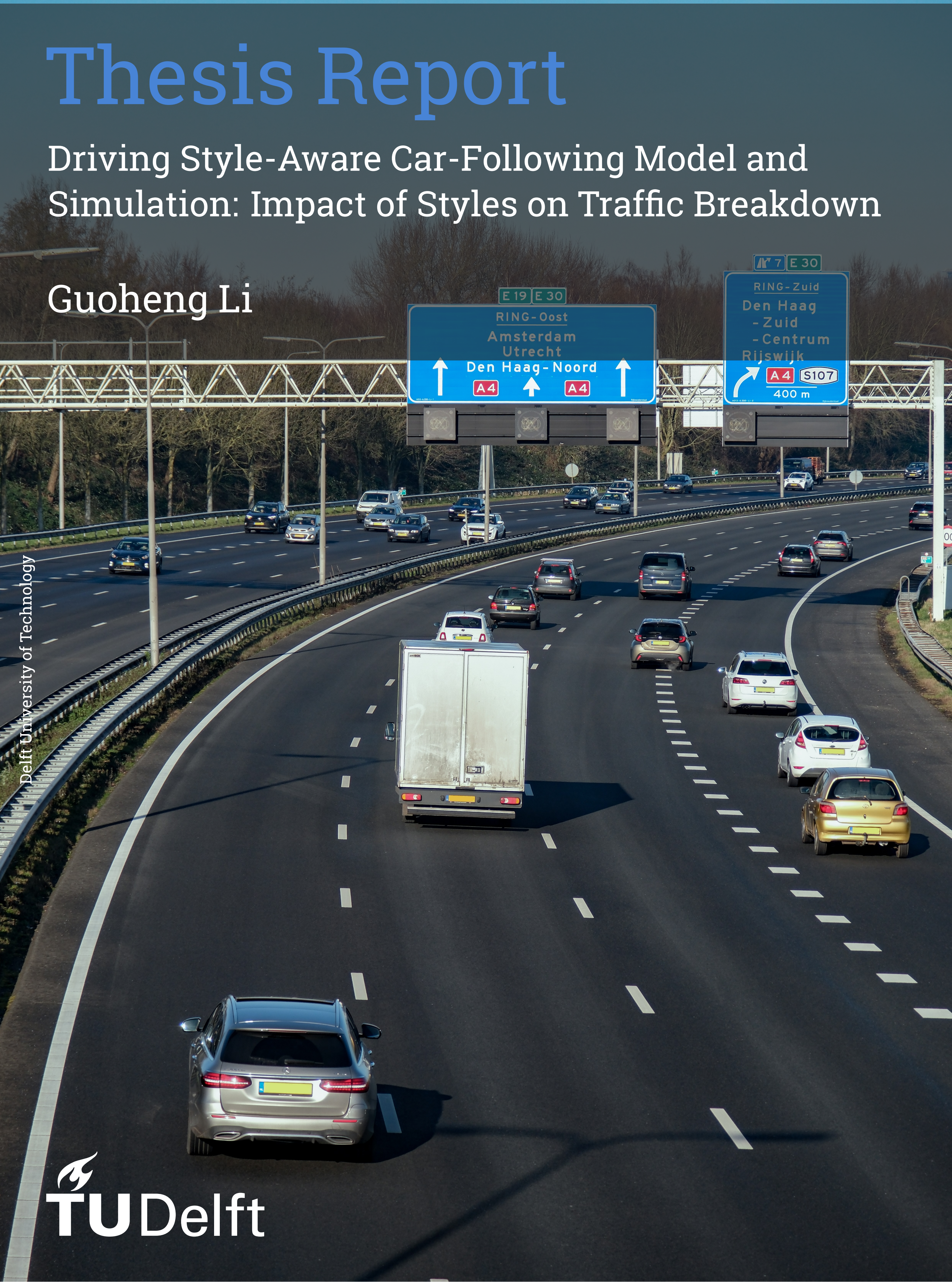
Thesis Report

Driving Style-Aware Car-Following Model and Simulation: Impact of Styles on Traffic Breakdown

Guoheng Li



Delft University of Technology

TU Delft

Thesis Report

Driving Style-Aware Car-Following Model and Simulation: Impact of Styles on Traffic Breakdown

by

Guoheng Li

Student Name	Student Number
Guoheng Li	5939259

Project duration:	February, 2025 – November, 2025	
Supervisory Team:	Dr. V.L. Knoop,	Responsible Supervisor
	Dr. A. Dabiri,	Supervisor
	Y. Jiao,	Supervisor

Cover image by Guoheng Li

Preface

This thesis marks the completion of my master's degree in Transport, Infrastructure, and Logistics at Delft University of Technology. Over the past two years, I have had the opportunity to explore various aspects of traffic flow theory, human driving behavior, and machine learning. I have always been fascinated by traffic flow and motivated to integrate data science methodologies into the study of transportation systems. These topics attracted me because they lie at the intersection of engineering, data science, and human factors—fields that together shape the future of mobility.

Working on this thesis, it has been both intellectually demanding and deeply rewarding. The research challenged me to combine theoretical insights with extensive data analysis, and to translate complex behavioral dynamics into computational models. Through this process, I have not only developed a stronger technical foundation but also gained a deeper appreciation for how data-driven methods can help us understand and improve real-world traffic systems.

I would like to express my sincere gratitude to my supervisors, Dr. Victor Knoop, Yiru Jiao, and Dr. Azita Dabiri, for their invaluable guidance and encouragement throughout this project. Victor, for his sharp insights and patience in discussions that helped me refine my ideas; Yiru, for her constructive feedback and constant support; and Azita, for her advice and inspiring enthusiasm. I am also grateful to Dr. Marco Rinaldi, the external examiner, for taking the time to evaluate this work and provide his professional perspective.

Beyond the academic side, I would like to thank my friends and colleagues at TU Delft for making these two years memorable—with shared struggles, laughter, and countless coffee breaks that made research life more enjoyable. My deepest gratitude goes to my family for their unwavering support from afar, and for always believing in me.

This master's journey has been a significant step in my academic and personal growth. While it marks the completion of one chapter, it also opens the door to new opportunities to continue exploring the fascinating world of traffic engineering.

Guoheng Li
Delft, November 2025

Summary

Traffic congestion has become a pervasive challenge worldwide, imposing severe economic and safety consequences on modern transportation systems. While infrastructure limitations are often identified as the direct cause, the diversity of human driving behaviors plays a considerable role in congestion as well. Differences in drivers' aggressiveness, anticipation, and reaction patterns shape the collective dynamics of traffic flow and strongly influence the onset of breakdown. Conventional car-following (CF) models, which characterize how vehicles follow one another through deterministic equations, capture speed and spacing dynamics effectively but fall short in representing the heterogeneity intrinsic to human driving. This limitation restricts conventional CF models from explaining how individual differences propagate in large-scale congestion phenomena.

To address this gap, this thesis develops a style-aware car-following model that integrates human behavioral diversity into data-driven motion prediction. The model is built upon the Transformer architecture and jointly learns motion prediction and driving-style embeddings—latent representations that characterize the preference of driving. A dedicated CF-oriented Zen-Traffic dataset is constructed and cleaned through kinematic filtering and car-following validation to ensure the reliability and physical consistency of the training data. Trained on this dataset, the proposed model significantly outperforms classical CF models, including the Intelligent Driver Model (IDM), as well as deep learning baselines such as LSTM and a plain Transformer, particularly in long-duration (90 seconds) simulation.

The learned driving-style embeddings are organized into four distinct and interpretable groups in two principal behavioral dimensions: aggressiveness and stability. The two dimensions align with established human-factor theories in traffic psychology, thus enabling an interpretable behavioral mapping of driver heterogeneity. Incorporating driving-style embeddings allows the data-driven CF model to generate diverse driving behavior patterns.

Building on the style-aware CF model, platoon-scale simulations were conducted to explore how microscopic behavioral heterogeneity affects macroscopic traffic performance. The experiments involving platoons with diverse driving styles and vehicle types reveal a clear stability–efficiency trade-off. Conservative styles suppress congestion waves but cause higher fuel consumption due to longer low-speed operations, while aggressive drivers appear fuel-efficient in isolation but induce instability when interacting with others. Mixing the two styles offers a practical compromise: when the stable aggressive and stable conservative styles are kept at around 50%–50% proportion, the platoon achieves both reduced delays and controlled fuel consumption. Unstable styles, however, amplify oscillations most severely compared to the other styles, destabilizing the flow and accelerating breakdown. Furthermore, the distribution of trucks in a platoon also shapes traffic stability as well: although clustered formations can occasionally reduce overall delay, dispersed formations generally enhance resilience and recovery, offering a more robust strategy under dense traffic.

In conclusion, this thesis establishes an interpretable framework that embeds human driving heterogeneity into the CF model. By combining data-driven CF modeling, behavioral embeddings, and simulation-based analysis, this thesis provides a novel perspective on how individual behavioral diversity influences collective traffic outcomes. The findings highlight that understanding and managing driving-style heterogeneity is essential to achieving stable, efficient, and sustainable traffic flow.

The main components of this research, including the model architecture, simulation framework, and algorithmic implementations, are publicly available at https://github.com/liheng423/style-cf.

Use of Generative AI

In accordance with the TU Delft Open Publishing Policies and COPE policies, the author discloses the use of generative AI tools in the preparation of this thesis. ChatGPT (OpenAI, GPT-5) was used as an auxiliary tool for language refinement, idea exploration, as well as for code completion assistance. All generated content was carefully reviewed, edited, and verified by the author to ensure accuracy, integrity, and originality. The author retains full responsibility for all ideas, analyses, and conclusions presented in this work.

Contents

I Style-aware Car-following Model

Conclusion and Discussion

Appendix

Nomenclature

Symbols

Symbol	Definition	Unit
PART I		
Δt	The duration between two records in the dataset	[s]
x_t	Ground-truth longitudinal position of the vehicle's front bumper (front end) at time t	[m]
v_t	Ground-truth speed at timestep t	[m/s]
a_t	Ground-truth acceleration at timestep t	[m/s^2]
l	Length of vehicle	[m]
Δx	Gross spacing between leader and follower ($x_t^L - x_t^F$, including l^L)	[m]
Δv	Difference of speed between a CF pair	[m/s]
$\cdot^F$	Variables related to the follower in a CF pair (e.g. l^F)	-
$\cdot^L$	Variables related to the leader in a CF pair (e.g. l^L)	-
$\hat{\cdot}$	Predicted value of model (e.g. $\hat{x}_t$, $\hat{v}_t$)	-
h	Time headway	[sec]
τ	Reaction time	[sec]
σ	Standard deviation	-
j	Jerkiness	m/s^3
$\bar{\cdot}$	Mean value	-
PART II		
q	Flow	[veh/hr]
k	Density	[veh/km]
v	Velocity	[m/s]
k_{jam}	Jam density	[veh/km]
k_{crit}	Critical density	[veh/km]
q_{max}	Traffic volume capacity	[veh/km]
FC	Fuel consumption	[L/km]

1

Introduction

1.1. Background

Traffic breakdown, or traffic congestion, has become a universal issue with far-reaching economic and safety implications. A key—but often underemphasized—factor in this problem is driving behavior. Variations in driver tendencies, such as aggressiveness versus caution, differing following distances, and varied reaction times, play a critical role in exacerbating congestion. For instance, these behavioral differences can lead to erratic driving patterns, increased likelihood of abrupt braking, and ultimately a higher risk of collisions. In the Netherlands, economic losses due to travel delays surged by 49% in 2023 compared to 2022 (Rijkswaterstaat, 2024), while congestion has also been linked to increased rear-end collisions as vehicles are forced to operate in close proximity at variable speeds (SWOV, 2022).

Breakdown typically occurs when the volume of incoming traffic exceeds a road's capacity. While infrastructure limitations (such as lane reductions, on-ramps, tunnels, and roadway sags) are often to blame, poor driving behavior significantly contributes to it. Instead of—or in addition to—expanding road capacity, suggesting a better way of driving (e.g., maintaining shorter, yet safe, headways) offers a promising approach to alleviate congestion.

Given the prominent role of driving behavior in congestion, researchers have long turned to car-following (CF) models to understand the microscopic interactions between vehicles. CF models aim to capture the response of a following vehicle to the movements of its leader on a single lane. Traditional models—including the Gazis-Herman-Rothery (GHR) model, safety distance models, Helly models, and action point models (Brackstone and McDonald, 1999)—primarily focus on quantifiable states such as speed, acceleration, and position. However, these models often struggle to account for the full spectrum of human driving behaviors, which are far more complex and varied than can be represented by a handful of parameters (Higgs and Abbas, 2015).

Recent advances in data collection and analysis have spurred the development of data-driven CF models. Unlike traditional approaches that rely on rigid mathematical formulations, data-driven models can learn directly from high-resolution car-following data. This enables them to capture nuanced driving behaviors—such as individual variations in reaction time, acceleration patterns, and headway preferences—that are pivotal in understanding and mitigating traffic breakdown (Papathanasopoulou and Antoniou, 2015). By better representing the complexity of real-world driving, these human-like models provide deeper insights into how different driving styles contribute to traffic oscillations and bottlenecks.

1.2. Motivation

Building on these data-driven CF models, it is feasible to develop a style-aware Car-following model, which can offer more insights into the styles, such as discovering new styles. Furthermore, the model can be incorporated into a simulation to examine how individual driving behavior shapes breakdown. By modeling interactions among multiple vehicles under different conditions, researchers can analyze how collective behaviors influence breakdown patterns, assess the impact of potential interventions, and refine traffic-management strategies. Such simulations improve our understanding of breakdown and allow systematic testing of how specific driving styles amplify or dampen traffic oscillations.

In summary, recognizing and modeling diverse driving styles is key to **understanding and improving breakdown**. Data-driven CF modeling and multi-vehicle simulations enable a more complete view of breakdown mechanisms and support the development of more effective, human-centric traffic management solutions.

1.3. Research Questions

This study aims to address the following research question:

How do different driving styles shape traffic breakdown in car-following?

To answer the major question, this study proposes the following sub-questions:

1. **Driving styles**: What is the heterogeneity in driving styles, and how do we distinguish them?
2. **Car-following**: What are the key characteristics in existing CF models, and why are data-driven CF models necessary?
3. **Styles and their Impacts on Traffic**: Which driving styles (or vehicle types), and which combinations thereof, contribute to traffic breakdown and less energy efficienct under given traffic conditions?

1.4. Main Contributions

- The study introduces a style-embedding framework that automatically extracts structured representations of driving styles from trajectory data. It establishes an interpretable style space—organized along two human-factor dimensions (aggressive–timid and stable–unstable)—providing a principled and explainable foundation for analyzing and integrating driving style characteristics into subsequent tasks.
- A car-following model is developed that integrates the style embedding (token) derived from the

embedder. This style-aware model outperforms both classical and state-of-the-art benchmarks in prediction accuracy. Besides, the model allows specifying an agent as either a truck or a passenger car by adjusting its vehicle length parameter.

- To investigate traffic breakdown holistically, the study develops fully automated evaluation indicators that combine microscopic indicators with calibrated macroscopic fundamental diagrams, thereby enabling large-scale collection and analysis of experimental results.
- The characteristics of different driving styles and their compositions in a platoon are examined in the context of traffic breakdown. The results demonstrate that stable and conservative drivers contribute most effectively to improving overall traffic at the cost of higher fuel consumption. Furthermore, the distinctive role of trucks in traffic flow is investigated.

1.5. Thesis Structure

This study is divided into two parts: (i) style identification and car-following modeling (Research Questions 1 and 2), and (ii) the impacts of driving styles on traffic breakdown (Research Question 3). See Figure 1.1 for more information.

In the first part, this study establishes a concrete definition of driving style, framing it as a consistent and preferred way of driving. Unlike conventional approaches that characterize heterogeneity qualitatively, this work provides a quantitative formulation. Trajectories are clustered based on their embeddings, which are learned representations trained specifically to encode driving style. To improve interpretability, each cluster is interpreted according to relevant human factors in car-following, as identified in prior studies.

Through this formulation, the study effectively reveals the heterogeneity in driving styles and distinguishes them within the learned embedding space. At the same time, the resulting clusters are endowed with plausible and interpretable meanings grounded in empirical driving behavior research.

Regarding car-following modeling, this study conducts a comparative analysis of classical and state-of-the-art CF models against the proposed style-based Transformer model. The performance limitations of the benchmark models are attributed to their architectural constraints and design choices, highlighting their shortcomings in capturing human-like behavior. These findings underscore the necessity of data-driven approaches—particularly the proposed style-aware CF model—for modeling realistic and individualized car-following behavior.

In the second part, building on the style-aware car-following model and the driving styles extracted from the dataset, the study investigates the connections between driving styles and traffic breakdowns. The analysis relies on a series of platoon simulations that incorporate different vehicle types (truck and passenger car) and driving styles (stable/unstable and aggressive/timid). These experiments aim to provide a general profile of how different styles contribute to traffic congestion. In addition, specific platoon compositions—such as alternating-style platoons—are examined to highlight recommended formations that can enhance overall traffic efficiency.

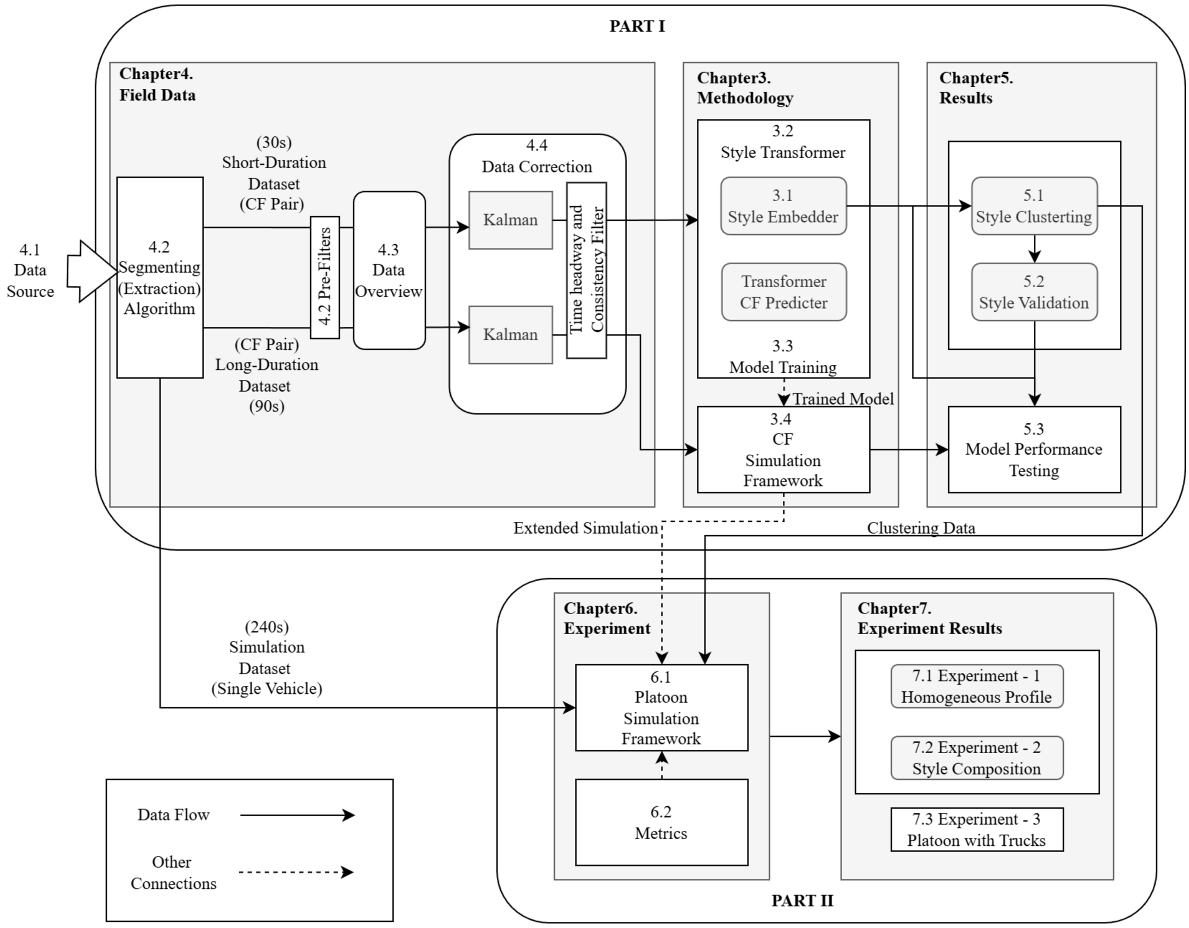


Figure 1.1: Overview of the Thesis Structure

2

Literature Review

To conduct a comprehensive literature review and address the primary research question, the following keywords are utilized in Scopus and Google Scholar to cover all relevant aspects of the study (Table 2.1)

Given the growing prominence of data-driven models in recent years, the search results are occasionally sorted by publication date to ensure that no relevant data-driven CF studies are overlooked.

This literature review addresses three subtopics relevant to the study. It first defines the concept of driving style and examines existing research on its clustering and characterization. It then explores the relationship between driving style and traffic breakdown in the current research, highlighting how individual behavioral variations may contribute to flow instability. Finally, it discusses key considerations for incorporating driving style into car-following models and reviews the state-of-the-art approaches in this domain.

2.1. Heterogeneous Driving Styles

Congestion can arise from the interaction of heterogeneous driving styles, making it essential to define and analyze driving style as a key factor in traffic dynamics.

Review	Keywords
Congestion	Congestion Dissipation Car-following, Traffic Congestion, Stop-and-go waves, Traffic Oscillation
Driving Style	Driving style analysis, Driving behavior classification
Historical CF Development	Car-Following Model, Car-following Congestion
Data-driven CF	Data-Driven Car-Following, Deep Learning Car-Following

Table 2.1: Keywords used in literature review

To clarify, according to the definition by Sagberg et al. (2015), driving style refers to a consistent set of behavioral preferences inherent to the driver and independent of the driving context, whereas driving behavior is context-dependent and shaped by driving context, such as traffic density. Accordingly, driving style captures stable traits—such as preferred following distance and reaction time—that systematically influence car-following behavior and are typically embedded in microscopic traffic models such as the IDM and Newell's model.

2.1.1. Human-factor in Driving

Human factors are a key topic in this thesis, explaining why drivers behave differently under various conditions. Since the car-following model is a psycho-physical model involving complex human factors, it is important to clearly specify which factors have been recognized in previous research. These factors help to interpret the clustering of driving styles.

In Saifuzzaman and Zheng (2014), several human factors are mentioned. Some of these can be quantitatively observed in the trajectory data, and will be used to qualitatively describe a specific type of driver, while others cannot. Table 2.2 lists the details of these factors.

Table 2.2: Human factors in driving style, categorized by observability from trajectory data

Factor	**Description**
Observable from trajectory data	
Reaction time	Can be derived using DTW by pairing two speed sequences
Temporal anticipation	Drivers can predict traffic situations for the next few seconds
Aggressiveness	Measured by time headway and reaction time
Driving skills	Assessed by the standard deviation of headway (σ_h) and reaction time (σ_τ)
Distraction	Partially reflected by variations in σ_h and σ_τ
Not directly observable from trajectory data	
Socio-economic characteristics	e.g., age, gender, income, education
Estimation errors	Spacing and speed are perceived with limited accuracy
Perception threshold	Humans cannot detect minor changes in stimuli
Spatial anticipation	Drivers consider multiple vehicles ahead, not directly measurable
Context sensitivity	Driving style may vary depending on traffic context
Imperfect driving	Same driver may behave inconsistently under similar conditions

2.1.2. Clustering Driving Style

Driving behavior is considered the outcome of an intricate psychological process (Ranney, 1999). Each cluster's attributes and peculiarities can be described phenomenologically using a set of criteria (e.g., average headway, acceleration, etc.). This clustering forms the foundation for analyzing drivers prone to causing congestion, which is of our interest.

The heterogeneity in driving behaviors significantly influences model performance. Ossen and Hoogendoorn (2011) demonstrates that this heterogeneity impacts the effectiveness of conventional microscopic models, highlighting their limitations. This necessitates a shift toward more generic models.

Efforts to identify clusters of potential driving styles continue, with no definitive conclusions yet reached. Building on the GHR model, Higgs and Abbas (2015) suggests the existence of approximately 30 distinct clusters of car drivers, each characterized by unique parameter settings. Moreover, this research highlights that truck driving behaviors are generally more homogeneous than those of passenger cars. Additionally, Chen and Chen (2019) identified three significant clusters based on the features of speed maintenance, lateral acceleration, braking, and longitudinal acceleration maneuvers. Attempts have been made to explain these clusters by driver attributes such as gender and age, yet the associations remain tentative.

Therefore, the clustering of driving styles remains at a very preliminary stage, with neither an established number of clusters nor a clear understanding of the mechanisms behind such clustering.

2.2. Driving Style in Traffic Breakdown

As stated, congestion arises when road capacity is insufficient to accommodate the traffic volume, often due to factors such as incidents or accidents that reduce capacity. In traffic flow theory, congestion is defined as a condition where the density of vehicles on a given road segment exceeds the critical density.

Traffic oscillations are a common cause of congestion and are frequently observed on roadways. The oscillation is termed as stop-and-go waves as well. Drivers typically experience the alternating stop-and-go phases while driving through congested areas but often remain unaware of the underlying causes (e.g. bottleneck, accident) of the congestion. When such waves propagate upstream without dissipating, they can escalate and intensify the congestion, pushing the local traffic state into the congested regime (Suh and Yeo, 2016).

The onset of the oscillation is ascribed to be longitudinal (e.g. instability in driving behaviors) and lateral (e.g. lane-changing maneuver) induced by spontaneous driving behaviors or external factors (e.g. bottleneck), the growth or dissipation of the waves are statistically verified to be independent of the trigger of such waves (Zheng et al., 2011).

As perturbations could occur in the normal traffic flow regardless of being spontaneous or external, the developing stage is more of interest in research. Instead of originally conjectured to be asymmetric acceleration and deceleration when leaving and entering the oscillation, the growth/dissipation of the oscillation is deemed to be driving patterns (behaviors) in more later studies (Laval and Leclercq, 2010). Simultaneously, Laval and Leclercq (2010) suggests that the actions of drivers, such as those who are aggressive or timid, contribute to the intensification of oscillations. Hence, it is crucial for us to explore which specific driving patterns worsen the stop-and-go waves, beyond merely focusing on aggressive and timid drivers.

In prior research (Laval and Leclercq, 2010; Zheng et al., 2011; Suh and Yeo, 2016), the classification of driving behaviors remains at a relatively basic level, focusing on factors such as reaction time, minimum spacing, and acceleration. While this simplicity aids in analysis, it may overlook broader clusters of driving patterns beyond merely aggressive and timid behaviors that exacerbate the oscillation into congestion.

2.3. Development of Conventional Car-following (CF)

To explore the impact of driving style, we need a tool to extract relevant behavioral features from trajectory data. Car-following (CF) models serve this role by translating raw trajectories into interpretable variables like acceleration, headway, and reaction time. These outputs allow us to analyze driving style and its influence on traffic breakdown. Therefore, CF model is essential for the experiments that follow.

Early car-following (CF) models were designed to describe longitudinal vehicle interactions through deterministic rules, often assuming homogeneous driver behavior. The GHR model (Chandler et al., 1958) represents a foundational attempt, where acceleration is a function of spacing, relative speed, and own speed. Despite its historical importance, its lack of robustness and contradictory parameter estimations limited its long-term relevance (Brackstone and McDonald, 1999).

Later developments introduced behavioral considerations. Collision avoidance models such as Gipps and IDM (Gipps, 1981; Treiber et al., 2000) separated driving into distinct regimes (e.g., free-flow and interaction), relying on intuitive parameters like reaction time and deceleration. While more realistic, they remain highly sensitive to calibration and often assume uniform behavioral patterns.

Other models, such as Helly's linear model, incorporate driver perception thresholds, aligning with the notion of action points (Brackstone and McDonald, 1999). Newell's model (Newell, 2002), though parsimonious and analytically efficient, simplifies car-following behavior as a fixed delay in trajectory replication—again overlooking inter-driver variability.

Across these formulations, a common limitation remains: the inability to account for driver heterogeneity and complex cognitive processes. Several empirically observed behavioral phenomena are often omitted to preserve mathematical simplicity:

1. **Memory effect** is initially proposed assuming the driver reacts to the speed of its leader in a period of time, rather than merely in a time instant (Lee, 1966). The model successfully smoothens the acceleration profile. However, due to the complexity of solving an integral, the memory effect stayed unnoticed till the advent of data-driven models. The concept of memory can be extended to the actions (speed, acceleration) by the follower previously, not merely the speed of the leader.
2. **Reaction time** is the time lag between the maneuver from the leader and the reaction made by the follower. Firstly specified in Chandler et al. (1958), the reaction time becomes a well-acknowledged effect existing in CF.
3. **Psychological mechanism** contends that driving is fundamentally a cognitive and perceptual process, characterized by inherent uncertainties since drivers are not fully aware of the precise distance or speed of the vehicle in front of them. Or the other way around, the drivers don't react till the limit is reached (i.e., distinguishable enough by a human driver). Therefore, the fuzzy-logic framework and Leutzbach and Wiedemann (1986) are consequently formulated. However, the models are criticized for their mathematically unsound and complicated (Brackstone and McDonald, 1999).
4. **Heterogeneity in driving behaviors** is undeniably a widely recognized phenomenon. Nonetheless, psychological factors play a significant role behind the scenes, resulting in CF models applying a different set of parameters to account for this variability. A related topic is driving behavior

clustering, which will be discussed in further detail later.

5. **Asymmetric effect** is believed to be one possible culprit in exacerbating traffic oscillation at the developing stage. This asymmetric response can intensify traffic oscillations, making it a key phenomenon to capture in CF modeling. The effect implies the drivers don't behave the same when accelerating and decelerating.

These limitations motivate the need for more flexible, data-driven approaches that can implicitly capture complex behavioral traits without predefined functional forms.

2.4. Data-Driven CF Model

Conventional car-following (CF) models have become increasingly intricate to better capture observed traffic phenomena. For instance, the GHR model assumes continuous interaction with the leader, while models such as Gipps and IDM incorporate distinct regimes, including free-flow and interaction phases. However, introducing additional regimes complicates the analytical framework and poses challenges in integrating measurement data. These limitations have spurred a transition toward data-driven models, which provide greater flexibility and reduce sensitivity to predefined regimes (Papathanasopoulou and Antoniou, 2015). Moreover, psychological studies on driving behavior suggest that drivers' behavior may evolve over time depending on the prevailing traffic conditions (Ranney, 1999). These developments highlight the looming bottlenecks in conventional CF models and pave the way for exploring alternative solutions.

Early data-driven CF models based on shallow architectures like MLPs and Elman networks showed improved accuracy over classical models (Simonelli et al., 2009; Colombaroni and Fusco, 2014), but struggled to incorporate temporal dependencies. Some also attempted to model driver psychology via fuzzy rules (Chong et al., 2013), yet lacked the capacity to fully capture the memory effect and complex behavioral dynamics.

Initial experiments primarily leverage the capabilities of time-series data, yet they are constrained by basic and shallow neural network designs like MLPs, which inherently lack memory mechanisms and aren't specifically optimized for handling time-series information. In the deep-learning era, though, temporal features can be seamlessly integrated with specialized neural network architectures, enhancing data-driven results significantly.

Recurrent Neural Networks (RNNs) and Long-Short Term Memory (LSTM) models are widely recognized within the time-series deep-learning domain. Zhou et al. (2017) introduces an RNN model for forecasting CF, asserting that RNNs can intrinsically capture the **heterogeneity in driving behavior** (Phenomena 4), unlike IDM's parameters which reflect a generalized scenario and struggle to predict specific driving patterns such as aggressive driving. Huang et al. (2018) places emphasis on utilizing the LSTM's ability to capture **asymmetric effects** (Phenomena 5). Wang et al. (2019) developed a similar time-series GRU network to study its performance in reproducing the **action point** (Phenomena 3). On the other hand, Ma and Qu (2020) employed a sequence to sequence model to break the output length limit, which is required to be less or equal to the input size, in LSTM and similar time-series model. On top of that, the **reaction time** (Phenomena 2) is externally calibrated and plugged into the model.

