated signalized intersection. Transportation Research Part C: Emerging Technologies 121, 102846. https://doi.org/10.1016/j.trc.2020.102846

Yao, H., Li, X., Yang, X., 2023. Physics-Aware Learning-Based Vehicle Trajectory Prediction of Congested Traffic in a Connected Vehicle Environment. IEEE Trans. Veh. Technol. 72, 102–112. https://doi.org/10.1109/TVT.2022.3203906

Yao, Z., Hu, R., Wang, Y., Jiang, Y., Ran, B., Chen, Y., 2019. Stability analysis and the fundamental diagram for mixed connected automated and human-driven vehicles. Physica A: Statistical Mechanics and its Applications 533, 121931. https://doi.org/10.1016/j.physa.2019.121931

Yue, X., Shi, H., Zhou, Y., Li, Z., 2024. Hybrid car following control for CAVs: Integrating linear feedback and deep reinforcement learning to stabilize mixed traffic. Transportation Research Part C: Emerging Technologies 167, 104773. https://doi.org/10.1016/j.trc.2024.104773

Zhang, J., Mao, S., Yang, L., Ma, W., Li, S., Gao, Z., 2024. Physics-informed deep learning for traffic state estimation based on the traffic flow model and computational graph method. Information Fusion 101, 101971. https://doi.org/10.1016/j.inffus.2023.101971

Zhao, W., He, T., Chen, R., Wei, T., Liu, C., 2023. State-wise Safe Reinforcement Learning: A Survey. https://doi.org/10.48550/arXiv.2302.03122

Zheng, S.-T., Jiang, R., Tian, J., Li, X., Treiber, M., Li, Z.-H., Gao, L.-D., Jia, B., 2022. Empirical and experimental study on the growth pattern of traffic oscillations upstream of fixed bottleneck and model test. Transportation Research Part C: Emerging Technologies 140, 103729. https://doi.org/10.1016/j.trc.2022.103729

Zhou, H., Zhang, Shanghang, Peng, J., Zhang, Shuai, Li, J., Xiong, H., Zhang, W., 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. Proceedings of the AAAI Conference on Artificial Intelligence 35, 11106–11115. https://doi.org/10.1609/aaai.v35i12.17325

Zhou, M., Qu, X., Li, X., 2017. A recurrent neural network based microscopic car following model to predict traffic oscillation. Transportation Research Part C: Emerging Technologies 84, 245–264. https://doi.org/10.1016/j.trc.2017.08.027

Zhou, Y., Wang, M., Ahn, S., 2019. Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability. Transportation Research Part B: Methodological 128, 69–86. https://doi.org/10.1016/j.trb.2019.07.001