Some studies also explore hybrid approaches, where models like IDM are incorporated into deep networks as regularizers (Mo et al., 2021; Geng et al., 2023). While these improve generalization and robustness, their reliance on predefined physical rules limits the ability to fully capture diverse and complex driving behaviors.

Table 2.3: Existing studies summary in deep learning CF framework (subscript denotes the length of the time series).

Reference	Model	Inputs	Outputs	Loss	Memory (s)
Zhou et al. (2017)	RNN	$[\Delta x_t, \Delta v_t, v_t^F]_{t-10:t}$	a_{t+1}^F	MSE(a_t^F)	1
Morton et al. (2017)	LSTM + Gaussian Mixture	$[\Delta x, \Delta v, v^F, a^F]_{t-100:t}$	Distribution(a_{t+1}^F)	MSE(v_t^F)	10
Huang et al. (2018)	LSTM	$[\Delta x, \Delta v, v^F]_{t-100:t}$	$[\Delta x_{t+1}, v_{t+1}^F]$	MSE(x_t^F) + MSE(v_t^F)	10
Wang et al. (2019)	GRU	$[\Delta x, \Delta v, v^F]_{t-10:t}$	a_{t+1}^F or v_{t+1}^F	MSE(x_t^F)	10
Ma and Qu (2020)	Seq2Seq (LSTM)	$[\Delta x, \Delta v, v^F]_{t-50:t}$	$a_{t+1:t+h}^F$	MSE(x_t^F)	5
Mo et al. (2021)	LSTM + IDM	$[\Delta x, \Delta v, v^F]$	a_{t+1}^F	MSE(a_t^F)	1
Geng et al. (2023)	Transformer + IDM	$[\Delta x, \Delta v, v^F, x^F]_{1:t}$, $[x^L, v^L]_{t:t+h}$	$x_{t+1:t+h}^F$	MSE(x_t^F)	5

Overall, deep learning models have demonstrated notable improvements in predictive performance and behavioral fidelity by incorporating temporal dynamics and driver heterogeneity. They also show promise in capturing rich driving information by incorporating possible phenomena in CF, which are related to human factors, though doing so requires careful architectural design and task formulation in training.

2.5. Research Gap

1. While driving styles and behaviors have been acknowledged as significant, there is a lack of studies that incorporate clustering during the training phase in data-driven models. In current practice, clustering occurs independently of the training phase, using standard clustering methods, followed by the development of several models tailored to different driving patterns (Zhu et al., 2018; Zhou et al., 2017). Additionally, the clustering remains rather basic, often characterized simply as timid or aggressive (Laval and Leclercq, 2010; Zhang et al., 2022). Adopting a data-driven model allows deeper insight into the clusters produced in an unsupervised manner through post-hoc analysis of the identified driving style groups.

2. Existing data-driven CF models rarely incorporate components explicitly dedicated to representing driving styles or ensuring that style-related features are meaningfully reflected in the model outputs (Zhou et al., 2017; Zhu et al., 2018; Ma and Qu, 2020). Furthermore, few studies discuss how input design can facilitate the learning of driving styles, leaving the relationship between model inputs and style representation largely unexplored.

3. The relationship between driving style and traffic breakdown remains elusive in existing research. Although some preliminary progress has been made through analytical experiments based on the

Newell CF model, these studies often neglect many influencing factors to preserve model parsimony (Zheng et al., 2011; Laval and Leclercq, 2010; Chen et al., 2014). Therefore, this study aims to investigate this relationship using a fully data-driven modeling and simulation approach. This involves designing breakdown experiments to examine how heterogeneous driving styles influence traffic flow stability and the onset of breakdown.

Part I

Style-aware Car-following Model

3

Methodology

3.1. Style Embedder

Although the existence of driving behavior is well acknowledged in the literature, there is no consensus on how to quantitatively define driving style, which is shaped by complex psychological and behavioral processes (Ranney, 1999). Following Sagberg et al. (2015), this study defines driving style as a habitual way of driving—a recurrent pattern of behavior across driving occasions that encompasses both automatized and consciously controlled actions. In contrast, driving behavior is understood as context-dependent reactions.

Given data limitations—specifically, the short trajectory lengths per driver—we assume that driving behavior may vary with traffic context, whereas driving style remains stable over time. To assess this assumption, we leverage a data-driven embedding model to examine whether long-term style traits can be inferred from observed trajectories.

However, traditional models such as the IDM often assume that drivers behave optimally, which fails to capture real-world variability. Moreover, they rely on predefined, context-insensitive parameters (e.g., average time headway). In contrast, our approach learns latent driving style representations directly from raw trajectory data, aiming to provide empirical evidence on whether driving style constitutes a stable driver-level trait or a context-dependent phenomenon.

3.1.1. Driving Style Features

Previous studies have revealed links between human factors and driving styles (Table 2.2). In this study, these factors are incorporated as input features for the data-driven style embedder model.

Meanwhile, previous studies have linked certain quantities to driving styles, which are widely used in classical car-following models. Instead of letting the data-driven model discover these patterns implicitly, we incorporate domain knowledge into feature selection to enhance interpretability. The selected features serve both as behavioral indicators and as a basis for understanding the learned style embeddings.

Human factors play a significant role in car-following behavior. While some of these factors cannot be directly examined due to the limitations of trajectory-only data, others—such as reaction time—are implicitly embedded in the observations. Quantifying these latent factors is essential for advancing this research. Although using hand-crafted features introduces potential issues, such as definitional ambiguity and sensitivity to noise, it remains necessary to define several style-related metrics. These features allow for the integration of domain knowledge into the model and support the interpretation of both the learned embeddings and the resulting style clusters.

Note that both reaction time τ and time headway h are extracted as time series and directly used as input features for the style embedder (see Table 3.3 for the style embedder's complete input series).

Reaction time

Reaction time τ is defined as the temporal gap between a leader's action and the corresponding response of the follower. In the literature, Sharma et al. (2018) propose a dynamic time warping (DTW) based estimation. However, this method may fail in cruising regimes—when both vehicles travel at nearly the same speed—since the leader–follower trajectories become indistinguishable (see Appendix Dynamic Time Warping for details).

To address this limitation, we introduce a dynamic upper bound on the DTW-estimated reaction time. The upper bound is determined by the inter-vehicle spacing in combination with an assumed traffic-wave speed (setting $w = -15$ km/h), using the ground-truth $w = \Delta x/\tau$. This ensures that the estimated delay does not exceed a physically plausible value. This modification prevents the DTW method from producing unrealistic results in near-cruising scenarios, while preserving its ability to capture reaction dynamics in other contexts.

Time headway

Time headway h, as defined in Equation 4.3, can be used to describe driver aggressiveness, driving skills, and other human factors in car-following behavior. For instance, the Intelligent Vehicle Model (IDM) includes the desired time headway as a key parameter. Therefore, in this study, time headway is selected as one of the style-related features.

3.1.2. Driving Style Embedding Model

According to Table 2.2, some factors—such as socio-economic characteristics, estimation errors, spatial anticipation, and context sensitivity—are not available in the dataset. Therefore, including these factors in the analysis is not feasible in this study; however, this does not imply that they do not contribute to the diversity of driving styles. Other factors are implicitly represented in the dataset, but there is still no consensus on how to quantify them. This study aims to develop methods to quantify these factors and analyze their contribution to the clustering of driving styles, regardless of whether they are directly used as input data. These factors are expected to provide a meaningful interpretation of the classification based on embeddings produced by a deep learning model.

Driving style is often conceptualized as a time-independent characteristic. Consistent with this view, the present study also treats driving style as a time-independent latent factor. Since driving style is not directly observable, it typically requires calibration when used as input for downstream models. In contrast, this study proposes a data-driven embedding model that automatically infers driving style from vehicle trajectory data.

Rather than performing direct temporal aggregation—which risks discarding temporally sensitive features such as reaction delays—this study leverages a neural attention mechanism to preserve and exploit the temporal dependencies. Specifically, an attention layer (see Figure A.5) is trained to capture correlations between different timesteps within a trajectory sequence.

The encoder component of the Transformer architecture (Figure A.4), namely the left-hand block, serves as the core of the driving style embedding model. While the encoder outputs a sequence of hidden representations across time, a single style representation (i.e., the driving style token or embedding) is ultimately obtained by aggregating the encoder's output over the temporal dimension. This aggregated token serves as a compact, time-independent representation of the driver's behavior. The complete process is illustrated in (Figure 3.1).

The model takes as input a sequence with shape $(\text{Time}, \text{Feature})$, setting $T_{\text{style}} = T_{\text{hist}}$ (see Figure 3.1), and produces a fixed-length vector representation, where the output dimensionality is user-defined. Notably, the resulting embedding is time-independent, meaning it does not retain any explicit temporal information. Moreover, the model is capable of processing sequences of arbitrary length along the time axis, providing flexibility in handling vehicle trajectories of varying durations.

The details of the architecture are illustrated in Table A.1.

3.2. Style-aware CF Model

Based on the plain Transformer model described in Section Transformer Model, the Style-aware CF model extends this architecture by incorporating a style embedding. This embedding serves as additional information to regulate the model's behavior. Ideally, the model should be able to recognize the input and utilize this information effectively.

Rather than merely relying on the Transformer model for prediction, driving style can also be extracted from time series data. However, the style embedding is represented in a time-independent format, meaning it lacks temporal information. Integrating the processed decoder series, the encoder output, and the style embedding presents a significant challenge. To address this, the following section explains the proposed approach in detail.

3.2.1. Integrating Style to CF model

The architecture of the style-aware CF model is illustrated in Figure 3.1, and architecture specifications are provided in Appendix Style-CF Model Architecture.

The model consists of two major parts: i) a style embedder, an independent Transformer encoder, which produces a time-independent style token. The CF Transformer integrates the style token from the embedder and produces the acceleration prediction.

Worth noticing, the model takes three inputs: encoder and decoder sequences, representing the standard car-following transformer structure, and a style series, which serves as complementary input.

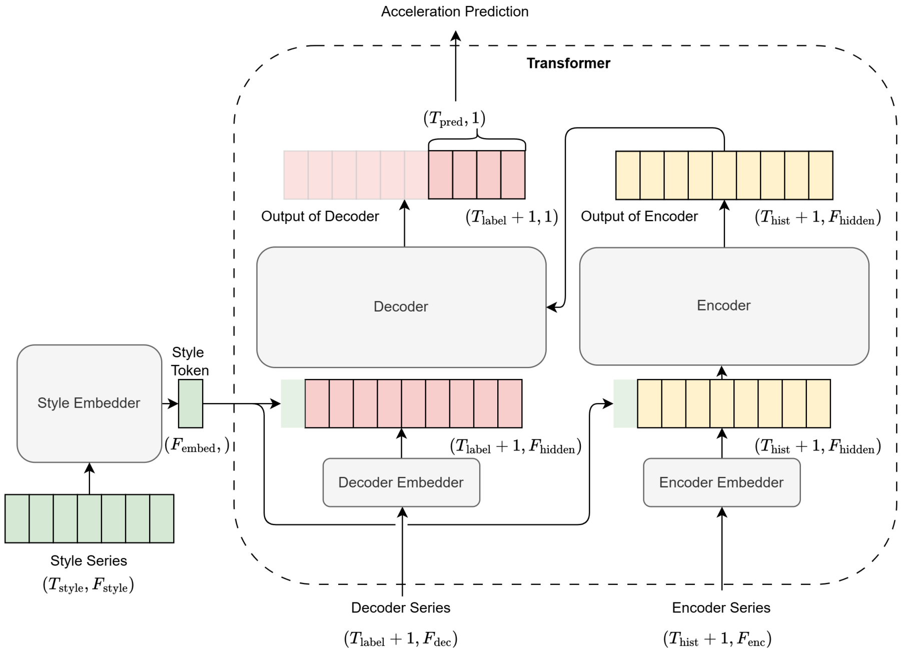

Acceleration Prediction
Transformer
$(T_{\text{pred}}, 1)$
Output of Decoder
$(T_{\text{label}} + 1, 1)$
Output of Encoder
$(T_{\text{hist}} + 1, F_{\text{hidden}})$
Decoder
Encoder
Style Token
Style Embedder
$(F_{\text{embed}},)$
$(T_{\text{label}} + 1, F_{\text{hidden}})$
$(T_{\text{hist}} + 1, F_{\text{hidden}})$
Decoder Embedder
Encoder Embedder
Style Series
$(T_{\text{style}}, F_{\text{style}})$
Decoder Series
$(T_{\text{label}} + 1, F_{\text{dec}})$
Encoder Series
$(T_{\text{hist}} + 1, F_{\text{enc}})$

Figure 3.1: Integrating the style embedding (token) into the Transformer CF model (The details of the architecture can be referred to section A.4)

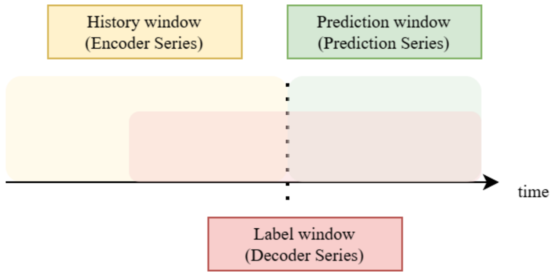


Figure 3.2: Definitions of inputs and outputs of the Transformer model (also of Transformer part in Style-aware CF model)

The style series is originally a time sequence but is transformed into a time-independent embedding (token) through the style embedder before being prepended into the model. This design allows the style embedder part to participate in training. As a result, the model produces two outputs: i) a style embedding, useful for clustering, driving style analysis, other downstream tasks, and optionally for loss function computation; ii) the predicted acceleration in the prediction window, used in car-following prediction and loss function.

This design allows the attention mechanism to dynamically decide whether and how to utilize the style information, and such a utilization can be later revealed in the attention matrix.

3.2.2. Inputs and Outputs Definitions

Since the Transformer architecture requires multiple inputs, which could be confusing at the beginning, the details of the definition and the segmentation of **a single training sample** are further illustrated.

Similar to the standard Transformer input, the structures of the encoding and decoding series in Style-aware CF model follow the plain Transformer architecture. In this study, specifically, we define the input series in the following notations.

Figure 3.2 shows the structure of one training sample. The dashed line separates the historical window T_{hist} from the prediction window T_{pred}. The label window T_{label} includes the leader's behavior that the follower reacts to. Specifically, the part of the label window on the left (i.e., inside the history window) is further defined as the History Label window, and the part on the right (i.e., inside the prediction window) is defined as the Future Label window.

Importantly, although the label window spans both past and future relative to the decoder, the masking mechanism restricts visibility to only the earlier time steps within the prediction horizon (Appendix Causality and Masking). In effect, the visibility matrix becomes lower-triangular padding with 0 (visible), allowing access to past information while blocking all future positions. This scheme is applied to features related to the leader, i.e. v^L in this study.

To prevent data leakage, features related to the follower vehicle—such as its velocity—are not used in the decoder input during the prediction window. These features are masked using average values from the historical segment to avoid introducing future knowledge. This scheme is applied to features related to the follower, i.e. v^F in this study.

The key difference between the plain Transformer CF model and the Style-aware CF model lies in the additional style series input (Figure 3.1), which is constructed using style-relevant features identified in Table 3.1. Specifically, the selected features—$(\tau_t, h_t, \Delta x_t, v_t^F)$—capture both driving style (τ_t, h_t) and regime $(\Delta x_t, v_t^F)$. During training, v_t^F in the prediction window is masked out to prevent data leakage. The organized details of inputs, outputs, and model structure can be found in Table 3.3.

3.3. Model Training

3.3.1. Setup

The Style-aware CF model was trained on a workstation equipped with an Intel i7-12700K CPU, an NVIDIA RTX 3070Ti GPU (8 GB VRAM), and 32 GB RAM, using PyTorch 2.1 with CUDA 12.1. Training was conducted for 30 epochs with a batch size of 64. The Adam optimizer was employed with an initial learning rate of 10^{-4} and a weight decay of 10^{-5}. kinematic position loss (see Equation 3.1) was used as the training objective, and a step-based learning rate scheduler reduced the learning rate by a factor of 0.1 every 10 epochs. See Table 3.3 for hyper-parameter details.

On this setup, the full training required approximately 26 minutes. Model checkpoints were saved according to the lowest validation loss, and all experiments were run with a fixed random seed, 42 in this training, to ensure reproducibility. The training process demonstrated stable convergence, with validation loss plateauing after roughly 20 epochs.

All code and scripts used for model training, evaluation, and simulation are publicly available at `https://github.com/liheng423/style-cf`, ensuring full reproducibility of the results.

3.3.2. Loss function

Since the model predicts acceleration while the quantities of interest are the follower's velocity and, at times, its position, the kinematic relations must be incorporated into the loss function. This ensures that both speed and position can be consistently recovered from the predicted accelerations. With ground-truth position and speed available, defining a suitable loss function becomes straightforward. In this study, we focus on the accuracy of the follower's position, which requires two levels of integration from acceleration. As a result, achieving accurate and stable position prediction also depends on the quality of predicted speed and acceleration, especially under longer prediction horizons.

The corresponding loss function is given in Equation 3.1 and the formulation follows Equation A.2:

$$\begin{aligned} \hat{v}_t^{\mathrm{F}} &= v_0^F + \Delta t \sum_{i=1}^{T_{\text{pred}}} \hat{a}_i^F \\ \hat{x}_t^{\mathrm{F}} &= x_0^F + \Delta t \sum_{i=1}^{T_{\text{pred}}} \hat{v}_i^F \\ \text{Loss} &= \frac{1}{T_{\text{pred}}} \sum_{i=1}^{T_{\text{pred}}} \left(\hat{x}_i^F - x_i^F\right)^2 \end{aligned} \tag{3.1}$$

Here, $\hat{x}_t^F$, $\hat{v}_t^F$, and $\hat{a}_t^F$ denote the model predictions of position, speed, and acceleration, respectively, while x_t^F, v_t^F, and a_t^F represent the corresponding ground-truth values recorded in the dataset. Note that the model requires initialization values—x_0^F, v_0^F, and a_0^F—which correspond to the follower's state at the timestep immediately preceding the prediction window.

Namely, the loss function corresponds to the mean squared error (MSE) of the follower's position over the prediction window. With this training approach, physical consistency is preserved throughout the process. As a result, the predicted acceleration remains realistic, free from extreme outliers or jitters artifacts, and aligns more closely with real-world driver behavior.

3.4. Testing: Model Evaluation Framework

To properly evaluate a style-aware car-following (CF) model, the following requirements must be satisfied:

1. **Recursive prediction:** The model performance should be evaluated over a long prediction horizon, relying solely on its own past predictions rather than real-world data inputs. This allows us to observe how the model behaves autonomously without external data guidance.
2. **Independent style evaluation:** The performance of the style embedder should be assessed separately to verify whether it effectively captures and represents the intended driving styles.
3. **Style-informed testing:** During CF model testing, style information must be explicitly provided to the model, enabling an examination of whether incorporating style awareness leads to measurable performance improvements.

This naturally leads to the following sections. First, we introduce the concept of recursive prediction. Next, we present the methodology for evaluating the style embedder. Finally, we describe in detail the design of the testing experiments.

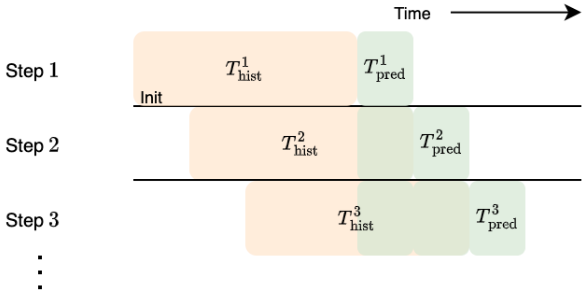


Figure 3.3: Recursive Prediction over time

3.4.1. Car-following Simulation: Recursive Prediction

The Style Transformer model and the corresponding benchmark models (see Section Benchmark models) are short-duration prediction models, meaning they are trained with a prediction horizon of only a few seconds (e.g., style-aware CF predicts 4 seconds in the future). This choice is primarily driven by considerations of computational cost. However, strong performance over a short prediction horizon does not necessarily translate to reliable performance over longer durations, as prediction errors can accumulate over time. Long-duration prediction poses a greater challenge for models trained on short-duration datasets and thus serves as an ideal criterion to evaluate their generalization capability. More importantly, this study aims to investigate the influence of driving styles on congestion formation. The long-duration evaluation is, thus, preferred.

Recursive Prediction

As a single-agent system, recursive prediction is essential for achieving long-duration forecasting. However, since the trained models are primarily optimized for short-duration prediction, a recursive prediction algorithm is developed to decompose the long-duration prediction task into a sequence of short-duration steps.

The overall process is illustrated in Figure 3.3. The long-duration trajectory is predicted recursively, with each iteration advancing the time window by T_{pred} seconds. The algorithm terminates automatically upon reaching the end of the sequence. A key requirement of this recursive framework is the initialization step, where the initial historical segment (the yellow block T^1_{hist}) is filled with ground-truth data, as the model requires a history window to begin prediction.

It is important to note that after the first prediction step, the subsequent input windows (the green blocks) start to contain the model's own predictions instead of ground-truth values, as they overlap with previous outputs. Although the model initially relies on ground-truth input, the prediction horizon is sufficiently long—90 seconds per clip—so the influence of ground-truth data is negligible in the overall sequence in the long term.

Simulation Algorithm

Based on the idea of recursive prediction, this report proposes a car-following simulation algorithm that generates the follower vehicle's trajectory given the leader's movements. The details are presented in Algorithm 1, where the simulation produces the complete trajectory of the follower within the specified time range.

Algorithm 1 Recursive Prediction with Sliding History

```
Inputs:
    groundtruth data $d^L_{[1:T]}, d^F_{[1:T]}$, including position, velocity and acceleration
    model $M$
Params: $T_{\text{hist}}$ (history length), $T_{\text{pred}}$ (prediction horizon), $T$ (length of time series)
Output: Predicted self movements $\hat{y}_{[1:T]}$ (including position, velocity and acceleration)
Initialize a default training movement series $x^F_{[1:T]} \leftarrow$ empty series $O_{[1:T]}$
Initialize first history window $x^F_{[1:T_{\text{hist}}]} \leftarrow d^F_{[1:T_{\text{hist}}]}$
Initialize the start timestep $t_0 \leftarrow T_{\text{hist}}$
for $k = 1$ to $\lfloor (T - t_0)/T_{\text{pred}} \rfloor$ do
    $t \leftarrow t_0 + (k-1)\,T_{\text{pred}}$
    Prepare input $D \leftarrow (d^L_{[t-T_{\text{hist}}:t+T_{\text{pred}}]},\ x^F_{[t-T_{\text{hist}}]})$
    Predict $\hat{y}_{[\,t:\,t+T_{\text{pred}}]} \leftarrow M(D)$
    Append $\hat{y}^{[\,t:\,t+T_{\text{pred}}\,]}$ to $\hat{y}$
    Update history window: $x^F_{[t:t+T_{\text{pred}}]} \leftarrow \hat{y}_{[t:t+T_{\text{pred}}]}$
end for
return $\hat{y}$
```

3.4.2. Testing: Style Embedder

Driving style embedding, or style token (Figure 3.1), is time-independent and captures the driving style after training. Due to its time independence, the style embeddings can be directly clustered in the embedding space.

Although the exact representation of the embeddings is unknown, clustering algorithms can still be applied to differentiate driving segments. In this thesis, the K-means algorithm is employed.

To avoid data leakage, the dataset used for the following analysis is the long-duration dataset (Figure 1.1), which is fully independent from the training and validation data.

Styles Clustering and Visualization

K-means is a widely used unsupervised clustering algorithm that partitions a set of n data points into K non-overlapping clusters. It minimizes within-cluster variance by iteratively updating cluster assignments and centroids. Specifically, each data point is assigned to its nearest cluster center, which is subsequently updated as the mean of the points within the cluster. This process repeats until convergence, typically when cluster assignments stabilize or the objective function no longer changes.

K-means clustering is conducted on the predefined embedding space of driving-style segments. The clustering operates directly on the learned embeddings. The number of clusters is selected based on the elbow method, and the quality of the resulting partition is assessed using the Silhouette Score, which measures both intra-cluster cohesion and inter-cluster separation, and further reveals the distribution of driving styles.

To visualize the clustering structure, the high-dimensional embedding representations are projected onto a 2D space using Principal Component Analysis (PCA). While the resulting plot provides an intuitive overview of cluster distribution, it may not fully reflect separability due to possible non-linear

Symbol	Quantity	Related human factors
$\bar{h}$	Average time headway	Reflects aggressiveness and possibly driving proficiency
$\bar{\tau}$	Average reaction time	Associated with aggressiveness, driving proficiency, and temporal anticipation
σ_{τ}	Standard deviation of reaction time	Related to driving proficiency and distraction, as variability may indicate inconsistent behavior over time
σ_{h}	Standard deviation of time headway	Also associated with driving proficiency and distraction, similar to σ_{τ}
l^{F}	Length of follower vehicle	Truck drivers exhibit different behavior compared to passenger car drivers (Ossen and Hoogendoorn, 2011)
l^{L}	Length of leader vehicle	Driver behavior may differ when following a truck rather than a passenger car

Table 3.1: Driving style-related car-following quantities

structures in the original space. The visualization thus serves as a qualitative reference for assessing the clustering results. Note, PCA is only used for visualization, and has no impact on any downstream tasks.

Style Interpretation and Validation

After visualizing the clustering results, it is essential to interpret the characteristics of each group. Accordingly, Table 3.1 presents the style-related indicators. These features are designed based on the human factors in car-following (CF), as listed in Table 2.2, because they are informative for uncovering the underlying driving styles reflected in the trajectories. These indicators thus serve as the foundation for interpreting the behavioral characteristics of each cluster.

Furthermore, after identifying their characteristics, we need to validate that the differences between groups indeed exist in terms of driving style, rather than merely in the embedding space—which could be misleading or influenced by noise. To this end, we design the token-replacement experiments: that is, we replace the tokens of a sample with the style tokens (and vehicle type tokens) from other groups, then observe the resulting behavioral changes and compare the effects. The details are provided in Section Truck as Follower and Section Cross-over Test case: Effectiveness of the embedding.

To illustrate this more intuitively, we further provide two examples showing that the observed differences originate from genuine variations in driving style, rather than superficial disparities in the embedding space. The details are provided in Section Example: Style Counterfactual Test.

3.4.3. Incorporating Style in Testing

Different models require different input structures, and to ensure a fair comparison, the following setup decisions were made:

Non-style-aware Testing: Due to the limited size of the testing set and the fact that LSTM

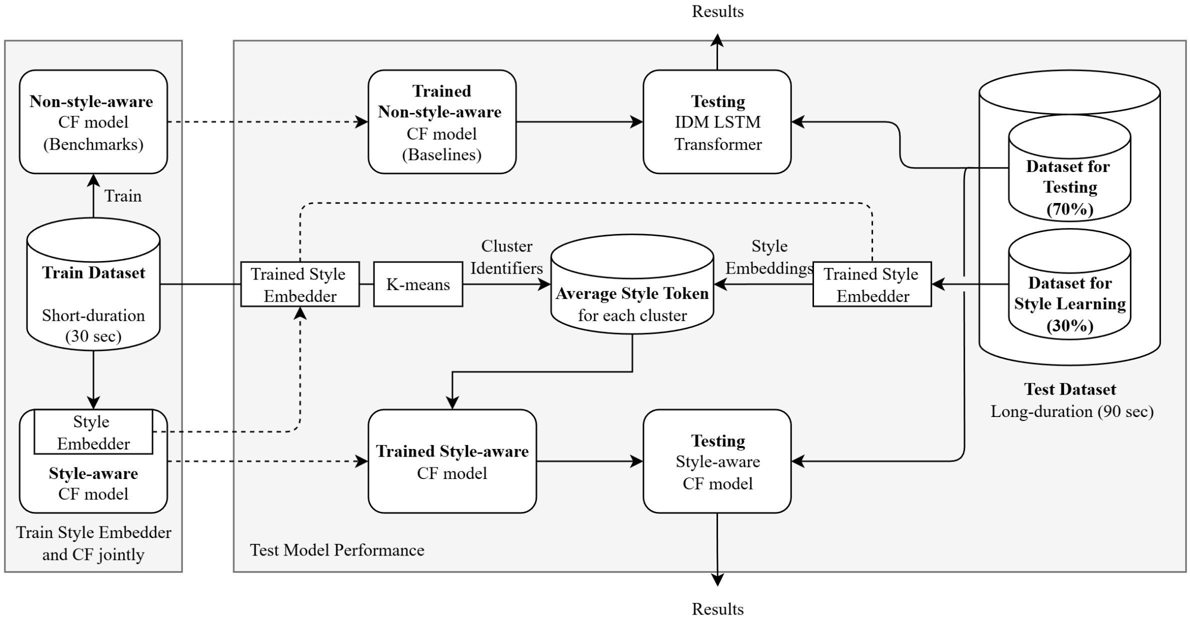

Results
Non-style-aware CF model (Benchmarks)
Train
Train Dataset
Short-duration (30 sec)
Style Embedder
Style-aware CF model
Train Style Embedder and CF jointly
Trained Non-style-aware CF model (Baselines)
Testing IDM LSTM Transformer
Trained Style Embedder
K-means
Cluster Identifiers
Average Style Token for each cluster
Style Embeddings
Trained Style Embedder
Dataset for Testing (70%)
Dataset for Style Learning (30%)
Test Dataset
Long-duration (90 sec)
Trained Style-aware CF model
Testing Style-aware CF model
Test Model Performance
Results

Figure 3.4: Training and testing data handling involving style information

and plain Transformer models do not incorporate explicit style information, we ensure a comparable setting by using average-style representations. Specifically, the IDM model is calibrated using the mean parameter values across all trajectories in the training set, as described in Table 3.2. The process of training and testing dataset handling is illustrated in Figure 3.4.

Incorporating Style-specific embeddings into Testing: For the Style Transformer, however, group-specific average embeddings are employed, as illustrated in Figure 3.4. The testing dataset is divided into two subsets: 30% of the data is used for learning driving styles. Since these trajectories span 90 seconds, they provide more stable and informative representations for style learning. To avoid bias when computing the average style token for each group, 30% of the data from each group is evenly sampled. Meanwhile, clustering is performed using the trained style embedder and the k-means algorithm based on the training dataset, which offers group IDs for each vehicle. Finally, the mean style embedding for each group (derived from the training set) is calculated and directly provided to the style-aware CF model, skipping the embedder stage. In this setup, the tokens obtained from training act as global style priors—their group identities, rather than the tokens themselves, are used in subsequent steps.

During testing, the tokens are re-estimated from the Style-Learning subset (30% of the test data) under fixed cluster assignments to achieve stable style representation, while avoiding any data leakage from the testing.

Note, the training dataset covers all vehicle IDs present in the test data, thereby preventing situations where the driving style of a specific trajectory remains unidentified in the clustering process based on training dataset. Further details regarding model inputs and outputs can be found in Table 3.3.

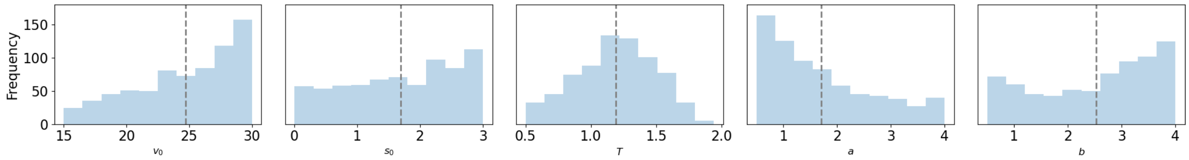


Figure 3.5: Calibrated parameter distribution of IDM

Parameter	v_0^F	Δx_0	h_0	$a_{\max}$	b
Zen-data Calibrated Values	24.70 m/s	1.70 m	1.19 s	1.70 m/s^2	2.53 m/s^2

Table 3.2: Calibrated IDM parameters on Zen-traffic Data

3.4.4. Benchmark models

Three models are considered in this study: the rule-based Intelligent Driver Model (IDM), a Long Short-duration Memory (LSTM) network, and a Transformer-based model.

IDM

The IDM serves (Treiber et al., 2000) as a baseline, simulating acceleration based on current velocity, spacing, and leader speed. It is interpretable, grounded in domain knowledge, and its five parameters directly reflect driving style, but it only supports one-step prediction. For details, see Appendix Intelligent Driver Model.

To avoid using an uncalibrated IDM and drawing biased conclusions about the proposed model, the IDM is calibrated using the same training dataset.

Calibration is performed per car-following (CF) pair using the genetic algorithm (Mitchell, 1996), meaning the algorithm searches for the best-fitting parameter set for each CF pair individually. The detailed procedure of the genetic algorithm is omitted here, as it falls outside the scope of this study.

This calibration yields a distribution of IDM parameters across all CF pairs, as shown in Figure 3.5. It can be roughly observed that the parameters calibrated through IDM are scattered, indicating that driving behaviors are heterogeneous in this dataset. Since IDM serves as the benchmark model for comparison with the model proposed in this study, we use the average parameter values to represent an idealized average driver.

To ensure a fair cross-model comparison, the IDM is constrained to adopt uniform parameters across all CF pairs, as is also the case for the other benchmark models. Instead, the average values from the parameter distribution are adopted as the final IDM setting. This calibrated IDM will serve as the benchmark model in the following analysis.

In total, 1000 trajectories are used for calibration, which are randomly sampled from the training dataset. Since the dataset is large and only the average parameter setting is required, calibration is performed on this subset of trajectories.

LSTM

The LSTM model (Hochreiter and Schmidhuber, 1997) uses a 6-second historical window with features including vehicle velocities, spacing, and lengths to account for heterogeneous vehicle types. In this thesis, the hidden states are used as outputs and further compressed by a linear layer into a one-dimensional time series representing the predicted acceleration. However, the model cannot access future leader behavior due to the sequential structure, limiting its long-duration prediction ability. For details, see Appendix LSTM Model and the architecture can be found in Table A.3.

Transformer

The Transformer model (Vaswani et al., 2017) extends this by incorporating future leader behavior through a label window while masking follower movements to prevent data leakage. It enables longer prediction horizons and better handles long-range dependencies, though it requires careful input segmentation and masking design. However, it doesn't take driving style into consideration in its architecture and is unable to offer any insights into driving style. For details, see Appendix Transformer Model, and the architecture can be found in Table A.2.

3.4.5. Overview of Models

An overview of the hyperparameters, along with the model inputs and outputs, is provided in Table 3.3.

<table>
<tr><th>Model
Parameter</th><th>LSTM</th><th>Transformer</th><th>Style-aware CF</th><th>IDM</th></tr>
<tr><td colspan="5"><b>Hyper-parameter Setting</b></td></tr>
<tr><td>Learning rate</td><td>0.001</td><td>0.001</td><td>0.001</td><td>-</td></tr>
<tr><td>Batch size</td><td>64</td><td>64</td><td>64</td><td>-</td></tr>
<tr><td>Epochs</td><td>20</td><td>30</td><td>30</td><td>-</td></tr>
<tr><td>Dropout</td><td>0.0</td><td>0.1</td><td>0.1</td><td>-</td></tr>
<tr><td>Activation Function</td><td>Sigmoid</td><td>ReLU</td><td>ReLU</td><td>-</td></tr>
<tr><td>Maximum Norm Value</td><td>10.0</td><td>10.0</td><td>10.0</td><td>-</td></tr>
<tr><td>Optimizer</td><td>Adam</td><td>Adam</td><td>Adam</td><td>-</td></tr>
<tr><td colspan="5"><b>Data Segmentation Setting</b> (seconds)</td></tr>
<tr><td>Prediction window (T_{pred})</td><td>2</td><td>4</td><td>4</td><td>0.1</td></tr>
<tr><td>History window (T_{hist})</td><td>6</td><td>6</td><td>6</td><td>0.1</td></tr>
<tr><td>Label window (T_{label})</td><td>-</td><td>4 + 4</td><td>4 + 4</td><td>-</td></tr>
<tr><td>Style window (T_{hist})</td><td>-</td><td>-</td><td>6</td><td>-</td></tr>
<tr><td colspan="5"><b>Features</b></td></tr>
<tr><td>History window(input)</td><td colspan="3">$v^F, \Delta v, \Delta x, l^F, l^L$</td><td>$v^F, v^L, \Delta x$</td></tr>
<tr><td>History Label window (input)</td><td>-</td><td colspan="2">v^L, v^F</td><td>-</td></tr>
<tr><td>Future Label window (input)</td><td>-</td><td colspan="2">$v^L, \langle v^F \rangle_{T_{\text{hist}}}$</td><td>-</td></tr>
<tr><td>Style Series (input)</td><td>-</td><td>-</td><td>$(\tau, h, \Delta x, v^F)$</td><td>-</td></tr>
<tr><td>Prediction window (output)</td><td>a^F</td><td>a^F</td><td>a^F</td><td>a^F</td></tr>
</table>

Table 3.3: Detailed parameters in training process ($\langle v^F \rangle_{T_{\text{hist}}}$ indicates the average value of v^F in the history window)

Note that IDM requires the parameters as well. The parameters are shown in the Table 3.2.

4

Field Data

This thesis seeks to examine the mechanisms underlying congestion formation as influenced by the interaction between various driving styles. Therefore, a robustly trained, style-specific data-driven car-following model is essential as the foundation of this investigation. This necessitates the use of high-caliber training data. The current chapter will elucidate the methodology for obtaining a reliable dataset.

4.1. Data Requirements and Source

4.1.1. Data Requirements

To guarantee that the data fully meets the requirements for subsequent tasks, two key requirements must be satisfied:

- **Consistency:** Accurate position and speed must be reliably derived from the acceleration values, ensuring that the trajectories are kinematically plausible.
- **Car-following:** All leader–follower pairs must genuinely be engaged in a car-following relationship, such that the follower's behavior reflects a response to the leader's actions.

These two requirements are critical for training and testing a car-following model. With acceleration as the output and position-based loss (assuming a kinematic model inside the loss function), consistency is essential to avoid invalid supervision. Likewise, genuine car-following interactions must be ensured; otherwise, if the leader and follower are effectively uncoupled, the model cannot learn meaningful behavior.

4.1.2. Data Source

The Zen traffic dataset contains vehicle trajectory data collected on Osaka motorways, covering a range of traffic states from free-flow to congestion (Figure B.1 and Figure B.2). This diversity enables comprehensive training data for car-following models. The dataset also includes lane-change events and road layouts, supporting future research.

However, inconsistencies exist (Section Processing: Data Consistency), and real-world motorway factors—such as lane changes and off-ramp entries—may interfere with pure car-following dynamics. Thus, rigorous filtering is essential to isolate relevant trajectories.

The dataset spans three routes in Osaka: Route 4 (5 scenes), Route 11 (5 scenes), and Route 13 (6 scenes). While each route varies in geometry, all feature merging sections and diverse traffic conditions, making them suitable for studying car-following behavior across traffic regimes.

Data from a section of Route 13 using the Moving Light Guide System (MLGS) is excluded to avoid bias. Road slopes and detailed geometries are beyond this study's scope.

Finally, The Zen dataset provides detailed road geometry and ramp locations, enabling the exclusion of trajectories in weaving areas where merging behavior introduces uncertainty. Lane ID information, which reflects lane-changing behavior, is available but not utilized in this study.

4.2. CF-related Trajectories Extraction (Segmentation)

Table 4.1: Filtering for extracting valid car-following data

Filters	**Description**
No weaving area	Trajectories that pass through weaving areas are excluded.
No lane change in follower	The follower in a car-following pair must not change lanes during the selected time window.
No MLGS system	Trajectories passing through the MLGS area while the system is enabled are excluded.

To prepare uniform-length samples for model training, the remaining trajectories are segmented after initial filtering. Raw trajectories often include disruptions from lane changes or other maneuvers (Table 4.1), disrupting car-following continuity. Varying durations of the vehicle trajectories also hinder the unified data format, which is ideal for training.

To address this, we extract fixed-duration segments with either fixed or randomized step sizes to balance sample diversity and redundancy. Each segment includes a time series of both target and leader vehicles and is filtered for quality.

As the output of the segmentation algorithm, the extracted dataset format is shown as Table B.1. Note that, since the raw dataset doesn't comprise the acceleration data, and considering the Kalman filtering will be applied in the next step, the acceleration will be introduced and derived using the Figure A.1.

Since the Zen-traffic trajectory data is collected at different times and date, there are multiple trajectory dataset on one route. Eventually, this data is concatenated into a single dataset. Eventually, a substantial number of valid CF trajectory pairs were collected. In total, 3 different datasets are extracted, a 30-second one is later used for the training and validation process (see Section Model Training), and a 90-second (long-duration) one is applied in testing (see Section Car-following Simulation: Recursive Prediction). The format of the extracted data is shown in Table B.1.

4.3. Segmented Car-following Data Overview

Various factors can affect data quality during collection, particularly for trajectories extracted from video footage. To assess the reliability of the dataset, the

Speed and Acceleration Profile

Speed distribution is an indicator to examine the traffic states in the dataset. Ideally, the dataset is required to include various traffic states, such as free-flow and congestion.

As shown in Figure 4.1, most vehicles operate at relatively low speeds, with an average speed of 46.8 km/h. This observation is consistent with the speed contour plots presented in Figure B.1 and Figure B.2, which indicates that congestion frequently occurs on the studied road segment. The alignment between the speed distribution and the observed traffic conditions supports the conclusion that the dataset captures a broad spectrum of traffic states, ranging from congested low-speed flows to free-flow high-speed conditions. Such variability is beneficial for model training, as it enables learning across diverse traffic scenarios.

Acceleration is not directly available in the raw dataset, which contains only vehicle positions and velocities. Consequently, acceleration must be derived from the recorded velocity profiles using forward differentiation (see Figure A.1). Given that the data originates from video footage, it is likely that position is the only directly observed variable, while velocity is computed through additional temporal processing. As a result, the derived velocity—and subsequently the acceleration—may be affected by noise. Therefore, analyzing the distribution of acceleration, particularly identifying abnormal or extreme values, is essential for assessing the quality and reliability of the dataset.

As shown in Figure 4.1, the acceleration values fall within a reasonable range, indicating that the dataset does not suffer from severe issues with abnormal values—an observation consistent with empirical traffic studies that report typical acceleration values within (-8, 5) m/s^2 (Montanino and Punzo, 2015).

Consistency

However, while data-driven models typically output acceleration, model validation is usually performed using position and velocity. This requires the dataset to maintain kinematic consistency to ensure reliable derivation of these variables. Thus, kinematic inconsistency between acceleration, velocity, and position, which can hinder the training process, remains a concern. To address this, Section Processing: Data Consistency will introduce a method for improving consistency to an acceptable range.

Car-following or not: trajectory examples

To validate the extracted dataset, two representative trajectories (Figure 4.2) were selected for detailed examination to identify and analyze potential issues within the current data.

The left panel clearly demonstrates a car-following behavior, characterized by a noticeable lag in the follower's velocity relative to the leader's, indicating that the follower is actively responding to and

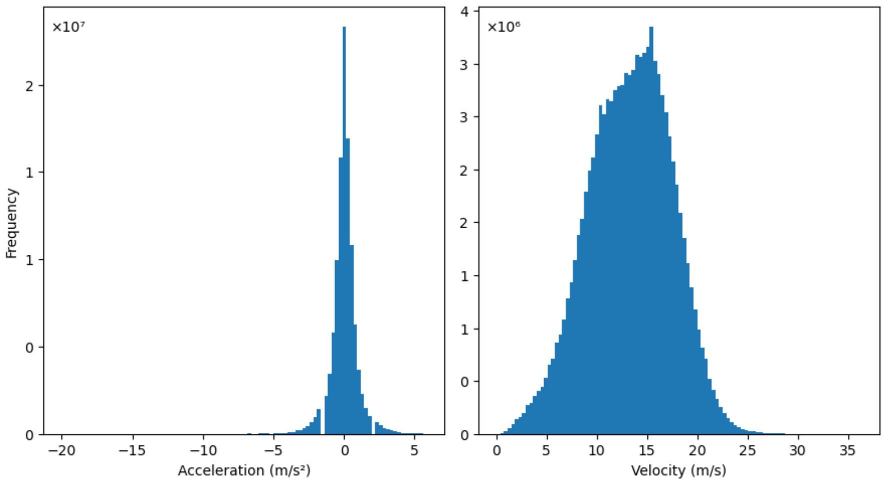


Figure 4.1: Acceleration and speed distribution (both leader and follower)

mimicking the leader's actions. The inter-vehicle gap remains within a reasonable range, further supporting this interpretation. The calculated time headway for the follower is approximately 1.2 seconds, which aligns well with typical car-following conditions.

Conversely, the right panel reveals a scenario where such a car-following relationship is absent. The follower's velocity shows little correlation with the leader's behavior, and the gap between the vehicles steadily increases over time. This pattern suggests external influences on the follower, such as interference from lane-changing vehicles or an impending lane change due to proximity to an off-ramp. The follower's time headway in this case is significantly larger, approximately 2.89 seconds, reinforcing the conclusion that the vehicles are not engaged in car-following.

These contrasting cases underscore the importance of filtering the dataset to exclude trajectories where genuine car-following is not observed. Such filtering is essential to ensure data quality and improve the robustness of subsequent modeling and analysis efforts.

Based on these observations, the vehicle pairs that are not in the CF, and the trajectories with abnormal acceleration values are filtered out in Table 4.3.

4.4. Processing: Data Consistency

In this study, consistency refers to the error between derived values and recorded values of lower-order traffic variables like speed and position.

The raw dataset contains recorded values for both velocity and position per vehicle per timestep. However, it's unclear whether the consistency between them is maintained throughout each trajectory. In other words, the recorded velocity might not match the position data. While this issue isn't serious over short time periods, it becomes problematic for long-duration prediction as errors accumulate over time. Eventually, the data can become significantly distorted since it accumulates all errors from the beginning.

To effectively measure the extent of the inconsistency, the absolute deviation between the recorded value and the value derived from the kinematic model in Equation 4.1 is chosen as the inconsistency

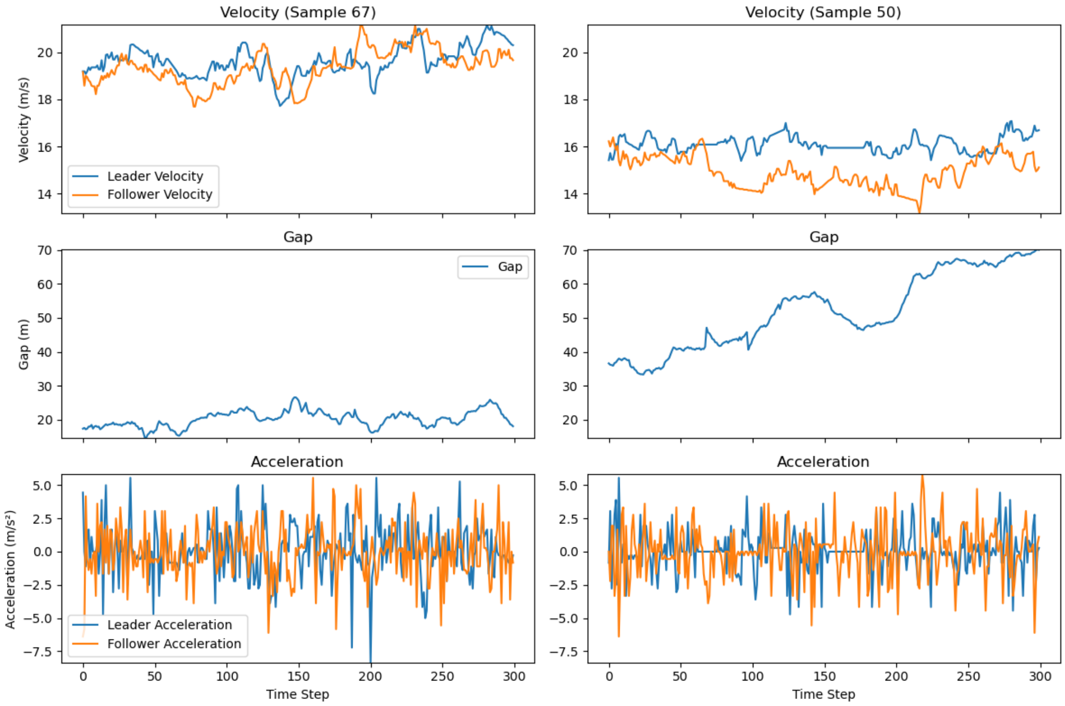


Figure 4.2: Trajectory examples in raw dataset

error. In the expanded form,

$$\text{error}_t = x_t - x_0 - \Delta t \sum_{i=1}^{t}(v_0 + \Delta t \sum_{j=1}^{i} a_j) \tag{4.1}$$

the error can be calculated at every timestep using Equation 4.1. Instead of using the average error, the errors in the latter timestep are more informative than the beginning since they accumulate over time. The error at the very last timestep is adopted as the inconsistency error.

As is shown in the Figure 4.4, the orange distribution refers to the position error in the raw dataset. The error is normally distributed, with the maximum value reaching 30 meters. This implies the recorded speed and the position data are not consistent, and in some samples this could hinder the training by implicating unnecessary inconsistency error.

4.4.1. Kalman Filter Setup

To mitigate this effect while preserving both speed and position information, we employ Kalman filtering embedded with a 1-D and first-order kinematic model (Equation 4.2) to enforce kinematic consistency between these variables. See section A.2 for details of the Kalman Filtering model and algorithm involved in this study.

$$\begin{bmatrix} x_t^{KF} \\ v_t^{KF} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ v_{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} a_{t-1} \tag{4.2}$$

In Equation 4.2, x_t^{KF} and v_t^{KF} denote the filtered position and velocity at timestep t, respectively.

Symbol	Value	Description
P_0	$\begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$	State covariance matrix: we assume the noise in position doesn't affect the noise in velocity.
R	$\begin{bmatrix} 10 & 0 \\ 0 & 100 \end{bmatrix}$	Measurement noise covariance matrix: it's assumed the position is more accurate than velocity since the data is collected in video footage and the position is directly accessible.
Q	$\begin{bmatrix} s_{\text{pos}}^2 & 0 \\ 0 & s_{\text{speed}}^2 \end{bmatrix}$	Process noise covariance: $s_{\text{pos}} = 0.5\, a_{\max} \Delta t^2$, $s_{\text{speed}} = a_{\max} \Delta t$, modeling unmodeled acceleration.

Table 4.2: Parameter setting of the Kalman filter (units on the diagonals: position [m^2], velocity [$(\text{m/s})^2$])

Δt is the discrete time step, and a_{t-1} is the measured acceleration at the previous step, used as an external input to update the velocity.

To set up the Kalman filter, the parameters relevant to the uncertainty must be determined beforehand. Based on the experience and knowledge of the dataset, the parameters are shown in Table 4.2 along with the reasons behind them. The parameters are tuned to ensure the trends in speed and position curves match the raw data as well.

4.4.2. Results

Following the indicated parameters, three example trajectories (Figure 4.3) are selected to be presented in the thesis. Note that the Kalman filter doesn't involve the acceleration as a part of the state. The accelerations in the following analysis are derived from the filtered speed.

After the filtering, acceleration becomes stable with most of the jitter smoothed. Meanwhile, the trends in speed and spacing are well-preserved, ensuring the training data is not overly manipulated. It can be observed that some peaks and dents appear to occur earlier or later. This could be the result of fixing the inconsistency error.

Figure 4.4 illustrates the results of the enhanced consistency. The filtering successfully narrows the position error distribution down, from around 30 to less than 15 on the edge. Nevertheless, some inconsistencies remain, as a balance between genuineness and consistency must be maintained: for data points that cannot be reliably corrected, we set their error at the threshold level and subsequently discard them.

However, some trajectories cannot be effectively corrected by Kalman filtering, even though they appear normal in the raw dataset. These trajectories are severely affected by consistency issues. When Kalman filtering is applied, it introduces noticeable jitter into the processed data rather than improving it (Figure 4.5). To prevent such artificial disturbances in the filtered dataset, an acceleration range filter is introduced accordingly.

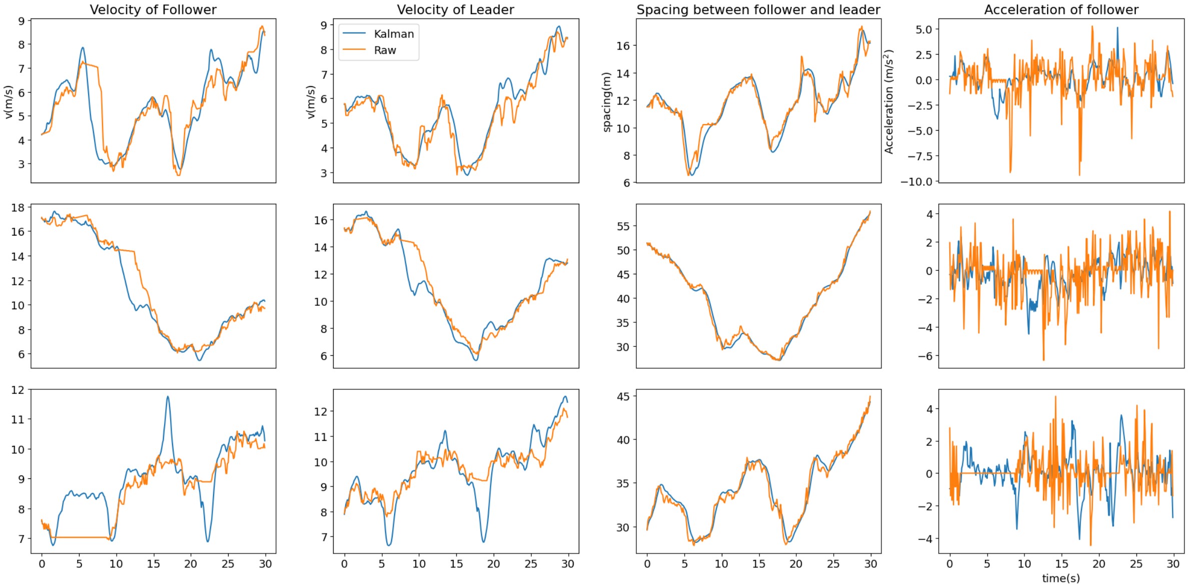


Figure 4.3: Examples of filtered trajectories (Orange: Raw, Blue: Kalman)

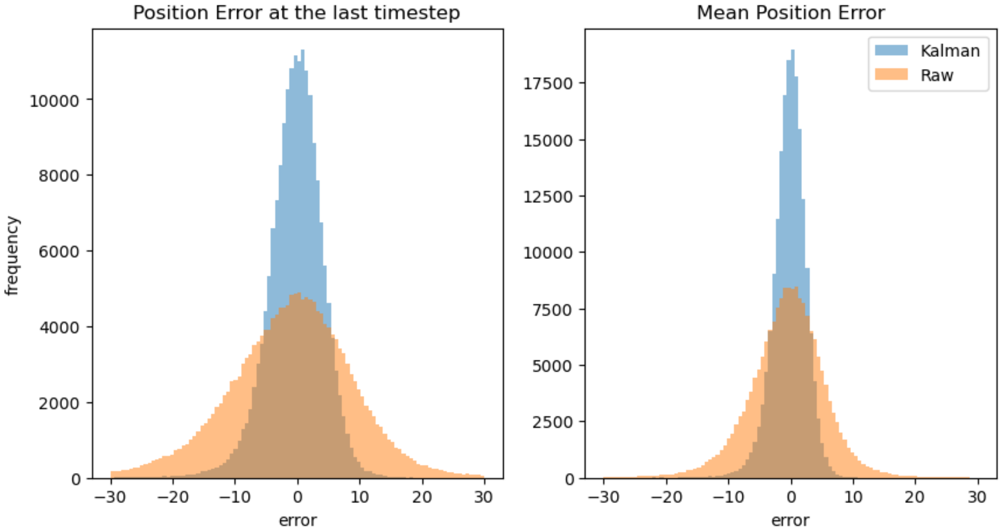


Figure 4.4: position inconsistency error before and after the Kalman filtering (error unit: [m])

Filter	Range	Description
In CF relationship	$h \in (0,2)$ s	Average time headway (Equation 4.3) of the full trajectory
position inconsistency error	error $\in (0,2)$ m	Position inconsistency error (Equation 4.1) at the last timestep
abnormal jitters	$a \in (-10,6)$ m/s^2	any trajectory includes an acceleration outside the range (i.e., abnormal jitters), the trajectory shall be discarded

Table 4.3: filters applied for refining dataset

4.4.3. Summary

In section Segmented Car-following Data Overview, the rough data overview reveals that the car-following data might include pairs that don't in fact have a CF relationship, which means the follower doesn't react to the behavior of the leader. Based on the intermediate conclusions we drew from that section, we set up the following filters to ensure the data only contains the pairs with a strong relationship.

Apart from the mentioned pre-processing filtering (Table 4.1), Table 4.3 shows the filters that are set up to ensure the training data contains meaningful car-following clips.

The average time headway filter is to ensure the follower and leader are, in fact, in the car-following relationship. Worth mentioning, when the follower vehicle runs at a relatively low speed, the time headway could become extremely large; therefore, the time headway h is calculated as Equation 4.3.

$$h = \frac{\Delta x}{\max(v_{\text{Follower}}, 0.1)} \tag{4.3}$$

Position inconsistency errors, as previously mentioned, are useful for identifying trajectories with significant noise that cannot be effectively corrected by the Kalman filtering process. Rather than attempting further repair, discarding such trajectories is a more rational choice.

In the Kalman filtering procedure, based on the chosen parameter settings, the kinematic model is given greater weight than the measurements. As the raw data often suffers from position inconsistency, the Kalman filter attempts to reconcile these inconsistencies, which can result in abnormal jitters in the filtered trajectories. These fluctuations do not reflect realistic driving behavior, as the resulting acceleration may reach up to 8 m/s^2 or more (see Figure 4.5). Although these jitters do not appear to directly affect the velocity, they are used as ground truth in downstream tasks, which negatively impacts model training. Therefore, trajectories with such poorly filtered noise are excluded from further analysis in the dataset.

4.5. Processed Dataset

In addition to ensuring data quality through Kalman filtering and pre-processing, it is necessary to examine the datasets used for model training and evaluation. Two datasets are introduced in this section: a *short-duration* dataset with a 30-second window and a *long-duration* dataset with a 90-second

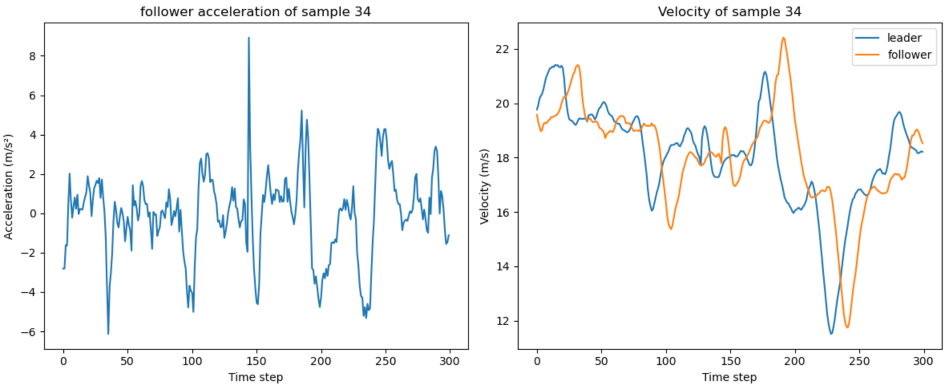


Figure 4.5: Jitters in acceleration

window.

The 30-second dataset is primarily designed to facilitate efficient model training by reducing computational cost and increasing the number of usable samples. Shorter time windows help minimize the impact of lane-changing and other non–car-following behaviors, thereby ensuring that the training data primarily reflects stable following interactions.

In contrast, the 90-second dataset is used to evaluate model performance over longer prediction horizons and to enhance the learning of driving styles. Longer sequences allow the model to capture more persistent behavioral patterns and temporal dependencies, which are essential for style embedding and long-term trajectory prediction experiments.

Note that there is no temporal overlap between the two datasets for a certain CF pair to prevent data leakage. However, overlapping in vehicle IDs is acceptable—and even beneficial—since the model can later recognize and transfer the same drivers' styles learned from the training dataset to the testing dataset, allowing consistent application of learned style patterns across different temporal contexts.

Lastly, two major components in dataset require further examination to ensure the successful execution of the downstream tasks.

Vehicle type can substantially influence car-following (CF) behavior, as trucks and passenger cars differ in length, weight, and acceleration capability. Drivers tend to maintain larger time headways and lower speeds when following trucks, reflecting cautious behavior due to reduced visibility and slower vehicle responses. To distinguish between them, a vehicle length threshold of 7 meters is applied, following common classification practices.

In addition to vehicle type, three indicators—time headway, acceleration, and speed—are analyzed to describe following dynamics. Their distributions reveal how closely a follower reacts to its leader and how different vehicle types contribute to behavioral variability. Examining these distributions helps verify whether the data-driven model captures both average tendencies and variations observed in real traffic.

Follower Leader	**Truck**	**Passenger Car**	Total
Truck	1786	7944	9730
Passenger Car	4245	30232	34477
Total	6031	38176	44207

Table 4.4: number of vehicle type in the short-duration (train) dataset

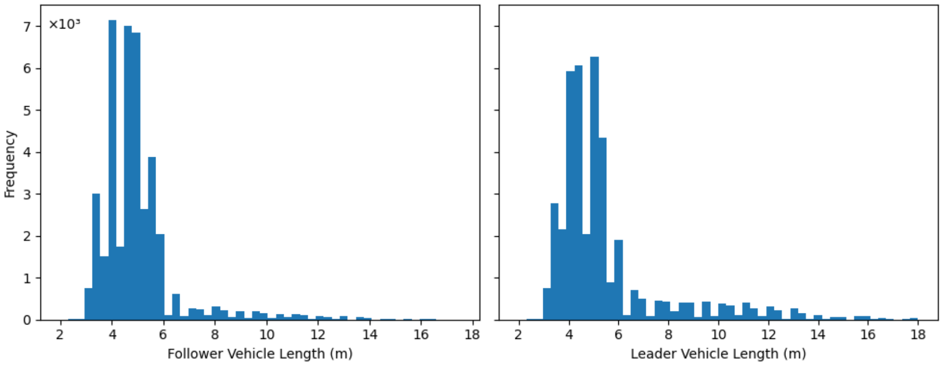


Figure 4.6: Vehicle Length in filtered data

4.5.1. Training Dataset (Short-duration in 30 seconds)

Vehicle Type

The Table 4.4 and Figure 4.6 present the vehicle profile in the filtered dataset. It is evident that passenger cars constitute the majority of the data, but a substantial number of trucks are also present. This diversity helps ensure that the dataset reflects a wider range of driving behaviors and vehicle dynamics, which is beneficial for developing more robust and generalizable models.

Indicators

After Kalman filtering, the overall distribution of average time headway remains largely unchanged (see Figure 4.7, 2)). As discussed in Section Car-following or not: trajectory examples, a large time headway may suggest that the vehicle pair is not engaged in a car-following (CF) relationship. At the same time, since the model requires a sufficient amount of training data, we adopt a threshold of 2.0 seconds to

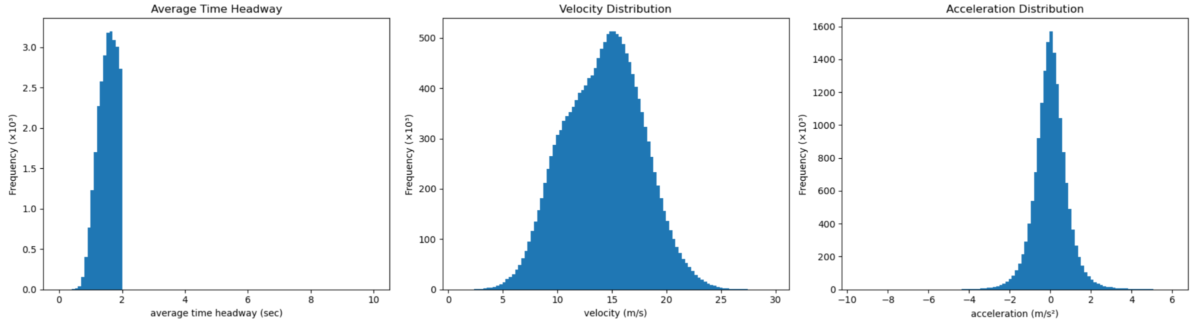


Figure 4.7: Distribution of average time headway, velocity and acceleration per sample (both leader and follower) in training and validation dataset

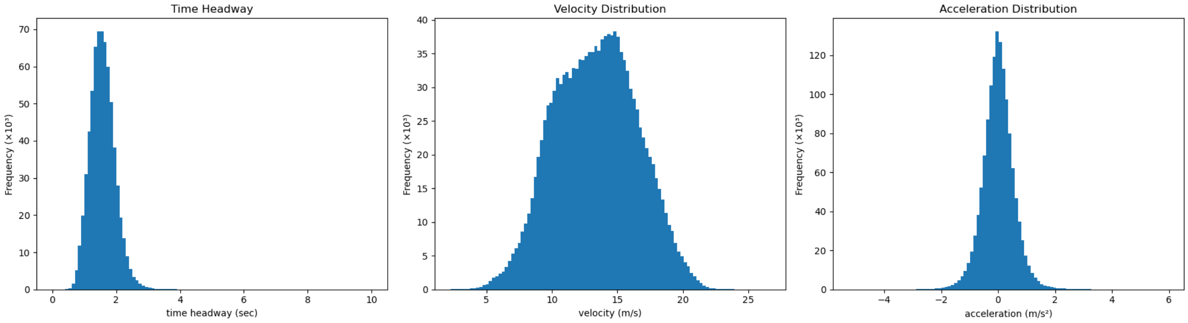


Figure 4.8: Distribution of time headway, velocity, and acceleration per sample (both leader and follower) in testing data

distinguish between CF and non-CF pairs. Specifically, if the time headway exceeds 2.0 seconds, the corresponding trajectories are excluded from further analysis.

Similarly, to ensure that the Kalman filter does not introduce additional noise into the speed and acceleration signals, these two profiles are presented in Figure 4.7. Note that, unlike average time headway, these two distributions are not aggregated over the time dimension. As shown, both speed and acceleration remain within reasonable ranges, consistent with the observations reported in Montanino and Punzo (2015).

4.5.2. Testing Dataset (Long-duration in 90-second)

Follower Leader	**Truck**	**Passenger Car**	Total
Truck	84	686	770
Passenger Car	194	3409	3603
Total	278	4095	4373

Table 4.5: Number of vehicle types in the filtered long-duration (test) dataset

Vehicle Type

While the vehicle type distribution largely matches that of the training set (Table 4.5), there are very few samples with trucks as follower vehicles in the test set. This is likely due to the limited presence of trucks and the short 90-second trajectory duration, during which trucks may be frequently overtaken and thus lack a stable leader. Nevertheless, in the style embedding experiments, truck behavior can still be incorporated by explicitly modifying the model input to reflect truck-related features.

Indicators

As shown in Figure 4.8, the test dataset covers a broad range of traffic states—from low-speed congestion to high-speed free flow—similar to the training data. This consistency helps avoid bias in the final evaluation.

4.6. Conclusion

This chapter introduces two main issues identified in the dataset: data inconsistency and the presence of non-car-following (non-CF) relationships. Data inconsistency introduces additional noise into the deep learning model, which integrates a kinematic model within its loss functions. Such noise significantly hinders the training process. Furthermore, follower-leader pairs without a car-following relationship represent cases where the follower vehicle is effectively in free flow; thus, the follower's behavior cannot be reliably predicted based on the leader's maneuvers.

While applying the Kalman filter to mitigate inconsistency, abnormal jitters appeared in the filtered data. This is likely caused by the conflict between the noisy raw data and the kinematic model embedded in the filter attempting to suppress this noise.

To identify and remove non-CF pairs, the average time headway was chosen as an indicator, with a threshold of 2.0 seconds used to distinguish between CF and non-CF pairs. Concurrently, jittering was further reduced by filtering out trajectories with abnormal acceleration values. The data handling procedure is also illustrated in Figure 1.1.

The resulting dataset is now cleaned and consistent, suitable for direct input into model training. The following chapter will focus on driving style, which is a core aspect of the model and plays a significant role in traffic congestion—the primary objective of this study.

5

Embedding and CF Model Evaluation

Based on the trained CF model and the style-embedder, this section evaluates two aspects: (i) the quality of the clustering and the influence of the extracted style embeddings, and (ii) the joint performance of the CF model with the style embedder to assess how these embeddings enhance prediction accuracy.

5.1. Style Embedding Results

This section introduces style embedding and its clustering results, followed by their impact on the style-CF model. Style embeddings, extracted from a longer testing dataset (90 seconds), offer more information and traffic states than the training dataset (30 seconds), forming the basis for PART II analysis.

Truck-led car-following pairs are excluded from truck embedding extraction and clustering because they are rare in the dataset, with only 770 occurrences out of 3603 in the test dataset, see Table 4.5. Additionally, when driving behind a truck, it becomes more challenging to discern the follower's driving style, as the behavior is largely influenced by the truck. Therefore, this omission improves precision in analyzing followers' behavior.

5.1.1. Clustering Results

The results in Figure 5.1 show relatively well-separated clusters. P2 and P3, which may reflect greater variability in behavior. P1 and P4 appear compact and contain the most samples, suggesting they might capture a common and stable driving style.

These results reinforce the view that driving style in the embedding space is not dichotomous but continuous, consistent with Sagberg et al. (2015). The smooth transitions and overlapping boundaries suggest that driver behavior is better understood as a spectrum, rather than a set of discrete categories, such as merely aggressive vs. timid.

Meanwhile, Figure 5.2 illustrates the cluster composition of follower vehicle types. The left bar

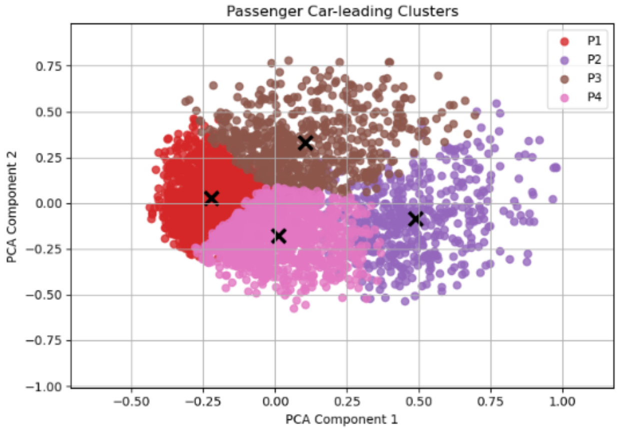


Figure 5.1: Clustering of the driving style embedding in reduced dimensions (Silhouette Score: 0.1878 in embedding space)

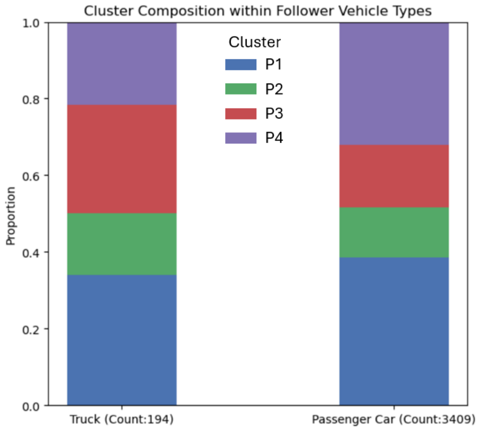


Figure 5.2: Share of styles in each group

Table 5.1: Cluster-wise summary statistics (PC: Passenger Car)

	Cluster P1	Cluster P2	Cluster P3	Cluster P4
$\bar{\tau}$	1.77 ± 0.39	2.00 ± 0.42	2.18 ± 0.40	1.71 ± 0.40
$\bar{h}$	1.31 ± 0.33	1.43 ± 0.28	1.65 ± 0.22	1.12 ± 0.30
$\Delta\bar{x}$	15.11 ± 4.87	19.73 ± 5.30	21.14 ± 5.06	14.63 ± 4.66
$\bar{v}^F$	11.68 ± 2.69	13.68 ± 2.75	12.76 ± 2.68	13.14 ± 2.84
σ_τ	1.01 ± 0.19	1.14 ± 0.17	1.14 ± 0.15	1.04 ± 0.20
σ_h	0.28 ± 0.12	0.40 ± 0.44	0.40 ± 0.17	0.30 ± 0.09
Truck Count	66	31	55	42
PC Count	1315	446	554	1094

represents the proportion of trucks classified into each cluster, whereas the right bar corresponds to passenger cars. For trucks, the four clusters are distributed more evenly. Although $P1$ still accounts for a notable proportion, $P3$ appears to take a higher share compared to $P3$ in passenger cars.

In contrast, most passenger car samples are concentrated in $P1$ and $P4$. This suggests that passenger car drivers predominantly exhibit $P1/P4$-type driving styles. It should be noted, however, that the number of trucks is considerably smaller than that of passenger cars, which may influence the observed proportions.

5.1.2. Interpretation: Cluster Profile

After obtaining the clusters, a tentative interpretation is provided. Based on our explanation for traffic variables and their relationships to human factors (see Table 3.1), the following table Table 5.2 summarizes the possible interpretation of each cluster based on the features described in Driving Style Features and their corresponding potential human factors discussed in Table 2.2. The details of the style feature statistics are also provided in Table 5.1 for reference, along with the interpretation.

Table 5.2: Interpretation of each cluster based on driving style and associated human factors

Cluster	Driving style	Key features	Inferred human factors
P1	Steady but impatient	Short $\bar{h}$ (1.31 s), small spacing (15.11 m), low σ_h	Mild aggressiveness, efficient car-following, strong focus on speed maintenance
P2	Slightly hesitant	Moderate $\bar{h}$ (1.43 s), longer $\bar{\tau}$ (2.00 s), high in variance (σ_τ and σ_h)	Occasional hesitation, longer anticipation, balanced between caution and efficiency
P3	Defensive and strongly conservative	Longest $\bar{\tau}$ (2.18 s), largest spacing (21.14 m)	Most cautious group, safety-oriented, delayed responses, stable but least efficient
P4	Fast-reacting but unstable	Lowest $\bar{h}$ (1.12 s), lowest $\bar{\tau}$ (1.71 s), higher variance (σ_τ and σ_h) compared to $P1$	Aggressive, impulsive decisions, high responsiveness but less stable control

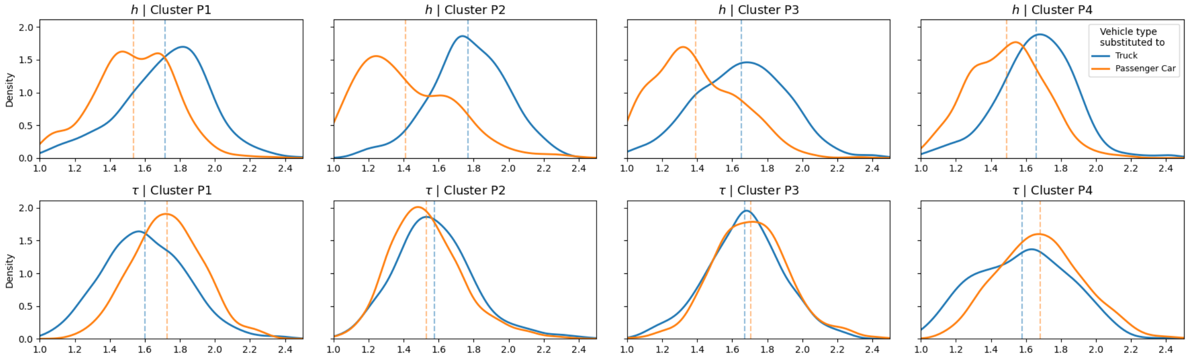


Figure 5.3: Replacing the follower with a truck (i.e., $l^F = 10$ m) per cluster

5.2. Driving Style Embedding Validation

The driving style embedder is jointly trained with the car-following model, allowing it to automatically extract meaningful style representations as long as they help improve prediction accuracy.

Following the presentation of prediction results, we now focus on validating whether the learned embeddings are indeed functionally meaningful. In particular, we examine whether the style token is effectively utilized by the model and whether it captures interpretable aspects of driving behavior.

To this end, we adopt three complementary validation perspectives to assess whether the style embedding contributes to the model's predictions and meaningfully characterizes individual driving styles.

5.2.1. Truck as Follower

Considering the scarcity of truck followers in the test set (Table 4.5), a direct statistical analysis is infeasible. However, since vehicle length is included as an input feature, we emulate truck behavior by modifying the follower's length ($l^F = 10$ m) in the model input. Note that clustering is no longer applicable under this setting.

As shown in Figure 5.3, replacing the follower with a truck ($l^F = 10$ m) causes clear shifts in the model-predicted distributions of headway (h) and reaction time (τ) across all clusters (P1–P4).

In the top row, the headway distributions (blue for trucks, orange for passenger cars) show only minor shifts: the peak positions remain similar, indicating that the model predicts comparable following distances for both vehicle types. However, in the bottom row, the reaction-time distributions consistently move leftward for trucks, meaning that the artificial truck followers exhibit shorter reaction times. This pattern is evident across all clusters except for Cluster P2 (in the second column), which likely includes more trajectories with highly dynamic or volatile leader movements. In such cases, followers need to stay highly alert to maintain stability. However, since trucks are less agile than passenger cars, they tend to exhibit slightly longer reaction times, leading to a minor deviation from the general trend.

These observations suggest that while the model maintains similar car-following spacing behavior regardless of vehicle type, it systematically predicts faster reactions for trucks. This aligns with the notion that truck drivers, often professionally trained and more attentive, respond more promptly to leader behavior, even though their overall following distance remains comparable (Ossen and Hoogendoorn, 2011).

5.2.2. Cross-over Test case: Effectiveness of the embedding

To examine how driving style influences behavior under comparable traffic conditions, the crossover is conducted within each true cluster—i.e., the original style group to which the samples belong—so that scene-related factors (e.g., leader motion, traffic density) remain controlled. In Figure 5.4, each subplot shows the distribution of samples from a given true cluster when their original embedding is replaced by the centroid of another cluster. Rows correspond to the original (true) cluster label, while columns indicate the substituted embedding (cluster centroid).

This analysis is motivated by the possibility that cluster embeddings encode both scene and behavioral attributes. By systematically swapping embeddings, we aim to isolate the behavioral component and assess its influence on average time headway $\bar{h}$. Observing a single column reveals how one embedding modulates samples from various clusters, while a single row illustrates how the same group responds to different embeddings. Significant shifts in $\bar{h}$ distribution across embeddings suggest that the learned embedding space captures behavioral traits independently of scene context.

As shown in Figure 5.4, embeddings P2 and P3 produce the most distinct effects on behavior, reflected in the highest and lowest values of average reaction time ($\bar{\tau}$) and time headway ($\bar{h}$), respectively. According to the cluster interpretation table (Table 5.2), P2 corresponds to conservative drivers—characterized by long headways and delayed reactions—while P3 represents reactive and aggressive behavior. These patterns are consistently reflected across multiple source clusters, supporting the semantic validity of the learned embedding.

These results confirm that the style embedding plays a decisive role in modulating car-following behavior. By substituting embeddings, the driver behavior can be systematically altered, demonstrating the interpretability and controllability of the embedding space.

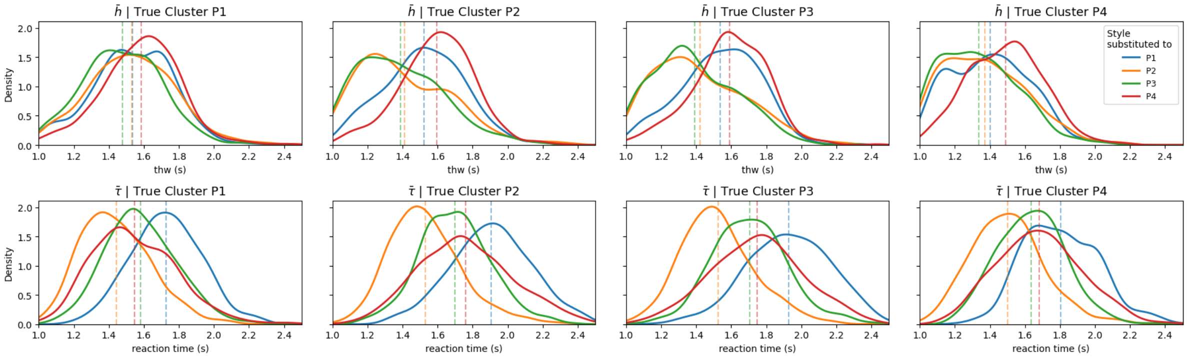


Figure 5.4: Cluster Centroid Embedding Substitution Test(each cluster evaluated using its embedding centroid, dashed lines indicate the mean values, and solid lines the distribution)

5.2.3. Example: Style Counterfactual Test

To visualize the behavioral differences induced by embedding replacement, two illustrative experiments are presented. Pair 1 compares embeddings from cluster P1 and P2—two behaviorally distinct groups—to highlight the contrast. Pair 2 compares a passenger car and a truck follower, both using the same embedding but differing in vehicle length (l^F). All experiments are conducted on the same sample (sample 10 from cluster P1). For clarity, the position trajectories are flattened by subtracting a reference motion term $-9.5t$.

As shown in Figure 5.5, the P1 embedding leads to slightly earlier reactions and closer spacing during the cruising regime (after 40 seconds), consistent with its profile in Table 5.2. In contrast, Figure 5.6 shows that the truck follower reacts more quickly than the passenger car but maintains a larger gap during deceleration phases (around 10–20 seconds). This behavior is attributable to the longer vehicle length and aligns with earlier findings on reaction time differences, reinforcing the conclusion that follower type modulates model behavior even when the style embedding is held constant.

5.3. CF Model Testing

Based on the aforementioned long-duration prediction task and evaluation objectives, the experimental setup has already been established. The proposed framework is applicable to all models, irrespective of whether they are data-driven or not. In this chapter, our focus shifts to long-duration prediction. Therefore, the following indicators are introduced to provide a comprehensive evaluation of model performance.

5.3.1. Error Statistics

Given the heterogeneity of driving styles and the dynamic nature of traffic—such as acceleration, deceleration, and cruising—prediction performance can vary across contexts. Aggressive following or larger headways, for example, affect the driving states, which is a huge uncertainty in model prediction. To capture this variability, we report not only the mean and standard deviation but also the 25th, 50th (median), and 75th percentiles (Q1–Q3) for each error metric in Table 5.3. These statistics provide a more complete view of error distribution and model robustness under diverse conditions. On top of that, the distributions of the metrics are plotted for reference Figure 5.7.

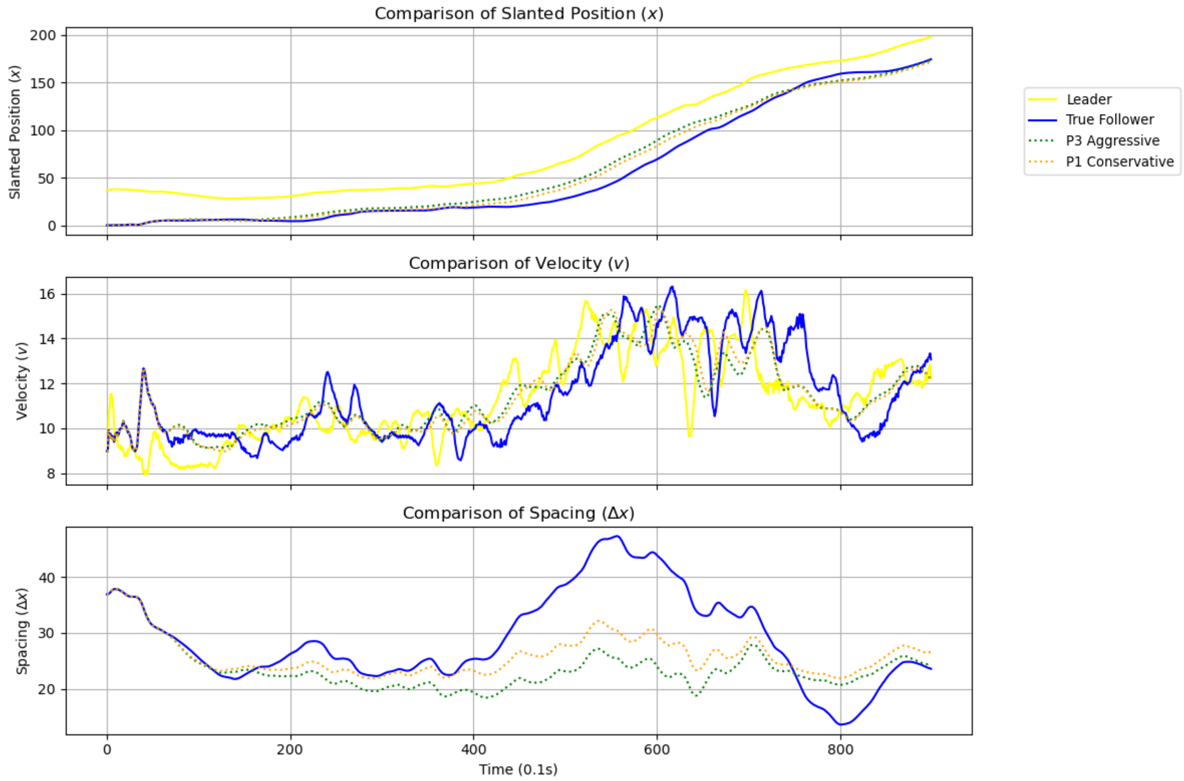


Figure 5.5: Embedding P2 and P3 follower, P2 $\bar{h} = 1.67$, $\bar{\tau} = 1.57$; P3 $\bar{h} = 1.58$, $\bar{\tau} = 1.41$

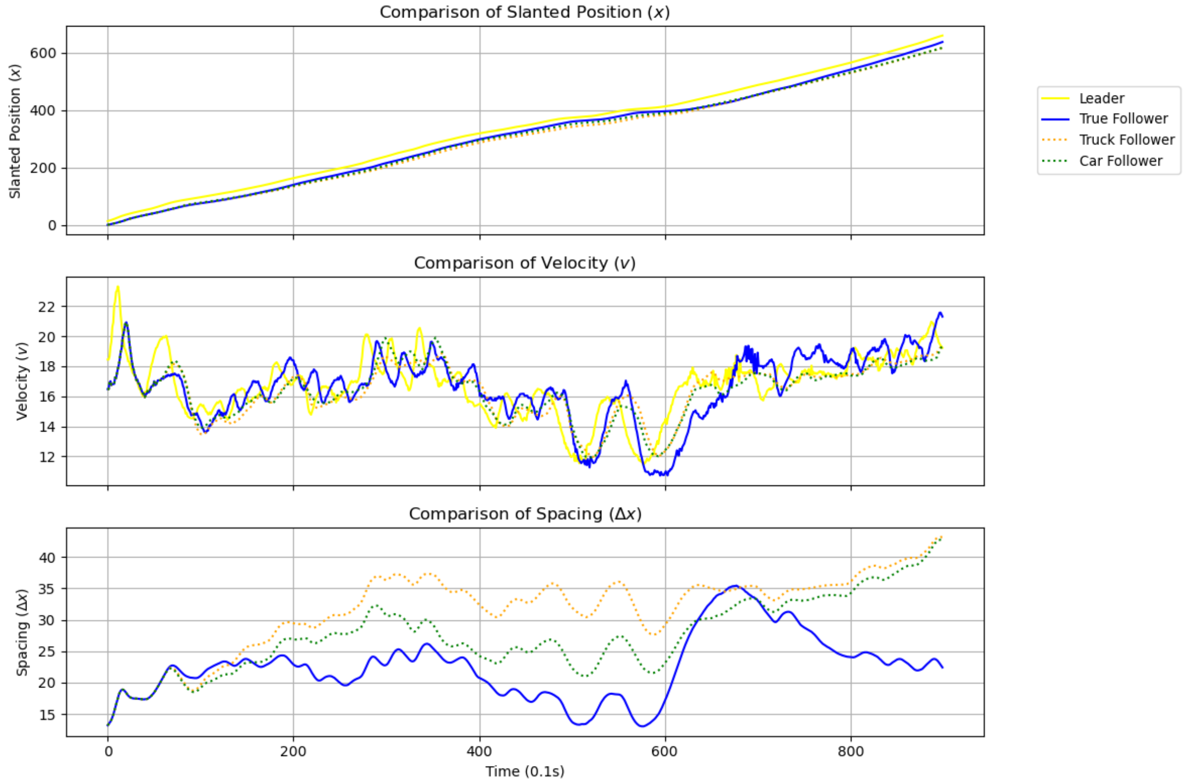


Figure 5.6: Truck and passenger car follower, Truck $\bar{h} = 1.70$, $\bar{\tau}$ = 1.82; Passenger car $\bar{h} = 1.42$, $\bar{\tau} = 1.79$ (Trajectory Sample 14)

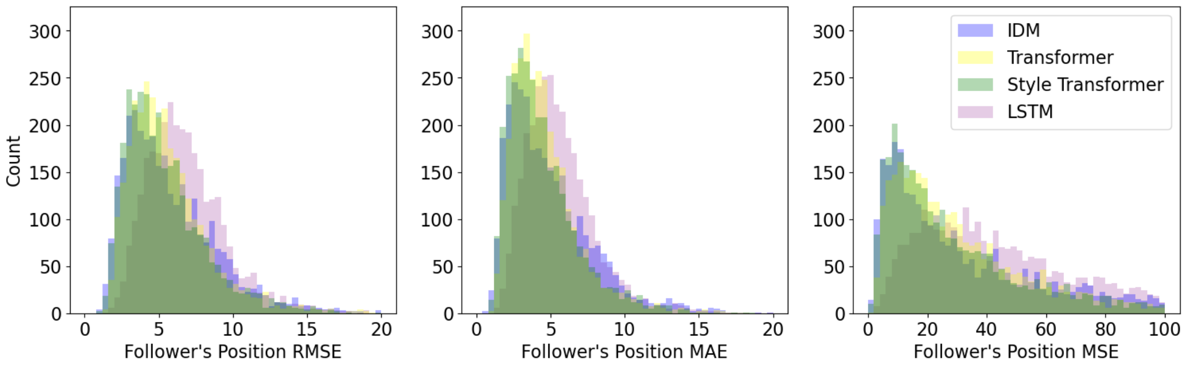


Figure 5.7: Follower's Position Metrics

Among the error metrics, the Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) are employed to quantify prediction accuracy. MSE penalizes larger errors more due to squaring, making it sensitive to outliers, while MAE offers a direct measure of average error magnitude. RMSE, as the square root of MSE, maintains unit consistency with the original variable. All metrics are computed based on the predicted vehicle position x_t^F, which serves as the primary evaluation variable. As the lowest-order variable in the loss function (Equation 3.1), it already incorporates all propagated noises, and has been widely used in previous CF studies (such as Huang et al. (2018)). The metrics are computed per individual trajectory to facilitate statistical analysis over the trajectories that are collected in different driving states.

The error metrics are computed as follows:

$$\mathrm{MSE} = \frac{1}{T}\sum_{t=1}^{T}\left(x_t^F - \hat{x}_t^F\right)^2, \tag{5.1}$$

$$\mathrm{MAE} = \frac{1}{T}\sum_{t=1}^{T}\left|x_t^F - \hat{x}_t^F\right|, \tag{5.2}$$

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(x_t^F - \hat{x}_t^F\right)^2}. \tag{5.3}$$

5.3.2. Evolution of Errors

Since we adopt a recursive prediction approach—where the output at each timestep serves as input for the next—prediction errors can accumulate over time. This compounding effect may cause a model that performs well initially to diverge significantly in the long term. Therefore, it is crucial to evaluate how prediction errors evolve as the forecast horizon extends. To this end, we visualize the error evolution curves for all four models over time. These curves allow us to assess the stability and robustness of each model under extended prediction durations, providing insight into both short-term accuracy and long-duration reliability.

Figure 5.8 shows the position MAE evolution over time for all models. The top plot presents the mean MAE at each prediction timestep, with shaded areas indicating one standard deviation ($\mu \pm \sigma$),

Metric	**Statistic**	**Style-aware CF**	**Transformer**	**LSTM**	**IDM**
MSE [m^2]	Mean	**36.54**	39.29	65.49	43.54
	Std Dev	45.03	43.95	58.97	49.81
	Q1 (25%)	11.50	13.90	24.59	12.11
	Q2 (Median)	22.16	25.92	41.39	26.01
	Q3 (75%)	43.65	47.97	69.31	56.80
RMSE [m]	Mean	**5.18**	5.26	6.79	5.64
	Std Dev	2.69	2.41	2.68	3.04
	Q1 (25%)	3.42	3.49	4.92	3.35
	Q2 (Median)	4.59	4.76	6.32	4.94
	Q3 (75%)	6.30	6.45	8.17	7.30
MAE [m]	Mean	**4.25**	4.28	5.57	4.82
	Std Dev	2.36	2.04	2.32	2.83
	Q1 (25%)	2.72	2.83	3.96	2.72
	Q2 (Median)	3.71	3.85	5.16	4.12
	Q3 (75%)	5.21	5.27	6.70	6.20

Table 5.3: Performance statistics of different models on prediction metrics (rounded to 2 decimals).

capturing both the average trend and variability. The bottom plot displays boxplots at every 10-second interval, illustrating the error distribution across samples as the prediction horizon increases.

Figure 5.8 illustrates the evolution and distribution of the position MAE error across the prediction horizon. In the top plot, we observe that although all models show increasing error trends over time, the LSTM model exhibits significantly larger variance, as shown by the broader confidence band. This is likely due to its lack of access to the prediction window information, resulting in instability in long-duration forecasting.

In the lower box plot, the error distribution across time further supports this observation. Both LSTM and the plain Transformer models demonstrate higher error variance, which suggests their sensitivity to short-term behavioral fluctuations—such as sudden acceleration or deceleration—present in individual driving styles. In contrast, the Style-aware CF and calibrated IDM models maintain relatively stable error distributions over time. This robustness can be attributed to the inclusion of explicit style information, either through style tokens or calibrated parameters, which prevents the models from overreacting to transient, style-driven perturbations.

5.3.3. Example Trajectories

To evaluate the models under realistic traffic scenarios, two testing examples were selected: one representing high-speed and one low-speed traffic. These two examples were chosen to capture the broader

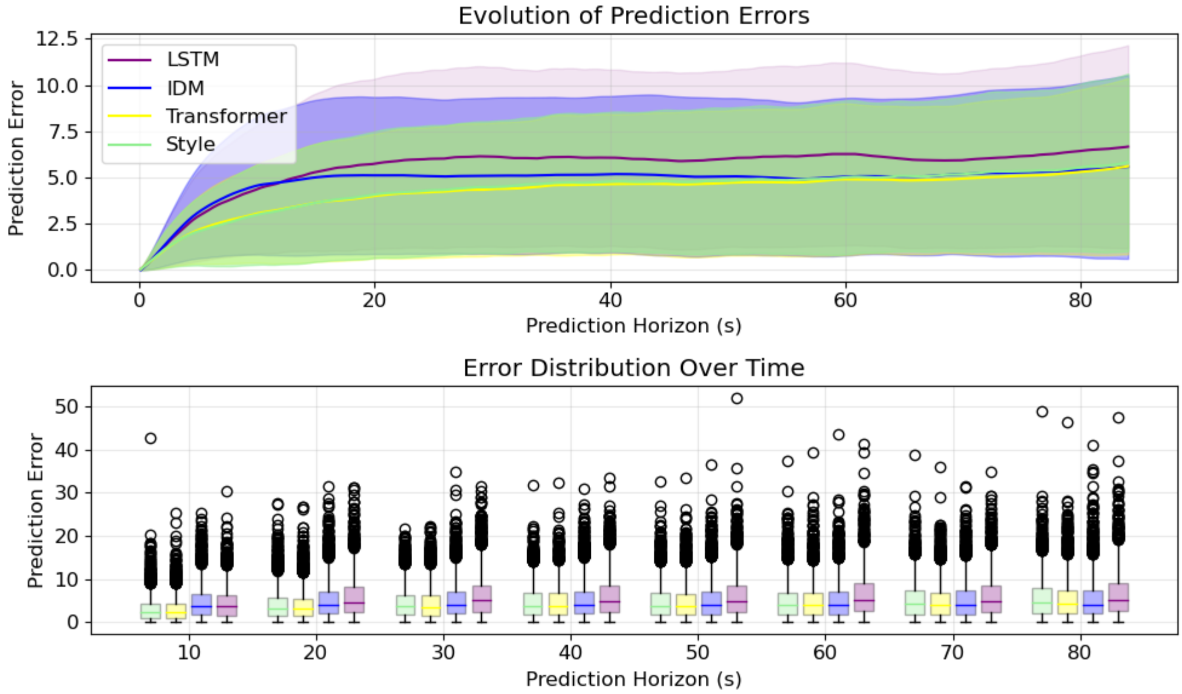


Figure 5.8: Evolution of the position MAE error over time

macroscopic characteristics of real-world traffic collectively. Additionally, to ensure coverage of various microscopic CF regimes—such as acceleration, deceleration, and cruising—the time window of each testing case was set to 90 seconds. This duration ensures that the leader's behavior is sufficiently rich, thereby reducing potential bias in evaluation. Note that non-CF samples have been excluded beforehand.

Several conclusions can be drawn from the experimental results:

1. As shown in Figure 5.9 and Figure 5.10 velocity diagrams, IDM responds to the leader without delay, especially when the leader accelerates (see velocity panel in Figure 5.10 around 100 s for an example). The result might be that IDM operates in a memoryless manner: each output is generated based on the current time step alone. In contrast, both the Transformer and Style-aware CF models can capture response delay, as they are trained with input and output sequences. This allows them to extract the temporal characteristics of human reaction time embedded in the data.

2. From Figure 5.9 and Figure 5.10 speed and spacing plots, it can be observed that all models are capable of learning to react to the leader. This is evidenced by the consistency in the spacing curves, which show synchronized responses to the leader's actions. However, the difference lies in personalized characteristics such as desired spacing (see slanted position plots) and preferred speed, which vary across models.

3. Both IDM and the Style-aware CF are guided by driving style information: IDM is parameterized with calibrated values representing average driver behavior, while the Style-aware CF uses a learned style token as input. In essence, both capture the mean driving style from the training

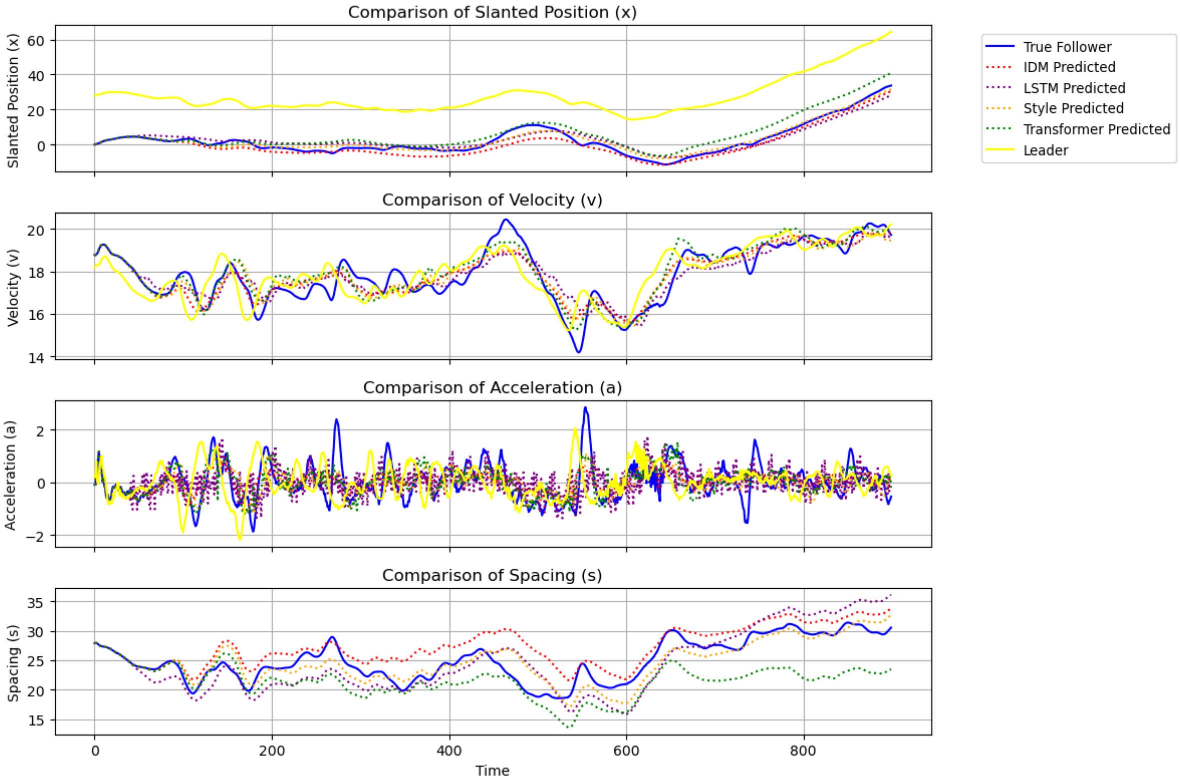


Figure 5.9: Trajectory comparison across models (Sample 200, slanted term: $-17.5t$)

set. However, IDM performs worse than the plain Transformer, but Style CF outperforms the rest. This could be the result of the parameter space of IDM being limited compared to the style token. Furthermore, IDM uses the uniform style, in other words, the same parameters across all the samples, which could be biased to some trajectories.

4. The raw acceleration signals (Figure 5.9 and Figure 5.10) are inherently noisy. However, the final converged models are able to produce smoothed acceleration profiles. This is attributed to the design of the loss function as well as the Kalman filtering process, which reduces inconsistencies in the data. The resulting smoothed acceleration curves are more consistent with natural human driving behavior. In contrast, the LSTM model fails to eliminate this noise and exhibits more volatile predictions. This behavior may imply a lack of convergence, which can be attributed to the absence of explicit future context within the prediction window—an inherent limitation of the LSTM architecture.

5.3.4. Attention Matrix

As is specified, the Transformer model maintains a certain interpretability, which is the attention mechanism (Figure A.5). In the time series prediction task, the attention matrix reveals which input timesteps, from both the provided labels and the historical window, the model attends to when generating predictions for each timestep within the prediction window. Inputs with higher attention weights contribute more to the subsequent computations and ultimately have a greater influence on the predicted outputs.

Figure 5.11 illustrates attention maps from Transformer Layer 1, revealing how different modules

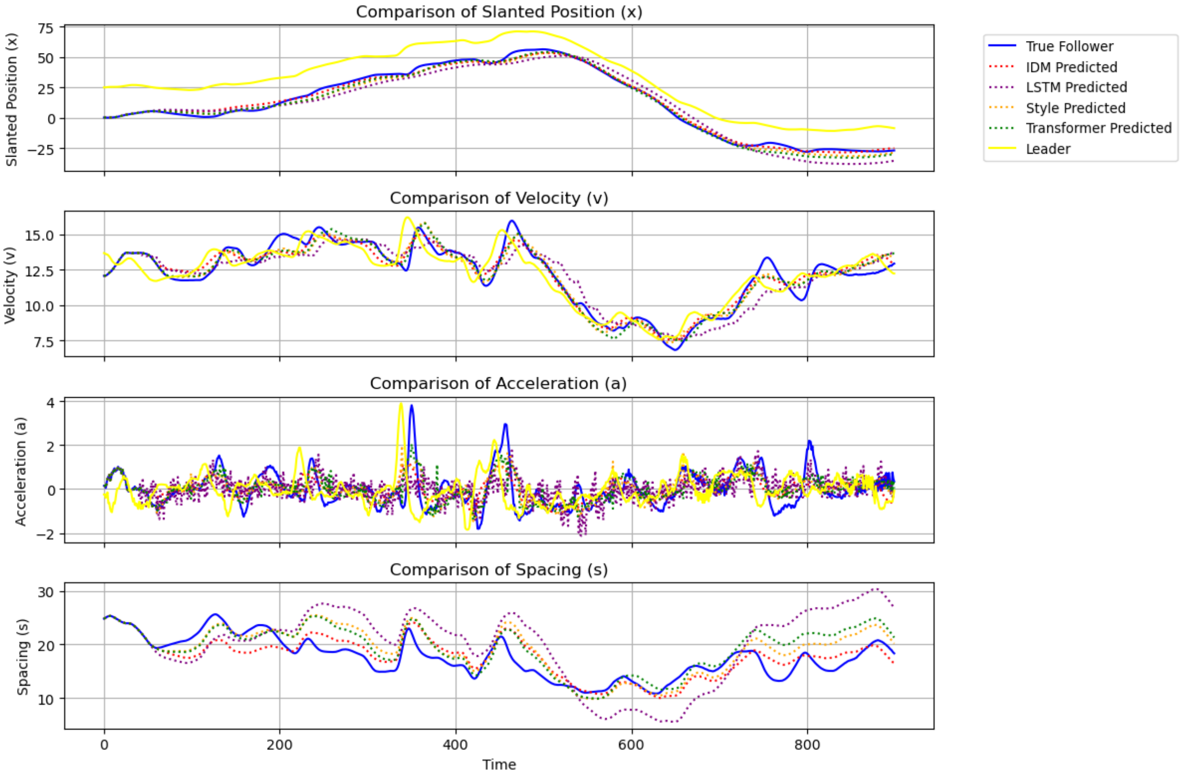


Figure 5.10: Trajectory comparison across models (Sample 230, slanted term: $-12.5t$)

handle temporal and contextual information during trajectory prediction. The y-axis indicates query positions—either the latter half of the prediction window (left and middle) or the full input (right). The x-axis shows key positions, beginning with a style token followed by input time steps (1–6s).

The attention maps clarify distinct roles. In the decoder cross-attention (left), the prepended style token is largely ignored for near-term queries (< 2 s) but is increasingly referenced at longer horizons (after approximately 2.5 seconds along the query axis, where the leftmost column in the key series corresponds to the prepended style token), where it provides complementary global guidance and supports more stable long-term forecasting. Note that the upper-right region is masked to enforce causality. Although the upper-triangular area appears brighter, data leakage is prevented, as explained in Appendix Causality and Masking.

In the decoder self-attention (middle), the pattern appears relatively diffuse rather than strictly lower-triangular. This arises because causal masking allows each newly generated step to access progressively updated context within the prediction window. Consequently, brighter regions appear in the lower-triangular area (key positions beyond 4 s), indicating the decoder's use of newly available information, while the darker upper-triangular region confirms that future inputs are effectively masked and excluded from attention.

By contrast, the encoder self-attention (right) presents a pronounced autoregressive diagonal across the entire historical window, highlighting its role in capturing strong temporal dependencies and faithfully propagating sequential information.

These patterns show that each attention module plays a distinct role: the encoder captures de-

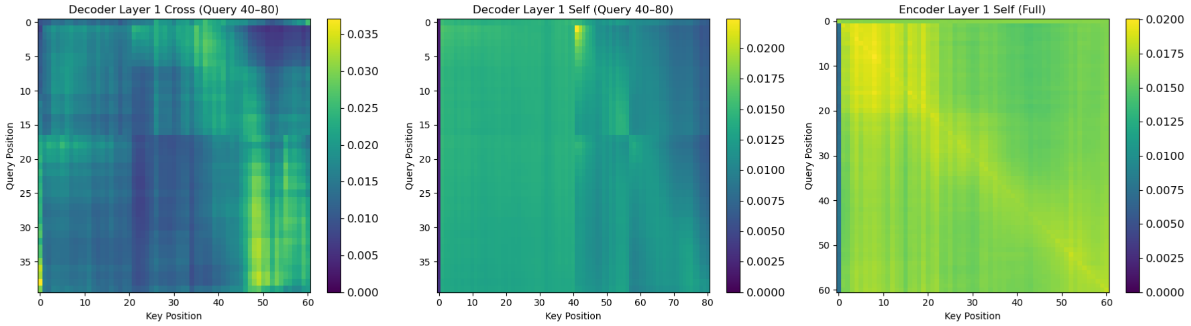


Figure 5.11: Attention matrix example (Sample 233, each timestep denotes 0.1 second)

terministic input dependencies, the decoder self-attention reflects fuzzy behavioral reactivity, and the decoder cross-attention integrates global style information.

5.4. Conclusion

In this chapter, we presented the long-duration experimental setup and framework, along with its corresponding results. The motivation for adopting long-duration evaluation stems from the intended application of the CF models in simulation environments, where long-duration performance is more aligned with the requirements of CF tasks.

The experimental findings lead to the following conclusions:

The experimental results highlight several key findings. First, the attention mechanism in Transformer-based models effectively captures weak autoregressive patterns in the decoder (Figure 5.11), supporting the view that a follower's behavior depends on temporally distributed cues rather than the leader's state at a single instant. This capacity allows Transformer models to learn reaction time characteristics, a feature absent in memoryless models like IDM.

Second, the style token plays a critical role in the Style-aware CF, contributing prominently in the cross-attention module and guiding behavior generation based on global driver characteristics.

Third, both the Style-aware CF demonstrates robustness to error accumulation over time, maintaining stable performance despite recursive input noise. In contrast, the LSTM is more vulnerable to such degradation, likely due to the lack of explicit style encoding. These models may overfit to short-term fluctuations, mistaking local behavior for generalizable patterns.

Among all models evaluated, the Style-aware CF achieves the best overall performance in terms of both accuracy and stability. Unlike IDM, which models reaction time as a fixed parameter, the Style-aware CF learns it implicitly and adjusts for different driving styles. Unlike LSTM, it captures context within the prediction window. And unlike the plain Transformer, it is grounded in a global representation of driving style. This combination makes the Style-aware CF uniquely suited for robust long-duration car-following prediction.

Part II

Evaluation of Driving Style Impacts on Traffic Breakdown

6

Experiment Design

To address the final research question—*Which driving styles, and which combinations thereof, contribute to breakdown generation under given traffic conditions?*—we now turn to Part II.

In Part I, we established a unified car-following model that embeds driving style and classified driving style into four representative categories: stable vs. unstable and aggressive vs. timid. Building on this classification and modeling framework, Part II investigates how these distinct driving styles, individually and in combination, influence platoon dynamics and the emergence of traffic breakdown.

6.1. Experiment Framework

6.1.1. Overview

Building on the platoon-scale simulator—which models an entire platoon rather than a single car-following pair—we now outline the experimental pipeline.

Each experimental setup has two components: (i) the leading-vehicle trajectory and (ii) the platoon composition. The leading-vehicle trajectory is treated as a control variable because it directly shapes the dynamics of the entire platoon (e.g., sustained cruising at low or high speed, accelerations, or stop-and-go patterns). The platoon composition is likewise influential: distinct driving styles and vehicle types, as well as their ordering within the platoon, can yield markedly different collective responses.

1. **Specify the setup.** Define a platoon with vehicles drawn from distinct driving-style classes and select a leading-vehicle trajectory recorded from real-world data.
2. **Assemble the platoon.** Randomly sample driving styles from the designated class set to form the target platoon, and choose one leader trajectory that spans as many traffic regimes as possible.
3. **Run the simulation.** Simulate for 4 minutes ($240\,\mathrm{s}$) and record the full state/output.

4. **Evaluate and analyze.** Compute the proposed metrics and analyze the outcomes to draw conclusions.

6.1.2. Platoon Simulation

The experiment builds on the style-transformer car-following model and its simulation (Algorithm 2). A simple car-following setup is insufficient to address the research question. Instead, the framework is extended to a platoon context, where the dynamics of the entire platoon are governed by the model. The following simulation design serves this purpose:

Algorithm 2 Platoon Recursive Prediction with Sliding History

```
Inputs:
    groundtruth data $D_{N\times T}$, including position, velocity and acceleration
    model $M$
Params: $T_{\text{hist}}$ (history length), $T_{\text{pred}}$ (prediction horizon), $T$ (length of time series), number of vehicle $N$
Output: Predicted platoon movements $\hat{Y}_{N\times T}$ (including position, velocity and acceleration)
$X_{N\times T} \leftarrow$ empty series $O_{N\times T}$          ▷ Initialize a default training movement series
$X_{2,1:T_{\text{hist}}} \leftarrow D_{2,1:T_{\text{hist}}}$          ▷ Initialize first follower's history window
$t_0 \leftarrow T_{\text{hist}}$          ▷ Initialize the start timestep
Initialize the head vehicle $\hat{Y}_{1,:} \leftarrow D_{1,:}$
for $k = 1$ to $\lfloor (T - t_0)/T_{\text{pred}} \rfloor$ do
    for $n = 2$ to $N$ do          ▷ From head to tail vehicle
        $t \leftarrow t_0 + (k-1)\,T_{\text{pred}}$
        $D \leftarrow (D_{n-1,\, t-T_{\text{hist}}:t+T_{\text{pred}}},\, x_{n,\, t-T_{\text{hist}}})$          ▷ Prepare input
        $\hat{Y}_{n,\, t:\, t+T_{\text{pred}}} \leftarrow M(D)$          ▷ Predict
        Force positive velocity $\hat{Y}_{n,\, t:\, t+T_{\text{pred}}}$
        Force no collision $\hat{Y}_{n,\, t:\, t+T_{\text{pred}}}$
        $X_{n,\, t:t+T_{\text{pred}}} \leftarrow \hat{Y}_{n,\, t:t+T_{\text{pred}}}$          ▷ Update history window
    end for
end for
return $\hat{Y}$
```

There are 3 points worth noticing in the platoon simulation compared to the car-following simulation:

1. The simulation operates in discrete time, where the state of every vehicle is updated at each timestep.

2. Within each timestep, vehicles are updated sequentially from head to tail, ensuring that every follower responds to the most recent state of its leader.

3. A collision-avoidance constraint is imposed to ensure that no two vehicles occupy the same spatial position. In rare cases where the model prediction implies that a follower would bump into its leader—an event scarcely represented in the training data—the vehicle is forcibly stopped by setting its velocity to zero.

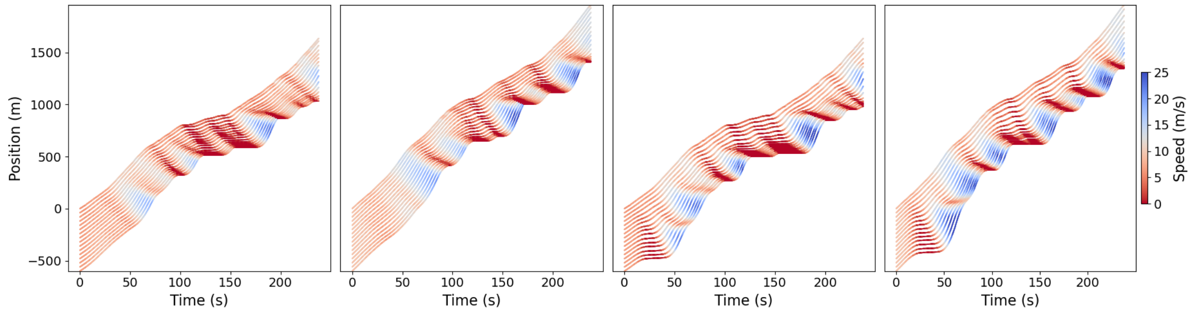


Figure 6.1: Example Simulation Results: Two Diagrams on the Left (Passenger Vehicle Platoons) vs. Two on the Right (Truck Platoons)

4. To complement this rule, a non-negativity constraint is applied on velocity. Specifically, whenever the predicted velocity becomes negative, which may arise as a side-effect of the model attempting to correct collisions, the value is reset to zero to prevent unrealistic backward motion.

To ensure the follow-up experiments are built on a well-rounded simulation environment, the following simulation examples (see Figure 6.1) are randomly drawn for visualization.

6.2. Metrics for Traffic Breakdown

Traffic breakdowns are often difficult to evaluate and quantify at the microscopic level. Moreover, there is no widely accepted standard for measuring their magnitude. Therefore, the metrics introduced here aim to characterize one or multiple breakdowns from different perspectives.

The indicators employed in the experiments are summarized in Table 6.1.

Indicator	Unit	Significance Test	Description
Delay	[s]	two-way ANOVA	Time loss compared to benchmark (Newell model); centered across leader trajectories
ST	[s]	two-way ANOVA	Total time in standstill regime ($v < 1$ m/s), centered across leader trajectories
FC	[L/km]	two-way ANOVA	Fuel consumption per kilometer
w_{acc}	[km/h]	t-test	Propagation velocity of accelerating traffic waves, detected via event-matching algorithm
w_{dec}	[km/h]	t-test	Propagation velocity of decelerating traffic waves, detected via event-matching algorithm
q_{max}	[veh/h]	WLS t-test	Capacity, inferred from fitted fundamental diagram
k_{jam}	[veh/km]	WLS t-test	Jam density, from congested branch intercept
k_{crit}	[veh/km]	WLS t-test	Critical density, derived from triangular FD assumption
b	[km/h]	WLS t-test	Slope of calibrated congested branch (absolute value gives backward wave speed)

Table 6.1: Indicators employed in the platoon experiments (ANOVA: analysis of variance, WLS: weighted least squares)

6.2.1. Trajectory-level Measures

Delay

Total time spent (TTS) is a widely used indicator in traffic engineering. However, it does not fully capture the dynamics of traffic breakdown. To address this limitation, this study introduces the concept of *delay*.

Delay is defined with respect to a benchmark scenario in which every vehicle in the platoon precisely replicates the leader's trajectory according to Newell's car-following model. Under this assumption, a benchmark TTS can be computed. By comparing it with the actual TTS obtained from observed trajectories, the difference reflects the additional time arising from deviations from ideal disturbance-free propagation:

$$\text{Delay} = \text{TTS}_{\text{actual}} - \text{TTS}_{\text{benchmark}} \tag{6.1}$$

where

$$\begin{aligned} TTS_{\text{actual}} &= \Delta t \cdot \sum_{n=1}^{N} \sum_{k=0}^{K-1} \mathbf{1}_{\{x_n^k < L\}}, \\ TTS_{\text{benchmark}} &= \Delta t \cdot \sum_{n=1}^{N} \sum_{k=0}^{K-1} \mathbf{1}_{\{x_n^{*,k} < L\}}. \end{aligned} \tag{6.2}$$

Here x_n^k is the actual position of vehicle n at time $t_k = k\Delta t$ ($k = 0, \ldots, K-1$, $K = T/\Delta t$), and $x_n^{*,k}$ is the benchmark position from Newell's model, obtained as a time–space shifted copy of its leader. The threshold L is determined as 500 meters before the last recorded position of the heading vehicle, with $\mathbf{1}_{\{x<L\}}$ indicating whether a vehicle is in the area of interest. This boundary is selected to ensure the travel time of each vehicle can be taken into account at the end of the simulation. For consistency, the benchmark shift is set so that the leader's benchmark and actual trajectories cross L at the same time.

Total Time Spent in Standstill

Standstill indicates the severity of traffic breakdown by measuring how long vehicles are nearly immobile. In this study, a standstill regime is defined as $v < 1$ m/s and $-0.1 < a < 0.1$ m/s^2 (see Table 6.2).

The total standstill time (ST) is computed as

$$\text{ST} = \Delta t \cdot \sum_{n=1}^{N} \sum_{k=0}^{K-1} \mathbf{1}_{\{v_n^k < 1 \text{ m/s},\}}. \tag{6.3}$$

Energy Efficiency

Beyond capacity loss and travel delays, traffic breakdown also has profound implications for energy efficiency. Stop-and-go patterns generated by breakdown events force vehicles into frequent acceleration and deceleration, which markedly increases fuel consumption. Such inefficiencies not only raise operational costs for drivers but also exacerbate environmental externalities at the system level. Examining energy efficiency alongside breakdown dynamics therefore enables a more comprehensive evaluation of traffic performance, linking operational stability with sustainability considerations.

The VT-Micro model (Rakha et al., 2004) estimates instantaneous fuel consumption and emissions as a function of a vehicle's speed and acceleration, using log-transformed polynomial regression calibrated with extensive chassis dynamometer data. The model distinguishes between positive and negative acceleration regimes, which improves accuracy in capturing transient driving behaviors and enables second-by-second fuel consumption estimation consistent with laboratory measurements.

With the calibrated coefficients, the model requires only vehicle speed and acceleration as inputs. In this study, the model is directly applied using the parameter set provided in Zegeye et al. (2013), which was calibrated on passenger vehicles and light trucks consistent with the vehicle types considered here.

The fuel consumption is denoted as FC in this study, and the details of the method and its corresponding parameter set can be found in the Appendix VT-Micro Fuel Consumption Estimation.

Significance Testing: Two-way ANOVA

To ensure a fair comparison of the indicator values across different experimental groups, the analysis must account for the fact that each group was observed under varying leader movements. Since the trajectory or motion pattern of the leading vehicle directly influences the follower's dynamic response, the observed differences in the indicator may partially stem from the leader-specific environment rather than intrinsic group characteristics. To disentangle these effects, a two-way analysis of variance (ANOVA) without interaction terms is adopted.

The two-way ANOVA can be represented as a linear model under the treatment (reference) coding scheme. Let $y_{p\ell i}$ denote the response variable (e.g., delay or standstill time) for the i-th observation (a single run of the simulation) within platoon p and leader environment ℓ. Each categorical factor is encoded using dummy (indicator) variables to capture deviations from a chosen reference level. Specifically, the model without interaction terms can be expressed as:

$$y_{p\ell i} = \mu + \sum_{p \neq p_0} \alpha_p D_{ip} + \sum_{\ell \neq \ell_0} \gamma_\ell D_{i\ell} + \varepsilon_{p\ell i}, \tag{6.4}$$

where p_0 and ℓ_0 denote the reference platoon and the reference leader environment, respectively. D_{ip} and $D_{i\ell}$ are binary indicator variables such that

$$D_{ip} = \begin{cases} 1, & \text{if observation } i \text{ belongs to platoon } p, \\ 0, & \text{otherwise,} \end{cases} \qquad D_{i\ell} = \begin{cases} 1, & \text{if observation } i \text{ belongs to} \ell, \\ 0, & \text{otherwise.} \end{cases}$$

Their inclusion allows the model to represent the mean deviation of each level relative to the chosen reference.

Here, μ represents the mean response of the reference combination (i.e., the reference platoon under the reference leader environment), α_p denotes the mean difference between platoon p and the reference platoon after controlling for leader effects, γ_ℓ represents the deviation associated with leader environment ℓ, and $\varepsilon_{p\ell i}$ is the residual error term, assumed to follow $\varepsilon_{p\ell i} \sim \mathcal{N}(0, \sigma^2)$.

Under this formulation, each coefficient α_p directly quantifies how the mean indicator value of platoon p differs from that of the reference platoon, averaged across the same set of leader environments.

Similarly, each γ_ℓ captures the mean shift introduced by leader ℓ relative to the reference leader. The corresponding t-tests for α_p and γ_ℓ therefore, evaluate the significance of platoon-specific and leader-specific deviations, respectively, while maintaining a common linear modeling framework consistent with the classical two-way ANOVA design.

6.2.2. Microscopic: Traffic Wave

Traffic waves, which propagate from downstream to upstream, serve as key indicators of traffic breakdown because they mark transitions between free-flow and congested states. The appearance of a decelerating wave signals the onset of congestion, whileana accelerating wave reflects the recovery from low-speed states. In this study, these two wave types are analyzed separately, with decelerating waves being particularly relevant for breakdown formation.

The primary variable of interest is the wave speed, defined as the upstream propagation velocity. Wave speed has been shown to depend on driving characteristics, such as reaction time (Chen et al., 2014). Other wave characteristics, such as duration, are not considered, since vehicle agents are trained to maintain car-following behavior, ensuring that waves always propagate upstream to the last vehicle without premature dissipation.

To identify waves and their speeds from trajectory data, we propose an automated detection algorithm (see Algorithm 4 fothe r complete version), which operates in three steps:

1. Event detection: Each vehicle's velocity series $\tilde{v}_i(t)$ is smoothed with a window W_s, and regime-change events $\mathcal{T}_i$ (decelerating or accelerating) are identified as local extrema within a look-around window W_ℓ. These events can be identified by detecting whether the acceleration values lie outside the threshold range after smoothing.

2. Pairwise matching: For each adjacent vehicle pair $(i, i+1)$, follower events τ_{i+1} are matched to leader events τ_i whenever $|\tau_{i+1} - \tau_i| \leq \tau_{\max}$. For matched events, the temporal lag, spatial gap, and wave speed are computed as

$$\Delta t = \tau_{i+1} - \tau_i, \quad \Delta x = x_{i+1}(\tau_{i+1}) - x_i(\tau_i), \quad w = \Delta x / \Delta t. \tag{6.5}$$

3. Wave chaining: Valid pairwise matches are chained downstream from the head vehicle to form complete wave sequences. A chain must span at least $L_{\min}$ consecutive vehicles. The wave speed is summarized as the median of the pairwise speeds c along the chain.

To this end, the algorithm can calculate the detected wave speeds with 2 categories – decelerating wave speed w_{acc} and accelerating wave speed w_{dec}.

Significance Testing: t-test

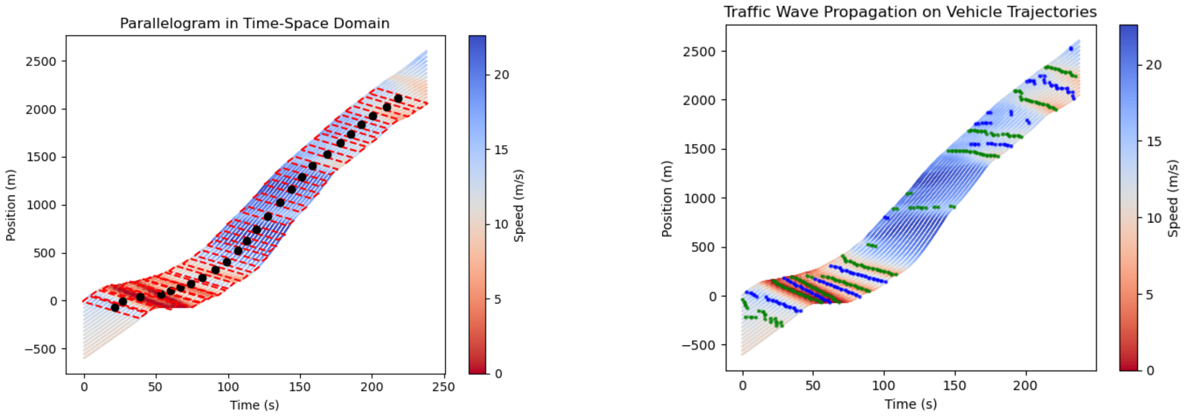
To evaluate whether the average wave speed differs across multiple experimental groups, pairwise comparisons are conducted using two-sample t-tests. One group is selected as the reference (baseline) condition, and the mean wave speed of each remaining group is compared against this reference. This approach allows for identifying which specific groups show statistically significant differences in wave speed relative to the baseline. The t-test is chosen because it provides a simple and interpretable way to quantify mean differences while maintaining statistical rigor under approximately normal and independent observations. Before applying the t-test, the normality of the wave speed distributions is examined, as the data are collected under different leaders and may not strictly follow a normal distribution.

6.2.3. Macroscopic Variable Calibration

Derivation of Parameters in Fundamental Diagram

Macroscopic traffic variables, namely flow, density, and speed, are commonly used to characterize the overall condition of a platoon within a specified time–space region. According to Edie's definition (Edie, 1965), these variables can be consistently retrieved for any arbitrary region in the time–space plane.

To capture the dynamics of a moving platoon, parallelograms are adopted as the basic measurement units. As illustrated in Figure 6.2a, parallelograms can cover the full vehicle trajectories while minimizing the inclusion of void areas. This property ensures both representativeness and accuracy in the measurement process. The detailed implementation of the parallelogram construction algorithm is provided in the Appendix Automatic Construction of Parallelograms Along a Platoon.



(a) Generated parallelograms in time-space domain

(b) Detected traffic waves in time-space domain (blue: deceleration wave, green: acceleration wave)

Figure 6.2: Example of parallelogram generation and traffic wave detection in time-space domain (Sample 16)

By applying this approach, the flow, density, and speed of the platoon can be extracted as a time series, where each element corresponds to the aggregated traffic variables derived from a single parallelogram.

To examine the macroscopic traffic characteristics, the congested branch of the fundamental diagram is approximated using a linear fit, which provides clearer insights than a simple scatter representation. A weighted least squares (WLS) regression is applied across experiments, where experiment identifiers are incorporated as dummy variables, to obtain a robust estimate of the congested-branch relationship for each experiment group:

$$q(k) = a + bk \tag{6.6}$$

with q denoting flow and k denoting density. The absolute slope $|b|$ corresponds to the backward wave speed $w = -b$, indicating the rate at which a disturbance propagates upstream.

Based on the triangular fundamental diagram assumption, the critical density can be directly inferred from this fitted congested branch. The free-flow speed v_f is obtained as the 90th percentile of the leading-vehicle speed, which provides a stable representation of the typical free-flow condition while suppressing occasional extreme values or measurement noise. Given v_f and the jam density $k_j = a/w$ (the density where $q(k) = 0$), The critical density is calculated as

$$k_{\text{crit}} = \frac{w\,k_j}{v_f + w} = \frac{a}{v_f + w} \tag{6.7}$$

and the corresponding capacity follows as

$$q_{\max} = v_f\,k_{\text{crit}} \tag{6.8}$$

This formulation avoids refitting a separate free-flow branch and allows direct style-wise comparisons of k_{crit} and $q_{\max}$ using the parameters a and b obtained from the robust congested-branch fit.

Significance Testing: WLS

To formally evaluate whether the slope of the congested branch of the fundamental diagram differs across driving styles, a weighted least squares (WLS) regression model with platoon–density interactions was employed.

This approach was chosen for three reasons. First, by including all four style groups within a single model, we can conveniently specify a reference category and assess the pairwise differences in slopes among the groups within a unified framework. Second, as discussed previously, the congested branch of the fundamental diagram is well approximated by a linear function; hence, a least squares–based approach is appropriate for estimating the slope parameters. Third, inspection of the observed density–flow scatter plots revealed that the data are not uniformly distributed: samples concentrate around the critical density (k_{crit}), whereas observations at higher densities are sparse. If fitted without adjustment, the regression would be overly influenced by the mid-density region and fail to capture the characteristics of the high-density regime. To mitigate this imbalance, we applied a weighting scheme that assigns larger weights to under-represented (high-density) observations.

Weighting Strategy: Formally, each observation i was assigned a weight w_i inversely proportional to the local sampling frequency of density:

$$w_i = \frac{1}{n_b(k_i)},$$

where $n_b(k_i)$ denotes the number of observations within the density bin b that contains k_i. The weights were subsequently normalized to have a unit mean, i.e.,

$$w_i^* = \frac{w_i}{\frac{1}{N}\sum_{j=1}^{N} w_j},$$

so that the overall scale of the regression remains unchanged. This weighting strategy effectively reduces the dominance of the dense mid-range samples and ensures that the estimated slopes are not biased toward regions with disproportionately frequent observations.

Model: Let q_i denote the flow and k_i the density measured in parallelogram i (see Figure 6.2a), and let D_{is} be dummy variables indicating the experiment s ($s = 2, \ldots, S$, with platoon 1 as the reference group). After centering density at its sample mean k_0 to improve numerical stability, the model is specified as

$$q_i = \beta_0 + \beta_1(k_i - k_0) + \sum_{s=2}^{S} \gamma_s D_{is} + \sum_{s=2}^{S} \delta_s \big[(k_i - k_0)D_{is}\big] + \varepsilon_i, \tag{6.9}$$

where ε_i is an error term. The coefficient β_1 represents the slope of the congested branch for the reference platoon, while δ_s captures the difference in slope between experiments s and the reference group. Pairwise t–tests on the model parameters are conducted to examine the differences in slopes across platoons. For simplicity and clearer interpretation, all comparisons are made against a preselected reference level, rather than performing all possible pairwise contrasts.

For each experiment s, the fitted parameters yield a group-specific linear relationship on the congested branch based on the Equation 6.6:

$$q(k) = a_s + b_s k \tag{6.10}$$

where a_s and b_s are obtained by back-transforming the centered coefficients:

$$a_s = (\beta_0 - \beta_1 k_0) + \gamma_s - \delta_s k_0, \qquad b_s = \beta_1 + \delta_s. \tag{6.11}$$

The backward wave speed is then $w_s = -b_s$. Assuming a triangular fundamental diagram, the jam density is $k_{j,s} = a_s / w_s$ and the critical density follows as

$$k_{\text{crit},s} = \frac{w_s k_{j,s}}{v_{f,s} + w_s} = \frac{a_s}{v_{f,s} + w_s}, \tag{6.12}$$

where the free–flow speed $v_{f,s}$ is taken as the 90th percentile of the leading–vehicle speed in experiment s (fixed to 50.4 km/h), providing a robust representation of typical free-flow conditions. Confidence intervals for $k_{\text{crit},s}$, $q_{\text{max},s}$ and for pairwise differences between styles are obtained by nonparametric bootstrap resampling of the parallelogram observations.

6.3. Experiment setup

6.3.1. Data Used

The data used in Part II consists of trajectories from the Zen-traffic dataset (see Figure 1.1 for the concept). These trajectories serve as the leader vehicle, guiding the movements of the entire platoon. Concurrently, the style embeddings and their associated cluster information, derived in the previous section, are provided as inputs to the style-aware CF model, ensuring that agents drive according to the desired styles.

Leaders Trajectory Profile

The movements of the leading vehicle largely determine the dynamics of the entire platoon. Hence, it is essential to examine the proportion of different movement regimes within the collected trajectories. In this dataset, each trajectory spans four minutes, providing sufficient simulation time for a traffic breakdown to emerge.

Vehicle movements are categorized into 8 regimes (see Table 6.2). Some regimes, such as cruising and mild acceleration, have little impact on platoon stability, whereas others, such as sharp deceleration, can significantly disturb it.

Figure 6.3 illustrates an example trajectory segmented into regimes according to the taxonomy in Table 6.2, using the PRAVT algorithm (Sharma et al., 2018) developed for car-following regime segmentation.

	Acceleration			Deceleration				
Regime	Mild	Moderate	Sharp	Mild	Moderate	Sharp	Cruise	Stand.
Code	A	Am	As	D	Dm	Ds	C	S
a**(m/s^2)**	$[0.1, 0.7]$	$[0.7, 1.5]$	$(1.5, \infty)$	$[-0.7, -0.1]$	$[-1.5, -0.7]$	$(-\infty, -1.5]$	$[-0.1, 0.1]$	
v**(m/s)**	-	-	-	-	-	-	$v > 1$	$(0, 1]$

Table 6.2: Regime definitions, thresholds, and codes (Stand. = Standstill)

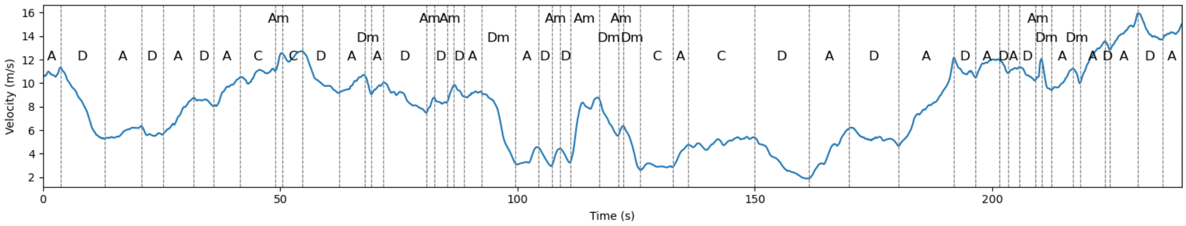


Figure 6.3: An example of the regime segmentation for one trajectory

Accordingly, 10 samples were selected, each with a duration of 240 seconds, to serve as the experimental dataset. As summarized in Table 6.3, the dataset spans all regimes, thereby capturing the full range of traffic conditions. In particular, the proportion of mild deceleration (31.15%) ensures that the leading vehicle is capable of inducing the intended traffic breakdown. Moreover, the dataset also contains extreme cases of strong acceleration (6.38%) and deceleration (6.45%), which, in line with the study's objectives, are expected to trigger larger disturbances within the platoon.

Table 6.3: Time share of different regimes

	A	D	C	Dm	Am	As	Ds	S
Duration (s)	774.72	747.71	563.93	146.96	134.06	18.91	8.00	5.70
Share (%)	32.28	31.15	23.50	6.12	5.59	0.79	0.33	0.24

Style Profile

The classification results are presented in subsection 5.1.2. Specifically, Table 5.1 shows the style classification result, and Table 5.2 provides a descriptive interpretation of each cluster. These tables summarize both the detailed car-following characteristics and their qualitative meanings. Each passenger car–leading group contains at least 400 samples, while each truck-leading group includes around 150 samples, both of which constitute sufficiently large pools for subsequent sampling and analysis.

Building on these classification results, the experiments draw random samples from the relevant style groups at each repetition (see Table 6.4), thereby ensuring that the heterogeneity within each style group is preserved.

6.3.2. Experiments

This experiment investigates how driving style influences traffic breakdown in platoons under car-following conditions. Because the number of possible style permutations grows combinatorially, we restrict attention to a set of practically relevant platoon compositions and design the following experiments accordingly.

The tested setups are summarized in Table 6.4. In total, three main experiments were designed, each addressing a specific research question. Within each main experiment, a series of sub-experiments were conducted to explore related aspects of the question. For example, Experiment 1 is divided into four sub-experiments. Each sub-experiment includes five simulations performed under the same leading-vehicle profile, and in total, ten different leading-vehicle profiles are tested. In other words, 50 simulations are conducted in Experiment 1-a. Note that both the simulations and the sub-experiments are independent and **do not interfere** with one another. Once a simulation is completed, the simulation environment is reset before running the next one.

Notations used in the Table 6.4 require additional clarification:

1. $T1$ denotes a truck agent that follows style group 1 but with a vehicle length of 9 meters, whereas $P1$ follows the same style group but with a length of 5 meters. Note that T and P do not correspond to any particular driving style; they are randomly assigned in each simulation run.
2. $[\cdots]$ denotes an ordered list, whereas $\{\cdots\}$ denotes an unordered set that only specifies the proportion of styles.
3. Subscripts indicate the number of elements in the list. For example, $[P1, P1, \cdots]^{16}$ represents a list of 16 vehicles.
4. $P1(25\%)$ indicates that 25% of the vehicles in the platoon belong to style $P1$.

Experiment 1

In a style-homogeneous platoon, what is the impact on potential traffic disturbances?

Existing studies often classify driving styles into two broad categories—aggressive and timid (Laval and Leclercq, 2010). This dichotomy is also reflected in the clustering results presented earlier (see Section Interpretation: Cluster Profile). A key question, therefore, is whether timid drivers destabilize traffic more than their aggressive counterparts. The experiments conducted in this study are designed to shed light on this issue.

Moreover, the clustering results indicate that driving styles extend beyond the simple aggressive–timid division (see Table 5.2). To capture these additional behavioral groups, the experiments are organized around homogeneous platoon compositions, enabling a systematic examination of their distinct traffic impacts.

Experiment 2

Is an alternating composition more efficient than a homogeneous composition?

In real-world traffic, driving styles are often more complex and stochastic than a simple homogeneous setting. In this experiment, we investigate mixed driving style compositions under two scenarios: an alternating arrangement and a homogeneous arrangement. The findings from Experiment 1 indicate that aggressive drivers generate faster traffic waves but experience shorter standstill times, whereas conservative drivers produce slower waves yet longer standstill durations. Hence, it is of interest to examine the performance of platoons composed of both styles.

To extend the analysis, all six alternating style pairs were tested, given the four styles P1–P4. Among them, the alternating composition of P1 and P3 yielded the most favorable delay performance, outperforming not only the other alternating pairs but also the homogeneous compositions in Experiment 1. Motivated by this result, an additional experiment was conducted to examine mixed platoons with varying proportions of P1 and P3 drivers (25%, 50%, and 75%, respectively).

Experiment 3

How does truck placement affect platoon efficiency?

Trucks are of particular interest in traffic engineering, yet most existing car-following models (Zhou et al., 2017; Huang et al., 2018) fail to account for passenger cars and trucks jointly. Building on the style-transformer model, this experiment examines the impact of trucks within passenger-car platoons.

A key question concerns the distribution of trucks: on motorways, they are often observed either clustered together within a platoon of passenger cars or dispersed individually throughout the flow. This experiment investigates which distribution pattern—clustered or dispersed—yields greater traffic efficiency.

In Experiment 3, our experiments show that when platoon size is fixed, positioning the truck group at the tail of the platoon yields the lowest overall delay, because the trucks exert minimal influence on the passenger cars ahead. However, such an ideal arrangement rarely occurs in real traffic. To explore more practical scenarios, we compared clustered (3-e and 3-f), alternating (3-g), and random (3-h) truck allocations.

Table 6.4: Experiment set-up summary (Follow-up exp. is for acquiring more indicator samples to achieve statistical significance.)

Experiment	Composition	Category	Iteration (Repetitions × Leaders)
1-a	$[P1, P1 \cdots]^{16}$	aggressive and stable	15 × 10
1-b	$[P2, P2 \cdots]^{16}$	timid but unstable	15 × 10
1-c	$[P3, P3 \cdots]^{16}$	timid and stable	15 × 10
1-d	$[P4, P4 \cdots]^{16}$	aggressive but unstable	15 × 10
2-a	$[P1, P2, P1, P2, \dots]^{16}$	mixed stability	15 × 10
2-b	$[P1, P3, P1, P3 \cdots]^{16}$	stable	15 × 10
2-c	$[P1, P4, P1, P4 \cdots]^{16}$	mixed stability	15 × 10
2-d	$[P2, P3, P2, P3 \cdots]^{16}$	mixed stability	15 × 10
2-e	$[P2, P4, P2, P4 \cdots]^{16}$	unstable	15 × 10
2-f	$[P3, P4, P3, P4 \cdots]^{16}$	mixed stability	15 × 10
2-b-1	$\{P1(25\%), P3(75\%)\}^{16}$	stable (P3 biased)	15 × 10
2-b-2	$\{P1(50\%), P3(50\%)\}^{16}$	stable (balanced)	15 × 10
2-b-3	$\{P1(75\%), P3(25\%)\}^{16}$	stable (P1 biased)	15 × 10
3-a	$[T1, T1 \cdots]^{16}$	aggressive and stable	15 × 10
3-b	$[T2, T2 \cdots]^{16}$	timid but unstable	15 × 10
3-c	$[T3, T3 \cdots]^{16}$	timid and stable	15 × 10
3-d	$[T4, T4 \cdots]^{16}$	aggressive but unstable	15 × 10
3-e	$[P, \cdots]^4[T, \cdots]^4[P, \cdots]^4$	clustered trucks at center	15 × 10
3-f	$[T, T \cdots]^8[P, P \cdots]^8$	clustered trucks as leader	15 × 10
3-g	$[T, P, T, P \cdots]^{16}$	alternating	15 × 10
3-h	$\{T(50\%), P(50\%)\}^{16}$	random	15 × 10

7

Experiment Result

7.1. Experiment 1 - Style Homogeneous Platoon

Platoons composed of different driving styles may induce traffic breakdown in distinct ways. Four major style groups, identified in Table 5.2, are examined in this experiment.

7.1.1. Trajectory-level Measures

Delay serves as a general indicator to assess whether a particular platoon composition outperforms others. Although it overlooks many details of traffic breakdown, such as wave speed, it provides a holistic view of overall platoon performance. Similarly, a standstill is a common feature of traffic breakdown and, to some extent, reflects the severity of congestion.

Table 7.1: Difference of indicators compared to the reference level (1-c), * indicates the difference compared to the reference level (in bold text) significant (p < 0.001)

Experiment	$\lvert w_{\text{dec}} \rvert$ (m/s)	$\lvert w_{\text{acc}} \rvert$(m/s)	ST (s)	Delay (s)	FC (L/km)
	t-test			WLS + t-test	
1-a	+0.55*	+0.56*	-26*	+11*	-0.48*
1-b	+0.51*	+0.22*	+77*	+45*	+0.43*
1-c	**5.68**	**5.29**	**228**	**835**	**0.69**
1-d	+1.01*	+1.05*	+38*	+41*	-0.45*

Table 7.1 reports the results on delay and standstill time, and wave speeds (in absolute value). Overall, Group 1-a (aggressive–stable) and Group 1-d (aggressive–unstable) exhibit faster wave propagation, with both decelerating and accelerating occurring quickly. Consequently, their standstill durations are relatively short, suggesting that although breakdowns emerge rapidly, they also dissipate quickly, which can be beneficial for overall platoon performance. By contrast, the conservative–unstable style

(Group 1-b) generates slower wave propagation and a smaller spatial extent of disturbance, but at the cost of substantially longer standstills and higher congestion impact.

From the perspective of centered delay, shorter standstill times for aggressive groups do not necessarily imply superior performance. Instead, the conservative–stable platoon (Group 1-c) achieves the lowest delay, a result that is statistically significant compared with the other groups. This finding suggests that while stable conservative drivers may induce longer standstills, the overall severity of breakdown is mitigated.

In terms of fuel consumption, generally the conservative driving style groups (b and c) consume more fuel than the aggressive, which might indicate such a driving style ends up with more time in low-speed gliding, leading to high fuel consumption but stabilizing the traffic.

Meanwhile, group 1-c and 1-d stand out from the rest groups. The 1-c platoon shows a stable but slow-reacting pattern with long stops and extended low-speed gliding, which prolongs operation in energy inefficient regimes and leads to the highest fuel consumption per distance despite the lowest overall delay.

In contrast, the 1-d follows a stop–go pattern with rapid accelerations and decelerations. Despite a relatively high standstill time ($v < 1$ m/s), it minimizes low-speed crawling and quickly returns to efficient cruising, showing that high ST does not necessarily imply high FC; by reducing low-speed operation and increasing efficient cruising, 1-d achieves lower fuel consumption per distance.

7.1.2. Detected Traffic Wave

Traffic waves generated by the four driving styles exhibit substantial differences, highlighting the need to further examine their characteristics. Since traffic waves play a pivotal role in the onset and development of congestion, their analysis can provide critical insights into the behavioral properties of platoons.

It is important to clarify that the traffic waves considered in this section differ from those in the fundamental diagram. In the latter, wave speed is typically calibrated using macroscopic quantities, whereas in this study, the waves are empirically detected and subjected to statistical testing. Moreover, the detected waves can be categorized into accelerating and decelerating waves, a distinction that is essential for understanding the propagation dynamics of traffic breakdown.

Figure 7.1 illustrates the accelerating and decelerating wave speeds, together with their 95% confidence intervals obtained through bootstrapping. The corresponding numerical values of the wave speeds (in absolute value) are reported in Table 7.1.

The scatter of mean accelerating vs. decelerating wave speeds (m/s, with 95% CIs) yields three consistent findings. First, the relationship is nearly symmetric: points cluster around the reference line $y=x$, and most lie slightly below it, indicating that accelerating waves tend to be marginally faster in magnitude than decelerating waves (asymmetry on the order of a few tenths km/h).

Second, styles differ systematically in wave–speed magnitude. Group 1-d exhibits the largest wave speeds among all groups and is clearly separated from the others, a pattern consistent with an *aggressive–unstable* style; by contrast, Group 1-c shows the smallest magnitudes, aligning with a *conservative–*

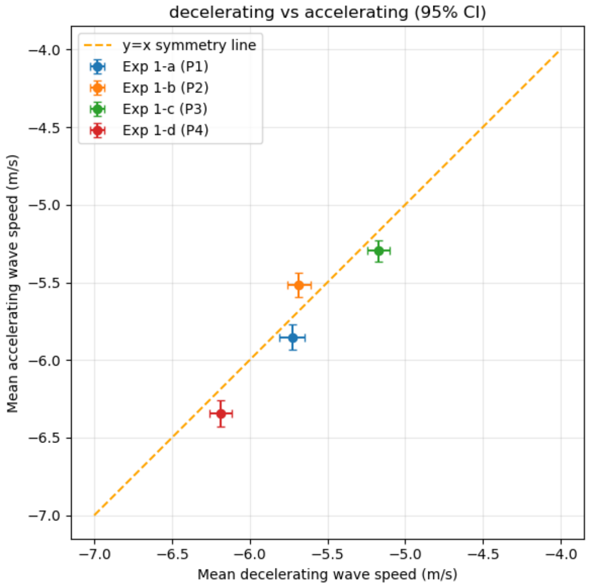


Figure 7.1: Wave speed comparison (Margins indicate the 95% CI.)

Table 7.2: Fundamental-diagram parameters estimated from WLS (reference = 1-c). Significance indicates difference from the reference group(* denotes $p < 0.001$).

Exp.	a	b	k_{jam}	k_{crit}	$q_{\max}$
	(veh/hr)	(km/hr)	(veh/km)	(veh/km)	(veh/hr)
	WLS + t-test		Bootstrap difference testing		
1-a	1493.38*	-24.34*	61.36*	19.06*	1029.40*
1-b	1294.70	-18.73	69.11	17.80	961.25
1-c	**1393.23**	**-19.52**	**71.37**	**18.95**	**1023.31**
1-d	1412.90*	-23.62*	59.82*	18.20*	982.94*

stable style.

Third, Group 1-b appears most fragile: its point and 95% CI approach (and may intersect) the line $y=x$, implying a nontrivial probability that the accelerating wave becomes slower than the decelerating wave; such an inversion suggests a higher propensity for deeper or more persistent breakdown relative to the other styles.

7.1.3. Macroscopic Traffic Variables

Macroscopic traffic variables offer an additional perspective on the simulation outcomes. This section presents the calibration of the fundamental diagram and the associated traffic variables, which serve as the basis for analyzing the differences across driving styles.

The analysis of the fundamental diagram parameters (Table 7.2 and Figure 7.2) suggests several conclusions. Note that the analysis is only carried out between the groups with a significant difference.

First, differences in wave speed primarily reflect drivers' reaction characteristics: faster-responding drivers (Group 1-a and Group 1-d) are associated with steeper wave speeds, whereas slower-responding

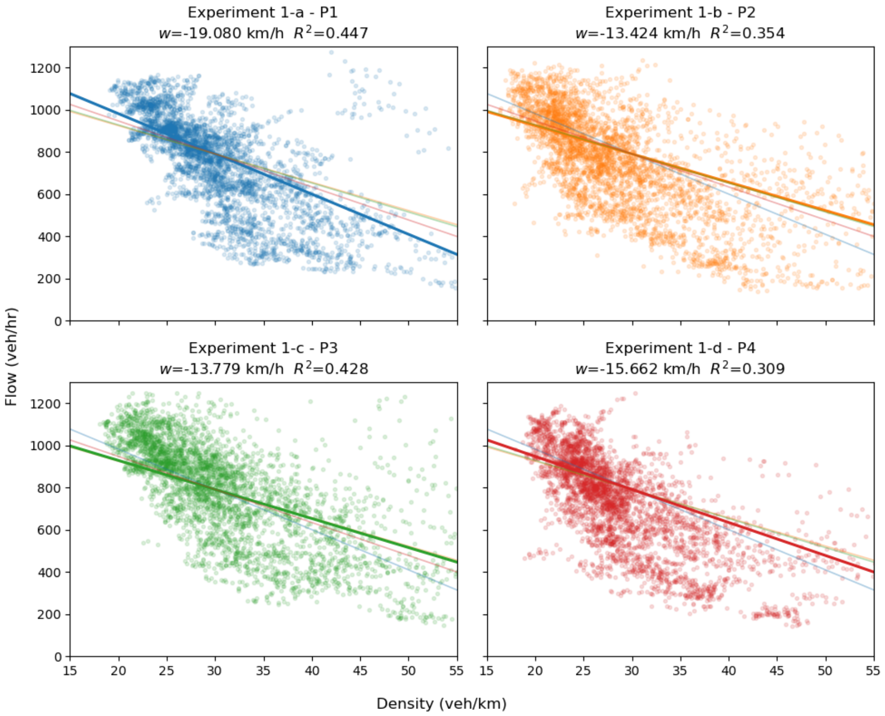


Figure 7.2: Calibrated Fundamental Diagrams (Congested Branch)

drivers (Group 1-b and Group 1-c) correspond to gentler slopes. Again, this result aligns with the finding in Section Detected Traffic Wave.

Second, the jam density reveals the platoon's compression capacity under congestion: defensive drivers (Group 1-c) can sustain the highest density, while unstable drivers (Group 1-d) lead to the sparsest conditions.

Third, the maximum flow is shaped by the trade-off between efficiency and stability. Group 1-a achieves the highest flow due to efficient car-following, Group 1-b yields the lowest flow because of hesitation and variability, Group 1-c maintains a moderate flow supported by cautious but consistent behavior, and Group 1-d's instability offsets the benefit of quick reactions.

Finally, at the capacity point, platoons composed of more stable drivers (Group 1-a and Group 1-c) are less prone to breakdown, because their critical densities are significantly higher than the unstable groups. This can be attributed to their lower variance in reaction time and headway over time, which helps sustain persistent driving patterns in homogeneous platoons. This conclusion could be applied in regulating traffic flow during high-volume periods, when traffic flow is unstable.

7.2. Experiment 2 - Mixing or Homogeneous Patterns

Experiment 1 revealed that the conservative–stable platoon delivered the best overall performance in terms of delay. Two additional insights emerged: (i) conservative drivers tend to generate larger breakdown areas than aggressive drivers, primarily due to their lower wave speeds; (ii) although breakdowns caused by aggressive drivers are usually shorter and less frequent, their higher magnitude often results in greater delays, usually accompanied by higher fuel consumption.

Motivated by these findings, Experiment 2 shifts the focus from homogeneous platoons to mixed compositions of two distinct driving styles. The objective is to test whether certain combinations can achieve even better performance. The investigation begins with alternating patterns, where two styles are interleaved throughout the platoon (yielding 6 possible style pairs). After identifying the most effective style pair, proportion tests are conducted to determine whether specific ratios outperform both homogeneous and alternating configurations.

Trajectory-level Indicators

Following the statistical testing procedure, Table 7.3 summarizes the results. The ANOVA confirms that each indicator shows a significant effect, and the subsequent HSD test is applied to identify pairwise differences of interest.

It should be noted that the data of Experiment 1 reported in Table 7.3 differ from the raw delay values presented in Table 7.1. This adjustment arises because, in Experiment 2, the data are obtained from 15×10 iterations (repetitions × leaders).

Table 7.3: Wave speed, standstill time, and delay for each platoon composition (* denotes the difference is significant ($p < 0.001$) compared to the reference level 2-b))

Experiment	Composition	$\lvert w_{\text{dec}} \rvert$ (m/s)	$\lvert w_{\text{acc}} \rvert$ (m/s)	Standstill (s)	Delay (s)	FC (L/km)
		t-test			WLS + t-test	
2-a	$[P1, P2, P1, P2, \dots]^{16}$	+0.20*	+0.28*	+44*	+23*	+0.24*
2-b	$[P1, P3, P1, P3, \dots]^{16}$	**5.51**	**5.61**	**321.9**	**492.0**	**0.98**
2-b-1	$\{P1(25\%), P3(75\%)\}^{16}$	-0.20*	-0.17*	+17*	-4	+0.45*
2-b-2	$\{P1(50\%), P3(50\%)\}^{16}$	-0.09	+0.02	-3	-2	+0.02
2-b-3	$\{P1(75\%), P3(25\%)\}^{16}$	+0.06	+0.11	-19*	+1	-0.31*
2-c	$[P1, P4, P1, P4, \dots]^{16}$	+0.38*	+0.49*	+6	+20*	-0.35*
2-d	$[P2, P3, P2, P3, \dots]^{16}$	-0.07	-0.10	+71*	+10*	+1.19*
2-e	$[P2, P4, P2, P4, \dots]^{16}$	+0.31*	+0.40*	+81*	+32*	+0.69*
2-f	$[P3, P4, P3, P4, \dots]^{16}$	+0.09	+0.19*	+40*	+16*	+0.41*
1-a	$[P1, P1, \dots]^{16}$	+0.21*	+0.24*	-39*	+1	-0.55*
1-b	$[P2, P2, \dots]^{16}$	+0.17	-0.09	+129*	+35*	+1.86*
1-c	$[P3, P3, \dots]^{16}$	-0.34*	-0.32*	+29*	-10*	+0.86*
1-d	$[P4, P4, \dots]^{16}$	+0.67*	+0.73*	+43*	+31*	-0.03

The results in Table 7.3 provides the following observations:

- Experiment set 2-b-x shows that the presence of P3 drivers has a stabilizing effect on traffic recovery. A higher proportion of P3 substantially reduces delay.
- In contrast, the inclusion of P2 drivers proves detrimental. Pure P2 or P2-dominated compositions (e.g., 2-d and 2-e) exhibit markedly longer delays and extended standstill durations.
- A longer standstill time does not necessarily imply greater delay (see experiments 2-b-x), as platoons with more conservative drivers (e.g., P3) may experience extended standstill yet achieve faster overall recovery due to their stabilizing effect.

In other words, the findings emphasize three key points: (i) P3 serves as a *stabilizer*, effectively mitigating congestion; (ii) P2 operates as a *drag*, slowing down recovery; and (iii) heterogeneous platoons—especially those mixing P1 and P3—do not substantially outperform their homogeneous counterparts.

It is also worth noting that the relationship between the proportion of P3 drivers and delay reduction is not strictly linear. While increasing the share of P3 generally improves efficiency, the marginal benefit becomes more pronounced once the proportion exceeds 75%.

Considering Energy Use

Beyond delay and wave dynamics, fuel consumption (FC) reveals an important trade-off. Although P3 is consistently identified as a *stabilizer*, its FC is notably higher. As shown in the proportion experiments (2-b-x), increasing the share of P3 yields a substantial reduction in delay, yet once the proportion of P3 exceeds 50%, FC rises sharply. This effect can be attributed to P3's tendency to remain in low-speed operating regimes, which increases energy use.

By contrast, P1 exhibits the most favorable performance in terms of FC. In the P1–P3 mixed platoons (2-b-x), when the share of P3 remains at or below 50%, the differences in FC are not statistically significant. This suggests that combining the two stable styles—P1 and P3—while keeping the proportion of P3 moderate, can strike a balance between minimizing delay and controlling fuel consumption.

7.3. Experiment 3 - Impact of Truck in Platoon

7.3.1. Profiles

Before entering the patterns of truck allocation in a platoon, it is useful to first investigate their homogeneous profiles, which can provide insights into the subsequent experiments. Therefore, as an additional setup, 3 homogeneous platoons are designed, each composed exclusively of one of the 3 different styles where a truck is the leader (see Table 5.2).

It is worth noting that the style transformer model allows the vehicle length to be specified, such that an agent behaves as a truck when its length is set to a larger value (9 meters in this study), which is denoted as "T", compared to the "P", which denotes the passenger car agents. See Table 5.2 for details.

Characteristics of Truck Platoon

Table 7.4: Difference of indicators compared to reference level (reference group: 3-a, * denotes the difference is significant $(p < 0.01)$ compared to the reference level)

Experiment	Composition	w^{dec}	w^{acc}	Standstill	Delay	FC
		(m/s)	(m/s)	(s)	(s)	(L/km)
		t-test		WLS + t-test		
3-a	$[T1, T1 \cdots]^{16}$	**6.12**	**5.11**	**21**	**9**	**3.22**
3-b	$[T2, T2 \cdots]^{16}$	+0.13*	+0.15*	-38*	+1	-0.84*
3-c	$[T3, T3 \cdots]^{16}$	+0.14*	+0.12	-41*	-19*	-0.69*
3-d	$[T4, T4 \cdots]^{16}$	+0.05	-0.05	-7*	-16*	+0.13*

Generally, the Figure 7.3 illustrates the main findings. Homogeneous truck platoons exhibit a larger discrepancy between decelerating and accelerating wave speeds compared to passenger-car platoons, indicating that congestion propagates more rapidly upstream while dissipating more slowly at

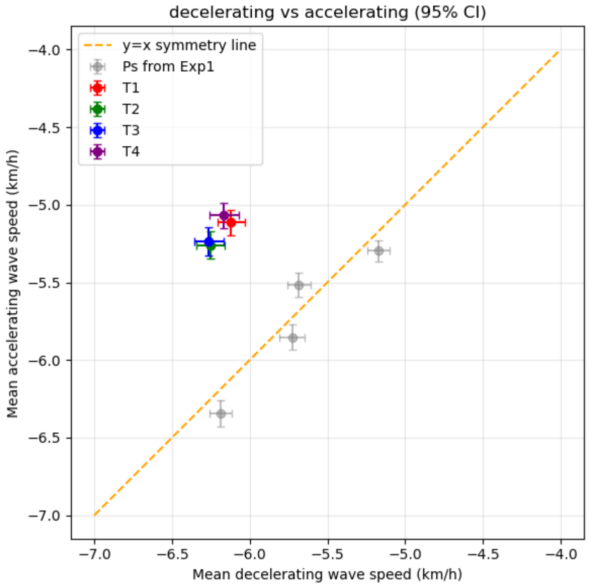


Figure 7.3: Wave speed comparison (Margins indicate the 95% CI.)

the breakdown front. Furthermore, their accelerating wave speeds are significantly lower than those of passenger cars, as trucks typically restart more slowly. In contrast, the decelerating wave speeds of truck platoons are overall slightly faster than those of passenger cars, which can be explained by the fact that, although trucks have inferior braking performance, truck drivers often benefit from higher vantage points and professional training, enabling them to anticipate traffic conditions and react earlier. This results in traffic waves that propagate somewhat faster in truck platoons.

Meanwhile, Table 7.4 reveals their characteristics from another angle. T4's fast but unstable reactions and short gaps trigger longer full stops, whereas T3's conservative spacing and early braking absorb disturbances and keep the standstill brief. T1 maintains tight yet steady following, yielding a moderate standstill, and T2 balances caution and responsiveness, falling between T1 and T3.

Table 7.4 shows that the earlier conclusion remains valid for truck agents: a conservative driving style is preferable, as it results in the lowest delay among all styles. The delay suggests that, when operating in clusters, adopting a stable conservative driving strategy (T3), such as a higher but stable time headway, can help reduce the overall delay in truck platoons.

The fuel consumption is overall higher in truck platoons than in passenger car platoons. Among them, the least-delay group 3-c exhibits the highest fuel consumption, which again indicates that the stable conservative group spends more time in the low-speed regime, thereby incurring greater energy consumption.

7.3.2. Impact of Truck Placement: Clustered or Dispersed

This experiment investigates the impact of truck allocation within a platoon. As summarized in Table 6.4, three allocation patterns (3-e, 3-f, 3-g, 3-h) are tested to simulate the influence of trucks on single-lane traffic.

The findings Table 7.5 indicate that clustering trucks within a platoon is a configuration with both potential benefits and significant risks. On the one hand, clustered formations can reduce delays under certain conditions; for instance, in our 16-vehicle experiments, placing the truck cluster in the middle of the platoon (3-e) resulted in relatively favorable delay consumption outcomes compared to alternating formations. However, this advantage comes with a critical limitation: the efficiency of clustering strongly depends on the exact position of the cluster (e.g., compare 3-e and 3-f). In real traffic, where truck positions cannot be precisely controlled, such benefits may not be consistently realized. Besides, the 3-e platoon exhibits the highest fuel consumption, likely because it remains trapped in the low-speed regime for extended periods.

On the other hand, **clustering substantially increases the standstill time of the platoon relative to dispersed arrangements**. This poses a serious safety concern, as longer standstill times imply that congestion may propagate further upstream and persist for longer durations. In single-lane traffic, therefore, clustered trucks should avoid being trapped in passenger-car-dense regions, since this could amplify breakdowns and delays rather than mitigate them.

By contrast, **dispersed formations generally produce more resilient traffic dynamics**. Although their delay performance is not as favorable as in 3-e, the lower standstill times indicate that breakdowns are less severe and recovery occurs more quickly. Considering that alternating patterns are relatively unsafe, the most practical and robust strategy for trucks—particularly under high-demand, single-lane conditions—is to remain randomly distributed across the platoon.

At the macroscopic level, however, when the overall proportion of trucks is fixed, no statistically significant differences are observed in the estimated fundamental diagram parameters (e.g., a, b, and k_{crit}) across allocation patterns. This indicates that macroscopic traffic characteristics and wave propagation are primarily determined by the truck proportion itself, rather than by the specific arrangement of trucks within the platoon.

Table 7.5: Difference of indicators compared to the reference level (reference group: 3-e, * denotes the difference is significant ($p < 0.01$) compared to the reference level)

Experiment	Composition	$\|w_{\text{dec}}\|$	$\|w_{\text{acc}}\|$	ST	Delay	FC
		(m/s)	(m/s)	(s)	(s)	(L/km)
		t-test		WLS + t-test		
3-e (clustered)	$[P\cdots]^4[T\cdots]^8[P\cdots]^4$	**6.0**	**5.4**	**442**	**632**	**4.25**
3-f (clustered)	$[T,T\cdots]^8[P,P\cdots]^8$	+0.05	+0.13*	+21*	+65*	-0.64*
3-g (dispersed)	$[T,P,T,P\cdots]^{16}$	-0.15*	+0.12	-35*	+9*	-1.08*
3-h (dispersed)	$\{T(50\%),P(50\%)\}^{16}$	-0.11*	+0.08	-43*	+0	-1.20*

7.4. Conclusion

In experiment 1, the following conclusions are drawn:

1. Aggressive drivers generate higher wave speeds (both starting and stopping), whereas conservative drivers produce slower waves. This implies that breakdowns induced by aggressive drivers extend farther in space but persist for shorter durations, and vice versa.

2. Stable drivers—whether aggressive or conservative—result in less delay, with the conservative–stable platoon performing best overall.
3. Aggressive drivers generally use less fuel than conservative drivers, suggesting that conservative drivers spend longer periods gliding at low speeds, thereby operating in less efficient conditions.
4. Starting waves are generally faster than stopping waves; however, unstable platoons face the risk of inversion, leading to more severe breakdowns.
5. Fundamental diagram analysis confirms style-dependent differences: stable groups are less prone to breakdown at capacity, owing to their higher critical density.

Based on the revealed driving styles, the following conclusions are drawn for experiment 2:

1. P3 drivers act as a *stabilizer*: the more P3 drivers in the platoon, the more favorable the impact on traffic flow. The reduction in delay is not strictly linear—once the proportion of P3 exceeds 50%, the decrease in delay becomes particularly significant. However, P3 also shows the poorest performance in terms of fuel consumption (FC), as its driving style tends to remain in low-speed regimes. Therefore, while P3 contributes to stability, its share in the platoon must be carefully considered.
2. P2 drivers act as a *drag*: as an unstable driving style, their presence prolongs standstill periods and substantially increases delay, making P2-dominated platoons the least efficient configurations.
3. P1 drivers exhibit the most favorable fuel efficiency. Consequently, a balanced mixture of P1 and P3—while keeping the share of P3 at or below 50%—offers a practical compromise, achieving congestion mitigation through stability while avoiding excessive fuel consumption.

Lastly, the following findings are revealed regarding the influence of trucks:

1. Generally, truck platoons generate faster decelerating than accelerating waves, causing congestion to extend further in both time and space and making breakdowns more difficult to dissipate compared with passenger-car platoons. Nevertheless, a stable and conservative driving style (T3) is recommended for homogeneous truck platoons, as it contributes to maintaining platoon stability.
2. Clustering trucks is a double-edged sword: while it can reduce delays if positioned favorably, it also leads to longer standstill times and greater fragility, making its benefits unreliable under realistic traffic conditions. This pattern is therefore more suitable under low traffic volumes. By contrast, dispersed trucks produce more resilient dynamics during breakdowns and recover more quickly, making this arrangement preferable in high-volume traffic.

Conclusion and Discussion

8

Conclusion

8.1. Part I Conclusion

This part presents a data-driven framework for modeling car-following (CF) behavior, incorporating driving style as a latent, time-independent factor. A clean and consistent dataset is first constructed through preprocessing steps, including segmentation, Kalman filtering, and the removal of invalid CF samples, to ensure high-quality learning input.

A Transformer-based style embedding model is then proposed to extract latent behavioral representations from vehicle trajectories. The embeddings are trained jointly with the CF model, ensuring functional relevance and behavioral consistency. Clustering and interpretation of these embeddings reveal their alignment with meaningful driving features such as average time headway, reaction time, and behavioral variability.

To evaluate model generalization beyond short-term prediction, a long-term recursive prediction framework is introduced. While all models are trained on short prediction windows, their performance is assessed over extended horizons. This long-term setup reveals the compounding nature of prediction errors, providing a more rigorous test of model robustness. The Style-aware CF outperforms baseline models—IDM, LSTM, and plain Transformer—achieving superior accuracy and stability over time. Its improved performance is attributed to the incorporation of style embeddings.

Furthermore, the learned embedding space is validated through crossover experiments, which demonstrate that altering the style token alone could systematically modulate car-following behavior. Moreover, the findings support the view that driving style lies on a continuous spectrum, rather than fitting into discrete categories such as aggressive versus conservative, as no clear boundaries between the styles are observed.

Overall, this framework integrates domain knowledge with neural modeling to achieve both performance and interpretability. It provides a foundation for simulating the impact of driver heterogeneity on traffic dynamics—an issue explored in the following part.

8.2. Part II Conclusion

This part builds on the style-aware car-following model and its extension to platoon-scale simulation to investigate traffic breakdown mechanisms. A series of experiments are designed with controlled platoon compositions and truck allocations, and their outcomes are evaluated using a set of microscopic indicators (delay, standstill time, wave speed) and macroscopic indicators (fundamental diagram calibration). Together, this methodological framework provides a unified and statistically robust approach to linking microscopic driver heterogeneity with macroscopic traffic flow dynamics.

Experiment 1 shows that differences in driving style fundamentally shape wave propagation and congestion dynamics. Drivers with more aggressive tendencies generate faster waves, causing congestion to spread further in space but to dissipate more quickly, whereas more conservative tendencies produce slower waves that persist longer. Stability emerges as the decisive factor: stable groups consistently lead to lower delays, with conservative–stable groups performing best overall. Yet this benefit comes at the expense of fuel efficiency, since conservative drivers tend to spend extended periods gliding at low speeds. Moreover, starting waves are generally faster than stopping waves, although unstable platoons risk wave inversion, which aggravates breakdowns.

Experiment 2 reveals a critical trade-off between stability and energy use. A higher proportion of stabilizing drivers reduces delays substantially, and the effect becomes particularly pronounced once their share surpasses half of the platoon. However, this comes with significantly higher fuel consumption, as stabilizing behavior often results in prolonged low-speed operation. By contrast, fuel-efficient drivers achieve the lowest energy use but provide less stability. Mixing the two styles offers a practical compromise: when the stabilizing type is kept at a moderate proportion, the platoon achieves both reduced delays and controlled fuel consumption.

Experiment 3 highlights the distinctive role of heavy vehicles. Compared with passenger-car platoons, truck groups tend to generate faster decelerating than accelerating waves, which makes congestion spread farther and persist longer. Nevertheless, a stable and conservative driving pattern remains beneficial for homogeneous truck platoons, helping to sustain overall stability. The spatial distribution of trucks within a mixed platoon also matters: clustering trucks can occasionally reduce delays but increases fragility and standstill times, making this arrangement less reliable under dense traffic. By contrast, dispersing trucks across the platoon enhances resilience and leads to faster recovery, offering a more robust strategy in high-volume conditions.

8.3. Contribution

Scientific

From a scientific perspective, this thesis demonstrates how microscopic heterogeneity in driver behavior systematically shapes macroscopic traffic outcomes. By embedding driving style into a car-following model and scaling up to controlled platoon experiments, it becomes possible to link behavioral traits to measurable effects such as delay, wave speed, standstill duration, and fuel consumption.

The results highlight three general mechanisms: (i) stability: stable and cautious behavior acts as a buffer against congestion, mitigating the formation and amplification of traffic waves. However, this benefit comes at the expense of higher fuel consumption, since stability is often achieved through prolonged low-speed operation; (ii) aggressiveness: more impatient or volatile driving tendencies accelerate the onset of breakdown and delay recovery. While such behavior may appear more fuel-efficient in isolation, it undermines overall traffic stability and increases vulnerability to stop–go oscillations; and (iii) style arrangement in platoon: the spatial distribution of heterogeneous driving styles plays a decisive role. Certain mixtures can balance delay and fuel efficiency, whereas unfavorable clustering—particularly of heavy vehicles—can amplify fragility. By contrast, dispersed arrangements improve resilience and allow the system to recover more quickly from disruptions.

Moreover, this thesis contributes several methodological advances that support both reproducibility and extensibility of traffic flow research. It first delivers a consistent and filtered version of the Zen-traffic dataset, together with its implementation, providing a reliable basis for car-following studies. Second, it introduces a style-aware car-following model and its implementation, enabling the integration of latent behavioral heterogeneity into predictive frameworks. Third, it develops a traffic-wave detection procedure and an automated method for generating platoon-based parallelograms to compute macroscopic variables, such as traffic flow and density, thereby forming a unified framework that can also be applied to calibrating fundamental diagrams from trajectory data.

Together, these findings provide a scientifically grounded framework for integrating behavioral heterogeneity into both microscopic traffic theory and macroscopic traffic management strategies.

Practical

The results also yield practical insights for traffic operations and everyday driving. For companies managing truck platoons, stability-oriented driving strategies should be encouraged, as they reduce breakdown risk and enhance predictability. However, clustering trucks tightly is not recommended in dense traffic, since this configuration increases fragility and standstill durations. A more robust strategy is to maintain moderate dispersion of trucks across the platoon, which improves recovery and reduces system-wide vulnerability under heavy demand.

For individual drivers, the findings suggest that stable conservative driving—characterized by prolonged low-speed gliding—may appear safe but is energetically inefficient and contributes to higher fuel consumption. Conversely, unstable or impatient behavior triggers stop–go waves that slow down everyone. The most effective practice lies in maintaining stable and moderate headways, avoiding unnecessary speed fluctuations, and responding smoothly to traffic changes. In other words, driving in a stable yet efficient manner not only reduces personal fuel use but also contributes to smoother traffic flow and fewer breakdowns at the collective level.

9

Discussion

9.1. Limitations

While this study advances the understanding of driving style and its impact on traffic breakdown through a data-driven modeling and simulation framework, several limitations should be acknowledged in terms of data coverage, methodological scope, and experimental design.

First, the training dataset may lack sufficient coverage of extreme cases, such as emergency braking. Although the car-following data were collected from real-world motorway trajectories, resulting in 44,207 pairs, the frequency of such rare events is relatively low and thus diluted in the dataset. Moreover, this study did not supplement the training data with synthesized corner cases. As a result, the model occasionally exhibits unrealistic behavior in simulation. To address this issue, the car-following and platoon simulations in this study regulate the model outputs by imposing constraints and overriding them when safety-critical violations occur, such as collisions with the leader.

Second, the driving style embedding is extracted from trajectory data through a data-driven style embedder model, which is jointly trained with the car-following model. Nevertheless, the classification in such a high-dimensional space may be subjective and sensitive to hyperparameter settings, such as the number of clusters. Although this study provides a plausible interpretation of each group using variables like time headway and reaction time (see Table 5.2) and relates them to relevant human factors (see Table 2.2), the interpretation does not fully capture the complexity of driving behavior and thus remains partly subjective.

Third, this study focuses exclusively on longitudinal car-following behavior. However, traffic breakdown is not solely determined by car-following but is also strongly influenced by lane-changing maneuvers. Driving style may also manifest in lane-changing decisions, such as gap acceptance or aggressiveness in merging, which were beyond the scope of this work. Nevertheless, these factors play a crucial role in congestion formation and propagation, and their omission limits the comprehensiveness of the current framework.

Lastly, a key limitation of the experimental design is its idealized nature. The platoon compositions were constructed according to predefined rules (e.g., homogeneous, alternating, or clustered), whereas in real traffic, vehicle distributions are highly random. The experiments also assumed that all vehicles remain in a strict car-following state, where each follower continuously responds to its leader. In practice, especially under free-flow conditions, such a configuration rarely occurs for an entire platoon. These simplifications inevitably create a gap between the experimental scenarios and actual traffic conditions. Nevertheless, the experiments still provide valuable insights by demonstrating that driving style exerts a substantial influence on overall traffic efficiency.

9.2. Future Work

Building on the limitations identified in this study, several avenues for future research can be pursued.

First, the scarcity of extreme cases in the training data could be mitigated by incorporating synthesized trajectories, for example generated by classical models such as IDM or other equivalent car-following formulations. Although such data would not originate from real-world observations, it would still be informative for the model to learn how to handle rare but safety-critical situations, such as emergency braking or near-collision scenarios. This approach would enhance the robustness of the model under corner cases that are underrepresented in naturalistic datasets.

Second, future work should aim to improve the clustering techniques used for driving style classification. The current framework relies on manual hyperparameter tuning, such as selecting the number of clusters, which introduces a degree of subjectivity. More advanced approaches could reduce this dependence by embedding the classification process directly into the training pipeline, thereby achieving a more consistent and precise representation of driving styles. In addition, the integration of interpretable machine learning techniques or external behavioral studies could further strengthen the explanatory power of the resulting clusters.

Third, this study only considers car-following, which is essentially a uni-dimensional maneuver. While car-following provides a plausible mechanism for traffic breakdown, it does not fully capture the complexity of real traffic dynamics. Lane-changing, for instance, can significantly alter the behavior of following vehicles and has been shown to contribute to traffic breakdown (Zheng et al., 2011). Therefore, extending the current framework to incorporate both longitudinal and lateral maneuvers would yield a more comprehensive understanding of style-dependent traffic phenomena.

Lastly, the experimental design could be made more realistic by relaxing the current idealized assumptions about platoon composition and vehicle interaction. While introducing additional randomness may obscure some of the causal effects observed in controlled settings, it would better reflect real-world conditions and provide more externally valid insights. Running experiments in larger and more heterogeneous traffic environments, including multi-lane setups and diverse vehicle classes, would thus represent a meaningful step toward translating the findings of this study into practical traffic management and control strategies.

Bibliography

Mark Brackstone and Mike McDonald. Car-following: a historical review. *Transportation Research Part F: Traffic Psychology and Behaviour*, 2(4):181–196, 1999. ISSN 1369-8478. doi: 10.1016/S1369-8478(00)00005-X.

Robert E Chandler, Robert Herman, and Elliott W Montroll. Traffic dynamics: studies in car following. *Operations research*, 6(2):165–184, 1958. doi: 10.1287/opre.6.2.165.

Danjue Chen, Soyoung Ahn, Jorge Laval, and Zuduo Zheng. On the periodicity of traffic oscillations and capacity drop: The role of driver characteristics. *Transportation Research Part B: Methodological*, 59:117–136, January 2014. ISSN 0191-2615. doi: 10.1016/j.trb.2013.11.005.

Kuan-Ting Chen and Huei-Yen Winnie Chen. Driving Style Clustering using Naturalistic Driving Data. *Transportation Research Record*, 2673(6):176–188, June 2019. ISSN 0361-1981. doi: 10.1177/0361198119845360. Publisher: SAGE Publications Inc.

Linsen Chong, Montasir M. Abbas, Alejandra Medina Flintsch, and Bryan Higgs. A rule-based neural network approach to model driver naturalistic behavior in traffic. *Transportation Research Part C: Emerging Technologies*, 32:207–223, July 2013. ISSN 0968-090X. doi: 10.1016/j.trc.2012.09.011.

Chiara Colombaroni and Gaetano Fusco. Artificial Neural Network Models for Car Following: Experimental Analysis and Calibration Issues. *Journal of Intelligent Transportation Systems*, 18(1):5–16, January 2014. ISSN 1547-2450. doi: 10.1080/15472450.2013.801717.

L. C. Edie. Discussion of traffic stream measurements and definitions. pages 139–154, 1965.

Maosi Geng, Junyi Li, Yingji Xia, and Xiqun (Michael) Chen. A physics-informed Transformer model for vehicle trajectory prediction on highways. *Transportation Research Part C: Emerging Technologies*, 154:104272, September 2023. ISSN 0968-090X. doi: 10.1016/j.trc.2023.104272.

P.G. Gipps. A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological*, 15(2):105–111, 1981. ISSN 0191-2615. doi: 10.1016/0191-2615(81)90037-0.

Bryan Higgs and Montasir Abbas. Segmentation and clustering of car-following behavior: Recognition of driving patterns. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):81–90, 2015. doi: 10.1109/TITS.2014.2326082.

Sepp Hochreiter and Jurgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735.

Xiuling Huang, Jie Sun, and Jian Sun. A car-following model considering asymmetric driving behavior based on long short-term memory neural networks. *Transportation Research Part C: Emerging Technologies*, 95:346–362, 2018. ISSN 0968-090X. doi: 10.1016/j.trc.2018.07.022.

Rudolph E Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

Jorge A. Laval and Ludovic Leclercq. A mechanism to describe the formation and propagation of stop-and-go waves in congested freeway traffic. *Philosophical Transactions of the Royal Society A*, 368 (1928):4519–4541, 2010. doi: 10.1098/rsta.2010.0138.

Gentry Lee. A generalization of linear car-following theory. *Oper. Res.*, 14(4):595–606, August 1966. ISSN 0030-364X. doi: 10.1287/opre.14.4.595.

Wilhelm Leutzbach and Rainer Wiedemann. Development and applications of traffic simulation models at the karlsruhe institut fuer verkehrswesen. *Traffic Engineering and Control*, 27(5):270 – 273, 275, 1986.

Lijing Ma and Shiru Qu. A sequence to sequence learning based car-following model for multi-step predictions considering reaction delay. *Transportation Research Part C: Emerging Technologies*, 120:102785, November 2020. ISSN 0968-090X. doi: 10.1016/j.trc.2020.102785.

Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, 1996. ISBN 0-262-13316-4. doi: doi.org/10.7551/mitpress/3927.001.0001.

Zhaobin Mo, Rongye Shi, and Xuan Di. A physics-informed deep learning paradigm for car-following models. *Transportation Research Part C: Emerging Technologies*, 130:103240, September 2021. ISSN 0968-090X. doi: 10.1016/j.trc.2021.103240.

Marcello Montanino and Vincenzo Punzo. Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns. *Transportation Research Part B: Methodological*, 80: 82–106, 2015. ISSN 0191-2615. doi: 10.1016/j.trb.2015.06.010.

Jeremy Morton, Tim A. Wheeler, and Mykel J. Kochenderfer. Analysis of Recurrent Neural Networks for Probabilistic Modeling of Driver Behavior. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1289–1298, May 2017. ISSN 1558-0016. doi: 10.1109/TITS.2016.2603007. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

G. F. Newell. A simplified car-following theory: a lower order model. *Transportation Research Part B: Methodological*, 36(3):195–205, March 2002. ISSN 0191-2615. doi: 10.1016/S0191-2615(00) 00044-8.

Saskia Ossen and Serge P. Hoogendoorn. Heterogeneity in car-following behavior: Theory and empirics. *Transportation Research Part C: Emerging Technologies*, 19(2):182–195, 2011. ISSN 0968-090X. doi: 10.1016/j.trc.2010.05.006. Emerging theories in traffic and transportation and methods for transportation planning and operations.

Vasileia Papathanasopoulou and Constantinos Antoniou. Towards data-driven car-following models. *Transportation Research Part C: Emerging Technologies*, 55:496–509, 2015. ISSN 0968-090X. doi: 10.1016/j.trc.2015.02.016. Engineering and Applied Sciences Optimization (OPT-i) - Professor Matthew G. Karlaftis Memorial Issue.

Hesham Rakha, Kyoungho Ahn, and Antonio Trani. Development of VT-Micro model for estimating hot stabilized light duty vehicle and truck emissions. *Transportation Research Part D: Transport and Environment*, 9(1):49–74, 2004. ISSN 1361-9209. doi: https://doi.org/10.1016/S1361-9209(03)00054-3.

Thomas A. Ranney. Psychological factors that influence car-following and car-following model development. *Transportation Research Part F: Traffic Psychology and Behaviour*, 2(4):213–219, December 1999. ISSN 1369-8478. doi: 10.1016/S1369-8478(00)00010-3.

Rijkswaterstaat. Rapportage Rijkswegennet. 3e periode 2023, 1 september - 31 december. Technical report, Rijkswaterstaat, 2024. In Dutch.

Fridulv Sagberg, Selpi, Giulio Francesco Bianchi Piccinini, and Johan Engström. A Review of Research on Driving Styles and Road Safety. *Human Factors*, 57(7):1248–1275, November 2015. ISSN 0018-7208. doi: 10.1177/0018720815591313. Publisher: SAGE Publications Inc.

Mohammad Saifuzzaman and Zuduo Zheng. Incorporating human-factors in car-following models: A review of recent developments and research needs. *Transportation Research Part C: Emerging Technologies*, 48:379–403, November 2014. ISSN 0968-090X. doi: 10.1016/j.trc.2014.09.008.

Hiroaki Sakoe. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26:159–165, 1978. doi: 10.1109/TASSP.1978.1163055.

Anshuman Sharma, Zuduo Zheng, and Ashish Bhaskar. A pattern recognition algorithm for assessing trajectory completeness. *Transportation Research Part C: Emerging Technologies*, 96:432–457, November 2018. ISSN 0968-090X. doi: 10.1016/j.trc.2018.09.027.

Fulvio Simonelli, Gennaro Nicola Bifulco, Valerio De Martinis, and Vincenzo Punzo. Human-like adaptive cruise control systems through a learning machine approach. In *Applications of Soft Computing*, pages 240–249, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-540-88079-0.

Jonghae Suh and Hwasoo Yeo. An empirical study on the traffic state evolution and stop-and-go traffic development on freeways. *Transportmetrica A: Transport Science*, 12(1):80–97, 2016. doi: 10.1080/23249935.2015.1101508.

SWOV. Traffic congestion and roadworks. *SWOV-Fact sheet, May 2022*, May 2022. Place: The Hague Publisher: SWOV.

Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev. E*, 62:1805–1824, Aug 2000. doi: 10.1103/PhysRevE.62.1805.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Xiao Wang, Rui Jiang, Li Li, Yi-Lun Lin, and Fei-Yue Wang. Long memory is important: A test study on deep-learning based car-following model. *Physica A: Statistical Mechanics and its Applications*, 514:786–795, January 2019. ISSN 0378-4371. doi: 10.1016/j.physa.2018.09.136.

S. K. Zegeye, B. De Schutter, J. Hellendoorn, E. A. Breunesse, and A. Hegyi. Integrated macroscopic traffic flow, emission, and fuel consumption model for control purposes. *Transportation Research Part C: Emerging Technologies*, 31:158–171, 2013. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2013.01.002.

Yifan Zhang, Xinhong Chen, Jianping Wang, Zuduo Zheng, and Kui Wu. A Generative Car-following Model Conditioned On Driving Styles. *Transportation Research Part C: Emerging Technologies*, 145:103926, December 2022. ISSN 0968090X. doi: 10.1016/j.trc.2022.103926. arXiv:2112.05399 [cs].

Zuduo Zheng, Soyoung Ahn, Danjue Chen, and Jorge Laval. Freeway traffic oscillations: Microscopic analysis of formations and propagations using wavelet transform. *Transportation Research Part B: Methodological*, 45(9):1378–1388, 2011. ISSN 0191-2615. doi: 10.1016/j.trb.2011.05.012.

Mofan Zhou, Xiaobo Qu, and Xiaopeng Li. A recurrent neural network based microscopic car following model to predict traffic oscillation. *Transportation Research Part C: Emerging Technologies*, 84: 245–264, 2017. ISSN 0968-090X. doi: 10.1016/j.trc.2017.08.027.

Meixin Zhu, Xuesong Wang, and Yinhai Wang. Human-like autonomous car-following model with deep reinforcement learning. *Transportation Research Part C: Emerging Technologies*, 97:348–368, December 2018. ISSN 0968-090X. doi: 10.1016/j.trc.2018.10.024.

Appendix

A

Methodology

A.1. Variable Definition

The kinematic variables recorded in the data are in a discrete format, and their physical meanings may differ depending on the context. For instance, the speed may refer to average speed during the time interval, or the instantaneous speed at the recorded time instance. This nuance may sound trivial, but it implies two different ways to derive the position based on their values. Therefore, an unambiguous definition of these variables is necessary for the following work.

In Figure A.1, we present a discrete representation of key kinematic definitions that aligns with the data storage format in our dataset. The position x corresponds to the **instantaneous position** as recorded in the dataset. Velocity is defined as the **average velocity** over each time interval using forward differentiation, while acceleration is defined as the **discrete forward differentiation between consecutive velocity values**.

For car-following prediction models, the acceleration series typically serves as the output variable. Based on our operational definitions, we derive position from acceleration through the following equations:

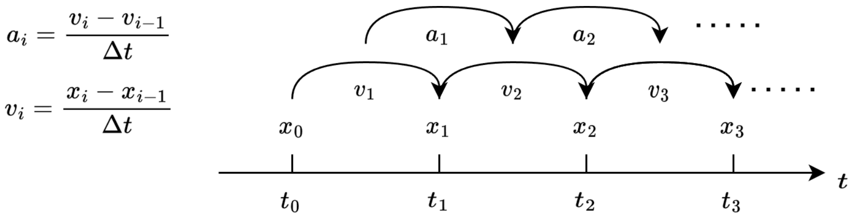

$a_i = \frac{v_i - v_{i-1}}{\Delta t}$
$v_i = \frac{x_i - x_{i-1}}{\Delta t}$
a_1
a_2
v_1
v_2
v_3
x_0
x_1
x_2
x_3
t
t_0
t_1
t_2
t_3

Figure A.1: Relationships between kinematic variables

$$v_t = v_0 + \Delta t \sum_{i=1}^{t} a_i \tag{A.1}$$

$$x_t = x_0 + \Delta t \sum_{i=1}^{t} v_i \tag{A.2}$$

Notably, since velocity is defined as an average quantity instead of instantaneous velocity, the position computation doesn't involve acceleration directly. This formulation maintains consistency with our discrete data representation while preserving the physical interpretation of kinematic relationships.

Using this definition, the output, which is in acceleration, can be derived to a lower-order variable, such as velocity and position. These two variables are easier and intuitive to evaluate.

A.2. Kalman Filter

The Kalman Filter (Kalman, 1960) is an efficient algorithm that estimates the state of a dynamic system by combining predictions from a system model with noisy measurements. It operates in two steps: prediction (forecasting the next state) and update (correcting with new data). This recursive process provides optimal estimates while tracking uncertainty. Widely used in navigation, robotics, and tracking, it handles noisy data effectively and works in real-time without storing past measurements.

This study employs a Kalman filter to address significant data inconsistency issues in the dataset. By optimally fusing noisy measurements with kinematic model predictions, the filter produces a refined dataset with reduced noise and improved physical consistency. The approach mitigates the observed inconsistencies while preserving the underlying motion characteristics.

The matrix format of the model is more accessible to the filter. According to kinematic model (Equation A.2), the corresponding matrix format can be easily transformed (Equation A.3):

$$\begin{bmatrix} x_t^{KF} \\ v_t^{KF} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ v_{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} a_{t-1} \tag{A.3}$$

$$\text{where} \quad \boldsymbol{s}_t = \begin{bmatrix} x_t \\ v_t \end{bmatrix} \quad F = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} \tag{A.4}$$

$$\tag{A.5}$$

where $\boldsymbol{s}$ is the state of the filter, including speed and position. Note that the Kalman filter is a recursive model, which doesn't store any previous steps in memory. To meet this requirement, the model is adapted into the recursive format. After obtaining the model in matrix format, there are still two core steps to set the filter up.

Prediction uses the output from the very last step as input of the model.

$$\boldsymbol{s}_t^- = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \boldsymbol{s}_{t-1} + \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} a_{t-1} \tag{A.6}$$

After deriving the estimated state, the covariance matrix is necessary for the next step.

$$P_t^- = FP_{t-1}F^T + Q \tag{A.7}$$

$$\text{where} \quad P_t = \begin{bmatrix} \sigma_x^2 & \sigma_{xv} \\ \sigma_{vx} & \sigma_v^2 \end{bmatrix} \quad Q = \begin{bmatrix} q_x^2 & 0 \\ 0 & q_v^2 \end{bmatrix} \tag{A.8}$$

P_t depicts the belief in the estimated state, and reveals their correlations as well. The Kalman filter requires an initialization of it, which is the initial belief in the states. The parameter setting will be introduced later in this section. Q indicates the noise along the process, which is another parameter that needs to be specified in the setup.

Update step involves the computation of the Kalman Gain K_t, which minimizes the P_t, and it indicates which value, either prediction or measurement, is more reliable.

$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1} \tag{A.9}$$

$$\text{where} \quad H = \begin{bmatrix} h_x \\ h_v \end{bmatrix} \quad R = \begin{bmatrix} r_x^2 & r_{xv} \\ r_{vx} & r_v^2 \end{bmatrix} \tag{A.10}$$

h_x and h_v indicate the observability of position and speed. And R indicates the presumed noise in the dataset, which is a parameter required in the setup.

State update

$$\boldsymbol{s}_t = \boldsymbol{s}_t^- + K_t \left(\boldsymbol{z}_t - H\boldsymbol{s}_t^- \right) \tag{A.11}$$

Covariance Update

$$P_t = (I - K_t H) P_t^- \tag{A.12}$$

In the update stage, the state, $\boldsymbol{s}_t$, which will be used as the input in the next timestep, is required to be derived as well. In the meantime, as the filter assumes the kinematic model F, it could propagate noises from one variable to another, which requires the state covariance matrix P to be updated, where I is the identity matrix.

Notably, this model only involves speed and position (retrieved from Table 4.4), which runs along the motorway, without taking the geometry of the road into account. In other words, these variables are scalars. In the meantime, the Kalman filter is applied to each vehicle, both the leader and the follower independently.

A.3. Dynamic Time Warping

DTW (Dynamic Time Warping) is a technique that aligns two temporal sequences by warping the time axis to find an optimal match between them, even if they vary in speed or timing. This flexibility allows DTW to effectively compare vehicle trajectories or driver behavior patterns that are not perfectly

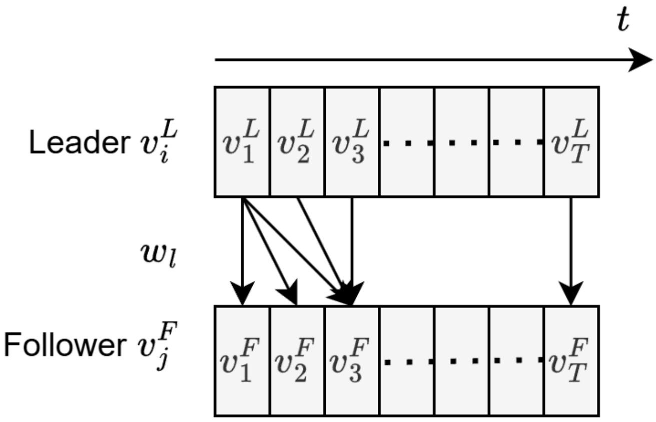


Figure A.2: Illustration of DTW algorithm

synchronized. By applying DTW to the leader and follower trajectories, it is possible to estimate the reaction time as the temporal offset that yields the best alignment between the two sequences.

Dynamic Time Warping (DTW) is a widely used algorithm for measuring the similarity between two temporal sequences that may vary in speed or timing (Sakoe, 1978). Unlike simple point-to-point distance metrics, DTW allows flexible alignment by stretching or compressing the time axis of the sequences to find the best possible match.

Formally, given two sequences $X = (x_1, x_2, \ldots, x_N)$ and $Y = (y_1, y_2, \ldots, y_M)$, DTW computes a warping path $W = (w_1, w_2, \ldots, w_L)$, where each $w_l = (i, j)$ represents an alignment between element x_i in X and element y_j in Y. The warping path is constrained to start at $(1, 1)$, end at (N, M), and be monotonically increasing in both indices. The optimal warping path minimizes the cumulative distance:

$$DTW(X, Y) = \min_W \sum_{l=1}^{L} d(x_{i_l}, y_{j_l}),$$

where $d(\cdot, \cdot)$ is the Euclidean distance between points.

Reaction time estimation: In car-following scenarios, the warping path from DTW can be used to estimate the reaction time of the follower. For each time step t in the leader's trajectory, we identify the set of aligned time steps in the follower's trajectory based on the warping path:

$$w_t = \{(t, j) \in W\}.$$

The average temporal shift between the aligned indices reflects the delay in the follower's response. The estimated reaction time at time t is defined as:

$$\tau_t = \frac{1}{|w_t|} \sum_{(t,j) \in w_t} \Delta t \cdot (j - t),$$

where Δt is the sampling interval. This formulation provides a data-driven and robust way to quantify the temporal delay without requiring explicit labeling of reaction events.

This approach provides a robust way to quantify reaction time from trajectory data without requiring explicit recording of driver responses.

A.4. Style-CF Model Architecture

A.4.1. Style Embedder

The architecture of the style embedder (left half in Figure 3.1) is shown in Table A.1. The module takes time series tensors as input, in this thesis, $F_{\text{style}} = (\tau, h, \Delta x, v^F)$. As a result, the module outputs a style embedding after attention pooling, which aggregates the information over time.

Table A.1: Architecture of the `Style Embedder` module

Component	**Input → Output**	**Description**
Input projection	$(B, T_{\text{hist}}, F_{\text{Style}}) \to (B, T, 256)$	Linear projection to embedding space.
Transformer encoder ($L = 2$, $H = 8$)	$(B, T_{\text{hist}}, 256) \to (B, T_{\text{hist}}, 256)$	Two-layer Transformer encoder with 8 heads, capturing temporal dependencies.
Time scoring (`time_fc`)	$(B, T_{\text{hist}}, 256) \to (B, T_{\text{hist}}, 1)$	Linear layer generating attention scores for each time step.
Attention pooling	$(B, T_{\text{hist}}, 256) \to (B, 256)$	Weighted sum over time using softmax-normalized attention weights.
Output projection	$(B, 256) \to (B, 256)$	Linear mapping followed by ℓ_2 normalization.

A.4.2. CF Transformer

The CF Transformer part (right half in Figure 3.1) adopts an encoder–decoder Transformer structure conditioned on a driving-style token. Table A.2 summarizes the architecture of its key components.

A.5. Benchmark Models

A.5.1. LSTM Model

The Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) model (Figure A.3) is a type of recurrent neural network (RNN) capable of capturing long-range temporal dependencies in sequential data. In traffic prediction tasks (Huang et al., 2018), LSTM can model the evolution of vehicle trajectories by learning from past motion sequences. Its memory cell structure allows it to preserve important temporal patterns over time.

The Long Short-Term Memory (LSTM) network processes sequential data through the following equations at each time step t:

Table A.2: Architecture details of the `CF Transformer` model (used in Style-CF and plain Transformer CF)

Module	Architecture	Description and Output
Embedder	Linear layer: $(d_{\text{in}} \to d_{\text{model}})$ + Shared positional embedding: Embedding$(T_{\text{enc\|dec}}, d_{\text{model}})$	(Both encoder and decoder embedder) Projects historical input sequence into d_{model} space and adds temporal positional encodings. The embedded sequence is concatenated with the style token before entering the encoder.
Encoder	`nn.TransformerEncoder` (from `nn.Transformer`) Layers: $N_{\text{enc}} = 1$ Attention heads: 8 Feedforward dim: 1024 Activation: ReLU, Dropout: 0	Processes the concatenated encoder embeddings and the style token to extract temporal features and context representation (memory) for the decoder.
Decoder	`nn.TransformerDecoder` (from `nn.Transformer`) Layers: $N_{\text{dec}} = 1$ Attention heads: 8 Feedforward dim: 1024 Activation: ReLU, Dropout: 0 Causal mask applied on target sequence	Generates future trajectory predictions autoregressively, conditioned on encoder memory and the style token. The output is projected via a linear layer $(d_{\text{model}} \to 1)$ to produce the final predicted variable (e.g., speed).

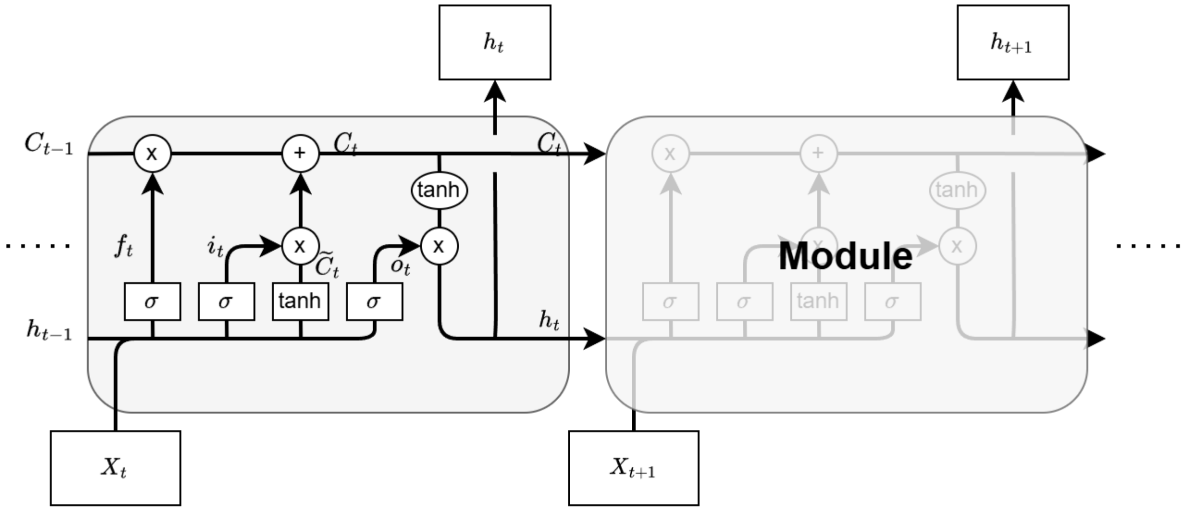


Figure A.3: Repeating Module in LSTM

$$
\begin{aligned}
i_t &= \sigma(W_i[h_{t-1}, x_t] + b_i) \quad \text{(Input gate)} \\
f_t &= \sigma(W_f[h_{t-1}, x_t] + b_f) \quad \text{(Forget gate)} \\
\tilde{C}_t &= \tanh(W_C[h_{t-1}, x_t] + b_C) \\
C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad \text{(Cell state)} \\
o_t &= \sigma(W_o[h_{t-1}, x_t] + b_o) \quad \text{(Output gate)} \\
h_t &= o_t \odot \tanh(C_t)
\end{aligned}
$$

Notation: σ=sigmoid, $\odot$=element-wise multiply, $\tanh$=hyperbolic tangent, h=hidden state, x=input, C=cell state, W_*=weight matrices, b_*=bias vectors, $[\cdot, \cdot]$=vector concatenation, subscript t=current timestep, subscript $t{-}1$=previous timestep.

The diagram (Figure A.3) shows the core structure of an LSTM cell, which processes sequential data using gating mechanisms. It maintains a cell state (C) for long-term memory and a hidden state (h) for short-term output. Three gates—input, forget, and output—use sigmoid activations to control information flow, while a $tanh$ function generates candidate values. This structure helps LSTMs capture long-range dependencies and avoid vanishing gradients, with the same cell repeated across time steps.

The architecture of LSTM is summarized in the Table A.3.

Table A.3: Architecture of the LSTM.

Component	**Configuration**
Hidden state size h_t	64
Number of LSTM layer	1
Input x_t	$[v^F, \Delta v, \Delta x, l^F, l^L]$
Output y_t	a_t^F (aggregated along hidden state axis)

A.5.2. Transformer Model

The overall Transformer architecture is shown in Figure A.4. It consists of an encoder-decoder structure, where both components are composed of N stacked layers. Each layer includes a multi-head attention mechanism and a feed-forward sublayer, both of which are wrapped with residual connections and layer normalization. The feed-forward sublayer consists of two linear transformations with a ReLU activation in between.

To incorporate temporal information, positional encodings are added to both the encoder and decoder inputs. In this study, we adopt learned positional embeddings rather than fixed sinusoidal encodings, in order to increase the model's flexibility and representational capacity.

The encoder processes the historical vehicle trajectory, generating contextualized representations through self-attention and feed-forward operations. The decoder, on the other hand, receives the forecast window and applies masked self-attention to prevent access to future time steps, thereby ensuring causal predictions. It then performs cross-attention over the encoder outputs, followed by its own feed-forward layers. Finally, the decoder output is passed through a linear projection layer to produce the predicted acceleration sequence.

Attention Mechanism

The attention mechanism (Vaswani et al., 2017) dynamically computes the relevance of each input token to others in a sequence. For a given set of queries Q, keys K, and values V matrices (Figure A.5), the attention weights are computed as:

$$A = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right), \tag{A.13}$$

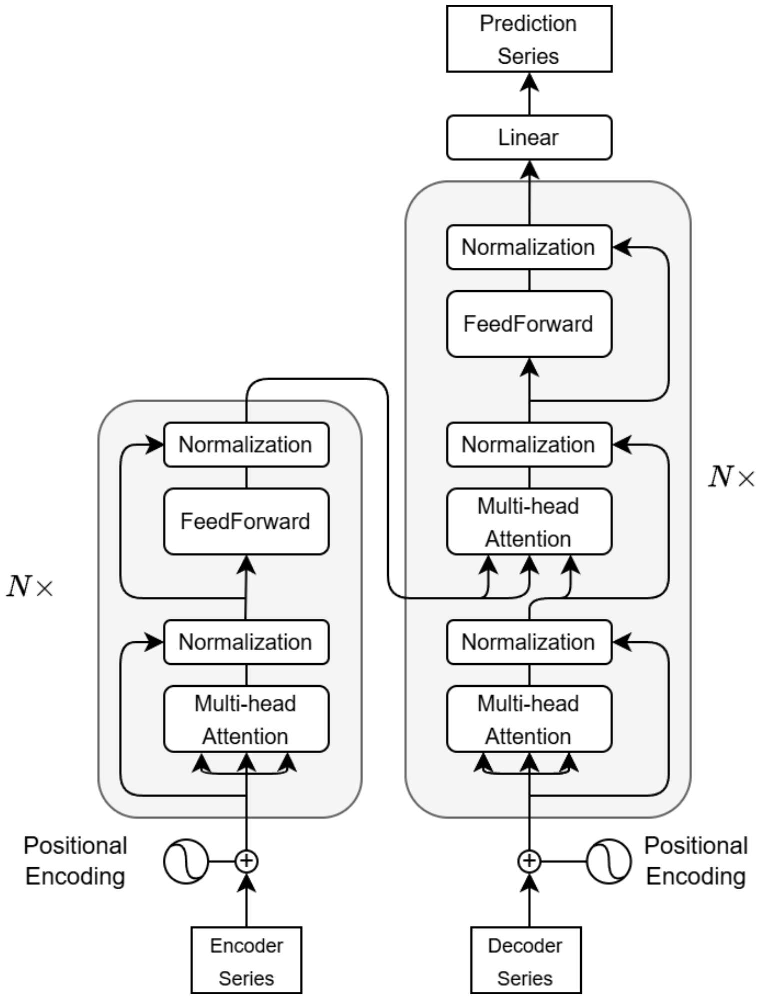


Figure A.4: Architecture of the Transformer model

These weights determine how much focus the model should place on different time steps when making predictions. Multi-head attention extends this mechanism by projecting the inputs into multiple subspaces, allowing the model to learn diverse types of temporal relationships in parallel.

In the context of car-following tasks, attention weights offer interpretability by revealing which past time steps most influence the predicted vehicle behavior, potentially aligning with the driver's reaction time.

Causality and Masking

To ensure that future information is not leaked during prediction, causal masking (Vaswani et al., 2017) is applied to the decoder's attention layers. This prevents each time step from attending to future tokens. Mathematically, the masked attention is computed as:

$$A = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}} + M\right), \tag{A.14}$$

where M is an upper-triangular mask matrix defined as:

$$M_{i,j} = \begin{cases} 0, & j \leq i, \\ -\infty, & j > i, \end{cases}$$

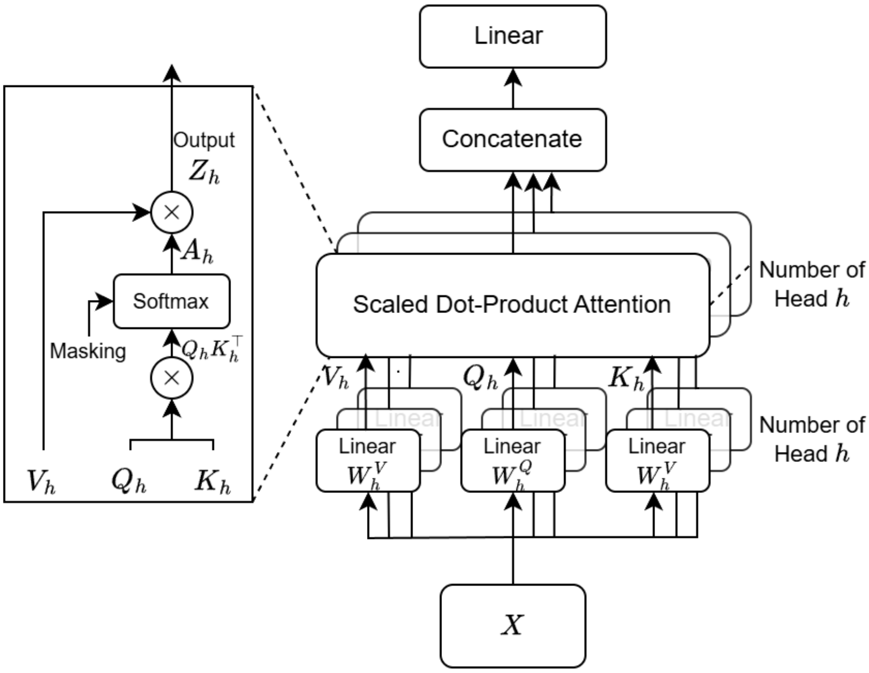


Figure A.5: The concept of attention mechanism

This masking strategy ensures that each decoder step only attends to itself and preceding positions. In our model, masking is only applied in the decoder, as the encoder processes fully observed historical data and does not require temporal restrictions.

A.5.3. Intelligent Driver Model

The Intelligent Driver Model (IDM) is a car-following model that describes the longitudinal acceleration of a vehicle based on its own speed, the distance to the leading vehicle, and the relative velocity. The acceleration a is given by:

$$a_t^F = a_{\max}\left[1 - \left(\frac{v_t^F}{v_0^F}\right)^\delta - \left(\frac{s^*(v_t^F, \Delta v_t)}{\Delta x_t}\right)^2\right] \tag{A.15}$$

where:

- v_t^F: current velocity of the follower vehicle
- v_0^F: desired velocity of the follower
- $a_{\max}$: maximum acceleration
- δ: acceleration exponent
- Δx_t: actual gap to the leading vehicle at time t
- $\Delta v_t = v_t^F - v_t^L$: relative velocity to the leading vehicle

The desired dynamic gap $s^*(v_t^F, \Delta v_t)$ is defined as:

$$s^*(v_t^F, \Delta v_t) = s_0 + v_t^F T + \frac{v_t^F \Delta v_t}{2\sqrt{a_{\max} b}} \tag{A.16}$$

where:

- s_0: minimum spacing
- T: desired time headway
- b: comfortable deceleration

This formulation allows the vehicle to accelerate smoothly toward the desired speed and decelerate safely when approaching slower traffic.

A.6. VT-Micro Fuel Consumption Estimation

The VT-Micro model (Rakha et al., 2004) is a microscopic emission and fuel consumption model that predicts the instantaneous rate of a vehicle based on its second-by-second speed $v(t)$ and acceleration $a(t)$. The model is formulated as an exponential function of polynomial terms of $v(t)$ and $a(t)$:

$$J_{\mathrm{FC}}(t) = \exp\left(\tilde{v}(t)^{\top} P_{\mathrm{FC}} \tilde{a}(t)\right), \tag{A.17}$$

where $J_{\mathrm{FC}}(t)$ denotes the estimated instantaneous fuel consumption rate, $\tilde{v}(t) = [1, v(t), v(t)^2, v(t)^3]^{\top}$ and $\tilde{a}(t) = [1, a(t), a(t)^2, a(t)^3]^{\top}$ are polynomial expansions of speed and acceleration, and P_{FC} is the parameter matrix calibrated from chassis dynamometer data.

In this study, the VT-Micro model is applied with the calibrated parameter matrix for fuel consumption (P_{FC}) as provided in Zegeye et al. (2013). The matrix used is

$$P_{\mathrm{FC}} = 0.01 \begin{bmatrix} -753.7 & 44.3809 & 17.1641 & -4.2024 \\ 9.7326 & 5.1753 & 0.2942 & -0.7068 \\ -0.3014 & -0.0742 & 0.0109 & 0.0116 \\ 0.0053 & 0.0006 & -0.0010 & -0.0006 \end{bmatrix}, \tag{A.18}$$

which corresponds to passenger vehicles and light trucks, consistent with the vehicle types considered in this study.

A.7. Automatic Construction of Parallelograms Along a Platoon

To evaluate local traffic states in a physically meaningful way, the observation window must align with the underlying wave propagation and vehicle motion. Instead of using a fixed rectangular window in the time–space domain, we construct parallelogram whose orientation follows the characteristic directions of the traffic flow.

Specifically, the long edge of the parallelogram aligns with the wave propagation direction (slope w), while the short edge aligns with the representative vehicle trajectory (slope v_{plat}). The temporal projection of the short edge H_t is fixed by the user (defining the observation duration), whereas the spatial projection of the long edge L_x is automatically determined from the platoon geometry. By iterating over the simulation time, a series of parallelograms is generated, each representing a localized spatio-temporal region for computing Edie's flow, density, and speed. See Algorithm 3 for details.

Algorithm 3 Automatic Construction of Moving Parallelograms

```
Input: Platoon trajectories $traj \in \mathbb{R}^{N \times T \times 3}$; sampling interval $\Delta t$; wave speed $w$; fixed temporal projection of short edge $H_t$; time range $[t_{\min}, t_{\max}]$; step size $\Delta T$.
Output: Sequence of parallelograms $\mathcal{P} = \{P_1, P_2, \dots, P_M\}$.
Initialize $\mathcal{P} \leftarrow [\,]$
for $t_c \leftarrow t_{\min}$ to $t_{\max}$ step $\Delta T$ do
    Obtain $x_{\text{head}}(t_c)$ and $x_{\text{tail}}(t_c)$ from $traj$
    Compute center $x_c \leftarrow \frac{1}{2}\big(x_{\text{head}}(t_c) + x_{\text{tail}}(t_c)\big)$
    Define wave direction $\vec{v}_{\text{wave}} \leftarrow (1, w)$
    Find intersection points $(t_h, x_h)$ and $(t_t, x_t)$ along $\vec{v}_{\text{wave}}$ with head/tail trajectories
    $L_x \leftarrow |x_h - x_t|$
    Update center: $t_c \leftarrow \frac{1}{2}(t_h + t_t)$, $x_c \leftarrow \frac{1}{2}(x_h + x_t)$
    Estimate representative speed $v_{\text{plat}} \leftarrow \text{mean}(v_{\text{head}}, v_{\text{tail}})$
    Construct parallelogram $P$ using the center $(t_c, x_c)$, fixed temporal projection of short edge $H_t$, and spatial projection of long edge $L_x$
        The long edge is aligned with the wave direction $(1, w)$, and the short edge with the vehicle direction $(1, v_{\text{plat}})$:
        $P \leftarrow$ generate parallelogram$(t_c, x_c, H_t, L_x, w, v_{\text{plat}})$
    Append $P$ to $\mathcal{P}$
end for
return $\mathcal{P}$
```

A.8. Wave Detection Algorithm

To quantify how traffic waves propagate through a platoon, we develop an automated detection algorithm (Algorithm 4) that extracts individual wave events from trajectory data and reconstructs complete wave sequences. This enables a consistent estimation of both decelerating and accelerating wave speeds across vehicles.

Algorithm 4 Wave Detection and Chaining

```
Require: Vehicle trajectories $\{x_i(t), v_i(t)\}$, parameters $(W_s, W_\ell, \tau_{\max}, L_{\min})$

Stage 1: Regime-change detection
for $i = 1$ to $N$ do
    Smooth $v_i(t)$ with window $W_s$
    Detect regime-change times $\mathcal{T}_i$ using window $W_\ell$
end for

Stage 2: Event linking and chain propagation
Initialize an empty list of chains $\mathcal{C}$
for $i = 1$ to $N - 1$ do
    for each event $\tau_i \in \mathcal{T}_i$ not yet assigned to any chain do
        Initialize a new chain $C \leftarrow \{(i, \tau_i)\}$
        Set $j \leftarrow i$
        while there exists $\tau_{j+1} \in \mathcal{T}_{j+1}$ s.t. $|\tau_{j+1} - \tau_j| \leq \tau_{\max}$ do
            Append $(j + 1, \tau_{j+1})$ to $C$
            Mark $\tau_{j+1}$ as used
            Update $j \leftarrow j + 1$
        end while
        Add chain $C$ to $\mathcal{C}$
    end for
end for

Stage 3: Chain filtering and wave characterization
Keep only chains with length $\geq L_{\min}$
For each chain $C$, compute wave speed $w = \frac{\Delta x_{\text{tot}}}{\Delta t_{\text{tot}}}$
return All valid wave chains and their speeds $w$
```

B

Overview of Field Data

Field	Unit	Description
Timestamp	0.1s	Each CF trajectory pair lasts for 30 seconds
Leader ID	-	-
Leader Position	meter	Relative position originated from the first timestep of the trajectory
Leader Speed	m/s	-
Leader Length	m	The length of the leader vehicle
Follower ID	-	-
Follower Position	meter	Relative position originated from the first timestep of the trajectory
Follower Speed	m/s	-
Follower Length	m	The length of the follower vehicle

Table B.1: Example of one sample from the extracted dataframe

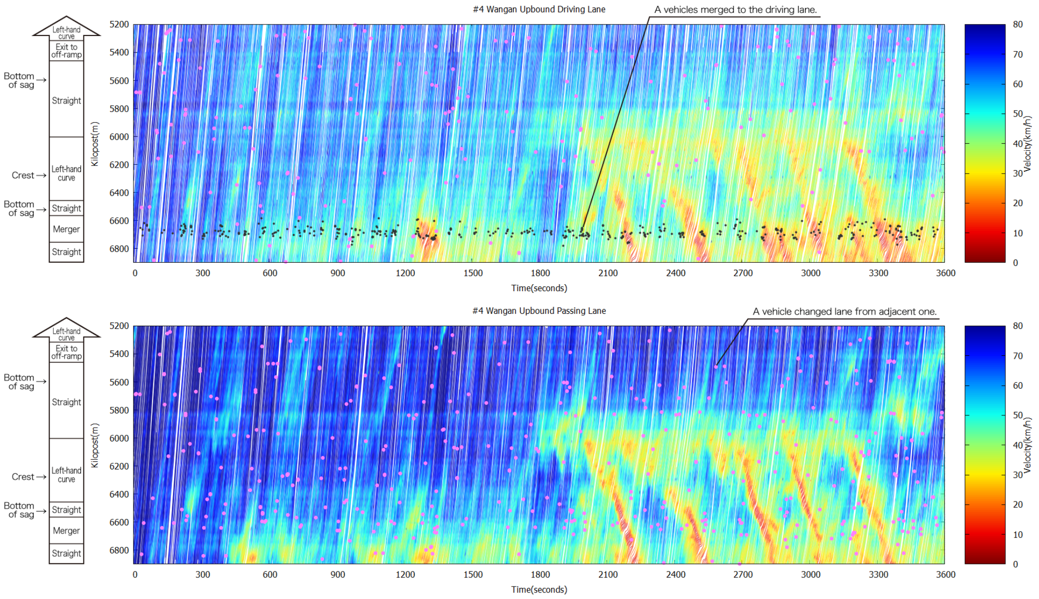


Figure B.1: Speed contour plot on Route 4 Scene 5 (Retrieved from Zen traffic data)

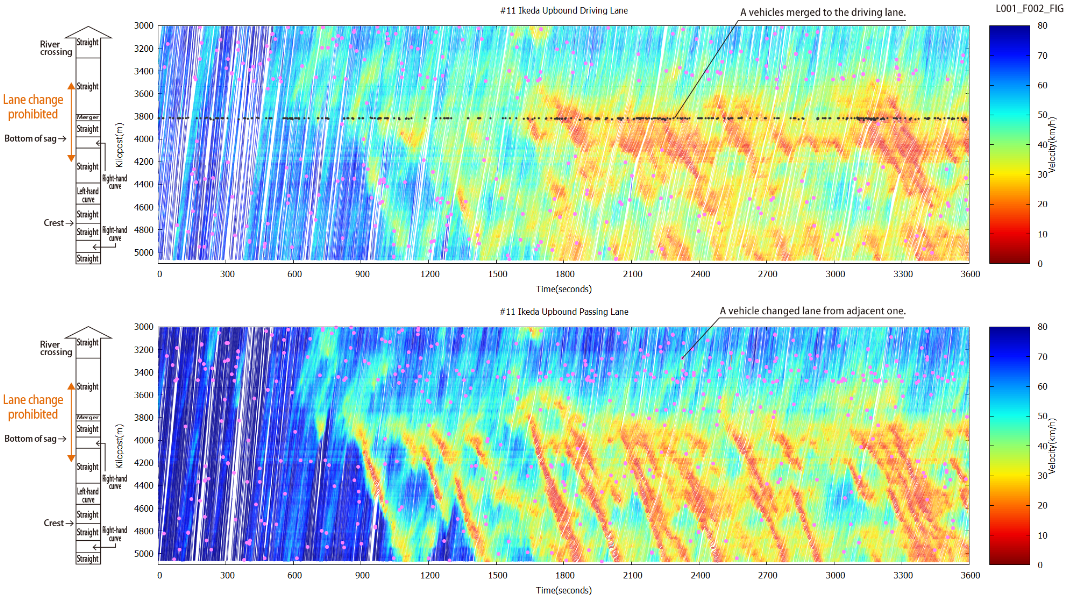


Figure B.2: Speed contour plot on Route 11 Scene 2 (Retrieved from Zen traffic data)